

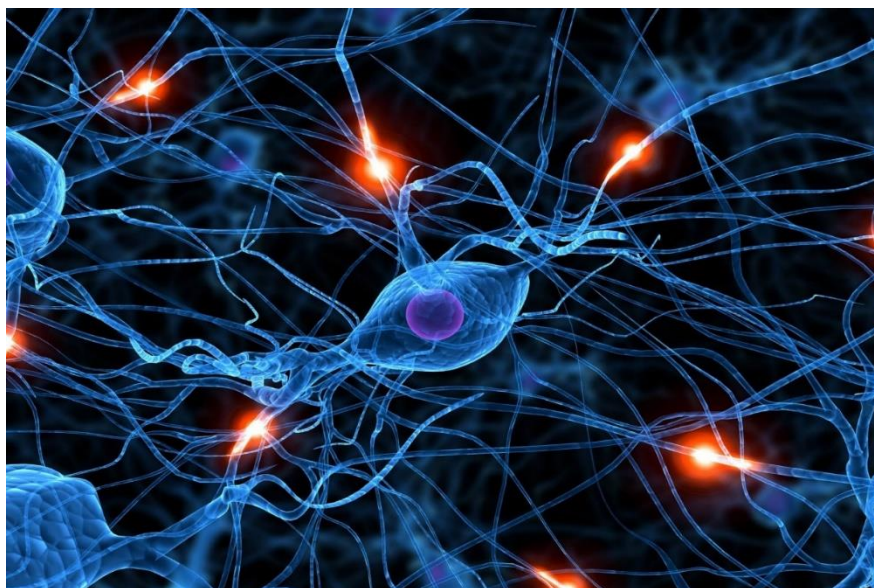
ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΤΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
Πρόγραμμα Μεταπτυχιακών Σπουδών
“Πληροφορικά Συστήματα & Υπηρεσίες”
Κατεύθυνση
“Προηγμένα Πληροφοριακά Συστήματα”



UNIVERSITY OF PIRAEUS
DEPARTMENT OF DIGITAL SYSTEMS
Postgraduate Programme
“Information Systems & Services”
Specialization
“Advanced Information Systems”

Διπλωματική Εργασία

**«Διαδικτυακή εφαρμογή για την πρόβλεψη μελλοντικών τιμών στο
Χρηματιστήριο με χρήση Νευρωνικών Δικτύων»**



Νίκος Αυγέρος

Επιβλέπων: Μιχαήλ Φιλιππάκης, ΠΑΠΕΙ

Αθήνα, Φλεβάρης 2022

Περίληψη

Η παρούσα διπλωματική εργασία αφορά στην ανάπτυξη διαδικτυακής εφαρμογής για την πρόβλεψη των μελλοντικών τιμών του Χρηματιστηρίου με τη χρήση τεχνικών νευρωνικών δικτύων. Στα πλαίσια της διπλωματικής εργασίας, αναπτύχθηκαν νευρωνικά δίκτυα διαφόρων αρχιτεκτονικών με σκοπό τόσο την ακριβέστερη πρόβλεψη των μελλοντικών τιμών κλεισίματος όσο και την τάση για διάφορες μετοχές, κυρίως όμως για τις IBM, Tesla και Google. Κατά την εκπαίδευση των μοντέλων, πραγματοποιήθηκαν διάφορα πειράματα στις παραμέτρους των δικτύων με σκοπό την καλύτερη απόδοση των μοντέλων. Για κάθε μοντέλο που αναπτύχθηκε δοκιμάστηκε η απόδοσή του σε πραγματικό σύνολο δεδομένων ελέγχου, ώστε να συγκριθεί και να αξιολογηθεί. Η διαδικτυακή εφαρμογή κάνει χρήση αυτών των μοντέλων, δίνοντας τη δυνατότητα στον χρήστη να επιλέξει το μοντέλο που τον ενδιαφέρει για πρόβλεψη της τιμής κλεισίματος μιας μετοχής για τον χρονικό ορίζοντα που δίδεται από την εφαρμογή.

Στην πρώτη ενότητα της εργασίας προσφέρεται την απαραίτητη γνώση που αφορά τόσο στη μηχανική όσο και στη βαθιά μηχανική μάθηση, για την ανάπτυξη αποδοτικών μοντέλων πρόβλεψης. Αναλύονται οι αρχιτεκτονικές των τεχνητών νευρωνικών δικτύων, καθώς και ο τρόπος εκπαίδευσης τους για την ορθή επιλογή κατάλληλων μοντέλων για πρόβλεψη.

Στη δεύτερη ενότητα εξηγούνται βασικές έννοιες των χρονοσειρών, αναλύονται τα χαρακτηριστικά τους, καθώς και αναφέρονται παραδοσιακές μέθοδοι πρόβλεψης που εφαρμόζονται για την αναγνώριση της συμπεριφοράς της χρονοσειράς.

Στην τρίτη ενότητα αναπτύσσονται τα τεχνητά νευρωνικά δίκτυα βάση της θεωρίας των δύο πρώτων ενοτήτων και κατόπιν αξιολόγησης εντοπίζονται τα βέλτιστα μοντέλα που αναπτύχθηκαν. Επιπροσθέτως, εξάγουμε συμπεράσματα από την γενική απόδοση των μοντέλων βαθιάς μηχανικής μάθησης για την πρόβλεψη των μελλοντικών τιμών των μετοχών.

Τέλος, στην τέταρτη ενότητα αναπτύσσεται η διαδικτυακή εφαρμογή και παρουσιάζονται οι δυνατότητές της και το γραφικό περιβάλλον. Έγινε προσπάθεια το περιβάλλον να είναι φιλικό προς το χρήστη, παρουσιάζοντας διαγράμματα των αποτελεσμάτων που προκύπτουν από τις επιλογές που διαθέτει η εφαρμογή.

Για την ανάπτυξη των τεχνητών νευρωνικών δικτύων και της διαδικτυακής εφαρμογής, χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python, η οποία τα τελευταία χρόνια έχει υψηλή δημοτικότητα στην ανάπτυξη μοντέλων μηχανική και βαθιάς μάθησης. Οι βασικότερες βιβλιοθήκες που χρησιμοποιήθηκαν είναι οι Pandas, NumPy, Matplotlib, Streamlit, Keras και Tensorflow, οι οποίες προσφέρουν τις απαραίτητες μεθόδους, κλάσεις και συναρτήσεις που χρειαστήκαμε για την υλοποίηση της εφαρμογής.

Λέξεις κλειδιά: Βαθιά Μάθηση, Νευρωνικά δίκτυα, Χρονοσειρές, LSTM, GRU, SimpleRNN.

Περιεχόμενα

1	Τεχνητά Νευρωνικά Δίκτυα	6
1.1	Εισαγωγή στην Τεχνητή Νοημοσύνη.....	6
1.1.1	Μηχανική Μάθηση (Machine Learning).....	6
1.1.2	Νευρωνικά Δίκτυα (Neural Networks).....	7
1.1.3	Βαθιά Μάθηση (Deep Learning).....	7
1.2	Δομή και Αρχιτεκτονικές Τεχνητών Νευρωνικών Δικτύων.....	7
1.2.1	Συνάρτηση Ενεργοποίησης (Activation Function).....	9
1.2.2	Στρώματα Νευρωνικού Δικτύου (NN Layers)	10
1.2.3	Μονοστρωματικά Νευρωνικά Δίκτυα εμπρόςθιας τροφοδότησης (SLP Feedforward)	11
1.2.4	Πολυστρωματικά Νευρωνικά Δίκτυα εμπρόςθιας τροφοδότησης (MLP Feedforward).....	11
1.2.5	Αναδρομικά Νευρωνικά Δίκτυα (RNN).....	12
1.2.6	Συνελεκτικά Νευρωνικά Δίκτυα (CNN)	14
1.3	Εκπαίδευση Τεχνητών Νευρωνικών Δικτύων.....	15
1.3.1	Μάθηση με επίβλεψη (Supervised Learning).....	16
1.3.2	Μάθηση χωρίς επίβλεψη (Unsupervised Learning)	17
1.3.3	Ενισχυτική Μάθηση (Reinforcement Learning).....	17
1.4	Κανόνες εκπαίδευσης.....	17
1.4.1	Κανόνας Hebbian (Hebb's Rule).....	18
1.4.2	Κανόνας Delta (Delta Rule)	18
1.4.3	Αλγόριθμος οπισθοδιάδοσης σφάλματος (Back Propagation).....	18
1.4.4	Κανόνας ανταγωνιστικής μάθησης (Kohonen's Learning Law).....	19
1.5	Αλγόριθμοι βελτιστοποίησης	19
1.5.1	Κατάβαση Κλίσης (Gradient Descent).....	19
1.5.2	Στοχαστική Κατάβαση Κλίσης (Stochastic Gradient Descent).....	20
1.5.3	RMSProp	20
1.5.4	ADAM.....	20
2	Χρονοσειρές	20
2.1	Εισαγωγή στις χρονοσειρές.....	20
2.2	Συνιστώσες χρονοσειρών	22
2.2.1	Τάση (Trend).....	22
2.2.2	Εποχικότητα (Seasonality)	22
2.2.3	Κυκλικότητα (Cyclicality).....	22
2.2.4	Τυχαίες τιμές (Random variations).....	23
2.3	Στασιμότητα (Stationarity).....	23
2.4	Αιτιοκρατία και στοχαστικότητα (Determinism and Stochasticity).....	23
2.5	Γραμμικότητα και μη γραμμικότητα (Linearity and Non-linearity).....	24
2.6	Ανάλυση χρονοσειρών	24

2.7	Μέθοδοι Πρόβλεψης Χρονοσειρών	25
2.7.1	Μέθοδοι εξομάλυνσης (Smoothing methods)	25
2.7.2	Μέθοδοι διάσπασης (Decomposition)	26
2.7.3	Ανάλυση ARIMA (ARIMA Analysis)	29
3	Ανάπτυξη Νευρωνικών Δικτύων για πρόβλεψη.....	29
3.1	Εισαγωγή μοντελοποίησης Νευρωνικών Δικτύων	29
3.2	Συλλογή δεδομένων.....	30
3.3	Επεξεργασία δεδομένων	30
3.4	Ανάλυση δεδομένων.....	32
3.5	Κατασκευή Νευρωνικών Δικτύων	34
3.5.1	Μετοχή της Google (GOOGL).....	34
3.5.2	Μετοχή της TESLA (TSLA)	43
3.5.3	Μετοχή της IBM (IBM)	51
3.6	Αξιολόγηση	59
4	Ανάπτυξη εφαρμογής πρόβλεψης μετοχών.....	60
4.1	Γραφική απεικόνιση της διαδικτυακής εφαρμογής.....	61
5	Επίλογος	72
5.1	Συμπεράσματα.....	72
5.2	Μελλοντικές επεκτάσεις.....	72
	Βιβλιογραφία	74
	Διαδικτυακοί σύνδεσμοι.....	74

1 ΤΕΧΝΗΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

1.1 Εισαγωγή στην Τεχνητή Νοημοσύνη

Η Τεχνητή Νοημοσύνη αποτελεί μια γνωστική περιοχή της Πληροφορικής, που ασχολείται με την υλοποίηση ευφύων υπολογιστικών συστημάτων, μιμούμενα της ανθρώπινης συμπεριφοράς. Με άλλα λόγια, αφορά στην ικανότητα των μηχανών να μαθαίνουν μέσα από την εμπειρία και να προσαρμόζονται σε νέα εισαγόμενα δεδομένα, ώστε να εκτελούν ανθρωπομορφικά έργα.

Ένα σύστημα τεχνητής νοημοσύνης μπορεί να πλεονεκτεί συγκριτικά με τον άνθρωπο, καθώς ένα τέτοιο σύστημα μπορεί να είναι διαθέσιμο ανά πάσα χρονική στιγμή, ενώ ο χρόνος εκπαίδευσης είναι ταχύτερος και πραγματοποιείται με μεγαλύτερη ευκολία σε σχέση με τον άνθρωπο που η εκπαίδευσή του μπορεί να είναι μακροχρόνια και χρονοβόρα. Επιπλέον, ένα τέτοιο σύστημα μπορεί να είναι αντικειμενικό, αφού χρησιμοποιώντας τα ίδια δεδομένα και κάτω από αντίξοες συνθήκες μπορεί να εξάγει τα ίδια αποτελέσματα. Στον αντίποδα, αυτό ταυτόχρονα αποτελεί μειονέκτημα, καθώς ένα σύστημα τεχνητής νοημοσύνης δίνοντας τα ίδια συμπεράσματα με τη χρήση του ίδιου συνόλου δεδομένων αδυνατεί να πρωτοτυπήσει και να διαθέσει έμπνευση. Επιπλέον, ένας σημαντικός παράγοντας απόδοσης για ευφυή συστημάτων αφορά στην ποιότητα των δεδομένων, νοώντας ότι αν τα δεδομένα είναι ανακριβείς τότε αυτό θα αντικατοπτρίζεται και στα αποτελέσματα. Μια επιπλέον πρόκληση στη χρήση συστημάτων τεχνητής νοημοσύνης είναι ο περιορισμός στην εκτέλεση καθηκόντων. Ειδικότερα, ένα σύστημα που έχει αναπτυχθεί για να παίζει σκάκι αδυνατεί να παίζει πασιέντζα, ενώ ένα σύστημα που ανιχνεύει απάτες στις ασφαλιστικές απαιτήσεις δεν έχει τη δυνατότητα να ανιχνεύσει φορολογικές απάτες. Επομένως, τα συστήματα τεχνητής νοημοσύνης επικεντρώνονται σε έναν στόχο και αδυνατούν να συμπεριφερθούν όπως ο άνθρωπος. Ο λόγος που αναπτύσσονται τέτοιου είδους συστήματα είναι για την υποστήριξη στην λήψη αποφάσεων και για την αντιμετώπιση πολύπλοκων προβλημάτων, χωρίς να αποτελούν υποκατάστατο των ανθρώπων.

1.1.1 Μηχανική Μάθηση (Machine Learning)

Η Μηχανική Μάθηση συνιστά μια προηγμένη μορφή ανάλυσης και υπολογισμού δεδομένων αξιοποιώντας την εξαιρετική ταχύτητα επεξεργασίας και τις τεχνικές αναγνώρισης προτύπων των υπολογιστών για την εύρεση και εκμάθηση νέων τάσεων στα δεδομένα. Η μηχανική μάθηση αποτελεί πεδίο της επιστήμης των υπολογιστών που αφορά στην τεχνική προγραμματισμού εμπνευσμένη από την τεχνητή νοημοσύνη. Ειδικότερα, ο υπολογιστής μαθαίνει και βελτιώνεται μέσα από τα δεδομένα από τα οποία έχει πρόσβαση με σκοπό την επίλυση πολύπλοκων προβλημάτων. Η μέθοδος εκμάθησης της μηχανής αντικατοπτρίζει τον τρόπο που μαθαίνει ο άνθρωπος μέσα από τη ζωή.

1.1.2 Νευρωνικά Δίκτυα (Neural Networks)

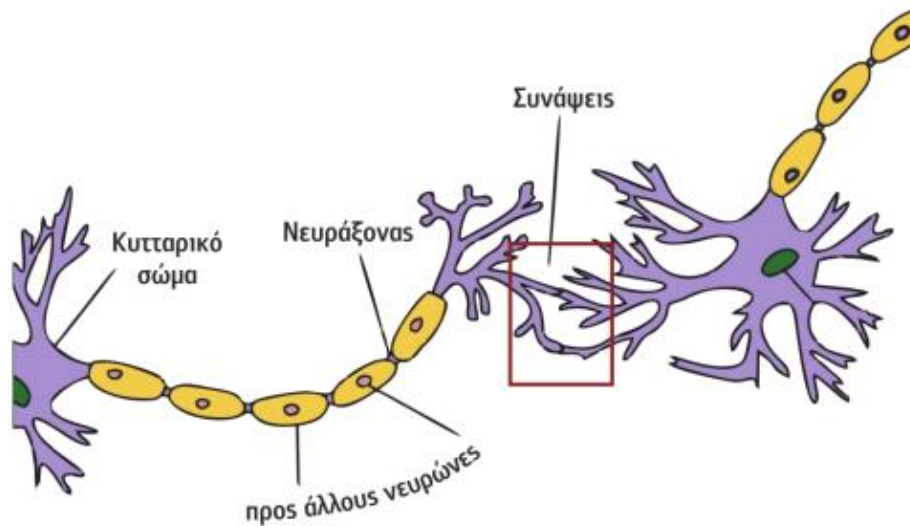
Τα Νευρωνικά Δίκτυα αφορούν στον τύπο μηχανικής μάθησης που αποτελείται από αλληλοσυνδεδεμένους κόμβους ή νευρώνες, οι οποίοι επεξεργάζονται τις πληροφορίες που δέχονται από εξωτερικά δεδομένα και στη συνέχεια προωθούν τις πληροφορίες μεταξύ κάθε κόμβου. Η διαδικασία απαιτεί πολλαπλές διελεύσεις στα δεδομένα προκειμένου να βρεθούν συνδέσεις και να γίνει εξαγωγή νοήματος από ακαθόριστα δεδομένα.

1.1.3 Βαθιά Μάθηση (Deep Learning)

Η Βαθιά μάθηση κάνει χρήση τεράστιων Νευρωνικών Δικτύων, τα οποία διαθέτουν πολλά επίπεδα μονάδων επεξεργασίας, αξιοποιώντας τις εξελίξεις στην υπολογιστική ισχύ και τις βελτιωμένες τεχνικές εκπαίδευσης για την μάθηση πολύπλοκων μορφών σε μεγάλες ποσότητες δεδομένων. Οι κοινές εφαρμογές της περιλαμβάνουν την αναγνώριση εικόνας και ομιλίας.

1.2 Δομή και Αρχιτεκτονικές Τεχνητών Νευρωνικών Δικτύων

Τα Τεχνητά Νευρωνικά Δίκτυα αποτελούν μαθηματικά μοντέλα εμπνευσμένα από τα βιολογικά μοντέλα του ανθρώπινου εγκεφάλου, μιμούμενα τη συμπεριφορά των νευρώνων. Σε βιολογικό επίπεδο ένας νευρώνας αποτελεί ένα κελί, το οποίο δέχεται και στέλνει ηλεκτρικά σήματα. Οι νευρώνες είναι τοποθετημένοι σε ομάδες, όπου κάθε ομάδα συνιστά ένα νευρωνικό δίκτυο αποτελούμενο από χιλιάδες διασυνδεδεμένους νευρώνες. Ο ανθρώπινος εγκέφαλος αποτελείται από δισεκατομμύρια νευρώνες που συνδέονται μεταξύ τους και επικοινωνούν μέσα από τους δενδρίτες που λαμβάνουν τα ηλεκτρικά σήματα. ως είσοδο και τους νευράξονες που στέλνουν το σήμα ως έξοδο. Το νευρικό κύτταρο ή σώμα αθροίζει τα εισερχόμενα σήματα και όταν λαμβάνει κατάλληλο σήμα, τότε στέλνει τα ηλεκτρικά σήματα μέσω από του νευροάξονα προς τους γειτονικούς νευρώνες. Το σήμα που εκπέμπει ένας νευρώνας μπορεί να θεωρηθεί ως δυαδικό, καθώς το κυτταρικό σώμα μπορεί να είναι είτε ήρεμο είτε να στέλνει σήματα.



Εικόνα 1.1: Κυτταρικό σώμα ανθρώπινου εγκεφάλου

Στα Τεχνητά Νευρωνικά Δίκτυα, κάθε νευρώνας αποτελεί ουσιαστικά μια συνάρτηση, που δέχεται τιμές εισόδου και παράγει μια έξοδο και η δομή του μοιάζει αρκετά με το βιολογικό μοντέλο. Τα βασικά στοιχεία που αποτελείται το μοντέλο του τεχνητού νευρώνα είναι τα εξής:

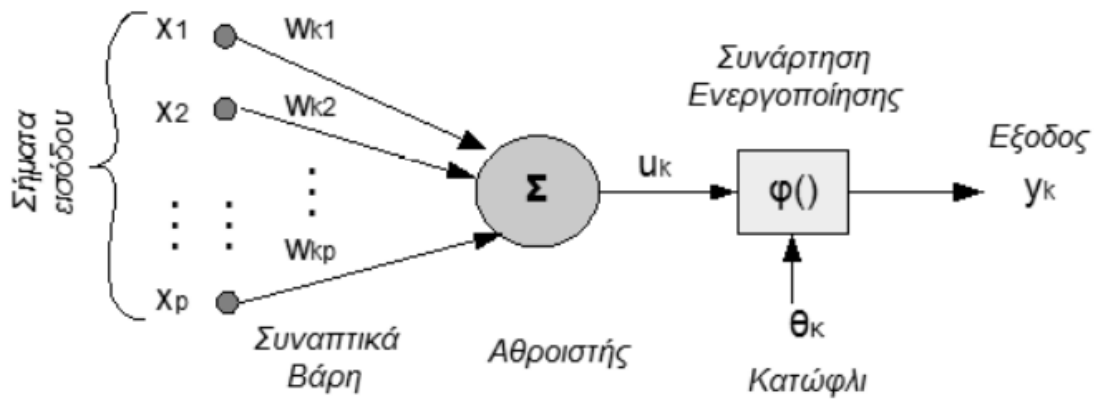
- Τις συνάψεις, χαρακτηρίζονται από βάρη, όπου σε κάθε είσοδο αντιστοιχεί ένα βάρος. Ειδικότερα, ένα σήμα x_k στην είσοδο της σύναψης j συνδεδεμένο στον νευρώνα k , πολλαπλασιάζεται με το βάρος w_{kj} .
- Τον αθροιστή u_k , ο οποίος αθροίζει τα σήματα εισόδου, αφού έχουν πολλαπλασιαστεί με τα βάρη. Επομένως, κατά την εισαγωγή στον νευρώνα ή αλλιώς κόμβο του δικτύου νοείται το σταθμικό άθροισμα των αποτελεσμάτων εξόδων των κόμβων που συνδέονται μαζί του, του οποίου το αποτέλεσμα καλείται τοπικό πεδίο.

$$u_k = \sum_{j=0}^m w_{kj} x_j$$

- Τη συνάρτηση ενεργοποίησης φ , για τη μείωση του εύρους της εξόδου του νευρώνα μεταξύ των διαστημάτων $[0, 1]$ ή $[-1, 1]$.

$$\varphi(u_k)$$

- Το κατώφλι ή πόλωση θ_k που αφορά στη μια επιπλέον είσοδο με τιμή 1 και συναπτικό βάρος w_{k0} . Το κατώφλι αποτελεί εξωτερικό παράγοντα και μόλις το άθροισμα ξεπεράσει την τιμή κατωφλίου, τότε ο νευρώνας παράγει μια έξοδο, την οποία στέλνει ως ηλεκτρικό σήμα στους γειτονικούς νευρώνες, μέσω του σημείου ένωσης που ονομάζεται. Έτσι λέμε ότι ο νευρώνας ενεργοποιείται.



Εικόνα 1.2: Τεχνητός νευρώνας

Σύμφωνα με τα ανωτέρω, μια τυπική συνάρτηση που αποτελεί την έξοδο του νευρώνα προκύπτει ως εξής,

$$y_k = \varphi(u_k)$$

1.2.1 Συνάρτηση Ενεργοποίησης (Activation Function)

Όπως αναφέρθηκε ανωτέρω, τα σήματα εισόδου x_i που δέχεται το δίκτυο, πολλαπλασιάζονται με τα αντίστοιχα βάρη και αθροίζονται, με αποτέλεσμα να σχηματίζεται ένα βεβαρημένο άθροισμα net .

$$net = \sum_i w_i x_i$$

Τα δεδομένα εισόδου μπορεί να αφορούν στην έξοδο άλλου τεχνητού νευρώνα ή στις αρχικές εισόδους. Τα βάρη w_i αφορούν στην ισχύ της σύνδεσης, όπου σε αυτά είναι αποθηκευμένη η γνώση των νευρωνικών δικτύων. Μια απλή περίπτωση για την κατανόηση της λειτουργίας της συνάρτησης ενεργοποίησης θα ήταν εάν η έξοδος έπαιρνε τις διακριτές τιμές Ναι ή Όχι, δηλαδή 0 ή 1, τότε θα μπορούσαμε να ορίζουμε μία τιμή κατωφλίου (θ), όπου όταν η είσοδος είναι μεγαλύτερη ή ίση του μηδενός, τότε η έξοδος θα ήταν 1 και αντιθέτως, όταν η είσοδος ήταν μικρότερη του μηδενός, άρα αρνητική, τότε η έξοδος θα ήταν 0. Κατόπιν, του γραμμικού υπολογισμού, ο νευρώνας πραγματοποιεί τη συνάρτηση ενεργοποίησης. Τυπικά παραδείγματα αποτελούν τα εξής:

- Ταυτοτική συνάρτηση, κατά την οποία δεν εκτελείται καμία ενέργεια και η έξοδος του νευρώνα είναι ο γραμμικός συνδυασμός.

$$f(net) = net$$

- Βηματική συνάρτηση , κατά την οποία εάν η τιμή του γραμμικού συνδυασμού είναι μεγαλύτερη της τιμής πόλωσης ή κατωφλίου, τότε ο νευρώνας αποστέλλει παλμό (ON), ενώ αν είναι μικρότερη τότε δεν εκτελεί καμία ενέργεια και θεωρείτο αδρανής (OFF).

$$f(net) = \begin{cases} 0 & : net < 0 \\ 1 & : net \geq 0 \end{cases}$$

- Σιγμοειδής (Sigmoid), η οποία αποτελεί μια ήπια μορφή της βηματική συνάρτησης.

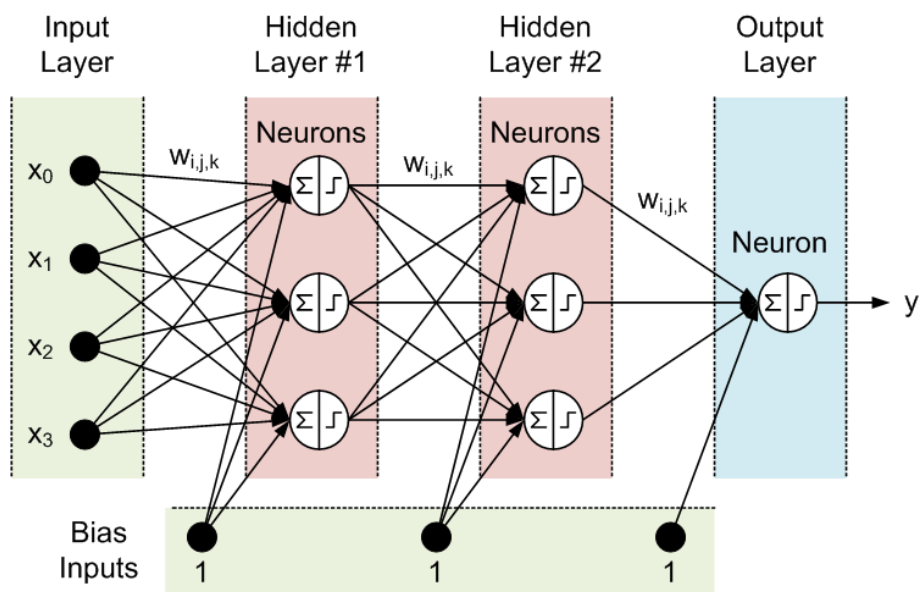
$$f(net) = \frac{1}{1 + e^{-net}}$$

- Ανορθωμένη Γραμμική Μονάδα (ReLU), γνωστή και ως ενεργοποίηση ράμπας, αποτελεί τη δημοφιλέστερη συνάρτηση ενεργοποίησης, καθώς υπολογίζει γρηγορότερα και ευκολότερα την τελική έξοδο του νευρώνα. Ειδικότερα, θέτει όλες τις αρνητικές τιμές ίσες με 0.

$$f(u) = \max(0, u)$$

1.2.2 Στρώματα Νευρωνικού Δικτύου (NN Layers)

Οι νευρώνες ενός δικτύου χρησιμοποιούνται για να εξυπηρετήσουν έναν σκοπό όσον αφορά τον πραγματικό κόσμο. Ειδικότερα, μερικοί νευρώνες διασυνδέουν το δίκτυο με το περιβάλλον λαμβάνοντας δεδομένα εισόδου (Input Layer), άλλοι νευρώνες παρέχουν δεδομένα εξόδου (Output Layer), ενώ το υπόλοιπο δίκτυο παραμένει κρυφό (Hidden Layer).

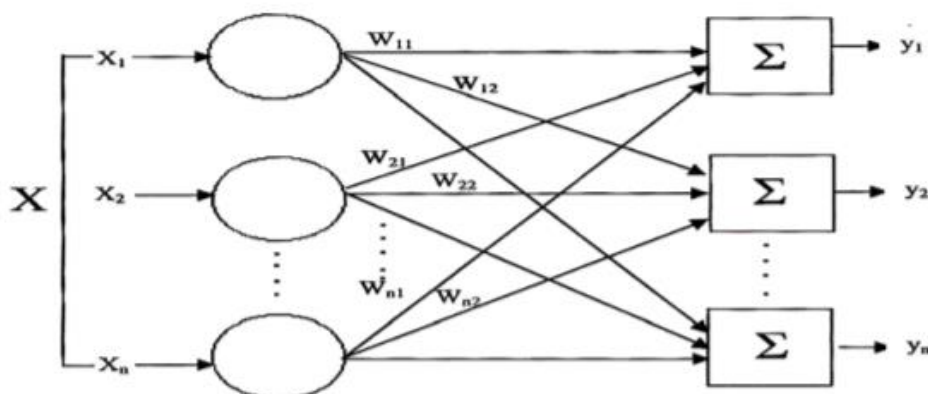


Εικόνα 1.3: Στρώματα νευρωνικού δικτύου

Οι νευρώνες του δικτύου ομαδοποιούνται σε στρώματα (neuron layers), όπου κάθε νευρώνας των κρυφών στρωμάτων συνδέεται με τους προηγούμενους και τους επόμενους νευρώνες. Οι πληροφορίες που εισάγονται σε ένα νευρωνικό δίκτυο μεταβιβάζονται από το ένα στρώμα στο αμέσως επόμενο. Ένα δίκτυο μπορεί να περιέχει κανένα, ένα ή περισσότερα κρυμμένα στρώματα, αλλά αν υπάρχει αυξημένος αριθμός στρωμάτων, τότε το δίκτυο μπορεί να έχει πρόβλημα.

1.2.3 Μονοστρωματικά Νευρωνικά Δίκτυα εμπρόσθιας τροφοδότησης (SLP Feedforward)

Τα μονοστρωματικά δίκτυα (Single Layer Perceptron) αποτελούν την απλούστερη μορφή αρχιτεκτονικής ενός νευρωνικού δικτύου, αποτελούμενο από δύο στρώματα, ένα στρώμα εισόδου και ένα εξόδου. Γενικά, το μονοστρωματικά δίκτυα μπορεί να περιέχουν περισσότερους κόμβους, αλλά στην απλούστερη μορφή τους, ένα δίκτυο αποτελείται από έναν και μοναδικό νευρώνα, το οποίο ονομάζεται στοιχειώδης αισθητήρας (elementary perceptron). Το επίπεδο εισόδου δεν εκτελεί κανέναν υπολογισμό και τροφοδοτεί απευθείας το στρώμα εξόδου, στο οποίο κάθε νευρώνας παράγει ένα άθροισμα από τα δεδομένα εισόδου πολλαπλασιασμένα με τα αντίστοιχα βάρη. Η αρχιτεκτονική αυτή ονομάζεται προς τα εμπρός τροφοδοτούμενο (Feedforward), καθώς η τροφοδότηση των τιμών εισόδου πραγματοποιείται μόνο από το επίπεδο εισόδου και όχι το αντίστροφο.

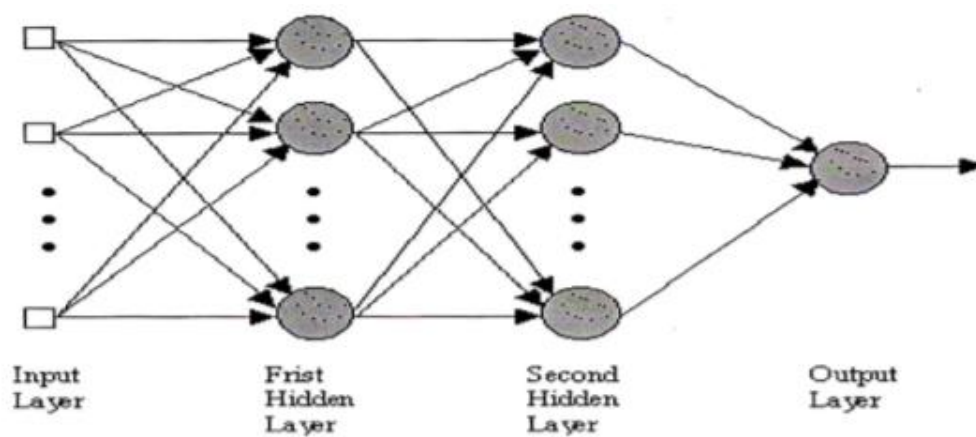


Εικόνα 1.4: Μονοστρωματικό νευρωνικό δίκτυο

1.2.4 Πολυστρωματικά Νευρωνικά Δίκτυα εμπρόσθιας τροφοδότησης (MLP Feedforward)

Τα πολυστρωματικά ή αλλιώς πολυεπίπεδα δίκτυα (Multilayer Perceptrons) είναι περισσότερο πολύπλοκα και προσφέρουν μεγαλύτερη υπολογιστική ικανότητα, καθώς μπορεί να σχηματιστούν από ομάδες μονοστρωματικών δικτύων. Η ροή του σήματος είναι πρόσθιας τροφοδότησης, όπως στα μονοστρωματικά, με τη διαφορά ότι περιέχουν περισσότερα στρώματα κόμβων, εκτός των επιπέδων εισόδου και εξόδου. Κατά αυτή τη δομή, τα κρυφά στρώματα παρέχουν περισσότερη υπολογιστική ισχύ με αποτέλεσμα το δίκτυο να είναι περισσότερο αποδοτικό, καθώς

έχει την ικανότητα να αφομοιώνει περισσότερα δεδομένα εισόδου μέσω του αυξημένου αριθμού συνάψεων και της πολυπλοκότητας των αλληλεπιδράσεων που δημιουργούνται μεταξύ των κόμβων. Κατά την είσοδο των δεδομένων, το στρώμα εισόδου τροφοδοτεί το πρώτο κρυφό στρώμα με τις τιμές εισόδου πολλαπλασιασμένες με τα αντίστοιχα βάρη. Κατόπιν, το πρώτο κρυφό επίπεδο εκτελεί τους υπολογισμούς, αθροίζοντας τα δεδομένα και στη συνέχεια τροφοδοτεί το δεύτερο κρυφό στρώμα, όπου τα επεξεργασμένα δεδομένα πολλαπλασιάζονται πάλι με τα βάρη διασύνδεσης. Τα σήματα που εξάγει το πρώτο κρυφό στρώμα αποτελούν είσοδο για το αμέσως επόμενο και η διαδικασία αυτή πραγματοποιείται μέχρις ότου τα επεξεργασμένα δεδομένα να φτάσουν στο στρώμα εξόδου, όπου θα επεξεργαστούν μια τελευταία φορά και το δίκτυο θα παράγει μια έξοδο.

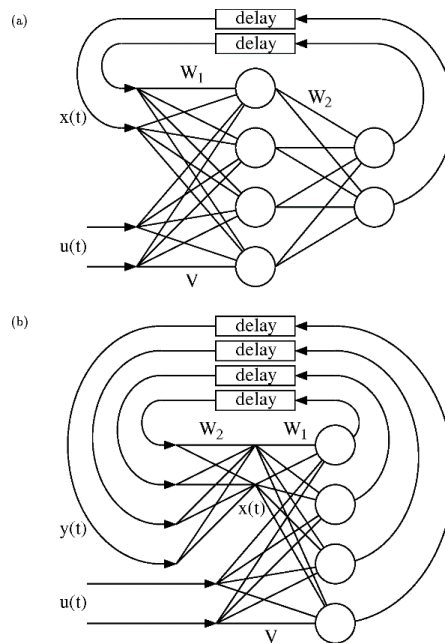


Εικόνα 1.5: Πολυστρωματικό νευρωνικό δίκτυο

1.2.5 Αναδρομικά Νευρωνικά Δίκτυα (RNN)

Η αρχιτεκτονική των αναδρομικών δικτύων (Recurrent Neural Networks) διαφέρει από τα προς τα εμπρός τροφοδοτούμενα δίκτυα, καθώς εμπεριέχει έναν βρόγχο ανάδρασης (feedback loop), ο οποίος ανατροφοδοτεί το σήμα εξόδου στην είσοδο του νευρώνα που υλοποιείται. Ο ρόλος ύπαρξης ενός τέτοιου βρόγχου είναι η βελτίωση της απόδοσης του δικτύου και στην ικανότητά του να εκπαιδευτεί.

Ειδικότερα, δίδεται η δυνατότητα αυτοανάδρασης, καθώς ένας νευρώνας μπορεί να δέχεται ως είσοδο την ίδια του την έξοδο, ενώ σε πιο πολύπλοκες αρχιτεκτονικές αναδρομικών δικτύων, οι συνδέσεις ανάδρασης να πραγματοποιούνται όχι μόνο από και προς νευρώνες εισόδου και εξόδου στο δίκτυο, αλλά και από και προς κρυφούς νευρώνες. Επιπροσθέτως, οι βρόγχοι ανάδρασης περιλαμβάνουν τη χρήση στοιχείων καθυστέρησης (delay units) σε κάποιες διασυνδέσεις, επιτρέποντας στο δίκτυο να αποκτήσει μη γραμμική δυναμική συμπεριφορά, το οποίο στοιχείο είναι απαραίτητο για συγκεκριμένες εφαρμογές.



Εικόνα 1.6: Αναδρομικό νευρωνικό δίκτυο

Τα συμβατικά αναδρομικά δίκτυα παρουσιάζουν το πρόβλημα στο να μάθουν μακροσκελείς αλληλουχίες που προκύπτει από το πρόβλημα της εξαφάνισης της κλίσης (vanishing gradient). Αυτό συμβαίνει, διότι κατά την εκπαίδευση οι νευρώνες αποκτούν μηδενική κλίση οδηγώντας τις υπόλοιπες κλίσεις των προηγούμενων στρωμάτων προς το μηδέν, με αποτέλεσμα τη συρρίκνωση και εξαφάνιση των τιμών τους έπειτα από μερικά χρονικά βήματα. Επομένως σε μεγάλες χρονικές στιγμές, η συνεισφορά των εν λόγω κλίσεων είναι μηδενική δημιουργώντας πρόβλημα στο δίκτυο.

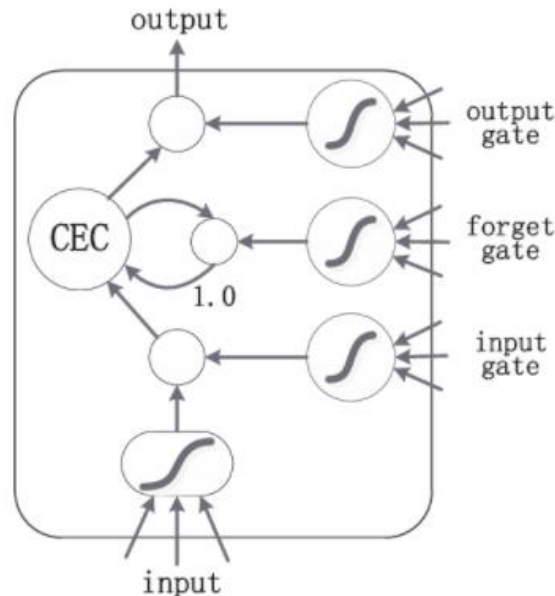
1.2.5.1 Μακράς βραχυπρόθεσμης μνήμης δίκτυα (LSTMNs)

Τα μακράς βραχυπρόθεσμης μνήμης δίκτυα (Long Short-Term Memory Networks) αποτελούν μια ειδική αρχιτεκτονική αναδρομικών δικτύων και αναπτύχθηκαν για να αντιμετωπίσουν το πρόβλημα της εξαφάνισης της κλίσης, δηλαδή να μπορούν μάθουν από μακροσκελείς αλληλουχίες, επιτυγχάνοντας καλύτερα αποτελέσματα. Η διαφορά τους από τα συμβατικά αναδρομικά δίκτυα είναι συνάρτηση που διαθέτουν για τον υπολογισμό της κρυφής κατάστασης. Ειδικότερα, ο συνηθισμένος νευρώνας που γνωρίζουμε σε προηγούμενες αρχιτεκτονικές δικτύων, αντικαθίσταται με μια πιο σύνθετη μονάδα που αφορά στο μπλοκ μνήμης του LSTM δικτύου και κάθε μπλοκ περιέχει ένα ή περισσότερα κελιά μνήμης. Τα κελιά των LSTM, τα οποία λαμβάνουν ως είσοδο την προηγούμενη κατάσταση και την τρέχουσα είσοδο, αποφασίζουν τι πρέπει να διατηρηθεί και τι να διαγραφεί και κατόπιν συνδυάζουν την προηγούμενη κατάσταση, την τρέχουσα μνήμη και την είσοδο. Τα κύρια χαρακτηριστικά που συνθέτουν το μπλοκ μνήμης είναι τα εξής:

- **Constant Error Carousel (CEC):** Αποτελεί μια κεντρική μονάδα, η οποία διαθέτει μια επαναλαμβανόμενη σύνδεση που αφορά στον βρόγχο ανατροφοδότησης με χρονική στιγμή ίση με ένα και η ενεργοποίηση του σφάλματος λειτουργεί ως μνήμη για τις προηγούμενες

πληροφορίες. Η CEC λύνει το πρόβλημα της εξαφάνισης της κλίσης που προκύπτει στα αναδρομικά δίκτυα, δίδοντας τη δυνατότητα να μαθαίνει μακροσκελείς ακολουθίες.

- **Input Gate:** Αφορά στην πολλαπλασιαστική μονάδα, η οποία προστατεύει τις αποθηκευμένες πληροφορίες του CEC από πιθανές άσχετες εισόδους.
- **Output Gate:** Αφορά στην πολλαπλασιαστική μονάδα, η οποία προστατεύει τις υπόλοιπες μονάδες από ενδεχόμενες παρεμβολές που μπορεί να προκύπτουν από τις αποθηκευμένες πληροφορίες που βρίσκονται στο CEC.



Εικόνα 1.7: Μακράς βραχυπρόθεσμης μνήμης δίκτυα

1.2.5.2 Gated Recurrent Units (GRUs)

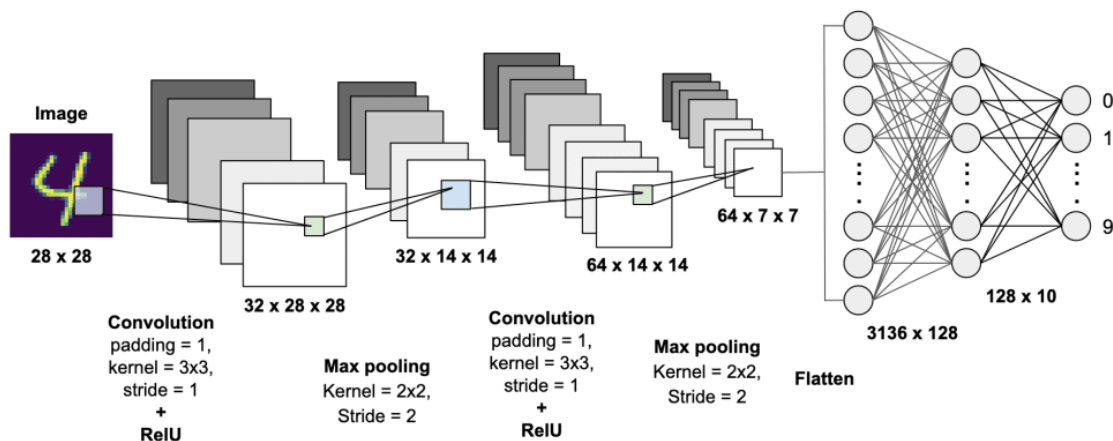
Ως Gated Recurrent Unit (GRU) ονομάζουμε την αρχιτεκτονική τεχνητών νευρωνικών δικτύων που παρουσιάστηκε για πρώτη φορά το 2014 από τον Cho και το κύριο χαρακτηριστικό της είναι η δυνατότητα της να προσαρμόζει την κάθε μονάδα ενός επαναλαμβανόμενου (recurrent) νευρωνικού δικτύου, με στόχο την απαθανάτιση των εξαρτήσεων που παρατηρούνται σε διαφορετικές χρονικές περιόδους. Όπως και ο LSTMN σχεδιάστηκε με σκοπό να επαναφέρει ή να αναβαθμίζει το περιεχόμενο της μνήμης του. Κάθε GRU έχει μία πύλη επαναφοράς $r_j t$ και μία πύλη αναβάθμισης $z_j t$, οι οποίες αντιστοιχούν στις πύλες μνήμης και εισόδου ενός LSTMN. Το GRU εκθέτει πλήρως το περιεχόμενο της μνήμης του σε κάθε χρονικό σημείο και ισορροπεί μεταξύ του προηγούμενου περιεχομένου της μνήμης του και του νέου περιεχομένου της μνήμης του χρησιμοποιώντας αυστηρά ολοκλήρωση διαρροής (leaky integration).

1.2.6 Συνελεκτικά Νευρωνικά Δίκτυα (CNN)

Τα συνελεκτικά νευρωνικά δίκτυα (Convolutional Neural Networks) διαφέρουν σημαντικά από τις προαναφερθέντα αρχιτεκτονικές και η ιδιαιτερότητά τους αφορά στην ικανότητα τους να

αναγνωρίζουν διαφορετικά στοιχεία σε εικόνες με μεγάλη ακρίβεια, καθώς ως είσοδο δέχονται έναν πολυδιάστατο πίνακα και όχι ένα διάνυσμα μιας διάστασης. Ειδικότερα, ένα συνελεκτικό δίκτυο έχει τρισδιάστατη μορφή, η οποία αλλάζει από επίπεδο σε επίπεδο. Τα επίπεδα που απαρτίζουν το δίκτυο μπορεί να είναι ένα ή περισσότερα συνελεκτικά επίπεδα (convolutional layers), στα οποία εισάγονται πολλές φορές αναμεσά τους τα συγκεντρωτικά επίπεδα (pooling layer) με σκοπό την περαιτέρω μείωση των παραμέτρων και πιθανόν κοντά στην έξοδο ένα ή περισσότερα πλήρως συνδεδεμένα επίπεδα (fully connected layers) ή μερικώς συνδεδεμένα επίπεδα (partially connected layers).

Στο συνελεκτικό επίπεδο, κάθε μονάδα λαμβάνει σήματα από ένα μέρος των μονάδων από το προηγούμενο επίπεδο μέσα από ένα σύνολο συναπτικών βαρών που είναι κοινό για όλες τις μονάδες του επιπέδου. Ακολουθώντας, στο εισερχόμενο σήμα εφαρμόζεται η μαθηματική συνάρτηση της συνέλιξης μεταδίδοντας το σήμα στο αμέσως επόμενο επίπεδο. Κατά την εφαρμογή της συνέλιξης, μειώνεται ο αριθμός των παραμέτρων σε κάθε επίπεδο, ώστε το δίκτυο να είναι πιο αποδοτικό υπολογιστικά και να υπάρχει το περιθώριο εφαρμογής περισσότερων επιπέδων. Στο συγκεντρωτικό επίπεδο, κάθε μονάδα δέχεται σήμα από το ένα μέρος των μονάδων του συνελεκτικού επιπέδου και πραγματοποιεί υπολογισμούς, όπως την εύρεση της μέσης ή μέγιστης τιμής, παρέχοντάς τα ως σήμα σε μοναδικό νευρώνα του επόμενου επιπέδου. Τα συνελεκτικά νευρωνικά χρησιμοποιούν σχετικά μικρή προ-επεξεργασία συγκριτικά με άλλους αλγορίθμους ταξινόμησης εικόνων και η δυνατότητα να επεξεργαστούν πολλαπλές συστοιχίες δεδομένων με μεταβλητά μεγέθη είναι σημαντικό πλεονέκτημα σε εφαρμογές Βαθιάς Μάθησης.



Εικόνα 1.8: Συνελεκτικό νευρωνικό δίκτυο

1.3 Εκπαίδευση Τεχνητών Νευρωνικών Δικτύων

Κατόπιν ανάπτυξης ενός τεχνητού νευρωνικού δικτύου, έρχεται η στιγμή που το δίκτυο πρέπει εκπαιδευτεί για να εξάγει συμπεράσματα. Κατά την εκπαίδευση, το δίκτυο μαθαίνει μέσα από ένα πλήθος από παραδείγματα που καλείται να αναγνωρίσει και η διαδικασία εκπαίδευσης επαναλαμβάνεται σταδιακά, μεταβάλλοντας το δίκτυο από την εμπειρία που αποκτάει μέσα από την

επανάληψη. Ο τρόπος εκμάθησης είναι παρόμοιος με τον τρόπο που μαθαίνει ο άνθρωπος μέσα από τις εμπειρίες του και αλληλοεπιδρώντας με το περιβάλλον. Στον ανθρώπινο εγκέφαλο, η δομή της σύνδεσης μεταξύ των νευρώνων μεταβάλλεται κατά το χρόνο εκμάθησης, όπου ορισμένες συνδέσεις παραμένουν σταθερές, ώστε να μπορούν να ανακαλέσουν την πληροφορία που έμαθαν. Κατά την εκπαίδευση των νευρωνικών δικτύων, οι τιμές των βαρών μεταβάλλονται, ενώ ο αριθμός των νευρώνων και των συνδέσεων, που προσδιορίζονται κατά την κατασκευή του δικτύου, παραμένουν σταθεροί. Επιπροσθέτως, κατόπιν της εκπαίδευσης του δικτύου υπάρχει ένα ορισμένο σφάλμα, το οποίο αφορά στην απόδοση του δικτύου, δηλαδή πόσο ακριβείς είναι η έξοδος του δικτύου συγκριτικά με την αναμενόμενη έξοδο. Οι παράγοντες που καθορίζουν την εκπαίδευση του δικτύου είναι οι αλγόριθμοι εκπαίδευσης και σε ορισμένες περιπτώσεις κάποιος εξωτερικός παράγοντας που παρεμβαίνει στην εκπαίδευσή του. Στη συνέχεια αναφέρονται μερικές από τις βασικές κατηγοριοποιήσεις εκπαίδευσης των δικτύων.

1.3.1 Μάθηση με επίβλεψη (Supervised Learning)

Η μάθηση με επίβλεψη αποτελεί μια διαδικασία κατά την οποία οι αλγόριθμοι μάθησης δέχονται ως είσοδο ένα σύνολο δεδομένων εκπαίδευσης, όπου η επιθυμητή έξοδος είναι γνωστή στον εκπαιδευτή. Κατά αυτόν τον τρόπο, η μηχανή συνδέει το σύνολο εισόδου με αυτό της εξόδου σε μια περίοδο επαναληπτικής μάθησης, αποσκοπώντας σε μια γενίκευση του κανόνα για να υλοποιεί συσχετίσεις μεταξύ εισόδου και εξόδου, έτσι ώστε όταν τροφοδοτείται με νέα δεδομένα να παρέχει σωστές εξόδους. Σε πρακτικό επίπεδο, ο εκπαιδευτής επιβλέποντας το μοντέλο πραγματοποιεί κατάλληλη παραμετροποίηση, ώστε η μηχανή να διορθώνει τα λάθη και να παράγει σωστές εξόδους.

Η μάθηση με επίβλεψη χρησιμοποιείται κυρίως για την επίλυση προβλημάτων κατηγοριοποίησης ή ταξινόμησης και παλινδρόμησης. Στο πρόβλημα της κατηγοριοποίησης το υπολογιστικό σύστημα καλείται να κατατάξει σε ένα προκαθορισμένο σύνολο κλάσεων τα δεδομένα εισόδου που δέχεται. Εν αντιθέσει, στην παλινδρόμηση μέσω της εκπαίδευσης μιας συνάρτησης, η τιμή εξόδου είναι συνεχής και όχι διακριτή. Η παλινδρόμηση χρησιμοποιείται σε προβλήματα πρόβλεψης, βάση μια συνάρτησης από ανεξάρτητες μεταβλητές προβλέποντας την εξαρτημένη τιμή, όπως την τιμή των μετοχών ή της θερμοκρασίας.

Στα νευρωνικά δίκτυα, κατά την είσοδο ενός διανύσματος δεδομένων συγκρίνεται η έξοδος του δικτύου με την επιθυμητή έξοδο, ενώ το σφάλμα που υπολογίζεται, τροφοδοτείται προς τα πίσω μέσα στο δίκτυο, προσαρμόζοντας συνεχώς τα βάρη σύμφωνα με τον αλγόριθμο μείωσης του σφάλματος, ούτως ώστε να είναι στο χαμηλότερο επιθυμητό επίπεδο. Στη διαδικασία μάθησης με επίβλεψη, το δίκτυο τροφοδοτείται συνεχώς με το ίδιο σύνολο δεδομένων, επαναπροσδιορίζοντας τα βάρη συνδέσεων για την ελάττωση του σφάλματος, ακολουθώντας μια επαναληπτική δομή που τροποποιούνται τα βάρη του δικτύου, ανάλογα τον αλγόριθμο κατάρτισης που χρησιμοποιείται για την προσαρμογή αυτή.

1.3.2 Μάθηση χωρίς επίβλεψη (Unsupervised Learning)

Η μάθηση χωρίς επίβλεψη συνιστά μια διαδικασία, όπου το υπολογιστικό σύστημα δέχεται ένα σύνολο εισόδου χωρίς να είναι γνωστό το επιθυμητό σύνολο εξόδου. Σκοπός του συστήματος είναι η εξαγωγή μοτίβων, όπου ο εκπαιδευτής δεν γνωρίζει τη σωστή απάντηση και το σύστημα βρίσκει ενδιαφέρουσες τάσεις μέσα από το σύνολο εκπαίδευσης. Η διαδικασία αντικατοπτρίζει τον τρόπο της άτυπης μάθησης που μαθαίνει ο άνθρωπος αλληλοεπιδρώντας με το περιβάλλον. Οι αλγόριθμοι μάθησης χωρίς επίβλεψη χρησιμοποιούνται κυρίως για την αντιμετώπιση προβλημάτων ομαδοποίησης και ανάλυση συσχετίσεων. Κατά την ομαδοποίηση, το υπολογιστικό σύστημα ταξινομεί τα δεδομένα σε ομάδες, βάση των κοινών χαρακτηριστικών μεταξύ των δεδομένων εισόδου και χρησιμοποιείται στην στατιστική ανάλυση, την αναγνώριση προτύπων, στην μείωση των δεδομένων, στην παραγωγή και έλεγχο μίας υπόθεσης κ.α. Η ανάλυση κανόνων συσχέτισης αφορά στην εξαγωγή κανόνων και συσχετίσεων μεταξύ των δεδομένων εισόδου που χαρακτηρίζουν τις ιδιότητές τους, όπου βρίσκει εφαρμογή στην εξόρυξη δεδομένων, στην αγοραστική συμπεριφορά του καταναλωτή κ.α.

1.3.3 Ενισχυτική Μάθηση (Reinforcement Learning)

Η ενισχυτική μάθηση αφορά στην υβριδική προσέγγιση της μάθησης με επίβλεψη, καθώς το υπολογιστικό σύστημα μαθαίνει μια στρατηγική ενεργειών αλληλοεπιδρώντας άμεσα με το περιβάλλον. Η Ενισχυτική Μάθηση εφαρμόζεται σε για την αντιμετώπιση προβλημάτων Σχεδιασμού, όπως ο έλεγχος κίνησης ρομπότ, στη μάθηση επιτραπέζιων παιχνιδιών κ.α. Το υπολογιστικό σύστημα δεν επιβλέπεται από κάποιον σχετικά με την επιλογή των ενεργειών που θα ακολουθήσει, αντιθέτως ανακαλύπτει μόνο του τις ενέργειες που μεγιστοποιούν το κέρδος. Η διαδικασία αποτελεί προσέγγιση δοκιμής και σφάλματος, βασιζόμενο στο σύστημα ανταμοιβής με πόντους παρόμοιο με τα παιχνίδια, στοχεύοντας στη βέλτιστη διαδρομή ελαχιστοποιώντας τους πόρους και μεγιστοποιώντας τον χρόνο.

1.4 Κανόνες εκπαίδευσης

Η προσαρμογή των συνοπτικών βαρών μεταξύ δύο επιτυχημένων κύκλων εκπαίδευσης (epochs) σύμφωνα με τα πρότυπα εισόδου σε ένα δίκτυο πραγματοποιείται με τη μορφή εκπαίδευσης που ακολουθεί το δίκτυο, γνωστό ως κανόνα εκπαίδευσης. Οι κανόνες εκπαίδευσης αφορούν στους μαθηματικούς αλγόριθμους που έχουν ως σκοπό την ανανέωση των βαρών που συνδέουν το δίκτυο, έπειτα από τον υπολογισμό του σφάλματος. Μερικοί από τους σημαντικότερους κανόνες είναι οι εξής:

1.4.1 Κανόνας Hebbian (Hebb's Rule)

Κατά τον κανόνα Hebbian, το βάρος μεταξύ των νευρώνων ενισχύεται όταν δύο κόμβοι είναι ενεργοί, δηλαδή έχουν το ίδιο πρόσημο και ο ένας λαμβάνει είσοδο από τον άλλον. Ο κανόνας του Hebb περιγράφεται ως εξής:

$$w'_{ij} = w_{ij} + ax_iy_i$$

Σύμφωνα με τον τύπο, το διορθωμένο συναπτικό βάρος w'_{ij} υπολογίζεται από το άθροισμα του αρχικού συναπτικού βάρους και του αποτελέσματος του γινομένου που προκύπτει από την είσοδο x_i του νευρώνα i στο νευρώνα j , την έξοδο y_i του νευρώνα j και του ρυθμού εκπαίδευσης a που είναι πάντοτε θετικός. Επομένως, ανάλογα τη συχνότητα που διεγείρει ο νευρώνας i τον νευρώνα j , τότε αναπτύσσεται μεταξύ τους μια μεταβαλλόμενη σύνδεση. Αν οι νευρώνες ενεργοποιούνται ταυτόχρονα, τότε τα συναπτικά βάρη αυξάνονται, ενώ σε αντίθετη περίπτωση, η ισχύς της σύνδεσης ελαττώνεται, δηλαδή τα συναπτικά βάρη μειώνονται.

1.4.2 Κανόνας Delta (Delta Rule)

Ο κανόνας Δέλτα είναι ένας από τους πιο γνωστούς και συνηθέστερους κανόνες που χρησιμοποιείται στην εκπαίδευση των δικτύων, χρησιμοποιώντας τη βηματική συνάρτηση. Αποτελεί μια παραλλαγή του κανόνα Hebbian και ο υπολογισμός των βαρών πραγματοποιείται ως εξής:

$$w_i = w_{old} - d * (a_j - a_i)$$

Στον ανωτέρω τύπο, η νέα τιμή του βάρους w_i υπολογίζεται από την αφαίρεση της παλαιάς τιμής του βάρους w_{old} και του γινομένου μεταξύ του ρυθμού εκμάθησης d και της διαφοράς της επιθυμητής εξόδου a_j από την τρέχουσα έξοδο a_i . Ο ρυθμός εκμάθησης καθορίζει πόσο γρήγορα συγκλίνει η εκπαίδευση και η τιμή του είναι πάντα μεγαλύτερη του μηδενός. Ο κανόνας Delta βασίζεται στη συνεχή τροποποίηση των βαρών μειώνοντας τη διαφορά μεταξύ της εξόδου του δικτύου και της επιθυμητής εξόδου. Η μεταφορά του σφάλματος μεταφέρεται από το ένα στρώμα στο αμέσως προηγούμενο στρώμα μέχρι να φτάσει το σφάλμα στο πρώτο επίπεδο. Στην αρχή το δίκτυο προσδιορίζει τυχαίες τιμές στα βάρη και συγκρίνοντας το αποτέλεσμα της εξόδου με την επιθυμητή έξοδο και στη συνέχεια προσαρμόζει τα κατάλληλα βάρη μέχρις ότου επιτευχθεί το χαμηλότερο επιθυμητό σφάλμα.

1.4.3 Αλγόριθμος οπισθοδιάδοσης σφάλματος (Back Propagation)

Η μάθηση με οπισθοδιάδοση του σφάλματος είναι ο πιο συνηθισμένος τρόπος για την εκπαίδευση ενός νευρωνικού δικτύου. Κατά τη διάρκεια της εκπαίδευσης, μόλις το δίκτυο ολοκληρώσει τη διαδικασία και εξάγει ένα αποτέλεσμα, υπολογίζεται το σφάλμα ανάλογα με τη συνάρτηση κόστους που έχει οριστεί. Στη συνέχεια, υπολογίζει την μερική παράγωγο της

συνάρτησης σφάλματος ως προς την είσοδο. Κατόπιν, με τον κανόνα της αλυσίδας υπολογίζονται τα σφάλματα σε όλα τα προηγούμενα επίπεδα μέχρι την είσοδο, τέλος μιας εποχής. Γνωρίζοντας πια όλες τις μερικές παραγώγους, μπορούμε να ανανεώσουμε τα βάρη και την πόλωση σε όλους τους νευρώνες όλων των επιπέδων προς την κατεύθυνση της μείωσης του σφάλματος εξόδου.

1.4.4 Κανόνας ανταγωνιστικής μάθησης (Kohonen's Learning Law)

Ο κανόνας ανταγωνιστικής μάθησης, γνωστός και ως κανόνας Kohonen, χρησιμοποιείται στα δίκτυα που ακολουθούν μη εποπτευόμενη μέθοδο μάθησης και ο σκοπός εφαρμογής του ήταν για τη δημιουργία του αυτό-οργανωμένου (self-organizing) δικτύου. Οι νευρώνες εξόδου του δικτύου ανταγωνίζονται μεταξύ τους για το ποιος θα ενεργοποιηθεί και ο νευρώνας με τη μεγαλύτερη έξοδο ορίζεται ως νικητής, έχοντας την ικανότητα να παρεμποδίσει άλλους νευρώνες και να διεγείρει του γειτονικούς του. Η έξοδος προκύπτει από τον νικητή νευρώνα και μόνο ο νικητής και οι γειτονικοί νευρώνες μπορούν να ανανεώσουν τα συναπτικά βάρη, νοώντας ότι έχουν το δικαίωμα να μάθουν. Ο κανόνας εκμάθησης μπορεί να εκφραστεί μαθηματικά ως εξής:

$$\Delta w_{ji} = \begin{cases} \eta(x_i - w_{ki}) \\ 0 \end{cases}$$

Όπου, η μεταβολή Δw_{ji} του βάρους από τη σύνδεση των νευρώνων j και i , ενώ ο ρυθμό εκμάθησης η αφορά στην σταθερή παράμετρο. Σε περίπτωση που ο νευρώνας νικήσει, τότε κάθε κόμβος εισόδου που αφορά στον συγκεκριμένο νευρώνα, αφήνει ένα ανάλογο του βάρους του, το οποίο επανακατανέμεται στους ενεργούς κόμβους εισόδου του, ενώ αντίθετα αν ο νευρώνας δεν επικρατήσει στον ανταγωνισμό, τότε παραμένει αδρανής.

1.5 Αλγόριθμοι βελτιστοποίησης

Οι βελτιστοποιητές εφαρμόζονται με σκοπό την μείωση του βαθμού απώλειας, τροποποιώντας τις τιμές των βαρών κατά την εκπαίδευση, ώστε το εκάστοτε δίκτυο να είναι αποδοτικό και να παράγει αξιόπιστα αποτελέσματα. Αυτό επιτυγχάνεται προσαρμόζοντας τον ρυθμό μάθησης για τις παραμέτρους του εκάστοτε μοντέλου με στόχο την ελαχιστοποίηση της συνάρτησης κόστους.

1.5.1 Κατάβαση Κλίσης (Gradient Descent)

Η κατάβαση κλίσης είναι ένας τρόπος για την ελαχιστοποίηση μιας αντικειμενικής συνάρτησης που παραμετροποιείται από τις παραμέτρους του μοντέλου παράμετροι, ενημερώνοντας τις παραμέτρους στην αντίθετη κατεύθυνση της κλίσης του αντικειμενική συνάρτηση στις παραμέτρους. Ο ρυθμός εκμάθησης καθορίζει το μέγεθος των βημάτων που κάνουμε για να φτάσουμε σε ένα τοπικό ελάχιστο.

1.5.2 Στοχαστική Κατάβαση Κλίσης (Stochastic Gradient Descent)

Ο αλγόριθμος Στοχαστική Κατάβαση Κλίσης (SGD) λύνει το πρόβλημα με τους περιττούς υπολογισμούς, καθώς ανανεώνει τις παραμέτρους για κάθε ένα από τα παραδείγματα εκπαίδευσης του συνόλου δεδομένων. Οι συχνά παρόμοιες παράγωγοι, πριν από κάθε ενημέρωση, εξαλείφονται, αυξάνοντας κατακόρυφα τη χρονική αποδοτικότητα του συστήματος μας. Αυτό μετατρέπει τον SGD σε κατάλληλο αλγόριθμο για μηχανική μάθηση εξαιτίας του μεγάλου όγκου των δεδομένων. Λόγω των συνεχών ενημερώσεων, μπορεί ο αλγόριθμος να είναι ευάλωτος σε ακραία δεδομένα εισόδου. Αυτό αποτελεί πλεονέκτημα, καθώς ο θόρυβος που εντοπίζεται στις ενημερώσεις, καταφέρνοντας να ξεφύγει από τα τοπικά ελάχιστα. Επιπλέον, η σταδιακή μείωση του ρυθμού εκμάθησης, οδηγεί το νευρωνικό δίκτυο σε ακόμα καλύτερα αποτελέσματα.

1.5.3 RMSProp

Ο RMSProp αποτελεί βελτιωμένη έκδοση του αλγορίθμου Adagrad, όπου η διαδικασία ενημέρωσης του RMSProp είναι παρόμοια με το Adagrad. Ο RMSProp, που διαιρεί το ρυθμό μάθησης για ένα βάρος με τον τρέχοντα μέσο όρο των μεγεθών των πρόσφατων κλίσεων για αυτό το βάρος. Το RMSProp δεν κάνει διόρθωση της προκατάληψης, η οποία προκαλεί σημαντικά προβλήματα όταν δημιουργείται το πρόβλημα της αραιής κλίσης.

1.5.4 ADAM

Ο βελτιστοποιητής ADAM είναι μια άλλη ευρέως χρησιμοποιούμενη μέθοδος που προσαρμόζει τον ρυθμό εκμάθησης προσαρμοστικά για κάθε παράμετρο. Ειδικότερα, αφορά στον συνδυασμό διαφορετικών μεθόδων βελτιστοποίησης κλίσης. Πέραν του ότι υπολογίζει έναν εκθετικά αποσυντιθέμενο μέσο όρο περασμένων τετραγωνικών διαβαθμίσεων, όπως συμβαίνει με τους Adadelta και RMSProp, αλλά επίσης ο παίρνει έναν εκθετικά αποσυντιθέμενο μέσο όρο παλαιότερων κλίσεων, ο οποίος είναι παρόμοιος με το Momentum.

2 ΧΡΟΝΟΣΕΙΡΕΣ

2.1 Εισαγωγή στις χρονοσειρές

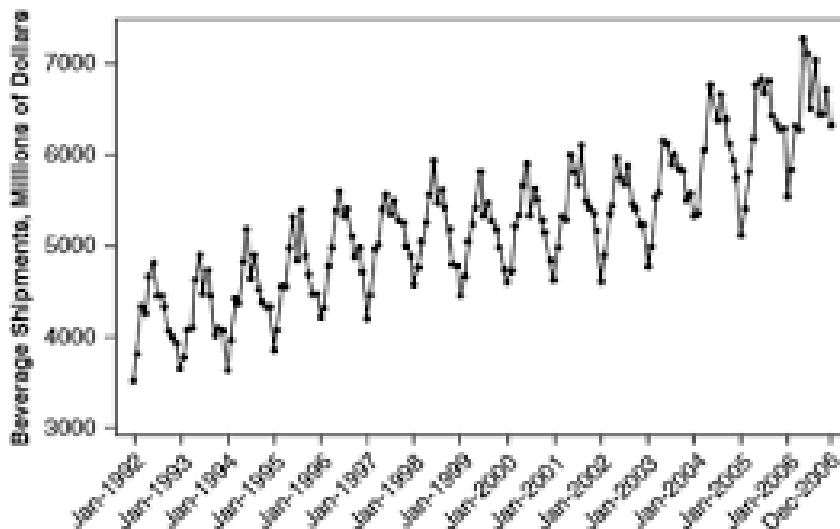
Η ανάλυση χρονοσειρών και η πρόβλεψη μελλοντικών τιμών τους αποτελούν ένα από τα πιο σημαντικά προβλήματα που απασχολεί επιστήμονες διαφόρων πεδίων, όπως υγείας, οικονομίας, περιβαλλοντικών και κοινωνικών επιστημών, επιχειρήσεων και βιομηχανίας, καθώς και επενδυτές που επιθυμούν να διαμορφώσουν τα χαρτοφυλάκιά τους με σκοπό την κερδοφορία. Όσων αφορά τα προβλήματα πρόβλεψης μπορούμε να τα ταξινομήσουμε σε βραχυπρόθεσμα, τα οποία διαρκούν λιγότερο του ενός έτους, σε μεσοπρόθεσμα που αφορούν στα ένα με δύο έτη και μακροπρόθεσμα που επεκτείνονται σε περισσότερα από δύο έτη. Ο λόγος της ανάγκης πρόβλεψης μελλοντικών τιμών

ή γεγονότων είναι ιδιαίτερα σημαντικός, καθώς αποτελεί σημαντικό παράγοντα στο στρατηγικό σχεδιασμό και στη λήψη αποφάσεων των επιχειρήσεων και επενδυτών.

Μια χρονοσειρά αφορά στο διαδοχικό σύνολο των τιμών ενός χαρακτηριστικού για ένα χρονικό διάστημα. Ειδικότερα, αποτελεί το σύνολο των δεδομένων, το οποίο καταγράφεται διαχρονικά για ίσες χρονικές περιόδους, εκφράζοντας την εξέλιξη των τιμών ενός χαρακτηριστικού ή αλλιώς μιας μεταβλητής. Στα μαθηματικά, η χρονοσειρά εκφράζεται από το σύνολο παρατηρήσεων N από τα ζεύγη των τιμών του χαρακτηριστικού A κατά τις χρονικές περιόδους T , δηλαδή $\{(t_1, a_1), (t_2, a_2), \dots, (t_n, a_n)\}$, όπου t αφορά τη χρονική στιγμή και a αφορά στην εκάστοτε τιμή παρατήρησης του χαρακτηριστικού A .

Οι χρονοσειρές διακρίνονται σε διακριτές και συνεχείς ανάλογα με την καταγραφή του συνόλου τιμών που έχει πραγματοποιηθεί στο χρόνο. Σε μια διακριτή χρονοσειρά, οι παρατηρήσεις καταγράφονται σε ορισμένες χρονικές στιγμές, όπως οι τιμές των μετοχών ανά ημέρα, ενώ στις συνεχείς οι τιμές καταγράφονται συνεχώς, όπως συμβαίνει στη μέτρηση της θερμοκρασίας ή στη συνεχή παρακολούθηση σεισμών. Επιπλέον, ανάλογα την ποσότητα των μεταβλητών που έχουν οι χρονοσειρές προς ανάλυση, διακρίνονται σε μονοδιάστατες όταν πρόκειται σε μία μεταβλητή και σε πολυδιάστατες για περισσότερες από μια μεταβλητές. Πιο συγκεκριμένα, η μέτρηση της θερμοκρασίας αποτελεί μονοδιάστατη χρονοσειρά, καθώς η παρατήρηση γίνεται σε μια μεταβλητή, ενώ αν συμπεριλάβουμε στην ανάλυση και τη μέτρηση της πίεσης, τότε αποτελεί πολυδιάστατη χρονοσειρά, αφού η παρατήρηση πραγματοποιείται σε δύο μεγέθη του ίδιου συστήματος, δηλαδή της θερμοκρασίας και της πίεσης.

Μια γενική εικόνα της χρονοσειράς αποτελεί η γραφική παράστασή της που δημιουργείται από την ένωση των ζευγών των παρατηρήσεων στο ορθοκανονικό σύστημα αξόνων, όπου στον άξονα x παρουσιάζονται οι χρονικές στιγμές και στον άξονα y οι τιμές του χαρακτηριστικού της χρονοσειράς, όπως φαίνεται στην Εικόνα 2.1.



Εικόνα 2.1: Απεικόνιση χρονοσειράς

Η γραφική παράσταση αποτελεί βασικό εργαλείο στην μελέτη και ανάλυση της χρονοσειράς, καθώς μπορούμε να εξάγουμε συμπεράσματα με γρήγορο και εύκολο τρόπο, διακρίνοντας με μια ματιά τη μεταβολή των τιμών με την πάροδο του χρόνου.

2.2 Συνιστώσες χρονοσειρών

Μια βασική προϋπόθεση στη μελέτη των χρονοσειρών αποτελεί η ανίχνευση των συνιστωσών της χρονοσειράς που αφορούν στην τάση (trend), στην εποχικότητα (seasonality), στην κυκλικότητα (cyclical) και στις τυχαίες τιμές (Random variations). Η γραφική παράσταση μας δίνει μια γρήγορη εικόνα σχετικά με την μελέτη των εν λόγω χαρακτηριστικών και την κατάσταση της χρονοσειράς.

2.2.1 Τάση (Trend)

Οι μεταβολές της μέσης τιμής μιας μεταβλητής κατά τη διάρκεια μιας ορισμένης χρονικής περιόδου ορίζεται ως τάση της χρονοσειράς. Αυτή η μακροχρόνια μεταβολή μπορεί να παρουσιάζει αύξηση, μείωση ή να παραμένει σταθερή και συνήθως εκτιμάται με μια ευθεία γραμμή ή καμπύλη. Στην περίπτωση που η τάση εμφανίζεται ως ευθείας γραμμή, τότε θεωρούμε ότι η τάση είναι γραμμική, ενώ στην περίπτωση της καμπύλης, η τάση είναι μη γραμμική. Επιπροσθέτως, η ύπαρξη επαρκούς αριθμού παρατηρήσεων είναι απαραίτητο κριτήριο για να εξάγουμε ασφαλή συμπεράσματα για το αν μια χρονοσειρά παρουσιάζει τάση ή όχι.

2.2.2 Εποχικότητα (Seasonality)

Οι εποχικές διακυμάνσεις που παρατηρούνται εντός του έτους σε μια χρονοσειρά ονομάζεται εποχικότητα. Η εποχικότητα αναφέρεται σε επαναλαμβανόμενα μοτίβα, τα οποία είναι γνωστά και σταθερά και πραγματοποιούνται με τον ίδιο τρόπο χρόνο με το χρόνο, όπως για παράδειγμα η κατανάλωση πετρελαίου θέρμανσης αυξάνεται κατά τους χειμερινούς μήνες, ενώ παρουσιάζει μείωση τους θερινούς μήνες. Τις περισσότερες φορές, η εποχικότητα διακρίνεται καλύτερα έπειτα από την απαλοιφή της τάσης και το μοτίβο της διακρίνεται σε προσθετικό ή πολλαπλασιαστικό. Στο προσθετικό μοτίβο, η εποχική μεταβολή έχει το ίδιο απόλυτο μέγεθος, ανεξάρτητα από το μέγεθος της τρέχουσας τιμής της χρονοσειράς, ενώ στο πολλαπλασιαστικό, έχει το ίδιο σχετικό μέγεθος σε σχέση με την τρέχουσα τιμή της χρονοσειράς.

2.2.3 Κυκλικότητα (Cyclicality)

Η κυκλικότητα εκφράζεται σαν κυκλικές διακυμάνσεις σε περίοδο μεγαλύτερη του έτους. Ειδικότερα, η μεταβολή οφείλεται σε εξωγενείς παράγοντες, όπως για παράδειγμα σε μια οικονομία, το Ακαθάριστο Εγχώριο Προϊόν μπορεί να παρουσιάζει περιοδικές ταλαντώσεις κατά τη διάρκεια μιας οικονομικής ύφεσης ή κρίσης που διαρκεί για μεγάλα χρονικά διαστήματα. Σε αντίθεση με την

εποχικότητα, η άνοδος και η πτώση των τιμών στις κυκλικές διακυμάνσεις δεν είναι σταθερή και συνήθως οι περιοδικές ταλαντώσεις είναι τουλάχιστον δυο ετών.

2.2.4 Τυχαίες τιμές (Random variations)

Οι παρατήσεις που είναι απομονωμένες και διακρίνονται ως απότομες αλλαγές στη χρονοσειρά αποτελούν τυχαίες τιμές της χρονοσειράς. Αναλυτικότερα, οι τυχαίες τιμές αποτελούν σημεία στο σύνολο δεδομένων που δεν ακολουθούν την τάση, την εποχικότητα και την κυκλικότητα της χρονοσειράς και διαφέρουν σημαντικά από τα υπόλοιπα δεδομένα. Διαγραμματικά μπορούμε να τις διακρίνουμε και ως ακραίες τιμές, δηλαδή σαν ξαφνικές αλλαγές που δεν επαναλαμβάνονται, όπως για παράδειγμα οι απεργίες, οι αρρώστιες, οι πόλεμοι, αποτελούν γεγονότα που είναι αδύνατο να προβλεφθούν. Η αντιμετώπιση τέτοιου είδους τιμών σε ένα σύνολο δεδομένων χρειάζεται ιδιαίτερη προσοχή, καθώς επηρεάζουν σημαντικά ένα μοντέλο πρόβλεψης, ενώ χρειάζεται εμπειρική ικανότητα και κριτική σκέψη για τη βέλτιστη διαχείρισή τους.

2.3 Στασιμότητα (Stationarity)

Με τον όρο στασιμότητα εννοούμε ότι δεν υπάρχει συστηματική αλλαγή του μέσου όρου και της διακύμανσης στο χρόνο, αντιθέτως τα εν λόγω στατιστικά μεγέθη της χρονοσειράς παραμένουν σταθερά. Στα προβλήματα του πραγματικού κόσμου, η υπόθεση ύπαρξης της μη στάσιμης χρονοσειράς είναι εξαιρετικά δύσκολο να υιοθετηθεί και επομένως χρειάζεται ειδική μεταχείριση, καθώς αποτελεί σημαντικό πρόβλημα τόσο στην ανάλυση όσο και στην πρόβλεψη των μελλοντικών τιμών. Η ύπαρξη της στάσιμης χρονοσειράς αποτελεί σημαντικό παράγοντα για την ανάπτυξη αποδοτικών μοντέλων πρόβλεψης, επομένως απαιτείται η μετατροπή μιας μη στάσιμης χρονοσειράς σε στάσιμη.

Τη στασιμότητα μπορούμε να τη διακρίνουμε σε αυστηρώς στάσιμη (strong stationarity) και σε ασθενώς στάσιμη (weak stationarity). Η αυστηρώς στάσιμη χρονοσειρά ορίζεται η κατανομή ενός αριθμού τυχαίων μεταβλητών της στοχαστικής διαδικασίας είναι ίδια όπως τις μετατοπίζουμε στον άξονα του δείκτη χρόνου. Με άλλα λόγια, η από κοινού κατανομή των τυχαίων μεταβλητών εξαρτάται μόνο από τις χρονικές μετατοπίσεις. Η ασθενώς στάσιμη χρονοσειρά ισχύει όταν η μέση τιμή και η αυτοδιακύμανση των τυχαίων παρατηρήσεων δεν εξαρτώνται από την χρονική στιγμή.

2.4 Αιτιοκρατία και στοχαστικότητα (Determinism and Stochasticity)

Στον πραγματικό κόσμο, όλες οι χρονοσειρές περιέχουν κάποιο είδος θορύβου και με αυτή την έννοια οι χρονοσειρές χαρακτηρίζονται στοχαστικές. Στόχος της ανάλυσης μιας χρονοσειράς είναι η αναγνώριση του αιτιοκρατικού μέρους του συστήματος που προκύπτει από τη χρονοσειρά γεγονός που αποτελεί και τη μεγαλύτερη πρόκληση κατά την ανάλυση. Επομένως, το σύστημα χαρακτηρίζεται στοχαστικό όταν η αιτιοκρατία που παράγει η χρονοσειρά εμπεριέχεται κρυμμένη

μέσα στο θόρυβο. Με άλλα λόγια, ένα αιτιοκρατικό μοντέλο μας επιτρέπει να προβλέψουμε μια τιμή με βεβαιότητα, γνωρίζοντας τις υπόλοιπες παραμέτρους. Για παράδειγμα, σε μια σχέση κεφαλαίου, επιτοκίου, χρόνου και τόκου μπορούμε να αναπτύξουμε ένα μαθηματικό μοντέλο, ώστε να προβλέψουμε τον τόκο όταν γνωρίζουμε το κεφάλαιο, το επιτόκιο και τον χρόνο. Αντιθέτως σε ένα στοχαστικό μοντέλο, για μια σχέση μεταξύ της τιμής και ζήτησης ενός προϊόντος είναι δύσκολο να προβλέψουμε τη ζήτηση, αφού μπορεί να εξαρτάται από άλλους παράγοντες εκτός της τιμής του προϊόντος, όπως η εμφάνιση υποκατάστατων προϊόντων στην αγορά μπορεί να προκαλέσει μείωση της ζήτησης ή το εισόδημα των καταναλωτών μπορεί εξίσου να επηρεάσει τη ζήτηση του προϊόντος.

2.5 Γραμμικότητα και μη γραμμικότητα (Linearity and Non-linearity)

Η εξέλιξη μιας μεταβλητής στο χρόνο μπορεί να ακολουθεί γραμμικό ή μη γραμμικό σύστημα, ανάλογα με τον τρόπο που οι ανεξάρτητες μεταβλητές επηρεάζουν τις διακυμάνσεις και αυξομειώσεις της πορείας της μεταβλητής του συστήματος. Ένα σύστημα εμφανίζει γραμμικότητα όταν όλοι οι όροι είναι γραμμικοί ως προς τις μεταβλητές της και διαγραμματικά δημιουργούν μια ευθεία γραμμή. Επομένως, αναφερόμαστε σε γραμμικό μοντέλο ονομάζεται όταν η τρέχουσα τιμή της χρονοσειράς αποτελεί γραμμική συνάρτηση των προηγούμενων παρατηρήσεών της. Σε αντίθετη περίπτωση το σύστημα αναφέρεται μη γραμμικό, ενώ μπορούμε να προβλέψουμε την τιμή της με μεγαλύτερη ακρίβεια συγκριτικά με το γραμμικό σύστημα, καθώς μπορεί να δημιουργεί ποικίλες καταστάσεις χωρίς την επίδραση του θορύβου.

2.6 Ανάλυση χρονοσειρών

Στόχος της ανάλυσης χρονοσειρών είναι η αναγνώριση του συστήματος που προκύπτει από τη χρονοσειρά, ώστε στη συνέχεια να δημιουργηθεί ένα ικανό μοντέλο πρόβλεψης των μελλοντικών τιμών της. Ιδανικά για την ανάπτυξη ενός ακριβή μοντέλου πρόβλεψης, πρέπει να θεωρήσουμε ότι οι τιμές της χρονοσειράς είναι ανεξάρτητες μεταξύ τους και το σύστημα που παρατηρούμε αποτελεί λευκό θόρυβο. Με τον όρο λευκό θόρυβο αναφερόμαστε σε μια χρονοσειρά που οι τιμές της είναι εντελώς τυχαίες και ταυτίζεται με τον όρο iid (independent and identically distributed).

Σε μια χρονοσειρά είναι δυνατό να μην υπάρχουν όλες οι προαναφερθέντα συνιστώσες παρά μόνο κάποιες από αυτές. Στη φάση της ανάλυσης μπορούμε να εφαρμόσουμε δυο μαθηματικά μοντέλα που υποδεικνύουν τον τρόπο που οι συνιστώσες τις χρονοσειράς καθορίζουν τις παρατηρήσεις της. Η εξέταση των συνιστωσών πραγματοποιείται με το προσθετικό και το πολλαπλασιαστικό μοντέλο. Ειδικότερα, το προσθετικό μοντέλο είναι κατάλληλο όταν το μέγεθος των εποχικών διακυμάνσεων δεν διαφέρει με το επίπεδο της χρονοσειράς, ενώ το πολλαπλασιαστικό μοντέλο είναι κατάλληλο όταν το μέγεθος των εποχικών μετατοπίσεων αυξάνεται ή μειώνεται ανάλογα με το μέσο επίπεδο της χρονοσειράς. Τα μαθηματικά υποδείγματα αποτυπώνονται ως εξής:

- Προσθετικό μοντέλο (Additive model):

$$Y_t = T_t + S_t + C_t + R_t$$

- Πολλαπλασιαστικό μοντέλο (Multiplicative model):

$$Y_t = T_t * S_t * C_t * R_t$$

Όπου, Y αφορά στην τιμή της μεταβλητής, T στην τάση, S στην εποχικότητα, C στην κυκλικότητα και R στις τυχαίες τιμές κατά τη χρονική στιγμή t .

2.7 Μέθοδοι Πρόβλεψης Χρονοσειρών

Για την πρόβλεψη των μελλοντικών τιμών μια μεταβλητής στο χρόνο μπορεί να εφαρμοστεί ένα μεγάλος αριθμός μεθόδων ανά περίπτωση, οι οποίες διακρίνονται σε τρεις κατηγορίες.

2.7.1 Μέθοδοι εξομάλυνσης (Smoothing methods)

Η κύρια ιδέα της εξομάλυνσης μιας χρονοσειράς είναι η απομάκρυνση ή η μείωση της επίδρασης του θορύβου από τη χρονοσειρά, ώστε να αναδειχθούν οι υπόλοιπες συνιστώσες της, όπως η τάση, η εποχικότητα και η κυκλικότητα για να αναγνωριστεί καλύτερα ο τρόπος συμπεριφοράς της. Υπάρχουν διάφορες τεχνικές εξομάλυνσης, οι οποίες διακρίνονται σε δύο βασικές κατηγορίες, κινητοί μέσοι όροι και την εκθετική εξομάλυνση.

2.7.1.1 Κινητοί Μέσοι Όροι

Απλός Μέσος Όρος: Πρόκειται για την απλούστερο τρόπο εξομάλυνσης, καθώς υπολογίζουμε τον μέσο όρο των παρατηρήσεων του δείγματος για να προβλέψουμε τη μελλοντική τιμή. Η διαδικασία της πρόβλεψης είναι αρκετά απλή και δεν χρησιμοποιείται ευρέως.

Κινητός Μέσος Όρος: Ένας εναλλακτικός τρόπος εξομάλυνσης και προτιμότερος από τον απλό κινητό μέσο αποτελεί ο κινητός μέσος όρος. Κατά τη συγκεκριμένη μέθοδο, υπολογίζουμε τον μέσο όρο των n πρόσφατων παρατηρήσεων του δείγματος και τον χρησιμοποιούμε για να προβλέψουμε την επόμενη περίοδο. Για κάθε νέα παρατήρηση που εισέρχεται στο δείγμα, καταργούμε την παλαιότερη παρατήρηση και επανυπολογίζουμε μια νέα μέση τιμή, η οποία αποτελεί την πρόβλεψη της επόμενης περιόδου.

Σταθμισμένος Κινητός Μέσος Όρος: Αποτελεί παρόμοια μέθοδο με τη αυτή του κινητού μέσου όρου με τη διαφορά ότι οι τιμές του υπολογισμού του μέσου όρου διαθέτουν βάρη, τα οποία διαφοροποιούνται ανάλογα την περίοδο. Ειδικότερα, τα βάρη που δίδονται είναι μεγαλύτερα για τις τιμές που είναι πιο κοντά στην υπό εξέταση περίοδο και μικρότερα για τις τιμές που βρίσκονται σε προγενέστερες περιόδους με αποτέλεσμα κάθε νέο σημείο που εισέρχεται στις παρατηρήσεις να συνεισφέρει σταδιακά στο μέσο όρο και να αφαιρείται από αυτόν προοδευτικά κατά την είσοδο νέων σημείων.

2.7.1.2 Εκθετική Εξομάλυνση

Απλή Εκθετική Εξομάλυνση: Οι τιμές των πιο πρόσφατων παρατηρήσεων έχουν μεγαλύτερη βαρύτητα σε σχέση με μεταγενέστερων περιόδων και οι συντελεστές βαρύτητας παρουσιάζουν εκθετική κατανομή. Η μέθοδος διαθέτει μια παράμετρο, η οποία ονομάζεται σταθερά εξομάλυνσης α και δέχεται τιμές από 0 έως 1, όπου όσο μεγαλύτερο είναι η τιμή του α τόσο μεγαλύτερο βάρος έχει η πιο πρόσφατη τιμή. Η μέθοδος απλής εκθετικής εξομάλυνσης απαιτεί μικρότερο αριθμό δεδομένων, συγκριτικά με τη μέθοδο του κινητού μέσου, και εφαρμόζεται σε χρονοσειρές που δεν περιέχουν τάση και εποχικότητα.

Εκθετική Εξομάλυνση με προσαρμογή στην Τάση: Η συγκεκριμένη μέθοδος εφαρμόζεται σε χρονοσειρές που περιέχουν τάση και όχι εποχικότητα και είναι γνωστή ως μέθοδος Holt. Ειδικότερα, χρησιμοποιεί τρεις διαφορετικές συναρτήσεις που εφαρμόζονται για την πρόβλεψη των μελλοντικών τιμών μιας χρονοσειράς. Η πρώτη αφορά στην εξομάλυνση των τιμών της χρονοσειράς, η δεύτερη λαμβάνει υπόψη την τάση, ενώ η τρίτη συνάρτηση αφορά στην πρόβλεψη. Η μέθοδος διαθέτει δυο παραμέτρους τη σταθερά εξομάλυνσης α και τη εξομάλυνση της τάσης β , οι οποίες δέχονται τιμές από 0 έως 1. και συγκριτικά με την απλή εκθετική εξομάλυνση, μας επιτρέπει να προβλέψουμε περισσότερες από μια μελλοντικές περιόδους.

Εκθετική Εξομάλυνση με προσαρμογή στην Τάση και στην Εποχικότητα: Η εν λόγω μέθοδος είναι γνωστή και ως μέθοδος Holt-Winters και αποτελεί επέκταση της εκθετικής εξομάλυνσης του Holt, η οποία αναπτύχθηκε για χρονοσειρές που περιέχουν τάση και εποχικότητα. Η μέθοδος χρησιμοποιεί μια επιπλέον συνάρτηση που αφορά στη συνιστώσα της εποχικότητας, ενώ χρησιμοποιεί τρεις παραμέτρους α , β και γ για την εξομάλυνση των τιμών της χρονοσειράς, της τάση και της εποχικότητας αντίστοιχα.

2.7.2 Μέθοδοι διάσπασης (Decomposition)

Στόχος της συγκεκριμένης μεθόδου είναι ο διαχωρισμός των συνιστωσών της χρονοσειράς για να προσδιοριστεί η επίδρασή τους στις τιμές της μεταβλητής. Κατά την αποσύνθεση της χρονοσειράς αναλύονται η τάση, η εποχικότητα, η κυκλικότητα και η μη κανονικότητα, ώστε να αναγνωριστεί ο τρόπος που διαμορφώνεται η εξέλιξη της χρονοσειράς και να προβούμε σε αξιόπιστες προβλέψεις. Όπως έχει ήδη αναφερθεί, οι χρονοσειρές διακρίνονται σε δυο μοντέλα, το προσθετικό και το πολλαπλασιαστικό. Αν οι συνιστώσες της χρονοσειράς αλληλοσυσχετίζονται, τότε αναφερόμαστε στο πολλαπλασιαστικό μοντέλο, ενώ σε αντίθετη περίπτωση στο προσθετικό μοντέλο. Ειδικότερα στο προσθετικό μοντέλο, αν η τάση μεταβάλλεται, τότε δεν επηρεάζεται την εποχικότητα ανεξάρτητα από τον ρυθμό μεταβολής της τάσης.

2.7.2.1 Ανάλυση εποχικότητας

Βασική προϋπόθεση για την ανάλυση της εποχικότητας είναι ο υπολογισμός των εποχικών δεικτών, τεχνική που μας επιτρέπει την απαλοιφή των εποχικών διακυμάνσεων από τις παρατηρήσεις της χρονοσειράς. Για τον υπολογισμό των εποχικών δεικτών αρχικά πρέπει να υπολογίσουμε τον απλό κινητό μέσο όρο που θα περιλαμβάνει ίδιο αριθμό περιόδων με την υπό μελέτη εποχικότητα. Αναλυτικότερα, αν τα δεδομένα έχουν συλλεχθεί σε μηνιαία βάση, τότε ο κινητός μέσος όρος θα πρέπει να δημιουργηθεί για δώδεκα περιόδους, ενώ αν η καταγραφή έχει πραγματοποιηθεί σε τρίμηνα, τότε ο κινητός μέσος όρος θα δημιουργηθεί για τέσσερα περιόδους. Ένα πρόβλημα που προκύπτει από τον υπολογισμό του κινητού μέσου όρου, όταν ο αριθμός των περιόδων του έτους είναι άρτιος αριθμός είναι η χρονική αντιστοιχία του με τις τιμές της χρονοσειράς. Για την αντιμετώπιση, υπολογίζουμε τον κεντρικό κινητό μέσο που προκύπτει από τον μέσο όρο δυο διαδοχικών τιμών του κινητού μέσου όρου. Εν συνεχεία, υπολογίζουμε τους εποχικούς δείκτες διαιρώντας τον απλό ή κεντρικό κινητό μέσο όρο ανά περίπτωση με τις τιμές της χρονοσειράς. Σε μαθηματική μορφή αποτυπώνεται ως εξής:

$$S_t = \frac{Y_t}{MA_t \text{ ή } CMA_t}$$

Όπου S_t αφορά στον εποχικό δείκτη της περιόδου, Y_t η τιμή της χρονοσειράς της συγκεκριμένης περιόδου, MA_t ο κινητός μέσος όρος σε περίπτωση που ο αριθμός των περιόδων είναι άρτιος και CMA_t ο κεντρικός κινητός μέσος όρος σε περίπτωση που ο αριθμός των περιόδων είναι ζυγός. Ο εποχικός δείκτης που προκύπτει αναφέρεται σε συγκεκριμένη περίοδο και πρέπει να αντιστοιχηθεί με τους εποχικούς δείκτες των επόμενων ετών. Επομένως, υπολογίζοντας τον μέσο όρο των αντίστοιχων εποχικών δεικτών, δημιουργούμε έναν τυπικό εποχικό δείκτη. Τέλος, για την απαλοιφή της εποχικότητας αρκεί να διαιρέσουμε τους τιμές της χρονοσειράς με τους προσαρμοσμένους τυπικούς δείκτες. Αυτό μαθηματικά εκφράζεται ως εξής:

$$SAY_t = \frac{Y_t}{SA_i}$$

Όπου SAY_t είναι η τιμή της χρονοσειράς απαλλαγμένη από τις εποχικές διακυμάνσεις, Y_t η τιμή της χρονοσειράς και SA_i ο προσαρμοσμένος τυπικός δείκτης.

2.7.2.2 Ανάλυση τάσης

Λαμβάνοντας υπόψιν το γραμμικό υπόδειγμα που εκφράζει την τάση, η μορφή της γραμμικής συνάρτησης εκφράζεται ως εξής:

$$Y_t = a + b_t + e_t$$

Όπου a η σταθερά της τάσης, b η κλίση της τάσης και e το τυχαία σφάλμα. Οι συντελεστές a και b μπορούν να προσδιοριστούν με τη μέθοδο των ελάχιστων τετραγώνων.

Για να εκτιμήσουμε την τάση υπολογίζουμε αρχικά τον συντελεστή b ως εξής:

$$b = \frac{n \sum XY - \sum X \sum Y}{n \sum X^2 - (\sum X)^2}$$

Στη συνέχεια, υπολογίζουμε τη σταθερά a από τον τύπο:

$$a = \frac{\sum Y}{n} - b \frac{\sum X}{n}$$

Τέλος, για να εκτιμήσουμε την τάση αρκεί να αντικαταστήσουμε τις τιμές a και b στο υπόδειγμα που εκφράζει την απλή γραμμική παλινδρόμηση, όπου σε μαθηματικούς όρους υπολογίζεται ως εξής:

$$T_t = a + bX_t$$

2.7.2.3 Ανάλυση κυκλικότητας

Για να μπορέσουμε να απομονώσουμε την κυκλικότητα της χρονοσειράς θα πρέπει να έχουν προηγηθεί η απαλλαγή της εποχικότητας και της τάσης. Αυτό πραγματοποιείται με την διαίρεση των απαλλαγμένων τιμών από την εποχικότητα με τις τιμές της τάσης, όπως υπολογίστηκαν ανωτέρω. Ειδικότερα, ο τύπος είναι ο ακόλουθος:

$$TAY_t = SAY_t + T_t$$

Όπου TAY_t αφορά στις τιμές που προκύπτουν από την απαλλοιφή της τάσης και της εποχικότητας, SAY_t στις τιμές απαλλαγμένες από την εποχικότητα και T_t στις τιμές της τάσης.

2.7.2.4 Ανάλυση μη κανονικότητας

Για να απομονώσουμε τη μη κανονικότητα της χρονοσειράς που αφορά στους τυχαίους παράγοντες και αποτελεί την τελευταία συνιστώσα της χρονοσειράς, αρκεί να διαιρέσουμε τις τιμές που είναι απαλλαγμένες από την εποχικότητα και την τάση με τον αντίστοιχο σταθμικό κεντρικό μέσο όρο και αποτυπώνεται ως εξής:

$$CAY_t = \frac{TAY_t}{WA_t}$$

Όπου CAY_t είναι οι τιμές απαλλαγμένες από την εποχικότητα, την τάση και την κυκλικότητα, TAY_t οι τιμές απαλλαγμένες από την εποχικότητα και την τάση και WA_t ο σταθμικός κεντρικός κινητός μέσος όρος.

2.7.3 Ανάλυση ARIMA (ARIMA Analysis)

Το μοντέλο ARIMA αποτελεί ακρόνυμο του Auto Regressive Integrated Moving Average και η μεθοδολογία ανάπτυξης και κατασκευής του είναι γνωστή και ως Box-Jenkins. Η διαδικασία δημιουργίας ενός μοντέλου ARIMA βασίζεται στη μεθοδολογία Box-Jenkins που αφορά στην επαναλαπτητική διαδικασία και περιλαμβάνει τρία στάδια.

1. **Ταυτοποίηση:** Το μοντέλο δέχεται τρεις παραμέτρους d , p και q που αφορούν στον αριθμό των διαφορών που απαιτούνται, ώστε η χρονοσειρά να μετατραπεί από μη στάσιμη σε στάσιμη, στην τάξη της αυτοπαλίνδρομης διαδικασίας και στην τάξη της διαδικασίας του κινητού μέσου. Ο καθορισμός της παραμέτρου d πραγματοποιείται είτε στατιστικό έλεγχο για μοναδιαία ρίζα, γνωστό ως Dickey Fuller test είτε κατασκευάζοντας τη συνάρτηση αυτοσυσχέτισης και την γραφική της παράσταση. Ο καθορισμός των παραμέτρων p και q υπολογίζονται με τη χρήση διαγραμμάτων της αυτοσυσχέτισης (ACF) και της μη αυτοσυσχέτισης (PACF).
2. **Εκτίμηση μοντέλου:** Το επόμενο στάδιο της διαδικασίας αφορά στην εκτίμηση των παραμέτρων q και p του υποδείγματος ARIMA. Αν η χρονοσειρά είναι αυτοπαλίνδρομη χρησιμοποιούμε τη μέθοδο των ελάχιστων τετραγώνων, ενώ αν είναι περιέχει όρους κινητών μέσων, τότε χρησιμοποιούμε μη γραμμικές μεθόδους εκτίμησης.
3. **Διαγνωστικός έλεγχος:** Στο συγκεκριμένο στάδιο πραγματοποιείται έλεγχος για το πόσο καλά ταιριάζει το μοντέλο στα δεδομένα μας. Αν το μοντέλο είναι ικανοποιητικό, τα κατάλοιπα πρέπει να ακολουθούν τη διαδικασία του λευκού θόρυβου, δηλαδή να μην υπάρχει αυτοσυσχέτιση μεταξύ τους. Ο έλεγχος των καταλοίπων πραγματοποιείται με τον έλεγχο Q των Box-Pierce ή με τον έλεγχο Ljung-Box.

3 ΑΝΑΠΤΥΞΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ ΓΙΑ ΠΡΟΒΛΕΨΗ

3.1 Εισαγωγή μοντελοποίησης Νευρωνικών Δικτύων

Στο παρόν κεφάλαιο, θα κατασκευάσουμε κυρίως αναδρομικά μοντέλα νευρωνικών δικτύων με διαφορετικές αρχιτεκτονικές, τα οποία θα εφαρμόσουμε σε πραγματικά δεδομένα με σκοπό να προβλέψουμε τις μελλοντικές τιμές των μετοχών του χρηματιστηρίου. Κατόπιν, θα μελετήσουμε την αποδοτικότητά τους διαφοροποιώντας τις παραμέτρους τους και θα εξάγουμε συμπεράσματα συγκρίνοντας τα αποτελέσματα που προκύπτουν. Για την επίτευξη αυτού, θα χρησιμοποιήσουμε την γλώσσα προγραμματισμού Python, η οποία περιέχει πληθώρα βιβλιοθηκών για την ανάλυση και επεξεργασία των δεδομένων και για την κατασκευή νευρωνικών δικτύων. Ειδικότερα, οι βιβλιοθήκες που θα χρησιμοποιήσουμε για την ανάλυση και επεξεργασία των δεδομένων είναι οι Pandas, NumPy και Matplotlib, ενώ για την κατασκευή των νευρωνικών δικτύων θα χρησιμοποιήσουμε τις TensorFlow και Keras.

3.2 Συλλογή δεδομένων

Το πρώτο στάδιο αφορά στη συλλογή των δεδομένων, όπου ανακτώνται όλα τα απαραίτητα δεδομένα που χρειαζόμαστε, τα οποία μπορεί να προέρχονται από διάφορες πηγές. Η ανάκτηση των δεδομένων για τις ιστορικές τιμές των μετοχών του χρηματιστηρίου μπορεί να γίνει με διάφορους τρόπους είτε μέσα από την επίσημη ιστοσελίδα <https://finance.yahoo.com/> που μπορούμε να τα αποθηκεύσουμε τοπικά σε μορφή csv είτε από διαθέσιμες βιβλιοθήκες, όπως `yfinance` και `pandas_datareader`. Στη συγκεκριμένη εργασία, δοκιμάστηκαν και οι τρεις τρόποι, αλλά για την κατασκευή των μοντέλων χρησιμοποιήθηκε η βιβλιοθήκη `pandas_datareader`, όπου αναπτύχθηκε μια αυτοματοποιημένη διαδικασία που αποθηκεύει τοπικά τις τιμές των μετοχών σε μορφή csv. Αυτό επιτεύχθηκε με τη χρήση της συνάρτησης `DataReader`, με παραμέτρους το σύμβολο της μετοχής που μας ενδιαφέρει, τη πηγή των δεδομένων και τις ημερομηνίες που μας ενδιαφέρουν.

Οι μετοχές προς μελέτη δεν επιλέχθηκαν τυχαία, αλλά με κριτήριο την πορεία τους στον χρόνο, ώστε να εργαστούμε επάνω σε χρονοσειρές που παρουσιάζουν διαφορετική συμπεριφορά, όπως θα δούμε στη συνέχεια. Στον παρακάτω πίνακα παρουσιάζονται οι μετοχές με το αντίστοιχο σύμβολο τους.

Πίνακας 3.1: Σύμβολα μετοχών

Σύμβολο μετοχής	Όνομα εταιρείας
GOOGL	Google Inc.
TSLA	Tesla Inc.
IBM	International Business Machine Corporation

Η πηγή δεδομένων που επιλέχθηκε είναι από τη yahoo και οι ημερομηνίες ξεκινούν από την 01/01/2016 έως την 30/09/2021. Στο σύνολο των δεδομένων που επιστρέφει η `DataReader` για κάθε μετοχή περιέχονται επτά στήλες (`Date`, `Open`, `High`, `Low`, `Close`, `Volume`, `Adj Close`), όπου η `Date` αφορά της ημερομηνίες και είναι `index` στο σύνολο δεδομένων, η `Name` το σύμβολο της μετοχής, η στήλη `Volume` είναι ακέραιου τύπου, ενώ υπόλοιπες στήλες είναι τύπου πραγματικών αριθμών.

3.3 Επεξεργασία δεδομένων

Κατόπιν ανάκτησης των δεδομένων, ακολουθεί η προ-επεξεργασία τους, το οποίο αποτελεί ένα από τα βασικότερα βήματα για την κατασκευή νευρωνικών δικτύων και γενικότερα μοντέλων μηχανικής μάθησης. Τα δεδομένα μας περιέχουν αριθμητικές τιμές και μερικά από τα προβλήματα που μπορεί να προκύπτουν και πρέπει να αντιμετωπιστούν με διαφορετική προσέγγιση ανά περίπτωση είναι ο εντοπισμός θορύβου, οι ακραίες και οι ελλιπείς τιμές. Με την έννοια του θορύβου εννοούμε ότι έχουν περαστεί εσφαλμένες τιμές κατά την καταγραφή είτε από απροσεξία είτε από υπολογιστικό λάθος. Οι ακραίες τιμές αποτελούν τιμές που είναι διαφορετικές σε σχέση με το σύνολο

των δεδομένων και μπορούν να εντοπιστούν εύκολα και γρήγορα με τη χρήση διαγραμμάτων, όπως το θηκόγραμμα. Οι ελλιπείς τιμές αφορούν σε κενές τιμές ή σε μερικές περιπτώσεις μπορεί να αποτελούν μια ή περισσότερες συγκεκριμένες τιμές στο σύνολο δεδομένων που αντιπροσωπεύουν τις ελλιπείς τιμές.

Στα σύνολο των δεδομένων δεν παρατηρήθηκε καμία ελλιπή ή ακραία τιμή γεγονός που κάνει τη διαδικασία πρόβλεψης αρκετά πιο εύκολη. Ακολουθώντας, από τα δεδομένα που έχουμε στη διάθεσή μας, μας ενδιαφέρουν μόνον η στήλη Close, δηλαδή η τιμή κλεισίματος. Επομένως αναφερόμαστε σε μονοδιάστατη χρονοσειρά, καθώς η παρατήρηση γίνεται σε μια μεταβλητή, όπως αναφέρθηκε στο προηγούμενο κεφάλαιο.

Στη συνέχεια, για την εκπαίδευση και αξιολόγηση των μοντέλων που θα κατασκευάσουμε θα χωρίσουμε τα δεδομένα σε δεδομένα εκπαίδευσης και ελέγχου. Τα δεδομένα που έχουμε στη διάθεσή μας αποτελούνται από 1453 παρατηρήσεις για κάθε μετοχή που σημαίνει ότι διαθέτουμε αρκετά μεγάλο αριθμό παρατηρήσεων για την εκπαίδευση του μοντέλου. Συνηθίζεται να χωρίζουμε το σύνολο δεδομένων κατά 70 ή 80 τοις εκατό σε δεδομένα εκπαίδευσης και το 30 ή 20 τοις εκατό αντίστοιχα σε δεδομένα ελέγχου, ώστε να υπάρχει αρκετός αριθμός σε δεδομένα για την εκπαίδευση των μοντέλων. Στη δική μας περίπτωση, θα χρησιμοποιήσουμε για δεδομένα ελέγχου τις παρατηρήσεις του έτους 2021 και από το έτος 2016 έως το 2020 για δεδομένα εκπαίδευσης. Αυτό σημαίνει ότι έχουμε αρκετά μεγάλο αριθμό παρατηρήσεων για εκπαίδευση, ώστε να αποφευχθεί το πρόβλημα της υπερπροσαρμογής (overfitting) και συγκεκριμένα 1259 για δεδομένα εκπαίδευσης και 194 για δεδομένα ελέγχου.

Για τη μείωση του εύρους των τιμών της τιμής κλεισίματος εφαρμόζουμε την τεχνική της κανονικοποίησης των δεδομένων χρησιμοποιώντας την κλάση MinMaxScaler από τη βιβλιοθήκη sklearn, καθώς το πρόβλημα που καλούμαστε να αντιμετωπίσουμε αφορά πρόβλημα παλινδρόμησης, όπου θα χρησιμοποιήσουμε τη σιγμοειδή συνάρτηση στον νευρώνα εξόδου για την πρόβλεψη των τιμών της εκάστοτε μετοχής.

Τέλος, δεδομένου ότι το πρόβλημα που καλούμαστε να αντιμετωπίσουμε είναι η πρόβλεψη μελλοντικών τιμών των μετοχών, χρειάζεται να δημιουργήσουμε ένα νέο σύνολο δεδομένων καθορίζοντας τι πρέπει να θυμάται το μοντέλο που θα κατασκευάσουμε προκειμένου να προβλέψει την επόμενη τιμή κλεισίματος. Ειδικότερα, θα ξεκινήσουμε χρησιμοποιώντας 60 timesteps που σημαίνει ότι το μοντέλο για κάθε χρονική στιγμή t το μοντέλο θα κοιτάζει τις 60 προηγούμενες τιμές και βάση της συμπεριφορά της πορείας τιμής κλεισίματος θα προβλέπει την αμέσως επόμενη τιμή. Ο καθορισμός του αριθμού των timesteps είναι σημαντικός, καθώς το μοντέλο πρέπει να έχει αρκετό αριθμό τιμών για να αναγνωρίσει τη συμπεριφορά της πορείας της μετοχής και να εκπαιδευτεί σωστά.

3.4 Ανάλυση δεδομένων

Σε αυτό το στάδιο εξετάζουμε τα χαρακτηριστικά που παρουσιάζουν οι χρονοσειρές, όπως συζητήθηκαν στο δεύτερο κεφάλαιο, προσπαθώντας να καταλάβουμε τη συμπεριφορά της τιμής κλεισίματος των μετοχών που αποτελεί το σημαντικότερο χαρακτηριστικό που ενδιαφέρει τους συμβαλλόμενους με την εκάστοτε μετοχή.

Στα διαγράμματα που ακολουθούν παρουσιάζεται η πορεία της τιμής κλεισίματος για τις προς μελέτη μετοχές που αναφέρθηκαν ανωτέρω, ενώ διακρίνουμε το σύνολο εκπαίδευσης μπλε χρωματισμό και ελέγχου με κόκκινο χρωματισμό.



Γράφημα 3.1: Τιμή κλεισίματος Google

Για τη μετοχή της Google, διακρίνουμε ανοδική πορεία της τιμής κλεισίματος, ενώ παρουσιάζεται μία απότομη πτώση της τιμής κατά την αρχή του έτους 2020, όπου στη συνέχεια συνεχίζει να αυξάνεται με μικρές διακυμάνσεις. Η τιμή της Google σε διάστημα τεσσάρων ετών φαίνεται να έχει σχεδόν εξαπλασιαστεί, καθώς το 2016 ξεκίνησε με περίπου 500 δολάρια και κλείνοντας το τρίτο τετράμηνο του 2021 η τιμή φτάνει περίπου τα 2,800 δολάρια.



Γράφημα 3.3: Τιμή κλεισίματος IBM

Η μετοχή της IBM φαίνεται να μην παρουσιάζει κάποια τάση ή εποχικότητα, ενώ διακρίνουμε ότι από το έτος 2016 έως την τελική ημερομηνία η τιμή της μετοχής έχει αυξηθεί ελάχιστα. Ακόμη, έχει σημειώσει μεγάλες διακυμάνσεις στο χρονικό διάστημα που μελετάμε διαδραματίζοντας μια ασταθή πορεία, γεγονός που αποτελεί πρόκληση για την πρόβλεψη της πορείας της τιμής στον χρόνο.



Γράφημα 3.2: Τιμή κλεισίματος Tesla

Η μετοχή της Tesla παρουσιάζει απότομη αυξητική τάση με κάποιες διακυμάνσεις το έτος 2020, και στη συνέχεια, ενώ επιτυγχάνει τη μέγιστη τιμή στις αρχές του 2021, εμφανίζει πτωτική και ασταθή πορεία.

3.5 Κατασκευή Νευρωνικών Δικτύων

Σε αυτό το σημείο θα κατασκευάσουμε αναδρομικά δίκτυα χρησιμοποιώντας τις κλάσεις SimpleRNN, LSTM και GRU από την Keras. Για κάθε μοντέλο χρησιμοποιούμε τον ίδιο αριθμό κύκλων εκπαίδευσης ίσο με 100, αλλά διαφοροποιείται από τον αριθμό των νευρώνων, τον κανόνα βελτιστοποίησης και τον αριθμό των κρυφών επιπέδων. Οι κανόνες βελτιστοποίησης που θα χρησιμοποιήσουμε είναι ο ADAM, όπου είναι ο συνηθέστερος, ο RMSProp που συστήνεται από την Keras, και ο SGD. Επιπλέον, ο αριθμός των παραδειγμάτων εκπαίδευσης ανά παρτίδα (batch size) ισούται με 32, το σύνολο δεδομένων εισόδου αποτελείται από τον αριθμό των timesteps, όπου ορίστηκε στις 60 προηγούμενες χρονικές στιγμές από την χρονική στιγμή t που θέλουμε να προβλέψουμε και το επίπεδο εξόδου προβλέπει την χρονική στιγμή t , δηλαδή την αμέσως επόμενη τιμή. Για την αξιολόγηση των μοντέλων χρησιμοποιήθηκε η ρίζα του μέσου τετραγωνικού σφάλματος (RMSE). Η υψηλή διάσταση των δικτύων μπορεί να οδηγήσει σε υπερπροσαρμογή, επομένως θα χρησιμοποιήσουμε την τεχνική της απομόνωσης τυχαίων νευρώνων με τη χρήση της κλάσης Dropout, όπου θα εισάγουμε ως παράμετρο την τιμή 0.2 ώστε να αγνοηθεί το 20 τοις εκατό των νευρώνων κατά τη διαδικασία της εκπαίδευσης.

3.5.1 Μετοχή της Google (GOOGL)

3.5.1.1 Νευρωνικά Δίκτυα με ένα επίπεδο

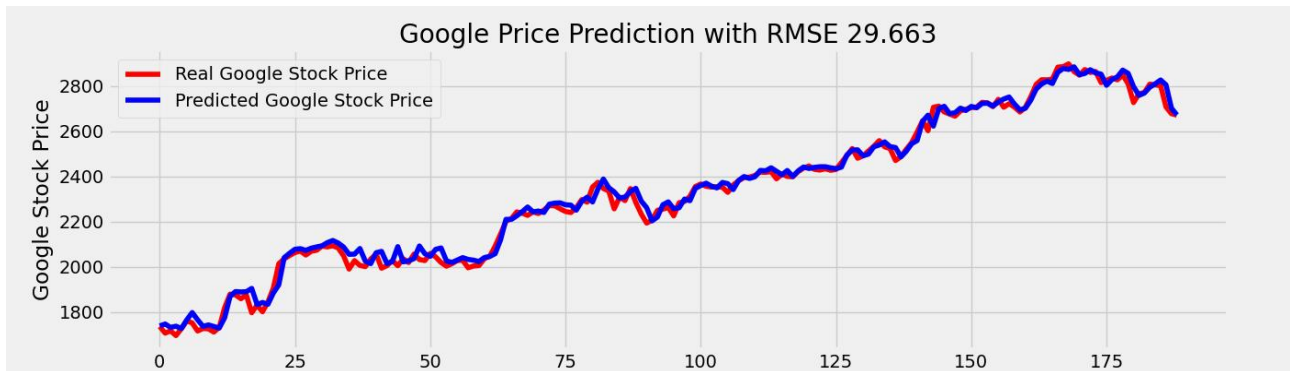
Τα αποτελέσματα του LSTM με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.2: Αποτελέσματα μοντέλου LSTM για την μετοχή της Google

Google LSTM - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	44.592	35.952	69.636
100	35.043	30.640	44.744
150	36.204	29.976	54.387
200	30.505	65.482	46.308
250	29.663	48.906	44.326

Από τα παραπάνω αποτελέσματα διακρίνουμε ότι για τον κανόνα βελτιστοποίησης ADAM όσο αυξάνεται ο αριθμός των νευρώνων τόσο μικρότερο είναι το σφάλμα, ενώ έχει και την καλύτερη απόδοση συγκριτικά με τους υπόλοιπους κανόνες βελτιστοποίησης. Επίσης, ο RMSProp είχε καλή απόδοση για μικρότερο αριθμό νευρώνων με βέλτιστο τους 150 που βρίσκεται αρκετά κοντά με τις προβλέψεις του ADAM. Ο SGD δεν είχε συγκριτικά καλές αποδόσεις, σημειώνοντας μεγαλύτερο σφάλμα, όμως παρουσίασε εξίσου καλή πρόβλεψη της τάσης της μετοχής. Στη συνέχεια,

παρουσιάζεται η σύγκριση μεταξύ του συνόλου δεδομένων ελέγχου και των προβλέψεων του βέλτιστου μοντέλου LSTM που κατασκευάστηκε με ένα επίπεδο για την Google.



Γράφημα 3.4: Αποτελέσματα μοντέλου LSTM για την μετοχή της Google με σφάλμα 29.633

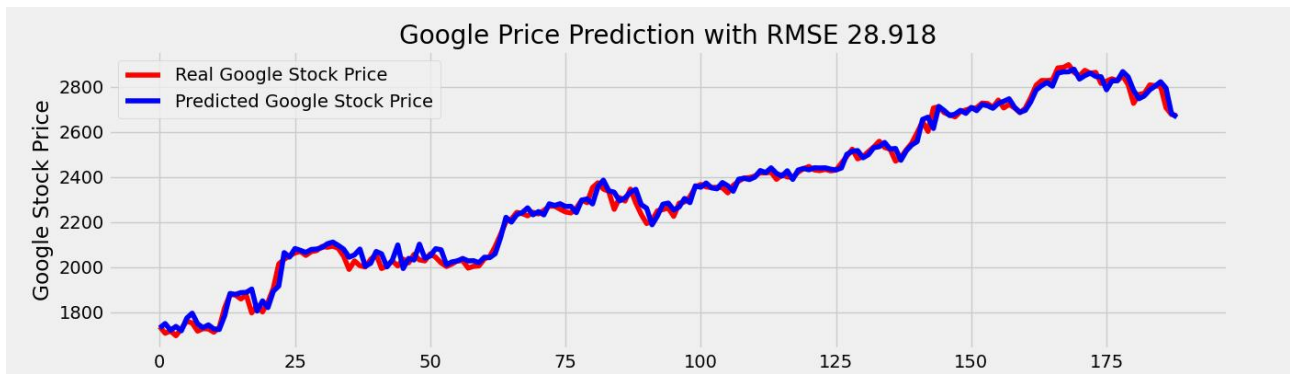
Παρατηρούμε ότι το μοντέλο έχει πολύ καλά αποτελέσματα, καθώς οι προβλέψεις σχεδόν ακολουθούν ακριβώς το σύνολο δεδομένων ελέγχου. Σε γενικές γραμμές, ένα απλό δίκτυο LSTM ανεξάρτητα από τον αριθμό των νευρώνων που δοκιμάστηκαν και τον κανόνα βελτιστοποίησης παρατηρούμε ότι τα σφάλματα είναι αρκετά κοντά μεταξύ τους και δεν διαφέρουν σημαντικά από το βέλτιστο μοντέλο.

Τα αποτελέσματα του GRU με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.3: Αποτελέσματα μοντέλου GRU για την μετοχή της Google

Google GRU - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	37.912	32.398	38.901
100	28.918	51.728	34.972
150	29.322	42.100	35.879
200	31.952	57.099	35.697
250	31.346	37.186	36.446

Το νευρωνικό δίκτυο GRU είχε καλύτερη απόδοση με 100 νευρώνες στο πρώτο επίπεδο και με κανόνα βελτιστοποίησης ADAM. Στο διάγραμμα φαίνεται η πρόβλεψη του μοντέλου σε σύγκριση με τα δεδομένα ελέγχου.



Γράφημα 3.5: Αποτελέσματα μοντέλου GRU για την μετοχή της Google με σφάλμα 29.332

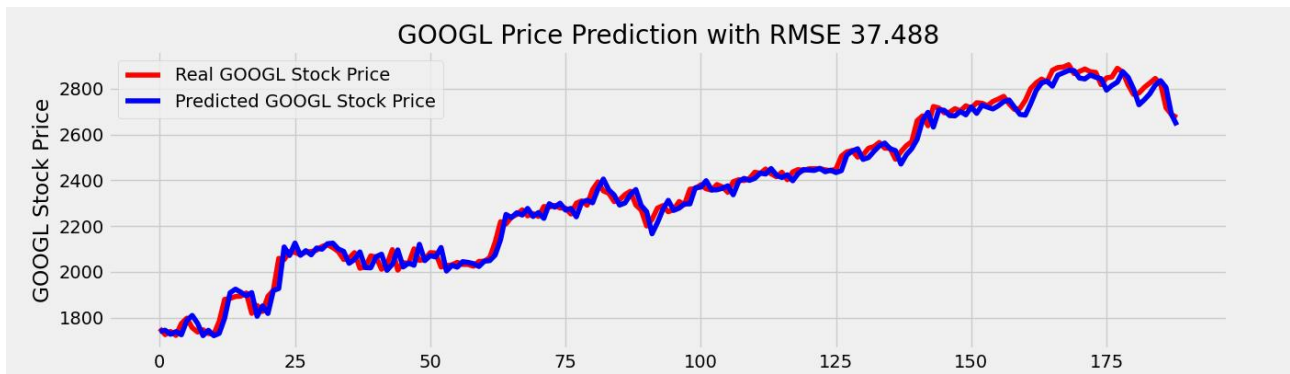
Από το διάγραμμα είναι προφανές ότι το μοντέλο GRU έχει εξίσου πολύ καλά αποτελέσματα με προβλέψεις που ακολουθούν σχεδόν ακριβώς το σύνολο δεδομένων για έλεγχο.

Συνεχίζοντας, τα αποτελέσματα του SimpleRNN με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.4: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Google

Google SimpleRNN - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	68.236	45.823	310.855
100	47.594	59.826	92.190
150	38.635	101.776	171.587
200	42.224	37.488	100.342
250	38.328	114.729	109.492

Από τα αποτελέσματα της κλάσης SimpleRNN φαίνεται ότι ο ADAM είχε τα καλύτερα αποτελέσματα στο σύνολό του, αλλά την καλύτερη απόδοση την είχε ο RMSProp με RMSE ίσο με 37.488. Όπως και στα προηγούμενα δίκτυα, παρατηρούμε ότι καλύτερα αποτελέσματα εξαγουν οι ADAM και RMSProp, που σημειώνουν σχετικά καλές προβλέψεις ανεξάρτητα από τον αριθμό των νευρώνων στο επίπεδο. Σημαντική διαφορά στην απόδοση που παρατηρούμε για το SimpleRNN δίκτυο είναι ότι δεν αποδίδει σχετικά καλά με τον SGD. Στο διάγραμμα διακρίνουμε τα αποτελέσματα των προβλέψεων του καλύτερου μοντέλου για το SimpleRNN δίκτυο συγκριτικά με το σύνολο δεδομένων ελέγχου.



Γράφημα 3.6: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Google με σφάλμα 37.488

Όπως διακρίνουμε και οπτικά, το βέλτιστο μοντέλο SimpleRNN παρουσιάζει παρόμοια πορεία στις προβλέψεις με το σύνολο δεδομένων για έλεγχο.

3.5.1.2 Νευρωνικά Δίκτυα με δύο επίπεδα

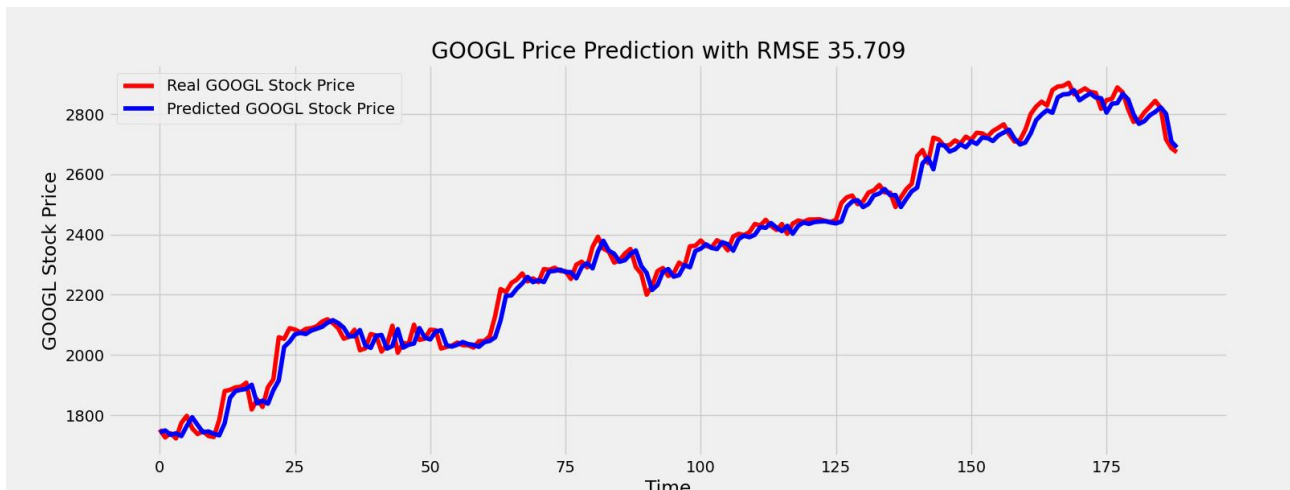
Σε αυτό το σημείο προσθέτουμε για τα παραπάνω μοντέλα ένα επιπλέον επίπεδο με διαφορετικό αριθμό νευρώνων.

Τα αποτελέσματα του LSTM φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.5: Αποτελέσματα μοντέλου LSTM για την μετοχή της Google

Google LSTM - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	44.141	44.138	132.11
50, 100	51.600	86.492	132.057
50, 150	57.813	99.152	110.460
100, 100	70.078	40.401	86.879
100, 150	35.709	89.329	88.014

Από τα αποτελέσματα βλέπουμε ότι το βέλτιστο μοντέλο LSTM με δυο επίπεδα είναι με 100 νευρώνες στο πρώτο και 150 στο δεύτερο επίπεδο και με τον κανόνα ADAM. Στο διάγραμμα βλέπουμε τις προβλέψεις του βέλτιστου μοντέλου.



Γράφημα 3.7: Αποτελέσματα μοντέλου LSTM για την μετοχή της Google με σφάλμα 35.709

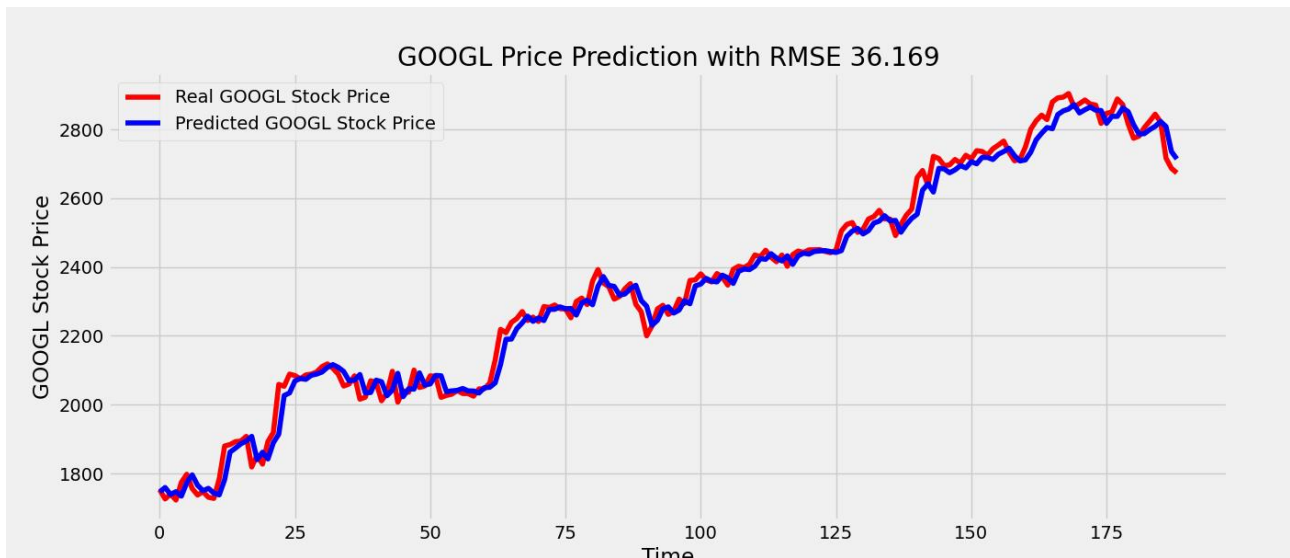
Το μοντέλο δείχνει να ανταποκρίνεται πολύ καλά στο σύνολο δεδομένων ελέγχου, αναγνωρίζοντας σε πολύ μεγάλο βαθμό τη συμπεριφορά της χρονοσειράς εντοπίζοντας και τις μικρές διακυμάνσεις που προκύπτουν.

Τα αποτελέσματα του GRU φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.6: Αποτελέσματα μοντέλου GRU για την μετοχή της Google

Google GRU - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	67.948	57.162	58.789
50, 100	43.419	54.071	59.479
50, 150	40.773	40.312	52.941
100, 100	72.398	53.663	54.528
100, 150	36.169	112.559	54.861

Το βέλτιστο GRU μοντέλο για δυο επίπεδα είναι με 100 νευρώνες στο πρώτο επίπεδο, 150 νευρώνες στο δεύτερο επίπεδο και με τον κανόνα ADAM.



Γράφημα 3.8: Αποτελέσματα μοντέλου GRU για την μετοχή της Google με σφάλμα 36.169

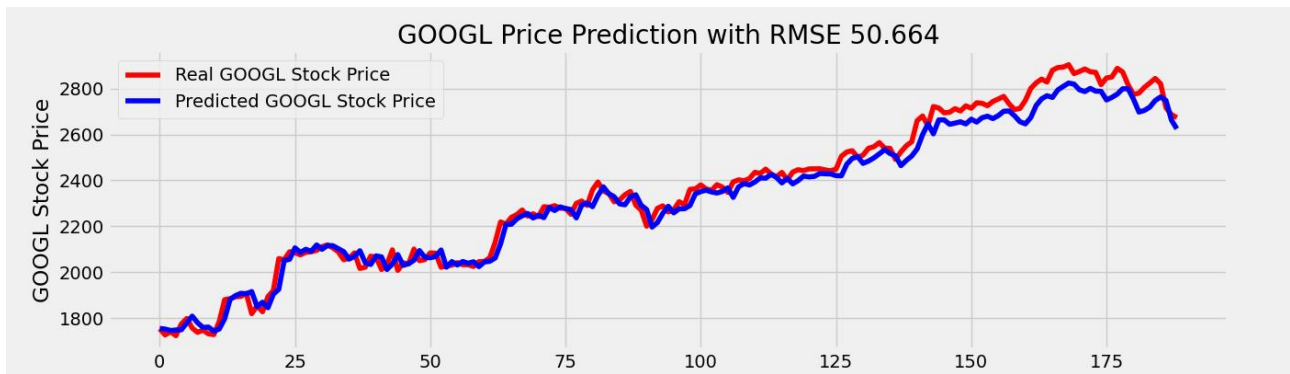
Όπως φαίνεται, οι προβλέψεις του μοντέλου έχουν επιτύχει σε μεγάλο βαθμό την πραγματική πορεία των τιμών της μετοχής Google.

Τα αποτελέσματα του SimpleRNN φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.7: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Google

Google SimpleRNN - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	124.748	102.474	298.447
50, 100	116.485	231.057	207.097
50, 150	149.498	176.905	163.909
100, 100	50.664	125.585	258.539
100, 150	76.361	387.287	200.250

Το μοντέλο SimpleRNN δείχνει να μην ανταποκρίνεται το ίδιο καλά σε σχέση με τα μοντέλα LSTM και GRU όσο αυξάνεται η διάσταση και η πολυπλοκότητα των μοντέλων. Παρόλα αυτά το βέλτιστο μοντέλο έχει επιτύχει πολύ καλά αποτελέσματα με 100 νευρώνες στα δυο επίπεδα και κανόνα ADAM.



Γράφημα 3.9: Αποτελέσματα μοντέλου *SimpleRNN* για την μετοχή της *Google* με σφάλμα 50.664

Όπως διακρίνουμε από το διάγραμμα, το μοντέλο έχει αρχίσει να αποκλείει περισσότερο από τις πραγματικές τιμές μετά από 135 ημέρες και για αυτόν τον λόγο το σφάλμα αρχίζει να αυξάνεται με την πάροδο του χρόνου.

3.5.1.3 Νευρωνικά Δίκτυα με τρία επίπεδα

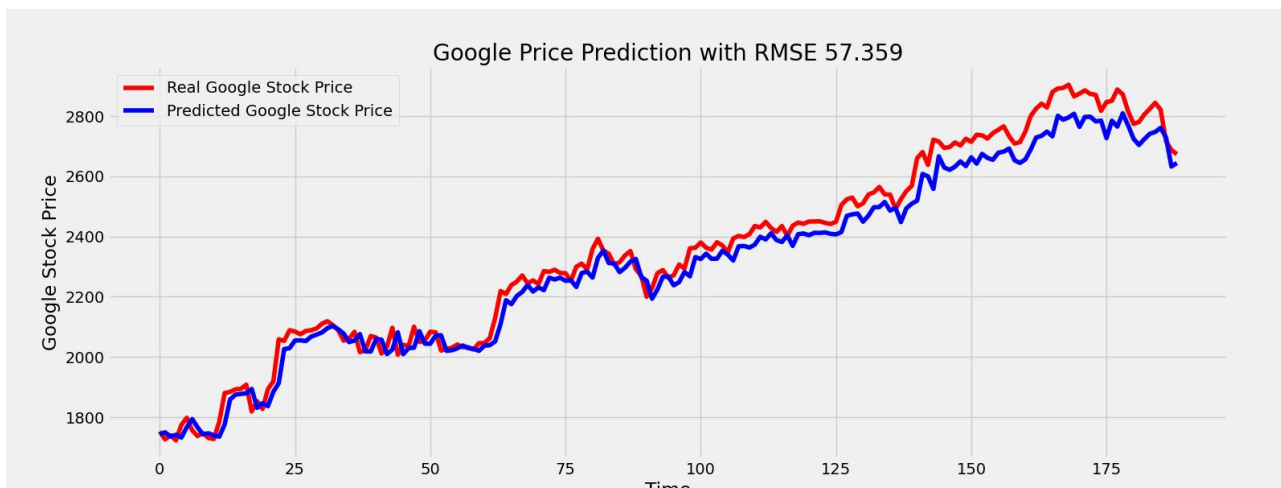
Συνεχίζοντας, προσθέτουμε ένα τρίτο επίπεδο στα μοντέλα, αυξάνοντας κατά αυτόν τον τρόπο της πολυπλοκότητά τους και τις διαστάσεις τους.

Τα αποτελέσματα του LSTM με τρία επίπεδα παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.8: Αποτελέσματα μοντέλου *LSTM* για την μετοχή της *Google*

Google LSTM - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	76.670	121.308	181.831
50, 50, 150	102.080	129.006	173.979
50, 100, 100	63.131	174.353	145.305
50, 100, 150	74.815	213.801	135.686
100, 100, 100	57.359	99.456	139.739
100, 100, 150	100.64	148.105	135.887

Από τα αποτελέσματα διακρίνουμε ότι το σφάλμα έχει αρχίσει να αυξάνεται λόγω των υψηλών διαστάσεων του μοντέλου. Παρόλα αυτά το βέλτιστο μοντέλο έχει πολύ καλά αποτελέσματα με 100 νευρώνες στα επίπεδα με τον ADAM.



Γράφημα 3.10: Αποτελέσματα μοντέλου LSTM για την μετοχή της Google με σφάλμα 57.359

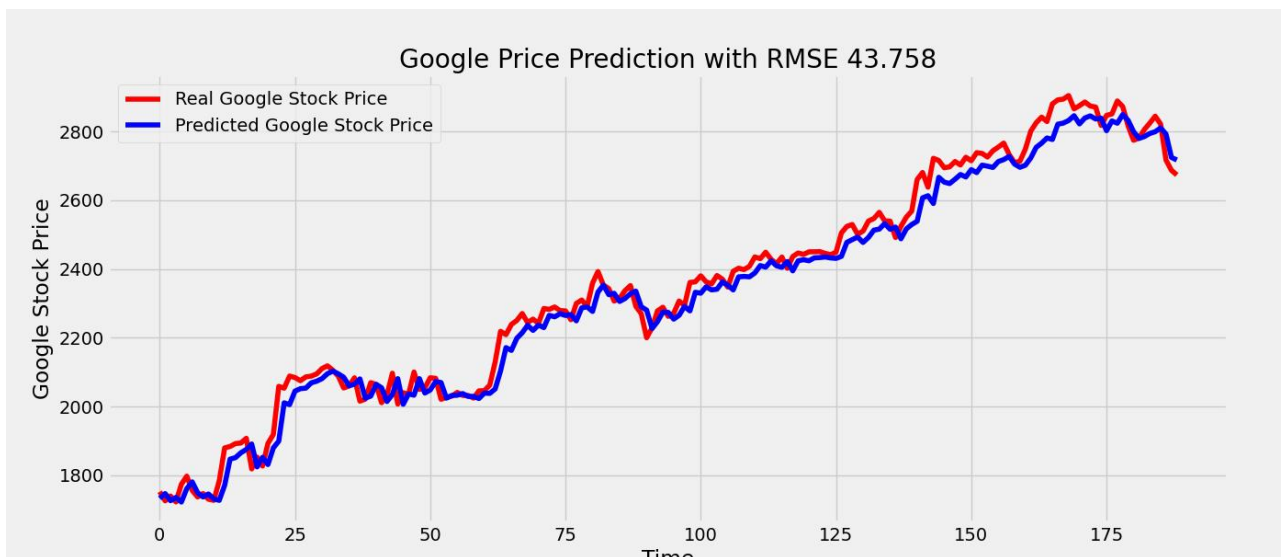
Από το διάγραμμα παρατηρούμε ότι το βέλτιστο μοντέλο επιτυγχάνει πολύ καλές προβλέψεις, αλλά με την πάροδο του χρόνου η απόκλιση μεταξύ των προβλέψεων και των πραγματικών τιμών αρχίζει να μεγαλώνει.

Τα αποτελέσματα του μοντέλου GRU φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.9: Αποτελέσματα μοντέλου GRU για την μετοχή της Google

Google GRU - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	94.111	104.754	86.905
50, 50, 150	76.864	218.335	77.295
50, 100, 100	64.210	51.782	79.592
50, 100, 150	49.481	59.048	71.279
100, 100, 100	43.758	79.525	70.989
100, 100, 150	62.748	45.376	55.387

Συγκριτικά με το LSTM το νευρωνικό δίκτυο GRU ανταποκρίνεται πολύ καλά σε γενικές γραμμές με όλους τους κανόνες βελτιστοποιήσεις εκτός από μερικές περιπτώσεις. Φαίνεται ότι όσο μεγαλώνει ο αριθμός των νευρώνων στα επίπεδα τόσο καλύτερα αποτελέσματα έχει το μοντέλο. Το βέλτιστο μοντέλο είναι το ίδιο με το LSTM, δηλαδή με 100 νευρώνες ανά επίπεδο και με τον ADAM, αλλά με αρκετά μικρότερο σφάλμα.



Γράφημα 3.11: Αποτελέσματα μοντέλου GRU για την μετοχή της Google με σφάλμα 43.758

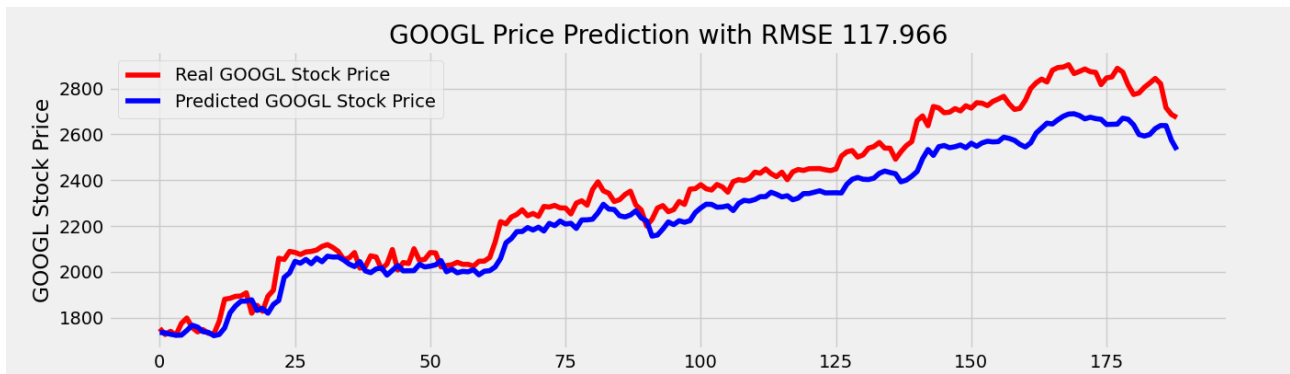
Στο διάγραμμα του βέλτιστου μοντέλου GRU διακρίνουμε ότι προβλέπει με επιτυχία την τάση και τις διακυμάνσεις των πραγματικών τιμών με μικρές αποκλίσεις.

Τα αποτελέσματα του μοντέλου SimpleRNN φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.10: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Google

Google SimpleRNN - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	143.207	361.015	195.208
50, 50, 150	145.359	342.579	269.605
50, 100, 100	117.966	197.214	252.176
50, 100, 150	124.759	307.199	293.822
100, 100, 100	140.606	110.400	436.282
100, 100, 150	174.540	288.316	711.263

Από τον παραπάνω πίνακα φαίνεται ότι το SimpleRNN δεν έχει καλά αποτελέσματα συγκριτικά με τα μοντέλα LSTM και GRU. Ειδικότερα, έχει σημειώσει προβλέψεις με υψηλά σφάλματα, με τα καλύτερα αποτελέσματα τον ADAM. Για τον SGD όσο αυξάνεται η πολυπλοκότητα του μοντέλου τόσο αδυνατεί να προβλέψει την πορεία της μετοχής Google. Το βέλτιστο μοντέλο είναι με τον ADAM και με 50 νευρώνες στο πρώτο επίπεδο και 100 νευρώνες στα υπόλοιπα δυο επίπεδα.



Γράφημα 3.12: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Google με σφάλμα 117.966

Από το διάγραμμα βλέπουμε ότι το μοντέλο προβλέπει επιτυχώς την πορεία της μετοχής, αλλά αποκλείει όσο αυξάνεται ο χρονικό ορίζοντας.

3.5.2 Μετοχή της TESLA (TSLA)

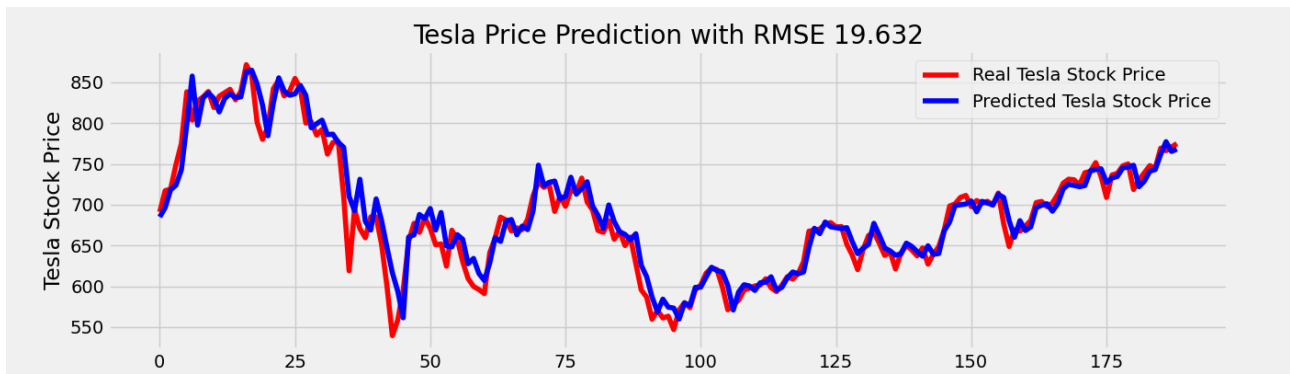
3.5.2.1 Νευρωνικά Δίκτυα με ένα επίπεδο

Τα αποτελέσματα του LSTM με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.11: Αποτελέσματα μοντέλου LSTM για την μετοχή της Tesla

Tesla LSTM - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	26.457	32.781	45.979
100	23.151	30.811	46.563
150	24.863	51.074	43.367
200	22.812	22.491	52.747
250	22.074	19.632	54.647

Όπως διακρίνουμε από τα αποτελέσματα ότι με τον κανόνα ADAM το δίκτυο LSTM είχε αρκετά καλή και σταθερή απόδοση παρά τον διαφορετικό αριθμό νευρώνων, αλλά την καλύτερη απόδοση την σημείωσε με τον RMSProp με αριθμό νευρώνων 250. Στο διάγραμμα που ακολουθεί διακρίνουμε οπτικά τα αποτελέσματα του συγκεκριμένου μοντέλου.



Γράφημα 3.13: Αποτελέσματα μοντέλου LSTM για την μετοχή της Tesla με σφάλμα 19.632

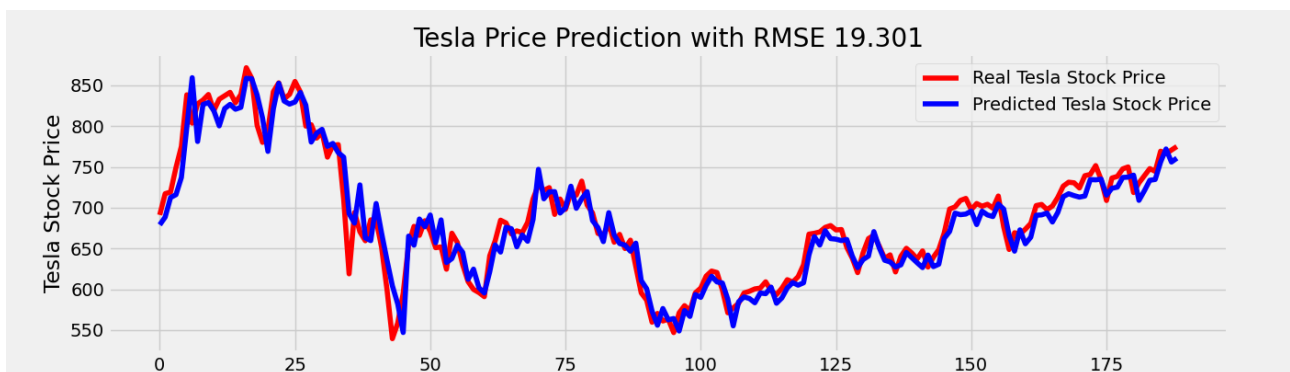
Από το διάγραμμα είναι φανερό ότι το βέλτιστο μοντέλο εντοπίζει τις διακυμάνσεις της τιμής κλεισίματος της μετοχής Tesla με πολύ χαμηλό σφάλμα.

Τα αποτελέσματα του GRU με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.12: Αποτελέσματα μοντέλου GRU για την μετοχή της Tesla

Tesla GRU - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	38.981	19.746	28.836
100	34.891	30.480	33.075
150	27.120	43.513	36.610
200	23.012	19.301	36.190
250	20.603	25.868	38.374

Το βέλτιστο μοντέλο είναι επίσης με τον RMSProp, αλλά με 200 νευρώνες στο επίπεδο. Παρατηρούμε ότι όλα τα μοντέλα που αναπτύχθηκαν έχουν επιτύχει πολύ καλές προβλέψεις. Στο διάγραμμα βλέπουμε τα αποτελέσματα του βέλτιστου μοντέλου.



Γράφημα 3.14: Αποτελέσματα μοντέλου GRU για την μετοχή της Tesla με σφάλμα 19.301

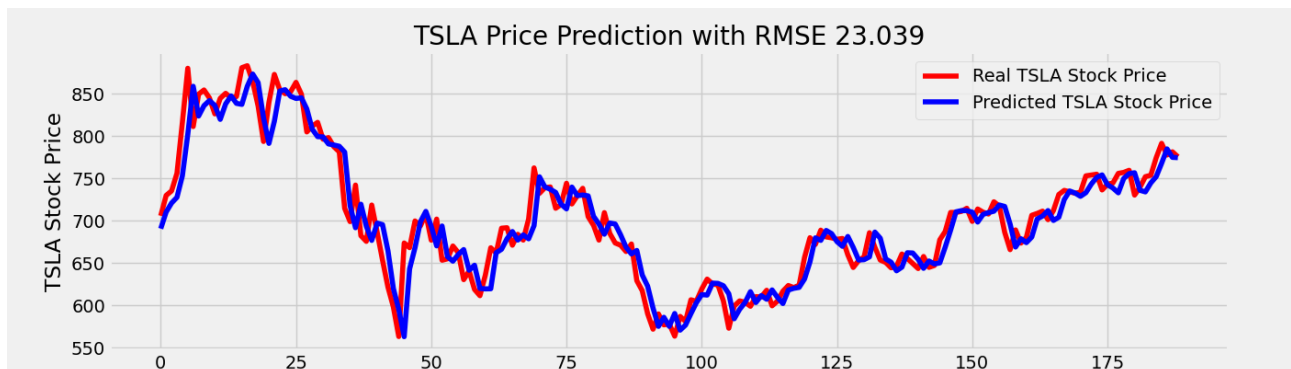
Τα αποτελέσματα του βέλτιστου μοντέλου είναι παρόμοια με το LSTM και επιτυγχάνει εξίσου πολύ καλά αποτελέσματα συγκρίνοντας τις πραγματικές τιμές.

Τα αποτελέσματα του SimpleRNN με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.13: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Tesla

Tesla SimpleRNN - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	23.039	23.901	37.885
100	27.796	25.065	47.217
150	25.551	24.671	34.419
200	23.508	25.730	37.003
250	34.658	35.537	42.289

Από τον παραπάνω πίνακα διακρίνουμε ότι τα αποτελέσματα του SimpleRNN με διαφορετικό αριθμό νευρώνων στο επίπεδο και διαφορετικό κανόνα βελτιστοποίησης ότι είναι πολύ καλά συγκριτικά τόσο με τα δεδομένα ελέγχου όσο και με τα παραπάνω δίκτυα που αναπτύχθηκαν. Για το συγκεκριμένο νευρωνικό δίκτυο το βέλτιστο μοντέλο αποτελείται από 50 νευρώνες και κανόνα βελτιστοποίησης ADAM.



Γράφημα 3.15: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Tesla με σφάλμα 23.039

Από το διάγραμμα βλέπουμε ότι και αυτό το μοντέλο αναγνωρίζει ορθά τις διακυμάνσεις και την πορεία της τιμή κλεισίματος Tesla με μικρό σφάλμα.

3.5.2.2 Νευρωνικά Δίκτυα με δύο επίπεδα

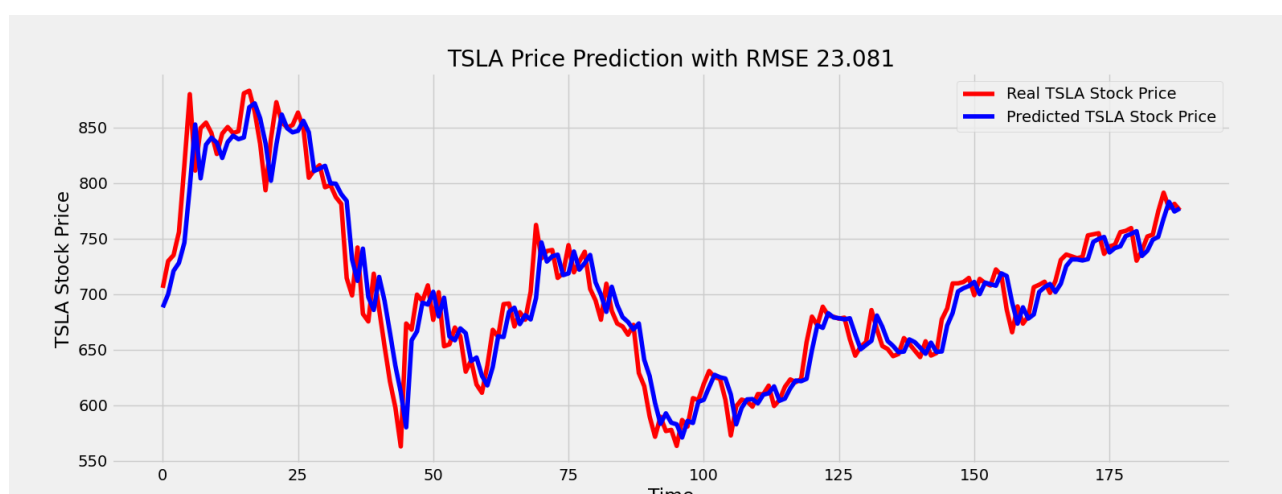
Σε αυτό το σημείο προσθέτουμε για τα παραπάνω μοντέλα ένα επιπλέον επίπεδο με διαφορετικό αριθμό νευρώνων.

Τα αποτελέσματα του LSTM φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.14: Αποτελέσματα μοντέλου LSTM για την μετοχή της Tesla

Tesla LSTM - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	26.783	26.116	47.774
50, 100	26.486	50.250	56.743
50, 150	47.562	24.869	59.985
100, 100	36.075	23.081	53.058
100, 150	27.525	24.145	53.939

Από τα αποτελέσματα βλέπουμε ότι το βέλτιστο μοντέλο LSTM με δύο επίπεδα είναι με 100 νευρώνες και στα δυο επίπεδα και με τον κανόνα RMSProp, όπως και για την μετοχή της Google. Στο διάγραμμα βλέπουμε τις προβλέψεις του βέλτιστου μοντέλου.



Γράφημα 3.16: Αποτελέσματα μοντέλου LSTM για την μετοχή της Tesla με σφάλμα 23.081

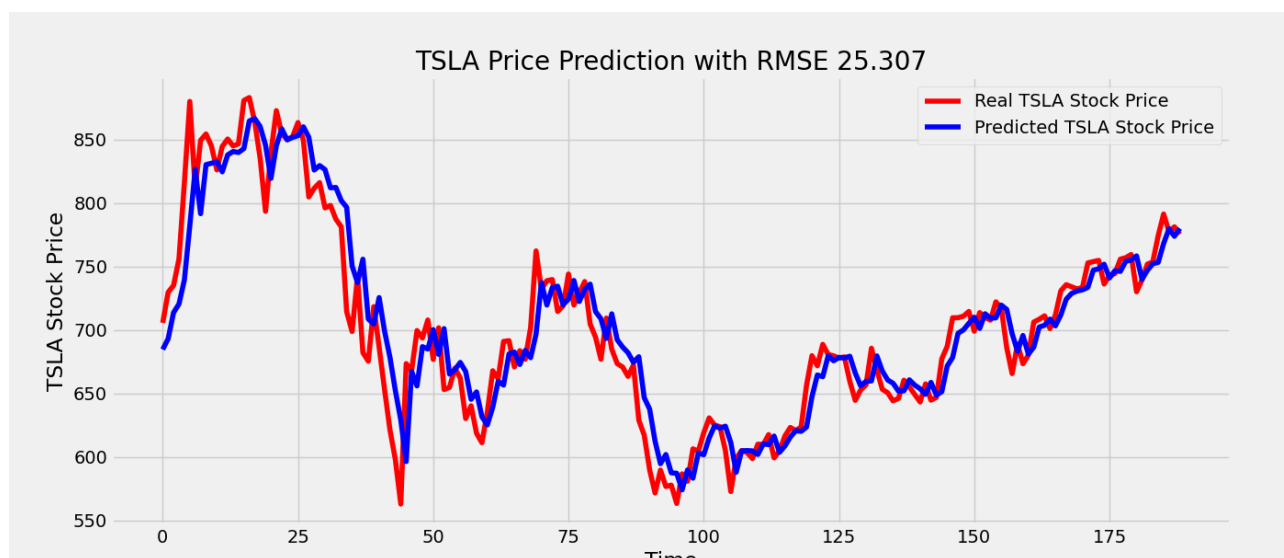
Το μοντέλο να ανταποκρίνεται πολύ καλά στο σύνολο δεδομένων ελέγχου, αναγνωρίζοντας σε πολύ μεγάλο βαθμό τη συμπεριφορά της χρονοσειράς, εντοπίζοντας τις διακυμάνσεις στην πορεία της μετοχής με μικρό σφάλμα.

Τα αποτελέσματα του GRU φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.15: Αποτελέσματα μοντέλου GRU για την μετοχή της Tesla

Tesla GRU - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	28.930	40.423	31.969
50, 100	36.121	33.869	32.110
50, 150	30.471	47.133	31.680
100, 100	25.307	34.644	34.876
100, 150	27.125	53.248	37.125

Το βέλτιστο GRU μοντέλο για δυο επίπεδα είναι με 100 νευρώνες σε κάθε επίπεδο και με τον κανόνα ADAM.



Γράφημα 3.16: Αποτελέσματα μοντέλου GRU για την μετοχή της Tesla με σφάλμα 25.307

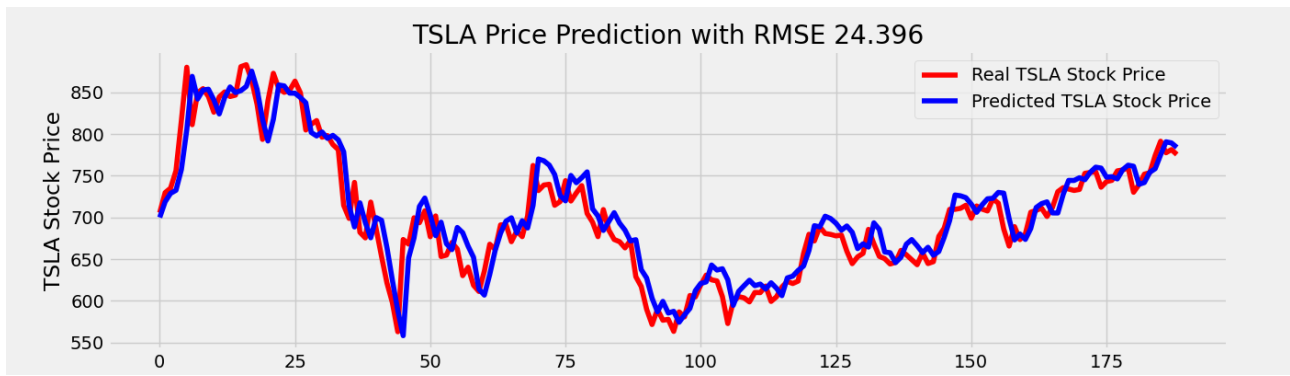
Όπως φαίνεται, οι προβλέψεις του μοντέλου έχουν επιτύχει σε μεγάλο βαθμό την πραγματική πορεία των τιμών της μετοχής Tesla και τις διακυμάνσεις της.

Τα αποτελέσματα του SimpleRNN φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.16: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Tesla

Tesla SimpleRNN - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	54.099	47.271	55.651
50, 100	24.396	56.852	46.458
50, 150	25.558	27.288	37.301
100, 100	26.427	24.666	40.427
100, 150	29.455	28.002	46.587

Το μοντέλο SimpleRNN δείχνει να ανταποκρίνεται εξίσου καλά σε σχέση με τα μοντέλα LSTM και GRU. Το βέλτιστο μοντέλο είναι με τον ADAM, με 50 νευρώνες στο πρώτο επίπεδο και 100 νευρώνες στο δεύτερο επίπεδο.



Γράφημα 3.17: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Tesla με σφάλμα 24.396

Όπως διακρίνουμε από το διάγραμμα, το βέλτιστο μοντέλο εντοπίζει εξίσου τη συμπεριφορά της πορείας της μετοχής με σχετικά μικρές αποκλίσεις από τις πραγματικές τιμές.

3.5.2.3 Νευρωνικά Δίκτυα με τρία επίπεδα

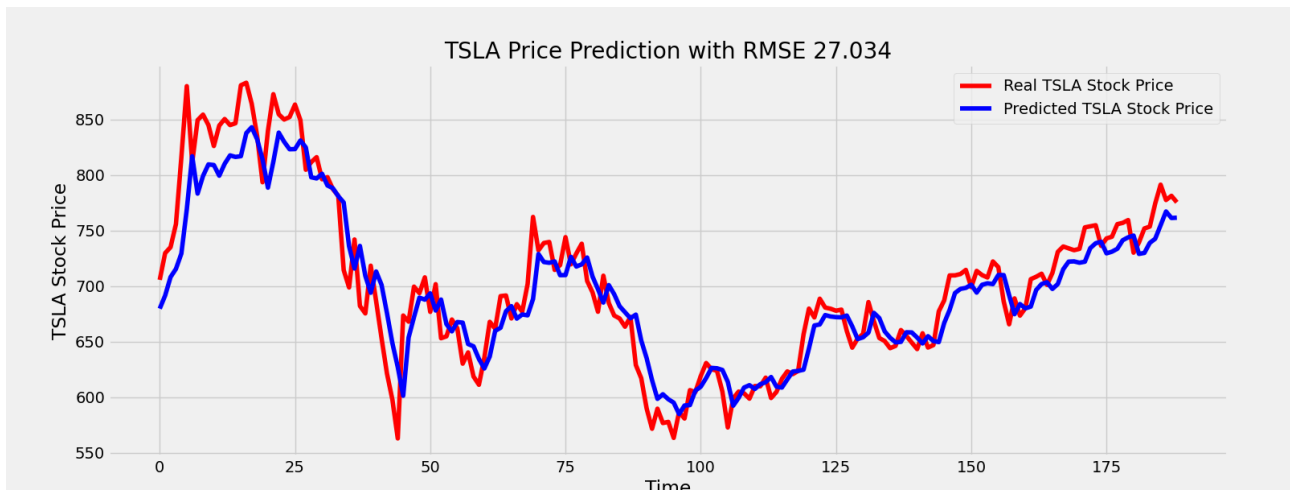
Σε αυτό το σημείο, προσθέτουμε το τρίτο επίπεδο στα παραπάνω μοντέλα, αυξάνοντας της πολυπλοκότητά τους.

Τα αποτελέσματα του LSTM με τρία επίπεδα παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.17: Αποτελέσματα μοντέλου LSTM για την μετοχή της Tesla

Tesla LSTM - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	34.748	34.960	67.065
50, 50, 150	46.847	59.422	66.211
50, 100, 100	35.898	30.919	62.334
50, 100, 150	30.153	32.842	64.016
100, 100, 100	27.034	37.983	67.836
100, 100, 150	28.907	59.702	65.997

Από τα αποτελέσματα το βέλτιστο GRU μοντέλο είναι με τον ADAM και 100 νευρώνες στα επίπεδά του. Επίσης, για τον SGD η απόδοση του μοντέλου φαίνεται αν μην επηρεάζεται ιδιαίτερα από τον διαφορετικό αριθμό νευρώνων έχοντας τη χαμηλότερη απόδοση συγκριτικά με τους ADAM και RMSProp.



Γράφημα 3.18: Αποτελέσματα μοντέλου LSTM για την μετοχή της Tesla με σφάλμα 27.034

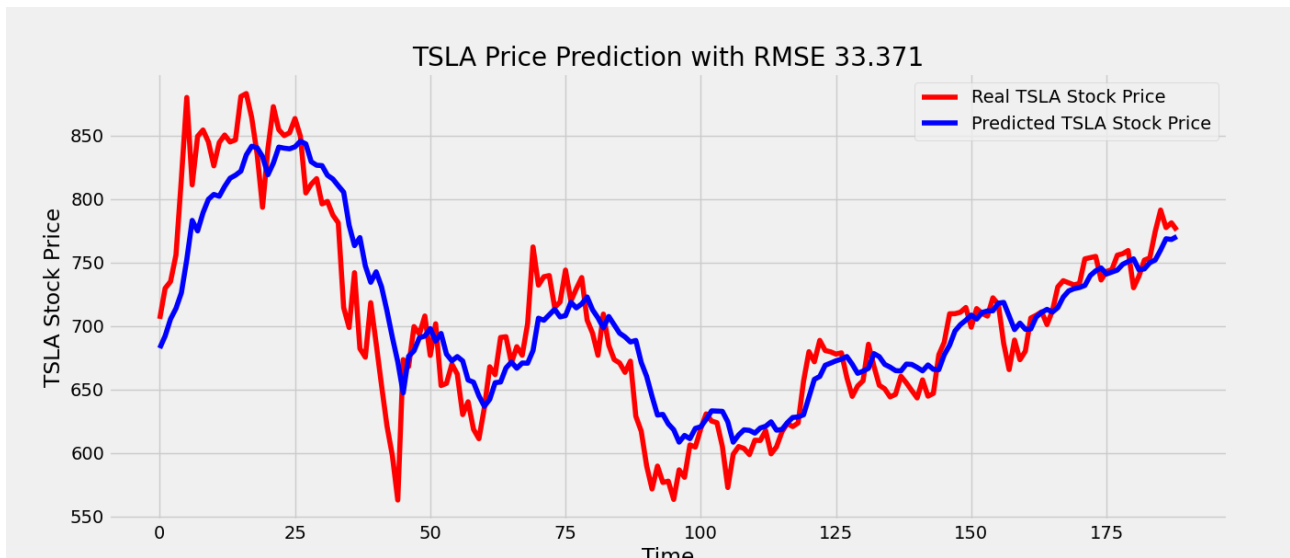
Από το διάγραμμα είναι προφανές ότι το βέλτιστο μοντέλο επιτυγχάνει πολύ καλές προβλέψεις, αλλά λόγω της πολυπλοκότητας αδυνατεί να προβλέψει τις μεγάλες διακυμάνσεις που προκύπτουν από την αρχή του έτους 2021, αλλά στη συνέχεια η απόκλιση αυτή αρχίζει να μειώνεται.

Τα αποτελέσματα του μοντέλου GRU φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.18: Αποτελέσματα μοντέλου GRU για την μετοχή της Tesla

Google GRU - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	45.177	33.371	37.082
50, 50, 150	37.997	39.340	37.242
50, 100, 100	46.763	47.932	39.430
50, 100, 150	45.489	101.361	40.948
100, 100, 100	53.311	79.327	39.312
100, 100, 150	39.761	38.626	40.745

Το νευρωνικό δίκτυο GRU ανταποκρίνεται πολύ καλά σε γενικές γραμμές με όλους τους κανόνες βελτιστοποιήσεις εκτός από μερικές περιπτώσεις. Το βέλτιστο μοντέλο είναι με τον κανόνα RMSProp και με 50 νευρώνες στο πρώτο, 50 στο δεύτερο και 100 στο τρίτο επίπεδο.



Γράφημα 3.19: Αποτελέσματα μοντέλου GRU για την μετοχή της Tesla με σφάλμα 33.371

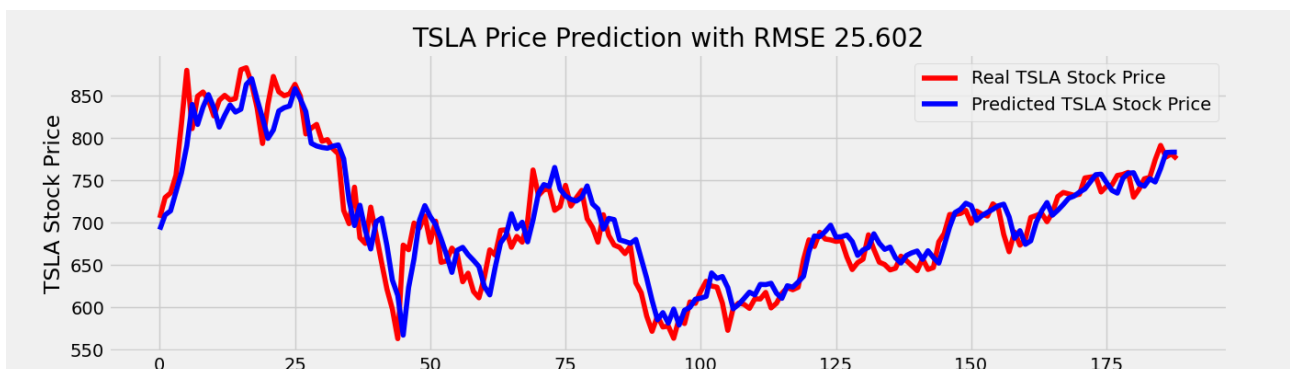
Στο διάγραμμα του βέλτιστου μοντέλου GRU διακρίνουμε ότι προβλέπει με επιτυχία την πορεία της τιμής κλεισίματος, αλλά σημειώνει σημαντική απόκλιση από τις πραγματικές τιμές για όλες τις περιπτώσεις που έχουμε δει μέχρι στιγμής μεταξύ των βέλτιστων μοντέλων.

Τα αποτελέσματα του μοντέλου SimpleRNN φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.19: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Tesla

Tesla SimpleRNN - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	78.088	50.614	73.593
50, 50, 150	31.373	45.990	40.920
50, 100, 100	26.519	48.340	41.217
50, 100, 150	25.602	33.219	187.141
100, 100, 100	29.742	30.665	72.990
100, 100, 150	41.173	141.375	133.348

Από τον παραπάνω πίνακα παρατηρούμε ότι το βέλτιστο μοντέλο SimpleRNN είναι με νευρώνες 50, 100 και 150 στα αντίστοιχα επίπεδα και κανόνα ADAM.



Γράφημα 3.20: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της Tesla με σφάλμα 25.602

Από το διάγραμμα βλέπουμε ότι το μοντέλο προβλέπει επιτυχώς την πορεία της μετοχής, καθώς επίσης έχει και την καλύτερη απόδοση από τα ανωτέρω μοντέλα LSMT και GRU με τρία επίπεδα.

3.5.3 Μετοχή της IBM (IBM)

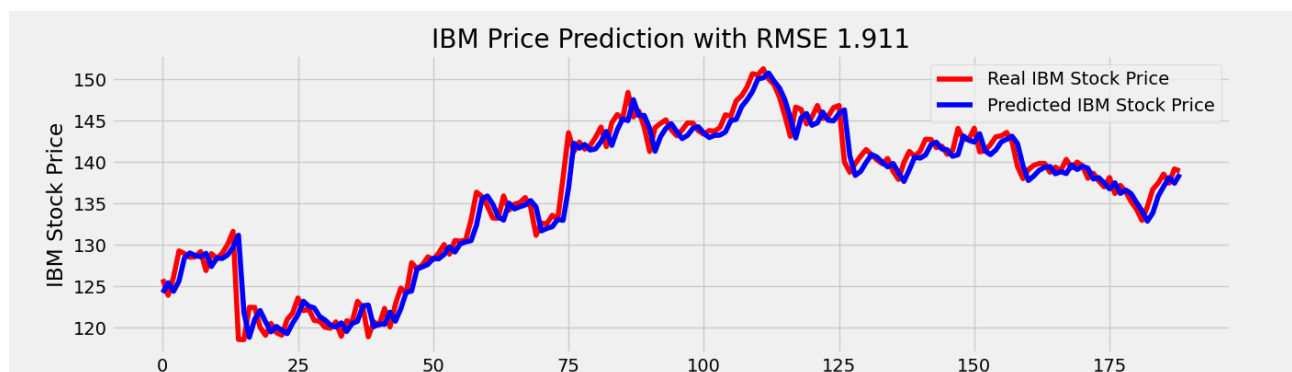
3.5.3.1 Νευρωνικά Δίκτυα με ένα επίπεδο

Τα αποτελέσματα του LSTM με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.20: Αποτελέσματα μοντέλου LSTM για την μετοχή της IBM

IBM LSTM - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	2.043	1.911	2.174
100	1.916	2.535	2.225
150	2.056	2.056	2.180
200	2.106	2.117	2.174
250	2.037	1.942	2.200

Την καλύτερη απόδοση την έχει το μοντέλο με κανόνα βελτιστοποίησης RMSProp και 50 νευρώνες στο επίπεδο. Όλα τα μοντέλα έχουν πολύ καλές αποδόσεις, όπως παρατηρήθηκε και στις προηγούμενες μετοχές.



Γράφημα 3.21: Αποτελέσματα μοντέλου LSTM για την μετοχή της IBM με σφάλμα 1.911

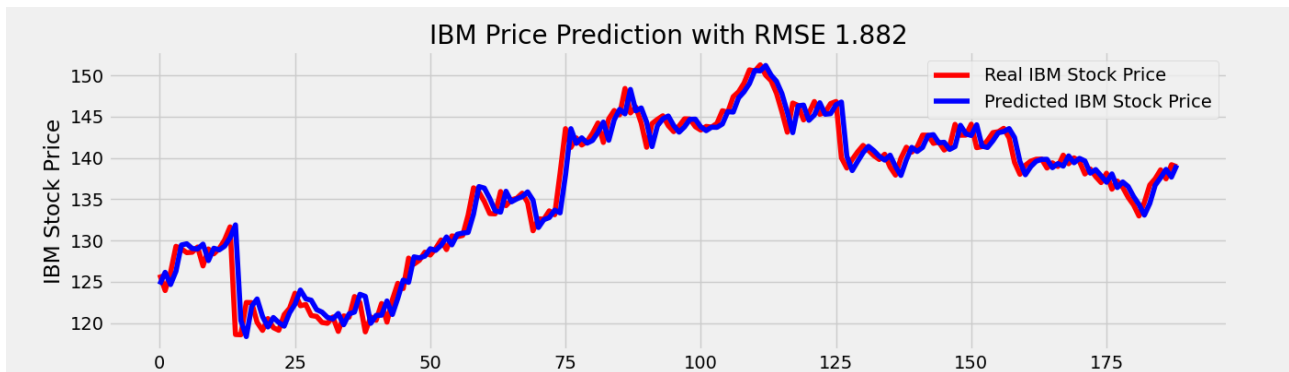
Από το διάγραμμα βλέπουμε ότι το μοντέλο προβλέπει την πορεία της τιμής κλεισίματος σε μεγάλο βαθμό συγκριτικά με τις πραγματικές τιμές σημειώνοντας πολύ μικρό σφάλμα.

Τα αποτελέσματα του GRU με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.21: Αποτελέσματα μοντέλου GRU για την μετοχή της IBM

IBM GRU - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	2.073	2.074	1.988
100	2.052	2.070	1.987
150	1.882	2.533	1.982
200	2.118	2.669	2.000
250	2.180	2.278	2.028

Εξίσου πολύ καλά αποτελέσματα έχει και το μοντέλο GRU σε όλα τις περιπτώσεις που δοκιμάστηκε. Το βέλτιστο μοντέλο είναι με 150 νευρώνες στο επίπεδο και με κανόνα ADAM.



Γράφημα 3.22: Αποτελέσματα μοντέλου GRU για την μετοχή της IBM με σφάλμα 1.882

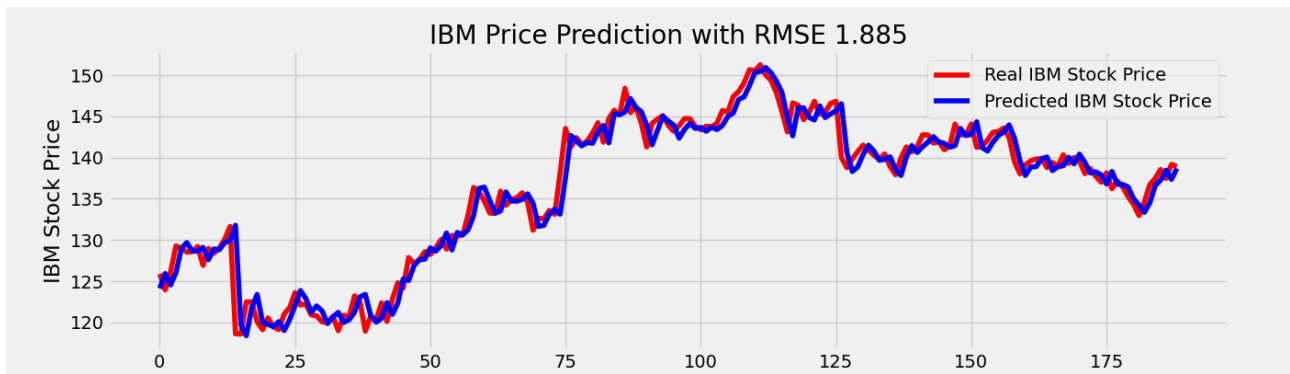
Στο διάγραμμα διακρίνουμε και οπτικά την απόδοση του μοντέλου GRU που σχεδόν εφάπτεται ακριβώς επάνω στα πραγματικά δεδομένα.

Τα αποτελέσματα του SimpleRNN με ένα επίπεδο παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.22: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της IBM

IBM SimpleRNN - One Layer - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50	1.918	2.268	2.905
100	1.900	1.926	2.193
150	1.885	2.839	2.163
200	1.918	2.357	2.658
250	1.947	6.374	2.311

Από τα αποτελέσματα βλέπουμε ότι το μοντέλο έχει πολύ καλές αποδόσεις παρόμοιες με τα LSTM και GRU για τον κανόνα ADAM, με τα σφάλματα να μην ξεπερνούν τα 2 δολάρια. Το βέλτιστο μοντέλο είναι με 200 νευρώνες στο επίπεδο με τον ADAM.



Γράφημα 3.23: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της IBM με σφάλμα 1.885

Από το διάγραμμα βλέπουμε τις προβλέψεις του βέλτιστου μοντέλου για την τιμή κλεισίματος της IBM. Συγκριτικά με τα προηγούμενα βέλτιστα μοντέλα, παρατηρούμε οπτικά την μικρή απόκλιση που έχουν οι προβλέψεις με τα δεδομένα ελέγχου.

3.5.3.2 Νευρωνικά Δίκτυα με δύο επίπεδα

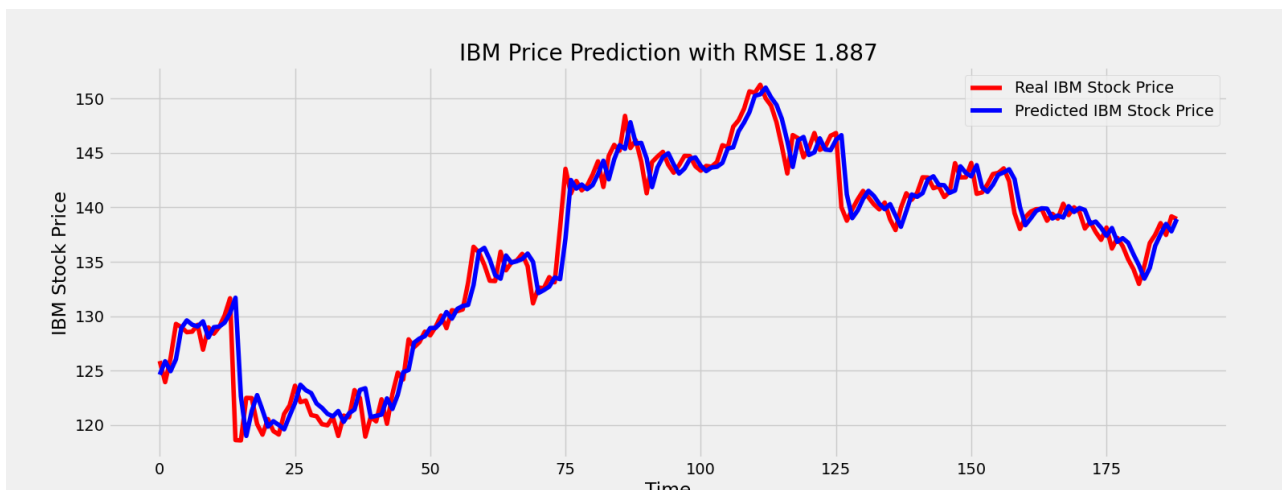
Για τα παραπάνω μοντέλα προσθέτουμε ένα επίπεδο με διαφορετικό αριθμό νευρώνων.

Τα αποτελέσματα του LSTM φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.23: Αποτελέσματα μοντέλου LSTM για την μετοχή της IBM

IBM LSTM - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	2.076	2.757	2.158
50, 100	1.940	3.184	2.666
50, 150	1.933	2.327	2.637
100, 100	1.934	2.259	2.757
100, 150	1.887	2.415	2.669

Από τα αποτελέσματα βλέπουμε ότι το μοντέλο με τον κανόνα ADAM έχει καλύτερες αποδόσεις σε σχέση με το RMSProp και SGD. Το βέλτιστο μοντέλο έχει 100 νευρώνες στο πρώτο επίπεδο και 150 στο δεύτερο επίπεδο, ενώ βλέπουμε ότι οι διαφορές για τον ADAM μεταξύ των διαφορετικών αριθμών νευρώνων είναι πολύ μικρές.



Γράφημα 3.24: Αποτελέσματα μοντέλου LSTM για την μετοχή της IBM με σφάλμα 1.887

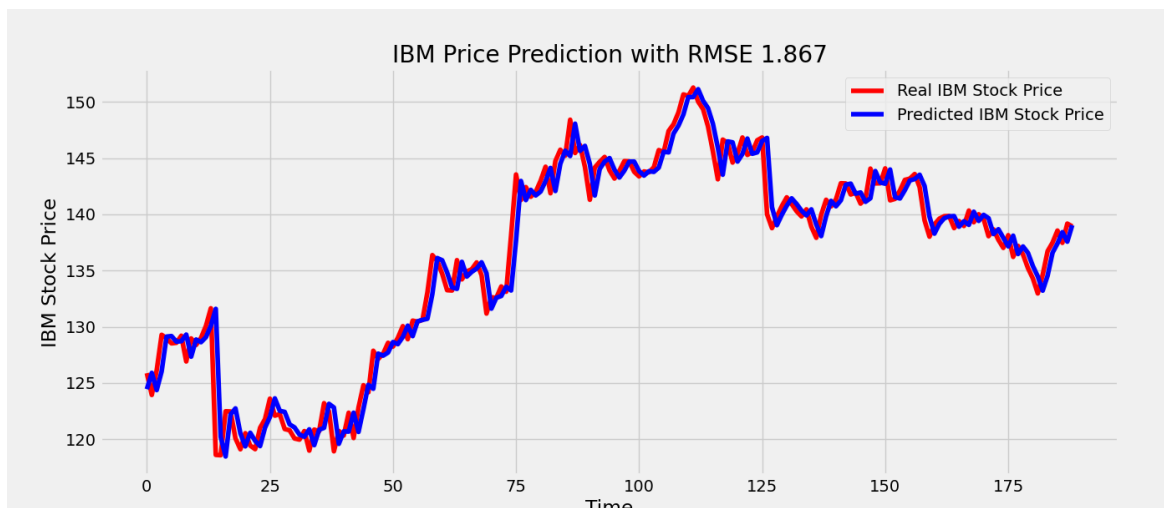
Στο διάγραμμα παρατηρούμε τις προβλέψεις του βέλτιστου μοντέλου SimpleRNN, έχοντας σχετικά μικρή απόκλιση στις αρχές του έτους για τις πρώτες ημέρες.

Τα αποτελέσματα του GRU φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.24: Αποτελέσματα μοντέλου GRU για την μετοχή της IBM

IBM GRU - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	1.878	2.632	2.411
50, 100	1.867	2.251	2.235
50, 150	2.355	1.993	2.272
100, 100	1.877	3.379	2.218
100, 150	1.934	3.110	2.239

Εξίσου το δίκτυο GRU έχει τις καλύτερες επιδόσεις συνολικά με τον ADAM για δύο επίπεδα, όπως συνέβη και με το LSTM. Το βέλτιστο μοντέλο είναι με 50 νευρώνες στο πρώτο επίπεδο και 100 στο δεύτερο επίπεδο.



Γράφημα 3.25: Αποτελέσματα μοντέλου GRU για την μετοχή της IBM με σφάλμα 1.867

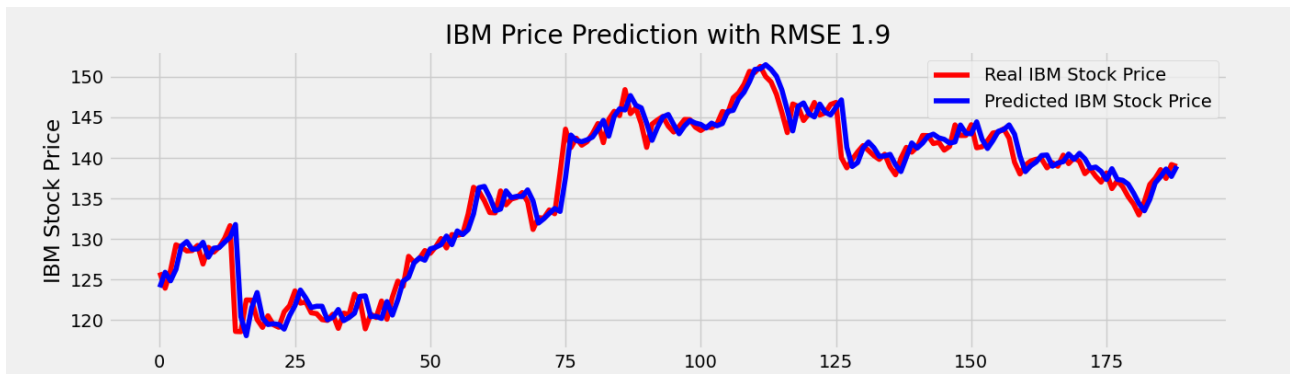
Στο διάγραμμα παρατηρούμε ότι τα αποτελέσματα που παράγει το βέλτιστο μοντέλο GRU με δύο επίπεδα εφάπτονται σχεδόν ακριβώς επάνω στις πραγματικές τιμές κλεισίματος των δεδομένων ελέγχου.

Τα αποτελέσματα του SimpleRNN φαίνονται στον παρακάτω πίνακα.

Πίνακας 3.25: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της IBM

IBM SimpleRNN - Two Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50	1.900	2.213	2.307
50, 100	1.916	2.313	2.625
50, 150	2.061	1.925	2.284
100, 100	1.933	2.272	2.361
100, 150	1.902	2.511	2.171

Το δίκτυο SimpleRNN επιτυγχάνει τις καλύτερες αποδόσεις με τον ADAM, όπως παρατηρήθηκε και στα προηγούμενα μοντέλα. Οι αποδόσεις είναι κοντά ανεξάρτητα από τον αριθμό νευρώνων στα επίπεδα του μοντέλου. Το βέλτιστο μοντέλο είναι με τον κανόνα ADAM και 50 νευρώνες στο επίπεδα.



Γράφημα 3.26: Αποτελέσματα μοντέλου *SimpleRNN* για την μετοχή της *IBM* με σφάλμα 1.900

Από το διάγραμμα διακρίνουμε ότι και αυτό το μοντέλο έχει πολύ καλή απόδοση εντοπίζοντας με μεγάλη ακρίβεια την πορεία και συμπεριφορά της χρονοσειράς.

3.5.3.3 Νευρωνικά Δίκτυα με τρία επίπεδα

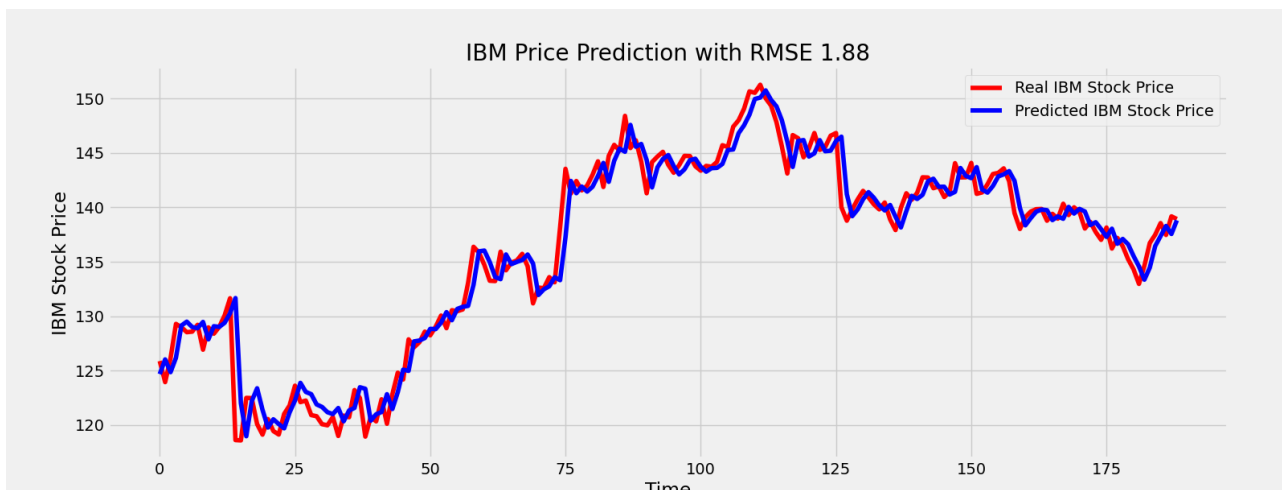
Συνεχίζοντας, δημιουργούμε περισσότερο πολύπλοκά τα ανωτέρω μοντέλα, προσθέτοντας ένα ακόμα επίπεδο, αυξάνοντας της πολυπλοκότητά τους.

Τα αποτελέσματα του LSTM με τρία επίπεδα παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.26: Αποτελέσματα μοντέλου *LSTM* για την μετοχή της *IBM*

IBM LSTM - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	2.641	1.986	3.075
50, 50, 150	2.424	1.880	3.101
50, 100, 100	2.189	2.035	3.077
50, 100, 150	1.965	2.105	3.194
100, 100, 100	1.925	2.301	3.036
100, 100, 150	1.913	2.546	3.124

Το μοντέλο που είχε τη βέλτιστη απόδοση είναι με βελτιστοποιητή RMSProp και 50 νευρώνες στο πρώτο και στο δεύτερο επίπεδο και 150 νευρώνες στο τρίτο επίπεδο. Παρότι έχουμε κατασκευάσει ένα δίκτυο με υψηλές διαστάσεις, επιτυγχάνει πολύ καλές προβλέψεις για όλες τις δοκιμές. Όσο αυξάνεται ο αριθμός των νευρώνων ανά επίπεδο τόσο καλύτερες αποδόσεις δείχνει να έχει το μοντέλο με τον ADAM, ενώ το αντίθετο συμβαίνει με τον RMSProp. Ο SGD δείχνει να έχει τις χαμηλότερες, αλλά σταθερές επιδόσεις για την μετοχή της IBM.



Γράφημα 3.27: Αποτελέσματα μοντέλου LSTM για την μετοχή της IBM με σφάλμα 1.880

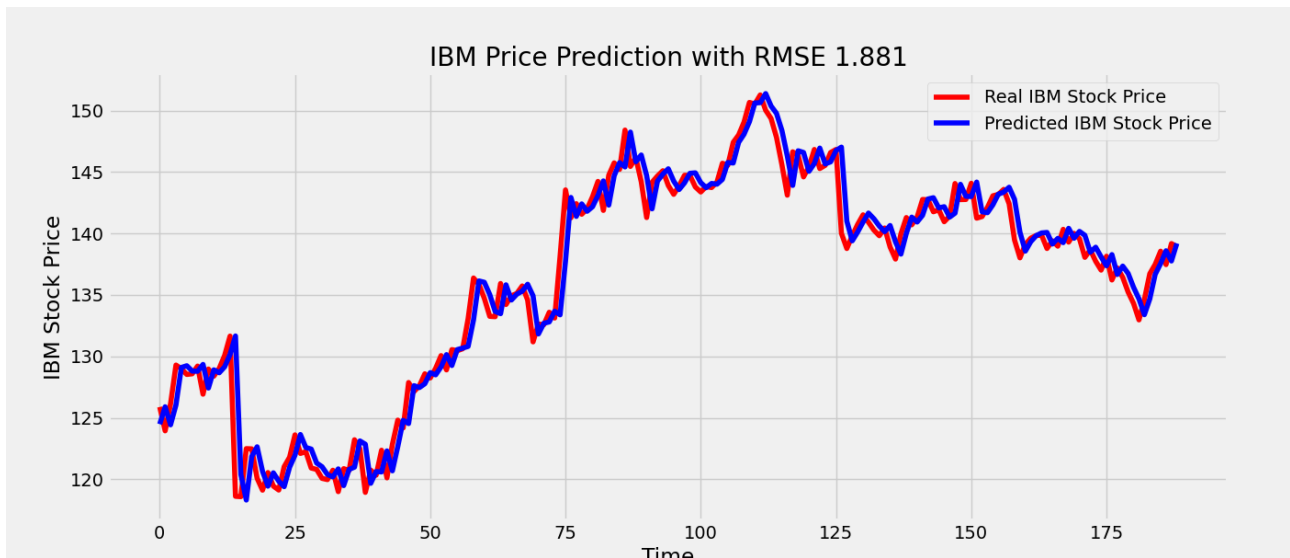
Από το διάγραμμα βλέπουμε και για αυτόν τον τύπο δικτύου, οι προβλέψεις του μοντέλου είναι πολύ ακριβής στην αναγνώριση της συμπεριφοράς της χρονοσειράς.

Τα αποτελέσματα του GRU με τρία επίπεδα παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.27: Αποτελέσματα μοντέλου GRU για την μετοχή της IBM

IBM GRU - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	1.889	2.920	2.951
50, 50, 150	1.980	3.397	2.531
50, 100, 100	1.881	2.052	2.558
50, 100, 150	1.873	1.883	2.685
100, 100, 100	2.228	4.118	2.455
100, 100, 150	1.943	2.096	2.479

Από τα αποτελέσματα που παρουσιάζονται στον ανωτέρω πίνακα, διακρίνουμε ότι ο ADAM έχει τις καλύτερες επιδόσεις συνολικά και το βέλτιστο μοντέλο προκύπτει με τον ADAM είναι με 50 νευρώνες στο πρώτο επίπεδο, 100 στο δεύτερο επίπεδο και 100 στο τρίτο επίπεδο.



Γράφημα 3.28: Αποτελέσματα μοντέλου GRU για την μετοχή της IBM με σφάλμα 1.881

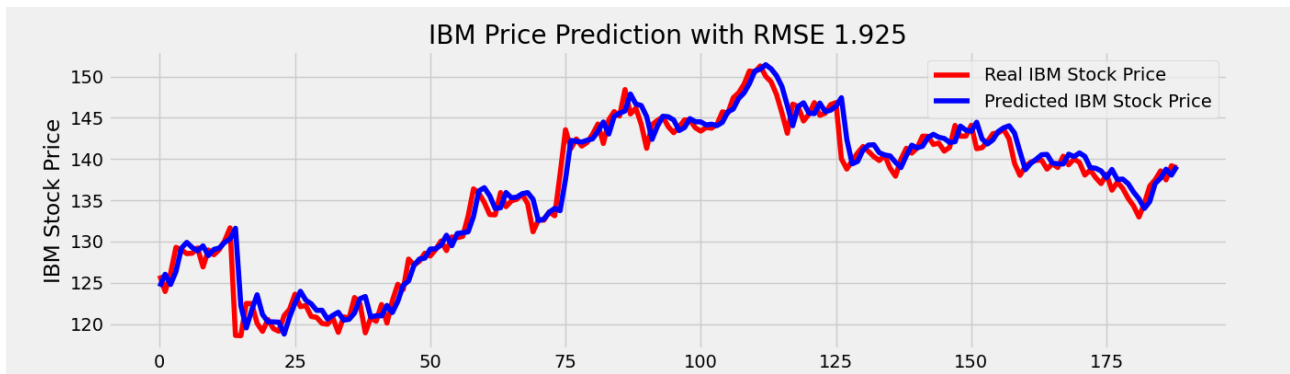
Από το διάγραμμα είναι εμφανές ότι το μοντέλο έχει εξαιρετική απόδοση για την πρόβλεψη της τιμής κλεισίματος με μικρό σφάλμα παρόλο που έχει αυξημένη πολυπλοκότητα.

Τα αποτελέσματα του SimpleRNN με τρία επίπεδα παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.28: Αποτελέσματα μοντέλου SimpleRNN για την μετοχή της IBM

IBM SimpleRNN - Three Layers - Root Mean Squared Error			
Units	ADAM	RMSPROP	SGD
50, 50, 100	2.865	2.415	2.288
50, 50, 150	2.327	2.351	2.377
50, 100, 100	2.099	2.079	2.385
50, 100, 150	1.925	3.118	10.758
100, 100, 100	2.045	3.718	3.436
100, 100, 150	2.799	2.614	8.905

Από τον πίνακα βλέπουμε ότι και το μοντέλο SimpleRNN έχει πολύ καλές επιδόσεις με τον ADAM, αλλά για με τον βελτιστοποιητή SGD σημειώνει πολύ κακές επιδόσεις σε μερικές περιπτώσεις.



Γράφημα 3.29: Αποτελέσματα μοντέλου *SimpleRNN* για την μετοχή της *IBM* με σφάλμα 1.925

Στο ανωτέρω διάγραμμα βλέπουμε ότι το μοντέλο *SimpleRNN* με τρία επίπεδα έχει πολύ καλά αποτελέσματα εντοπίζοντας στην πορεία της τιμής κλεισίματος της *IBM* για το έτος 2021.

3.6 Αξιολόγηση

Σε γενικές γραμμές, τα μοντέλα που κατασκευάστηκαν φαίνεται να προβλέπουν πολύ καλά την πορεία των τιμών κλεισίματος για τις μετοχές *Google*, *Tesla* και *IBM*. Παρατηρήθηκε ότι όσο περισσότερο προσθέτουμε επίπεδα στα μοντέλα, αυξάνοντας δηλαδή την πολυπλοκότητά τους, τόσο δυσκολότερο ήταν για τα μοντέλα να αναγνωρίσουν το μοτίβο της πορείας την χρονοσειράς. Εκτός από μερικές περιπτώσεις, σχεδόν όλα τα μοντέλα κατάφεραν να προβλέψουν την πορεία της εκάστοτε μετοχής ανεξάρτητα από τον αριθμό νευρώνων και τον αριθμό κρυφών νευρώνων που χρησιμοποιήθηκαν, επιτυγχάνοντας σε μεγάλο βαθμό τον σκοπό τους. Ειδικότερα, οι μετοχές διαφοροποιούνται μεταξύ τους τόσο ως προς το εύρος των τιμών κλεισίματος όσο ως προς το μοτίβο που ακολουθεί η κάθε χρονοσειρά, γεγονός που μας αποδεικνύει ότι τα αναδρομικά νευρωνικά δίκτυα μπορούν να αναγνωρίσουν τις διάφορες συμπεριφορές με μεγάλη επιτυχία.

Από τους βελτιστοποιητές είδαμε ότι τα μοντέλα είχαν καλύτερες προβλέψεις με τους *ADAM* και *RMSProp* και λιγότερο καλές αποδόσεις με τον *SGD* που σε μερικές περιπτώσεις αδυνατούσαν τα μοντέλα να αναγνωρίσουν την μελλοντική πορεία της μετοχής.

Για την μετοχή της *Google* το βέλτιστο μοντέλο είναι το *GRU*, το οποίο περιλαμβάνει ένα επίπεδο με 150 νευρώνες και βελτιστοποιητή τον *Adam* και σφάλμα ίσο με 28.918, ενώ το δεύτερο καλύτερο μοντέλο είναι το *LSTM* με ένα με 250 νευρώνες και *RMSE* ίσο με 29.663. Την χειρότερη απόδοση την είχε το μοντέλο *SimpleRNN* με τρία επίπεδα και με βελτιστοποιητή τον *SGD*, όπου το *RMSE* ισούται με 711.263. Για το *SimpleRNN* στα τρία επίπεδα με τον *SGD*, παρατηρήθηκε ότι όσο αυξανόταν ο αριθμός των νευρώνων σημείωνε όλο και χειρότερη απόδοση, αδυνατώντας να προβλέψει την πορεία της μετοχής *Google*. Επιπροσθέτως, είχε μεγάλα σφάλματα και με τον *RMSProp*, ενώ αρκετά καλύτερη απόδοση είχε με τον *ADAM*.

Για την μετοχή της Tesla παρατηρήθηκε ότι το καλύτερο μοντέλο ήταν εξίσου με ένα επίπεδο, όπως παρατηρήθηκε και με τη μετοχή της Google. Ειδικότερα, το μοντέλο GRU είχε την καλύτερη απόδοση 200 νευρώνες με τον RMSProp και σφάλμα 19.301 δολάρια, ενώ δεύτερο καλύτερο μοντέλο με ελάχιστη διαφορά στα 19.632 δολάρια είναι το LSTM με 250 νευρώνες στο επίπεδο. Την χειρότερη απόδοση την είχε ξανά το μοντέλο SimpleRNN με τρία επίπεδα με RMSE ίσο με 187.141. Το συγκεκριμένο μοντέλο είχε 50, 100 και 150 νευρώνες στα επίπεδα και με βελτιστοποιητή τον SGD.

Για την μετοχής της IBM παρατηρήθηκε ότι τα μοντέλα με βελτιστοποιητή τον ADAM είχαν εξαιρετικές επιδόσεις συνολικά. Γενικότερα, τα καλύτερα μοντέλα που κατασκευάστηκαν για ίδιο αριθμό επιπέδων και νευρώνων είχαν παρόμοια απόδοση με ελάχιστη διαφορά στα σφάλματα. Το καλύτερο μοντέλο είναι το GRU με δύο επίπεδα, με 50 νευρώνες στο πρώτο επίπεδο και 100 στο δεύτερο και με RMSE 1.867. Το δεύτερο καλύτερο μοντέλο είναι ξανά το GRU με RMSE ίσο με 1.882, κανόνα βελτιστοποίησης τον ADAM και 150 νευρώνες στο επίπεδο. Το χειρότερο μοντέλο είναι το SimpleRNN με τρία επίπεδα με σφάλμα 10.758 δολάρια. Το συγκεκριμένο μοντέλο αποτελείται από 50 νευρώνες στο πρώτο επίπεδο, 100 νευρώνες στο δεύτερο επίπεδο και 150 στο τρίτο επίπεδο και βελτιστοποιητή τον SGD.

4 ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ ΠΡΟΒΛΕΨΗΣ ΜΕΤΟΧΩΝ

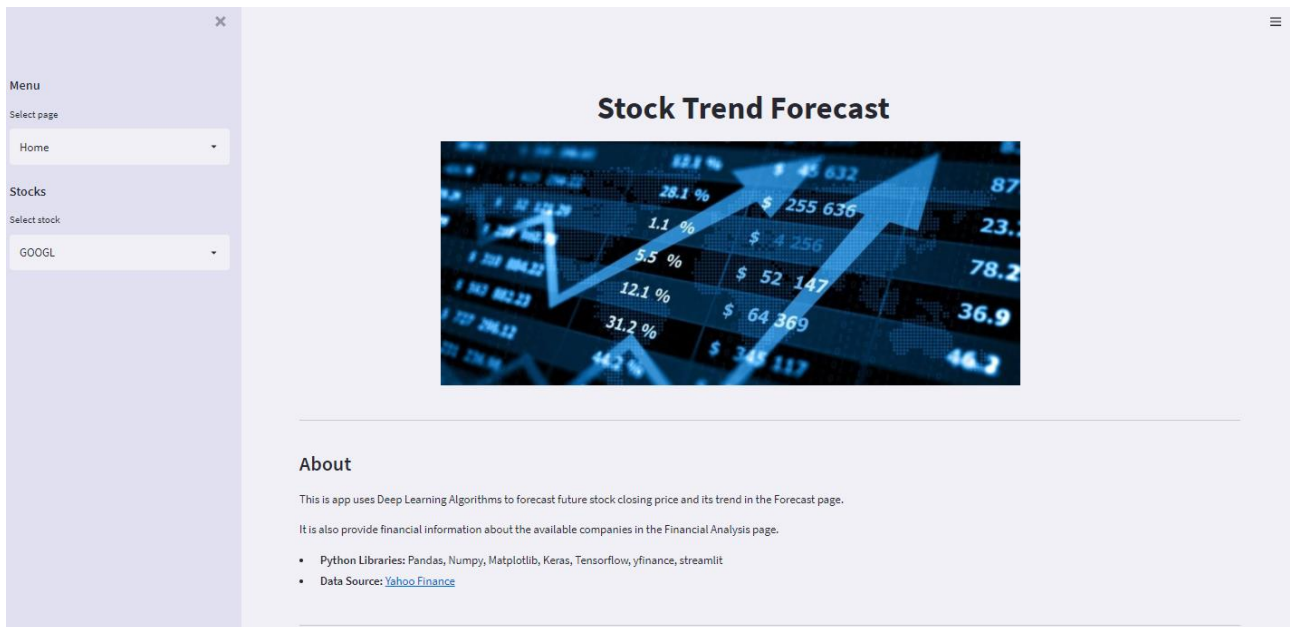
Στη συγκεκριμένη ενότητα αναπτύσσουμε μια διαδικτυακή εφαρμογή πρόβλεψης των μελλοντικών τιμών του χρηματιστηρίου που κάνει χρήση των βέλτιστων μοντέλων που αναπτύχθηκαν στην προηγούμενη ενότητα. Ο χρήστης θα έχει τη δυνατότητα να επιλέξει ανάμεσα στις διαφορετικές αρχιτεκτονικές των μοντέλων που αναπτύχθηκαν, τον αριθμό των κρυφών επιπέδων που θα διαθέτει το δίκτυο, καθώς και τις ημέρες που επιθυμεί προς πρόβλεψη .

Η εφαρμογή αναπτύχθηκε σε γλώσσα python με τη χρήση του ανοιχτού πακέτου streamlit, όπου χρησιμοποιείται για γρήγορη ανάπτυξη εφαρμογών ιστού μηχανικής μάθησης και επιστήμης των δεδομένων, χωρίς τη χρήση HTML και CSS από την πλευρά του προγραμματιστή. Επίσης, με απλές εντολές python μπορούμε να αναπτύξουμε γρήγορα και με εύκολο τρόπο μια διαδικτυακή εφαρμογή που θα κάνει χρήση τα νευρωνικά δίκτυα που αναπτύχθηκαν.

Για να χρησιμοποιήσουμε τα μοντέλα που κατασκευάσαμε κατά την εκπαίδευση, χρησιμοποιήσαμε τη βιβλιοθήκη pickle, ώστε να τα αποθηκεύσουμε σε δυαδική μορφή. Επιπροσθέτως, εκτός των μοντέλων, αποθηκεύσαμε και το αντικείμενο της κλάσης MinMaxScaler που χρησιμοποιήσαμε για τη μείωση του εύρους των τιμών, ώστε να το επαναχρησιμοποιήσουμε στις τιμές της εκάστοτε μετοχής που επιλέγει ο χρήστης.

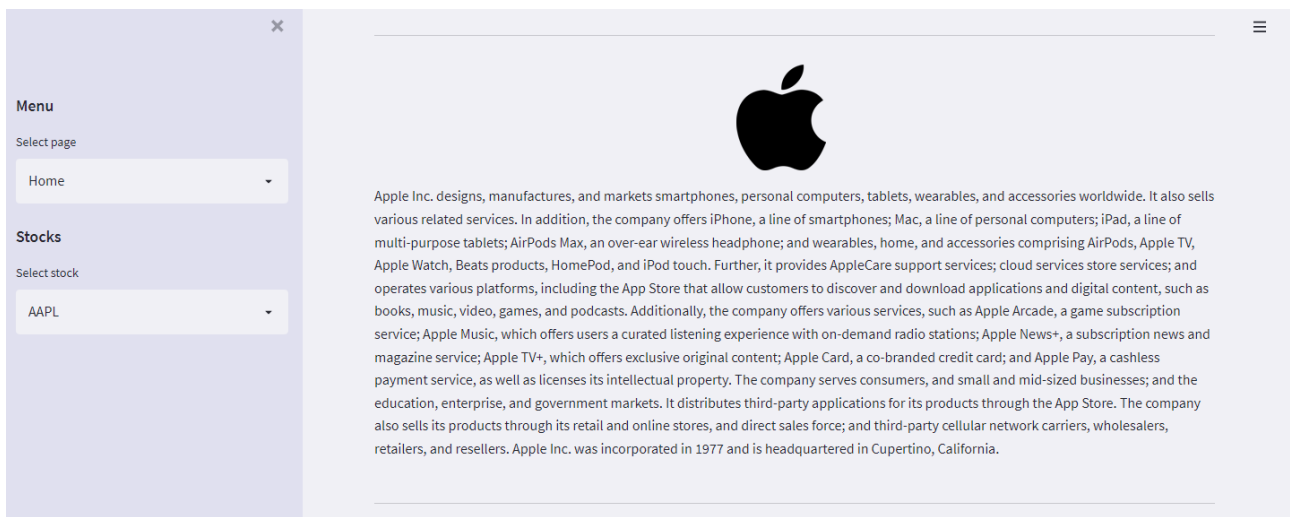
4.1 Γραφική απεικόνιση της διαδικτυακής εφαρμογής

Με την εκκίνηση της εφαρμογής εμφανίζεται η αρχική σελίδα Home όπου περιέχει τον τίτλο της εφαρμογής, μια περιγραφή σχετικά με τον σκοπό που εξυπηρετεί, τις βασικές βιβλιοθήκες της ρυθμον που χρησιμοποιούμε για την υλοποίησή της, καθώς και την πηγή άντλησης δεδομένων, μέσα από τη βιβλιοθήκη yfinance.



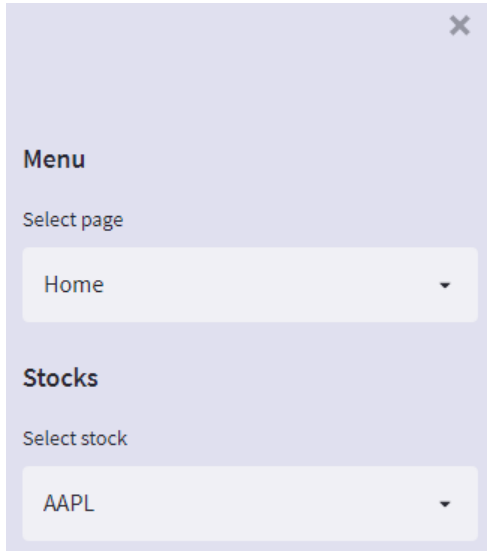
The screenshot shows the application's home page. On the left, there is a sidebar menu with a 'Menu' section containing a 'Select page' dropdown set to 'Home' and a 'Stocks' section with a 'Select stock' dropdown set to 'GOOGL'. The main content area features a large image titled 'Stock Trend Forecast' showing a blue line graph with upward-pointing arrows over a background of stock market data. Below the image, there is an 'About' section with the following text: 'This app uses Deep Learning Algorithms to forecast future stock closing price and its trend in the Forecast page. It is also provide financial information about the available companies in the Financial Analysis page.' Below this, there is a list of Python Libraries: Pandas, Numpy, Matplotlib, Keras, Tensorflow, yfinance, streamlit, and a Data Source: Yahoo Finance.

Ακριβώς κάτω από τις πληροφορίες που αφορούν την εφαρμογή, εμφανίζονται οι πληροφορίες της εκάστοτε εταιρείας που έχει επιλέξει ο χρήστης με λογότυπό της, όπως φαίνεται στην εικόνα. Ως προεπιλεγμένη μετοχή έχει ορισθεί τυχαία της Apple (AAPL) και το κείμενο δίδεται από το attribute info της κλάσης Ticker του πακέτου yfinance.

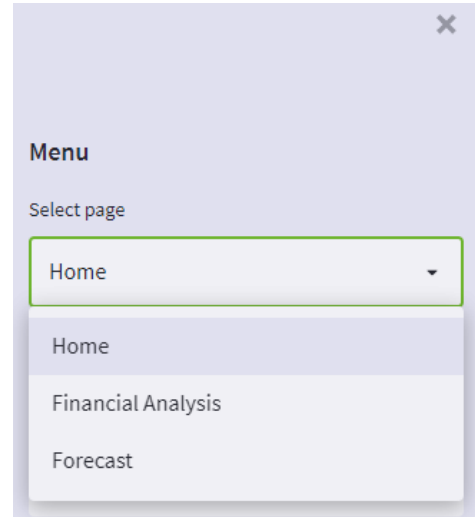


The screenshot shows the application's home page with the stock selected as 'AAPL'. The main content area features a large black Apple logo. Below the logo, there is a detailed paragraph of text describing Apple Inc. and its various products and services, including iPhones, iPads, Macs, AirPods, and various subscription services like Apple Music and Apple TV+.

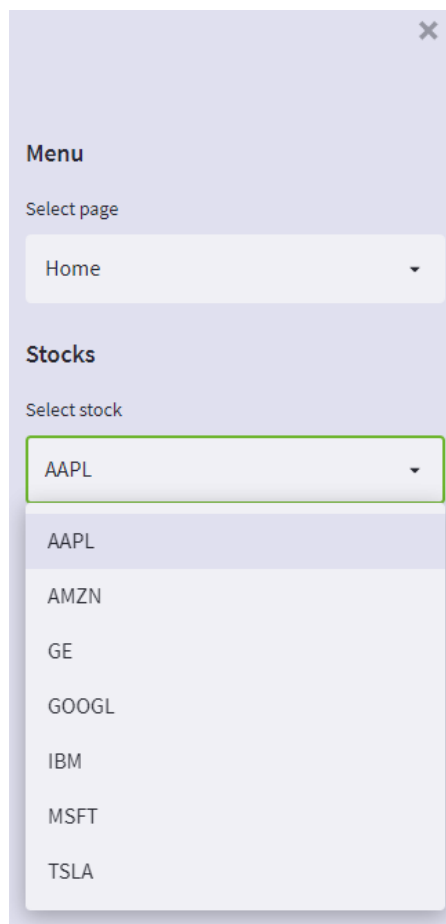
Οι σελίδες και η επιλογή των διαθέσιμων μετοχών είναι προσβάσιμες από το πλευρικό μενού της εφαρμογής, όπως εμφανίζεται στις παρακάτω εικόνες.



Εικόνα 4.3: Βασικό πλευρικό μενού

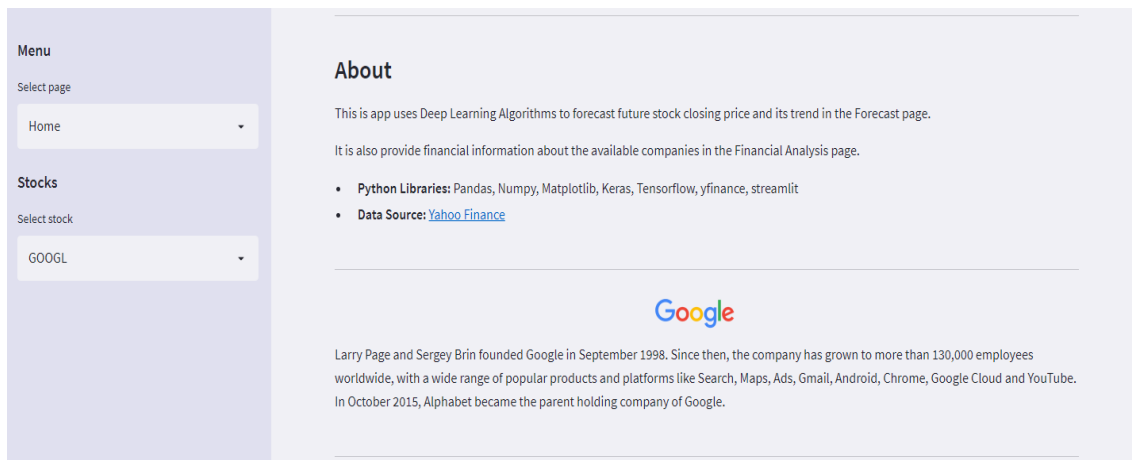


Εικόνα 4.4: Επιλογή σελίδας από το πλευρικό μενού



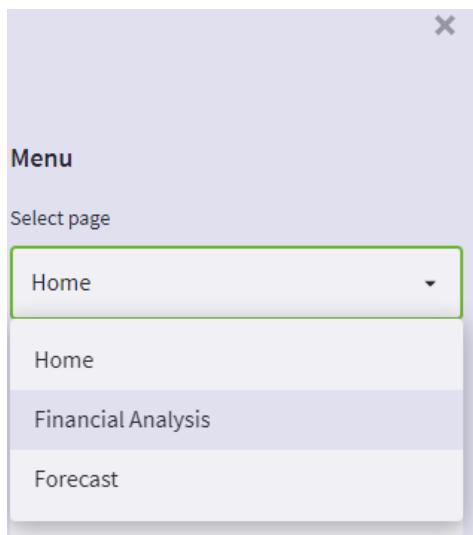
Εικόνα 4.5: Επιλογή μετοχής από το πλευρικό μενού

Με την επιλογή μιας μετοχής που είναι διαθέσιμες ανανεώνεται το κείμενο και το λογότυπο της αρχικής σελίδας. Ειδικότερα αν επιλέξει ο χρήστης τη μετοχής της Google (GOOGL), τότε η σελίδα ανανεώνεται όπως φαίνεται στην παρακάτω εικόνα 4.6.

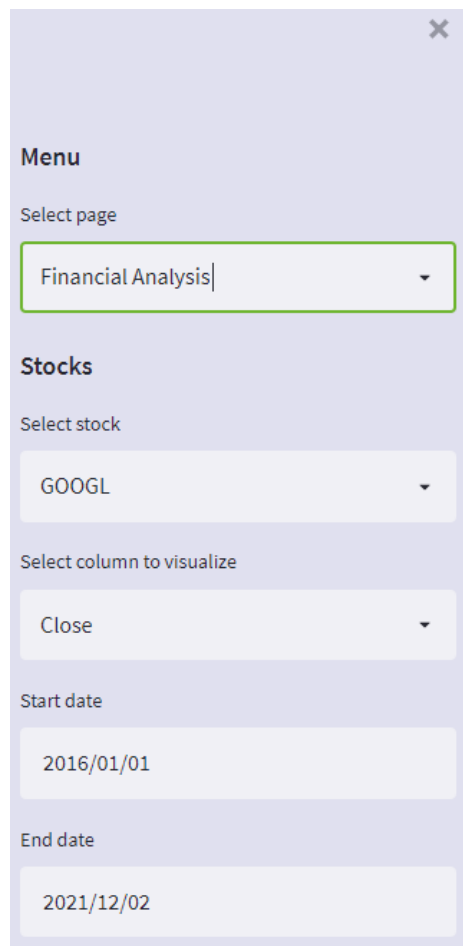


Εικόνα 4.6: Επιλογή μετοχής της Google από το πλευρικό μενού

Συνεχίζοντας την περιήγηση στην εφαρμογή, κατευθυνόμαστε στην αμέσως επόμενη σελίδα Financial Analysis από το πλευρικό μενού, όπως φαίνεται στην εικόνα 4.7 που αφορά στα χρηματοοικονομικές πληροφορίες της εταιρείας στην οποία ανήκει η μετοχή, καθώς επίσης και τα πραγματικά δεδομένα των τιμών της μετοχής, όπως δίδονται από την βιβλιοθήκη yfinance.



Εικόνα 4.7: Μετακίνηση στη σελίδα Financial Analysis




Εικόνα 4.8: Βασικό μενού σελίδας Financial Analysis


Κατά την κατεύθυνσή μας στη σελίδα Financial Analysis, το πλευρικό μενού αλλάζει όπως φαίνεται στην εικόνα 4.8. Από εδώ μπορεί ο χρήστης έχει περισσότερες επιλογές όπως την μετοχή, την στήλη, καθώς και τις ημερομηνίες που τον ενδιαφέρουν προς απεικόνιση. Τα δεδομένα εμφανίζονται με την συνάρτηση `download()` που διαθέτει η `yfinance`, με παραμέτρους την προεπιλεγμένη μετοχή, την ημερομηνία έναρξης ίδια με αυτή που αναπτύχθηκαν τα μοντέλα, δηλαδή την 01-01-2016 έως το σήμερα που ορίζεται κάθε φορά όταν τρέχει η εφαρμογή με τη συνάρτηση `today()` του πακέτο `datetime`. Στην εικόνα 4.9 που ακολουθεί παρουσιάζονται τα δεδομένα της μετοχής Google που έχουμε επιλέξει μεταξύ το χρονικό όριο που έχουμε ορίσει.

GOOGL Data from Yahoo Finance

Data Dimension: 1491 rows and 6 columns

	Open	High	Low	Close	Adj Close	Volume
2015-12-31	787.8200	788.3300	777.3200	778.0100	778.0100	1637600
2016-01-04	762.2000	762.2000	747.5400	759.4400	759.4400	3369100
2016-01-05	764.1000	769.2000	755.6500	761.5300	761.5300	2260800
2016-01-06	750.3700	765.7300	748.0000	759.3300	759.3300	2410300
2016-01-07	746.4900	755.3100	735.2800	741.0000	741.0000	3156600
2016-01-08	747.8000	750.1200	728.9200	730.9100	730.9100	2375300
2016-01-11	731.9500	735.0800	719.5600	733.0700	733.0700	2544800
2016-01-12	740.7500	748.3400	736.4300	745.3400	745.3400	2340800
2016-01-13	749.3400	753.0000	716.7800	719.5700	719.5700	2586400
2016-01-14	724.4400	739.8900	705.0000	731.3900	731.3900	2777900

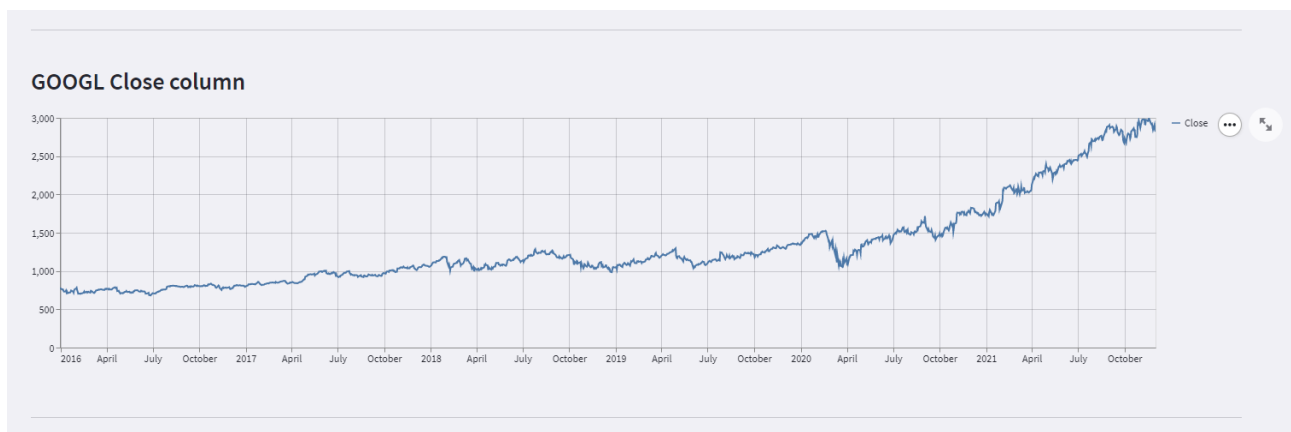
 Download Excel

 Download CSV

Εικόνα 4.9: Μετακίνηση στη σελίδα Financial Analysis

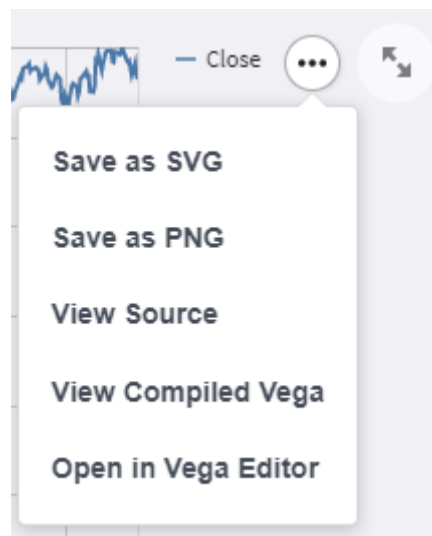
Στην παραπάνω εικόνα εμφανίζονται τα δεδομένα με τις τιμές της εκάστοτε μετοχής που έχει επιλεγεί, καθώς επίσης ο συνολικός αριθμός των στηλών και σειρών. Επιπλέον, δίδεται η δυνατότητα στον χρήστη να αποθηκεύσει τα δεδομένα σε μορφή xls ή csv, εφόσον το επιθυμεί.

Στη συνέχεια, κάτω από τα δεδομένα εμφανίζεται ένα διάγραμμα, όπως διακρίνουμε στην εικόνα 4.10, όπου απεικονίζονται τα δεδομένα για τις ημερομηνίες που έχει επιλέξει ο χρήστης, αλλά και για την στήλη που επιθυμεί. Ως προεπιλογή έχουμε ορίσει την τιμή κλεισίματος που είναι η στήλη, όπου πραγματοποιούνται οι προβλέψεις.



Εικόνα 4.10: Μετακίνηση στη σελίδα Financial Analysis

Μια καλή λειτουργία του συγκεκριμένου διαγράμματος είναι ότι μπορεί ο χρήστης να μετακινήσει τον κέρσορα επάνω στην γραμμή που αντικατοπτρίζει την πορεία της μετοχής και να εμφανίσει την ημερομηνία, την τιμή και την στήλη, όπως φαίνεται στην παραπάνω εικόνα. Επιπλέον, δίδεται η δυνατότητα μεγέθυνσης, ώστε είναι πιο ευδιάκριτη η πορεία της τιμής για μικρότερα χρονικά διαστήματα. Ακόμη, εάν επιλέξουμε το εικονίδιο με τις τρεις τελείες εμφανίζεται μια σειρά από επιλογές όπως φαίνεται στην εικόνα 4.11, ενώ το εικονίδιο με τα βέλη μεγαλώνει το διάγραμμα σε νέο παράθυρο `full_screen`.



Εικόνα 4.11: Επιλογές που δίδονται από τη συνάρτηση `line_chart`

Εν συνεχεία, στην σελίδα `Financial Analysis` εμφανίζονται χρηματοοικονομικές πληροφορίες που αφορούν την εταιρεία, όπως διάφορα νέα που δημοσιεύονται στην ιστοσελίδα της `yahoo finance`. Στην εικόνα 4.12 παρουσιάζονται τα νέα που αφορούν την εταιρεία `Google` από την οποία έχουμε ήδη επιλέξει.

News

	title	publisher	link	providerPublishTime
0	Ex-Google scientist Gebru opens AI institute year after tumultuous exit	Reuters	https://finance.yahoo.com/news/ex-google-scientist-gebru-opens-130446340.html	1638450286
1	Dow Jones Futures Pare Gains, Tech Futures Fall After Market's Bearish Reversal; Apple Retreats, Snowflake Jumps	Investor's Business Daily	https://finance.yahoo.com/m/627dc627-6ac2-3707-a40c-84b9a32492e5/dow-jones-futures-pare-gains%2C.html	1638450285
2	10 Best Stocks to Buy Today According to Wallace Weitz's Hedge Fund	Insider Monkey	https://finance.yahoo.com/news/10-best-stocks-buy-today-103328072.html	1638441208
3	Amazon (AMZN) Gains Competitive Edge with Graviton3 Instances	Zacks	https://finance.yahoo.com/news/amazon-amzn-gains-competitive-edge-021002583.html	1638411002
4	Facebook Users Rewrite Marcos History in Race to Succeed Duterte	Bloomberg	https://finance.yahoo.com/news/facebook-users-rewrite-marcos-history-00009168.html	1638403209
5	BMO's investment chief just predicted a 17% stock market gain for 2022 — ignore the bears and make room for these 3 'outperform' stocks	MoneyWise	https://finance.yahoo.com/news/bmos-investment-chief-just-predicted-223000602.html	1638397800
6	Square will change its name to Block as business expands	MarketWatch	https://finance.yahoo.com/m/9f2ca1a7-b01e-3901-9591-cd729ad770b8/square-will-change-its-name.html	1638396060
7	Facebook's Meta Ticker Trades in Canada, Ignores U.S. Delay	Bloomberg	https://finance.yahoo.com/news/facebook-meta-ticker-takes-over-154826041.html	1638394379

Εικόνα 4.12: Μετακίνηση στη σελίδα Financial Analysis

Η πρόσβαση στα συγκεκριμένα δεδομένα πραγματοποιείται με το attribute της κλάσης Ticker, όπου είναι τύπου dictionary και το μετατρέψαμε σε pandas DataFrame, ορίζοντας συγκεκριμένες στήλες προς εμφάνιση, όπως τον τίτλο, τον εκδότη, την ιστοσελίδα δημοσίευσης και την ώρα δημοσίευσης.

Κατόπιν, ακολουθούν επιπλέον πληροφορίες όπως φαίνονται από την εικόνα 4.13 έως την εικόνα 4.16.

Dividends and Splits			Major Holders	
	Dividends	Stock Splits	0	1
2014-04-03T00:00:00	0.0000	1.9980	0	0.15% % of Shares Held by All Insider
			1	79.60% % of Shares Held by Institutions
			2	79.72% % of Float Held by Institutions
			3	4487 Number of Institutions Holding Shares

Εικόνα 4.13: Μερίσματα της Google και ποσοστά των κυριότερων μετόχων.

Institutional Holders

	Holder	Shares	Date Reported	% Out	Value
0	Vanguard Group, Inc. (The)	22961780	2021-09-29T00:00:00	0.0763	61388778065
1	Blackrock Inc.	20380976	2021-09-29T00:00:00	0.0678	54488946955
2	FMR, LLC	13443290	2021-09-29T00:00:00	0.0447	35940904680
3	State Street Corporation	11095512	2021-09-29T00:00:00	0.0369	29664073242
4	Price (T.Rowe) Associates Inc	7328683	2021-09-29T00:00:00	0.0244	19593380574
5	Norges Bank Investment Management	5433638	2020-12-30T00:00:00	0.0181	9523211304
6	Geode Capital Management, LLC	5206549	2021-09-29T00:00:00	0.0173	13919812882
7	Wellington Management Group, LLP	4669223	2021-09-29T00:00:00	0.0155	12483261074
8	Capital Research Global Investors	3769465	2021-09-29T00:00:00	0.0125	10077740066
9	Northern Trust Corporation	3683032	2021-09-29T00:00:00	0.0122	9846659712

Εικόνα 4.14: Ιδρυματικοί Μέτοχοι

Recommendations

	Firm	To Grade	From Grade	Action
2014-12-17T00:00:00	Pivotal Research	Buy		up
2015-01-08T06:03:07	Stifel	Hold	Buy	down
2015-01-14T09:00:00	Credit Suisse	Outperform		main
2015-01-30T09:00:00	Citigroup	Buy		main
2015-03-02T14:04:06	Bank of America	Buy	Neutral	up
2015-03-06T08:07:11	Citigroup	Buy		main
2015-04-21T09:00:00	SunTrust Robinson Humphrey	Buy		main
2015-04-24T09:00:00	CRT Capital	Buy		main
2015-07-13T08:40:26	Pivotal Research	Hold	Buy	down
2015-07-16T06:18:29	BMO Capital	Outperform	Market Perform	up

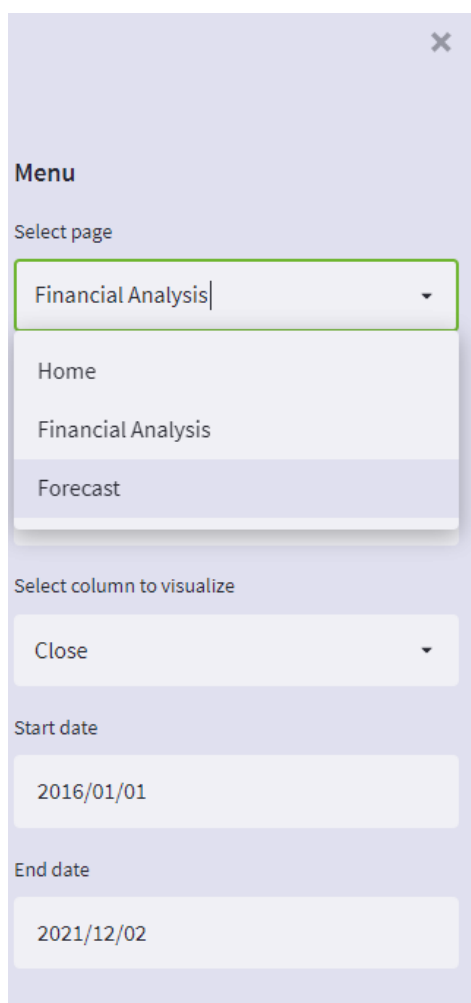
Εικόνα 4.15: Συστάσεις εταιρειών που αφορούν την μετοχή GOOGL

Balance Sheet

	2020-12-31 00:00:00	2019-12-31 00:00:00	2018-12-31 00:00:00	2017-12-31 00:00:00
Intangible Assets	1,445,000,000.0000	1,979,000,000.0000	2,220,000,000.0000	2,692,000,000.0000
Total Liab	97,072,000,000.0000	74,467,000,000.0000	55,164,000,000.0000	44,793,000,000.0000
Total Stockholder Equity	222,544,000,000.0000	201,442,000,000.0000	177,628,000,000.0000	152,502,000,000.0000
Other Current Liab	27,252,000,000.0000	22,159,000,000.0000	17,612,000,000.0000	10,651,000,000.0000
Total Assets	319,616,000,000.0000	275,909,000,000.0000	232,792,000,000.0000	197,295,000,000.0000
Common Stock	58,510,000,000.0000	50,552,000,000.0000	45,049,000,000.0000	40,247,000,000.0000
Other Current Assets	5,490,000,000.0000	4,412,000,000.0000	4,236,000,000.0000	2,983,000,000.0000
Retained Earnings	163,401,000,000.0000	152,122,000,000.0000	134,885,000,000.0000	113,247,000,000.0000
Other Liab	15,160,000,000.0000	14,478,000,000.0000	16,532,000,000.0000	16,641,000,000.0000
Good Will	21,175,000,000.0000	20,624,000,000.0000	17,888,000,000.0000	16,747,000,000.0000

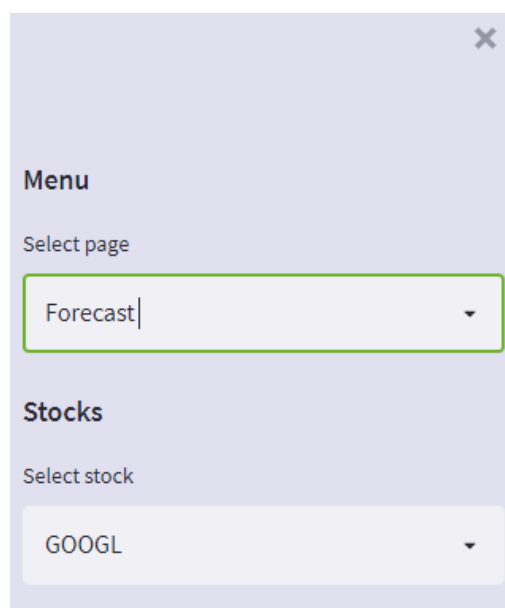
Εικόνα 4.16: Οικονομικός Ισολογισμός της Google

Για να κατευθυνθούμε στην τελευταία σελίδα της εφαρμογής, επιλέγουμε από το μενού την επιλογή Forecast. Με την ανακατεύθυνση στη σελίδα, το βασικό μενού αλλάζει, όπως φαίνεται στην εικόνα 4.18, το οποίο είναι παρόμοιο με το βασικό μενού της σελίδας Home.



A screenshot of a mobile application menu. The menu is titled "Menu" and has a close button (X) in the top right corner. It contains several sections: "Select page" with a dropdown menu showing "Financial Analysis" selected and a list of options: "Home", "Financial Analysis", and "Forecast" (highlighted); "Select column to visualize" with a dropdown menu showing "Close" selected; "Start date" with a text input field containing "2016/01/01"; and "End date" with a text input field containing "2021/12/02".

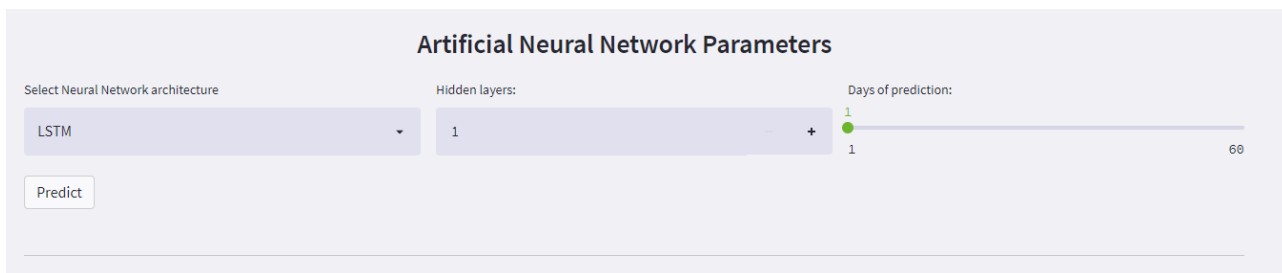
Εικόνα 4.17: Μετακίνηση στη σελίδα Forecast



A screenshot of a mobile application menu. The menu is titled "Menu" and has a close button (X) in the top right corner. It contains several sections: "Select page" with a dropdown menu showing "Forecast" selected; "Stocks" with a section titled "Select stock" and a dropdown menu showing "GOOGL" selected.

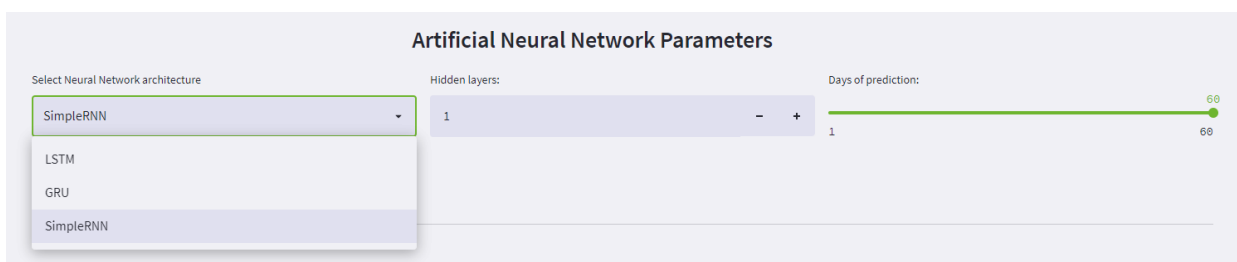
Εικόνα 4.18: Βασικό μενού της σελίδα Forecast

Στη σελίδα Forecast, μπορούμε να κάνουμε χρήση όλων των μοντέλων που κατασκευάστηκαν στην τρίτη ενότητα για να πραγματοποιήσουμε μελλοντικές προβλέψεις στην τιμή κλεισίματος της εκάστοτε μετοχής που έχουμε επιλέξει από το βασικό μενού. Όπως φαίνεται στην εικόνα 4.20, εισάγουμε τις κατάλληλες παραμέτρους για να επιλέξουμε το μοντέλο που μας ενδιαφέρει, καθώς επίσης και τον αριθμό των μελλοντικών ημερών που θέλουμε να προβλέψουμε.



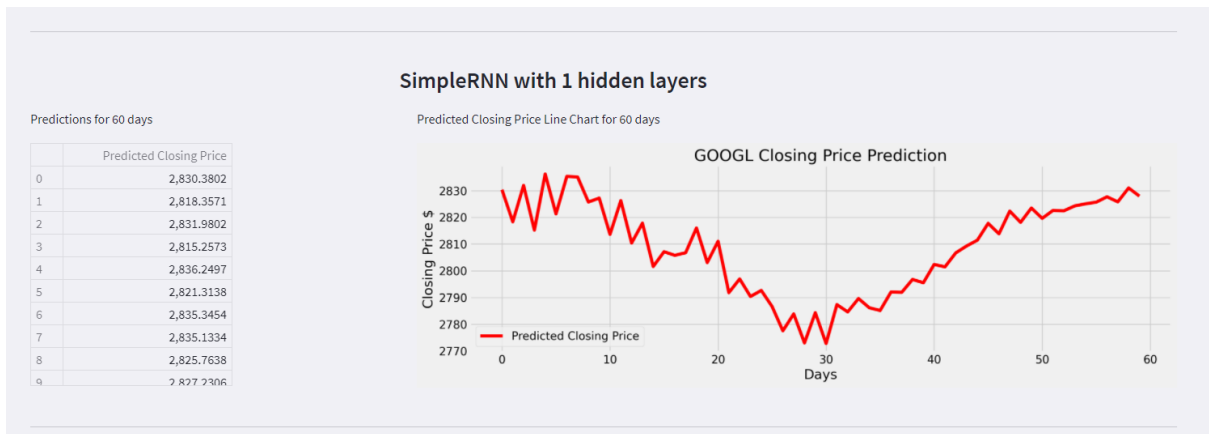
Εικόνα 4.19: Μετακίνηση στη σελίδα Forecast

Στην εικόνα 4.20, επιλέγουμε το μοντέλο SimpleRNN με ένα επίπεδο, ώστε να προβλέψουμε την τιμή της τιμής κλεισίματος της Google για τις επόμενες 60 ημέρες. Ο ελάχιστος και ο μέγιστος αριθμός των ημερών είναι καθορισμένος από το σύστημα, όπου αντιστοιχούν από 1 έως 60 ημέρες αντίστοιχα.



Εικόνα 4.20: Μετακίνηση στη σελίδα Forecast

Έχοντας επιλέξει το μοντέλο και τις ημέρες πρόβλεψης, πατάμε το κουμπί Predict, ώστε το εκάστοτε μοντέλο να εκτελεστεί και να εμφανίσει τα αποτελέσματα, όπως φαίνεται στην εικόνα 4.21.



Εικόνα 4.21: Μετακίνηση στη σελίδα Forecast

Στην εικόνα 4.21, παρουσιάζονται τα αποτελέσματα που παρήγαγε το μοντέλο για τη μετοχή της Google, όπου οι ακριβείς τιμές εμφανίζονται στον αριστερό πίνακα και στο διάγραμμα μπορούμε να διακρίνουμε την πορεία της μετοχής για το μέλλον.

5 ΕΠΙΛΟΓΟΣ

5.1 Συμπεράσματα

Η παρούσα διπλωματική εργασία είχε ως στόχο την ανάπτυξη διαδικτυακής εφαρμογής για την πρόβλεψη των μελλοντικών τιμών του Χρηματιστηρίου με τη χρήση μοντέλων βαθιάς μάθησης. Τα μοντέλα που αναπτύχθηκαν και αξιολογήθηκαν προκειμένου να ενσωματωθούν στην εφαρμογή, αφορούν κυρίως στα αναδρομικά νευρωνικά δίκτυα. Στη φάση της ανάπτυξης των μοντέλων, ακολούθησε μια σειρά από πειράματα για την εύρεση των καταλληλότερων παραμέτρων, με στόχο την κατασκευή ικανών μοντέλων πρόβλεψης διαφόρων αρχιτεκτονικών. Κατόπιν, ακολούθησε η φάση της ανάπτυξης της εφαρμογής με στόχο να είναι φιλική προς τον χρήστη και να πραγματοποιεί πρόβλεψη στις τιμές των μετοχών που ο ίδιος επιλέγει. Για την επίτευξη αυτού, η διαδικτυακή εφαρμογή χρησιμοποιεί τα ανωτέρω μοντέλα, όπου ο χρήστης μπορεί να προβλέψει την πορεία της τιμής κλεισίματος για την εκάστοτε μετοχή που έχει επιλέξει. Όπως αναφέρθηκε, ο χρήστης μπορεί να επιλέξει συγκεκριμένο εύρος ημερών προς πρόβλεψη, καθώς και συγκεκριμένες μετοχές που είναι διαθέσιμες από την εφαρμογή. Επιπλέον, για κάθε διαθέσιμη μετοχή παρουσιάζονται διάφορες χρηματοοικονομικές πληροφορίες που αφορούν την ίδια τη μετοχή και την εταιρεία έκδοσης, με στόχο την ενημέρωση του χρήστη.

5.2 Μελλοντικές επεκτάσεις

Όπως κάθε πληροφοριακό σύστημα έχει τη δυνατότητα επέκτασης και βελτίωσης, έτσι και η διαδικτυακή εφαρμογή που αναπτύχθηκε μπορεί εξίσου να αναβαθμιστεί. Οι μελλοντικές επεκτάσεις αφορούν τόσο το γραφικό περιβάλλον της εφαρμογής όσο και τα μοντέλα βαθιάς μάθησης που χρησιμοποιεί.

Σχετικά με τα μοντέλα πρόβλεψης, θα μπορούσαμε να πειραματιστούμε με διαφορετικούς ρυθμούς μάθησης των μοντέλων, καθώς η προκαθορισμένη τιμή είναι 0.001 που δίδεται από την κλάση `Optimizer` της `Keras`. Ένας διαφορετικός αριθμός μάθησης, ενδεχομένως να έδινε καλύτερα αποτελέσματα στα μοντέλα που αναπτύχθηκαν διατηρώντας τις υπόλοιπες παραμέτρους. Κατόπιν, τα μοντέλα εξετάστηκαν σε διαφορετικούς κύκλους εκπαίδευσης και σε διαφορετικό αριθμό `timesteps`, χωρίς να εμφανίζουν ιδιαίτερη βελτίωση. Επομένως, θα μπορούσαμε να επεκταθούμε και σε άλλες αρχιτεκτονικές, όπως `CNN` ή `MLP` μελετώντας την απόδοσή τους και εντάσσοντάς τα στην εφαρμογή. Τέλος, θα μπορούσαμε να χρησιμοποιήσουμε την κλάση `ModelCheckpoint` του πακέτου `callbacks` από την `Keras`, όπου κατά την εκπαίδευση αποθηκεύονται τα βάρη του βέλτιστου μοντέλου για συγκεκριμένο κύκλο εκπαίδευσης. Ειδικότερα, όταν εκπαιδεύουμε ένα μοντέλο με 100 κύκλους εκπαίδευσης (`epochs`), τα τελικά βάρη του μοντέλου είναι του τελευταίου κύκλου, ενώ του βέλτιστου μοντέλου μπορεί να είναι υποθετικά για 98 κύκλους εκπαίδευσης. Θέτοντας την παράμετρο `save_best_only` ίση με `True` της κλάσης `ModelCheckpoint` στην παράμετρο `callbacks` της μεθόδου

fit(), θα αποθηκεύαμε το βέλτιστο μοντέλο για κάθε νευρωνικό δίκτυο που κατασκευάσαμε και θα το συγκρίναμε με τα υπόλοιπα βέλτιστα μοντέλα

Όσον αφορά το γραφικό περιβάλλον της εφαρμογής, θα μπορούσαμε να δώσουμε στον χρήστη την πληροφορία του σφάλματος που έχει προκύψει από την εκπαίδευση του μοντέλου, ώστε να γνωρίζει την απόδοση για κάθε μοντέλο που επιλέγει. Επίσης, να εμφανίζει εκτός από το σφάλμα και το διάγραμμα της σύγκρισης μεταξύ των δεδομένων ελέγχου και των προβλέψεων, όπως παρουσιάστηκαν στην 3^η ενότητα. Ακόμη, θα μπορούσαμε να δημιουργήσουμε ένα επιπλέον μενού, όπου ο χρήστης θα μπορεί να κάνει εκπαίδευση και αποθήκευση νέων μοντέλων με γραφικό τρόπο, με σκοπό να χρησιμοποιήσει τα δικά του μοντέλα για πρόβλεψη. Τέλος, μια ακόμη δυνατότητα θα ήταν να συνδέσουμε την εφαρμογή με κάποια βάση δεδομένων, ώστε να αποθηκεύονται οι προβλέψεις του χρήστη σε έναν πίνακα, εφόσον το επιλέξει, για να έχει πρόσβαση στην πληροφορία ανά πάσα στιγμή.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Chollet F, (2018). Deep Learning with Python, Manning
2. Geron A., (2019). Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow of Deep Learning, O'Reilly
3. Michelucci, U (2018). Applied Deep Learning: A Case-Based Approach to Understanding Deep Neural Networks, Apress
4. Goodfellow I, Bengio Y, Courville A, (2016). Deep Learning (Adaptive Computation and Machine Learning series), The MIT Press
5. Buduma N, (2015). Fundamentals of Deep Learning, O'Reilly
6. Swamynathan M. (2017). Mastering Machine Learning with Python in Six Steps, Apress
7. Διαμαντάρας Κ., (2007). Τεχνητά Νευρωνικά Δίκτυα, Εκδόσεις Κλειδάριθμος.
8. Peter J. Brockwell, Richard A. Davis (2016). Introduction to TimeSeries and Forecasting 3rd Edition, Springer
9. Douglas C. Montgomery, Cheryl L Jennings, Kulahci M. (2015). Introduction to Time Series Analysis and Forecasting 2nd Edition, Wiley
10. Wilfredo Palma (2016). Time Series Analysis, Wiley
11. Russell S. & Norvig P., (2004). Τεχνητή Νοημοσύνη, μια σύγχρονη προσέγγιση, Εκδόσεις Κλειδάριθμος.
12. Haykin S., (2010). Νευρωνικά Δίκτυα και Μηχανική Μάθηση 3^η Έκδοση, Εκδόσεις Παπασωτηρίου.
13. Spall, J. C. (2003). Introduction to Stochastic Search and Optimization. Wiley. ISBN 0-471-33052-3.)
14. Diederik P. Kingma, Jimmy Ba (2014), Adam: A Method for Stochastic Optimization
15. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R, (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting
16. Abien Fred M. Agarap (2018). Deep Learning using Rectified Linear Units (ReLU)

ΔΙΑΔΙΚΤΥΑΚΟΙ ΣΥΝΔΕΣΜΟΙ

1. <https://www.python.org/>
2. <https://www.tensorflow.org/>
3. <https://www.kaggle.com/shreenidhipparagi/google-stock-prediction/code>
4. <https://medium.com/@zeeshanmulla/cost-activation-loss-function-neural-network-deep-learning-what-are-these-91167825a4de>
5. <https://iamtrask.github.io/2015/07/27/python-network-part2/>
6. <https://machinelearningmastery.com/use-timesteps-lstm-networks-time-series-forecasting/>

7. <https://machinelearningmastery.com/how-to-develop-multilayer-perceptron-models-for-time-series-forecasting/>
8. <https://towardsdatascience.com/time-series-forecasting-with-recurrent-neural-networks-74674e289816>
9. https://en.wikipedia.org/wiki/Artificial_neural_network
10. <https://algotrading101.com/learn/yfinance-guide/>
11. <https://streamlit.io/>
12. <https://numpy.org/>
13. <https://pandas.pydata.org/>
14. <https://keras.io/>