Υπόδειγμα εξωφύλλου του αντιτύπου ΜΔΕ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ - WEBINTELLIGENCE

Δημιουργία μοντέλων μηχανικής μάθησης για την πρόβλεψη τιμών ισοτιμίας νομισμάτων με τη χρήση forex dataset

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΕΥΣΤΡΑΤΙΟΥ ΜΠΙΛΗ

Επιβλέπων : Κωνσταντίνος Γουλιάνας
Αναπληρωτής Καθηγητής, ΔΙ.ΠΑ.Ε.

Θεσσαλονίκη, Φεβρουάριος 2022

Η σελίδα αυτή είναι σκόπιμα λευκή.

Υπόδειγμα φύλλου τίτλου (μπροστινή σελίδα) του αντιτύπου ΜΔΕ

ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ – WEB INTELLIGENCE

Δημιουργία μοντέλων μηχανικής μάθησης για την πρόβλεψη τιμών ισοτιμίας νομισμάτων με τη χρήση forex dataset

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΕΥΣΤΡΑΤΙΟΥ ΜΠΙΛΗ

Επιβλέπων : Κωνσταντίνος Γουλιάνας
Αναπληρωτής Καθηγητής, ΔΙ.ΠΑ.Ε.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή στις Choose a date.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Όνομα Επώνυμο	Όνομα Επώνυμο	'Ονομα Επώνυμο
Choose an item. ΔΙ.ΠΑ.Ε.	Choose an item. ΔΙ.ΠΑ.Ε.	Choose an item. ΔΙ.ΠΑ.Ε.

Θεσσαλονίκη, Μάρτιος 2022

Υπόδειγμα φύλλου τίτλου (πίσω σελίδα) του αντιτύπου που υποβάλλεται στις βιβλιοθήκες (διπλωματική εργασία)

(Υπογραφή)

.....

ΕΥΣΤΡΑΤΙΟΣ ΜΠΙΛΗΣ

Μηχανικός Πληροφορικής Α.Τ.Ε.Ι.Θ

Ευχαριστίες

Με την ολοκλήρωση της μεταπτυχιακής διπλωματικής μου εργασίας, θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες σε όλους όσους συνέβαλλαν στην εκπόνησή της.

Συγκεκριμένα, ευχαριστώ θερμά τους επιβλέποντες καθηγητές μου, κυρίους Κωνσταντίνο Γουλιάνα και Κωνσταντίνο Διαμαντάρα, για την εμπιστοσύνη που μου έδειξαν να αναλάβω το συγκεκριμένο θέμα της διπλωματικής, καθώς και την επιστημονική καθοδήγηση και τις υποδείξεις τους. Τέλος, θα ήθελα εκφράσω την ευγνωμοσύνη μου στην οικογένειά μου για όλη τη στήριξη, τη συμπαράσταση και την κατανόησή τους, καθ' όλη τη διάρκεια των σπουδών μου.

Περίληψη

Στην παρούσα εργασία, διερευνάτε το πρόβλημα της πρόβλεψης της μελλοντικής τιμής της ισοτιμίας Forex, EUR/USD χρησιμοποιώντας τεχνικές και μοντέλα μηχανικής μάθησης. Σκοπός της εργασίας είναι η προσπάθεια πρόβλεψης της τιμής EUR/USD σε time frame ημέρας και η εξαγωγή συμπεράσματος κατά πόσο και σε τι ποσοστό η πρόβλεψη είναι επιτυχημένη ή αποτυχημένη με χρήση μοντέλων μηχανικής μάθησης. Τα μοντέλα μηχανικής μάθησης που υλοποιήθηκαν είναι τα LSTM, SGD, BGR, XGB, Random Forest και Linear Regressor. Τα μοντέλα δημιουργήθηκαν και εκτελέστηκαν στην πλατφόρμα Kaggle, με δεδομένα κεριών ημέρας από την πλατφόρμα Metatrader5 και από το Kaggle. Όλα τα μοντέλα παρουσιάζουν παρόμοια ποσοστά επιτυχίας, με τις τιμές πρόβλεψης να είναι αρκετά κοντά τις περισσότερες φορές. Για κάθε μοντέλο που υλοποιήθηκε, υπολογίστηκαν τα errors mae (mean absolute error), mse (mean squared error) και το r^2 score. Η εφαρμογή υλοποιήθηκε με Python στο Django Framework όπου έγινε χρήση των γλωσσών προγραμματισμού Html, Css, Javascript και Bootstrap Framework για το Frontend κομμάτι. Επίσης χρησιμοποιήθηκαν πληθώρα βιβλιοθηκών Python.

Τα αποτελέσματα θεωρούνται αρκετά ικανοποιητικά δεδομένου ότι συνήθως, δεν υπάρχει πολύ μεγάλη απόκλιση της προβλεπόμενής τιμής σε σχέση με την πραγματική τιμή. Παρόλα αυτά, με βάση την παρούσα εργασία, μόνο η τιμή πρόβλεψης δεν είναι αρκετή για επαγγελματικό άνοιγμα εντολών καθώς θα πρέπει να ληφθούν και άλλες παράμετροι υπόψιν όπως και να βελτιωθεί η απόδοση των μοντέλων.

Λέξεις Κλειδιά: Machine Learning, Forex, trading

Η σελίδα αυτή είναι σκόπιμα λευκή.

Abstract

In the present paper, is examined the problem of predicting the future value of the Forex, EUR / USD exchange rate using machine learning techniques and models. The purpose of this study is to try to predict the EUR / USD price in day time frame and to draw a conclusion as to whether the forecast is successful or failed using machine learning models. The machine learning models implemented are LSTM, SGD, BGR, XGB, Random Forest and Linear Regressor. The models were created and executed on the Kaggle platform, with data from Metatrader5 platform and Kaggle. All models present similar success rates, with forecast values being quite close most of the time. For each model implemented, the errors mae (mean absolute error), mse (mean squared error) and the r2 score were calculated. The application was implemented with Python in the Django Framework where the programming languages Html, Css, Javascript and Bootstrap Framework were used for the Frontend part of the application. A variety of Python libraries were also used.

The results are considered quite satisfactory as, there is not much deviation of the predicted value from the actual value. However, the forecast value is not enough for professional opening trading orders as other parameters should be considered as well as the improvement of the performance of the models.

Keywords: Machine Learning, Forex, Trading

Η σελίδα αυτή είναι σκόπιμα λευκή.

Πίνακας περιεχομένων

1

Εισαγωγή

1.1 Αγορά Συναλλάγματος

Η αγορά συναλλάγματος είναι η μεγαλύτερη χρηματοπιστωτική αγορά στον κόσμο. Με ημερήσιο τζίρο που ξεπερνά τα 5 τρισεκατομμύρια δολάρια, αποτελεί τη ραχοκοκαλιά του διεθνούς εμπορίου [1]. Η αγορά είναι σε μεγάλο βαθμό ανεξέλεγκτη και οι συναλλαγές με συναλλαγματικές ισοτιμίες γίνονται σήμερα σχεδόν αποκλειστικά εξωχρηματιστηριακά (OTC), με την πλειοψηφία να πραγματοποιείται στα Δίκτυα Ηλεκτρονικών Επικοινωνιών (ECN). Τα ECN είναι ουσιαστικά χρηματιστήρια όπου οι συμμετέχοντες στην αγορά μπορούν να δουν ο ένας τις παραγγελίες του άλλου και να αλληλεπιδράσουν για να πουλήσουν ή να αγοράσουν νόμισμα, χρησιμοποιώντας πολλούς τύπους εντολών που κυμαίνονται από απλές οριακές εντολές έως σύνθετες συνδυασμένες στρατηγικές. Την τελευταία δεκαετία σημειώθηκαν σημαντικοί προόδοι στον τομέα της μηχανικής μάθησης και της τεχνητής νοημοσύνης. Πολλά οφείλονται στους ταχύτερους υπολογιστές, αλλά και στην πληθώρα των ηλεκτρονικών δεδομένων που έχουν δημιουργήσει μια βάση δοκιμών για πειράματα με αλγορίθμους μηχανικής μάθησης. Καθώς οι αλγόριθμοι μηχανικής μάθησης αρχίζουν να έχουν καλύτερη απόδοση από τον ανθρώπινο νου σε σύνθετες εργασίες [2], υπάρχει ένα αυξανόμενο ενδιαφέρον για τη χρήση μηχανικής μάθησης με σκοπό την μοντελοποίηση και την πρόβλεψη των κινήσεων της αγοράς, είτε για αντιστάθμιση κινδύνων, είτε για αναζήτηση ευκαιριών συναλλαγών.

Η μηχανική μάθηση μπορεί να εφαρμοστεί σε πολλούς τομείς της επιστήμης των υπολογιστών, όπως για την λήψη αποφάσεων, την πρόβλεψη, και ειδικότερα για την πρόβλεψη της τιμής των μετοχών ή της συναλλαγματικής ισοτιμίας. Με την μηχανική μάθηση επιτυγχάνεται η δημιουργία εκπαιδευμένων μοντέλων μέσω του υπολογιστή για την ταξινόμηση ή την ομαδοποίηση δεδομένων από το σύνολο δεδομένων που χρησιμοποιούνται, τα οποία καταλήγουν σε προβλέψεις. Με την άνθηση των εφαρμογών της Μηχανικής Μάθησης (ML) τα τελευταία χρόνια, έχει γίνει όλο και πιο δημοφιλής η εφαρμογή μοντέλων Machine Learning στον χρηματοοικονομικό κλάδο. Παρά τον ενθουσιασμό, η επιτυχία τους είναι αναμφισβήτητα λιγότερο ελπιδοφόρα σε σύγκριση με άλλους τομείς, ιδίως αν συγκριθεί με άλλους τομείς όπως πχ τον τομέα του εμπορίου. Στον τομέα του Forex γενικότερα, εξετάζεται η χρήση μεθόδων μηχανικής μάθησης για την πρόβλεψη της ισοτιμίας EUR/USD σε διάφορα time frames. Ειδικότερα, υλοποιούνται διάφορες προσέγγισεις πρόβλεψης των ισοτιμιών Forex που βασίζονται σε νευρωνικά δίκτυα και αλγορίθμους μηχανικής μάθησης για τη μοντελοποίηση πρόβλεψης τιμών ισοτιμίας Forex.[21][25][27][29]

Η μοντελοποίηση της πρόβλεψης τιμών Forex, αποτελεί ένα από τα θεμελιώδη προβλήματα στον χρηματοοικονομικό κόσμο. Δεν είναι μόνο ένα κρίσιμο κομμάτι για την αξιολόγηση των κινδύνων που ελλοχεύουν, αλλά επίσης ένα σημαντικό εργαλείο για το άνοιγμα εντολών Forex. Για παράδειγμα, η επιλογή ανοίγματος εντολών από τους επενδυτές (traders) συχνά κατοπτρίζουν την δικιά τους πρόβλεψη η οποία πολλές φορές θεωρείται αβάσιμη. Αλγόριθμοι υψηλής συχνότητας ανοίγματος εντολών θα πρέπει να υπολογίζουν διάφορες παραμέτρους όπως την διακύμανση τιμών ημέρας ώστε να κάνουν πρόβλεψη της τιμής ισοτιμίας σε πραγματικό χρόνο. Ένας επαγγελματίας trader, συχνά χρησιμοποιεί διάφορες παραμέτρους όπως π.χ οι διαφορές μεταξύ του μέγιστου και του ελάχιστου ενός κεριού (time frame), ως δείκτης μεταβλητότητας σε πραγματικές συναλλαγές. Η εφαρμογή της μηχανικής μάθησης στην πρόβλεψη τιμών Forex αποτελεί ένα ενεργό τομέα των οικονομικών, όπου τα νευρωνικά δίκτυα αποτελούν μία δημοφιλή κατηγορία μοντέλων.[30]

1.2 Μοντέλα μηχανικής μάθησης σε Forex data

Στην παρούσα εργασία παρουσιάζονται τα αποτελέσματα της χρήσης μοντέλων μηχανικής μάθησης με χρήση διαφόρων παραμέτρων όπως παρουσιάζονται παρακάτω στην εργασία. Οι προσεγγίσεις που υλοποιήθηκαν αποτελούν μία ενδεικτική λύση πρόβλεψης τιμών Forex συναλλάγματος EUR/USD οι οποίες θα μπορούσαν να βελτιωθούν περαιτέρω .

Σε αυτή τη διατριβή, χρησιμοποιούνται αλγόριθμοι μηχανικής μάθησης σε δεδομένα ημέρας μέσα από μία εφαρμογή που δημιουργήθηκε στα πλαίσια της παρούσας διπλωματικής, σε μια

προσπάθεια εντοπισμού προτύπων και πρόβλεψης των κινήσεων της αγοράς συναλλάγματος EUR/USD.

1.2.1 Συνεισφορά

Η συνεισφορά της διπλωματικής συνοψίζεται ως εξής:

1. Μελετήσαμε μοντέλα και μεθόδους μηχανικής μάθησης όπως τα LSTM, SGD, BGR, XGB, Random Forest, Linear Regressor.
2. Υλοποιήσαμε 6 αλγορίθμους μηχανικής μάθησης υπολογισμού πρόβλεψης της τιμής ισοτιμίας EUR/USD με δεδομένα ημέρας.
3. Αξιολογήσαμε την επίδοση των αλγορίθμων και βρήκαμε ότι όλα τα παραπάνω μοντέλα που υλοποιήθηκαν παρουσιάζουν παρόμοιου error αποτελέσματα.
4. Υλοποιήσαμε και ελέγξαμε όλους τους παραπάνω αλγορίθμους στην πλατφόρμα Kaggle.
5. Ενσωματώσαμε όλους του παραπάνω αλγορίθμους σε προ-εκπαιδευμένα μοντέλα τα οποία ενσωματώσαμε σε μία custom web app για την πρόβλεψη τιμών ισοτιμίας EUR/USD σε δεδομένα ημέρας.

1.3 Οργάνωση κειμένου

Εργασίες σχετικές με το αντικείμενο της διπλωματικής παρουσιάζονται στο Κεφάλαιο 2 . Το Κεφάλαιο 3 αναφέρεται στο θεωρητικό υπόβαθρο της μηχανικής μάθησης. Στο Κεφάλαιο 4 εξηγούνται τα μοντέλα μηχανικής μάθησης τα οποία περιγράφονται αναλυτικά όλα τα βήματα της δημιουργίας τους στο κεφάλαιο 5. Στο κεφάλαιο 6 αναφέρονται οι μετρικές αξιολόγησης των μοντέλων. Στην συνέχεια, στο κεφάλαιο 7, γίνεται αναφορά στις τεχνικές λεπτομέρειες καθώς και τις πλατφόρμες και τα προγραμματιστικά εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής που συνοδεύει την παρούσα εργασία. Στο προτελευταίο κεφάλαιο το 8, αναφέρονται τα συμπεράσματα και οι μελλοντικές εργασίες. Τέλος, στο κεφάλαιο 9 υπάρχει η βιβλιογραφία στην οποία βασίστηκε η συγκεκριμένη εργασία.

2

Σχετικές εργασίες

Με βάση την βιβλιογραφία ,αρκετοί συγγραφείς έχουν διερευνήσει τη χρήση μεθοδολογιών βαθιάς μάθησης και ενισχυτικής μάθησης (Reinforcement Learning) προκειμένου να δομηθούν αποτελεσματικά συστήματα συναλλαγών ισοτιμίας. Ακολουθούν ορισμένες επιστημονικές συνεισφορές που δείχνουν τα πλεονεκτήματα που μπορούν να προκύψουν από τη χρήση εποπτευόμενης βαθιάς μάθησης και αλγορίθμων που βασίζονται σε RL.

2.1 Deep Learning (LSTM) και RL Based Trading systems

Υπάρχουν αρκετές εργασίες στον χώρο του Forex με την χρήση μεθόδων-μοντέλων μηχανικής μάθησης με στόχο την πρόβλεψη τιμής ισοτιμιών. Συγκεκριμένα στο [3], οι συγγραφείς πρότειναν μια ενδιαφέρουσα εμπορική προσέγγιση που βασίζεται στη χρήση της βαθιάς ενισχυτικής μάθησης. Οι συγγραφείς πρότειναν έναν πράκτορα συναλλαγών που βασίζεται στη βαθιά μάθηση ενίσχυσης, για να λαμβάνει αυτόνομα αποφάσεις συναλλαγών μέσω μιας τροποποιημένης προσέγγισης ενός βαθύ Q-Network (DQN) και Actor- critic (A3C). Χρησιμοποίησαν ένα βαθύ Framework που βασίζεται στη χρήση ενός stacked denoising autoencoder (SDAEs) και LSTM) προκειμένου να σχεδιάσουν ισχυρούς μηχανισμούς που θα κάνουν τον πράκτορα συναλλαγών πιο πρακτικό στο πραγματικό περιβάλλον συναλλαγών. Τα αποτελέσματα επιβεβαίωσαν την αποτελεσματικότητα της προτεινόμενης προσέγγισης [3].

Στο [4], προτείνεται μια μέθοδος ημερησίας διαπραγμάτευσης πολλαπλών εντολών. Η βασική ιδέα της προτεινόμενης προσέγγισης είναι η χρήση μεθοδολογίας μάθησης βαθιάς ενίσχυσης πολλαπλών στόχων για την ημερησία αναπαράσταση σημάτων και ανοίγματος εντολών. Οι συγγραφείς στο [4] εφάρμοσαν ένα βαθύ νευρωνικό δίκτυο για να εξαγάγουν βαθιά χαρακτηριστικά της αγοράς που ακολουθείται από ένα πλαίσιο ενισχυτικής μάθησης (με ad-hoc LSTMs) ικανό να λαμβάνει συνεχείς αποφάσεις συναλλαγών. Προκειμένου να επιτευχθεί

μια καλή αντιστάθμιση μεταξύ κέρδους και κινδύνου, οι συγγραφείς πρότειναν μια προσέγγιση βελτιστοποίησης πολλαπλών στόχων που περιλαμβάνει δύο αντικειμενικές συναρτήσεις (μία για το κέρδος και μία για τον κίνδυνο) με διαφορετικά βάρη. Τα πειραματικά αποτελέσματα επιβεβαίωσαν ότι η προσέγγιση που αναφέρεται στο [4] είναι αποτελεσματική.

Στο [5], οι Chen et al. πρότειναν μια καινοτόμο μέθοδο που βασίζεται στην έννοια της «εμπορίας ενέργειας». Μέσω ενός ad-hoc μαθηματικού μοντέλου στρατηγικών εμπορίας ενέργειας ενός αγοραστή, στο προτεινόμενο ολιστικό μοντέλο αγοράς, η διαδικασία λήψης αποφάσεων του αγοραστή θα αναλυθεί ως διαδικασία απόφασης Markov, έτσι ώστε η συμμετοχή στην αγορά εντολών, να επιλυθεί με τεχνολογία εκμάθησης βαθιάς ενίσχυσης. Αυτή η προσέγγιση μπορεί εύκολα να επεκταθεί στις χρηματοπιστωτικές αγορές με ειδική αναφορά στις μετοχές εταιρειών ή στον τομέα της διαχείρισης ενέργειας. Ένας από τους πιο μελετημένους δείκτες στην ποσοτική χρηματοοικονομική είναι σίγουρα η χρηματοοικονομική αστάθεια. Υπάρχουν διάφοροι τρόποι για να υπολογιστεί ευρετικά η αστάθεια ενός συγκεκριμένου χρηματοοικονομικού μέσου. Πολλά συστήματα συναλλαγών βασίζονται στην εκτίμηση του συντελεστή μεταβλητότητας του χρηματοοικονομικού μέσου, έτσι ώστε να είναι πολύ σημαντικό να υπάρχει μια ισχυρή και αποτελεσματική μέθοδος για την πρόβλεψη της μεταβλητότητας.

Στο [6], οι συγγραφείς πρότειναν έναν αγωγό για την πρόβλεψη αστάθειας ενός ζεύγους νομισμάτων (INRUSD). Μέσω των αρχιτεκτονικών LSTM, η μεταβλητότητα του ζεύγους νομισμάτων INR/USD εκτιμήθηκε με επιτυχία. Η έρευνα που αναφέρθηκε στο [6] πρότεινε μια καινοτόμο προσέγγιση για την πρόβλεψη της ανοδικής ή καθοδικής κίνησης της καθημερινής μεταβλητότητας. Οι συγγραφείς συνέκριναν τον αλγόριθμο που βασίζεται στο LSTM με τα κλασικά νευρωνικά δίκτυα παλινδρόμησης, το SVM, το Random Forest, τους αλγόριθμους παλινδρόμησης, τα δέντρα αποφάσεων και τεχνικές ενίσχυσης. Η προσέγγιση που βασίζεται σε LSTMs επιβεβαιώθηκε ότι έχει την καλύτερη απόδοση.

Στο [7], οι συγγραφείς ανέλυσαν διάφορες στρατηγικές συναλλαγών που βασίζονται σε προσεγγίσεις που βασίζονται σε βαθιά μάθηση που εφαρμόζονται στο εμπόριο του σύνθετου δείκτη της Σαγκάης. Το αποτέλεσμα της έρευνας στο [7] επιβεβαίωσε ότι η καλύτερη στρατηγική συναλλαγών που βασίζεται στη χρήση του βαθιού νευρωνικού δικτύου είναι αυτή που δείχνει υψηλή ακρίβεια πρόβλεψης στην αγορά χαμηλής μεταβλητότητας, καθώς μπορεί να βοηθήσει τους επενδυτές να μειώσουν τον κίνδυνο ενώ επιτυγχάνουν ικανοποιητικές αποδόσεις. Στο [8], οι Chen και al. πρότειναν μια πολύ ενδιαφέρουσα ιδέα, την κλωνοποίηση της προηγούμενης στρατηγικής συναλλαγών που ήταν αποθηκευμένη στα οικονομικά αρχεία για να δημιουργηθεί ένα κερδοφόρο σύστημα συναλλαγών. Ούτως ή άλλως, λόγω των μεγάλων ποσοτήτων οικονομικών δεδομένων που εξάγονται, οι λογικές αποφάσεις και τα βασικά χαρακτηριστικά των στρατηγικών διαπραγμάτευσης ερμηνευτών είναι ιδιαίτερα δύσκολα. Για

αυτούς τους λόγους, οι συγγραφείς πρότειναν στο [8] να χρησιμοποιηθεί ένα σύστημα ενισχυτικής μάθησης (Reinforcement Learning) για μίμηση στρατηγικών συναλλαγών επαγγελματιών traders. Οι συγγραφείς σχεδίασαν το περιβάλλον RL (καταστάσεις, ενέργειες και ανταμοιβές) προκειμένου να εφαρμόσουν μια ad-hoc μέθοδο gradient πολιτικής ικανή να μιμηθεί τις στρατηγικές συναλλαγών ενός ειδικού. Τα πειραματικά αποτελέσματα δείχνουν ότι η προτεινόμενη χρήση ενισχυτικής μάθησης RL είναι σε θέση να εντοπίσει περίπου το 80% των σωστών αποφάσεων συναλλαγών, τόσο σε περιόδους εκπαίδευσης, όσο και σε περιόδους επικύρωσης.

2.2 Deep Computing Based Trading Systems

Στο [9], οι συγγραφείς πρότειναν μια ενδιαφέρουσα προσέγγιση που ονομάστηκε βασισμένη στο σύστημα A-trader (σύστημα πολλαπλών πρακτόρων για διαπραγμάτευση μετοχών). Οι συγγραφείς ανέλυσαν την εφαρμογή προσεγγίσεων βαθιάς μάθησης στο framework A-trader για τη δημιουργία κερδοφόρων στρατηγικών συναλλαγών στην αγορά συναλλάγματος. Ο αναλυόμενος αλγόριθμος βαθιάς μάθησης H20 φαίνεται να έχει μεγαλύτερη απόδοση σύμφωνα με τα πειραματικά αποτελέσματα που αναφέρονται στο [9]. Στο [10], οι συγγραφείς ανέλυσαν και συνέκριναν τέτοιες προσεγγίσεις μηχανικής μάθησης με νέες που προτάθηκαν από τους συγγραφείς και βασίστηκαν στη χρήση του 1D συνελικτικού νευρωνικού δικτύου (CNN). Τα προτεινόμενα μονοδιάστατα συνελικτικά επίπεδα επεξεργάζονται διαφορετικά χρηματοοικονομικά δεδομένα, όπως τιμές, όγκο συναλλαγών, εξάγοντας χαρακτηριστικά για να χρησιμοποιηθούν για τη δημιουργία των στρατηγικών συναλλαγών. Οι συγγραφείς αξιολόγησαν την απόδοση της μεθόδου τους με back-testing σε ιστορικά δεδομένα από τον Ιανουάριο του 2010 έως τον Οκτώβριο του 2017. Τα αποτελέσματα φαίνονται πολύ ελπιδοφόρα [10]. Για πολλά χρόνια, αρκετοί συγγραφείς έχουν μελετήσει τη χαοτική συμπεριφορά τέτοιων φαινομένων, συμπεριλαμβανομένης της δυναμικής της χρηματοπιστωτικής αγοράς [11].

Ο Lee στο [11] πρότεινε ένα Χαοτικό Τύπο-2 Μεταβατικό-Ασαφές Βαθύ Νευροταλαντευτικό Δίκτυο (Transient-Fuzzy Deep Neuro-oscillatory Network (CT2TFDNN)), για την παγκόσμια πρόβλεψη οικονομικών δεδομένων, συμπεριλαμβανομένων των κύριων κρυπτονομισμάτων, του συναλλάγματος, των μεγάλων εμπορευμάτων και αρκετών χρηματοοικονομικών δεικτών. Ο συγγραφέας εισήγαγε τη νέα έννοια των χαοτικών νευρικών ταλαντωτών που χρησιμεύουν ως «παροδικοί-ασαφείς νευρώνες εισόδου» του χρησιμοποιούμενου βαθιού νευρωνικού δικτύου. Αυτό το προτεινόμενο μοντέλο χρησιμοποιήθηκε από τον συγγραφέα για να προβλέψει την τάση της κίνησης των αγορών. Τα πειραματικά αποτελέσματα επιβεβαίωσαν ότι η προτεινόμενη προσέγγιση παρουσιάζει καλά αποτελέσματα.

Στο [12], οι συγγραφείς ανέλυσαν τη χρήση της μεθόδου «trailing» για τη δημιουργία αποτελεσματικών στρατηγικών συναλλαγών. Πιο αναλυτικά, οι συγγραφείς στο [12] πρότειναν μια νέα μέθοδο τελικής τιμής, θεωρώντας το πρόβλημα συναλλαγών ως πρόβλημα ελέγχου της τιμής. Η προτεινόμενη μέθοδος εφάρμοσε ισχυρούς παράγοντες που μπορούν να αντέξουν μεγάλες ποσότητες θορύβου χρονοσειρών, προσδιορίζοντας τις τάσεις των τιμών προκειμένου να εκτελούν κερδοφόρες εντολές. Η καμπύλη Profit and Loss (P&L) που αναφέρεται στο [12] επιβεβαίωσε ότι η προτεινόμενη προσέγγιση έχει πολύ καλή απόδοση στη χρηματοπιστωτική αγορά. Ούτως ή άλλως, σύμφωνα με τις περισσότερες από τις στρατηγικές συναλλαγών που προτείνονται στη βιβλιογραφία, η προσέγγιση που αναφέρεται στο [12] και τα περισσότερα από τα παραπάνω που περιγράφηκαν, αν και αποδίδουν πολύ καλά, δεν έχουν ακριβή ανάλυση τόσο της μέγιστης όσο και της δυναμικής ανεβάσματος – κατεβάσματος της τιμής [1,2], η οποία είναι απαραίτητη για την ορθή αξιολόγηση της προτεινόμενης στρατηγικής συναλλαγών, καθώς επιτρέπει την ποσοτικοποίηση του κινδύνου έκθεσης του επενδυτή.

3

Μηχανική μάθηση (Machine Learning)

Η μηχανική μάθηση (Machine Learning) είναι υποπεδίο της επιστήμης των υπολογιστών που αναπτύχθηκε από τη μελέτη της αναγνώρισης προτύπων και της υπολογιστικής θεωρίας μάθησης στην τεχνητή νοημοσύνη. Στην μηχανική μάθηση διερευνάται η μελέτη και η κατασκευή αλγορίθμων που μπορούν να μαθαίνουν από ένα σύνολο δεδομένων και να κάνουν προβλέψεις που σχετίζονται με τα δεδομένα. Τέτοιου είδους αλγόριθμοι λειτουργούν κατασκευάζοντας μοντέλα από πειραματικά δεδομένα, με σκοπό να κάνουν προβλέψεις βασιζόμενες στα δεδομένα ή να εξάγουν αποφάσεις που εκφράζονται ως το αποτέλεσμα.

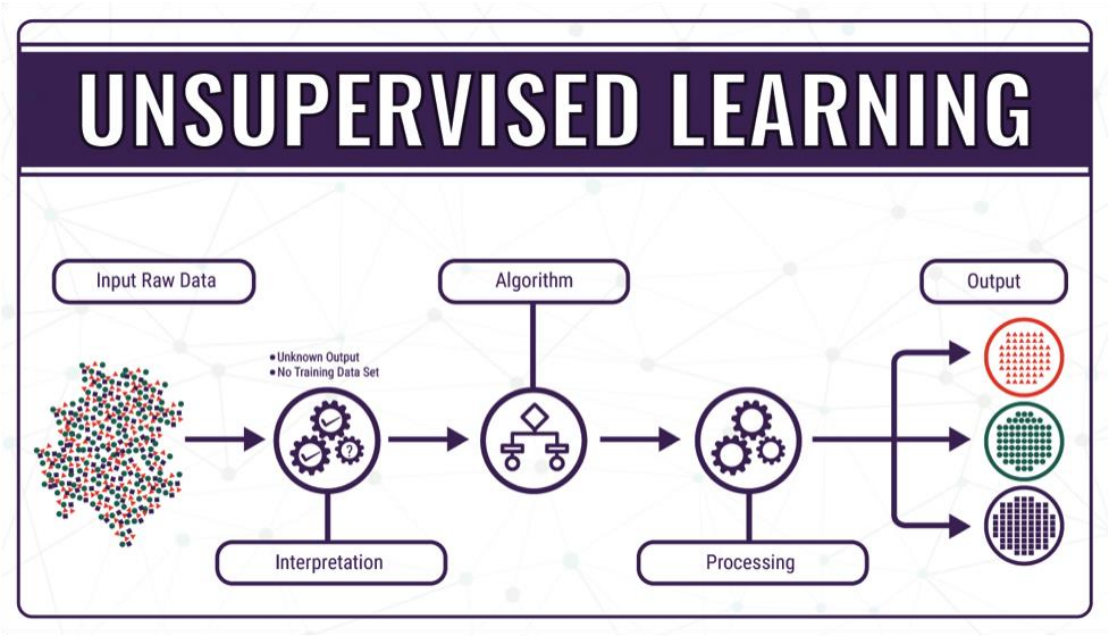
Η μηχανική μάθηση εφαρμόζεται με δεδομένα αριθμούς, φωτογραφίες, κείμενο, τραπεζικές συναλλαγές, εικόνες ανθρώπων ,δεδομένα χρονοσειρών, αναφορές πωλήσεων. Τα δεδομένα συλλέγονται και προετοιμάζονται για χρήση ως δεδομένα εκπαίδευσης από τα οποία θα εκπαιδευτεί το μοντέλο μηχανικής μάθησης. Συνήθως όσο περισσότερα δεδομένα έχουμε, τόσο καλύτερο είναι για το μοντέλο που θα εκπαιδευτεί.

Εφαρμόζεται σε μια σειρά από υπολογιστικές εργασίες, όπου ο σχεδιασμός όπως και ο προγραμματισμός των αλγορίθμων είναι ανέφικτος. Παραδείγματα εφαρμογών αποτελούν τα φίλτρα spam (spam filtering), η οπτική αναγνώριση χαρακτήρων (OCR), οι μηχανές αναζήτησης και η υπολογιστική όραση.

Η λειτουργία ενός συστήματος μηχανικής μάθησης μπορεί να είναι περιγραφική, που σημαίνει ότι το σύστημα χρησιμοποιεί τα δεδομένα για να εξηγήσει τι συνέβη. προγνωστικό, που σημαίνει ότι το σύστημα χρησιμοποιεί τα δεδομένα για να προβλέψει τι θα συμβεί, ή προδιαγραφική, που σημαίνει ότι το σύστημα θα χρησιμοποιήσει τα δεδομένα για να κάνει προτάσεις σχετικά με τη δράση που πρέπει να ληφθεί.[13][14][15]

3.1 Μάθηση χωρίς επίβλεψη (Unsupervised Learning)

Η μάθηση χωρίς επίβλεψη είναι ένας τύπος αλγόριθμων μηχανικής μάθησης που χρησιμοποιείται για την εξαγωγή συμπερασμάτων από τα σύνολα δεδομένων που αποτελούνται από δεδομένα εισόδου χωρίς ετικέτα. Πρόκειται στην ουσία για έναν τύπο μηχανικής εκμάθησης που χρησιμοποιείται για την εύρεση κρυφών μοτίβων χωρίς καμία εξωτερική είσοδο. Χρησιμοποιούνται μόνο τα ακατέργαστα δεδομένα για τον εντοπισμό δομών και σχέσεων μεταξύ των δεδομένων.



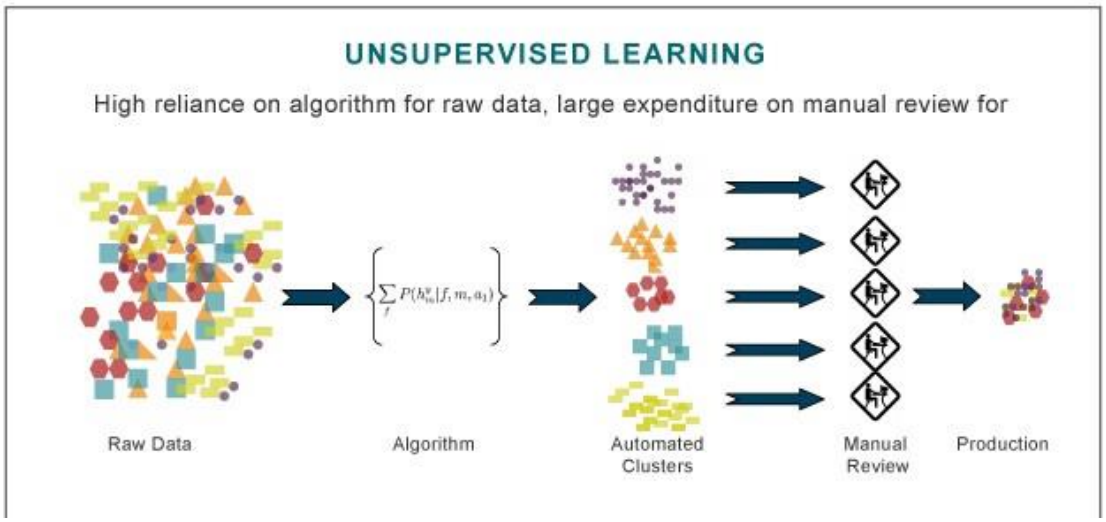
Εικόνα 1. Πηγή εικόνας:bigdata-madesimple.com

Μερικές φορές, μπορεί να μην είναι πρακτικό να παρέχονται πληροφορίες σχετικά με τους τύπους δεδομένων που θα λάβει ένα σύστημα υπολογιστή μηχανικής μάθησης. Επομένως, η εποπτευόμενη μάθηση μπορεί να μην είναι κατάλληλη στις περιπτώσεις όπου τα συστήματα υπολογιστών χρειάζονται τακτικές πληροφορίες σχετικά με νέους τύπους δεδομένων. Για παράδειγμα, οι χάκερ που εκτελούν επιθέσεις hacking σε χρηματοοικονομικά συστήματα ή σε διακομιστές τραπεζών έχουν την τάση να αλλάζουν τακτικά τα πρότυπα και τη φύση τους. Η μάθηση χωρίς επίβλεψη μπορεί να είναι πιο κατάλληλη για τέτοια σενάρια, επειδή τα συστήματα πρέπει να μάθουν γρήγορα από τις επιθέσεις και θα πρέπει να συμπεριλάβουν την πιθανότητα επερχόμενης επίθεσης.[16]

Μέθοδοι μάθησης χωρίς επίβλεψη χρησιμοποιούνται στη βιοπληροφορική για ανάλυση αλληλουχίας και γενετική ομαδοποίηση, στην εξόρυξη δεδομένων για εξόρυξη ακολουθιών και προτύπων, στην ιατρική για τμηματοποίηση εικόνων, για την αναγνώριση αντικειμένων από εικόνα μέσω του υπολογιστή. Με τη βοήθεια της μάθησης χωρίς επίβλεψη μπορούν να

επιτευχθούν δεδομένα πιο οργανωμένα και ευανάγνωστα. Ένα παράδειγμα μάθησης χωρίς επίβλεψη είναι η δυνατότητα αναγνώρισης εικόνων του Facebook.

Κάθε φορά που ο χρήστης ανεβάζει οποιαδήποτε φωτογραφία στο Facebook, το Facebook προτείνει το όνομα του ατόμου που είναι παρόν μαζί του στην εικόνα, έτσι ώστε το άτομο που βρίσκεται στην εικόνα να μπορεί να επισημανθεί. Επίσης προτείνεται το όνομα του ατόμου βγάζοντας τη φωτογραφία του συγκεκριμένου ατόμου στην φωτογραφία.



Εικόνα 2. Πηγή εικόνας: mandegar.info

Η ανάλυση συστάδων (clustering) είναι η πιο κοινή μέθοδος στη μάθηση χωρίς επίβλεψη. Είναι η διαδικασία όπου παρόμοιες οντότητες ομαδοποιούνται. Ο στόχος αυτής της τεχνικής μηχανικής εκμάθησης χωρίς επίβλεψη είναι να βρει ομοιότητες στο σημείο δεδομένων και να ομαδοποιήσει παρόμοια σημεία δεδομένων μαζί.

Χρησιμοποιείται για την ανάλυση διερευνητικών δεδομένων και για την εύρεση των κρυμμένων μοτίβων μέσα στα δεδομένα. Μοντελοποιούνται αυτές οι συστάδες χρησιμοποιώντας ένα μέτρο ομοιότητας (μετρήσεις όπως η Ευκλείδεια ή η πιθανολογική απόσταση).[14][15]

Οι κοινοί αλγόριθμοι ομαδοποίησης είναι οι εξής:

- Ιεραρχική ομαδοποίηση (Hierarchical clustering): Σε αυτήν, δημιουργείται μια πολυεπίπεδη ιεραρχία συστάδων δημιουργώντας ένα δέντρο συστάδων.
- Ομαδοποίηση K-Means (K-Means clustering): Τα δεδομένα διαιρούνται σε k διακριτές συστάδες με βάση την απόσταση από το κέντρο μιας συστάδας. Χρησιμοποιείται επίσης για την πρόβλεψη υποομάδων από ένα δεδομένο σύνολο δεδομένων.
- Μοντέλο Gaussian μείγματος (Gaussian mixture model): Σε αυτό, οι συστάδες μοντελοποιούνται ως μείγμα συστατικών πολυμεταβλητής κανονικής πυκνότητας.
- Αυτο-οργάνωση χαρτών (Self-organizing maps): χρησιμοποιεί νευρωνικά δίκτυα που μαθαίνουν την τοπολογία και τη διανομή των δεδομένων

- Κρυφά μοντέλα Markov (Hidden Markov models): Σε αυτό, τα παρατηρούμενα δεδομένα χρησιμοποιούνται για την ανάκτηση της ακολουθίας των καταστάσεων.

Όταν ομαδοποιούνται παρόμοιες οντότητες από κοινού, τότε σκιαγραφούμε τα χαρακτηριστικά διαφορετικών ομάδων. Με άλλα λόγια, αυτό μας δίνει τη γνώση σχετικά με τα υποκείμενα πρότυπα διαφορετικών ομάδων. Η ομαδοποίηση δεδομένων χωρίς ετικέτα έχει πολλές εφαρμογές, όπως, μπορούμε εύκολα να αναγνωρίσουμε τις διάφορες ομάδες ή τμήματα πελατών προκειμένου να αυξήσουμε τα κέρδη. Ένα ακόμη παράδειγμα αφορά την ομαδοποίηση εγγράφων που ανήκουν στο ίδιο θέμα.

Η ομαδοποίηση χρησιμοποιείται επίσης για τη μείωση της διάστασης των δεδομένων όταν έχουμε να κάνουμε με πολύ μεγάλο αριθμό μεταβλητών.

Υπάρχουν πολλοί αλγόριθμοι που έχουν αναπτυχθεί στο κομμάτι της ομαδοποίησης και συγκεκριμένα θα γίνει αναφορά στους:

1. K-means Clustering
2. Ιεραρχική Ομαδοποίηση

3.1.1 Ομαδοποίηση K-means

Η πρόβλεψη στην ομαδοποίηση K-means εξαρτάται από τον αριθμό των κέντρων συμπλέγματος που υπάρχουν και από την πλησιέστερη μέση τιμή (Ευκλείδεια απόσταση) μεταξύ των παρατηρήσεων. Χρησιμοποιούμε K-means στην τμηματοποίηση πελατών, τον εντοπισμό ασφαλιστικών απατών, τη μοντελοποίηση κόστους.[19] Η λειτουργία του K-means έχει ως εξής:

1. Ξεκινά με το K ως είσοδο. Το K είναι βασικά ένας αριθμός που δείχνει πόσα συμπλέγματα θέλουμε να σχηματίσουμε. Αν K=3 τότε θα υπάρχουν τρεις συστάδες και αν K=4, τότε θα υπάρχουν τέσσερις συστάδες.
2. Τώρα, με τη βοήθεια της Ευκλείδειας απόστασης (χρησιμοποιείται για να βρούμε ποιο κέντρο είναι πιο κοντά σε κάθε σημείο δεδομένων) μεταξύ σημείων δεδομένων και κεντροειδών (κεντρικό σημείο του δεδομένου συνόλου δεδομένων), θα αντιστοιχίσουμε κάθε σημείο δεδομένων στο πλησιέστερο σύμπλεγμα.
3. Στη συνέχεια υπολογίζουμε ξανά το κέντρο συστάδων.
4. Θα επαναλάβουμε τα βήματα 2 και 3 μέχρι να μην υπάρξουν επιπλέον αλλαγές.

Πλεονεκτήματα:

- Είναι αρκετά κατανοητό και είναι στιβαρό.
- Όταν τα σύνολα δεδομένων είναι διαφορετικά τότε αυτός ο αλγόριθμος δίνει το καλύτερο αποτέλεσμα.

3.1.2 Ιεραρχική Ομαδοποίηση

Χρησιμοποιούμε αυτή τη μέθοδο μάθησης χωρίς επίβλεψη για την πρόβλεψη υποομάδων εντός δεδομένων. Αυτό το κάνουμε βρίσκοντας την απόσταση μεταξύ κάθε σημείου δεδομένων και του πλησιέστερου γείτονά τους. Στη συνέχεια, κάθε σημείο δεδομένων συνδέεται με τον γείτονά του. Σε αντίθεση με την ομαδοποίηση K-means στην ιεραρχική ομαδοποίηση ξεκινάμε αναθέτοντας κάθε δεδομένο να δείχνει στο δικό του σύμπλεγμα (cluster). Στο επόμενο βήμα, τα δύο πλησιέστερα σημεία δεδομένων συγχωνεύονται σε ένα σύμπλεγμα (cluster).

1. Αρχικά, εκχωρούμε σε κάθε σημείο δεδομένων το δικό του σύμπλεγμα.
2. Στη συνέχεια, χρησιμοποιώντας την Ευκλείδεια απόσταση βρίσκουμε το πλησιέστερο ζεύγος του συμπλέγματος και στη συνέχεια το συγχωνεύουμε σε ενιαίο σύμπλεγμα.
3. Η απόσταση μεταξύ δύο πλησιέστερων συστάδων υπολογίζεται και ενώνεται έως ότου όλα τα σημεία συγκεντρωθούν σε ένα μόνο σύμπλεγμα.

Σε αυτήν την τεχνική, αποφασίζουμε τον καλύτερο δυνατό αριθμό συστάδων παρατηρώντας ποια οριζόντια γραμμή μπορεί να κόψει τις κάθετες γραμμές χωρίς να τέμνει μία συστάδα (cluster) και να καλύπτει τη μέγιστη απόσταση. Μπορούμε να βρούμε τις υποομάδες με τη βοήθεια του δενδρογράμματος.

Δενδρογράφημα: Το δεντρόγραμμα είναι ένα δενδρόγραμμα (χρησιμοποιείται βασικά στη βιολογία για την εμφάνιση της συστάδας μεταξύ γονιδίων ή δειγμάτων) που χρησιμοποιείται για την απεικόνιση της ιεραρχικής συστάδας - της σχέσης μεταξύ παρόμοιου συνόλου δεδομένων. Μπορεί να είναι γράφημα γραμμής ή στήλης.

3.2 Μάθηση με επίβλεψη (Supervised Learning)

Η εποπτευόμενη μάθηση είναι μια δημοφιλής έννοια της μηχανικής μάθησης που εφαρμόζεται σε πραγματικές περιστάσεις. Όπως υποδηλώνει το όνομα, πρέπει να επιβλέπουμε το μηχάνημά μας ενώ μαθαίνει ή εκπαιδεύεται να δουλεύει μόνο του. Για αυτό, απαιτούμε σύνολα δεδομένων (με ετικέτα δεδομένα εκπαίδευσης) για να κάνουμε προβλέψεις. Αυτά τα σύνολα δεδομένων αποτελούνται από τιμές εισόδου και εξόδου. Οι προβλέψεις γίνονται με βάση αυτά τα σύνολα δεδομένων. Κατασκευάζουμε ένα μοντέλο που κάνει προβλέψεις με βάση προηγούμενα δεδομένα για νέα σύνολα δεδομένων. Ας υποθέσουμε ότι παρέχουμε εικόνες ζώων και τις προσδιορίζουμε. Όταν το μοντέλο μας μάθει να κάνει ακριβείς προβλέψεις, τότε παρέχουμε νέες εικόνες ζώων. Αυτή τη φορά το μοντέλο, θα κάνει πρόβλεψη για νέες εικόνες με βάση την προηγούμενη εκπαίδευση.

Πρόκειται στην ουσία για μια μέθοδο διευκόλυνσης των μηχανών να ώστε να ταξινομούν αντικείμενα, καταστάσεις ή προβλήματα με βάση τα δεδομένα που τροφοδοτούνται στις μοντέλα. Σε αυτό, τροφοδοτούμε τα μοντέλα με μοτίβα, χρώμα, διαστάσεις αντικειμένων,

ανθρώπων ή καταστάσεων ή άλλες πληροφορίες ως δεδομένα έως ότου τα μοντέλα, εκτελέσουν τις ταξινομήσεις με ακρίβεια. Χρησιμοποιούμε εποπτευόμενη μάθηση όταν έχουμε μια καθορισμένη τιμή στόχο την οποία θέλουμε να προβλέψουμε.[14][16][26][32]

Η εποπτευόμενη μάθηση έχει δύο τύπους:

1. Ταξινόμηση (Classification)
2. Παλινδρόμηση (Regression)

3.2.1 Ταξινόμηση

Στην ταξινόμηση αναζητούμε βασικά μια έξοδο που έχει τη μορφή είτε π.χ «ναι» ή «όχι» ή «μαύρο» ή «λευκό». Όταν μιλάμε για ένα ταξινομημένο μοντέλο τότε προσπαθούμε να βγάλουμε κάποιο συμπέρασμα από την παρατήρηση. Βοηθά στον προσδιορισμό σε ποια κατηγορία ανήκει ένα συγκεκριμένο πράγμα. Στην ταξινόμηση, οι τιμές στόχου έχουν τη μορφή επισημασμένων και κατηγορικών δεδομένων. Η μέθοδος φιλτραρίσματος ανεπιθύμητων μηνυμάτων του Gmail εμπίπτει σε αυτήν την κατηγορία. Με τη βοήθεια της μεθόδου φιλτραρίσματος ανεπιθύμητων μηνυμάτων, το Gmail ταξινομεί εύκολα τα μηνύματα ως ανεπιθύμητα ή επιθυμητά. Οι πελάτες ηλεκτρονικού ταχυδρομείου χρησιμοποιούν φίλτρα ανεπιθύμητης αλληλογραφίας για να κρατούν τους χρήστες μακριά από την ανεπιθύμητη αλληλογραφία. Αυτά τα φίλτρα ανεπιθύμητης αλληλογραφίας ενημερώνονται συχνά.

Ομοίως, μπορούμε να ταξινομήσουμε τα αντικείμενα σε διαφορετικές κατηγορίες με βάση τα χαρακτηριστικά ή τις ιδιότητές τους.

3.2.2 Λογιστική παλινδρόμηση (Logistic Regression)

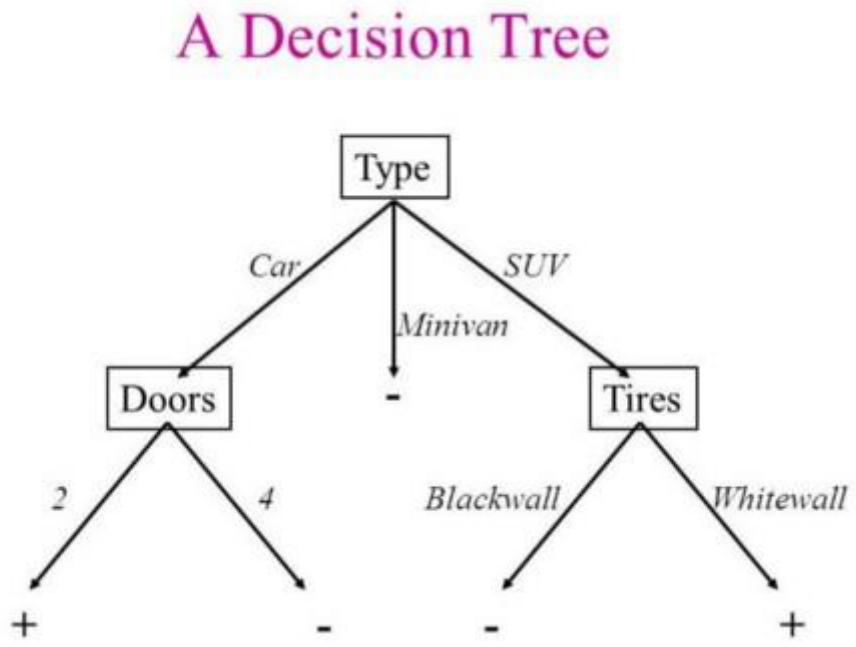
Χρησιμοποιείται για ταξινόμηση. Χρησιμοποιώντας λογιστική παλινδρόμηση μπορούμε να προβλέψουμε διακριτές τιμές. Είναι μια απλή προσέγγιση μηχανικής μάθησης που χρησιμοποιείται για την πρόβλεψη της τιμής μιας αριθμητικής κατηγορικής μεταβλητής με βάση τη σχέση της με τις μεταβλητές πρόβλεψης. Υπάρχουν λίγοι τύποι λογιστικής παλινδρόμησης:

1. Binary Logistic Regression: Έχει μόνο δύο πιθανά αποτελέσματα. Για παράδειγμα: Ανεπιθύμητα ή Μη Ανεπιθύμητα emails, Ναι ή Όχι
2. Ordinal Logistic Regression: Έχει αποτέλεσμα με τη μορφή παραγγελίας. Παράδειγμα: Βαθμολογίες από 1 έως 10.
3. Multinomial Logistic Regression: Έχει τρεις ή περισσότερες κατηγορίες. Για παράδειγμα: Ποιο χρώμα θα προτιμηθεί περισσότερο - μαύρο, μπλε, μωβ ή καφέ.

Για την πρόβλεψη, σε ποια κατηγορία ανήκει ένα συγκεκριμένο δεδομένο, ορίζουμε ένα όριο με βάση το οποίο γίνεται η ταξινόμηση. Για παράδειγμα: εάν η προβλεπόμενη τιμή > 2, τότε επισημάνετε την αλληλογραφία ως ανεπιθύμητη, διαφορετικά ως μη ανεπιθύμητη.

3.2.3 Δέντρο αποφάσεων

Είναι μια απλή δομή της οποίας οι μη τερματικοί κόμβοι αντιπροσωπεύουν δοκιμή σε ένα ή περισσότερα χαρακτηριστικά και οι τερματικοί κόμβοι αντικατοπτρίζουν τα αποτελέσματα αποφάσεων. Δημιουργείται επιλέγοντας ένα υποσύνολο στιγμιότυπων από ένα σετ εκπαίδευσης. Οι υπόλοιπες περιπτώσεις ελέγχουν την ακρίβεια του κατασκευασμένου δέντρου. Εάν το δέντρο απόφασης ταξινομήσει σωστά τα στιγμιότυπα, η διαδικασία τερματίζεται. Εάν τα στιγμιότυπα έχουν ταξινομηθεί εσφαλμένα, τα στιγμιότυπα προστίθενται στο επιλεγμένο υποσύνολο των στιγμιοτύπων εκπαίδευσης και δημιουργείται ένα νέο δέντρο. Αυτή η διαδικασία συνεχίζεται μέχρι να δημιουργηθεί ένα δέντρο το οποίο θα ταξινομεί σωστά όλα τα μη επιλεγμένα στιγμιότυπα ή θα δημιουργηθεί ένα δέντρο αποφάσεων από το ολόκληρο σύνολο εκπαίδευσης.[18]

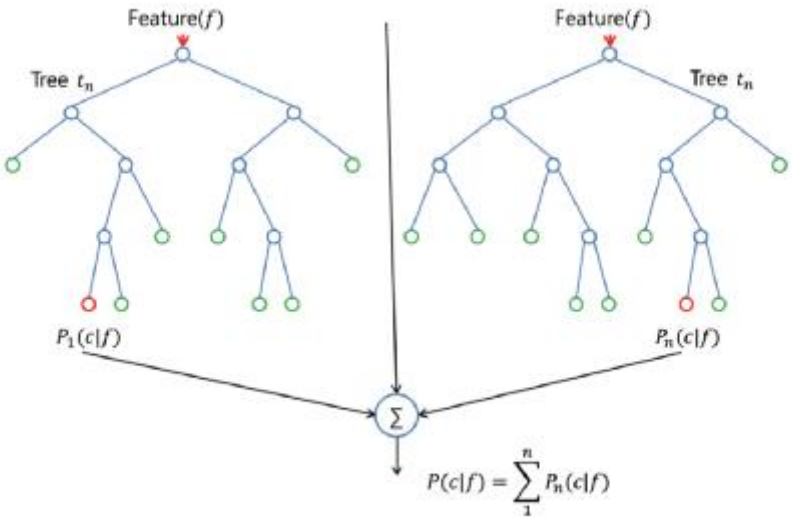


Εικόνα 3. Πηγή εικόνας techleer.com

3.2.4 Naïve Bayes

Ο αλγόριθμος Naïve Bayes βασίζεται στο θεώρημα του Bayes και μπορεί να είναι πολύ χρήσιμος για μεγάλα σύνολα δεδομένων. Προβλέπει πιθανότητες σχέσης για κάθε κλάση, όπως την πιθανότητα μια δεδομένη εγγραφή ή δεδομένα να ταιριάζουν σε μια συγκεκριμένη κλάση ή όχι. Η κλάση που έχει την υψηλότερη πιθανότητα είναι γνωστό ότι είναι η πιο πιθανή κατηγορία. Η υπόθεση στον ταξινομητή Naïve Bayes βασίζεται στην υπόθεση ότι όλα τα χαρακτηριστικά δεν σχετίζονται μεταξύ τους και η παρουσία ή η απουσία οποιουδήποτε χαρακτηριστικού δεν θα επηρεάσει την απουσία ή την παρουσία άλλου χαρακτηριστικού.

3.2.5 Τυχαίο δάσος (Random Forest)



Εικόνα 4. Πηγή εικόνας assignment.com

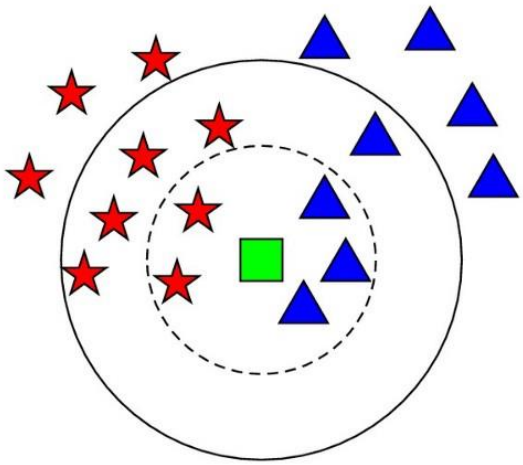
Όπως προκύπτει από το όνομα, δημιουργείται ένα δάσος με έναν αριθμό δέντρων. Μπορούμε να ονομάσουμε ένα δάσος καλό δάσος μόνο αν υπάρχει τεράστιος αριθμός δέντρων. Το ίδιο συμβαίνει και στο τυχαίο δάσος. Όσο μεγαλύτερος αριθμός δέντρων σε τυχαία δάση, τόσο πιο ακριβές αποτέλεσμα θα δημιουργηθεί από αυτή την προσέγγιση. Χρησιμοποιώντας ένα τυχαίο δάσος μπορούμε επίσης να χειριστούμε τις τιμές που λείπουν.

Ας υποθέσουμε ότι υπάρχει μια κυρία που ονομάζεται Ρία που θέλει να αρχίσει να διαβάζει μυθιστορήματα. Έτσι, θα ζητήσει από τη φίλη της που διαβάζει μυθιστορήματα την γνώμη της. Η φίλη της θα ρωτήσει τη Ρία για το είδος της ή τι της αρέσει και με βάση αυτό θα της προτείνει κάποια ονόματα μυθιστορημάτων και θα της πει ποια ήταν τα καλύτερα από αυτά που διάβασε. Τώρα η Ρία θα ξανασυμβουλευτεί λίγους φίλους της και οι φίλοι της θα της κάνουν κι αυτοί περισσότερες ερωτήσεις για τα γούστα της. Με βάση το ενδιαφέρον της Ρίας θα προτείνουν τα και αυτοί κάποια μυθιστορήματά. Τώρα η Ρία θα επιλέξει το μυθιστόρημα που προτάθηκε περισσότερο. Με την παραπάνω λογική του παραδείγματος, λειτουργεί ο αλγόριθμος Random Forest.[19][20]

3.2.6 Κ Πλησιέστερος Γείτονας (K Nearest Neighbor)

Οι προβλέψεις που έγιναν βασίζονται σε ποιο βαθμό οι εκπαιδευτικές παρατηρήσεις είναι παρόμοιες με τις νέες παρατηρήσεις. Σε αυτό το σημείο, βάζουμε ένα νέο σημείο δεδομένων στην πιο πιθανή γειτονική ομάδα του. Το K στο KNN είναι η ακέραια τιμή μεγαλύτερη από 1.

Κάθε φορά που έχουμε ένα νέο σημείο δεδομένων που θέλουμε να ταξινομήσουμε, θα υπολογίζουμε για να βρούμε σε ποια γειτονική ομάδα είναι πιο κοντά.



Εικόνα 5. KNN

Στην εικόνα μπορούμε να δούμε ότι υπάρχουν τρίγωνα ένα τετράγωνο και αστέρια. Το πράσινο τετράγωνο πρέπει να τοποθετηθεί σε μία από τις δύο ομάδες. Οπότε δημιουργούμε κύκλο γύρω από το τετράγωνο. Τα τρίγωνα πλησιέστερα στο τετράγωνο, είναι περισσότερα σε αριθμό από τα αστέρια, οπότε το τετράγωνο θα τοποθετηθεί στην ομάδα τριγώνων.

Το KNN έχει κάποιες υποθέσεις όπως:

- Τα σύνολα δεδομένων έχουν μικρό θόρυβο.
- Τα σύνολα δεδομένων έχουν σχετικά χαρακτηριστικά και φέρουν ετικέτα.

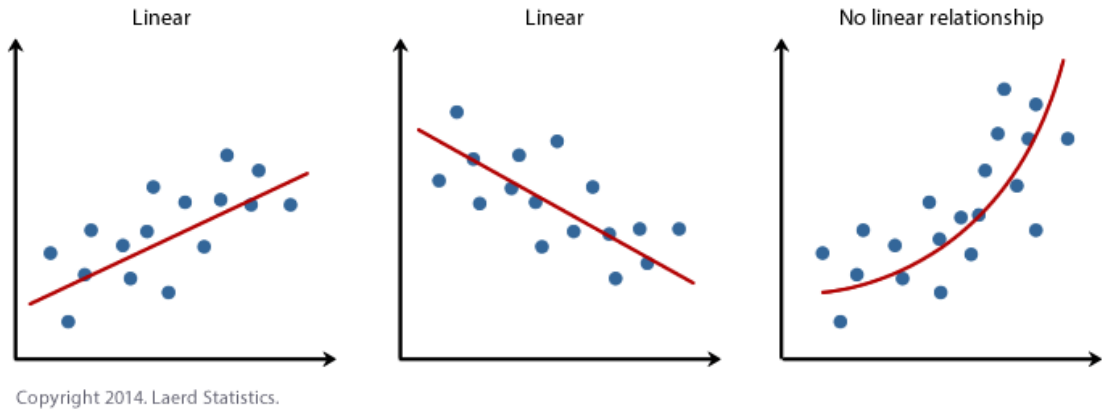
Ο αλγόριθμος KNN χρησιμοποιείται σε βάσεις δεδομένων όπου τα σημεία δεδομένων πρέπει να διαχωριστούν σε διάφορες κατηγορίες για την πρόβλεψη της ταξινόμησης ενός νέου σημείου δεδομένων. Μπορεί να χρησιμοποιηθεί ο αλγόριθμος KNN στο σύστημα πρόβλεψης τιμών μετοχών.

3.2.7 Παλινδρόμηση (Regression)

Στην παλινδρόμηση, έχουμε τη μεταβλητή στόχο ως αριθμητικά δεδομένα. Χρησιμοποιούνται δεδομένα του παρελθόντος για να γίνει η πρόβλεψη αντί να γίνει ταξινόμηση. Στην προσέγγιση ταξινόμησης είχαμε την τιμή εξόδου καθώς και την τιμή εισόδου στο σύνολο δεδομένων εκπαίδευσης. Αλλά στην παλινδρόμηση δεν έχουμε τιμή εξόδου στο σύνολο δεδομένων εκπαίδευσης. Για παράδειγμα: Ας υποθέσουμε ότι μετακομίζουμε σε μια νέα πόλη και θέλουμε να μάθουμε ποια είναι η τιμή των διαμερισμάτων στη συγκεκριμένη πόλη. Για αυτό τον σκοπό, παίρνουμε τιμές που δείχνουν τις τιμές των διαμερισμάτων.

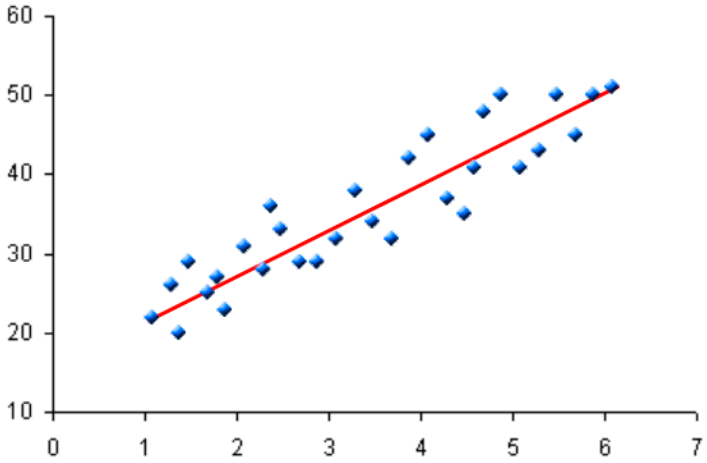
Σε αυτή την προσέγγιση έχουμε μια ανεξάρτητη μεταβλητή x, με βάση το x υπολογίζουμε την τιμή μιας εξαρτημένης μεταβλητής y.

3.2.7.1 Γραμμική παλινδρόμηση



Εικόνα 6. Πηγή εικόνας: statistics.laerd.com

Είναι μια προσέγγιση γραμμικής μοντελοποίησης για την εύρεση σχέσης μεταξύ μιας ή περισσότερων ανεξάρτητων μεταβλητών (προγνωστικών) που συμβολίζονται ως x και μιας εξαρτημένης μεταβλητής (στόχου) που συμβολίζεται ως y. Ονομάζεται γραμμική γιατί η εξίσωση δεν έχει μη γραμμική συνιστώσα. Μπορεί να ονομαστεί ως μέθοδος στατιστικής μηχανικής μάθησης που χρησιμοποιούμε για να ποσοτικοποιηθεί η πρόβλεψη με βάση τη σχέση μεταξύ αριθμητικών μεταβλητών. Έχει να κάνει με την εύρεση της καλύτερης γραμμής προσαρμογής με αναδρομικό τρόπο.

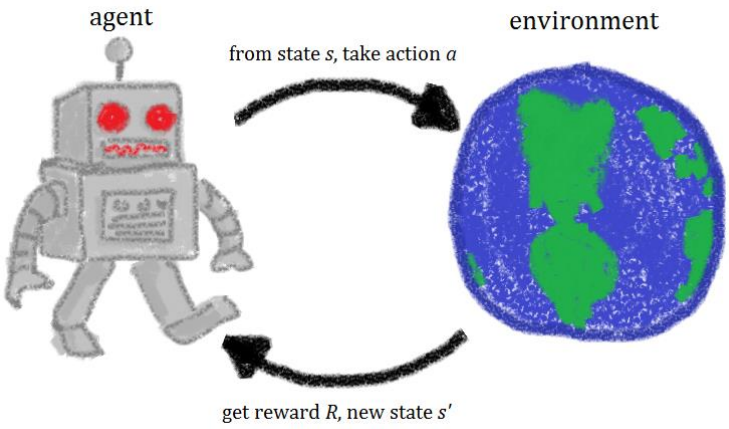


Εικόνα 7. Πηγή εικόνας- nchsbands.info

Όπως φαίνεται από την παραπάνω εικόνα, υπάρχουν πολλά σημεία στο γράφημα από όπου θα μπορούσε να τραβηχτεί η γραμμή. Εάν μπορεί να σχεδιαστεί μια γραμμή που έχει ελάχιστη απόσταση από όλα τα σημεία, τότε η γραμμή είναι γνωστό ότι είναι η καλύτερη προσαρμοσμένη γραμμή παλινδρόμησης.

3.3 Ενισχυτική μάθηση (Reinforcement learning)

Η ενισχυτική μάθηση είναι μια προσέγγιση της μηχανικής μάθησης που χρησιμοποιείται για το σκοπό της κατευθυνόμενης μάθησης και της λήψης αποφάσεων. Είναι εμπνευσμένη από τη συμπεριφορική ψυχολογία. Σε αυτήν την προσέγγιση, η μηχανή μαθαίνει από την άμεση αλληλεπίδραση με το περιβάλλον της χωρίς να εξαρτάται από κάποιο προκαθορισμένο σύνολο δεδομένων με ετικέτα. Ο στόχος πίσω από την Ενισχυτική Μάθηση είναι - ένας πράκτορας λογισμικού ή μία μηχανή που θα μπορούσε να μάθει από το περιβάλλον, αλληλεπιδρώντας με αυτό και λαμβάνοντας ανταμοιβές για την εκτέλεση ενεργειών. Σε αυτό, το μηχάνημα ή ο πράκτορας λογισμικού καθορίζει συνήθως την τελική συμπεριφορά μέσα σε ένα συγκεκριμένο πλαίσιο για τη μεγιστοποίηση της απόδοσης. Αυτός ο πράκτορας μπορεί να είναι ένα αυτο-οδηγούμενο αυτοκίνητο ή μια εφαρμογή που παίζει σκάκι. Όπως αναφέρθηκε παραπάνω, ο (πράκτορας) αλληλεπιδρά με το περιβάλλον του, λαμβάνει μια ανταμοιβή με βάση τον τρόπο με τον οποίο ενεργεί. Για παράδειγμα η οδήγηση με ασφάλεια στον προορισμό ή νίκη σε ένα παιχνίδι. Όταν εκτελείται λανθασμένα, όπως φεύγοντας από το δρόμο ή ματ για το σκάκι, ο πράκτορας λαμβάνει ποινή. Ο πράκτορας λαμβάνει αποφάσεις με τρόπο για να αξιοποιήσει στο έπακρο την ανταμοιβή του και να μειώσει τις ποινές μέσω δυναμικού προγραμματισμού. Αυτή η προσέγγιση έχει πλεονέκτημα στην τεχνητή νοημοσύνη. Κάθε πρόγραμμα τεχνητής νοημοσύνης μπορεί να μάθει χωρίς τη βοήθεια προγραμματιστή που καθοδηγεί τον πράκτορα σχετικά με τις ενέργειες που πρέπει να γίνουν. [31]



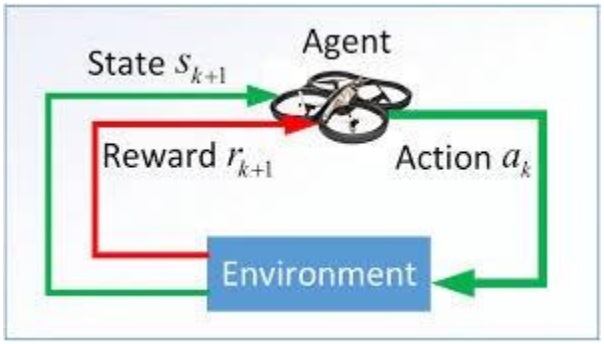
Εικόνα 8. Πηγή εικόνας: datanami.com

Η ενισχυτική μάθηση χρειάζεται για τους εξής λόγους:

- Λιγότερη ανθρώπινη προσπάθεια: Δεδομένου ότι, ο μαθητής (μηχανή) δεν ενημερώνεται για τις ενέργειες που πρέπει να γίνουν, αλλά αντίθετα, πρέπει να καθορίσει μόνος του ποια ενέργεια αποφέρει μεγάλες ανταμοιβές, επιχειρώντας τις

ενέργειες. Επομένως, υπάρχει μικρή ανάγκη για έναν άνθρωπο ειδικό να ενημερώνει την μηχανή.

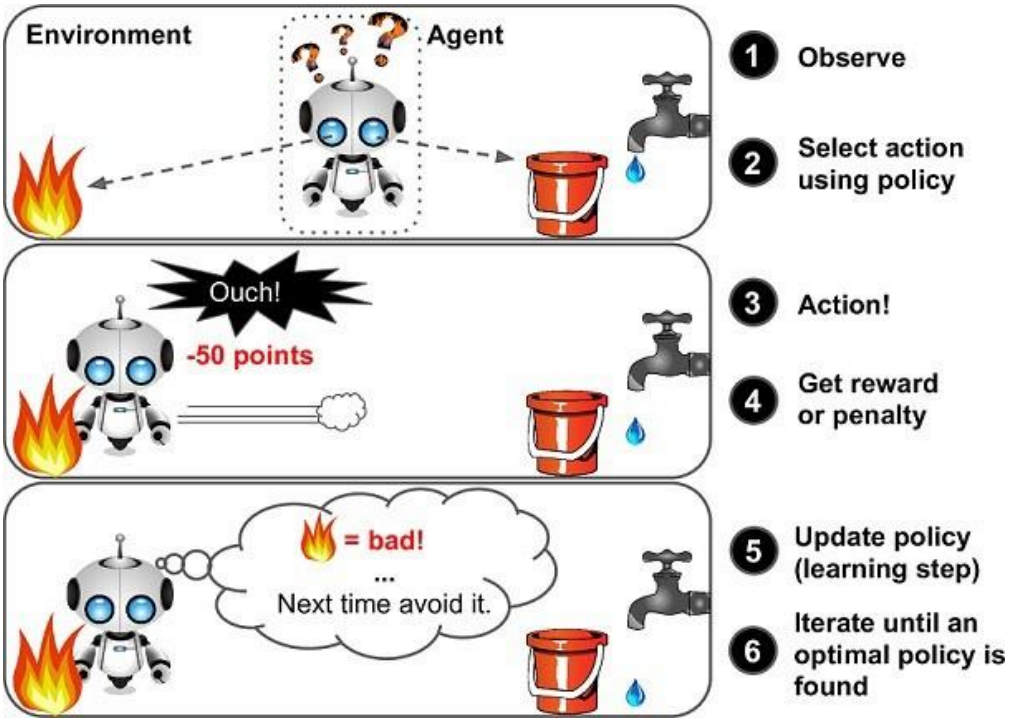
- Δεν χρειάζονται περίπλοκους κανόνες: Θα χρησιμοποιηθεί μικρότερος χρόνος για το σχεδιασμό μιας λύσης, επειδή δεν χρειάζεται να δημιουργηθούν περίπλοκοι κανόνες όπως κάνουμε με τα Expert Systems.
- Μέσω του Reinforcement Learning, το μηχάνημα ή ο πράκτορας λογισμικού είναι σε θέση να μαθαίνει αλληλεπιδρώντας και με βάση την αντίδραση από την κατάσταση. Αυτή η γνώση θα διατηρηθεί για πάντα και μπορεί επίσης να συνεχίσει να προσαρμόζεται σε βάθος χρόνου.
- Βοηθά στην αύξηση της αποτελεσματικότητας ενός εργαλείου ή προγράμματος.



Εικόνα 9. Πηγή εικόνας: researchgate.net

Υπάρχουν μερικά προκλητικά και ενδιαφέροντα περιστατικά, όπου ορισμένες ενέργειες επηρεάζουν όχι μόνο την άμεση ανταμοιβή αλλά και την επερχόμενη κατάσταση και μέσω αυτού όλες τις επόμενες ανταμοιβές. Αυτά τα δύο χαρακτηριστικά: η αναζήτηση δοκιμής και λάθους και η καθυστερημένη ανταμοιβή είναι τα διακριτικά χαρακτηριστικά της Ενισχυτικής Μάθησης.

Όπως αναφέρθηκε παραπάνω, στην ενισχυτική μάθηση, ο πράκτορας λογισμικού πρέπει να επιλέξει μια ενέργεια που θα μεγιστοποιήσει την ανταμοιβή μακροπρόθεσμα. Στην πράξη, αυτό γίνεται μαθαίνοντας να εκτιμάς την αξία μίας κατάστασης. Αυτή η εκτίμηση ρυθμίζεται με τη διάδοση της ανταμοιβής της επόμενης κατάστασης. Όταν όλες οι καταστάσεις και όλες οι ενέργειες δοκιμάζονται για αρκετό χρόνο, τότε ορίζεται μια βέλτιστη πολιτική και επιλέγεται η ενέργεια που μεγιστοποιεί την τιμή της επόμενης κατάστασης.



Εικόνα 10. Πηγή εικόνας: marutitech.com

Η ενισχυτική μάθηση είναι βασικά η αλληλεπίδραση μεταξύ δύο συστατικών - του περιβάλλοντος και του παράγοντα (αυτός που μαθαίνει). Ο μαθησιακός παράγοντας έχει δύο μηχανισμούς:

1. Εξερεύνηση: Όταν ο εκπαιδευτικός πράκτορας ενεργεί βάσει δοκιμής και σφάλματος, τότε είναι γνωστό ως εξερεύνηση.
2. Εκμετάλλευση: Όταν ενεργεί σύμφωνα με τις γνώσεις που έχει αποκομίσει από το περιβάλλον, τότε ονομάζεται εκμετάλλευση.

Το παιχνίδι σκάκι είναι ένας από τους καλύτερους τρόπους κατανόησης της ενισχυτικής μάθησης. Στο παιχνίδι του σκακιού, η μηχανή παίρνει μια απόφαση κάνοντας μια κίνηση. Στη συνέχεια έρχεται να μάθει αν η κίνηση ήταν κατάλληλη ή όχι. Και μετά σε αυτή τη βάση δίνεται η ανταμοιβή. Εάν η κίνηση είναι καλή, ας υποθέσουμε ότι ανταμείβονται +5 πόντοι. Και αν η κίνηση δεν είναι καλή, ας υποθέσουμε ότι αφαιρούνται σαν ποινή -5 πόντοι. Τώρα, το μηχάνημα θα μάθει ποια κίνηση είναι καλή και θα κάνει κινήσεις ανάλογα. Έτσι λειτουργεί η ενισχυτική μάθηση.

Το μαθηματικό πλαίσιο για τον ορισμό μιας λύσης στο σενάριο ενισχυτικής μάθησης ονομάζεται Διαδικασία Απόφασης Markov. Αυτό έχει:

- Σύνολο πολιτειών, S
- Σύνολο ενεργειών, A
- Συνάρτηση ανταμοιβής, R
- Πολιτική, π

- Αξία, V

Θα κάνουμε μια ενέργεια (A) για τη μετάβαση από την κατάσταση έναρξης στην κατάσταση λήξης (S). Σε αντάλλαγμα παίρνουμε ανταμοιβές (R) για κάθε ενέργεια. Αυτές οι ενέργειες μπορούν να οδηγήσουν σε θετική ή αρνητική ανταμοιβή. Το σύνολο των ενεργειών που κάνουμε καθορίζει την πολιτική μας (π) και οι ανταμοιβές που κερδίζουμε καθορίζουν την αξία μας (V). Πρέπει να μεγιστοποιήσουμε τις ανταμοιβές έτσι ώστε:

$$E(r_t \mid \pi, s_t)$$

για όλες τις πιθανές τιμές του S για χρόνο t.

4

Επεξήγηση Μοντέλων μηχανικής μάθησης

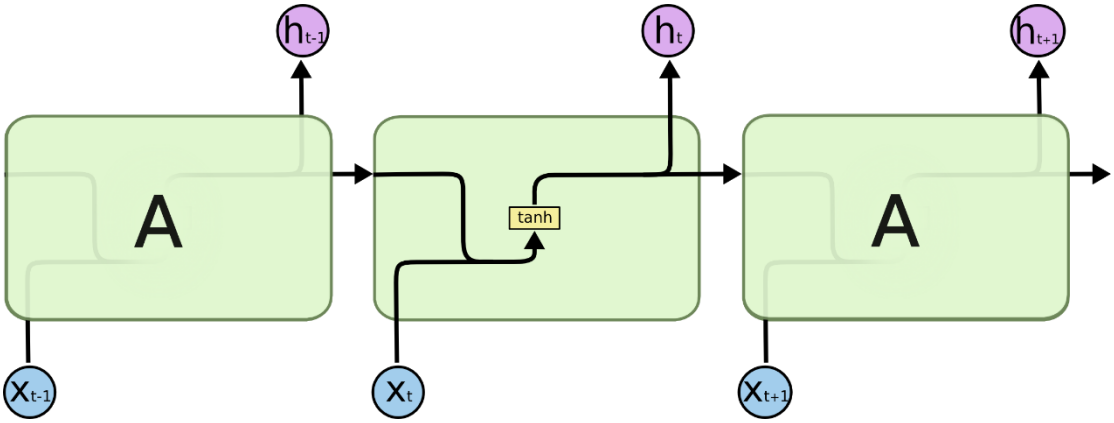
Σε αυτό το κεφάλαιο γίνεται αναφορά στην θεωρία των μοντέλων που υλοποιήθηκαν για την χρήση τους στην εφαρμογή πρόβλεψης τιμών ισοτιμίας του συναλλάγματος EUR/USD. Συγκεκριμένα υλοποιήθηκε μοντέλο νευρωνικού δικτύου LSTM και μοντέλα που βασίζονται στους αλγορίθμους Bagging Regressor, Linear Regression, SGD Regressor, Random Forest Regressor, Extreme Gradient Boosting.

4.1 Δίκτυο LSTM

Τα δίκτυα LSTM ή αλλιώς δίκτυα μακράς βραχυπρόθεσμης μνήμης (Long Short-Term Memory) αποτελούν ένα είδος RNN (Recurrent Neural Network) ικανό να μάθει μακροπρόθεσμες εξαρτήσεις. Εισήχθησαν από τους Hochreiter & Schmidhuber (1997) και βελτιώθηκαν και διαδόθηκαν από πολλούς ανθρώπους. Λειτουργούν εξαιρετικά καλά σε μια μεγάλη ποικιλία προβλημάτων και χρησιμοποιούνται πλέον ευρέως.

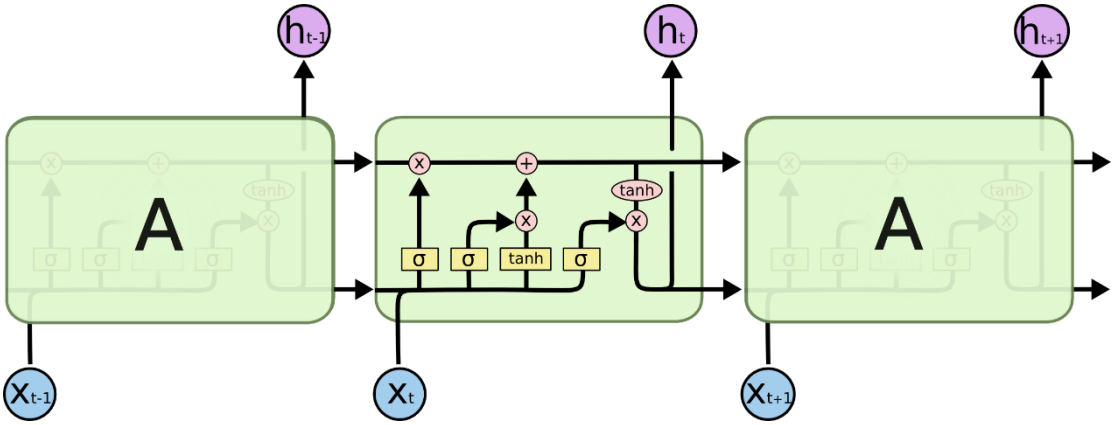
Τα LSTM έχουν σχεδιαστεί για την αποφυγή του προβλήματος της μακροπρόθεσμης εξάρτησης. Το να θυμούνται πληροφορίες για μεγάλες χρονικές περιόδους είναι πρακτικά η προεπιλεγμένη συμπεριφορά τους, όχι κάτι που δυσκολεύονται να μάθουν.[23]

Όλα τα επαναλαμβανόμενα νευρωνικά δίκτυα έχουν τη μορφή μιας αλυσίδας επαναλαμβανόμενων μονάδων νευρωνικού δικτύου. Στα τυπικά RNN (εικόνα 11), αυτή η επαναλαμβανόμενη ενότητα θα έχει μια πολύ απλή δομή, όπως ένα στρώμα tanh.[25]

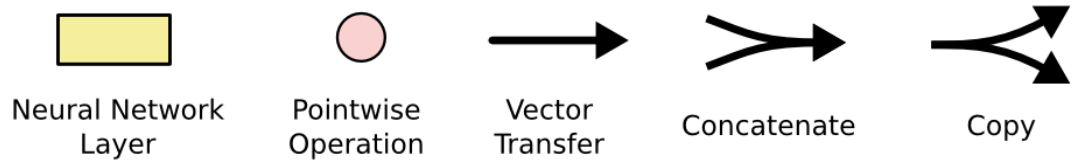


Εικόνα 11. Ένα τυπικό RNN δίκτυο.

Τα LSTM έχουν επίσης αυτή τη δομή της αλυσίδας, αλλά η επαναλαμβανόμενη μονάδα έχει διαφορετική δομή. Αντί να έχουμε ένα ενιαίο επίπεδο νευρωνικού δικτύου, υπάρχουν τέσσερα επίπεδα που αλληλεπιδρούν με έναν πολύ ιδιαίτερο τρόπο.



Εικόνα 12. Η επαναλαμβανόμενη μονάδα σε ένα LSTM περιέχει τέσσερα αλληλεπιδρώντα επίπεδα.

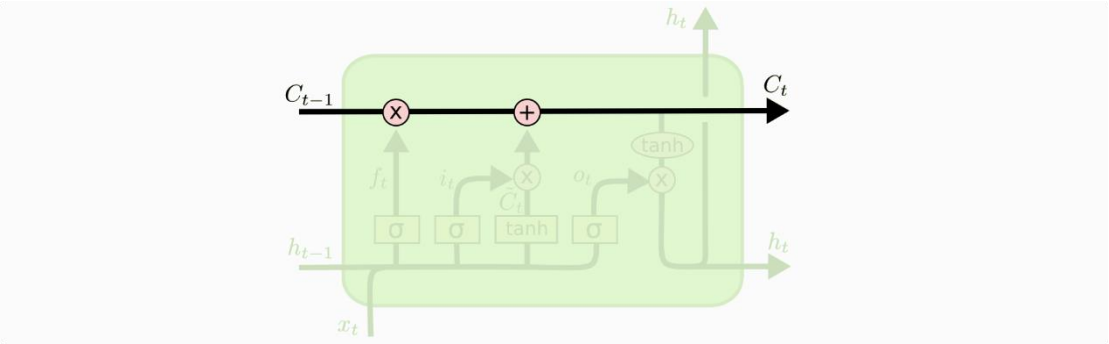


Διάγραμμα 1.

Στο παραπάνω διάγραμμα, κάθε γραμμή φέρει ένα ολόκληρο διάνυσμα, από την έξοδο ενός κόμβου έως τις εισόδους άλλων. Οι ροζ κύκλοι αντιπροσωπεύουν σημειακές πράξεις, όπως η πρόσθεση διανυσμάτων, ενώ τα κίτρινα πλαίσια είναι μαθημένα στρώματα νευρωνικού δικτύου. Η συγχώνευση γραμμών υποδηλώνει συνένωση, ενώ μια διχάλα γραμμών υποδηλώνει ότι το περιεχόμενό της αντιγράφεται και τα αντίγραφα πηγαίνουν σε διαφορετικές τοποθεσίες.

Το κλειδί για τα LSTM δίκτυα, είναι η κατάσταση του κελιού, η οριζόντια γραμμή που διατρέχει την κορυφή του διαγράμματος.

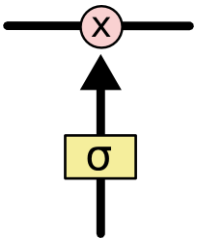
Η κατάσταση της κυψέλης μοιάζει με μεταφορικό ιμάντα. Διατρέχει κατευθείαν ολόκληρη την αλυσίδα, με μερικές μικρές γραμμικές αλληλεπιδράσεις. Είναι πολύ εύκολο για τις πληροφορίες να ρέουν κατά μήκος τους αμετάβλητες.



Εικόνα 13.

Το LSTM δίκτυο, έχει την ικανότητα να αφαιρεί ή να προσθέτει πληροφορίες στην κατάσταση της κυψέλης, που ρυθμίζεται προσεκτικά από δομές που ονομάζονται πύλες.

Οι πύλες είναι ένας τρόπος για να περάσουν προαιρετικά πληροφορίες. Αποτελούνται από ένα στρώμα σιγμοειδούς νευρικού δικτύου και μια πράξη πολλαπλασιασμού κατά σημείο.



Εικόνα 14.

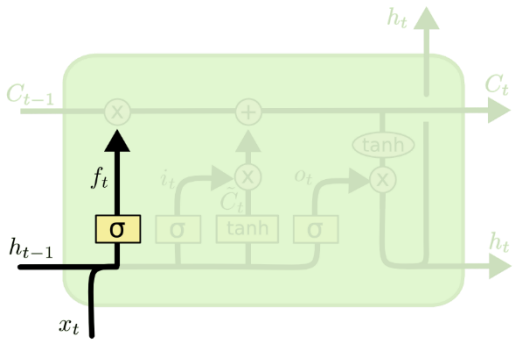
Το σιγμοειδές στρώμα εξάγει αριθμούς μεταξύ μηδέν και του ενός, περιγράφοντας πόσο από κάθε στοιχείο πρέπει να περάσει. Η τιμή του μηδέν σημαίνει "μην αφήσετε τίποτα να περάσει", ενώ η τιμή ένα σημαίνει "αφήστε τα πάντα να περάσουν".

Ένα LSTM δίκτυο έχει τρεις πύλες, για την προστασία και τον έλεγχο της κατάστασης των κυττάρων.

Το πρώτο βήμα στο LSTM δίκτυο, είναι να αποφασίσουμε ποιες πληροφορίες θα πετάξουμε από την κατάσταση κυψέλης. Αυτή η απόφαση λαμβάνεται από ένα σιγμοειδές στρώμα που ονομάζεται «στρώμα της πύλης ξεχασμού (forget gate layer)». Εξετάζει το ht−1 και xt, και εξάγει έναν αριθμό μεταξύ 0 και 1 για κάθε αριθμό στην κατάσταση κελιού Ct−1. Το 1 αντιπροσωπεύει "διατήρησε πλήρως το στοιχείο" ενώ το 0 αντιπροσωπεύει "πλήρης απόρριψη στοιχείου".

Για παράδειγμά έστω ότι έχουμε ένα γλωσσικό μοντέλο που προσπαθεί να προβλέψει την επόμενη λέξη με βάση τις προηγούμενες. Σε ένα τέτοιο πρόβλημα, η κατάσταση του κελιού μπορεί να περιλαμβάνει το φύλο του παρόντος θέματος, έτσι ώστε να μπορούν να

χρησιμοποιηθούν οι σωστές αντωνυμίες. Όταν βλέπουμε ένα νέο θέμα, θέλουμε να ξεχάσουμε το φύλο του παλιού θέματος.

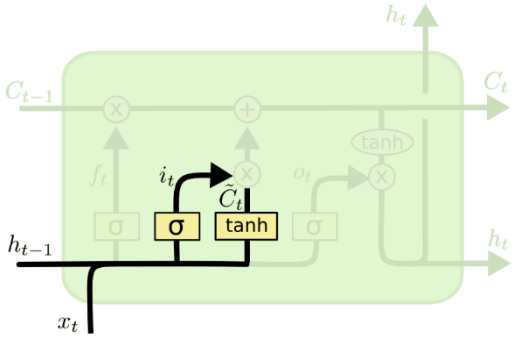


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Εικόνα 15.

Το επόμενο βήμα είναι να αποφασιστεί ποιες νέες πληροφορίες πρόκειται να αποθηκευτούν στην κατάσταση κυψέλης. Αυτό το στάδιο έχει δύο μέρη. Πρώτον, ένα σιγμοειδές στρώμα που ονομάζεται "στρώμα πύλης εισόδου" αποφασίζει ποιες τιμές θα ενημερωθούν. Στη συνέχεια, ένα στρώμα tanh δημιουργεί ένα διάνυσμα νέων υποψήφιων τιμών, C~t, που θα μπορούσε να προστεθεί στην κατάσταση. Στο επόμενο βήμα, γίνεται συνδυασμός των δύο παραπάνω βημάτων για να δημιουργηθεί η ενημέρωση για την κατάσταση.

Στο παράδειγμα του γλωσσικού μοντέλου, θα θέλαμε να προστεθεί το φύλο του νέου θέματος στην κατάσταση κελιού, για να αντικατασταθεί από το παλιό που θα έχει ξεχαστεί.



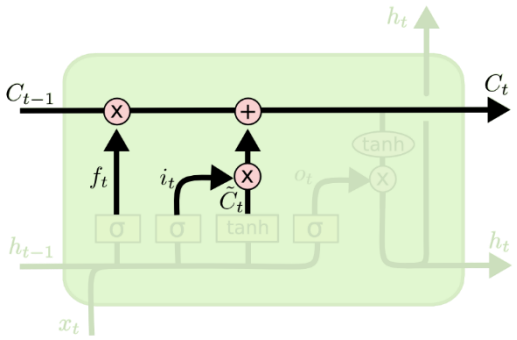
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Εικόνα 16.

Σε αυτό το σημείο ενημερώνεται η παλιά κατάσταση κελιού, Ct−1 , στη νέα κατάσταση κελιού Ct . Στα προηγούμενα βήματα έχει αποφασιστεί ήδη τι θα γίνει.

Πολλαπλασιάζουμε την παλιά κατάσταση επί ft, μη λαμβάνοντας υπόψη τα στοιχεία που αποφασίσαμε να ξεχάσουμε νωρίτερα. Μετά προσθέτετε το $i_t * C_t$. Αυτές είναι οι νέες υποψήφιες τιμές, οι οποίες κλιμακώνονται ανάλογα με το πόσο αποφασίσαμε να ενημερώσουμε την κάθε τιμή κατάστασης.

Στην περίπτωση του γλωσσικού μοντέλου, στο συγκεκριμένο βήμα θα είχαμε τις πληροφορίες σχετικά με το φύλο του παλιού θέματος και θα προσθέταμε τις νέες πληροφορίες, όπως αποφασίστηκε στα προηγούμενα βήματα.

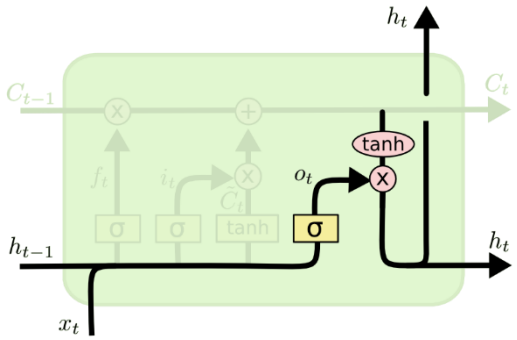


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Εικόνα 17.

Τέλος, πρέπει να αποφασιστεί τι output θα βγάλουμε. Αυτή η έξοδος (output) θα βασίζεται στην κατάσταση του κελιού, αλλά θα είναι μια φιλτραρισμένη έκδοση. Αρχικά, εκτελείται ένα σιγμοειδές στρώμα το οποίο αποφασίζει ποια μέρη της κατάστασης κελιού θα εξαχθούν. Στη συνέχεια, εισάγεται η κυτταρική κατάσταση μέσω tanh (για να είναι οι τιμές μεταξύ −1 και 1) ή οποία πολλαπλασιάζετε με την έξοδο της σιγμοειδούς πύλης.

Για το παράδειγμα του γλωσσικού μοντέλου, μόλις εισαχθεί ένα θέμα, το μοντέλο μπορεί να θέλει να εξάγει πληροφορίες σχετικές με ένα ρήμα, σε περίπτωση που αυτό είναι το επόμενο. Για παράδειγμα, μπορεί να βγάζει αν το θέμα είναι στον ενικό ή τον πληθυντικό, έτσι ώστε να γνωρίζουμε σε ποια μορφή θα πρέπει να συζευχθεί ένα ρήμα εάν αυτό ακολουθεί στη συνέχεια.



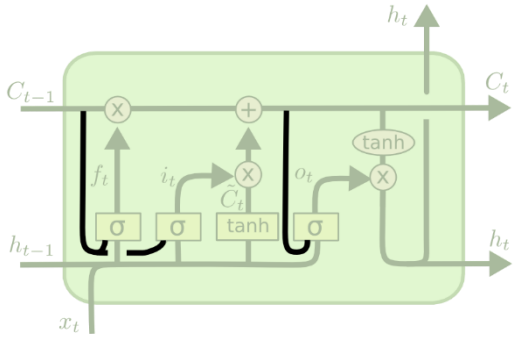
$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Εικόνα 18.

4.1.1 Παραλλαγές δικτύου LSTM

Όλα τα LSTM δίκτυα δεν είναι ακριβώς ίδια με το παραπάνω. Υπάρχουν μερικές μικρές αλλαγές- τροποποιήσεις στα δίκτυα LSTM οι οποίες θα περιγραφούν παρακάτω.

Μια δημοφιλής παραλλαγή LSTM, που εισήχθη από τους Gers & Schmidhuber (2000), προσθέτει τις «peephole connections». Αυτό σημαίνει ότι τα στρώματα πύλης κοιτάζουν την κατάσταση του κελιού.

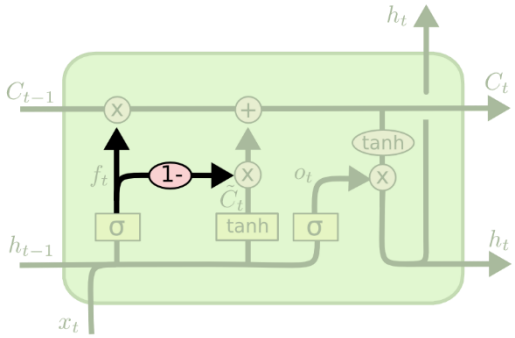


$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \; + \; b_o\right)$$

Εικόνα 19.

Το παραπάνω διάγραμμα προσθέτει peepholes σε όλες τις πύλες.

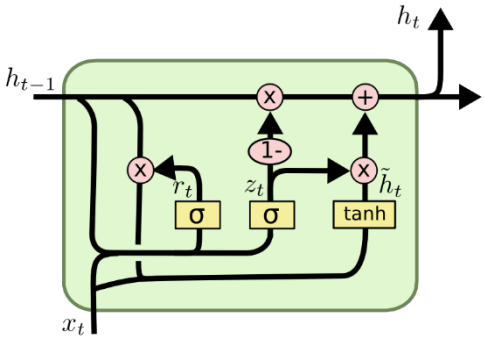
Μια άλλη παραλλαγή είναι η χρήση συζευγμένων πυλών λήψεως και εισόδου. Αντί να αποφασιστεί χωριστά τι να ξεχαστεί και τι νέα πληροφορία να προστεθεί, λαμβάνονται αυτές οι αποφάσεις μαζί. Η διαδικασία στο να ξεχαστεί η πληροφορία, γίνεται μόνο όταν πρόκειται να εισαχθεί νέα πληροφορία.



$$C_t = f_t * C_{t-1} + \boldsymbol{(1 - f_t)} * \tilde{C}_t$$

Εικόνα 20.

Μια ελαφρώς πιο εντυπωσιακή παραλλαγή του LSTM είναι η Gated Recurrent Unit, ή GRU, που εισήχθη από τους Cho, et al. (2014). Συνδυάζει τις πύλες λήψεως και εισόδου σε μια ενιαία "πύλη ενημέρωσης". Επίσης, συγχωνεύει την κατάσταση του κελιού και την κρυφή κατάσταση και κάνει κάποιες άλλες αλλαγές. Το μοντέλο που προκύπτει είναι απλούστερο από τα τυπικά μοντέλα LSTM και γίνεται ολοένα και πιο δημοφιλές.



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$
$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Εικόνα 21.

Αυτές είναι μερικές από τις πιο αξιοσημείωτες παραλλαγές LSTM. Υπάρχουν και άλλες παραλαγγές, όπως τα Depth Gated RNN από τους Yao, et al. (2015). Υπάρχει επίσης κάποια εντελώς διαφορετική προσέγγιση για την αντιμετώπιση μακροπρόθεσμων εξαρτήσεων, όπως τα Clockwork RNN από τους Koutnik, et al. (2014).

Στο ερώτημα ποια από τις παραλλαγές είναι η καλύτερη Greff, et al. (2015) στην σύγκριση δημοφιλών παραλλαγών LSTM, διαπιστώθηκε ότι όλες έχουν περίπου παρόμοια απόδοση. Στην εργασία Jozefowicz, et al. (2015) δοκιμάστηκαν περισσότερες από δέκα χιλιάδες αρχιτεκτονικές RNN, βρίσκοντας κάποιες που λειτουργούσαν καλύτερα από τα LSTM δίκτυα σε ορισμένες περιπτώσεις.[18][21]

4.2 Bagging Regressor

Ένας από τους αλγορίθμους που χρησιμοποιήθηκαν στην παρούσα εργασία για την δημιουργία μοντέλων μηχανικής μάθησης είναι ο Bagging Regreesor. Πρόκειται στην ουσία για έναν μετα-εκτιμητής συνόλου που προσαρμόζει τους βασικούς αναδρομείς σε τυχαία υποσύνολα του αρχικού συνόλου και στη συνέχεια συγκεντρώνει τις μεμονωμένες προβλέψεις τους (είτε με ψηφοφορία είτε με μέσο όρο) για να σχηματίσει μια τελική πρόβλεψη. Ένας τέτοιος μετα-εκτιμητής μπορεί τυπικά να χρησιμοποιηθεί ως τρόπος μείωσης της διακύμανσης ενός εκτιμητή μαύρου κουτιού (π.χ., ενός δέντρου αποφάσεων), εισάγοντας την τυχαιοποίηση στη διαδικασία κατασκευής του και στη συνέχεια δημιουργώντας ένα σύνολο από αυτό.[24][33]

Οι Bagging μέθοδοι βγαίνουν σε διαφορετικές εκδόσεις και διαφοροποιούνται μεταξύ τους λόγω του τρόπου με τον οποίο σχεδιάζουν τυχαία τα υποσύνολα του σετ εκπαίδευσης.

- Όταν τα τυχαία υποσύνολα του συνόλου δεδομένων σχεδιάζονται ως τυχαία υποσύνολα των δειγμάτων, τότε αυτός ο αλγόριθμος είναι γνωστός ως Επικόλληση (Pasting).
- Εάν τα δείγματα λαμβάνονται με αντικατάσταση, τότε η μέθοδος είναι γνωστή ως Bagging.
- Όταν τα τυχαία υποσύνολα του συνόλου δεδομένων σχεδιάζονται ως τυχαία υποσύνολα των χαρακτηριστικών, τότε η μέθοδος είναι γνωστή ως Τυχαίοι Υποχώροι (Random Subspaces).
- Τέλος, όταν οι εκτιμητές βάσης χτίζονται σε υποσύνολα δειγμάτων και χαρακτηριστικών, τότε η μέθοδος είναι γνωστή ως Τυχαίες επιδιορθώσεις (Random Patches).

4.3 Linear Regressor

Μία άλλη μέθοδος που χρησιμοποιήθηκε για την δημιουργία μοντέλων μηχανικής μάθησης είναι η Linear Regressor. Η Γραμμική Παλινδρόμηση ταιριάζει σε ένα γραμμικό μοντέλο με συντελεστές w = (w1, …, wp) ώστε να ελαχιστοποιήσει το υπολειπόμενο άθροισμα των τετραγώνων μεταξύ των παρατηρούμενων στόχων στο σύνολο δεδομένων και των στόχων που προβλέπονται από τη γραμμική προσέγγιση.

4.4 SGD Regressor

Το SGD σημαίνει Stochastic Gradient Descent όπου η διαβάθμιση της απώλειας εκτιμάται σε κάθε δείγμα κάθε φορά και το μοντέλο ενημερώνεται στην πορεία με ένα φθίνον χρονοδιάγραμμα δύναμης (γνωστός και ως ρυθμός εκμάθησης).

Ο κανονικοποιητής είναι μια έννοια που προστίθεται στη συνάρτηση απώλειας που συρρικνώνει τις παραμέτρους του μοντέλου προς το μηδενικό διάνυσμα, χρησιμοποιώντας είτε την τετραγωνισμένη ευκλείδεια νόρμα L2 είτε την απόλυτη νόρμα L1 ή συνδυασμό και των δύο (Elastic Net). Εάν η ενημέρωση παραμέτρων υπερβεί την τιμή 0,0 λόγω του κανονικοποιητή, η ενημέρωση περικόπτεται σε 0,0 για να επιτρέψει την εκμάθηση αραιών μοντέλων και την επίτευξη διαδικτυακής επιλογής δυνατοτήτων.

Η συγκεκριμένη υλοποίηση, λειτουργεί με δεδομένα που αντιπροσωπεύονται ως πυκνοί ανώμαλοι πίνακες (dense numpy arrays) τιμών κινητής υποδιαστολής για τα χαρακτηριστικά. [34]

4.5 Random Forest Regressor

Ο Random Forest είναι ένας μετα-εκτιμητής που ταιριάζει σε μια σειρά από δέντρα απόφασης ταξινόμησης, σε διάφορα υποδείγματα του συνόλου δεδομένων και χρησιμοποιεί τον μέσο όρο για να βελτιώσει την προγνωστική ακρίβεια και τον έλεγχο της υπερπροσαρμογής. Το μέγεθος του δευτερεύοντος δείγματος ελέγχεται με την παράμετρο max_samples εάν έχουμε το bootstrap=True (προεπιλογή), διαφορετικά ολόκληρο το σύνολο δεδομένων χρησιμοποιείται για τη δημιουργία κάθε δέντρου.

4.6 Gradient Boosting

Η GB δημιουργεί ένα προσθετικό μοντέλο με προοδευτικό σκηνικό τρόπο. Επιτρέπει τη βελτιστοποίηση αυθαίρετων συναρτήσεων διαφοροποιήσιμων απωλειών. Σε κάθε στάδιο των n_classes, τα δέντρα παλινδρόμησης προσαρμόζονται στην αρνητική κλίση της συνάρτησης

απώλειας διωνυμικής ή πολυωνυμικής απόκλισης. Η δυαδική ταξινόμηση είναι μια ειδική περίπτωση όπου δημιουργείται ένα μόνο δέντρο παλινδρόμησης.

4.6.1 Extreme Gradient Boosting

Η ενίσχυση κλίσης αναφέρεται σε μια κατηγορία αλγορίθμων μηχανικής μάθησης συνόλου, που μπορούν να χρησιμοποιηθούν για προβλήματα μοντελοποίησης πρόβλεψης ταξινόμησης ή παλινδρόμησης.

Τα σύνολα κατασκευάζονται από μοντέλα δέντρων αποφάσεων. Τα δέντρα προστίθενται ένα κάθε φορά στο σύνολο και ταιριάζουν για τη διόρθωση των σφαλμάτων πρόβλεψης που έγιναν από προηγούμενα μοντέλα. Αυτός είναι ένας τύπος μοντέλου μηχανικής εκμάθησης συνόλου που αναφέρεται ως boosting.

Τα μοντέλα προσαρμόζονται χρησιμοποιώντας οποιαδήποτε αυθαίρετη συνάρτηση διαφοροποιήσιμης απώλειας και αλγόριθμο βελτιστοποίησης gradient descent. Αυτό δίνει στην τεχνική το όνομά της, «ενίσχυση κλίσης», καθώς η κλίση απώλειας ελαχιστοποιείται καθώς το μοντέλο είναι κατάλληλο, σαν ένα νευρωνικό δίκτυο.

Το Extreme Gradient Boosting, ή XGBoost για συντομία, είναι μια αποτελεσματική εφαρμογή ανοιχτού κώδικα του αλγόριθμου ενίσχυσης κλίσης (gradient boosting algorithm). Ο XGBoost είναι ένας αλγόριθμος ανοιχτού κώδικα που έχει υλοποιηθεί σε Python.

Αναπτύχθηκε αρχικά από τον Tianqi Chen και περιγράφηκε από τους Chen και Carlos Guestrin στην εργασία τους του 2016 με τίτλο "XGBoost: A Scalable Tree Boosting System".

Είναι σχεδιασμένο να είναι υπολογιστικά αποδοτικό (π.χ. γρήγορο στην εκτέλεση) και εξαιρετικά αποτελεσματικό, ίσως πιο αποτελεσματικό από άλλες εφαρμογές ανοιχτού κώδικα.

Οι δύο κύριοι λόγοι για να χρησιμοποιήσετε το XGBoost είναι η ταχύτητα εκτέλεσης και η απόδοση του μοντέλου.

Το XGBoost κυριαρχεί σε δομημένα ή πινακοποιημένα σύνολα δεδομένων σχετικά με προβλήματα μοντελοποίησης πρόβλεψης ταξινόμησης και παλινδρόμησης. Η απόδειξη είναι ότι είναι ο αλγόριθμος που επιλέγεται περισσότερο από τους νικητές διαγωνισμών στην ανταγωνιστική πλατφόρμα επιστήμης δεδομένων Kaggle.

5

Δημιουργία μοντέλων μηχανικής μάθησης

Η μηχανική μάθηση μπορεί να εφαρμοστεί σε πολλούς τομείς της επιστήμης των υπολογιστών, όπως η λήψη αποφάσεων, η πρόβλεψη, και ειδικά η πρόβλεψη της τιμής των μετοχών ή της συναλλαγματικής ισοτιμίας κ.λ.π. Με την μηχανική μάθηση επιτυγχάνεται η δημιουργία εκπαιδευμένων μοντέλων μέσω του υπολογιστή για την ταξινόμηση ή τη ομαδοποίηση δεδομένων από το σύνολο δεδομένων που χρησιμοποιούνται. Με την έκρηξη των δεδομένων στις μέρες μας, ιδιαίτερα με τα μεγάλα δεδομένα (big data) των επιχειρήσεων, οι τεχνικές μηχανικής μάθησης μπορούν να βοηθήσουν τους ανθρώπους να κοιτάξουν σε βάθος και να αναλύσουν τα δεδομένα, προκειμένου να εξάγουν χρήσιμες πληροφορίες με βάση τα δεδομένα που έχουν υποστεί ανάλυση. Στη χρηματιστηριακή αγορά, μπορεί να προβλεφθεί ή άνοδος ή πτώση των τιμών ισοτιμίας μεταξύ νομισμάτων με τη βοήθεια της μηχανικής μάθησης και της παρατήρησης του συνόλου δεδομένων που έχουν συλλεχθεί για τον συγκεκριμένο σκοπό. Τα συστήματα τεχνητής νοημοσύνης χρησιμοποιούνται με μεθόδους μηχανικής μάθησης όπως οι μέθοδοι Λογιστικής Παλινδρόμησης, Perceptron, Naive Bayes. [2]. Στόχος των μοντέλων της μηχανικής μάθησης είναι η ταξινόμηση ή το σύνολο δεδομένων παλινδρόμησης σε εναλλακτικές κατηγορίες που ονομάζονται ως έξοδοι. Αυτές οι μεταβλητές στόχου (y) είναι συνήθως οι ονομαστικές (διακριτές) ή οι συνεχείς τιμές [1]. Για απλότητα, η έξοδος του y στο πρόβλημα ταξινόμησης μπορεί να θεωρηθεί ως δυαδικές τιμές όπως 0 ή 1.

5.1 Μοντέλο LSTM

Το μοντέλο LSTM που θα αναλυθεί εκτενέστερα παρακάτω, έχει σαν στόχο την πρόβλεψη της τιμής ισοτιμίας EUR/USD της επόμενης ημέρας, εξετάζοντας τις τελευταίες 20 τιμές δεδομένων των κεριών ημέρας. Έχει παρατηρηθεί εμπειρικά ότι η ύπαρξη της συνεχόμενης εισαγωγής τιμών, βελτιώνει την ακρίβεια του μοντέλου. Η εισαγωγή και άλλων time frames όπως εβδομάδας και ώρας της ημέρας ως χαρακτηριστικά, θα μπορούσε να βελτιώσει την

απόδοση του μοντέλου. Στην παρούσα εργασία έχουν χρησιμοποιηθεί δεδομένα κεριών ημέρας από το 1971 μέχρι και το 2020.

Στο μοντέλο που θα εκπαιδευτεί, εισάγονται δεδομένα όπως open, high, low, close, volume από την πλατφόρμα metatrader5. Τα δεδομένα αυτά, αναφέρονται σε ημερήσια κεριά της ισοτιμίας EUR/USD.

Ξεκινώντας την δημιουργία του μοντέλου, αρχικά φορτώνονται τα δεδομένα από excel αρχείο, το οποίο περιέχει ιστορικά δεδομένα κεριών ημέρας. Συγκεκριμένα περιλαμβάνει 5 στήλες με πεδία τα date, open, high, low, volume. Ενδεικτικά εμφανίζονται οι πρώτες 5 γραμμές και οι τελευταίες 5 γραμμές του αρχείου.

date	open	high	low	close	volume
1971-01-04	0.5369	0.5369	0.5369	0.5369	1
1971-01-05	0.5366	0.5366	0.5366	0.5366	1
1971-01-06	0.5365	0.5365	0.5365	0.5365	1
1971-01-07	0.5368	0.5368	0.5368	0.5368	1
1971-01-08	0.5371	0.5371	0.5371	0.5371	1

Πίνακας 1. Πέντε πρώτες γραμμές του αρχείου.

Ακολουθούν οι 5 τελευταίες γραμμές του αρχείου. Εμφανίζοντας τις πρώτες και τις τελευταίες γραμμές επιβεβαιώνεται ότι έχουν φορτωθεί σωστά όλα τα δεδομένα.

date	open	high	low	close	volume
2021-12-15	1.12575	1.12992	1.12217	1.12919	71224
2021-12-16	1.12905	1.13602	1.12811	1.13296	72728
2021-12-17	1.13284	1.13489	1.12344	1.12372	63263
2021-12-20	1.12356	1.13040	1.12341	1.12769	60738
2021-12-21	1.12743	1.13027	1.12609	1.12822	54420

Πίνακας 2. Πέντε τελευταίες γραμμές του αρχείου.

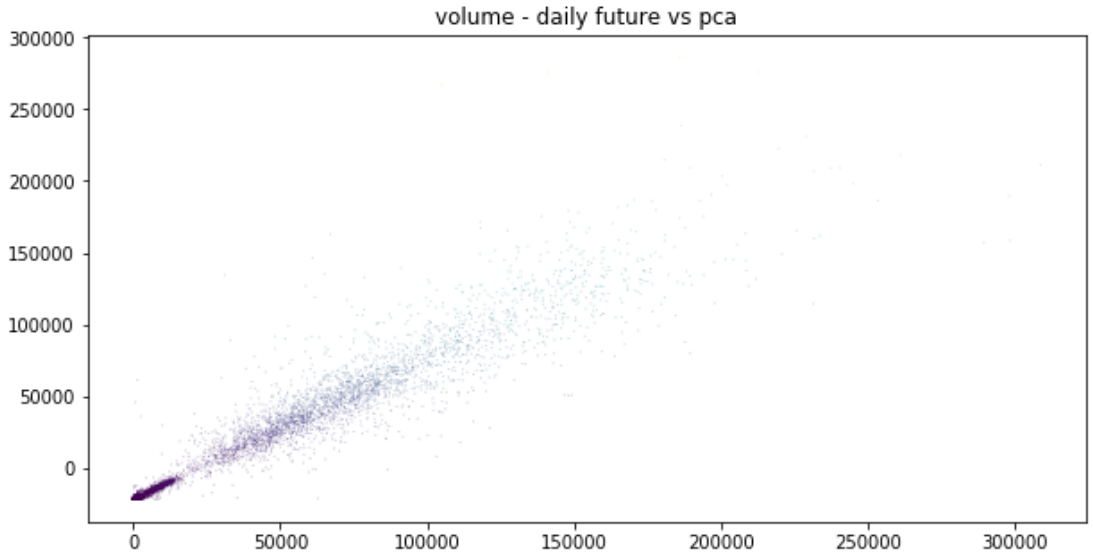
Η εμφάνιση των δεδομένων είναι πολύ χρήσιμη για τον έλεγχο των τιμών των δεδομένων κυρίως μετά από το update του αρχείου δεδομένων, που θα χρησιμοποιηθεί στο μοντέλο.

Στην συνέχεια χρησιμοποιούνται κάποια επιπρόσθετα χαρακτηριστικά τα οποία δίνουν περισσότερες πληροφορίες για τα δεδομένα, τα οποία βελτιώνουν την απόδοση του μοντέλου.

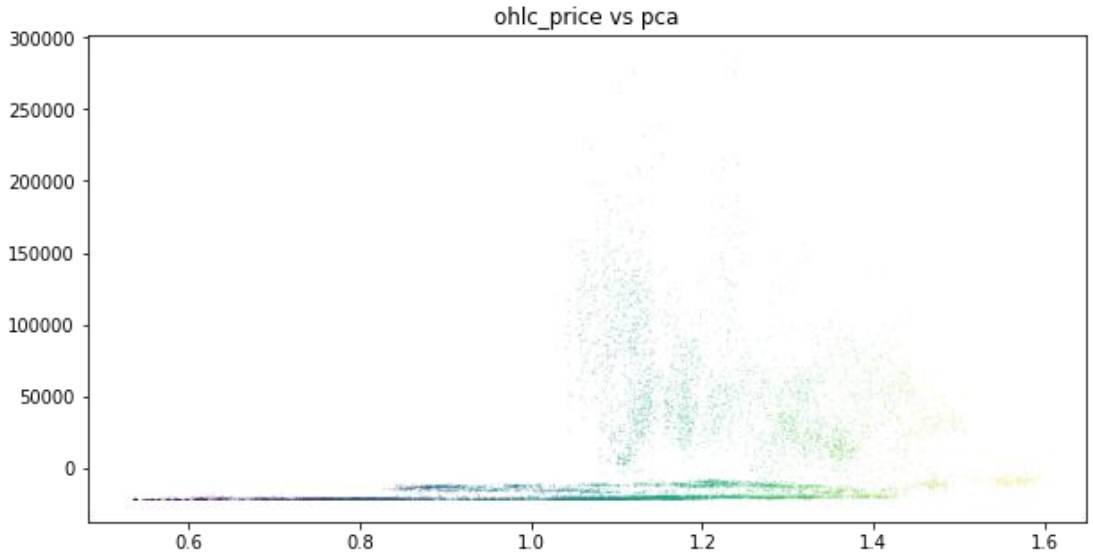
Συγκεκριμένα, στα υπάρχοντα δεδομένα προστέθηκαν πληροφορίες για δεδομένα:

- Εβδομάδας,
- στιγμής , (volume * (open - close),
- μέσου όρου,
- εύρος κεριού, (high - low),
- ανοίγματος-κλεισίματος-χαμηλού-υψηλού κεριού ((low + high + open + close)/4),
- διαφορά ανοίγματος κλεισίματος κεριού (open - close).

Επιπρόσθετα χρησιμοποιείται το PCA (Principal component analysis) το οποίο βελτιώνει λίγο την ακρίβεια του μοντέλου.



Εικόνα 22. Volume vs Pca.



Εικόνα 23. Ohlc vs Pca.

5.1.1 PCA

Το PCA χρησιμοποιείται για την αποσύνθεση ενός δεδομένου πολλών μεταβλητών σε ένα σύνολο διαδοχικών ορθογώνιων στοιχείων που εξηγούν ένα μέγιστο ποσό της διακύμανσης. Στο scikit-learn, το PCA υλοποιείται ως αντικείμενο μετασχηματιστή που μαθαίνει n στοιχεία με τη μέθοδο προσαρμογής του και μπορεί να χρησιμοποιηθεί σε νέα δεδομένα για να τα προβάλει σε αυτά τα στοιχεία.

Το PCA κεντράρει, αλλά δεν κλιμακώνει τα δεδομένα εισόδου για κάθε χαρακτηριστικό πριν από την εφαρμογή του SVD. Η προαιρετική παράμετρος whiten=True καθιστά δυνατή την

προβολή των δεδομένων στον μοναδικό χώρο ενώ κλιμακώνεται κάθε στοιχείο σε διακύμανση μονάδων. Αυτό είναι συχνά χρήσιμο εάν τα μοντέλα κάνουν ισχυρές υποθέσεις σχετικά με την ισοτροπία του σήματος. Αυτό συμβαίνει για παράδειγμα με τις μηχανές υποστήριξης διανυσμάτων με τον πυρήνα RBF και τον αλγόριθμο ομαδοποίησης K-Means.

Ο αρχικός πίνακας δεδομένων, τροποποιήθηκε με τις εισαγωγές των παραπάνω χαρακτηριστικών όπως φαίνεται στους πίνακες 3 και 4.

date	open	high	low	close	volume	hour	day	week	momentum	avg_price	range	ohlc_price	oc_diff	pca
1971-01-04	0.5369	0.5369	0.5369	0.5369	1	0	0	1	0.0	0.5369	0.0	0.5369	0.0	-22328.890625
1971-01-05	0.5366	0.5366	0.5366	0.5366	1	0	1	1	0.0	0.5366	0.0	0.5366	0.0	-22328.960938
1971-01-06	0.5365	0.5365	0.5365	0.5365	1	0	2	1	0.0	0.5365	0.0	0.5365	0.0	-22328.974609
1971-01-07	0.5368	0.5368	0.5368	0.5368	1	0	3	1	0.0	0.5368	0.0	0.5368	0.0	-22328.962891
1971-01-08	0.5371	0.5371	0.5371	0.5371	1	0	4	1	0.0	0.5371	0.0	0.5371	0.0	-22328.708984

Πίνακας 3. Πέντε πρώτες γραμμές τροποποιημένου πίνακα δεδομένων.

date	open	high	low	close	volume	hour	day	week	momentum	avg_price	range	ohlc_price	oc_diff	pca
2021-12-15	1.12575	1.12992	1.12217	1.12919	71224	0	2	50	-245.01056	1.126045	0.00775	1.126758	-0.00344	48894.039062
2021-12-16	1.12905	1.13602	1.12811	1.13296	72728	0	3	50	-284.36648	1.132065	0.00791	1.131535	-0.00391	50398.039062
2021-12-17	1.13284	1.13489	1.12344	1.12372	63263	0	4	50	576.95856	1.129165	0.01145	1.128722	0.00912	40933.015625

date	open	high	low	close	volume	hour	day	week	momentum	avg_price	range	ohlc_price	oc_diff	pca
2021-12-15	1.12575	1.12992	1.12217	1.12919	71224	0	2	50	-245.01056	1.126045	0.00775	1.126758	-0.00344	48894.039062
2021-12-20	1.12356	1.13040	1.12341	1.12769	60738	0	0	51	-250.84794	1.126905	0.00699	1.126265	-0.00413	38408.039062
2021-12-21	1.12743	1.13027	1.12609	1.12822	54420	0	1	51	-42.99180	1.128180	0.00418	1.128003	-0.00079	32090.035156

Πίνακας 4. Πέντε τελευταίες γραμμές τροποποιημένου πίνακα δεδομένων.

5.1.2 Pearson Correlation

Ο συντελεστής συσχέτισης Pearson είναι ένα μέτρο της γραμμικής σχέσης μεταξύ δύο χαρακτηριστικών. Είναι ο λόγος της συνδιακύμανσης των x και y προς το γινόμενο των τυπικών αποκλίσεων τους. Συχνά συμβολίζεται με το γράμμα r και ονομάζεται συντελεστής r του Pearson. Η τιμή του Pearson εκφράζεται μαθηματικά με την εξίσωση:

$r = \Sigma_i((x_i - \text{mean}(x))(y_i - \text{mean}(y))) \, (\sqrt{\Sigma_i(x_i - \text{mean}(x))^2} \, \sqrt{\Sigma_i(y_i - \text{mean}(y))^2})^{-1}$

Εδώ, το i λαμβάνει τις τιμές 1, 2, …, n. Οι μέσες τιμές των x και y συμβολίζονται με mean(x) και mean(y). Αυτός ο τύπος δείχνει ότι εάν μεγαλύτερες τιμές x τείνουν να αντιστοιχούν σε μεγαλύτερες τιμές y και αντίστροφα, τότε το r είναι θετικό. Από την άλλη πλευρά, εάν οι μεγαλύτερες τιμές x συνδέονται κυρίως με μικρότερες τιμές y και αντίστροφα, τότε το r είναι αρνητικό.

Ακολουθούν ορισμένα σημαντικά στοιχεία σχετικά με τον συντελεστή συσχέτισης Pearson:

- Ο συντελεστής συσχέτισης Pearson μπορεί να λάβει οποιαδήποτε πραγματική τιμή στην περιοχή $-1 \leq r \leq 1$.
- Η μέγιστη τιμή r = 1 αντιστοιχεί στην περίπτωση στην οποία υπάρχει μια τέλεια θετική γραμμική σχέση μεταξύ x και y. Με άλλα λόγια, μεγαλύτερες τιμές x αντιστοιχούν σε μεγαλύτερες τιμές y και αντίστροφα.
- Η τιμή r > 0 υποδηλώνει θετική συσχέτιση μεταξύ x και y.
- Η τιμή r = 0 αντιστοιχεί στην περίπτωση στην οποία δεν υπάρχει γραμμική σχέση μεταξύ x και y.
- Η τιμή r < 0 υποδηλώνει αρνητική συσχέτιση μεταξύ x και y.
- Η ελάχιστη τιμή r = −1 αντιστοιχεί στην περίπτωση που υπάρχει μια τέλεια αρνητική γραμμική σχέση μεταξύ x και y. Με άλλα λόγια, μεγαλύτερες τιμές x αντιστοιχούν σε μικρότερες τιμές y και αντίστροφα.

Τα παραπάνω γεγονότα μπορούν να συνοψιστούν:

Pearson's r	Τιμή Συσχέτιση μεταξύ x και y

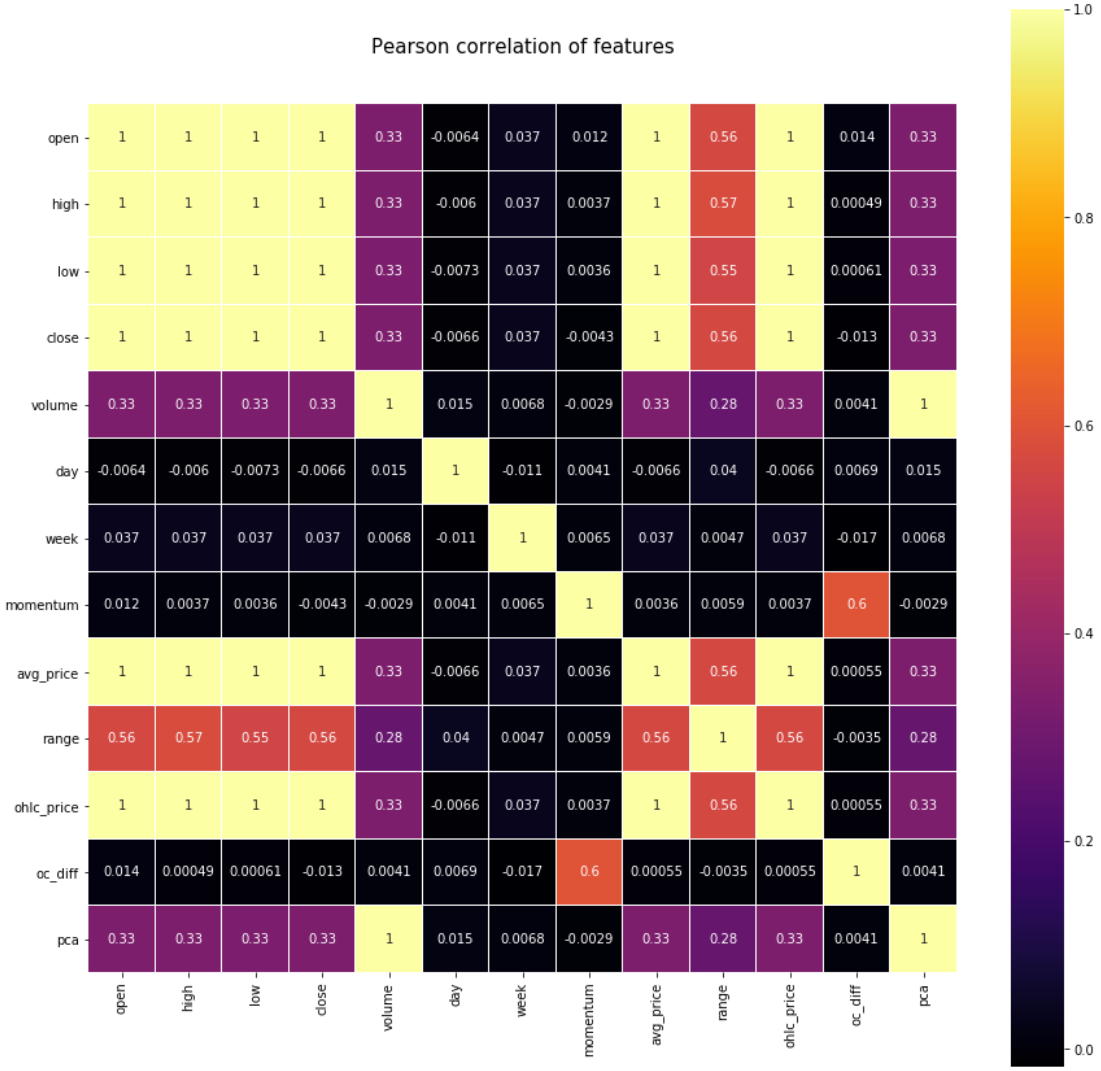
ίσο με 1	τέλεια θετική γραμμική σχέση
μεγαλύτερη από 0	θετική συσχέτιση
ίσο με 0	καμία γραμμική σχέση
μικρότερη από 0	αρνητική συσχέτιση
ίσο με -1 τέλεια	αρνητική γραμμική σχέση

Εν ολίγοις, μια μεγαλύτερη απόλυτη τιμή του r δείχνει ισχυρότερη συσχέτιση, πιο κοντά σε μια γραμμική συνάρτηση. Μια μικρότερη απόλυτη τιμή του r υποδηλώνει ασθενέστερη συσχέτιση.

Οι συντελεστές συσχέτισης είναι ένα ευρέως χρησιμοποιούμενο στατιστικό μέτρο στις επενδύσεις. Διαδραματίζουν πολύ σημαντικό ρόλο σε τομείς όπως η σύνθεση χαρτοφυλακίου, οι ποσοτικές συναλλαγές (trading) και η αξιολόγηση της απόδοσης. Για παράδειγμα, ορισμένοι διαχειριστές χαρτοφυλακίου θα παρακολουθούν τους συντελεστές συσχέτισης μεμονωμένων μετοχών στα χαρτοφυλάκιά τους, προκειμένου να διασφαλίσουν ότι η συνολική αστάθεια των χαρτοφυλακίων τους διατηρείται εντός αποδεκτών ορίων.

Ομοίως, οι αναλυτές μερικές φορές χρησιμοποιούν συντελεστές συσχέτισης για να προβλέψουν πώς ένα συγκεκριμένο περιουσιακό στοιχείο θα επηρεαστεί από μια αλλαγή σε έναν εξωτερικό παράγοντα, όπως η τιμή ενός εμπορεύματος ή ένα επιτόκιο.

Στην παρούσα εργασία, ο συντελεστής συσχέτισης Pearson correlation χρησιμοποιείται για τον προσδιορισμό συσχετίσεων μεταξύ των χαρακτηριστικών του συνόλου των δεδομένων.



Εικόνα 24. Pearson correlation

Στην παραπάνω εικόνα, απεικονίζεται η συσχέτιση των χαρακτηριστικών που θα χρησιμοποιηθούν στην δημιουργία του μοντέλου πρόβλεψης τιμής ισοτιμίας EUR/USD.

Τα δεδομένα για να χρησιμοποιηθούν για ανάλυση, θα πρέπει πρώτα να τροποποιηθούν. Για την τροποποίηση αρχικά, γίνεται χρήση της μεθόδου MinMaxScaler.

Στην ουσία πρόκειται για έναν εκτιμητή οποίος κλιμακώνει και μεταφράζει κάθε χαρακτηριστικό ξεχωριστά έτσι ώστε να βρίσκεται στο δεδομένο εύρος του σετ εκπαίδευσης, π.χ. μεταξύ μηδέν και ενός.

Ο μετασχηματισμός δίνεται από τον τύπο:

X_std = (X - X.min(άξονας=0)) / (X.max(άξονας=0) - X.min(άξονας=0))

X_scaled = X_std * (max - min) + min

όπου min, max = εύρος_χαρακτηριστικών.

Αυτός ο μετασχηματισμός χρησιμοποιείται συχνά ως εναλλακτική λύση στη μηδενική μέση, μοναδιαία κλίμακα διακύμανσης.

Στην προκειμένη περίπτωση έγινε χρήση της μεθόδου:

fit_transform(X[, y]) σύμφωνα με την οποία τα δεδομένα προσαρμόζονται και μετά μετατρέπονται.

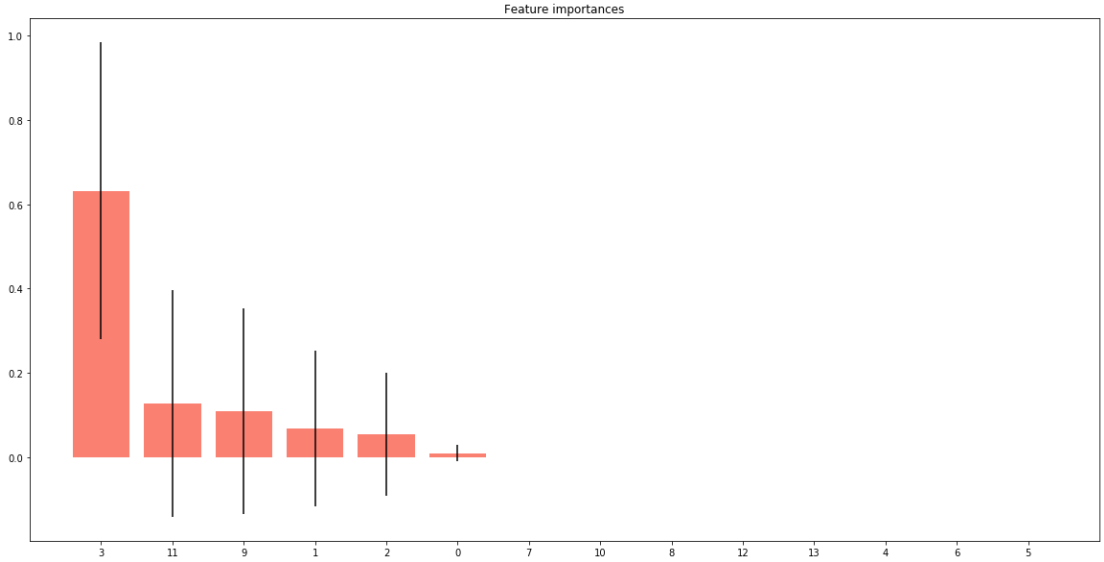
5.1.3 Feature importances

Στην περίπτωση του συγκεκριμένου dataset χρησιμοποιήθηκε η MinMaxScaler από το Skikit-Learn για να κάνουμε scale το dataset σε αριθμούς ενδιάμεσα από το 0 και το 1.

Στο επόμενο βήμα της δημιουργίας του μοντέλου δημιουργείται το data set το οποίο θα ελέγχει τις συσχετίσεις 20 κεριά πίσω (δηλαδή 20 μέρες πίσω).

```
def create_dataset(dataset, look_back=20):
    dataX, dataY = [], []
    for i in range(len(dataset)-look_back-1):
        a = dataset[i:(i+look_back)]
        dataX.append(a)
        dataY.append(dataset[i + look_back])
    return np.array(dataX), np.array(dataY)
```

Ένα χρήσιμο διάγραμμα για τα χαρακτηριστικά που θα χρησιμοποιηθούν για την δημιουργία του μοντέλου είναι το διάγραμμα σπουδαιότητας των χαρακτηριστικών. Το διάγραμμα βασίζεται στον μετα-εκτιμητή Forest Regressor για να υπολογιστεί η σπουδαιότητα των χαρακτηριστικών. Η εικόνα που ακολουθεί απεικονίζει την σπουδαιότητα των χαρακτηριστικών με την χρήση Random Forest.



Εικόνα 25. Σπουδαιότητα χαρακτηριστικών.

Οι σπουδαιότητες των χαρακτηριστικών, παρέχονται από το προσαρμοσμένο χαρακτηριστικό feature_importances_ και υπολογίζονται ως ο μέσος όρος και η τυπική απόκλιση της συσσώρευσης της μείωσης του θορύβου σε κάθε δέντρο.

5.1.4 Random Forest Regressor

Random forest regressor είναι ένας μετα-εκτιμητής που ταιριάζει σε μια σειρά από δέντρα απόφασης ταξινόμησης σε διάφορα υποδείγματα του συνόλου δεδομένων και χρησιμοποιεί τον μέσο όρο για να βελτιώσει την προγνωστική ακρίβεια και τον έλεγχο της υπερπροσαρμογής. Το μέγεθος του δευτερεύοντος δείγματος ελέγχεται με την παράμετρο max_samples εάν έχουμε την τιμή bootstrap=True (προεπιλογή), διαφορετικά ολόκληρο το σύνολο δεδομένων χρησιμοποιείται για τη δημιουργία κάθε δέντρου.

Στον random forest αλγόριθμο, κάθε δέντρο στο σύνολο δημιουργείται από ένα δείγμα που έχει σχεδιαστεί με αντικατάσταση (δηλαδή, ένα δείγμα bootstrap) από το σετ εκπαίδευσης.

Επιπλέον, κατά τον διαχωρισμό κάθε κόμβου κατά την κατασκευή ενός δέντρου, ο καλύτερος διαχωρισμός βρίσκεται είτε από όλα τα χαρακτηριστικά εισόδου είτε από ένα τυχαίο υποσύνολο μεγέθους max_features. (Υπάρχουν οδηγίες ρύθμισης παραμέτρων στο link https://scikit-learn.org/stable/modules/ensemble.html#random-forest-parameters).

Ο σκοπός αυτών των δύο πηγών τυχαιότητας είναι να μειώσουν τη διακύμανση του εκτιμητή random forest. Τα μεμονωμένα δέντρα απόφασης παρουσιάζουν συνήθως υψηλή διακύμανση και τείνουν να υπερισχύουν. Η εγχυόμενη τυχαιότητα στα δάση (forests) αποδίδει δέντρα απόφασης με αποσυνδεδεμένα σφάλματα πρόβλεψης. Λαμβάνοντας τον μέσο όρο αυτών των προβλέψεων, ορισμένα σφάλματα μπορούν να ακυρωθούν. Τα Random forests επιτυγχάνουν μειωμένη διακύμανση συνδυάζοντας διαφορετικά δέντρα, μερικές φορές με το κόστος μιας ελαφριάς αύξησης της προκατάληψης. Στην πράξη, η μείωση της διακύμανσης είναι συχνά σημαντική, οπότε προκύπτει ένα συνολικά καλύτερο μοντέλο.

Στο επόμενο βήμα χωρίζεται το data set ώστε το 90% των δεδομένων να χρησιμοποιηθούν για training και το 10% για test.

Μετά τον διαχωρισμό του dataset σειρά έχει η δημιουργία του νευρωνικού δικτύου. Συγκεκριμένα δημιουργείται ένα LSTM δίκτυο με το οποίο θα εκπαιδευτεί το συγκεκριμένο μοντέλο και αποτελεί ένα από τα μοντέλα που χρησιμοποιούνται στην παρούσα εργασία.

```
model = Sequential()
model.add(LSTM(20, input_shape=(X.shape[1], X.shape[2]), return_sequences=True))
model.add(LSTM(10, return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(4, return_sequences=False))
```

model.add(Dense(4, kernel_initializer='uniform', activation='relu'))

model.add(Dense(1, kernel_initializer='uniform', activation='relu'))

model.compile(loss='mean_squared_error', optimizer='adam', metrics=['mae', 'mse'])

Με τον παραπάνω κώδικα δημιουργείται ένα LSTM νευρωνικό δίκτυο. Τα LSTM αναμένουν τα δεδομένα να είναι σε μια συγκεκριμένη μορφή, συνήθως μια συστοιχία πίνακα. Ξεκινώντας, δημιουργούνται αρχικά δεδομένα 20 κεριών πίσω, τα οποία μετατρέπονται σε πίνακα χρησιμοποιώντας την βιβλιοθήκη NumPy. Στη συνέχεια, τα δεδομένα μετατρέπονται σε έναν πίνακα με δείγματα X για 20 χρονικές σημάνσεις (data frames).

Για την δημιουργία του συγκεκριμένου LSTM μοντέλου χρειάστηκε να χρησιμοποιηθούν κάποια modules από το Keras. Συγκεκριμένα χρησιμοποιήθηκαν :

- Sequential για την αρχικοποίηση του νευρωνικού δικτύου,
- Dense για την προσθήκη ενός πυκνά συνδεδεμένου στρώματος νευρωνικού δικτύου,
- LSTM για την προσθήκη του Long Short-Term Memory layer,
- Dropout για την προσθήκη dropout layers που αποτρέπουν την υπερτροφοδότηση (overfitting).

Το LSTM μοντέλο που δημιουργείται έχει 3 layers των 20, 10 και 4 νευρώνων και hidden layers των 4 και 1 νευρώνων.

Ο συντελεστής των Dropout Layers έχει προσδιοριστεί με την τιμή 0,2 που σημαίνει ότι το 20% των layers θα απορριφθεί.

Το attribute return_sequences έχει την τιμή True που σημαίνει ότι η έξοδος της κρυμμένης κατάστασης κάθε νευρώνα, αποτελεί είσοδο για το επόμενο LSTM layer.

Ο kernel_initializer χρησιμοποιείται για τον πίνακα βαρών (weights) που χρησιμοποιείται για την γραμμικό μετασχηματισμό των εισόδων (inputs).

Η συνάρτηση ενεργοποίησης είναι υπεύθυνη για τη μετατροπή της αθροιστικής σταθμισμένης εισόδου από τον κόμβο στην ενεργοποίηση του κόμβου ή στην έξοδο για αυτήν την είσοδο.

Η διορθωμένη συνάρτηση γραμμικής ενεργοποίησης ή ReLU για συντομία, είναι μια τμηματικά γραμμική συνάρτηση που θα εξάγει απευθείας την είσοδο εάν είναι θετική, διαφορετικά, θα βγάζει μηδέν. Έχει γίνει η προεπιλεγμένη λειτουργία ενεργοποίησης για πολλούς τύπους νευρωνικών δικτύων, επειδή ένα μοντέλο που το χρησιμοποιεί είναι πιο εύκολο να εκπαιδευτεί και συχνά επιτυγχάνει καλύτερη απόδοση.

Η τροποποίηση των προεπιλεγμένων παραμέτρων επιτρέπει να χρησιμοποιούνται μη μηδενικά όρια, να αλλάζεται η μέγιστη τιμή της ενεργοποίησης και να χρησιμοποιείται ένα μη μηδενικό πολλαπλάσιο της εισόδου για τιμές κάτω από το όριο.

Μετά την ρύθμιση των παραπάνω παραμέτρων γίνεται compile στο μοντέλο, χρησιμοποιώντας τον Adam έναν διαδεδομένο optimizer. Για το loss επιλέγεται το mean_squared_error όπου θα υπολογιστεί ο μέσος όρος των τετραγωνικών σφαλμάτων.

Στην συνέχεια γίνεται προσαρμογή του μοντέλου (fit) όπου θα τρέξει για 3000 εποχές με batch size 500.

```
callbacks_list = [checkpoint]
history = model.fit(trainX, trainY, epochs=200, batch_size=500, verbose=0,
callbacks=callbacks_list, validation_split=0.1)
```

5.1.5 Compile

Απαιτείται το compilation ενός μοντέλου για να οριστικοποιηθεί το μοντέλο και να είναι πλήρως έτοιμο για χρήση. Για τη μεταγλώττιση, πρέπει να καθορίσουμε έναν βελτιστοποιητή και μια συνάρτηση απώλειας. Μπορούμε να μεταγλωττίσουμε ένα μοντέλο χρησιμοποιώντας το χαρακτηριστικό compile.

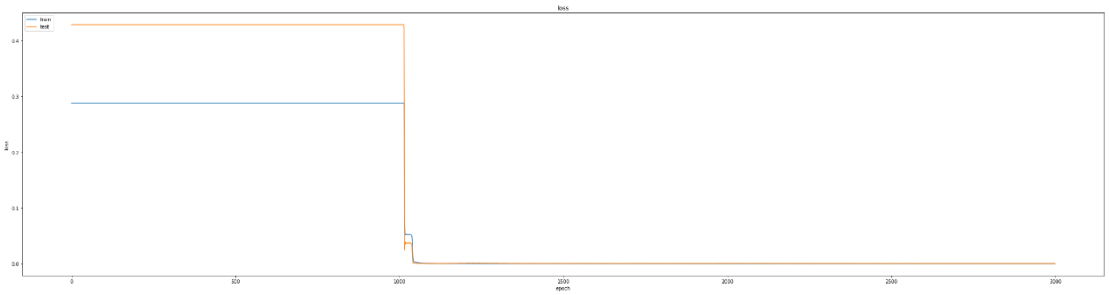
- optimizer: Σε αυτό το χαρακτηριστικό, θα περαστεί ο optimizer που θα χρησιμοποιηθεί για τον συγκεκριμένο μοντέλο. Υπάρχουν διάφοροι βελτιστοποιητές όπως SGD, Adam. Στο μοντέλο LSTM χρησιμοποιήθηκε ο βελτιστοποιητής Adam.
- loss: Σε αυτό το χαρακτηριστικό, θα γίνει χρήση μιας συνάρτηση απώλειας που έχει επιλεχθεί για το μοντέλο. Στο συγκεκριμένο μοντέλο επιλέχθηκε η mean_squared_error function η οποία υπολογίζει τον μέσο όρο των τετραγώνων μεταξύ των πραγματικών τιμών και των τιμών πρόβλεψης.
- metrics: Καθορίζεται μέσου του συγκεκριμένου χαρακτηριστικού, η μέτρηση με την οποία να βαθμολογηθεί το μοντέλο. Τα metrics που χρησιμοποιήθηκαν είναι τα mae (mean absolute error) και mse (mean squared error).

5.1.6 Optimizers

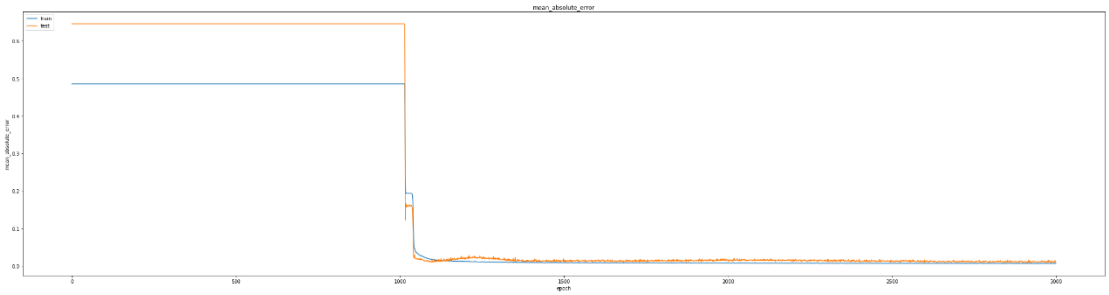
Στη μηχανική μάθηση, η βελτιστοποίηση είναι μια σημαντική διαδικασία που βελτιστοποιεί τα βάρη εισόδου συγκρίνοντας τη συνάρτηση πρόβλεψης και απώλειας. Το Keras παρέχει αρκετούς βελτιστοποιητές όπως:

- SGD − Βελτιστοποιητής στοχαστικής κλίσης κατάβασης.
- RMSprop − RMSProp optimizer.
- Adagrad − Adagrad optimizer.
- Adadelta − Adadelta optimizer.
- Adam − Adam optimizer.
- Adamax − Adamax optimizer from Adam.
- Nadam − Nesterov Adam optimizer.

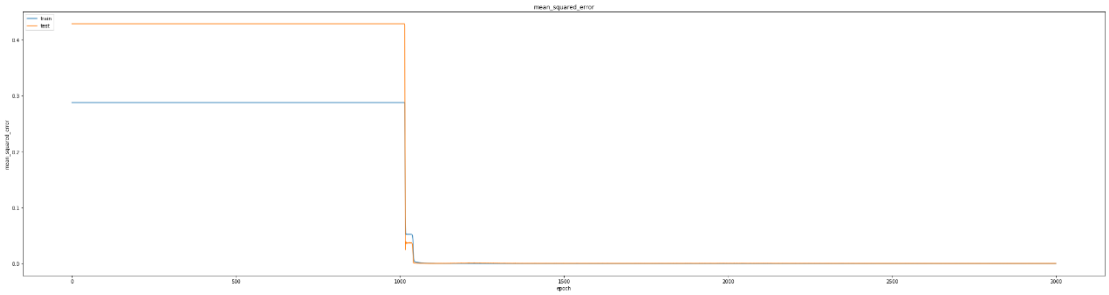
Μετά την εκπαίδευση του μοντέλου παρουσιάζονται τα συγκριτικά αποτελέσματα του train και test dataset.



Εικόνα 26. Loss train-test.



Εικόνα 27. mean absolute error train-test.



Εικόνα 28. Mean squared error train-test.

Μετά την εμφάνιση των αποτελεσμάτων το μοντέλο σώζεται σε αρχείο με κατάληξη .h5 ώστε να χρησιμοποιηθεί μεταγενέστερα στην Django εφαρμογή που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας.

5.2 Μοντέλο Linear Regression

Πριν ξεκινήσει η δημιουργία του μοντέλου, φορτώθηκαν τα ίδια δεδομένα που χρησιμοποιήθηκαν για την δημιουργία του προηγούμενου μοντέλου (πίνακας 1, πίνακας 2). Για την δημιουργία του μοντέλου μηχανικής μάθησης Linear Regression χρειάστηκε να εισαχθεί από το sklearn η βιβλιοθήκη LinearRegression.

Πριν προστεθούν έξτρα χαρακτηριστικά (που ίσως βοηθήσουν την απόδοση του μοντέλου), έγινε προσπάθεια να αφαιρεθεί ο θόρυβος των δεδομένων. Ο θόρυβος λογίζεται από τις άνω και κάτω ακραίες τιμές και υπολογίζεται από την διαφορά του μέσου όρου του volume από την $3^{η}$ τυπική απόκλιση.

#αφαίρεση θορύβου άνω άκρων

```
vol_cut_off = eu.volume.std()*3 + eu.volume.mean()
eu.volume[eu.volume > vol_cut_off] = vol_cut_off

#αφαίρεση θορύβου κάτω άκρων
vol_cut_off =  eu.volume.mean() - eu.volume.std()*3
eu.volume[eu.volume < vol_cut_off] = vol_cut_off
```

Στην συνέχεια χρησιμοποιήθηκαν παράμετροι για την ενίσχυση των δεδομένων που θα χρησιμοποιηθούν για την εκπαίδευση του μοντέλου όπως:

- μέσος όρος κλεισίματος 5 κεριών
- μέσος όρος κλεισίματος 30 κεριών
- μέσος όρος κλεισίματος 90 κεριών
- μέσος όρος κλεισίματος 365 κεριών,
- αναλογία μέσου όρου 5 με 30 κεριών,
- αναλογία μέσου όρου 5 με 90 κεριών,
- αναλογία μέσου όρου 5 με 3650 κεριών,
- αναλογία μέσου όρου 30 με 90 κεριών,
- αναλογία μέσου όρου 30 με 365 κεριών,
- αναλογία μέσου όρου 90 με 365 κεριών,
- αναλογία τυπικής απόκλισης 5 με 30 κεριών,
- αναλογία τυπικής απόκλισης 5 με 90 κεριών,
- αναλογία τυπικής απόκλισης 30 με 90 κεριών,
- αναλογία τυπικής απόκλισης 30 με 365 κεριών,
- αναλογία τυπικής απόκλισης 30 με 90 κεριών,
- αναλογία τυπικής απόκλισης 30 με 365 κεριών,
- αναλογία τυπικής απόκλισης 90 με 365 κεριών,
- τυπική απόκλιση volume 5 κεριών,
- τυπική απόκλιση volume 30 κεριών,
- τυπική απόκλιση volume 90 κεριών,
- τυπική απόκλιση volume 365 κεριών,

Μετά την εισαγωγή των χαρακτηριστικών με τα οποία θα τροφοδοτηθεί το μοντέλο, καθορίστηκε το εύρος του training set και test set.

```
start_train = datetime.datetime(1971, 1, 4)

end_train = datetime.datetime(2018, 12, 31)

data_train = data.loc[start_train:end_train]
```

Συγκεκριμένα, από το σύνολο των δεδομένων που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής στα πλαίσια της παρούσας διπλωματικής, χρησιμοποιήθηκαν στο training set τα δεδομένα από το 1971 μέχρι το 2018 όπως φαίνεται στο παραπάνω τμήμα κώδικα.

Στο κομμάτι του test set χρησιμοποιήθηκαν τα δεδομένα από το 2019 μέχρι το 2021.

start_test = datetime.datetime(2019, 1, 2)

end_test = datetime.datetime(2021,8,10)

data_test = data.loc[start_test:end_test]

Εφόσον έχουν καθοριστεί τα test set και data set, θα πρέπει τα δεδομένα να τυποποιηθούν για να χρησιμοποιηθούν από τα μοντέλα Machine Learning.

Ο λόγος για τον οποίο πρέπει να τυποποιηθούν τα δεδομένα βασίζεται στο γεγονός ότι οι μεταβλητές που μετρώνται σε διαφορετικές κλίμακες δεν συμβάλλουν εξίσου στη συνάρτηση προσαρμογής και εκμάθησης του μοντέλου και μπορεί να καταλήξουν να δημιουργούν μια προκατάληψη. Έτσι, για την αντιμετώπιση αυτού του δυνητικού προβλήματος, από άποψη χαρακτηριστικών, χρησιμοποιείται συνήθως ένας τυποποιητής με τιμές (μ=0, σ=1) πριν από την χρήση των δεδομένων από το μοντέλο ML.

Για την χρήση ενός τυποποιητή από το scikit-learn, πρέπει πρώτα να κατασκευαστεί ένας πίνακας εισόδου X που να περιέχει δεδομένα με το σχήμα X να είναι [number_of_samples, number_of_features] .

Όλες οι συναρτήσεις μηχανικής εκμάθησης (ML) της scikit-learn, αναμένουν ως είσοδο έναν numpy πίνακα X με αυτό το σχήμα, δηλαδή οι σειρές να είναι τα δείγματα και οι στήλες να είναι τα χαρακτηριστικά/μεταβλητές.

Η κύρια ιδέα είναι να τυποποιούνται τα χαρακτηριστικά-μεταβλητές του πίνακα X με μ = 0 και σ = 1, πριν την χρήση τους σε μοντέλο μηχανικής εκμάθησης. Έτσι, η StandardScaler() θα κανονικοποιήσει τα χαρακτηριστικά, δηλαδή κάθε στήλη του X, μεμονωμένα έτσι ώστε κάθε χαρακτηριστικό να έχει μ = 0 και σ = 1.

Η τύπος που χρησιμοποιείται κατά την χρήση της StandardScaler ενός δείγματος x υπολογίζεται ως:

$$z = (x - u) / s$$

όπου u είναι ο μέσος όρος των δειγμάτων εκπαίδευσης ή μηδέν εάν έχουμε την τιμή (with_mean=False), και s είναι η τυπική απόκλιση των δειγμάτων εκπαίδευσης ή ένα εάν έχουμε (with_std=False).

Το κεντράρισμα και η κλιμάκωση γίνονται ανεξάρτητα σε κάθε χαρακτηριστικό, υπολογίζοντας τα σχετικά στατιστικά στοιχεία για τα δείγματα στο σετ εκπαίδευσης. Ο μέσος όρος και η τυπική απόκλιση αποθηκεύονται στη συνέχεια για να χρησιμοποιηθούν σε μεταγενέστερα δεδομένα χρησιμοποιώντας μετασχηματισμό.

Η τυποποίηση ενός συνόλου δεδομένων είναι μια κοινή απαίτηση για πολλούς εκτιμητές της μηχανικής μάθησης.

Στον παρακάτω κώδικα που ακολουθεί, παραθέτονται οι εντολές της χρήσης της μεθόδου Linear Regression για την δημιουργία του μοντέλου.

```
from sklearn.linear_model import LinearRegression

lin = LinearRegression()

lin.fit(X_scaled_train, y_train)

predictions_lin = lin.predict(X_scaled_test)
```

Στον παραπάνω κώδικα, γίνεται χρήση 3 μεθόδων της LinearRegression, της fit και της predict.

Θα ακολουθήσει η περιγραφή των τριών αυτών μεθόδων όπου οι δύο τελευταίες, χρησιμοποιούνται κατά την δημιουργία όλων των μοντέλων Μηχανικής Μάθησης.

Η μέθοδος fit υλοποιείται από κάθε εκτιμητή και δέχεται μια είσοδο για τα δείγματα δεδομένων (X) και για εποπτευόμενα μοντέλα δέχεται επίσης ένα όρισμα για ετικέτες (δηλαδή δεδομένα στόχου y). Προαιρετικά, μπορεί επίσης να δεχθεί πρόσθετες ιδιότητες δείγματος όπως βάρη.

Οι μέθοδοι fit είναι συνήθως υπεύθυνες για πολλές λειτουργίες. Μία λειτουργία είναι να ξεκινούν διαγράφοντας τυχόν χαρακτηριστικά που είναι ήδη αποθηκευμένα στον εκτιμητή και στη συνέχεια να εκτελούν επικύρωση παραμέτρων και δεδομένων. Είναι επίσης υπεύθυνοι για την εκτίμηση των χαρακτηριστικών από τα δεδομένα εισόδου και για την αποθήκευση των χαρακτηριστικών του μοντέλου όπως επίσης και για την επιστροφή του προσαρμοσμένου εκτιμητή.

Τώρα που έχει εκπαιδευτεί το μοντέλο, το επόμενο βήμα περιλαμβάνει τις προβλέψεις για τις δοκιμές. Οι δοκιμές επιτυγχάνονται με την κλίση της μεθόδου predict() που ουσιαστικά θα χρησιμοποιήσει τις παραμέτρους που έμαθε από την fit() προκειμένου να εκτελέσει προβλέψεις στα δεδομένα δοκιμής.

Ουσιαστικά, η predict() θα εκτελέσει μια πρόβλεψη για κάθε στιγμιότυπο δοκιμής και συνήθως δέχεται μία μόνο είσοδο (X). Για τους ταξινομητές και τους οπισθοδρομητές, η προβλεπόμενη τιμή, θα βρίσκεται στον ίδιο χώρο με αυτόν που εμφανίζεται στο training set. Στους εκτιμητές ομαδοποίησης, η προβλεπόμενη τιμή θα είναι ένας ακέραιος αριθμός. Οι προβλεπόμενες τιμές των παρεχόμενων στιγμιότυπων δοκιμής θα επιστραφούν σε μορφή εξόδου πίνακα ή αραιού πίνακα.

Εάν επιχειρηθεί να εκτελεστεί η predict, χωρίς να εκτελεστεί πρώτα η fit, θα προκληθεί ένα σφάλμα exceptions.NotFittedError.

Στο παραπάνω απόσπασμα κώδικα χρησιμοποιείται ο αλγόριθμος Linear Regression για την δημιουργία του μοντέλου μηχανικής μάθησης.

Η Γραμμική παλινδρόμηση είναι ένας αλγόριθμος μηχανικής μάθησης που βασίζεται στην εποπτευόμενη μάθηση. Εκτελεί μια εργασία παλινδρόμησης και μοντελοποιεί μια τιμή πρόβλεψης στόχου με βάση ανεξάρτητες μεταβλητές. Χρησιμοποιείται κυρίως για την εύρεση της σχέσης μεταξύ των μεταβλητών και της πρόβλεψης. Τα διαφορετικά μοντέλα παλινδρόμησης διαφέρουν ανάλογα με το είδος της σχέσης μεταξύ εξαρτημένων και ανεξάρτητων μεταβλητών, που εξετάζουν και τον αριθμό των ανεξάρτητων μεταβλητών που χρησιμοποιούνται.

Η γραμμική παλινδρόμηση εκτελεί την εργασία πρόβλεψης μιας τιμής εξαρτημένης μεταβλητής (y) με βάση μια δεδομένη ανεξάρτητη μεταβλητή (x). Έτσι, αυτή η τεχνική παλινδρόμησης ανακαλύπτει μια γραμμική σχέση μεταξύ του x (είσοδος) και y (έξοδος).

Συνάρτηση υπόθεσης για Γραμμική παλινδρόμηση :

$y = \theta_1 + \theta_2 x$

Κατά την εκπαίδευση του μοντέλου μας δίνονται:

- x: εισαγωγή δεδομένων εκπαίδευσης (μία μεταβλητή εισόδου (παράμετρος)).
- y: ετικέτες σε δεδομένα (εποπτευόμενη μάθηση).
- Κατά την εκπαίδευση του μοντέλου – ο αλγόριθμος εντοπίζει στην καλύτερη γραμμή για να προβλέψει την τιμή του y για μια δεδομένη τιμή του x. Το μοντέλο βρίσκει την καλύτερη γραμμή προσαρμογής παλινδρόμησης βρίσκοντας τις καλύτερες τιμές θ1 και θ2.
- θ1: παρεμπόδιση.
- θ2: συντελεστής x.
- Μόλις εντοπιστούν οι καλύτερες τιμές θ1 και θ2, προκύπτει η καλύτερη γραμμή προσαρμογής. Έτσι, όταν τελικά χρησιμοποιηθεί το μοντέλο για πρόβλεψη, θα προβλέψει την τιμή του y για την τιμή εισόδου του x.
- Για να εντοπιστεί καλύτερη γραμμή προσαρμογής, μπορούν να ενημερωθούν οι τιμές θ1 και θ2. Με την επίτευξη της καλύτερης προσαρμογής γραμμής παλινδρόμησης, το μοντέλο στοχεύει να προβλέψει την τιμή y έτσι ώστε η διαφορά σφάλματος μεταξύ της προβλεπόμενης τιμής και της πραγματικής τιμής να είναι ελάχιστη. Επομένως, είναι πολύ σημαντικό να ενημερωθούν οι τιμές θ1 και θ2, για να εντοπιστεί, η καλύτερη τιμή που ελαχιστοποιεί το σφάλμα μεταξύ της προβλεπόμενης τιμής y (pred) και της αληθινής τιμής y (y).

$$minimize \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

$$J = \frac{1}{n}\sum_{i=1}^{n}(pred_i - y_i)^2$$

Η συνάρτηση κόστους (J) της Γραμμικής παλινδρόμησης είναι το ριζικό μέσο τετράγωνο σφάλμα (RMSE) μεταξύ της προβλεπόμενης τιμής y (pred) και της αληθινής τιμής y (y).

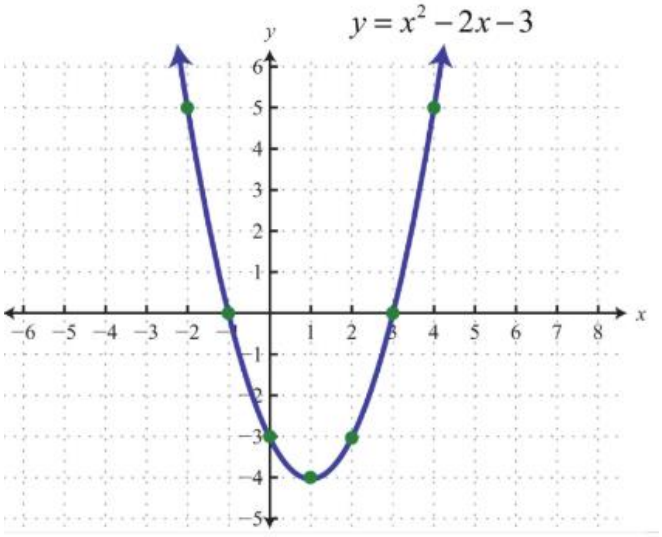
5.2.1 Gradient Descent

Για να ενημερωθούν οι τιμές θ1 και θ2 προκειμένου να μειωθεί η συνάρτηση κόστους (ελαχιστοποίηση της τιμής RMSE) και να επιτευχθεί η καλύτερη γραμμή προσαρμογής, το μοντέλο χρησιμοποιεί το Gradient Descent. Η ιδέα είναι να ξεκινάει με τυχαίες τιμές θ1 και θ2 και στη συνέχεια να ενημερώνει επαναληπτικά τις τιμές, φτάνοντας στο ελάχιστο κόστος.

5.3 Μοντέλο Stochastic Gradient Descent (SGD)

Ο Stochastic gradient descent είναι ένας πολύ δημοφιλής και κοινός αλγόριθμος που χρησιμοποιείται σε διάφορους αλγόριθμους Machine Learning, και το πιο σημαντικό είναι ότι αποτελεί τη βάση των Νευρωνικών Δικτύων.

Gradient, με απλούς όρους σημαίνει κλίση ή κλίση μιας επιφάνειας. Άρα κλίση κατάβασης κυριολεκτικά σημαίνει κατέβασμα μιας κλίσης για να φτάσει στο χαμηλότερο σημείο σε αυτήν την επιφάνεια.



Εικόνα 29. Παραβολή.

Σε ένα δισδιάστατο γράφημα, όπως η παραπάνω παραβολή, το χαμηλότερο σημείο της παραβολής εμφανίζεται στο x = 1. Ο στόχος του αλγόριθμου καθόδου με κλίση είναι να βρει

την τιμή του "x" έτσι ώστε το "y" να είναι ελάχιστο. Το "y" εδώ ορίζεται ως η αντικειμενική συνάρτηση στην οποία λειτουργεί ο αλγόριθμος καθόδου βαθμίδας, για να κατέβει στο χαμηλότερο σημείο.

5.3.1 Gradient Descent - ο αλγόριθμος

Ο στόχος της παλινδρόμησης, είναι να ελαχιστοποιήσει το άθροισμα των τετραγωνικών υπολειμμάτων. Μία συνάρτηση φτάνει την ελάχιστη τιμή της όταν η κλίση της είναι ίση με 0. Χρησιμοποιώντας αυτή την τεχνική, λύθηκε το πρόβλημα της γραμμικής παλινδρόμησης και μάθαμε το διάνυσμα βάρους. Το ίδιο πρόβλημα μπορεί να λυθεί με την τεχνική gradient descent.

"Η κατάβαση κλίσης είναι ένας επαναληπτικός αλγόριθμος, που ξεκινά από ένα τυχαίο σημείο μιας συνάρτησης και διανύει την κλίση της σε βήματα μέχρι να φτάσει στο χαμηλότερο σημείο αυτής της συνάρτησης.".

Αυτός ο αλγόριθμος είναι χρήσιμος σε περιπτώσεις όπου τα βέλτιστα σημεία δεν μπορούν να βρεθούν εξισώνοντας την κλίση της συνάρτησης με 0. Στην περίπτωση της γραμμικής παλινδρόμησης, μπορεί να υπολογιστεί το άθροισμα των υπολοίπων στο τετράγωνο ως συνάρτηση "y" και το διάνυσμα βάρους ως «x» στην παραπάνω παραβολή.

Υπάρχουν μερικά μειονεκτήματα του αλγόριθμου gradient descent. Πρέπει να ρίξουμε μια πιο προσεκτική ματιά στον υπολογισμό που κάνουμε για κάθε επανάληψη του αλγορίθμου.

Ας υποθέσουμε ότι έχουμε 10.000 σημεία δεδομένων και 10 χαρακτηριστικά. Το άθροισμα των υπολειμμάτων στο τετράγωνο αποτελείται από όσους όρους υπάρχουν τα σημεία δεδομένων, άρα 10000 όροι στην περίπτωσή μας. Πρέπει να υπολογίσουμε την παράγωγο αυτής της συνάρτησης σε σχέση με καθένα από τα χαρακτηριστικά, οπότε στην πραγματικότητα θα κάνουμε 10000 * 10 = 100.000 υπολογισμούς ανά επανάληψη. Είναι σύνηθες να κάνουμε 1000 επαναλήψεις και στην πραγματικότητα έχουμε 100.000 * 1000 = 100000000 υπολογισμούς για να ολοκληρώσουμε τις επαναλήψεις στον αλγόριθμο. Αυτό είναι σχεδόν ένα γενικό κόστος και ως εκ τούτου η κλίση είναι αργή σε τεράστια δεδομένα.

Ο αλγόριθμος Stochastic Gradient Descent λύνει το παραπάνω πρόβλημα. Κατά την επιλογή των σημείων δεδομένων σε κάθε βήμα για τον υπολογισμό των παραγώγων, το SGD επιλέγει τυχαία ένα σημείο δεδομένων από ολόκληρο το σύνολο δεδομένων σε κάθε επανάληψη για να μειώσει τους υπολογισμούς.

Είναι επίσης σύνηθες να γίνεται δειγματοληψία ενός μικρού αριθμού σημείων δεδομένων αντί για ένα μόνο σημείο σε κάθε βήμα και αυτό ονομάζεται «mini-batch» gradient descent. Η μίνι παρτίδα προσπαθεί να επιτύχει μια ισορροπία μεταξύ των θετικών στοιχείων του αλγορίθμου Gradient Descent και της ταχύτητας του SGD.

Η παραπάνω θεωρία, αποτέλεσε την βάση για την δημιουργία του μοντέλου SGD όπως παρουσιάζεται στον παρακάτω κώδικα:

```
from sklearn.linear_model import SGDRegressor
param_grid = {
   'penalty':['l1', 'l2', 'elasticnet'],
   "alpha": [1e-5, 3e-5, 1e-4],
   "eta0": [0.01, 0.03, 0.1],
}
sgd = SGDRegressor()
grid_search = GridSearchCV(sgd, param_grid, cv=5, scoring='neg_mean_absolute_error', n_jobs=-1)
grid_search.fit(X_scaled_train, y_train)
sgd_best = grid_search.best_estimator_
predictions_sgd = sgd_best.predict(X_scaled_test)
```

Αρχικά, γίνεται εισαγωγή της βιβλιοθήκης SGDRegressor από το sklearn. Στην συνέχεια καθορίζονται οι παράμετροι penalty, alpha και eta0. Το Penalty είναι μια ποινή που προστίθεται στη συνάρτηση απώλειας, που συρρικνώνει τις παραμέτρους του μοντέλου προς το μηδενικό διάνυσμα, χρησιμοποιώντας είτε την τετραγωνισμένη ευκλείδεια νόρμα L2 είτε την απόλυτη νόρμα L1 ή συνδυασμό και των δύο (Elastic Net). Εάν η ενημέρωση παραμέτρων υπερβεί την τιμή 0,0 λόγω του κανονικοποιητή, η ενημέρωση περικόπτεται σε 0,0 για να επιτρέψει την εκμάθηση αραιών μοντέλων και την επίτευξη διαδικτυακής επιλογής δυνατοτήτων.

Το GridSearchCV εφαρμόζει μια μέθοδο "fit" και "score". Εφαρμόζει επίσης "score_samples", "predict", "predict_proba", "decision_function", "transform" και "inverse_transform" εάν υλοποιούνται στον εκτιμητή που χρησιμοποιείται.

Οι παράμετροι του εκτιμητή που χρησιμοποιείται για την εφαρμογή αυτών των μεθόδων, βελτιστοποιούνται με διασταυρούμενη επικυρωμένη (cross-validated) αναζήτηση πλέγματος σε ένα πλέγμα παραμέτρων.

Η παράμετρος cv, καθορίζει τη στρατηγική διαχωρισμού διασταυρούμενης (cross-validated) επικύρωσης, όπου με την ακέραια τιμή 5, καθορίζεται ο αριθμός των Folds.

Η παράμετρος scoring με την επιλογή «neg_mean_absolute_error»αποτελεί μία στρατηγική, για την αξιολόγηση της απόδοσης του διασταυρούμενου (cross-validated) μοντέλου στο δοκιμαστικό σύνολο.

5.4 Μοντέλο BaggingRegressor (BGR)

Ο Bagging regressor είναι ένας μετα-εκτιμητής συνόλου που προσαρμόζει τους βασικούς αναδρομείς σε τυχαία υποσύνολα του αρχικού συνόλου και στη συνέχεια συγκεντρώνει τις μεμονωμένες προβλέψεις τους (είτε με ψηφοφορία είτε με μέσο όρο) για να σχηματίσει μια τελική πρόβλεψη. Ένας τέτοιος μετα-εκτιμητής μπορεί τυπικά να χρησιμοποιηθεί ως τρόπος μείωσης της διακύμανσης ενός εκτιμητή μαύρου κουτιού (π.χ., ενός δέντρου αποφάσεων), εισάγοντας την τυχαιοποίηση στη διαδικασία κατασκευής του και στη συνέχεια δημιουργώντας ένα σύνολο από αυτό.

Με το παρακάτω απόσπασμα κώδικα γίνεται χρήση του Bagging Regressor με τις παραμέτρους του.

```
bgr = BaggingRegressor(base_estimator=lin, n_estimators=100, oob_score=True, n_jobs=-1)
new_bgr=bgr.fit(X_scaled_train, y_train)
predictions_bgr = bgr.predict(X_scaled_test)
```

Η παράμετρος base_estimator χρησιμοποιείται σαν επιλογή βασικού εκτιμητή που ταιριάζει (fit) σε τυχαία υποσύνολα του συνόλου δεδομένων. Αν δεν έχει επιλεχθεί κανείς, τότε εξορισμού χρησιμοποιείται ο εκτιμητής DecisionTree Regressor.

Η παράμετρος oob_score χρησιμοποιείται για να καθοριστεί εάν θα χρησιμοποιηθούν δείγματα εκτός της συσκευασίας (out of bag) για την εκτίμηση του σφάλματος γενίκευσης.

Με την παράμετρο n_jobs καθορίζεται ο αριθμός των εργασιών που θα εκτελεστούν παράλληλα για προσαρμογή και πρόβλεψη. Η τιμή None=-1 σημαίνει χρήση όλων των επεξεργαστών.

5.5 Μοντέλο XGBoost (XGB)

Το επόμενο μοντέλο μηχανική μάθησης, κατασκευάστηκε με βάση τον αλγόριθμο XGB. Το XGBoost είναι συντομογραφία του "EXtreme Gradient Boosting". Το "eXtreme" αναφέρεται σε βελτιώσεις ταχύτητας, όπως παράλληλος υπολογισμός και επίγνωση της προσωρινής μνήμης που κάνει το XGBoost περίπου 10 φορές πιο γρήγορο από την παραδοσιακή ενίσχυση κλίσης (Gradient Boosting). Επιπλέον, το XGBoost περιλαμβάνει έναν μοναδικό αλγόριθμο εύρεσης διαχωρισμού για τη βελτιστοποίηση των δέντρων, μαζί με ενσωματωμένη τακτοποίηση που μειώνει την υπερπροσαρμογή. Σε γενικές γραμμές, το XGBoost είναι μια πιο γρήγορη και ακριβής έκδοση του Gradient Boosting.

Το Boosting αποδίδει καλύτερα από το bagging κατά μέσο όρο και το Gradient Boosting είναι αναμφισβήτητα ο καλύτερος boosting αλγόριθμος.

Ακολουθεί απόσπασμα κώδικα για την δημιουργία του μοντέλου XBG.

```
xgb_param_grid = {'learning_rate': [0.001, 0.01, 0.1, 1], 'n_estimators': [50, 100, 200, 300],
                  'subsample': [0.3, 0.5, 0.7, 1]}
grid_search = GridSearchCV(estimator=xgb, param_grid=xgb_param_grid,
                           scoring='neg_mean_squared_error', cv=4, verbose=1,  n_jobs=-1)
grid_search.fit(X_train, y_train)
xgb_best = grid_search.best_estimator_
xgb_best.fit(X_train,y_train)
predictions_xgb = xgb_best.predict(X_test)
```

Στον παραπάνω κώδικα, η παράμετρος learning_rate καθορίζει τον ρυθμό μάθησης. Συγκεκριμένα συρρικνώνει τη συνεισφορά κάθε δέντρου με το Learning_rate. Υπάρχει μία αντιστάθμιση μεταξύ Learning_rate και n_estimators.

Με την παράμετρο n_estimators δηλώνεται ο αριθμός των σταδίων ενίσχυσης προς εκτέλεση. Η ενίσχυση κλίσης (Gradinet Boosting) είναι αρκετά ανθεκτική στην υπερτροφοδότηση δεδομένων, επομένως ένας μεγάλος αριθμός συνήθως οδηγεί σε καλύτερη απόδοση.

Η παράμετρος subsample ερμηνεύεται σαν το κλάσμα των δειγμάτων που θα χρησιμοποιηθούν για την προσαρμογή των μεμονωμένων βασικών μαθητευόμενων. Εάν είναι μικρότερο από 1,0, αυτό έχει ως αποτέλεσμα την ενίσχυση της στοχαστικής κλίσης. Το μέρος του δείγματος αλληλεπιδρά με την παράμετρο n_estimators. Η επιλογή υποδείγματος < 1,0 οδηγεί σε μείωση της διακύμανσης και αύξηση της μεροληψίας.

Εφόσον καθοριστούν οι παράμετροι στην μέθοδο GridSearchCV (όπως περιεγράφηκε στην ενότητα SGD), εκτελούνται οι μέθοδοι fit και predict για να γίνει πρόβλεψη με βάση το μοντέλο που δημιουργήθηκε.

5.6 Random Forest (RF)

Το μοντέλο μηχανικής μάθησης που ακολουθεί, είναι βασισμένο στον αλγόριθμο Random Forest. Ο Random Forest ή Random Decision Forest είναι ένας εποπτευόμενος αλγόριθμος μηχανικής μάθησης, που χρησιμοποιείται για ταξινόμηση, παλινδρόμηση και άλλες εργασίες που χρησιμοποιούν δέντρα αποφάσεων.

Ο ταξινομητής Random Forest δημιουργεί ένα σύνολο δέντρων αποφάσεων από ένα τυχαία επιλεγμένο υποσύνολο του συνόλου εκπαίδευσης. Είναι βασικά ένα σύνολο δέντρων αποφάσεων (DT), από ένα τυχαία επιλεγμένο υποσύνολο του συνόλου εκπαίδευσης και στη συνέχεια συλλέγει τους ψήφους από διαφορετικά δέντρα αποφάσεων για να αποφασίσει την τελική πρόβλεψη.

Ακολουθεί απόσπασμα κώδικα από την δημιουργία του συγκεκριμένου μοντέλου.

```
param_grid = {"max_depth": [30, 50],     "min_samples_split": [5, 10, 20], }
rf = RandomForestRegressor(n_estimators=100)
grid_search = GridSearchCV(rf, param_grid, cv=5, scoring='neg_mean_absolute_error', n_jobs=-1)
grid_search.fit(X_train, y_train)
print(grid_search.best_params_)
# print(grid_search.best_score_)
rf_best = grid_search.best_estimator_
predictions_rf = rf_best.predict(X_test)
```

Ομοίως με τα μοντέλα SGD και XGB χρησιμοποιήθηκε ένα σύνολο παραμέτρων που καθορίζονται από το param_grid. Συγκεκριμένα η παράμετρος max_depth δηλώνει το μέγιστο βάθος του δέντρου. Αν δεν καθορίζεται, τότε οι κόμβοι επεκτείνονται μέχρι να γίνουν όλα τα φύλλα καθαρά ή έως ότου όλα τα φύλλα περιέχουν λιγότερα από τον αριθμό των min_samples_split δειγμάτων.

Η παράμετρος min_samples_split δηλώνει τον ελάχιστο αριθμό δειγμάτων που απαιτούνται, για τον διαχωρισμό ενός εσωτερικού κόμβου.

Μετά τον καθορισμό των παραμέτρων, χρησιμοποιήθηκαν οι μέθοδοι GridSearchCV, fit και predict για να μπορέσει το μοντέλο να κάνει την πρόβλεψη του.

6

Αξιολόγηση

Στην παρούσα διπλωματική δημιουργήθηκαν μοντέλα μηχανικής μάθησης με αλγορίθμους όπως BGR, Linear, Regression, Random, Forest, SGD, XGB και LSTM. Για κάθε μοντέλο, χρησιμοποιήθηκαν ίδιοι παράμετροι αξιολόγησης για να υπάρχει μια κοινή βάση αξιολόγησης της απόδοσής τους.

6.1 Παράμετροι αξιολόγησης

Για να μπορέσει να προσδιοριστεί κατά πόσο ένα μοντέλο μηχανικής είναι ακριβές, γίνεται χρήση συγκεκριμένων μετρικών για να ποσοτικοποιηθεί το πόσο ακριβές είναι.

Για κάθε μοντέλο που δημιουργήθηκε υπολογίστηκαν τα errors mae (mean absolute error), mse (mean squared error) και r2 score.

6.1.1 R2 Score

Η βαθμολογία R2 (R2 score) είναι ένα από τα μέτρα αξιολόγησης απόδοσης για μοντέλα μηχανικής μάθησης που βασίζονται σε παλινδρόμηση.

Πρόκειται για μια πολύ σημαντική μέτρηση που χρησιμοποιείται για την αξιολόγηση της απόδοσης ενός μοντέλου μηχανικής μάθησης που βασίζεται σε παλινδρόμηση. Προφέρεται ως R τετράγωνο και είναι επίσης γνωστός ως συντελεστής προσδιορισμού. Λειτουργεί με τη μέτρηση του ποσού της διακύμανσης στις προβλέψεις που εξηγούνται από το σύνολο δεδομένων. Στην ουσία, πρόκειται για την διαφορά μεταξύ των δειγμάτων στο σύνολο δεδομένων και των προβλέψεων που γίνονται από το μοντέλο.[28]

Η καλύτερη δυνατή βαθμολογία είναι 1,0 και μπορεί να είναι αρνητική (επειδή το μοντέλο μπορεί να είναι αυθαίρετα χειρότερο). Ένα σταθερό μοντέλο που προβλέπει πάντα την αναμενόμενη τιμή του y, αγνοώντας τα χαρακτηριστικά εισόδου, θα έπαιρνε βαθμολογία R^2 0,0.

Αν το $\hat{y}_i$ είναι η προβλεπόμενη τιμή του i-ου δείγματος και yi είναι η αντίστοιχη πραγματική τιμή για τα συνολικά n δείγματα, το εκτιμώμενο R^2 ορίζεται ως:

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ and $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} \epsilon_i^2$.

6.1.2 MSE (Mean Squared Error)

Η συνάρτηση mean_squared_error, υπολογίζει το μέσο τετραγωνικό σφάλμα. Πρόκειται για μια μέτρηση κινδύνου που αντιστοιχεί στην αναμενόμενη τιμή του τετραγωνικού (τετραγωνικού) σφάλματος ή απώλειας.[28]

Όσο μεγαλύτερος είναι ο αριθμός, τόσο μεγαλύτερο είναι το σφάλμα. Σφάλμα σε αυτή την περίπτωση σημαίνει τη διαφορά μεταξύ των παρατηρούμενων τιμών y1, y2, y3, … και των προβλεπόμενων pred(y1), pred(y2), pred(y3), … Υψώνεται στο τετράγωνο κάθε διαφορά $(\text{pred}(yn) - yn)$ $)^2$ ώστε οι αρνητικές και οι θετικές τιμές να μην αλληλοεξουδετερώνονται.

Αν $\hat{y}_i$ είναι η προβλεπόμενη τιμή του i-ου δείγματος, και y_i είναι η αντίστοιχη αληθινή τιμή, τότε το μέσο τετραγωνικό σφάλμα (MSE) υπολογίζεται πάνω σε n δείγματα ως

$$\text{MSE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2.$$

6.1.3 MAE (Mean absolute error)

Η συνάρτηση mean_absolute_error υπολογίζει το μέσο απόλυτο σφάλμα όπου αντιπροσωπεύει μια μέτρηση κινδύνου που αντιστοιχεί στην αναμενόμενη τιμή της απόλυτης απώλειας σφάλματος ή L1- απώλεια κανόνων.[28]

Αν $\hat{y}_i$ είναι η προβλεπόμενη τιμή του i-ου δείγματος, και y_i είναι η αντίστοιχη αληθινή τιμή, τότε το μέσο απόλυτο σφάλμα (MAE) υπολογίζεται πάνω σε n δείγματα ως

$$\text{MAE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} |y_i - \hat{y}_i| .$$

6.2 Οργάνωση πειραμάτων

Τα δεδομένα που χρησιμοποιήθηκαν για την εκπαίδευση των μοντέλων συγκεντρώθηκαν από την πλατφόρμα Metatrader5 και από την πλατφόρμα Kaggle. Συγκεκριμένα από το 1971 μέχρι το 2010 χρησιμοποιήθηκαν τα δεδομένα από την Kaggle και τα υπόλοιπα προστέθηκαν από την πλατφόρμα Metatrader5. Τα δεδομένα απαρτίζουν κεριά ημέρας συνολικά από το 1971 μέχρι το 2021 για όλες τις καθημερινές χωρίς να υπολογίζονται το Σάββατο και η Κυριακή.

Τα δεδομένα κάθε κεριού ημέρας έχουν σαν χαρακτηριστικά τις τιμές ανοίγματος, κλεισίματος, χαμηλού, υψηλού και όγκου συναλλαγών (volume) της κάθε ημέρας. Η αγορά Forex συναλλάγματος λειτουργεί από την Δευτέρα μέχρι και την Παρασκευή 24ώρες το 24ώρο.

Η τιμή ανοίγματος αναφέρεται στην πρώτη τιμή που καταγράφεται στο κερί ημέρας. Αντίστοιχα η τιμή κλεισίματος είναι η τελευταία τιμή του κεριού, πριν αλλάξει η ημέρα. Το υψηλό και χαμηλό ημέρας, αναφέρονται στην υψηλότερη τιμή και την χαμηλότερη τιμή αντίστοιχα που καταγράφτηκε κατά την διάρκεια της ημέρας.

Ο όγκος συναλλαγών (volume) είναι ο συνολικός αριθμός μίας ισοτιμίας που διαπραγματεύτηκε κατά τη διάρκεια μιας δεδομένης χρονικής περιόδου (στην προκειμένη περίπτωση ημέρας). Ο όγκος συναλλαγών δεν υποδεικνύει απλώς πόσες συναλλαγές πραγματοποιούνται, αλλά περιλαμβάνει επίσης το συνολικό ποσό της ισοτιμίας που αγοράστηκε ή πωλήθηκε κατά τη διάρκεια της συναλλαγής.

Συνήθως η ημερήσια αναφορά γίνεται μετά το κλείσιμο της αγοράς και είναι πιο ακριβής σε σχέση με μικρότερα time frames όπως π.χ ώρας.

Ο όγκος είναι σημαντικός επειδή είναι στενά συνδεδεμένος με τη ρευστότητα, η οποία έχει άμεσο αντίκτυπο στην ικανότητα του trader να ανοίγει και να κλείνει θέσεις γρήγορα και στην επιθυμητή τιμή. Τα αποτελέσματα της έντασης αλλάζουν ανάλογα με το αν η ένταση είναι υψηλή ή χαμηλή.

Ο υψηλός όγκος δείχνει ότι υπάρχουν πολλοί traders στην αγορά. Αν και αυτό δεν σημαίνει απαραίτητα ότι κάθε trader θα ανοίξει εντολές στις ίδιες θέσεις αλλά συνήθως δημιουργείται μια τάση. Συνήθως, εάν ο όγκος των συναλλαγών αυξηθεί, τότε η τιμή της αγοράς κινείται γενικά προς την ίδια κατεύθυνση – αλλά αυτό δεν σημαίνει απαραίτητα ότι ο υψηλός όγκος ισούται με υψηλές τιμές, επειδή μπορεί να οι traders να ανοίξουν εντολές πτώσης. Σε περιόδους υψηλού όγκου, οι τιμές τείνουν να αλλάζουν πιο γρήγορα λόγω του τεράστιου αριθμού ατόμων που αγοράζουν και πουλάνε συνάλλαγμα.

Ο χαμηλός όγκος υποδηλώνει ότι υπάρχουν λιγότεροι αγοραστές και πωλητές στην αγορά, γεγονός που μεταφράζεται σε λιγότερη ρευστότητα. Η χαμηλή ρευστότητα σημαίνει ότι

υπάρχει ρίσκο καθώς μπορεί οι traders να κολλήσουν σε ανοιχτές θέσεις και να μην μπορούν να κλείσουν τις εντολές.

Ενδεικτικά τα δεδομένα της αρχής και του τέλους του data set:

eu.head()

	open	high	low	close	volume
date					
1971-01-04	0.5369	0.5369	0.5369	0.5369	1
1971-01-05	0.5366	0.5366	0.5366	0.5366	1
1971-01-06	0.5365	0.5365	0.5365	0.5365	1
1971-01-07	0.5368	0.5368	0.5368	0.5368	1
1971-01-08	0.5371	0.5371	0.5371	0.5371	1

Πίνακας 5. 5 πρώτες εγγραφές του πινακα.

eu.tail()

	open	high	low	close	volume
date					
2020-12-15	1.12575	1.12992	1.12217	1.12919	71224
2020-12-16	1.12905	1.13602	1.12811	1.13296	72728
2020-12-17	1.13284	1.13489	1.12344	1.12372	63263
2020-12-20	1.12356	1.13040	1.12341	1.12769	60738
2020-12-21	1.12743	1.13027	1.12609	1.12822	54420

Πίνακας 6. 5 τελευταίες εγγραφές του πίνακα.

6.3 Αποτελέσματα

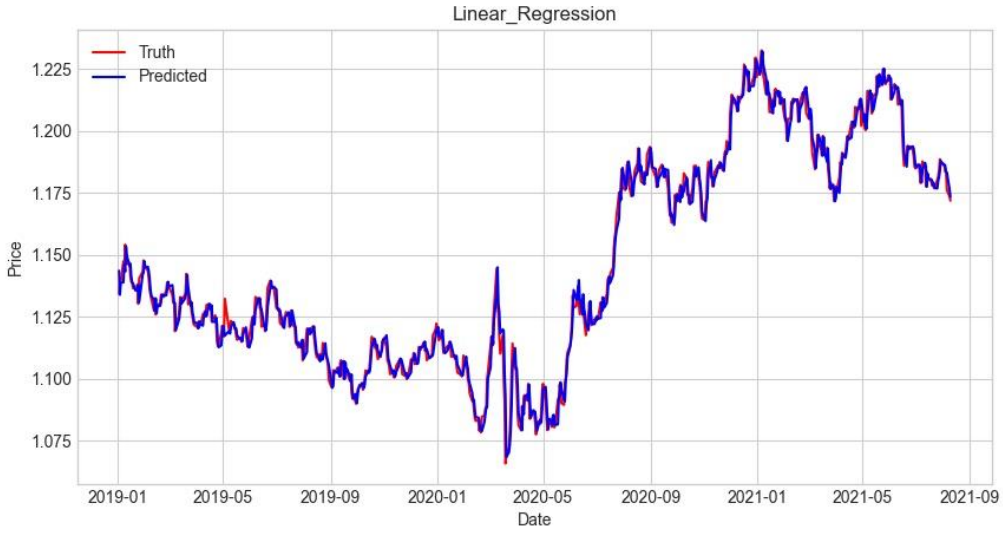
Στο συγκεκριμένο κεφάλαιο θα παρουσιαστούν τα αποτελέσματα του κάθε μοντέλου μηχανικής μάθησης. Για κάθε μοντέλο θα παρατεθούν η γραφική αναπαράστασης σύγκρισης της προβλεπόμενης τιμής σε αντιπαραβολή με την πραγματική τιμή και ο πίνακας error.

Αρχικά παραθέτονται τα αποτελέσματα του μοντέλου Linear Regression:

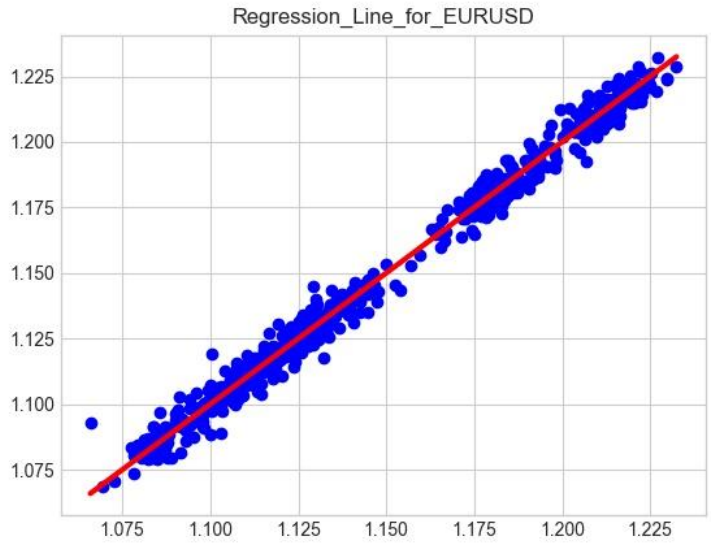
Linear Regrssion

MSE	MAE	R2
0,00002	0,003	0,989

Πίνακας 7. Error Linear Regression



Σχεδιάγραμμα 1. Linear_Regression vs πραγματική τιμή.



Σχεδιάγραμμα 2. Regression Line για το μοντέλο Linear Regression.

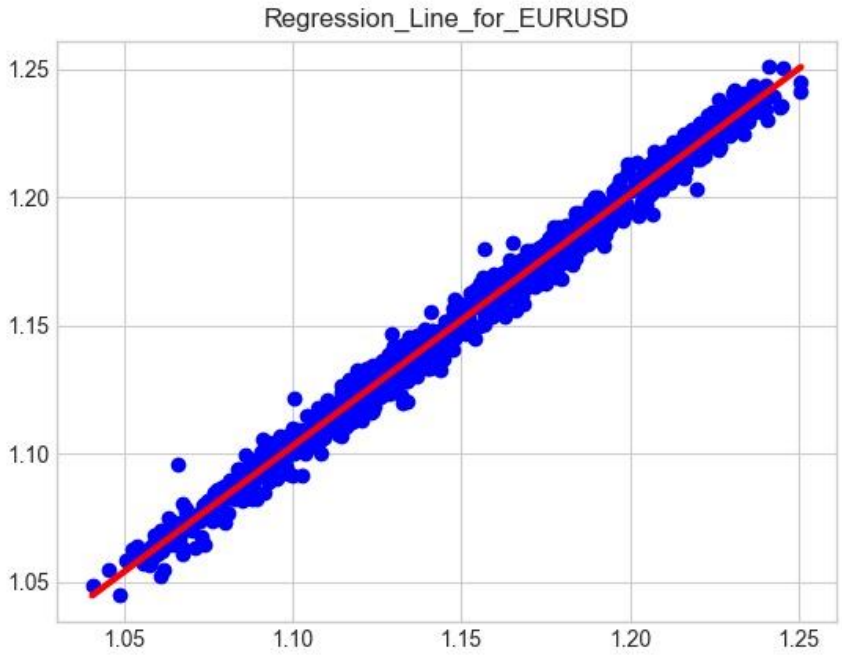
Ακολουθούν τα αποτελέσματα του μοντέλου BGR:

BGR

MSE	MAE	R2
0,00002	0,00378	0,98812

Πίνακας 8.Error Linear Regression.

Σχεδιάγραμμα 9. BGR vs πραγματική τιμή.



Σχεδιάγραμμα 10. Regression Line για το μοντέλο BGR

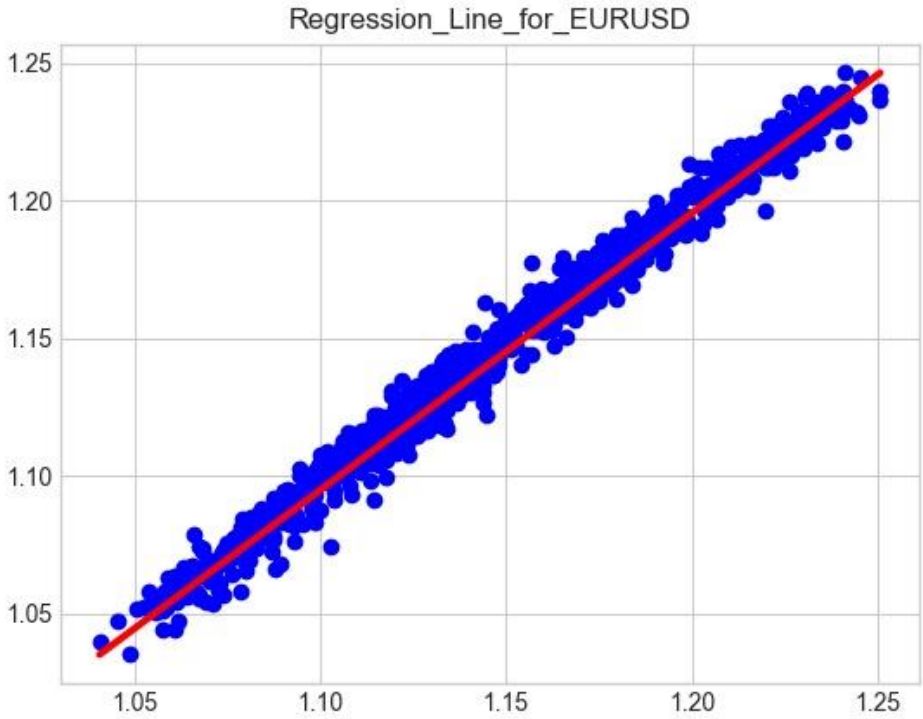
Αποτελέσματα μοντέλου SGD:

SGD

MSE	MAE	R2
0,00002	0,004	0,985

Πίνακας 9. Error Linear Regression

Σχεδιάγραμμα 11. SGD vs πραγματική τιμή



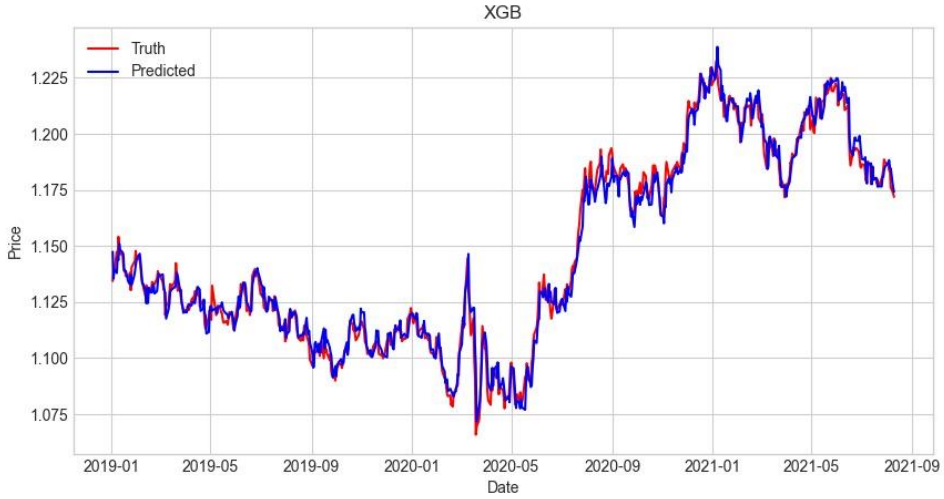
Σχεδιάγραμμα 12. Regression Line για το μοντέλο SGD.

Αποτελέσματα μοντέλου XGB:

XGB

MSE	MAE	R2
0,00002	0,003	0,988

Πίνακας 10. Error XGB.



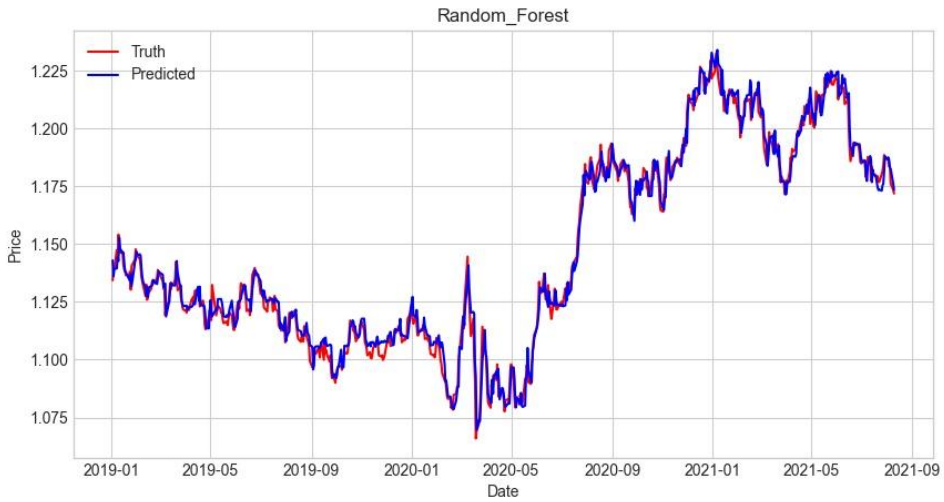
Σχεδιάγραμμα 13. XGB vs πραγματική τιμή

Αποτελέσματα μοντέλου Random Forest:

Random Forest

MSE	MAE	R2
0,00003	0,004	0,986

Πίνακας 11. Error Random Forest



Σχεδιάγραμμα 14. Random Forest vs πραγματική τιμή.

6.4 Σύνοψη συμπερασμάτων αξιολόγησης

Βάση των αποτελεσμάτων τον μοντέλων μηχανικής μάθησης που δημιουργήθηκαν στα πλαίσια της διπλωματικής εργασίας, προκύπτει το συμπέρασμα ότι τα αποτελέσματα όλων των μοντέλων είναι αρκετά κοντά με κάποιες διαφοροποιήσεις.

Συγκεντρωτικός πίνακας error

	MSE	MAE	R2
LIN	0,00002	0,003	0,989
RF	0,00003	0,004	0,986
XGB	0,00002	0,003	0,988
SGD	0,00002	0,004	0,985
BGR	0,00002	0,003	0,988
LSTM	0,00004	0,004	0,977

Πίνακας 12. Συγκεντρωτικός πίνακας error.

Από τον συγκεντρωτικό πίνακα αποτελεσμάτων των μοντέλων, προκύπτει ότι όλα τα μοντέλα παρουσιάζουν παρόμοια απόδοση. Αξίζει να επισημανθεί ότι κάθε φορά που τρέχει ένα μοντέλο με updated δεδομένα κεριών ημέρας, τα αποτελέσματα διαφοροποιούνται ελάχιστα. Το εύρος τιμών error για την τιμή του mean squared error κυμαίνεται από 0,00002 μέχρι 0,00005.

Σε επίπεδο ανοίγματος εντολών Forex συναλλάγματος, τα αποτελέσματα θεωρούνται αρκετά καλά αλλά χρήζουν περαιτέρω βελτίωσης για να θεωρηθούν αξιόλογα. Για το άνοιγμα εντολών μόνο η προβλεπόμενη τιμή δεν είναι αρκετή για να καθοριστεί το σημείο ανοίγματος και κλεισίματος των εντολών, παρόλα αυτά αποτελεί ένα πολύ σημαντικό δεδομένο ιδίως αν έχει μεγάλο ποσοστό επιτυχίας πρόβλεψης.

7

Τεχνικές λεπτομέρειες

Στο συγκεκριμένο κεφάλαιο θα ακολουθήσουν οι τεχνικές λεπτομέρειες υλοποίησης του project που περιεγράφηκε στα προηγούμενα κεφάλαια. Θα δοθούν λεπτομέρειες των προγραμμάτων, των βιβλιοθηκών και των πλατφόρμων που χρησιμοποιήθηκαν στα πλαίσια της παρούσας εργασίας. Θα γίνει αναφορά των γλωσσών προγραμματισμού που χρησιμοποιήθηκαν για το frond-end και το back-end κομμάτι της εφαρμογής που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας.

Για την υλοποίηση του Frond-end τμήματος ενός webapp χρησιμοποιούνται γλώσσες προγραμματισμού διεπαφής για να δημιουργηθεί αυτό που βλέπει ο τελικός χρήστης σε ένα πρόγραμμα περιήγησης. Για την ανάπτυξη του Back-end, χρησιμοποιούνται γλώσσες προγραμματισμού για την εκπλήρωση των αιτημάτων που γίνονται προς την πλευρά του διακομιστή.

7.1 Πλατφόρμες και προγραμματιστικά εργαλεία

Η υλοποίηση της εφαρμογής επιτεύχθηκε με την χρήση του Django Framework και την χρήση πληθώρας βιβλιοθηκών python, οι οποίες θα αναφερθούν στην συνέχεια του τρέχοντος κεφαλαίου. Το κομμάτι ανάπτυξης των μοντέλων μηχανική μάθησης υλοποιήθηκε στην πλατφόρμα Kaggle η οποία δίνει επεξεργαστική ισχύει αρκετά ανώτερη από ένα υπολογιστή home user.

7.1.1 Django Framework

Το Django είναι ένα δωρεάν και ανοιχτού κώδικα πλαίσιο εφαρμογών, που βασίζεται σε Python για την ανάπτυξη backend εφαρμογών ιστού και ιστοσελίδων.

Ακολουθεί το αρχιτεκτονικό μοτίβο Model View Template (MVT). Διαχωρίζει τον κώδικα σε τρία διαφορετικά μέρη - Μοντέλο, Προβολή και Πρότυπα.

- Μοντέλο: Το μοντέλο πρόκειται να λειτουργήσει ως διεπαφή των δεδομένων. Αποτελεί την λογική δομή δεδομένων πίσω από ολόκληρη την εφαρμογή και αντιπροσωπεύεται από μια βάση δεδομένων (γενικά σχεσιακές βάσεις δεδομένων όπως MySql, Postgres).
- Προβολή: Η προβολή είναι η διεπαφή χρήστη, αυτό που βλέπουμε στο πρόγραμμα περιήγησής κατά την επίσκεψη σε έναν ιστότοπο. Αντιπροσωπεύεται από αρχεία HTML/CSS/Javascript.
- Πρότυπο: Ένα πρότυπο αποτελείται από στατικά μέρη της επιθυμητής εξόδου HTML καθώς και από κάποια ειδική σύνταξη που περιγράφει πώς θα εισαχθεί δυναμικό περιεχόμενο.

Οι προγραμματιστές πρέπει να κωδικοποιήσουν τι πρέπει να εμφανίζεται στον χρήστη και το Django θα φροντίσει για όλες τις λεπτομέρειες του φόντου.

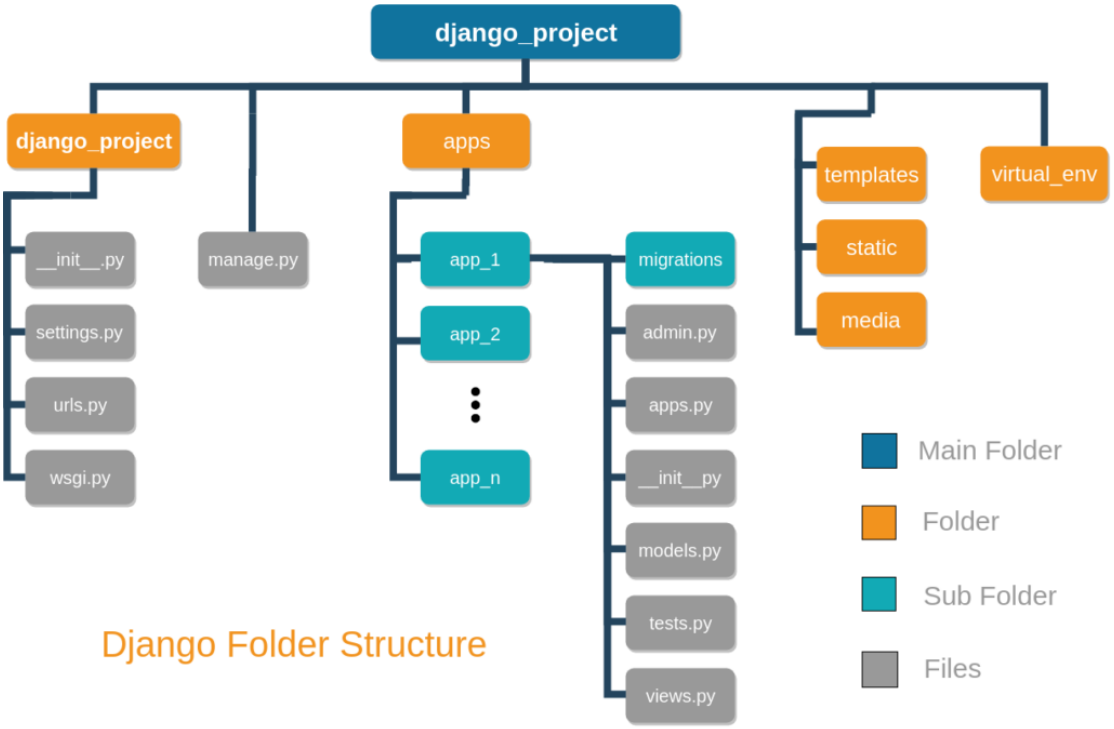
Ο πρωταρχικός στόχος του Django είναι να διευκολύνει τη δημιουργία πολύπλοκων ιστότοπων που βασίζονται σε βάσεις δεδομένων.

Το πλαίσιο web Python Django δίνει έμφαση στους ακόλουθους βασικούς τομείς:

- Επαναχρησιμοποίηση
- Συνδεσιμότητα εξαρτημάτων
- Λιγότερος κωδικός
- Χαμηλή σύζευξη
- Γρήγορη ανάπτυξη

Έχει ένα ευρύ φάσμα χρήσης, που εξαπλώνεται από τις εφαρμογές ηλεκτρονικού εμπορίου και μπορεί να χρησιμοποιηθεί για εφαρμογές μηχανικής μάθησης, για κινητά, την ανάπτυξη παιχνιδιών.

- Το mlapp/models.py είναι το μοντέλο όπου ορίζετε η βάση δεδομένων. Στην τρέχουσα εφαρμογή οι πληροφορίες εγγραφής του χρήστη της εφαρμογής, αποθηκεύονται σε βάση δεδομένων.
- Το mlapp/views.py είναι ο ελεγκτής. Μέσα στο views.py ορίζονται διάφορες συναρτήσεις και κλάσεις. Το Django προσφέρει υποστήριξη και για τα δύο. Μια συνάρτηση ορίζει τι συμβαίνει όταν δίνεται πρόσβαση σε μια συγκεκριμένη διεύθυνση URL και γίνεται αίτημα HTTP στον διακομιστή.
- Τα πάντα κάτω από το mlapp/templates/mlapp/ είναι πρότυπα (tempaltes) ή αρχεία HTML που ορίζουν την προβολή της εφαρμογής. Όταν μια συνάρτηση στο views.py κάνει render ένα αρχείο HTML, μπορεί να μεταβιβάσει αντικείμενα όπως μια λίστα η οποία μπορεί να εμφανιστεί με ειδική σύνταξη που χρησιμοποιείται στο Django framework. Μέσα σε κάθε template, μπορούν να χρησιμοποιηθούν στατικά αρχεία όπως CSS, αρχεία Javascript ή εικόνες που θα υποστηρίζουν την ιστοσελίδα - εφαρμογή.
- Το urls.py είναι το αρχείο διαμόρφωσης URL. Αυτό είναι το αρχείο επιτρέπει να αντιστοιχηθεί μια συγκεκριμένη διεύθυνση URL σε μια συγκεκριμένη συνάρτηση στο views.py. Ας υποθέσουμε ότι θέλουμε να συνδεθούμε σε έναν διακομιστή που λαμβάνει αιτήματα από τον τομέα mydomain.com. Εάν μεταβούμε στο mydomain.com/about-us, μια συγκεκριμένη συνάρτηση όπως η aboutus() θα δημιουργήσει μια απάντηση.



Εικόνα 30. Δομή Django Application.

7.1.2 Εγκατάσταση Djano - Pycharm IDE

Το PyCharm είναι ένα IDE πολλαπλών πλατφορμών (cross-platform) IDE που υποστηρίζεται στα λειτουργικά συστήματα Windows, macOS και Linux.

Το PyCharm είναι διαθέσιμο σε τρεις εκδόσεις: Professional, Community και Edu. Οι εκδόσεις Community και Edu είναι έργα ανοιχτού κώδικα και είναι δωρεάν, αλλά έχουν λιγότερες δυνατότητες. Το PyCharm Edu παρέχει μαθήματα και βοηθάει στην εκμάθηση προγραμματισμό με την Python. Η έκδοση Professional είναι εμπορική και παρέχει ένα εξαιρετικό σύνολο εργαλείων και λειτουργιών.

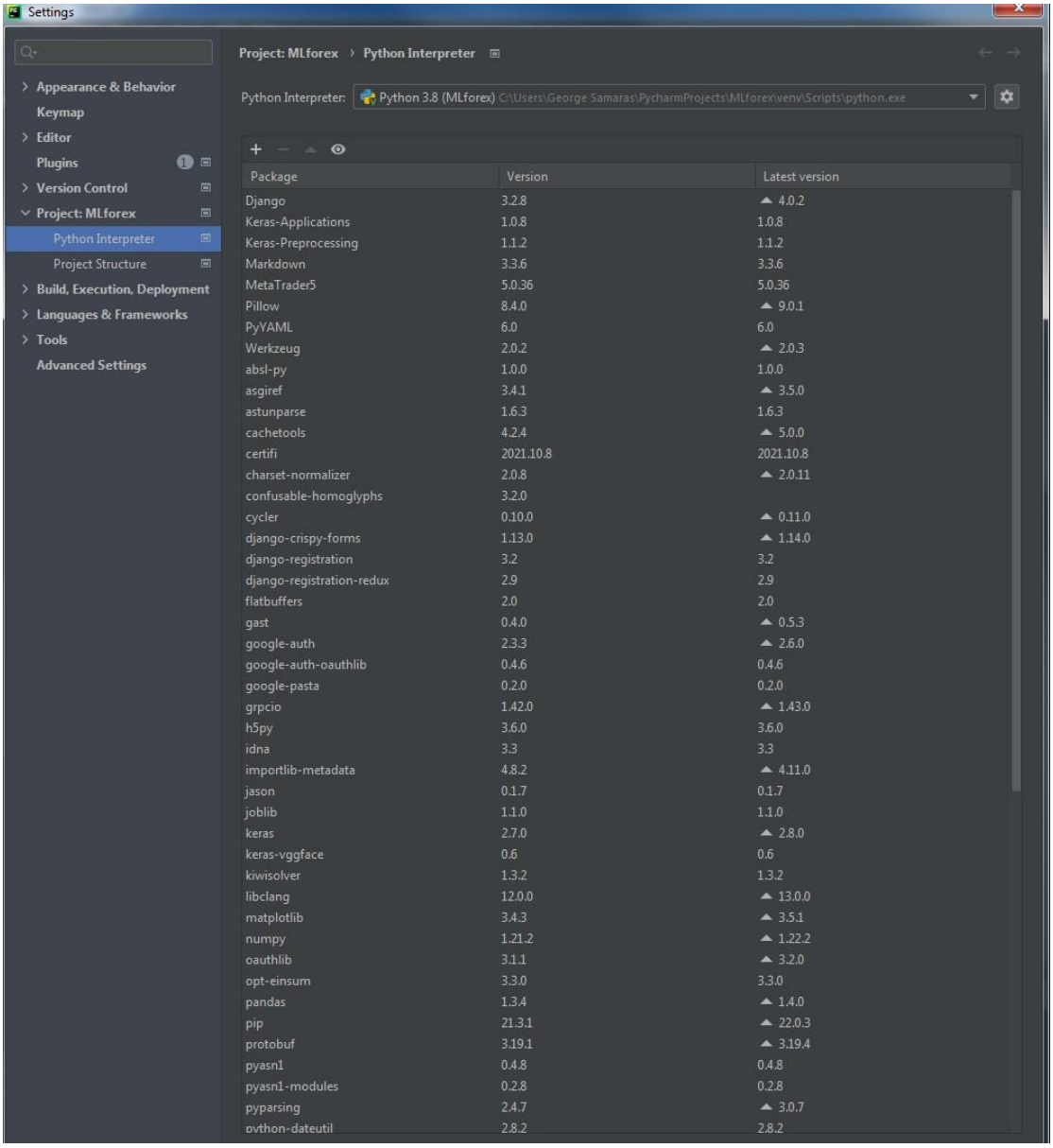
System requirements

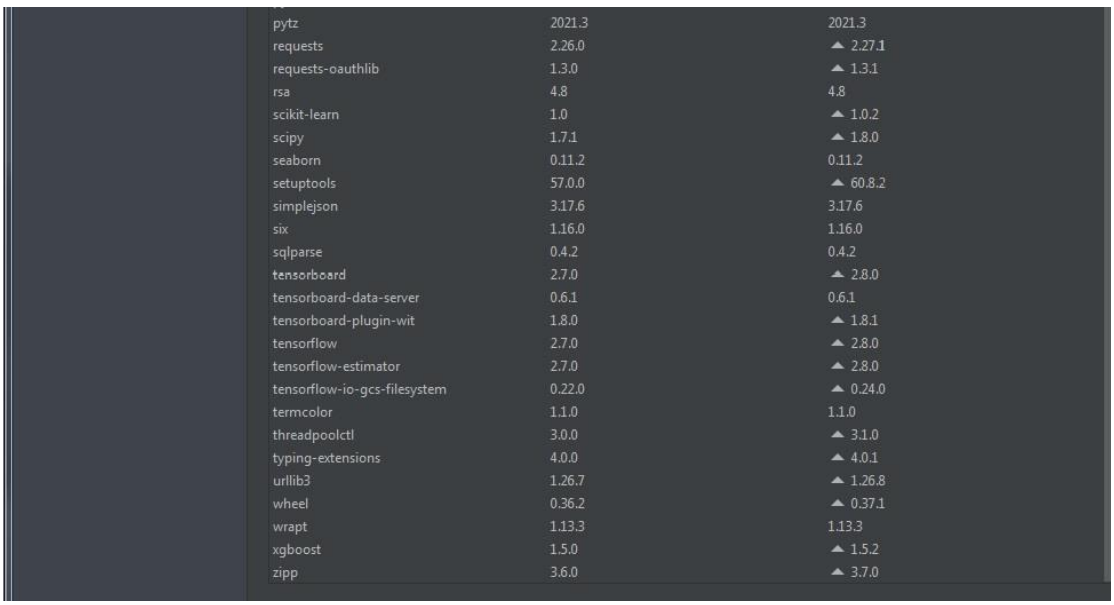
Requirement	Minimum	Recommended
RAM	4 GB of free RAM	8 GB of total system RAM
CPU	Any modern CPU	Multi-core CPU. PyCharm supports multithreading for different operations and processes making it faster the more CPU cores it can use.
Disk space	2.5 GB and another 1 GB for caches	SSD drive with at least 5 GB of free space
Monitor resolution	1024x768	1920×1080
Operating system	Officially released 64-bit versions of the following: • Microsoft Windows 8 or later • macOS 10.14 or later • Any Linux distribution that supports Gnome, KDE , or Unity DE. PyCharm is not available for some Linux distributions, such as RHEL6 or CentOS6, that do not include GLIBC ↗ 2.14 or later. Pre-release versions are not supported.	Latest 64-bit version of Windows, macOS, or Linux (for example, Debian, Ubuntu, or RHEL)

Εικόνα 31 .Requirements.

Για την επιτυχή εγκατάσταση του PyCharm IDE πρέπει να ικανοποιούνται οι παραπάνω απαιτήσεις. Το μηχάνημα στο οποίο έγινε εγκατάσταση, έχει Windows 10, επεξεργαστή i7 με 12GB RAM με ssd 256GB. Για την εγκατάσταση χρησιμοποιήθηκε το αρχείο exe που κατέβηκε από την επίσης σελίδα της jetbrains https://www.jetbrains.com/pycharm/download/#section=windows .

Η εγκατάσταση του Django Framework έγινε μέσα από το PyCharm IDE. Μέσα από το path file->settings->Python interpreter κάνοντας κλικ στο κουμπί + έγινε η εγκατάσταση των βιβλιοθηκών που χρειάστηκαν για να δημιουργηθεί η εφαρμογή που περιγράφτηκε στα προηγούμενα κεφάλαια. Η λίστα των βιβλιοθηκών απεικονίζεται στις παρακάτω εικόνες.


Settings
Project: MLforex › Python Interpreter
Appearance & Behavior
Keymap
Editor
Plugins
Version Control
Project: MLforex
Python Interpreter
Project Structure
Build, Execution, Deployment
Languages & Frameworks
Tools
Advanced Settings
Python Interpreter: Python 3.8 (MLforex) C:\Users\George Samaras\PycharmProjects\MLforex\venv\Scripts\python.exe
Package | Version | Latest version
Django | 3.2.8 | ▲ 4.0.2
Keras-Applications | 1.0.8 | 1.0.8
Keras-Preprocessing | 1.1.2 | 1.1.2
Markdown | 3.3.6 | 3.3.6
MetaTrader5 | 5.0.36 | 5.0.36
Pillow | 8.4.0 | ▲ 9.0.1
PyYAML | 6.0 | 6.0
Werkzeug | 2.0.2 | ▲ 2.0.3
absl-py | 1.0.0 | 1.0.0
asgiref | 3.4.1 | ▲ 3.5.0
astunparse | 1.6.3 | 1.6.3
cachetools | 4.2.4 | ▲ 5.0.0
certifi | 2021.10.8 | 2021.10.8
charset-normalizer | 2.0.8 | ▲ 2.0.11
confusable-homoglyphs | 3.2.0
cycler | 0.10.0 | ▲ 0.11.0
django-crispy-forms | 1.13.0 | ▲ 1.14.0
django-registration | 3.2 | 3.2
django-registration-redux | 2.9 | 2.9
flatbuffers | 2.0 | 2.0
gast | 0.4.0 | ▲ 0.5.3
google-auth | 2.3.3 | ▲ 2.6.0
google-auth-oauthlib | 0.4.6 | 0.4.6
google-pasta | 0.2.0 | 0.2.0
grpcio | 1.42.0 | ▲ 1.43.0
h5py | 3.6.0 | 3.6.0
idna | 3.3 | 3.3
importlib-metadata | 4.8.2 | ▲ 4.11.0
jason | 0.1.7 | 0.1.7
joblib | 1.1.0 | 1.1.0
keras | 2.7.0 | ▲ 2.8.0
keras-vggface | 0.6 | 0.6
kiwisolver | 1.3.2 | 1.3.2
libclang | 12.0.0 | ▲ 13.0.0
matplotlib | 3.4.3 | ▲ 3.5.1
numpy | 1.21.2 | ▲ 1.22.2
oauthlib | 3.1.1 | ▲ 3.2.0
opt-einsum | 3.3.0 | 3.3.0
pandas | 1.3.4 | ▲ 1.4.0
pip | 21.3.1 | ▲ 22.0.3
protobuf | 3.19.1 | ▲ 3.19.4
pyasn1 | 0.4.8 | 0.4.8
pyasn1-modules | 0.2.8 | 0.2.8
pyparsing | 2.4.7 | ▲ 3.0.7
python-dateutil | 2.8.2 | 2.8.2


pytz | 2021.3 | 2021.3
requests | 2.26.0 | ▲ 2.27.1
requests-oauthlib | 1.3.0 | ▲ 1.3.1
rsa | 4.8 | 4.8
scikit-learn | 1.0 | ▲ 1.0.2
scipy | 1.7.1 | ▲ 1.8.0
seaborn | 0.11.2 | 0.11.2
setuptools | 57.0.0 | ▲ 60.8.2
simplejson | 3.17.6 | 3.17.6
six | 1.16.0 | 1.16.0
sqlparse | 0.4.2 | 0.4.2
tensorboard | 2.7.0 | ▲ 2.8.0
tensorboard-data-server | 0.6.1 | 0.6.1
tensorboard-plugin-wit | 1.8.0 | ▲ 1.8.1
tensorflow | 2.7.0 | ▲ 2.8.0
tensorflow-estimator | 2.7.0 | ▲ 2.8.0
tensorflow-io-gcs-filesystem | 0.22.0 | ▲ 0.24.0
termcolor | 1.1.0 | 1.1.0
threadpoolctl | 3.0.0 | ▲ 3.1.0
typing-extensions | 4.0.0 | ▲ 4.0.1
urllib3 | 1.26.7 | ▲ 1.26.8
wheel | 0.36.2 | ▲ 0.37.1
wrapt | 1.13.3 | 1.13.3
xgboost | 1.5.0 | ▲ 1.5.2
zipp | 3.6.0 | ▲ 3.7.0

Εικόνα 32. Βιβλιοθήκες Python.

7.1.3 Kaggle

Είναι μια cross-sourced πλατφόρμα που προσελκύει, επιστήμονες από όλο τον κόσμο για την επίλυση επιστημονικών προβλημάτων, μηχανικής μάθησης και προγνωστικής ανάλυσης. Ξεκινώντας από τη Μελβούρνη, η Αυστραλία Kaggle μετακόμισε στη Silicon Valley το 2011, και στη συνέχεια εξαγοράστηκε από την Google το Μάρτιο του 2017. Το Kaggle είναι ο νούμερο ένα σταθμός για τους λάτρεις της επιστήμης δεδομένων σε όλο τον κόσμο.

Το Kaggle δίνει τη δυνατότητα σε επιστήμονες δεδομένων και άλλους προγραμματιστές να συμμετέχουν σε διαγωνισμούς μηχανικής εκμάθησης, να γράφουν, να μοιράζονται κώδικα και να φιλοξενούν σύνολα δεδομένων. Οι τύποι προβλημάτων της επιστήμης δεδομένων που δημοσιεύονται στο Kaggle μπορεί να είναι οτιδήποτε, από την προσπάθεια πρόβλεψης της εμφάνισης καρκίνου με την εξέταση των αρχείων ασθενών έως την ανάλυση συναισθημάτων που προκαλούνται από κριτικές ταινιών και πώς αυτό επηρεάζει την αντίδραση του κοινού.

Στην πλατφόρμα Kaggle δημιουργήθηκαν, τροποποιήθηκαν και σώθηκαν σε αρχεία όλα τα μοντέλα μηχανικής μάθησης που αναφέρονται στην συγκεκριμένη εργασία.

7.1.4 Front end

Η ανάπτυξη front-end, γνωστή και ως ανάπτυξη από την πλευρά του πελάτη, είναι η πρακτική της παραγωγής HTML, CSS και JavaScript για έναν ιστότοπο ή μια εφαρμογή Ιστού, έτσι ώστε ένας χρήστης να μπορεί να τα δει και να αλληλεπιδράσει άμεσα με τον ιστότοπο - εφαρμογή. Η πρόκληση που σχετίζεται με την ανάπτυξη διεπαφής είναι ότι τα εργαλεία και οι τεχνικές που χρησιμοποιούνται για τη δημιουργία της διεπαφής ενός ιστότοπου αλλάζουν συνεχώς και έτσι ο προγραμματιστής πρέπει να γνωρίζει συνεχώς τις νέες εξελίξεις του χώρου.

Ο στόχος του σχεδιασμού ενός ιστότοπου είναι να διασφαλιστεί ότι όταν οι χρήστες ανοίγουν τον ιστότοπο, βλέπουν τις πληροφορίες σε μια μορφή που είναι ευανάγνωστη. Αυτό περιπλέκεται περαιτέρω από το γεγονός ότι οι χρήστες χρησιμοποιούν πλέον μια μεγάλη ποικιλία συσκευών με διαφορετικά μεγέθη οθόνης και αναλύσεις, αναγκάζοντας έτσι τον σχεδιαστή να λάβει υπόψη αυτές τις πτυχές κατά το σχεδιασμό του ιστότοπου. Πρέπει να διασφαλίσουν ότι ο ιστότοπός τους εμφανίζεται σωστά σε διαφορετικά προγράμματα περιήγησης (cross-browser), διαφορετικά λειτουργικά συστήματα (cross-platform) και διαφορετικές συσκευές (cross-device), κάτι που απαιτεί προσεκτικό σχεδιασμό από την πλευρά του προγραμματιστή.

Συνήθως, ο τομέας της ανάπτυξης front-end περιλαμβάνει γλώσσες προγραμματισμού όπως HTML, CSS και JavaScript.

7.1.5 HTML

Το 1980, ο φυσικός Τιμ Μπέρνερς Λι, ο οποίος εργαζόταν στο CERN, επινόησε το ENQUIRE, ένα σύστημα χρήσης και διαμοιρασμού εγγράφων για τους ερευνητές του CERN, και κατασκεύασε ένα πρωτότυπό του. Αργότερα, το 1989, πρότεινε ένα σύστημα βασισμένο στο διαδίκτυο, το οποίο θα χρησιμοποιούσε υπερκείμενο. Έτσι, έφτιαξε την προδιαγραφή της HTML και έγραψε τον browser και το λογισμικό εξυπηρετητή στα τέλη του 1990. Τον ίδιο χρόνο, ο Μπέρνερς Λι και ο μηχανικός συστημάτων πληροφορικής του CERN Robert Cailliau συνεργάστηκαν σε μια κοινή προσπάθεια εύρεσης χρηματοδότησης, αλλά το έργο δεν υιοθετήθηκε ποτέ επίσημα από το CERN. Στις προσωπικές του σημειώσεις από το 1990, ο Μπέρνερς Λι αριθμεί «μερικές από τις πολλές χρήσεις του υπερκειμένου», όπως την γενική παρουσίαση πληροφοριών (π.χ. μια εγκυκλοπαίδεια), στοχευμένη δημοσίευση (κείμενα βοήθειας, τεκμηρίωσης, εκπαίδευσης, κλπ) μέχρι ακόμα και για καταγραφή προσωπικών σημειώσεων.

Συγκεκριμένα για το frond end κομμάτι της εφαρμογής χρησιμοποιήθηκε η HTML 5.

7.1.6 CSS

Η CSS (Cascading Style Sheets – διαδοχικά φύλλα ύφους ή επάλληλα φύλλα ύφους) είναι μια γλώσσα που ανήκει στην κατηγορία των γλωσσών φύλλων ύφους που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός ιστοτόπου. Η CSS είναι μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει στιλιστικά μια ιστοσελίδα δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και δίνει περισσότερες δυνατότητες σε σχέση με την html. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της CSS κρίνεται ως απαραίτητη.[36]

7.1.7 Bootsrap

Το Bootstrap είναι μια Βιβλιοθήκη HTML, CSS & JS που εστιάζει στην απλοποίηση της ανάπτυξης ενημερωτικών ιστοσελίδων (σε αντίθεση με τις εφαρμογές Ιστού). Ο πρωταρχικός σκοπός της προσθήκης του σε ένα έργο web είναι η εφαρμογή των επιλογών του Bootstrap για το χρώμα, το μέγεθος, τη γραμματοσειρά και τη διάταξη σε αυτό το έργο. Ως εκ τούτου, ο πρωταρχικός παράγοντας είναι εάν οι υπεύθυνοι προγραμματιστές βρίσκουν αυτές τις επιλογές σύμφωνα με τις προτιμήσεις τους. Μόλις προστεθεί σε ένα έργο, το Bootstrap παρέχει βασικούς ορισμούς στυλ για όλα τα στοιχεία HTML. Το αποτέλεσμα είναι μια ομοιόμορφη εμφάνιση για πεζογραφία, πίνακες και στοιχεία φόρμας στα προγράμματα περιήγησης ιστού. Επιπλέον, οι προγραμματιστές μπορούν να επωφεληθούν από τις κλάσεις CSS που ορίζονται

στο Bootstrap για να προσαρμόσουν περαιτέρω την εμφάνιση του περιεχομένου τους. Για παράδειγμα, το Bootstrap έχει προβλέψει πίνακες ανοιχτού και σκούρου χρώματος, επικεφαλίδες σελίδων, πιο εμφανή εισαγωγικά λέξης και κείμενο με επισήμανση.

Το Bootstrap έρχεται επίσης με πολλά στοιχεία JavaScript με τη μορφή πρόσθετων jQuery. Παρέχουν πρόσθετα στοιχεία διεπαφής χρήστη, όπως πλαίσια διαλόγου, συμβουλές εργαλείων και καρουζέλ. Κάθε στοιχείο Bootstrap αποτελείται από μια δομή HTML, δηλώσεις CSS και σε ορισμένες περιπτώσεις συνοδευτικό κώδικα JavaScript. Επεκτείνουν επίσης τη λειτουργικότητα ορισμένων υπαρχόντων στοιχείων διεπαφής, συμπεριλαμβανομένης για παράδειγμα μιας λειτουργίας αυτόματης συμπλήρωσης για πεδία εισαγωγής.[37]

Στην εφαρμογή που περιγράφεται στην παρούσα εργασία, χρησιμοποιήθηκε η έκδοση 5 του Bootsrap Framework.

7.1.8 Back end

Το Back-end Development αναφέρεται στην ανάπτυξη από την πλευρά του διακομιστή. Επικεντρώνεται σε βάσεις δεδομένων, scripting, αρχιτεκτονική ιστοσελίδας. Περιλαμβάνει παρασκηνιακές δραστηριότητες που συμβαίνουν κατά την εκτέλεση οποιασδήποτε ενέργειας σε έναν ιστότοπο. Μπορεί να είναι μια σύνδεση λογαριασμού ή μια αγορά από ένα ηλεκτρονικό κατάστημα. Ο κώδικας που γράφτηκε από προγραμματιστές back-end βοηθά τα προγράμματα περιήγησης να επικοινωνούν με πληροφορίες της βάσης δεδομένων.

7.1.9 Python

Η Python είναι μια γλώσσα προγραμματισμού διερμηνευόμενη (interpreted), γενικού σκοπού (general-purpose) και υψηλού επιπέδου. Ανήκει στις γλώσσες προστακτικού προγραμματισμού (Imperative programming) και υποστηρίζει τόσο το διαδικαστικό (procedural programming) όσο και το αντικειμενοστραφές (object-oriented programming) προγραμματιστικό υπόδειγμα (programming paradigm). Είναι δυναμική γλώσσα προγραμματισμού (dynamically typed) και υποστηρίζει συλλογή απορριμμάτων (garbage collection ή GC).

Δημιουργήθηκε από τον Ολλανδό Γκίντο βαν Ρόσσουμ (Guido van Rossum) στο ερευνητικό κέντρο Centrum Wiskunde & Informatica (CWI) το 1989[4] και κυκλοφόρησε για πρώτη φορά το 1991.

Ο κύριος στόχος της είναι η αναγνωσιμότητα του κώδικά της και η ευκολία χρήσης της. Το συντακτικό της επιτρέπει στους προγραμματιστές να εκφράσουν έννοιες σε λιγότερες γραμμές κώδικα από ότι θα ήταν δυνατόν σε γλώσσες όπως η C++ ή η Java. Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα αρκετές συνηθισμένες εργασίες και για την ταχύτητα εκμάθησης της. Μειονεκτεί στο ότι επειδή είναι διερμηνευόμενη είναι πιο αργή

από τις μεταγλωττιζόμενες (compiled) γλώσσες όπως η C και η C++. Για αυτόν τον λόγο δεν είναι κατάλληλη για δημιουργία κώδικα λειτουργικών συστημάτων.

Οι διερμηνευτές της Python είναι διαθέσιμοι για εγκατάσταση σε πολλά λειτουργικά συστήματα, επιτρέποντας στην Python την εκτέλεση κώδικα σε ευρεία γκάμα συστημάτων. Χρησιμοποιώντας εργαλεία τρίτων, όπως το Py2exe ή το Pyinstaller,[9] ο κώδικας της Python μπορεί να πακεταριστεί σε αυτόνομα εκτελέσιμα προγράμματα για μερικά από τα πιο δημοφιλή λειτουργικά συστήματα, επιτρέποντας τη διανομή του βασισμένου σε Python λογισμικού για χρήση σε αυτά τα περιβάλλοντα χωρίς να απαιτείται εγκατάσταση του διερμηνευτή της Python.

Η Python αναπτύσσεται ως ανοιχτό λογισμικό (open source) και η διαχείρισή της γίνεται από τον μη κερδοσκοπικό οργανισμό Python Software Foundation.[8] Ο κώδικας διανέμεται με την άδεια Python Software Foundation License η οποία είναι συμβατή με την GPL. Το όνομα της γλώσσας προέρχεται από την ομάδα των Άγγλων κωμικών Μόντυ Πάιθον και δεν έχει καμιά σχέση με το φίδι πύθωνα, παρότι το λογότυπό της παραπέμπει σε κάτι τέτοιο.[38]

8

Επίλογος

Μία από τις ερωτήσεις που χρειάστηκε να γίνουν για να εκπονηθεί η παρούσα εργασία, είναι αν μπορούν να γίνουν προβλέψεις στις χρηματοπιστωτικές αγορές χρησιμοποιώντας τεχνικές και αλγορίθμους μηχανική μάθηση.

Η έρευνα με βάση την βιβλιογραφία που μελετήθηκε για την εκπόνηση της διπλωματικής, δείχνει ότι οι τεχνικές μηχανικής μάθησης μπορούν να χρησιμοποιηθούν και να κάνουν αποτελεσματικές προβλέψεις στην ισοτιμία EUR/USD. Αποδείχθηκε ότι είναι δυνατόν να προβλεφθεί ως ένα βαθμό, η πιθανή τιμή που θα μπορούσε να αγγίξει η ισοτιμία μία μέρα μετά το κλείσιμο της τρέχουσας ημέρας.

8.1 Σύνοψη και συμπεράσματα

Στην παρούσα διπλωματική εργασία, έγινε μελέτη διαφόρων αλγορίθμων μηχανικής μάθησης, δημιουργήθηκαν μοντέλα και υλοποιήθηκε μία εφαρμογή ή οποία εξετάζει την απόδοση των μοντέλων μηχανικής μάθησης σε χρηματιστηριακά δεδομένα ισοτιμίας EUR/USD. Τα μοντέλα που δημιουργήθηκαν, εξετάστηκαν για την απόδοσή τους με βάση διάφορες μεθόδους ποσοτικής μέτρησης σφάλματος. Αναλυτικότερα για κάθε μοντέλο που υπολογίστηκαν τα errors mae (mean absolute error), mse (mean squared error) και το r2 score. Συγκεκριμένα εξετάστηκαν τα μοντέλα LSTM, SGD, BGR, XGB, Random Forest ,Linear Regressor, RNN και GRU.

Τα μοντέλα εξετάστηκαν, υλοποιήθηκαν και δοκιμάστηκαν σε data set δεδομένων ημέρας με παραμέτρους την τιμή ανοίγματος ημέρας, κλεισίματος ημέρας, υψηλή ημέρας, χαμηλή ημέρας και όγκο συναλλαγών. Στις παραμέτρους δεδομένων ημέρας προστέθηκαν και άλλες παράμετροι όπως αναφέρονται στο κεφάλαιο 5 με σκοπό την βελτίωση απόδοσης των μοντέλων.

Όλα τα αποτελέσματα των μοντέλων μηχανικής μάθησης είναι αρκετά κοντά με μικρή διαφορά καλύτερου αποτελέσματος του μοντέλου LSTM. Για την εξαγωγή συμπεράσματος πέρα από το θεωρητικό κομμάτι, ανοίχτηκαν εντολές σε demo account για τον έλεγχο απόδοσης των μοντέλων. Τα αποτελέσματα των εντολών δείχνουν ακρίβεια κερδισμένων εντολών κοντά στο 80%, αλλά πρέπει να τονιστεί ότι για το άνοιγμα των εντολών λήφθηκαν υπόψιν και άλλες παράμετροι όπως διάφοροι indicators (Bollinger bands, Stochastic oschillator). Η καλύτερη προσσέγιση τιμής έγινε με χρήση του μοντέλου LSTM καθώς φαίνεται να παρουσιάζει σταθερότερη απόδοση στην πρόβλεψη τιμής, έναντι των άλλων μοντέλων με μικρή όμως διαφορά.

8.2 Μελλοντικές επεκτάσεις

Στα πλαίσια μελλοντικής επέκτασης αξίζει να διερευνηθούν και να αναλυθούν ακόμη περισσότερα χαρακτηριστικά που να βελτιώνουν την απόδοση των μοντέλων πρόβλεψης της ισοτιμίας EUR/USD. Θα μπορούσαν να εισαχθούν διάφοροι indicators στα δεδομένα, οι οποίοι δίνουν αρκετές φορές χρήσιμες πληροφορίες για την κίνηση της ισοτιμίας καθώς για την τάση που πιθανότατα να ακολουθηθεί.

Σημαντική βελτίωση θα αποτελούσε η δημιουργία μοντέλων σε μικρότερα time frames όπως 15’, 30’ ώρας και 4 ωρών καθώς πιθανότατα να αποτύπωναν καλύτερα την στιγμιαία τάση, γεγονός που θα αποτελούσε σημαντική πληροφορία για την ευνοϊκότερη τιμή ανοίγματος της εντολής καθώς και την ευνοϊκότερη τιμή και ώρα κλεισίματος της εντολής.

Η τελευταία εξέλιξη της τεχνολογίας αλγορίθμων μηχανικής μάθησης είναι οι αλγόριθμοι transformers. Αν και αρχικά δημιουργήθηκαν για να χρησιμοποιηθούν στον τομέα natural language processing (NLP), θα μπορούσε να τροποποιηθεί ώστε να δημιουργηθεί ένα μοντέλο έχοντας σαν βάση τους αλγορίθμους transformers για την βελτίωση της απόδοσης με βάση τα μοντέλα που δημιουργήθηκαν παραπάνω.

Ένα πολύ σημαντικό κομμάτι στο forex αποτελούν τα νέα ή οι ειδήσεις τα οποία είναι υπεύθυνα για την δημιουργία τάσης. Τα προγραμματισμένα νέα όπως και κάποια tweets θα μπορούσαν να δημιουργήσουν ή να επηρεάσουν την τάση ημέρας οπότε παίζουν σημαντικό ρόλο στις κινήσεις του δείκτη ισοτιμίας και πρέπει να λαμβάνονται σοβαρά υπόψιν.

Με βάση την διπλωματική εργασία του Maxim Afteniy με τίτλο “predicting time series with transformers” παρουσιάζονται αποτελέσματα με μεγαλύτερη ακρίβεια συγκριτικά με τα LSTM μοντέλα. Η υλοποίηση ενός μοντέλου με transformers θα μπορούσε να αποτελέσει μία σημαντική βελτίωση στην απόδοση της πρόβλεψης τιμών ισοτιμίας.

9

Βιβλιογραφία

[1] BIS, 2013. Triennial Central Bank Survey Foreign Exchange Turnover in April 2013: preliminary global results. www.bis.org/publ/rpfx13fx.pdf.

[2] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D.. 2016. Mastering The Game of Go with

[3] Li, Y.; Zheng, W.; Zheng, Z. Deep Robust Reinforcement Learning for Practical Algorithmic Trading. IEEE Access 2019, 7, 108014–108022.

[4] Chen, T.; Su, W. Local Energy Trading Behavior Modeling With Deep Reinforcement Learning. IEEE Access 2018, 6, 62806–62814.

[5] Si, W.; Li, J.; Ding, P.; Rao, R. A Multi-objective Deep Reinforcement Learning Approach for Stock Index Future's Intraday Trading. In Proceedings of the 2017 10th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 9–10 December 2017; pp. 431–436.

[6] Kumar, P.H.; Patil, S.B. Forecasting volatility trend of INR USD currency pair with deep learning LSTM techniques. In Proceedings of the 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), Bengaluru, India, 20–22 December 2018; pp. 91–97.

[7] Ma, Y.; Han, R. Research on stock trading strategy based on deep neural network. In Proceedings of the 18th International Conference on Control,

Automation and Systems (ICCAS), PyeongChang, Korea, 17–20 October 2018; pp. 92–96.

[8] Chen, C.T.; Chen, A.; Huang, S. Cloning Strategies from Trading Records using Agent-based Reinforcement Learning Algorithm. In Proceedings of the IEEE International Conference on Agents (ICA), Singapore, 28–31 July 2018; pp. 34–37.

[9] Korczak, J.; Hemes, M. Deep learning for financial time series forecasting inA-Trader system. In Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS), Prague, Czech Republic, 3–6 September 2017; pp. 905–912.

[10] Wang, J.; Sun, T.; Liu, B.; Cao, Y.; Wang, D. Financial Markets Prediction with Deep Learning. In Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 17–20 December 2018; pp. 97–104.

[11] Lee, R.S.T. Chaotic Type-2 Transient-Fuzzy Deep Neuro-Oscillatory Network (CT2TFDNN) for Worldwide Financial Prediction. IEEE Trans. Fuzzy Syst. May 2019.

[12] Zarkias, K.S.; Passalis, N.; Tsantekidis, A.; Tefas, A. Deep Reinforcement Learning for Financial Trading Using Price Trailing. In Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 3067–3071.

[13] Kalyankar, G. D., Poojara, Shivananda R., Dharwadkar,Nagaraj V., 2017. Predictive analysis of diabetic patient data using machine learning and hadoop. [Online] pp. 619 - 624. Available from: https://ieeexplore.ieee.org/document/8058253 [Accessed 28th December 2018].

[14] Muller, A. C., Guido, Sarah (2017). Introduction to machine learning with python: a guide for data scientists, O'reilly media, inc., 1005 Gravenstein highway north, Sebastopol, ca 95472.

[15] Mitchell, T. M. (1997). Machine learning, mcgraw-hill science/engineering/math.

[16] Awad. Mariette, K. R. (2015). Efficient learning machines: theories, concepts, and applications for engineers and system designers, Apress, Berkeley, CA.

[17] Raschka, S. (2015). unlock deeper insights into machine learning with this vital guide to cutting-edge predictive analytics, Packt.

[18] Grus, J. (2015). data science from scratch, O'Reilly Media, INC., 1005 Gravenstein highway north, Sebastopol, CA 95472.

[19] Polat, K., Gunes, s., 2007. An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease. [Online] 17.(4), pp. 702-710. Available from: https://www.sciencedirect.com/science/article/pii/S1051200406001370.

[20] Yıldırım, D. C., Toroslu, I. H., & Fiore, U. (2021). Forecasting directional movement of Forex data using LSTM with technical and macroeconomic indicators. In Financial Innovation (Vol. 7, Issue 1). Springer Science and Business Media LLC. https://doi.org/10.1186/s40854-020-00220-2

[21] Chihab, Y., Bousbaa, Z., Chihab, M., Bencharef, O., & Ziti, S. (2019). Algo-Trading Strategy for Intraweek Foreign Exchange Speculation Based on Random Forest and Probit Regression. In Applied Computational Intelligence and Soft Computing (Vol. 2019, pp. 1–13). Hindawi Limited. https://doi.org/10.1155/2019/8342461 K. Orr. Data quality and systems theory. In Communications of the ACM, 41, 2, pp. 54-57, Feb. 1998.

[22] Pradeepkumar, D., & Ravi, V. (2016). FOREX Rate prediction using Chaos and Quantile Regression Random Forest. In 2016 3rd International Conference on Recent Advances in Information Technology (RAIT). 2016 3rd International Conference on Recent Advances in Information Technology (RAIT). IEEE. https://doi.org/10.1109/rait.2016.7507954

[23] Ahmed, S., Hassan, S.-U., Aljohani, N. R., & Nawaz, R. (2020). FLF-LSTM: A novel prediction system using Forex Loss Function. In Applied Soft Computing (Vol. 97, p. 106780). Elsevier BV. https://doi.org/10.1016/j.asoc.2020.106780 K. Orr. Data quality and systems theory. In Communications of the ACM, 41, 2, pp. 54-57, Feb. 1998.

[24] Yu, N., & Haskins, T. (2021). Bagging Machine Learning Algorithms: A Generic Computing Framework Based on Machine-Learning Methods for Regional Rainfall Forecasting in Upstate New York. In Informatics (Vol. 8, Issue 3, p. 47). MDPI AG. https://doi.org/10.3390/informatics8030047.

[25] Charkha, P. R. (2008). Stock Price Prediction and Trend Prediction Using Neural Networks. First International Conference on Emerging Trends in

Engineering and Technology (pp. 592–594). Ieee. doi:10.1109/ICETET.2008.223

[26] Kotsiantis SB (2007), Supervised Machine Learning: A Review of Classification Techniques, Informatica: 31, 249-268.

[27] Witten IH, Frank E, Hall MA (2011), Data mining: practical machine learning tools and techniques, Morgan Kaufmann Publishers.

[28] Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. In PeerJ Computer Science (Vol. 7, p. e623). PeerJ. https://doi.org/10.7717/peerj-cs.623

[29] Kumar Sarangi, P., Chawla, M., Ghosh, P., Singh, S., & Singh, P. K. (2022). FOREX trend analysis using machine learning techniques: INR vs USD currency exchange rate using ANN-GA hybrid approach. In Materials Today: Proceedings (Vol. 49, pp. 3170–3176). Elsevier BV. https://doi.org/10.1016/j.matpr.2020.10.960.

[30] Ranjit, S., Shrestha, S., Subedi, S., & Shakya, S. (2018). Comparison of algorithms in Foreign Exchange Rate Prediction. In 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS). 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS). IEEE. https://doi.org/10.1109/cccs.2018.8586826.

[31] Szepesvári, C. (2010). Algorithms for Reinforcement Learning. In Synthesis Lectures on Artificial Intelligence and Machine Learning (Vol. 4, Issue 1, pp. 1–103). Morgan & Claypool Publishers LLC. https://doi.org/10.2200/s00268ed1v01y201005aim009.

[32] Burkart, N., & Huber, M. F. (2021). A Survey on the Explainability of Supervised Machine Learning. In Journal of Artificial Intelligence Research.

[33] Kadiyala, A., & Kumar, A. (2018). Applications of python to evaluate the performance of bagging methods. In Environmental Progress & Sustainable Energy (Vol. 37, Issue 5, pp. 1555–1559). Wiley. https://doi.org/10.1002/ep.13018.

[34] Zhang, T. (2004). Solving large scale linear prediction problems using stochastic gradient descent algorithms. In Twenty-first international conference on Machine learning - ICML '04. Twenty-first international conference. ACM Press. https://doi.org/10.1145/1015330.1015332.

[35] https://el.wikipedia.org/wiki/HTML (προσπελάστηκε 5/1/2022)

[36] https://el.wikipedia.org/wiki/CSS (προσπελάστηκε 5/1/2022)

[37] https://www.geeksforgeeks.org/bootstrap (προσπελάστηκε 5/1/2022)

[38] https://el.wikipedia.org/wiki/Python (προσπελάστηκε 5/1/2022)