Τμήμα Μηχανικών Η/Υ και Πληροφορικής
Πανεπιστήμιο Πατρών
Πολυτεχνική Σχολή

Τομέας Εφαρμογών και Θεμελιώσεων της Επιστήμης των Υπολογιστών

Διδάσκων: Δημήτριος Κοσμόπουλος
Ακαδημαϊκό Έτος: 2025 – 2026
Ημ/νία Παράδοσης: 3/12/2025

Στατιστικές Μέθοδοι Μηχανικής Μάθησης

Μάθημα Επιλογής – CEID1509

1η Εργασία: Πρόβλεψη Τιμών Μετοχών με Γραμμική Παλινδρόμηση

Σωκράτης Μαντές | 1093421 | up1093421@ac.upatras.gr

1 Εισαγωγή

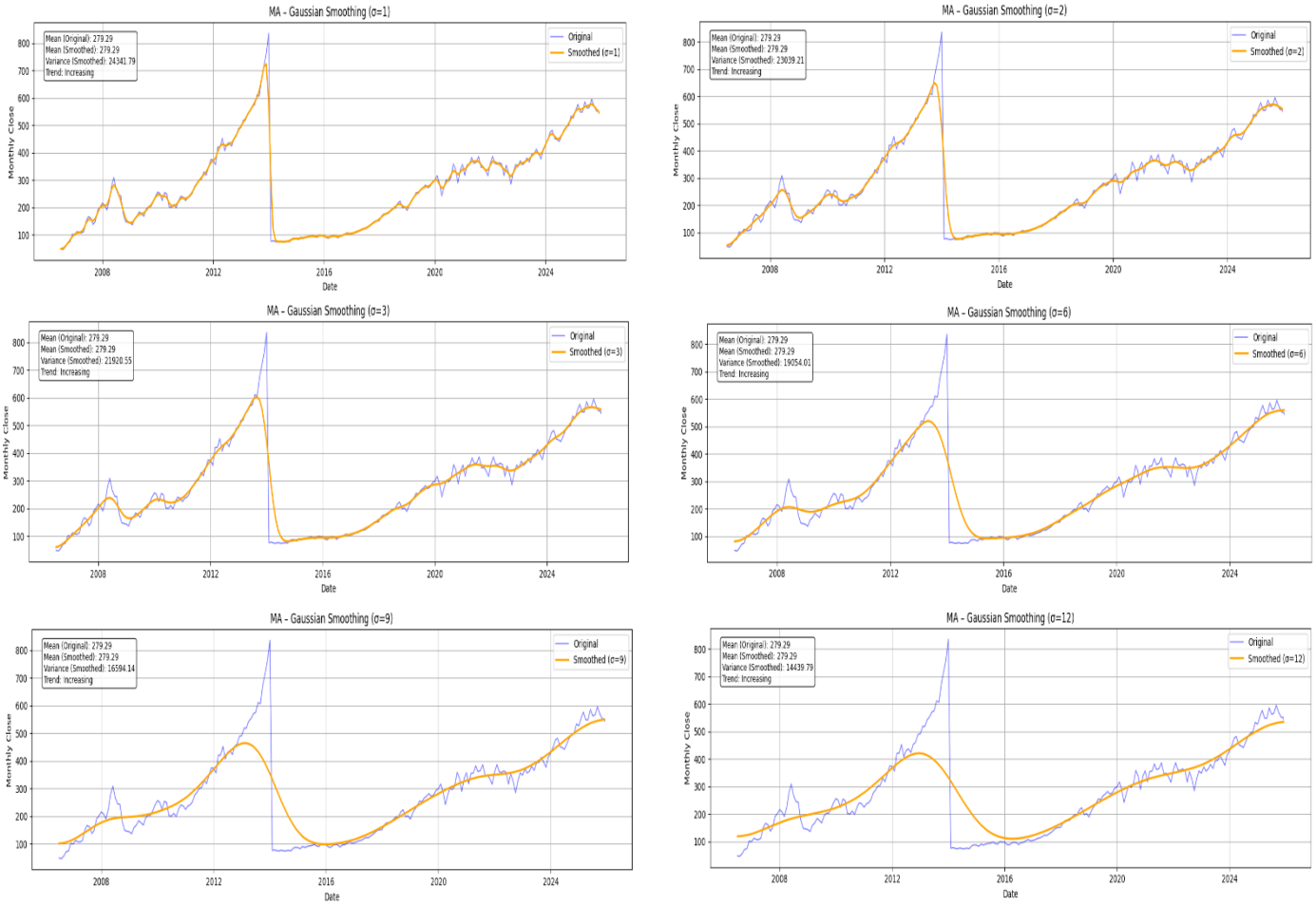
Το σύμβολο μετοχής που έχει επιλεχθεί είναι αυτό της Mastercard (MA) και στο dataset περιλαμβάνονται οι τιμές κλεισίματός της στο χρηματιστήριο από τις 24/05/2006 έως τις 28/11/2025. Τα ωμά ημερήσια δεδομένα αποθηκεύονται στο αρχείο close_prices_MA_daily_OHLCV.csv από όπου θα φορτώνονται στην συνέχεια για περαιτέρω προεπεξεργασία και στατιστική μελέτη. Ως training data ορίζουμε τις τιμές κλεισίματος μέχρι τις 31/12/2023, ως validation data τις τιμές από 01/01/2024 έως 28/11/2025.

Για την εκπαίδευση του μοντέλου επιλέχθηκαν τα **μηνιαία** και όχι ημερήσια δεδομένα, ώστε να γίνει η πρόβλεψη της **μέσης τιμής κλεισίματος του μήνα**, καθώς έχει νόημα οι είσοδοι και ο στόχος να είναι στην ίδια χρονική κλίμακα. Η μηνιαία κλίμακα λειτουργεί ουσιαστικά ως φυσικό *smoothing* πάνω στις ημερήσιες διακυμάνσεις, μειώνοντας τον τυχαίο θόρυβο και τα ακραία spikes των daily τιμών και αναδεικνύοντας καλύτερα τη μεσο-μακροπρόθεσμη τάση που ενδιαφέρει το πρόβλημα. Επιπλέον, τα μηνιαία lags (close_t-1, close_t-2, …) κουβαλάνε πιο διαφορετική πληροφορία μεταξύ τους σε σχέση με τα πολύ συσχετισμένα ημερήσια lags, με αποτέλεσμα πιο σταθερή εκπαίδευση και πιο ερμηνεύσιμους συντελεστές στα γραμμικά/πολυωνυμικά μοντέλα. Ο αριθμός των διαθέσιμων μηνών (≈234) είναι επαρκής για να εκπαιδευτούν μοντέλα με αρκετά lags χωρίς να «πνιγόμαστε» σε χιλιάδες daily παρατηρήσεις που θα απαιτούσαν πιο σύνθετες χρονοσειρές.

Data Preprocessing

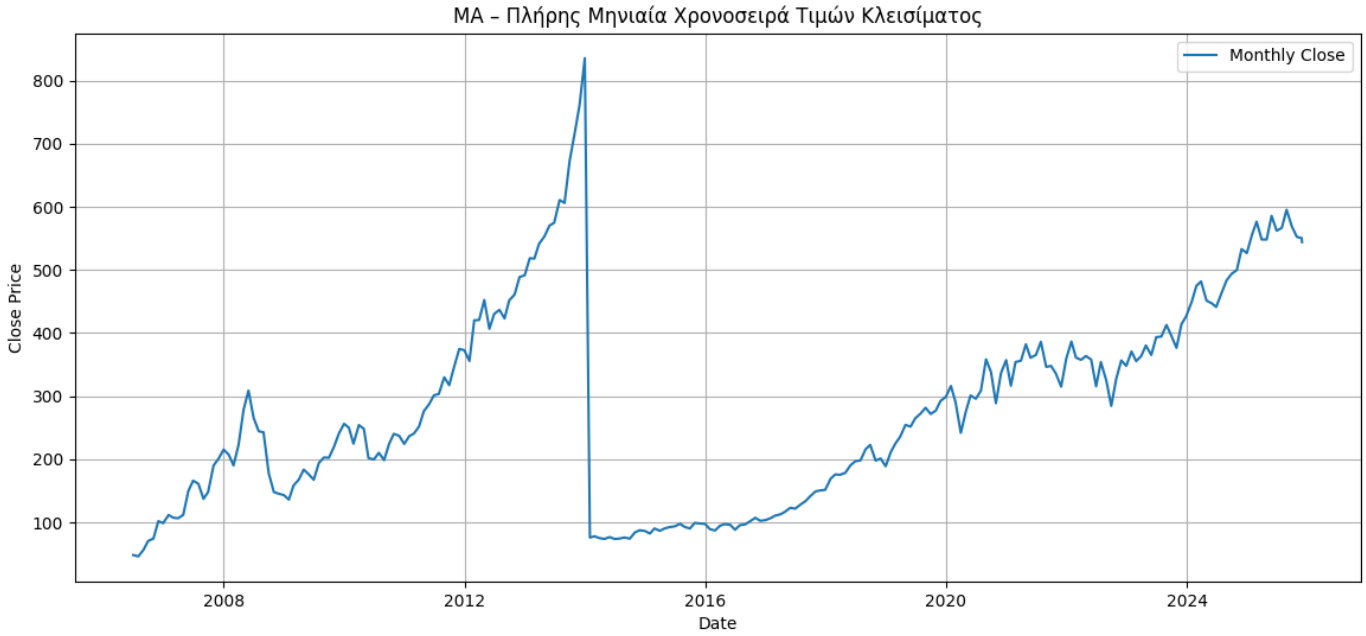
Ξεκινάμε την επεξεργασία με την εφαρμογή ενός Gaussian Filter στην αρχική χρονοσειρά με σκοπό να απομακρυνθούν οι ακραίες τιμές (outliers) και τυχόν θόρυβος και τελικά να προκύψει μια χρονοσειρά πιο εξομαλυμένη. Σκοπός είναι η τιμή της παραμέτρου sigma να είναι τέτοια ώστε να έχουμε ικανοποιητική αποθορυβοποίηση αλλά να μην συμβεί υπερβολικό smoothing στα δεδομένα και χαθεί έτσι σημαντική πληροφορία που υπάρχει ανάμεσα στις βραχυπρόθεσμες συσχετίσεις.

Για να κρίνουμε ποια τιμή sigma θα ήταν πιο κατάλληλη σχεδιάζουμε για ένα εύρος τιμών [1, 3, 6, 9, 12] την αρχική και την εξομαλυμένη χρονοσειρά, ενώ ταυτόχρονα υπολογίζουμε τη μέση τιμή, τη διασπορά και τη τάση (ανοδική ή καθοδική) σε κάθε εξομαλυμένη χρονοσειρά. Έτσι, μπορούμε να δούμε οπτικά σε ποια περίπτωση το smoothing δεν «καταστρέφει» την αρχική πληροφορία και τα short-term variations, καθώς μας ενδιαφέρει η βραχυπρόθεσμη πρόβλεψη τιμών κλεισίματος (επόμενη μέρα). Γενικά, μια μεγάλη τιμή του sigma, όπως επιβεβαιώνουμε και από την Εικόνα 1, εφαρμόζει ισχυρή εξομάλυνση κάνοντας τα δεδομένα πιο αργά στο να αντιδράσουν σε μικρές διακυμάνσεις. Συνεπώς, μας συμφέρει η επιλογή ενός χαμηλότερου sigma, όπως οι τιμές από 1 μέχρι 3. Επιλέγουμε να εργαστούμε με τιμή sigma = 2, καθώς έτσι το μοντέλο φαίνεται να μαθαίνει καλύτερα τις τρέχουσες τάσεις.



Το Gaussian Smoothing μας βοηθά να δούμε καθαρά τα βασικές τάσεις της μετοχής, όπως έντονη άνοδος, απότομη κρίση, ανάκαμψη και νέα άνοδος, φιλτράροντας τον μηνιαίο θόρυβο και αποφεύγοντας να εστιάσουμε υπερβολικά σε βραχυχρόνιες κινήσεις που μπορεί να είναι συγκυριακές.

Παρακάτω απεικονίζεται η πλήρης μηνιαία χρονοσειρά της τιμής κλεισίματος της μετοχής χωρίς καμία εξομάλυνση, οπότε βλέπουμε όλο τον «θόρυβο» και τις βραχυχρόνιες διακυμάνσεις. Στα πρώτα χρόνια παρατηρείται μια έντονα ανοδική πορεία, με αλλεπάλληλα υψηλότερα τοπικά χαμηλά και υψηλά μέχρι περίπου το 2013, όπου η τιμή φτάνει σε πολύ υψηλά επίπεδα. Στη συνέχεια εμφανίζεται μια εξαιρετικά απότομη πτώση, σχεδόν κατακόρυφη, που οδηγεί τη μετοχή σε πολύ χαμηλότερη ζώνη τιμών, το οποίο συνήθως συνδέεται με κάποιο σημαντικό εταιρικό γεγονός (π.χ. split, αναδιάρθρωση ή μεγάλη κρίση) και γι' αυτό πρέπει να είμαστε προσεκτικοί όταν συγκρίνουμε πριν και μετά. Από εκείνο το σημείο και μετά η σειρά δείχνει μια αρχική φάση σταθεροποίησης γύρω από σχετικά χαμηλές τιμές και στη συνέχεια, από το 2016 και μετά, ένα νέο ανοδικό σκέλος με αρκετές ενδιάμεσες διορθώσεις. Οι πτώσεις γύρω στο 2018–2019 και αργότερα την περίοδο της πανδημίας είναι ορατές ως απότομα «σκαλοπάτια» προς τα κάτω, αλλά συνολικά η κίνηση παραμένει ανοδική μέχρι τα τελευταία χρόνια, όπου η τιμή φαίνεται να κορυφώνεται και να διορθώνει ελαφρά. Γενικά, το γράφημα δείχνει μια μετοχή με σημαντική μεταβλητότητα και έντονες διακυμάνσεις, αλλά με μακροχρόνια θετική τάση, ειδικά στη μετα-κρίσης περίοδο.

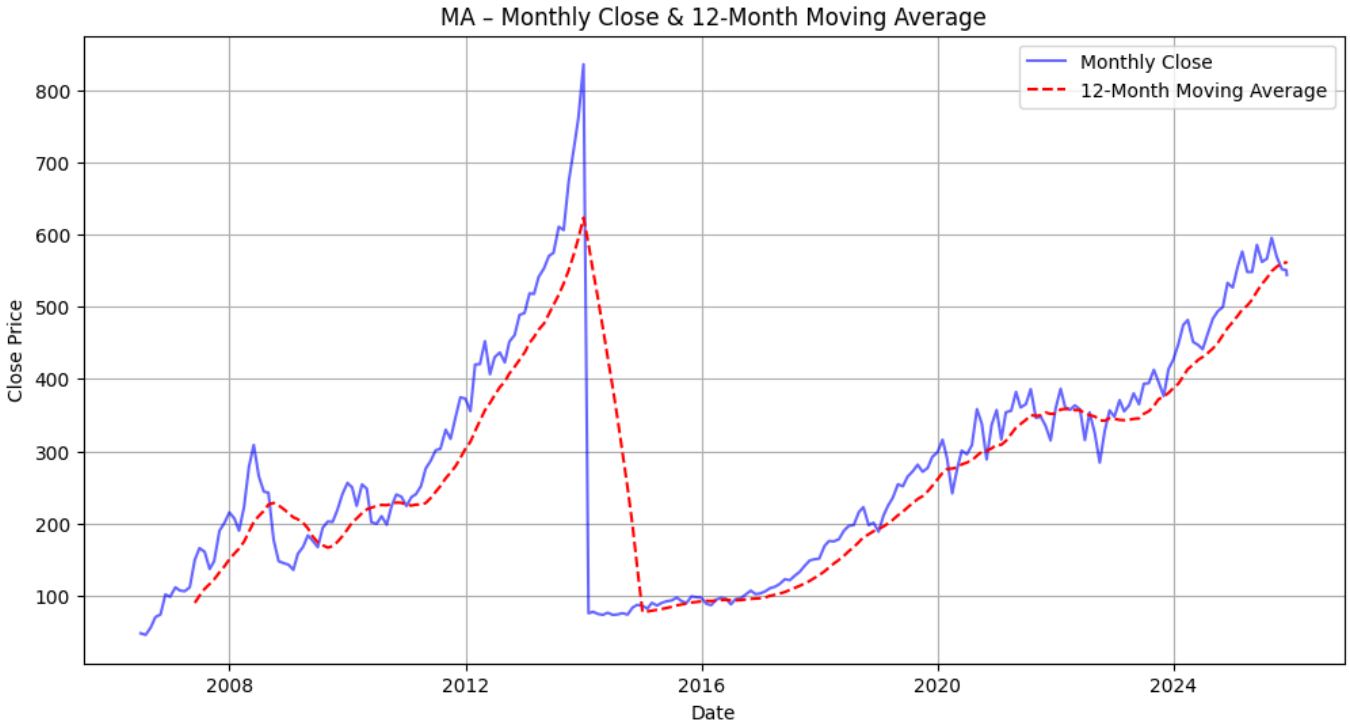


Το συμπέρασμα στο οποίο καταλήξαμε από την ανάλυση των διαγραμμάτων, αλλά και από τα πειράματα με διαφορετικό αριθμό καθυστερήσεων, μας καθοδηγεί και στην επιλογή των lagged features με τα οποία τροφοδοτούμε το μοντέλο γραμμικής παλινδρόμησης. Η δυναμική της μηνιαίας τιμής κλεισίματος αποτυπώνεται καλύτερα όταν το μοντέλο «βλέπει» τόσο τις πρόσφατες μεταβολές όσο και τη μεσοπρόθεσμη πορεία της μετοχής. Για αυτόν τον λόγο χρησιμοποιούμε ως χαρακτηριστικά τις ιστορικές μηνιαίες τιμές κλεισίματος και τον αντίστοιχο όγκο συναλλαγών, με καθυστερήσεις από 1 έως και N μήνες πίσω (close_t-1 … close_t-N και volume_t-1 … volume_t-N), και στη συνέχεια δοκιμάζουμε συστηματικά τιμές του N από 1 έως 12. Με αυτό τον τρόπο, δεν επιλέγουμε αυθαίρετα τον αριθμό των lags, αλλά αφήνουμε τα ίδια τα δεδομένα, μέσω των δεικτών σφάλματος στις περιόδους train και validation, να υποδείξουν ποιο εύρος καθυστερήσεων προσφέρει την καλύτερη ισορροπία μεταξύ προσαρμογής και γενίκευσης. Η χρήση τόσο των πρόσφατων μηνιαίων τιμών όσο και των παλαιότερων (π.χ. 6–12 μήνες πίσω) επιτρέπει στο μοντέλο να συλλάβει τα βασικά μοτίβα της σειράς, δηλαδή τις ισχυρές ανοδικές ή πτωτικές φάσεις αλλά και τις φάσεις σταθεροποίησης, χωρίς να χάνει την πληροφορία των βραχυπρόθεσμων κινήσεων που αποτυπώνονται στον μηνιαίο όγκο και στην πιο πρόσφατη τιμή κλεισίματος.

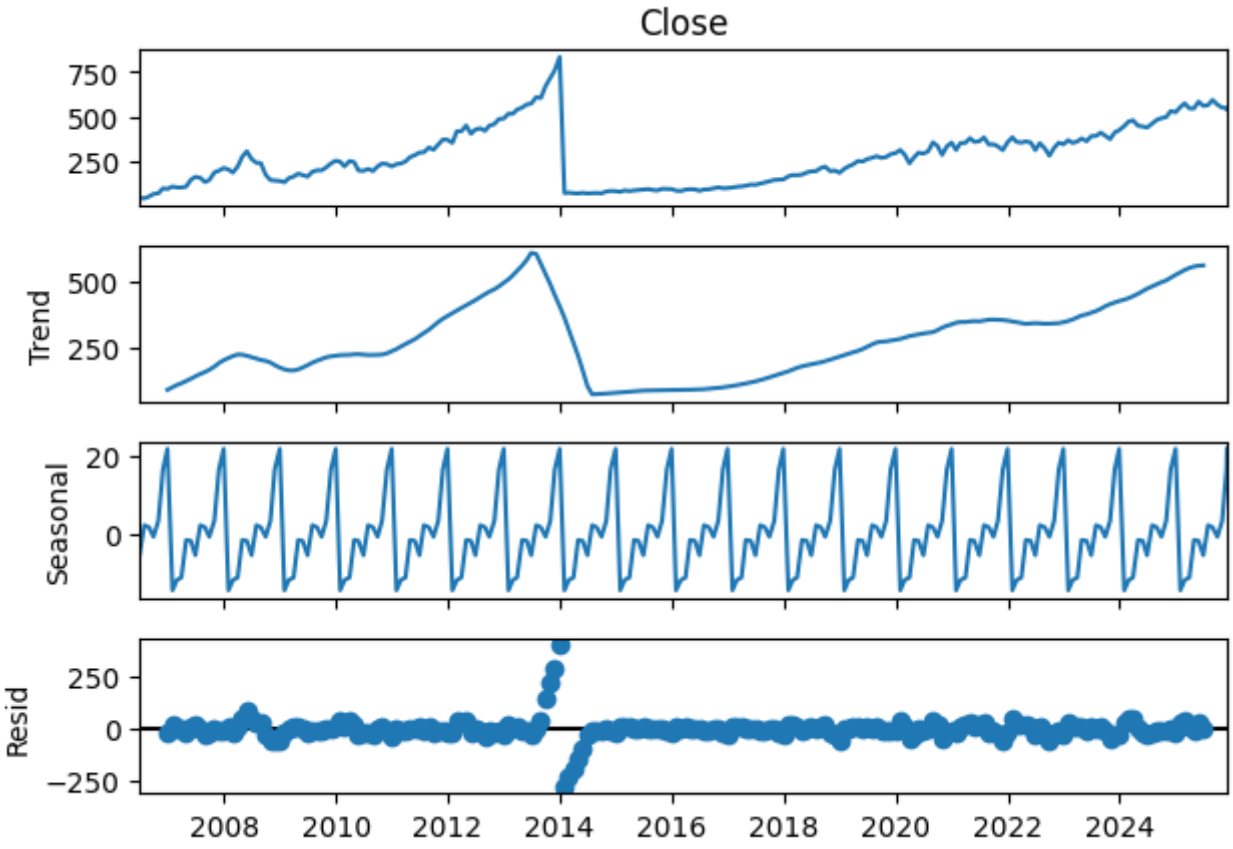
Statistic Study

Αρχικά, με την χρήση του Simple Moving Average (SMA) στοχεύουμε στο να εντοπίσουμε τη συνολική απόδοση της μετοχής σε μεσοπρόθεσμο ορίζοντα, καθώς απομονώνει τις πιο σημαντικές διακυμάνσεις, καθιστώντας τις τάσεις πιο ορατές. Όταν η τιμή κλεισίματος είναι πάνω από την τιμή του SMA, αυτό μπορεί να υποδηλώνει ανοδική τάση. Αντίθετα, όταν είναι κάτω από την SMA, μπορεί να υποδηλώνει πτωτική τάση. Συγκεκριμένα, παρατηρούμε μία έντονα ανοδική πορεία μέχρι περίπου το 2013, όπου τόσο η τιμή όσο και ο SMA ανεβαίνουν σταδιακά, και στη συνέχεια μια απότομη πτώση που «σπάει» αυτή τη μακροχρόνια τάση. Ο 12-μηνος SMA ακολουθεί αυτή την κίνηση με καθυστέρηση, λειαίνοντας την απότομη διόρθωση και δείχνοντας ότι η αλλαγή κατεύθυνσης δεν ήταν απλή βραχυχρόνια διακύμανση αλλά αλλαγή καθεστώτος. Μετά το χαμηλό, η τιμή σταδιακά σταθεροποιείται και

στη συνέχεια κινείται ξανά πάνω από τον SMA, υποδηλώνοντας μια νέα ανοδική φάση, η οποία γίνεται πιο καθαρά ορατή χάρη στη μετακίνηση της κόκκινης γραμμής προς τα πάνω και τη μικρότερη απόσταση μεταξύ των δύο καμπυλών. Έτσι, ο 12-μηνος SMA μας βοηθά να ξεχωρίσουμε τις πραγματικές μεσοπρόθεσμες τάσεις από τις βραχυχρόνιες διακυμάνσεις της μετοχής.



Έπειτα, επιχειρούμε να κάνουμε seasonal decomposition των δεδομένων για περιόδους ενός χρόνου. Από εδώ μπορούμε να εντοπίσουμε τη συνολική τάση, την εποχικότητα και τα residuals, δηλαδή τα υπόλοιπα δεδομένα αφού αφαιρεθούν η τάση και η εποχικότητα.



Συνολικά, τα αποτελέσματα της εποχικής αποσύνθεσης επιβεβαιώνουν ότι η συμπεριφορά της μετοχής καθορίζεται κυρίως από τη μακροχρόνια τάση και λιγότερο από την εποχικότητα. Η Τάση δείχνει καθαρά μια ισχυρή ανοδική φάση μέχρι το 2013, ένα απότομο «σπάσιμο» λόγω της μεγάλης πτώσης και, στη συνέχεια, μια περίοδο σταθεροποίησης και νέας ανόδου. Αυτό σημαίνει ότι, σε μέσο και μακροπρόθεσμο ορίζοντα, η μετοχή έχει θετική πορεία, αλλά με μία έντονη δομική διακοπή στη μέση του δείγματος, την οποία πρέπει να λάβουμε υπόψη σε οποιοδήποτε μοντέλο πρόβλεψης. Η Εποχικότητα εμφανίζει μεν ένα επαναλαμβανόμενο ετήσιο μοτίβο, ωστόσο οι διακυμάνσεις της είναι σχετικά μικρές σε σχέση με το επίπεδο των τιμών, άρα παίζει δευτερεύοντα ρόλο σε σχέση με την τάση. Τέλος, τα κατάλοιπα αναδεικνύουν ότι το μεγαλύτερο μέρος του «ανεξήγητου» θορύβου συγκεντρώνεται γύρω από τη μεγάλη πτώση, υποδηλώνοντας ένα ισχυρό εξωγενές σοκ, ενώ στις υπόλοιπες περιόδους παραμένουν σε πιο λογικά επίπεδα.

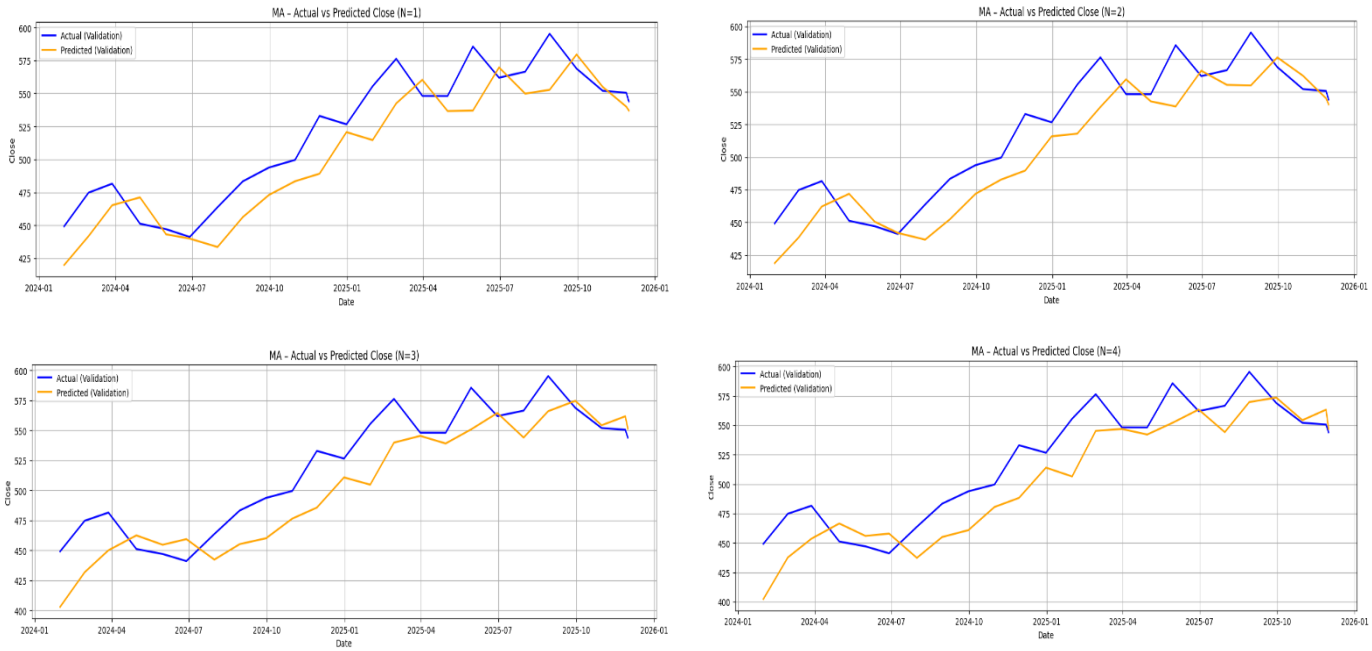
2 Γραμμική Παλινδρόμηση

Για την υλοποίηση του μοντέλου Γραμμικής Παλινδρόμησης ορίζουμε τη συνάρτηση linear_regression_model, η οποία παίρνει ως είσοδο μια τιμή N και το dataframe με τα δεδομένα και τα ταξινομεί χρονικά ως προς τη στήλη Date. Στη συνέχεια, για κάθε i από 1 μέχρι N, δημιουργεί δύο νέες στήλες: close_t-i και volume_t-i, που αντιστοιχούν στις τιμές Close και Volume i μήνες πίσω. Με αυτόν τον τρόπο, κάθε γραμμή του πίνακα "βλέπει" τις προηγούμενες N παρατηρήσεις, ώστε το μοντέλο να μπορεί να συλλάβει τη χρονική εξάρτηση της χρονοσειράς. Όσες γραμμές δεν έχουν πλήρες ιστορικό (λόγω των shifts) αφαιρούνται με dropna. Έπειτα, τα δεδομένα χωρίζονται χρονικά σε train (2018–2023) και validation (2024+), ώστε να ελέγξουμε την ικανότητα γενίκευσης στο πιο πρόσφατο διάστημα. Ορίζεται η λίστα χαρακτηριστικών (όλα τα lags close και volume) και κατασκευάζονται τα X_train, X_val και τα αντίστοιχα y.

Ακολουθεί ο χρονικός διαχωρισμός σε train και validation: τα δεδομένα από το 2018 έως και το 2023 χρησιμοποιούνται για εκπαίδευση, ενώ από το 2024 και μετά κρατούνται για έλεγχο σε "μελλοντική" περίοδο, χωρίς ανάμιξη των δύο συνόλων. Η συγκεκριμένη εξαετία επιλέχθηκε για εκπαίδευση, καθώς είναι αρκετά πρόσφατη, ώστε η συμπεριφορά της αγοράς σε αυτό το διάστημα να θεωρείται αντιπροσωπευτική για το τρέχον καθεστώς της μετοχής, αποφεύγοντας πολύ παλαιότερες περιόδους που μπορεί να διέπονται από διαφορετικές δομικές συνθήκες (π.χ. άλλα επίπεδα τιμών, διαφορετική μεταβλητότητα ή εταιρικά γεγονότα). Ταυτόχρονα, το 2018–2023 προσφέρει ικανοποιητικό αριθμό παρατηρήσεων για να εκπαιδευτεί σταθερά ένα γραμμικό μοντέλο με αρκετά lagged features, χωρίς όμως το train set να γίνεται υπερβολικά «ετερογενές». Επιπλέον, μέσα σε αυτή την περίοδο περιλαμβάνονται διαφορετικές φάσεις της αγοράς (π.χ. πιο ήρεμες περίοδοι, έντονες διακυμάνσεις, φάσεις ανόδου και διόρθωσης), κάτι που βοηθά το μοντέλο να μάθει μια πιο ρεαλιστική σχέση μεταξύ παρελθοντικών τιμών/όγκου και μελλοντικής τιμής κλεισίματος.

Έπειτα ορίζονται ρητά οι στήλες χαρακτηριστικών (όλα τα lags της Close και του Volume) και κατασκευάζονται οι πίνακες X και τα αντίστοιχα y για train και validation. Πριν δοθεί ο πίνακας στο μοντέλο, εφαρμόζεται StandardScaler, ο οποίος κανονικοποιεί κάθε χαρακτηριστικό ώστε να έχει συγκρίσιμη κλίμακα· αυτό είναι σημαντικό σε γραμμική παλινδρόμηση, γιατί

αποτρέπει το μοντέλο από το να δώσει δυσανάλογη έμφαση σε μεταβλητές με πολύ μεγάλα αριθμητικά μεγέθη (π.χ. όγκος). Πάνω στα κανονικοποιημένα δεδομένα εκπαιδεύεται το LinearRegression, και στη συνέχεια υπολογίζονται οι προβλέψεις τόσο στο train όσο και στο validation σετ. Από αυτές τις προβλέψεις εξάγονται τα σφάλματα MAE, MSE και RMSE για τις δύο περιόδους, ενώ αποθηκεύονται και τα βάρη (συντελεστές) και το bias του μοντέλου, ώστε να μπορούμε να γράψουμε ρητά την εξίσωση της παλινδρόμησης. Επίσης, βλέπουμε γραφικά το πόσο καλά προσεγγίζει η πρόβλεψη τις πραγματικές τιμές πάνω στη χρονοσειρά του συνόλου επικύρωσης. Η ίδια διαδικασία επαναλαμβάνεται για N από 1 έως 12, καταγράφοντας σε έναν πίνακα τα σφάλματα για κάθε περίπτωση και σχεδιάζοντας τα αντίστοιχα διαγράμματα. Με αυτόν τον τρόπο, δεν καταλήγουμε αυθαίρετα στον αριθμό των lags, αλλά αξιολογούμε συστηματικά πόσες καθυστερήσεις βελτιώνουν όντως την ικανότητα πρόβλεψης της γραμμικής παλινδρόμησης.



Ενδεικτικά για N = 1…4 το γραμμικό μοντέλο με τα lag features καταφέρνει να προσεγγίσει αρκετά καλά τη **γενική ανοδική τάση** της τιμής κλεισίματος στην περίοδο επικύρωσης. Επίσης, φαίνεται ότι το μοντέλο τείνει να είναι πιο **εξομαλυμένο**, δηλαδή υποεκτιμά τις κορυφώσεις και δεν αποτυπώνει πλήρως τις πιο απότομες πτώσεις, κάτι αναμενόμενο για μια απλή γραμμική παλινδρόμηση. Όσο αυξάνεται το N, οι προβλέψεις τείνουν να ευθυγραμμίζονται καλύτερα με τη συνολική πορεία, αλλά δεν εξαλείφονται πλήρως οι αποκλίσεις στις ακραίες τιμές. Αυτό υποδηλώνει ότι τα lag features δίνουν στο μοντέλο χρήσιμη πληροφορία για τη δυναμική της σειράς, αλλά υπάρχουν ακόμη μη γραμμικά στοιχεία ή εξωγενείς παράγοντες που δεν μπορούν να περιγραφούν μόνο με μια γραμμική σχέση των προηγούμενων μηνών.

Οι πιο πιθανοί λόγοι για την υποεκτίμηση των κορυφώσεων είναι:

- Η σχέση παρελθοντικές τιμές/όγκοι με την μελλοντική τιμή σπάνια είναι πραγματικά γραμμική. Η απλή γραμμική παλινδρόμηση τείνει να «ισιώνει» τις ακραίες κινήσεις και να δίνει πιο μέτριες προβλέψεις.

- Χρησιμοποιούμε μόνο lags της Close και του Volume. Δεν υπάρχουν άλλες μεταβλητές (π.χ. αποδόσεις, log-returns, μεταβλητότητα, τεχνικοί δείκτες, macro κ.λπ.), άρα το μοντέλο δεν βλέπει σημαντικές πληροφορίες που επηρεάζουν την τιμή.
- Η σειρά έχει έντονες αλλαγές καθεστώτος (π.χ. μεγάλη πτώση/άλμα). Ένα σταθερό γραμμικό μοντέλο με μία μόνο εξίσωση δύσκολα περιγράφει καλά και τις "ήρεμες" και τις "χαοτικές" περιόδους μαζί.
- Επιλέξαμε να εκπαιδεύσουμε από 2018–2023 και δοκιμάζουμε στο 2024+, άρα το μοντέλο ίσως δεν έχει «δει» αρκετά παρόμοια σενάρια με αυτά του validation (π.χ. νέα επίπεδα τιμών, διαφορετική μεταβλητότητα).

Metrics

	N	Train MAE	Train RMSE	Train MSE	Val MAE	Val RMSE	Val MSE
0	1	17.746597	22.647181	512.894829	20.531589	24.840400	617.045448
1	2	17.454499	22.253986	495.239895	20.148994	24.739971	612.066165
2	3	15.166165	19.962531	398.502655	22.554711	27.107902	734.838331
3	4	15.184301	19.816227	392.682852	21.248843	25.791126	665.182181
4	5	15.013905	19.736277	389.520635	21.222727	26.020962	677.090447
5	6	15.093457	19.465623	378.910487	19.875831	24.275312	589.290759
6	7	14.094525	18.015898	324.572579	19.489840	23.327580	544.175984
7	8	14.171530	17.842490	318.354460	19.315159	23.469652	550.824570
8	9	14.024577	17.750387	315.076238	19.604121	23.605251	557.207897
9	10	13.883635	17.518061	306.882445	20.876842	24.864092	618.223059
10	11	13.885684	17.243583	297.341147	22.143343	25.978823	674.899223
11	12	13.912793	16.964992	287.810951	23.738311	27.742756	769.660521

Γενικά, παρατηρούμε ότι για 3-4 lags το μοντέλο αποδίδει πολύ καλά στα δεδομένα του validation set, καθώς τα δεδομένα του 2024 παρουσιάζουν μια σταθερή τάση χωρίς απότομες μεταβολές.

Όσο αυξάνεται το **N**, τα σφάλματα στο **train** (Train MAE/RMSE/MSE) μειώνονται σταθερά, άρα το μοντέλο προσαρμόζεται όλο και καλύτερα στα ιστορικά δεδομένα. Στο **validation** όμως τα σφάλματα ακολουθούν σχήμα U, αρχικά βελτιώνονται, φτάνουν στο καλύτερο επίπεδο περίπου για **N=6–8** (ελάχιστο Val RMSE στο N=7, ελάχιστο Val MAE στο N=8) και μετά αρχίζουν ξανά να αυξάνονται, ένδειξη ότι για πολύ μεγάλα N το μοντέλο αρχίζει να υπερπροσαρμόζεται στον θόρυβο του train. Άρα, η καλύτερη ζώνη βρίσκεται γύρω στο **N≈7**, όπου πετυχαίνουμε καλή ισορροπία ανάμεσα σε χαμηλό σφάλμα στο validation και σχετικά απλό μοντέλο. Όσον αφορά τις παραμέτρους, στην επόμενη σελίδα θα παρουσιαστούν οι παράμετροι και η εξίσωση του μοντέλου Γραμμικής Παλινδρόμησης για κάθε διαφορετική τιμή του πλήθους των lagged features. Για τη πρόβλεψη θα χρησιμοποιηθεί το μοντέλο με την εξίσωση που αντιστοιχεί σε 7 lagged features. Οι παράμετροι του μοντέλου που θα χρησιμοποιηθεί σημειώνονται με έντονη γραφή πιο κάτω.

Προχωράμε στη πρόβλεψη για τα νέα δεδομένα με το μοντέλο που επιλέξαμε. Η συνάρτηση predict_next_month_close ταξινομεί το dataframe χρονικά με βάση τη Date και ελέγχει ότι υπάρχουν τουλάχιστον N μήνες διαθέσιμοι, ώστε να μπορέσουν να δημιουργηθούν τα αντίστοιχα lag features. Στη συνέχεια, «τραβάει» τις τελευταίες N τιμές της Close και του Volume, τις γυρίζει ανάποδα (ώστε η πιο πρόσφατη να είναι αυτή που αντιστοιχεί στο lag t-1 κ.λπ.) και τις συνενώνει σε ένα διάνυσμα εισόδου, με τη σειρά [close_lags, volume_lags]. Το διάνυσμα αυτό κανονικοποιείται με τον ίδιο scaler που είχε χρησιμοποιηθεί στην εκπαίδευση, ώστε η πρόβλεψη να είναι απολύτως συνεπής με το training set, και στη συνέχεια περνάει στο εκπαιδευμένο model.predict, από όπου προκύπτει η εκτιμώμενη τιμή κλεισίματος για τον επόμενο μήνα. Παράλληλα, βρίσκουμε την τελευταία διαθέσιμη ημερομηνία στο σύνολο δεδομένων, τη στρογγυλοποιούμε στο τέλος του μήνα και προσθέτουμε έναν ακόμη μήνα, ώστε να πάρουμε την ημερομηνία λήξης του επόμενου μήνα (π.χ. 31/12/2025).

Model Parameters for N=1:

Bias: 307.7215277777778

Weight for close_t-1: 66.1413417299311

Weight for volume_t-1: 3.3023728645944694

Model Equation:

Close_t = 307.7215277777778 + (66.1413417299311) * close_t-1 + (3.3023728645944694) * volume_t-1

Model Parameters for N=2:

Bias: 307.7215277777778

Weight for close_t-1: 52.27113713746613

Weight for close_t-2: 14.427224649833152

Weight for volume_t-1: 1.1564006923222006

Weight for volume_t-2: 2.0004959774884834

Model Equation:

Close_t = 307.7215277777778 + (52.27113713746613) * close_t-1 + (14.427224649833152) * close_t-2 + (1.1564006923222006) * volume_t-1 + (2.0004959774884834) * volume_t-2

Model Parameters for N=3:

Bias: 307.7215277777778

Weight for close_t-1: 42.50379021969372

Weight for close_t-2: -11.432236931503246

Weight for close_t-3: 36.39936468340156

Weight for volume_t-1: -1.6677177033099864

Weight for volume_t-2: -0.7601585105870745

Weight for volume_t-3: 4.218065036730242

Model Equation:

Close_t = 307.7215277777778 + (42.50379021969372) * close_t-1 + (-11.432236931503246) * close_t-2 + (36.39936468340156) * close_t-3 + (-1.6677177033099864) * volume_t-1 + (-0.7601585105870745) * volume_t-2 + (4.218065036730242) * volume_t-3

Model Parameters for N=4:

Bias: 307.7215277777778

Weight for close_t-1: 46.31211153121043

Weight for close_t-2: -11.716380747804617

Weight for close_t-3: 42.72484893296731

Weight for close_t-4: -9.92867626073722

Weight for volume_t-1: -1.5622509558894886

Weight for volume_t-2: -0.15027293276093173

Weight for volume_t-3: 4.66860560808709

Weight for volume_t-4: -0.7660925234620916

Model Equation:

Close_t = 307.7215277777778 + (46.31211153121043) * close_t-1 + (-11.716380747804617) * close_t-2 + (42.72484893296731) * close_t-3 + (-9.92867626073722) * close_t-4 + (-1.5622509558894886) * volume_t-1 + (-0.15027293276093173) * volume_t-2 + (4.66860560808709) * volume_t-3 + (-0.7660925234620916) * volume_t-4

Model Parameters for N=5:

Bias: 307.7215277777778

Weight for close_t-1: 47.16731253830539

Weight for close_t-2: -14.992943498655944

Weight for close_t-3: 42.825470922468554

Weight for close_t-4: -15.205936269682224

Weight for close_t-5: 7.659018034310494

Weight for volume_t-1: -1.5248153155022464

Weight for volume_t-2: -0.19013070987134473

Weight for volume_t-3: 4.148389352418651

Weight for volume_t-4: -1.3183760516163714

Weight for volume_t-5: 0.8740277600687454

Model Equation:

Close_t = 307.7215277777778 + (47.16731253830539) * close_t-1 + (-14.992943498655944) * close_t-2 + (42.825470922468554) * close_t-3 + (-15.205936269682224) * close_t-4 + (7.659018034310494) * close_t-5 + (-1.5248153155022464) * volume_t-1 + (-0.19013070987134473) * volume_t-2 + (4.148389352418651) * volume_t-3 + (-1.3183760516163714) * volume_t-4 + (0.8740277600687454) * volume_t-5

Model Parameters for N=6:

Bias: 307.7215277777778

Weight for close_t-1: 47.99448999509193

Weight for close_t-2: -14.582488587352358

Weight for close_t-3: 40.50195788372254

Weight for close_t-4: -14.05878661371698

Weight for close_t-5: 6.654648008544654

Weight for close_t-6: 1.1737436305379194

Weight for volume_t-1: -0.8533191554058311

Weight for volume_t-2: -0.4847102970153504

Weight for volume_t-3: 3.41912177693101

Weight for volume_t-4: -0.7659799996972001

Weight for volume_t-5: 2.685435906484751

Weight for volume_t-6: -3.7281694963357963

Model Equation:

Close_t = 307.7215277777778 + (47.99448999509193) * close_t-1 + (-14.582488587352358) * close_t-2 + (40.50195788372254) * close_t-3 + (-14.05878661371698) * close_t-4 + (6.654648008544654) * close_t-5 + (1.1737436305379194) * close_t-6 + (-0.8533191554058311) * volume_t-1 + (-0.4847102970153504) * volume_t-2 + (3.41912177693101) * volume_t-3 + (-0.7659799996972001) * volume_t-4 + (2.685435906484751) * volume_t-5 + (-3.7281694963357963) * volume_t-6

Model Parameters for N=7:

Bias: 307.7215277777778

Weight for close_t-1: 41.82572166197533

Weight for close_t-2: -9.417420485928925

Weight for close_t-3: 42.90754943373281

Weight for close_t-4: -20.997368023935312

Weight for close_t-5: 8.897871237405452

Weight for close_t-6: -7.678742408844959

Weight for close_t-7: 12.457386818706118

Weight for volume_t-1: -2.4175847928713545

Weight for volume_t-2: 1.7714922861693871

Weight for volume_t-3: 3.0556904099155067

Weight for volume_t-4: -2.334143317916483

Weight for volume_t-5: 3.0902705812716955

Weight for volume_t-6: -0.4165729423670446

Weight for volume_t-7: -6.802666246034741

Model Equation:

Close_t = 307.7215277777778 + (41.82572166197533) * close_t-1 + (-9.417420485928925) * close_t-2 + (42.90754943373281) * close_t-3 + (-20.997368023935312) * close_t-4 + (8.897871237405452) * close_t-5 + (-7.678742408844959) * close_t-6 + (12.457386818706118) * close_t-7 + (-2.4175847928713545) * volume_t-1 + (1.7714922861693871) * volume_t-2 + (3.0556904099155067) * volume_t-3 + (-2.334143317916483) * volume_t-4 + (3.0902705812716955) * volume_t-5 + (-0.4165729423670446) * volume_t-6 + (-6.802666246034741) * volume_t-7

Προχωράμε τώρα στη πρόβλεψη για τα νέα δεδομένα με το μοντέλο που επιλέξαμε για τον Δεκέμβριο 2025 και Γενάρη 2025. Αρχικά, ορίζουμε τη συνάρτηση predict_next_month_close. Αρχικά ταξινομεί το dataframe χρονολογικά με βάση τη στήλη Date και κάνει reset στο index, ώστε τα δεδομένα να είναι σε σωστή σειρά. Στη συνέχεια ελέγχει αν υπάρχουν τουλάχιστον N εγγραφές· αν όχι, σηκώνει σφάλμα γιατί δεν μπορεί να σχηματίσει όλα τα lags. Από τις στήλες Close και Volume παίρνει τις N πιο πρόσφατες τιμές, τις αντιστρέφει έτσι ώστε η πιο πρόσφατη να αντιστοιχεί σε lag t−1, και τις ενώνει σε ένα διάνυσμα εισόδου μήκους 2N. Αυτό το διάνυσμα περνάει πρώτα από τον scaler που είχε εκπαιδευτεί μαζί με το μοντέλο, ώστε οι τιμές να μπουν στην ίδια κλίμακα με τα δεδομένα εκπαίδευσης, και στη συνέχεια τροφοδοτείται στο γραμμικό μοντέλο για να παραχθεί η προβλεπόμενη τιμή κλεισίματος. Τέλος, η συνάρτηση βρίσκει την τελευταία ημερομηνία του dataframe, την προσαρμόζει στο τέλος του αντίστοιχου μήνα και προσθέτει έναν ακόμη μήνα, επιστρέφοντας έτσι ταυτόχρονα την ημερομηνία λήξης του επόμενου μήνα και την αντίστοιχη πρόβλεψη της Close. Στη συνέχεια βρίσκει την πραγματική τιμή κλεισίματος για τον Ιανουάριο 2025 από το df και υπολογίζει το απόλυτο και το ποσοστιαίο σφάλμα της πρόβλεψης σε σχέση με την πραγματικότητα.

3 Πολυωνυμική Παλινδρόμηση με Κανονικοποίηση Lasso

Για την υλοποίηση του μοντέλου Πολυωνυμικής Παλινδρόμησης με Κανονικοποίηση Lasso ορίζουμε τη συνάρτηση polynomial_regression_model_L1, η οποία παίρνει ως ορίσματα τον βαθμό του πολυωνύμου (degree) και τον αριθμό lags (N). Όπως και πριν, ξεκινάμε από ένα αντίγραφο του dataframe, το ταξινομούμε χρονικά και δημιουργούμε για κάθε i=1..N τα lagged features close_t-i και volume_t-i, ώστε κάθε γραμμή να «κουβαλά» τις N προηγούμενες τιμές της Close και του Volume. Αφού αφαιρεθούν οι γραμμές με NaN λόγω των shifts, τα δεδομένα χωρίζονται χρονικά σε train (μέχρι 31/12/2023) και validation (από

1/1/2024 και μετά). Στη συνέχεια ορίζουμε το σύνολο των χαρακτηριστικών (feature_cols), δηλαδή όλα τα lags της τιμής και του όγκου, και φτιάχνουμε τα X_train, X_val και τα αντίστοιχα y. Εδώ όμως δεν μένουμε μόνο στα «γραμμικά» lags: με το PolynomialFeatures δημιουργούμε όλες τις πολυωνυμικές επεκτάσεις μέχρι τον βαθμό degree (τετραγωνικοί όροι, διασταυρούμενοι όροι μεταξύ Close και Volume κ.λπ.), με include_bias=False ώστε να μην προστεθεί επιπλέον σταθερά. Έτσι το μοντέλο έχει τη δυνατότητα να περιγράψει μη γραμμικές σχέσεις μεταξύ των lagged features και της μελλοντικής τιμής κλεισίματος.

Σε αυτό το πλαίσιο, οι βασικές υπερπαράμετροι του μοντέλου L1 είναι ο βαθμός του πολυωνύμου (degree), ο αριθμός των χρονικών καθυστερήσεων N και η παράμετρος κανονικοποίησης α του Lasso. Για κάθε συνδυασμό degree και N παράγονται τα πολυωνυμικά χαρακτηριστικά πάνω στα lags της Close και της Volume, και η τιμή του α δεν επιλέγεται χειροκίνητα αλλά μέσω GridSearchCV με 5-fold cross–validation, ελαχιστοποιώντας το MSE στο train set. Με αυτό τον τρόπο προκύπτει ένας τριπλός συνδυασμός (degree*, N*, α*), όπου degree* και N* αντιστοιχούν στην περιοχή με τις χαμηλότερες τιμές Val RMSE / Val MAE, ενώ το α* είναι η τιμή κανονικοποίησης που βρέθηκε βέλτιστη από το grid search. Επειδή ο αριθμός των πολυωνυμικών χαρακτηριστικών αυξάνεται γρήγορα και υπάρχει κίνδυνος υπερπροσαρμογής, η L1 ποινή του Lasso «συρρικνώνει» πολλούς συντελεστές προς το μηδέν, κρατώντας κυρίως τα πιο σημαντικά χαρακτηριστικά. Αφού εκπαιδευτεί το τελικό μοντέλο πάνω σε όλα τα train δεδομένα, προκύπτει το διάνυσμα συντελεστών w και η σταθερά (bias/intercept), τα οποία ορίζουν πλήρως την πολυωνυμική παλινδρόμηση. Οι προβλέψεις σε train και validation χρησιμοποιούνται για τον υπολογισμό των μετρικών Train MAE/RMSE/MSE και Val MAE/RMSE/MSE, και τα αποτελέσματα συνοψίζονται σε πίνακες για βαθμούς πολυωνύμου 1–3 και N=1..6. Έτσι μπορούμε να συγκρίνουμε γραμμικό και μη γραμμικά επεκταμένο μοντέλο, να εντοπίσουμε πού εμφανίζεται overfitting και τελικά να επιλέξουμε συνδυασμό (degree, N, α) που πετυχαίνει χαμηλό σφάλμα στο validation χωρίς υπερβολική πολυπλοκότητα.

Metrics για Polynomial Degree 1:

	N	Train MAE	Train RMSE	Train MSE	Val MAE	Val RMSE	Val MSE
0	1	23.175515	56.720477	3217.212467	28.662989	33.678459	1134.238585
1	2	23.129353	56.658768	3210.216046	28.274349	33.344053	1111.825866
2	3	23.174998	56.602926	3203.891265	28.482347	33.429692	1117.544288
3	4	23.117198	56.580065	3201.303765	28.144222	33.061095	1093.036005
4	5	23.274595	56.467274	3188.553084	27.593893	32.575531	1061.165206
5	6	23.124113	56.422935	3183.547637	27.953487	32.897544	1082.248378

	alpha
0	0.001
1	0.001
2	10.000
3	10.000
4	10.000
5	10.000

Metrics για Polynomial Degree 2:

	N	Train MAE	Train RMSE	Train MSE	Val MAE	Val RMSE	Val MSE
0	1	24.587202	48.393434	2341.924486	72.199846	82.398599	6789.529086
1	2	22.519358	43.842489	1922.163847	39.537324	46.345017	2147.860573
2	3	21.237668	42.838623	1835.147579	37.912543	42.793272	1831.264130
3	4	20.270333	40.686385	1655.381924	38.401787	43.688963	1908.725450
4	5	19.860285	30.658875	939.966589	39.129087	52.184590	2723.231427
5	6	17.190478	26.575934	706.280251	46.573340	59.168532	3500.915161

	alpha
0	0.001
1	10.000
2	10.000
3	10.000
4	10.000
5	10.000

Metrics για Polynomial Degree 3:

	N	Train MAE	Train RMSE	Train MSE	Val MAE	Val RMSE
0	1	22.663678	34.286984	1175.597264	132.927774	160.667526
1	2	19.007485	25.821817	666.766242	65.376682	81.161329
2	3	16.003105	21.018333	441.770327	93.167991	117.414198
3	4	11.376585	15.724587	247.262634	73.022339	92.168568
4	5	5.840117	8.364675	69.967787	67.898509	84.066038
5	6	3.565625	5.510220	30.362527	80.887500	110.037501

	Val MSE	alpha
0	25814.053811	1.000
1	6587.161270	0.001
2	13786.093923	0.001
3	8495.044969	10.000
4	7067.098687	0.001
5	12108.251644	1.000

Αν κοιτάξουμε τα αποτελέσματα της L1 για τους τρεις βαθμούς πολυωνύμου, φαίνεται καθαρά ότι το degree=1 (γραμμικό Lasso πάνω στα lags) είναι αυτό που συμπεριφέρεται καλύτερα στο validation. Για degree=1 τα train errors είναι σχεδόν σταθερά γύρω στα ~56 RMSE, ενώ το Val RMSE μειώνεται σταδιακά καθώς αυξάνεται το N και φτάνει στο ελάχιστο περίπου για N=5 (και πολύ κοντά για N=6). Δηλαδή, με 5 μηνιαία lags της Close και του Volume το μοντέλο καταφέρνει να εκμεταλλευτεί την επιπλέον πληροφορία χωρίς να αρχίσει να υπερπροσαρμόζεται. Το GridSearch επέλεξε για τα μεγαλύτερα N σχετικά μεγάλο α≈10, κάτι που δείχνει ότι χρειάζεται αρκετά ισχυρή L1 ποινή ώστε να «μαζέψει» τους συντελεστές και να αποφύγει το overfitting.

Για degree=2 βλέπουμε άλλο μοτίβο: όσο αυξάνεται το N τα train errors πέφτουν σημαντικά, αλλά το Val RMSE έχει ένα ελάχιστο γύρω στο N=3 και μετά ξανανεβαίνει. Αυτό είναι κλασική ένδειξη ότι το πολυωνυμικό μοντέλο 2ου βαθμού με πολλά lags αρχίζει να προσαρμόζεται υπερβολικά στο train set. Επιπλέον, ακόμη και στο καλύτερο σημείο (degree=2, N≈3) το Val RMSE είναι σημαντικά μεγαλύτερο από αυτό του degree=1 (περίπου 43 αντί για ~33), άρα συνολικά ο βαθμός 2 δεν προσφέρει βελτίωση στην ικανότητα πρόβλεψης.

Τέλος, για degree=3 τα πράγματα είναι ακόμη χειρότερα: τα train errors γίνονται πολύ μικρά όσο αυξάνει το N, αλλά τα Val RMSE / Val MAE εκτοξεύονται (πολύ πάνω από 80–100), δείχνοντας ισχυρό overfitting, παρότι το GridSearch προσπαθεί να ρυθμίσει το α. Το μοντέλο 3ου βαθμού είναι προφανώς υπερβολικά σύνθετο για το μέγεθος και τη δομή των δεδομένων μας.

Συνοψίζοντας, με βάση τα metrics του validation η πιο ισορροπημένη επιλογή για το L1 Lasso είναι:

- degree = 1 (γραμμικό πολυώνυμο),
- N = 5 lags της Close και του Volume,
- α ≈ 10 (όπως προέκυψε από το GridSearch).

Αυτός ο συνδυασμός δίνει το χαμηλότερο Val RMSE/MAE, κρατά το μοντέλο σχετικά απλό και επωφελείται από την L1 κανονικοποίηση χωρίς έντονο overfitting.

Για την πρόβλεψη χρησιμοποιείται η predict_next_month_close_poly και κάνει το ίδιο πράγμα με την προηγούμενη predict, αλλά για το πολυωνυμικό μοντέλο).

- Παίρνει το df, το βάζει σε σωστή χρονική σειρά και ελέγχει ότι υπάρχουν τουλάχιστον N μήνες για να φτιάξει τα lags.
- Από τις στήλες Close και Volume κρατά τις N πιο πρόσφατες τιμές, τις αντιστρέφει ώστε η πιο πρόσφατη να είναι το lag t−1, και τις ενώνει σε ένα διάνυσμα εισόδου x_raw.
- Στη συνέχεια εφαρμόζει ακριβώς τους ίδιους μετασχηματισμούς που έγιναν στην εκπαίδευση: πρώτα scaler.transform (StandardScaler) και μετά poly.transform (PolynomialFeatures). Έτσι το διάνυσμα βρίσκεται στον ίδιο χώρο χαρακτηριστικών με αυτόν που είδε το μοντέλο όταν εκπαιδεύτηκε.
- Το τελικό x_poly δίνεται στο εκπαιδευμένο model.predict και παίρνουμε την προβλεπόμενη τιμή κλεισίματος για τον επόμενο μήνα.

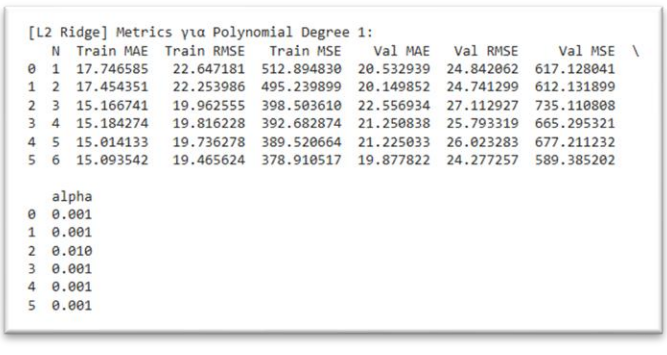
- Τέλος, με βάση την τελευταία ημερομηνία του dataframe, υπολογίζει την ημερομηνία λήξης του επόμενου μήνα και την επιστρέφει μαζί με την πρόβλεψη.

4 Πολυωνυμική Παλινδρόμηση με Κανονικοποίηση Ridge

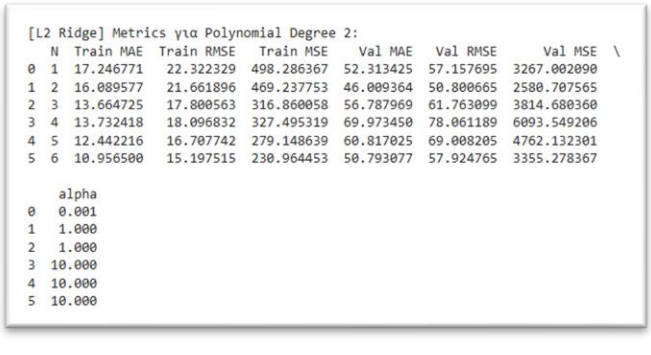
Τώρα υλοποιούμε ένα αντίστοιχο πλαίσιο με πριν, αλλά αυτή τη φορά για πολυωνυμική παλινδρόμηση με L2 κανονικοποίηση (Ridge), πάνω σε lags της Close και του Volume. Η συνάρτηση polynomial_regression_model_L2 δέχεται ως ορίσματα τον βαθμό του πολυωνύμου (degree), τον αριθμό των lags (N) και το dataframe με τα δεδομένα.

Στην αρχή κρατάμε μόνο τις βασικές στήλες Date, Close, Volume, τις ταξινομούμε χρονολογικά και κάνουμε reset στο index. Σε έναν βρόχο από 1 μέχρι N δημιουργούμε τα lagged features close_t-1…close_t-N και volume_t-1…volume_t-N με shift, πετάμε τις γραμμές με NaN ώστε να μείνουν μόνο πλήρη ιστορικά, και στη συνέχεια χωρίζουμε τα δεδομένα σε train (2018–2023) και validation (2024+). Τα features ορίζονται ως όλα τα lags της Close και του Volume, κατασκευάζουμε τα X_train, X_val και τα αντίστοιχα y και εφαρμόζουμε StandardScaler στα X, ώστε τα lags να βρεθούν στην ίδια κλίμακα – κρίσιμο βήμα για να λειτουργήσει σωστά η L2 ποινή. Πάνω στα κανονικοποιημένα X εφαρμόζουμε PolynomialFeatures μέχρι τον βαθμό degree, δημιουργώντας γραμμικούς, τετραγωνικούς και αλληλεπιδρώντες όρους μεταξύ Close και Volume. Επειδή ο αριθμός των χαρακτηριστικών αυξάνεται γρήγορα, χρειαζόμαστε κανονικοποίηση: το Ridge (L2) «τιμωρεί» μεγάλα weights και τα κρατά σε λογικές τιμές, χωρίς να τα μηδενίζει όπως το Lasso.

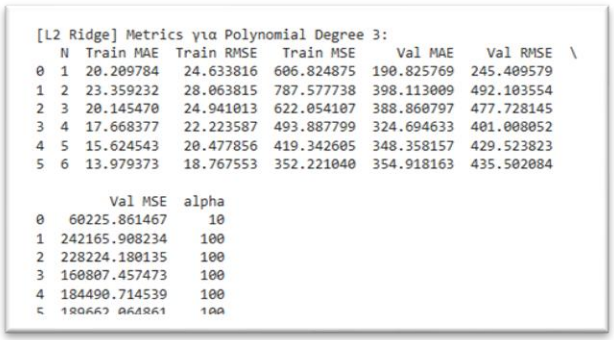
Οι βασικές υπερπαράμετροι του μοντέλου είναι ο βαθμός degree, ο αριθμός lags N και η παράμετρος κανονικοποίησης α. Για κάθε συνδυασμό degree, N χρησιμοποιούμε GridSearchCV με 5-fold cross-validation πάνω στο train set για να βρούμε το βέλτιστο α, ελαχιστοποιώντας το MSE. Έτσι προκύπτει ένας τελικός συνδυασμός (degree*_L2, N*_L2, α*_L2) που δίνει τη χαμηλότερη απόδοση σφάλματος στο validation. Το τελικό Ridge μοντέλο ορίζεται από το διάνυσμα συντελεστών w πάνω στα πολυωνυμικά χαρακτηριστικά και τη σταθερά (bias/intercept). Από τις προβλέψεις σε train και validation υπολογίζουμε τις μετρικές Train MAE, Train RMSE, Train MSE και Val MAE, Val RMSE, Val MSE, τις οποίες συνοψίζουμε σε πίνακες για degree=1..3 και N=1..6. Τα αποτελέσματα αυτά, μαζί με το αντίστοιχο α*_L2, μας επιτρέπουν να συγκρίνουμε διαφορετικές ρυθμίσεις, να εντοπίσουμε τυχόν overfitting και να επιλέξουμε τον συνδυασμό υπερπαραμέτρων που εξισορροπεί καλύτερα πολυπλοκότητα και απόδοση.



```
[L2 Ridge] Metrics για Polynomial Degree 1:
   N  Train MAE  Train RMSE   Train MSE    Val MAE   Val RMSE     Val MSE  \
0  1  17.746585   22.647181  512.894830  20.532939  24.842062  617.128041
1  2  17.454351   22.253986  495.239899  20.149852  24.741299  612.131899
2  3  15.166741   19.962555  398.503610  22.556934  27.112927  735.110808
3  4  15.184274   19.816228  392.682874  21.250838  25.793319  665.295321
4  5  15.014133   19.736278  389.520664  21.225033  26.023283  677.211232
5  6  15.093542   19.465624  378.910517  19.877822  24.277257  589.385202

   alpha
0  0.001
1  0.001
2  0.010
3  0.001
4  0.001
5  0.001
```



```
[L2 Ridge] Metrics για Polynomial Degree 2:
   N  Train MAE  Train RMSE   Train MSE    Val MAE   Val RMSE     Val MSE  \
0  1  17.246771   22.322329  498.286367  52.313425  57.157695  3267.002090
1  2  16.089577   21.661896  469.237753  46.009364  50.800665  2580.707565
2  3  13.664725   17.800563  316.860058  56.787969  61.763099  3814.680360
3  4  13.732418   18.096832  327.495319  69.973450  78.061189  6093.549206
4  5  12.442216   16.707742  279.148639  60.817025  69.008205  4762.132301
5  6  10.956500   15.197515  230.964453  50.793077  57.924765  3355.278367

    alpha
0   0.001
1   1.000
2   1.000
3  10.000
4  10.000
5  10.000
```



```
[L2 Ridge] Metrics για Polynomial Degree 3:
   N  Train MAE  Train RMSE   Train MSE     Val MAE    Val RMSE  \
0  1  20.209784   24.633816  606.824875  190.825769  245.409579
1  2  23.359232   28.063815  787.577738  398.113009  492.103554
2  3  20.145470   24.941013  622.054107  388.860797  477.728145
3  4  17.668377   22.223587  493.887799  324.694633  401.008052
4  5  15.624543   20.477856  419.342605  348.358157  429.523823
5  6  13.979373   18.767553  352.221040  354.918163  435.502084

         Val MSE  alpha
0   60225.861467     10
1  242165.908234    100
2  228224.180135    100
3  160807.457473    100
4  184490.714539    100
5  189662.064861    100
```

Για το Ridge (L2) βλέπουμε πολύ καθαρά ότι μόνο το degree = 1 (γραμμικό μοντέλο πάνω στα lags) συμπεριφέρεται ικανοποιητικά, ενώ τα degree 2 και 3 οδηγούν σε έντονο overfitting και πολύ μεγάλα σφάλματα στο validation.

- Για το Degree 1τα train errors είναι σχετικά σταθερά γύρω στα RMSE ≈ 19–22, ενώ τα Val RMSE κινούνται στην περιοχή 24–27. Όσο αυξάνεται το N, το train RMSE μειώνεται ελαφρά και το validation σφάλμα βελτιώνεται, με καλύτερη τιμή στο N=6 (Val RMSE ≈ 24.28, Val MSE ≈ 589, Val MAE ≈ 19.88). Επομένως, για degree=1 το N=6 δίνει την καλύτερη ισορροπία μεταξύ χαμηλού train error και χαμηλού validation error. Το GridSearch εδώ επιλέγει σταθερά α = 0.001, άρα μια ήπια αλλά παρούσα L2 ποινή είναι αρκετή για να σταθεροποιήσει το μοντέλο.
- Για το Degree 2 παρότι τα train errors μειώνονται (Train RMSE πέφτει έως ~15.2), τα Val RMSE είναι πολύ υψηλά (≈ 50–78) για όλα τα N. Αυτό σημαίνει ότι το πολυωνυμικό μοντέλο 2ου βαθμού, ακόμη και με Ridge, μαθαίνει υπερβολικά τις λεπτομέρειες του train set και γενικεύει πολύ χειρότερα από το απλό degree=1. Άρα δεν υπάρχει συνδυασμός N με degree=2 που να ανταγωνίζεται το degree=1.
- Για το Degree 3 τα train errors είναι σχετικά μικρά, αλλά τα Val RMSE φτάνουν σε τιμές > 200–400. Το μοντέλο 3ου βαθμού είναι υπερβολικά πολύπλοκο για τα διαθέσιμα δεδομένα, ακόμη και με μεγάλες τιμές α (10 ή 100).

Συνολικά, από τα metrics προκύπτει ότι το καταλληλότερο Ridge μοντέλο είναι:

- degree = 1 (γραμμική πολυωνυμική παλινδρόμηση),
- N = 6 lags της Close και του Volume,
- α = 0.001 (όπως δόθηκε από το GridSearch).

Αυτός ο συνδυασμός δίνει το μικρότερο Val RMSE / Val MSE ανάμεσα σε όλους τους δοκιμασμένους, με λογικό train error, και επομένως αποτελεί την προτεινόμενη επιλογή υπερπαραμέτρων για το L2 Ridge.

5 Μείωση Των Διαστάσεων

Εδώ δοκιμάζουμε διάφορες τεχνικές μείωσης διάστασης / επιλογής χαρακτηριστικών πάνω στο ίδιο πρόβλημα πρόβλεψης της μηνιαίας Close, πάντα με βασικό μοντέλο ένα Ridge. Πρώτα, η build_lagged_dataset παίρνει το αρχικό df και χτίζει το dataset με lags: κρατά μόνο Date, Close, Volume, τα ταξινομεί, και για i=1..N_best (π.χ. N_best=5) δημιουργεί close_t-i και volume_t-i. Αφού πεταχτούν τα NaN, γίνεται χρονικό split σε train (2018–2023) και validation (2024+). Επιστρέφονται τα X_train, X_val, y_train, y_val, τα ονόματα των features και οι ημερομηνίες του validation. Η βοηθητική compute_metrics υπολογίζει MAE, RMSE και MSE, ώστε να έχουμε ενιαίο τρόπο αξιολόγησης. Στη συνέχεια εφαρμόζουμε StandardScaler στα X (ίδιος scaler για όλες τις μεθόδους) και χτίζουμε ένα baseline Ridge (α=1.0) χωρίς καμία μείωση διάστασης. Υπολογίζονται τα train/val σφάλματα και αποτελούν το σημείο αναφοράς μας: τι πετυχαίνουμε όταν χρησιμοποιούμε όλες τις 2N εισόδους (όλα τα lags της Close και του Volume). Μετά δοκιμάζουμε τρεις διαφορετικές στρατηγικές μείωσης διάστασης, πάντα πάνω στα κανονικοποιημένα X.

Πρώτα, χρησιμοποιούμε PCA με n_components=0.95, δηλαδή κρατάμε τόσους κύριους άξονες όσοι χρειάζονται για να εξηγήσουν περίπου το 95% της διασποράς των lags. Παίρνουμε έτσι νέα χαρακτηριστικά (γραμμικούς συνδυασμούς των αρχικών lags), εκπαιδεύουμε ξανά Ridge πάνω σε αυτά και υπολογίζουμε τα αντίστοιχα train/val MAE–RMSE, μαζί με το πόσες συνιστώσες κρατήσαμε και τι ποσοστό διασποράς εξηγούν.

Ύστερα εφαρμόζουμε Factor Analysis με έναν μικρό αριθμό παραγόντων n_factors, που προσπαθούν να αποδώσουν τις συσχετίσεις μεταξύ των lags μέσω λίγων λανθανουσών μεταβλητών. Η διαφορά που έχει με το PCA είναι ότι αντί να μεγιστοποιεί απλώς τη διασπορά, προσπαθεί να εξηγήσει τις **συσχετίσεις** των χαρακτηριστικών. Πάνω σε αυτούς τους παράγοντες εκπαιδεύουμε πάλι Ridge και βλέπουμε πώς αλλάζουν τα σφάλματα.

Τέλος, δοκιμάζουμε την wrapper μέθοδο SequentialFeatureSelector με Ridge, που ξεκινά με κανένα χαρακτηριστικό και προσθέτει σταδιακά εκείνα τα lags που βελτιώνουν περισσότερο τη μέση απόδοση στο train, μέχρι να κρατήσει n_select (π.χ. 4). Έτσι παίρνουμε ένα υποσύνολο από τα αρχικά features (π.χ. συγκεκριμένα close_t-3, volume_t-1 κ.λπ.), ξαναεκπαιδεύουμε Ridge μόνο πάνω σε αυτά και αξιολογούμε σφάλματα.

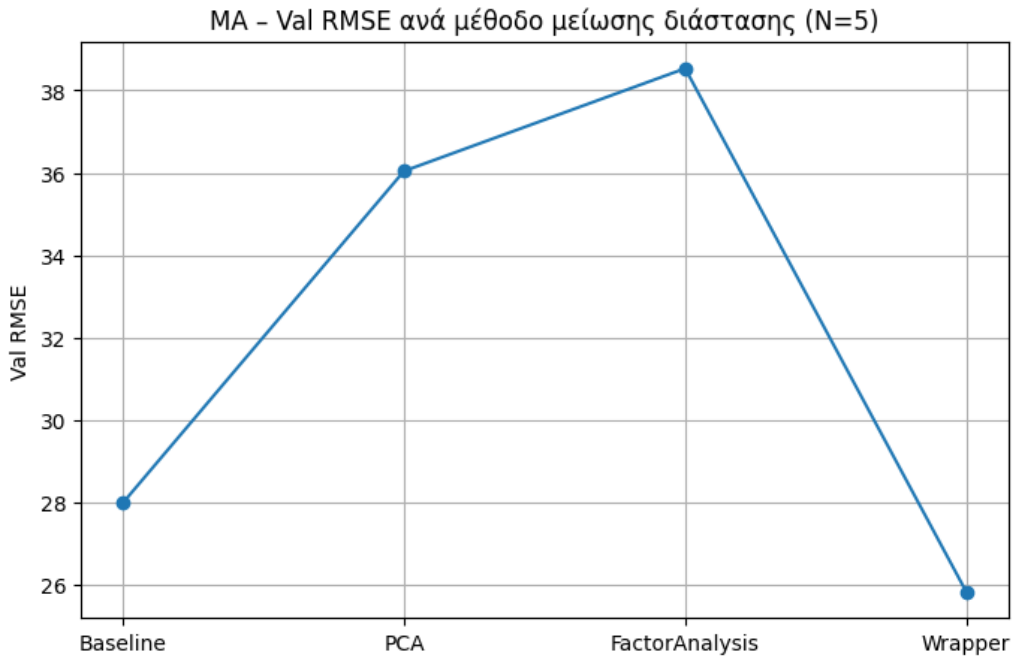
Στο τέλος, όλα τα αποτελέσματα συγκεντρώνονται στον πίνακα results_dimred, όπου για κάθε μέθοδο (Baseline, PCA, FactorAnalysis, Wrapper) καταγράφεται πόσες διαστάσεις χρησιμοποιεί, καθώς και οι μετρικές Train_MAE/RMSE και Val_MAE/RMSE/MSE. Ο πίνακας και το συνοδευτικό γράφημα της Val_RMSE ανά μέθοδο μας επιτρέπουν να συγκρίνουμε άμεσα αν η μείωση διάστασης βελτιώνει την γενίκευση σε σχέση με το baseline, ποια τεχνική αποδίδει καλύτερα για N_best lags και αν υπάρχει κέρδος από το να συμπυκνώσουμε ή να φιλτράρουμε τα lagged features σε λιγότερες, πιο «ουσιαστικές» διαστάσεις.

```
 PCA
PCA components: 6, explained variance: 98.11%
PCA - Train MAE:  16.87, RMSE: 22.47
PCA - Val   MAE:  29.46, RMSE: 36.02

 Factor Analysis (n_factors=2)
FA - Train MAE:  17.97, RMSE: 23.65
FA - Val   MAE:  31.80, RMSE: 38.51

 WRAPPER (Sequential Feature Selector)
Επιλεγμένα features: ['close_t-1', 'close_t-2', 'close_t-3', 'close_t-4']
Wrapper - Train MAE:  15.75, RMSE: 20.33
Wrapper - Val   MAE:  21.28, RMSE: 25.82
```

```
 Συνολικός πίνακας αποτελεσμάτων
           Model  Dimensionality  Train_MAE  Train_RMSE    Val_MAE  \
0       Baseline              10  15.133958   20.015323  22.878518
1            PCA               6  16.869969   22.471012  29.457560
2  FactorAnalysis               2  17.965430   23.654403  31.798850
3        Wrapper               4  15.747633   20.332216  21.284496

    Val_RMSE      Val_MSE
0  27.978543   782.798858
1  36.019657  1297.415673
2  38.514812  1483.390709
3  25.817762   666.556821
```

Συγκρίνοντας το Val RMSE των τεσσάρων μοντέλων παρατηρούμε τα εξής αποτελέσματα:

- Baseline (Ridge με όλα τα lags): Val RMSE ≈ 28. Αυτό είναι το σημείο αναφοράς· το μοντέλο βλέπει όλα τα 10 features (5 close + 5 volume) χωρίς καμία συμπίεση.
- PCA: Val RMSE ≈ 36. Η μείωση διάστασης μέσω κύριων συνιστωσών χειροτερεύει την πρόβλεψη· παρότι κρατά ~95% της διασποράς, φαίνεται ότι χάνει πληροφορία που είναι σημαντική για το σήμα πρόβλεψης.
- FactorAnalysis: Val RMSE ≈ 38.5. Η αναγωγή όλων των lags σε λίγους λανθάνοντες παράγοντες οδηγεί σε πολύ φτωχότερη προσαρμογή στο validation, άρα το μοντέλο γίνεται υπερβολικά «χονδροειδές».
- Wrapper (Sequential Feature Selector): Val RMSE ≈ 25.8, το χαμηλότερο από όλα. Επιλέγοντας ένα μικρό αλλά κατάλληλο υποσύνολο από τα αρχικά lags, η μέθοδος καταφέρνει να πετύχει καλύτερη γενίκευση από τόσο το baseline όσο και τις τεχνικές συμπίεσης



Συνεπώς, για N=5 η πιο αποδοτική στρατηγική είναι η wrapper μέθοδος επιλογής χαρακτηριστικών, ακολουθεί το baseline, ενώ PCA και Factor Analysis δεν βελτιώνουν την απόδοση στο σύνολο επικύρωσης.

6 Αποτελέσματα

Γραμμική Παλινδρόμηση

Προβλεπόμενη τιμή κλεισίματος για τον μήνα που λήγει στις 2025-12-31: 557.23 $

Προβλεπόμενη τιμή κλεισίματος για τον μήνα που λήγει στις 2025-01-31: 507.75 $

Πραγματική τιμή κλεισίματος για Ιανουάριο 2025: 555.43 $

Απόλυτο σφάλμα: 47.68 $

Ποσοστιαίο σφάλμα: 8.58%

Πολυωνυμική Παλινδρόμηση με Κανονικοποίηση Lasso

Προβλεπόμενη τιμή κλεισίματος για τον μήνα που λήγει στις 2025-12-31: 546.61 $

Προβλεπόμενη τιμή κλεισίματος για τον μήνα που λήγει στις 2025-01-31: 507.21 $

Πραγματική τιμή κλεισίματος για 01/2025: 555.43 $

Απόλυτο σφάλμα: 48.22 $

Ποσοστιαίο σφάλμα: 8.68%

Πολυωνυμική Παλινδρόμηση με Κανονικοποίηση Ridge

Προβλεπόμενη τιμή κλεισίματος για τον μήνα που λήγει στις 2025-12-31: 546.19 $

Προβλεπόμενη τιμή κλεισίματος για τον μήνα που λήγει στις 2025-01-31: 505.27 $

Πραγματική τιμή κλεισίματος για Ιανουάριο 2025: 555.43 $

Απόλυτο σφάλμα: 50.16 $

Ποσοστιαίο σφάλμα: 9.03%

Παρατίθεται το Github Repository με το συνολικό κώδικα στο αντίστοιχο Jupyter Notebook:
https://github.com/sokratismantes/Statistical-Methods-For-Machine-Learning/tree/main