

|  |
| --- |
| ΕΡΓΑΣΙΑ 1  Πρόβλεψη Τιμών Μετοχών με Γραμμική Παλινδρόμηση |

**ΘΕΩΡΙΑ ΑΠΟΦΑΣΕΩΝ**

**Ονοματεπώνυμο**: Χρυσαυγή Πατέλη

**ΔΙΔΑΣΚΩΝ** : Δημήτριος Κοσμόπουλος

**Α.Μ**.: 1084513

**Εξάμηνο**: 9o

**mail**: up1084513@ac.upatras.gr

Τμήμα Μηχανικών Η/Υ και Πληροφορικής

Ακαδημαϊκό Έτος 2024-2025

**ΠΕΡΙΕΧΟΜΕΝΑ**

[**1. Bιβλιοθήκες** 3](#_Toc183177075)

[**2. Προεπεξεραγασία** 4](#_Toc183177076)

[**3. Linear Regression** 8](#_Toc183177077)

[**3.1 Training-Validation** 8](#_Toc183177078)

[**3.2 Testing** 11](#_Toc183177079)

[**4. Lasso (L1 Κανονικοποίηση)** 12](#_Toc183177080)

[**4.1 Training-Validation** 12](#_Toc183177081)

[**4.2 Testing** 14](#_Toc183177082)

[**5.** **Ridge (L2 Κανονικοποίηση)** 15](#_Toc183177083)

[**5.1 Training-Validation** 15](#_Toc183177084)

[**5.2 Testing** 16](#_Toc183177085)

[**6.** **Συμπεράσματα** 17](#_Toc183177086)

# **Bιβλιοθήκες**

Οι βιβλιοθήκες που χρησιμοποιήθηκαν είναι:

* **requests:** χρησιμοποιείται για να κάνει HTTP αίτημα στο API της Alpha Vantage και να πάρει τα δεδομένα της μετοχής της google με μετοχικό σύμβολο GOOGL**.**
* **json:** χρησιμοποιείται για να διαχειριστεί αρχικά τα δεδομένα που είναι σε μορφή JSON.
* **csv:** χρησιμοποιείται για την ανάγνωση και την εγγραφή στο αρχείο csv στο οποίο βασίζεται το project (close\_prices.csv).
* **pandas:** χρησιμοποιεί DataFrames για την ανάλυση και διαχείριση δεδομένων.
* **matplotlib:** χρησιμοποιείται για την δημιουργία γραφημάτων (συνάρτηση **pyplot**) καθώς και για την πρόσβαση σε colormaps που επιτρέπουν την εφαρμογή χρωμάτων στα δεδομένα σε γραφήματα (συνάρτηση **cm**).
* **statsmodels:** παρέχει εργαλεία για στατιστική ανάλυση χρονοσειρών. Η συνάρτηση **seasonal\_decompose** χρησιμοποιείται για την αποσύνθεση της χρονοσειράς σε trend, seasonality και residual. Επίσης, χρησιμοποιείται η συνάρτηση **plot\_acf**, η οποία δημιουργεί ένα γράφημα που απεικονίζει την αυτοσυσχέτιση της χρονοσειράς για διάφορες τιμές lag.
* **scipy:** από συγκεκριμένη βιβλιοθήκη χρησιμοποιείται η συνάρτηση **gaussian\_filter1d** για την εφαρμογή Gaussian φίλτρου στα δεδομένα ώστε να τα εξομαλύνει.
* **sklearn:** από τηνσυγκεκριμένη βιβλιοθήκη χρησιμοποιούνται οι συναρτήσεις:
* **LinearRegression** είναι για την δημιουργία του μοντέλου γραμμικής παλινδρόμησης.
* **Lasso** είναι για την δημιουργία του μοντέλου που χρησιμοποιεί L1 κανονικοποίηση.
* **Ridge** είναι για την δημιουργία του μοντέλου που χρησιμοποιεί L2 κανονικοποίηση.
* **PolynomialFeatures** χρησιμοποιείται για να μετατρέψει τα χαρακτηριστικά σε πολυωνυμικά χαρακτηριστικά.
* **mean\_squared\_error, mean\_absolute\_error** είναι για τον υπολογισμό του μέσου τετραγωνικού σφάλματος και του μέσου απόλυτου σφάλματος.
* **numpy:** χρησιμοποιείται για επιστημονικούς υπολογισμούς και πιο συγκεκριμένα για τον υπολογισμό της τετραγωνικής ρίζας του μέσου τετραγωνικού σφάλματος (rmse).
* **joblib:** χρησιμοποιείται για την αποθήκευση και φόρτωση μοντέλων σε δυαδική μορφή.

# **2. Προεπεξεραγασία**

Στόχος της συγκεκριμένης εργασίας είναι η πρόβλεψη της τιμής κλεισίματος της επόμενης μέρας μιας μετοχής. Η μετοχή που επιλέχθηκε είναι της google class A και έχει σύμβολο **GOOGL**. Τα δεδομένα της μετοχής συλλέχθηκαν με αίτημα στο API της Alpha Vantage. Tα δεδομένα περιείχαν τις στήλες open, high, low, close, volume, ωστόσο η εργασία επικεντρώνεται στις τιμές κλεισίματος, οπότε αποθηκεύτηκαν σε ένα csv (close\_prices.csv) μόνο οι τιμές κλεισίματος (close), το οποίο είναι αυτό που θα χρησιμοποιηθεί. Το csv περιέχει τιμές από το 19-08-2004 μέχρι 18-11-2024.

Προκειμένου να μελετηθεί η μετοχή εκτενέστερα και να κατανοηθούν διάφορα μοτίβα και συσχετίσεις που περιέχει έγινε η παρακάτω προεπεξεργασία:

Αρχικά εξομαλύνθηκαν τα δεδομένα χρησιμοποιώντας gaussian φίλτρο με sigma=3, η επιλογή του sigma έγινε προκειμένου να επιτευχθεί μια ικανοποιητική εξομάλυνση των δεδομένων διατηρώντας ωστόσο αρκετές πληροφορίες και τάσεις της χρονοσειράς. Μια μεγαλύτερη τιμή θα οδηγούσε σε απώλεια σημαντικών πληροφοριών και τάσεων. Επιπλέον, δοκιμάστηκαν και άλλες τιμές sigma και τα σφάλματα κατά την αξιολόγηση των μοντέλων ήταν μεγαλύτερα.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Επίσης, αποτυπώνονται στην παρακάτω γραφική παράσταση οι τιμές κλεισίματος καθώς και οι εξομαλυμένες τιμές κλεισίματος που προκύπτουν από την χρήση gaussian φίλτρου με sigma=3. Η συγκεκριμένη γραφική δίνει μια συνολική εικόνα της συμπεριφοράς των τιμών κλεισίματος της μετοχής. Παρατηρείται:

1. Μέχρι το 2020 οι τιμές κλεισίματος εμφανίζουν διακυμάνσεις, αλλά και μια σταθερή ανοδική τάση.
2. Περίπου το 2021 παρατηρείται απότομη άνοδος της μετοχής.
3. Τέλος, στα μέσα του 2022 υπάρχει ξαφνική πτώση της μετοχής, η οποία στην συνέχεια σταθεροποιείται σε πολύ χαμηλές τιμές με ελάχιστες διακυμάνσεις σε σύγκριση με τις προηγούμενες περιόδους.

Εικόνα που περιέχει γραμμή, γράφημα, κείμενο, διάγραμμα

Περιγραφή που δημιουργήθηκε αυτόματα

Επιλέχθηκε η στατιστική μέθοδος **Seasonal Decomposition** χρησιμοποιώντας το προσθετικό μοντέλο (additive)**,** ώστε να διασπαστεί η χρονοσειρά στα βασικά της συστατικά (observed, trend, seasonal, residual). Με αυτόν τον τρόπο θα κατανοήσουμε καλύτερα την δομή και την τάση της μετοχής.

1. Αποτυπώνονται τα αρχικά δεδομένα (Observed), τα οποία αναλύθηκαν παραπάνω.

Εικόνα που περιέχει γραμμή, γράφημα, διάγραμμα, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

1. Στην επόμενη γραφική (Trend) αποτυπώνεται η μακροχρόνια συμπεριφορά της χρονοσειράς αφαιρώντας τι εποχικές διακυμάνσεις και τον θόρυβο. Παρατηρείται ότι μέχρι το 2021 η τάση είναι ανοδική και στην συνέχεια υπάρχει απότομη πτώση.

Εικόνα που περιέχει γραμμή, γράφημα, διάγραμμα, στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα

1. Στην γραφική Seasonal αποτυπώνεται η ετήσια εποχικότητα, η οποία είναι η επαναλαμβανόμενη διακύμανση των τιμών. Παρατηρείται ότι υπάρχει τακτική περιοδικότητα στις διακυμάνσεις των τιμών που επαναλαμβάνεται κάθε χρόνο.

Εικόνα που περιέχει στιγμιότυπο οθόνης, γραμμή, γράφημα, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

1. Τέλος η γραφική Residual αποτυπώνει τις τυχαίες διακυμάνσεις μετά την αφαίρεση της τάσης και της εποχικότητας. Μεγάλη αύξηση στις αποκλίσεις συμβαίνει κατα την περόδο του 2022 που υπήρχε η μεγάλη πτώση, τα προηγούμενα χρόνια τα residuals είναι σχετικά σταθερά.

Εικόνα που περιέχει κείμενο, γραμμή, γράφημα, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Ακόμη, δημιουργήθηκε η συνάρτηση αυτοσυχέτισης (**ACF**) χρησιμοποιώντας την βιβλιοθήκη **statsmodels**, προκειμένου να αποτυπωθεί η αυτοσυσχέτιση των δεδομένων της χρονοσειράς σε διαφορετικά χρονικά διαστήματα lags. Πιο συγκεκριμένα εξετάζει πόσο σχετίζεται μια τιμή με τις τιμές πριν από t ημέρες.

Παρατηρήσεις που προκύπτουν από το γράφημα:

1. Η ACF ξεκινά κοντά στο 1 στο lag 0, υποδεικνύοντας ισχυρή αυτοσυσχέτιση στην αρχή, κάτι που είναι αναμενόμενο, καθώς μια χρονοσειρά συσχετίζεται πάντα τέλεια με τον εαυτό της. Για μικρές τιμές εξακολουθεί να υπάρχει ισχυρή συσχέτιση.
2. Καθώς αυξάνουμε τα lag , η αυτοσυσχέτιση μειώνεται σταδιακά. Αυτό υποδηλώνει ότι οι ημερήσιες τιμές κλεισίματος της μετοχής της Google συσχετίζονται σε μεγάλο βαθμό σε μικρά χρονικά διαστήματα.
3. Η ACF φθίνει σταδιακά, μέχρι να πλησιάσει σχεδόν το μηδέν. Αυτό δείχνει ότι σε μεγαλύτερα χρονικά διαστήματα η συσχέτιση μεταξύ των τιμών μειώνεται.

Εικόνα που περιέχει κείμενο, γράφημα, γραμμή, διάγραμμα

Περιγραφή που δημιουργήθηκε αυτόματα

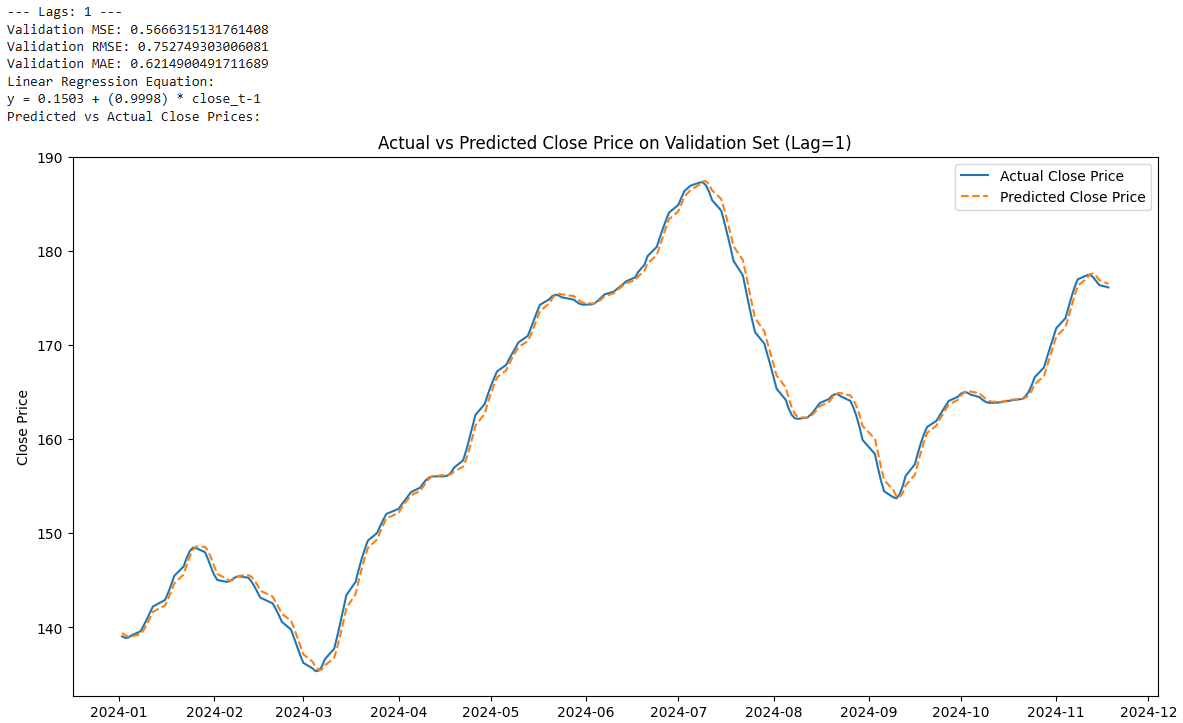
# **3. Linear Regression**

## **3.1 Training-Validation**

Αρχικά, εφαρμόζεται φιλτράρισμα στα δεδομένα χρησιμοποιώντας το gaussian filter με sigma=3 και δημιουργείται στο dataframe μια στήλη closed\_smoothed που περιέχει τις τιμές που θα χρησιμοποιηθούν. Στην συνέχεια, στο dataframe δημιουργούνται και στήλες που θα περιέχουν τα lagged features δηλαδή τις παρελθοντικές τιμές. Προκειμένου να βρεθεί το σωστό πλήθος lags και με βάση τις παρατηρήσεις που προέκυψαν από την προεπεξεργασία θα δοκιμαστεί το μοντέλο για διαφορετικό πλήθος παρελθοντικών τιμών από 1(close\_t-1) μέχρι 7 (close\_t-1…close\_t-7). Επιπλέον, διασπάται το dataframe σε σύνολο εκπαίδευσης (train set) και σύνολο επικύρωσης (validation set), το train set θα περιέχει τιμές πριν το 2024 και το validation set θα περιέχει τις τιμές μετά το 2024 μέχρι 18-11-2024.

* Το Χ\_train, Χ\_validation είναι τα lags και το y\_train, y\_validation είναι οι στόχοι, δηλαδή οι τιμές κλεισίματος της επόμενης μέρας.

Ύστερα, δημιουργείται το μοντέλο με την συνάρτηση **LinearRegression** και εκπαιδεύεται το μοντέλο γραμμικής παλινδρόμησης στα δεδομένα εκπαίδευσης με την μέθοδο **fit()** και πραγματοποιούνται προβλέψεις στα δεδομένα επικύρωσης με την μέθοδο **predict().** Tο μοντέλο αξιολογείται χρησιμοποιώντας τις μετρικές Μέσο τετραγωνικό σφάλμα (**MSE**),Τετραγωνική ρίζα του MSE (**RMSE**), Μέσο απόλυτο σφάλμα (**MAE**).

Η διαδικασία επαναλαμβάνεται για πλήθος lags από 1 μέχρι και 7 και τα αποτελέσματα για το κάθε μοντέλο αποθηκεύονται στην λίστα  **results\_list.** Για κάθε μοντέλο εκτυπώνεται η εξίσωση γραμμικής παλινδρόμησης και δημιουργείται η γραφική με τις προβλεπόμενες και τις πραγματικές τιμές για το σύνολο επικύρωσης. Παρατίθενται ενδεικτικά τα αποτελέσματα και οι γραφικές παρακάτω για lags από 1 μέχρι 4.

Εικόνα που περιέχει κείμενο, διάγραμμα, γράφημα, γραμμή

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, διάγραμμα, γραμμή, στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, διάγραμμα, στιγμιότυπο οθόνης, γράφημα

Περιγραφή που δημιουργήθηκε αυτόματα

Τέλος ανακτώνται από την λίστα results\_list τα σφάλματα (MSE, RMSE, MAE) για τις διάφορες τιμές των lags και δημιουργείται το παρακάτω γράφημα.

Εικόνα που περιέχει κείμενο, γραμμή, γράφημα, διάγραμμα

Περιγραφή που δημιουργήθηκε αυτόματα

Από το γράφημα φαίνεται ότι οι μετρικές σφάλματος (MSE, RMSE και MAE) μειώνονται σημαντικά από το lag 1 έως το lag 2 και στη συνέχεια φθάνουν σε χαμηλές και σταθερές τιμές από το lag 3 και μετά. Μέχρι το lag 4, τα σφάλματα έχουν φθάσει σε  σταθερή κατάσταση, γεγονός που υποδηλώνει ότι τα πρόσθετα lag  πέραν αυτού του σημείου δεν παρέχουν ουσιαστική βελτίωση της ακρίβειας. Οπότε γι'αυτό και θα χρησιμοποιηθεί το μοντέλο που έχει δημιουργηθεί με τα 4 lag. Το σφάλμα (mse) σε αυτήν την περίπτωση είναι πολύ μικρό και τείνει στο μηδέν **Validation MSE**: 0.0010647297112459037. αυτό σημαίνει ότι το μοντέλο έχει εκπαιδευτεί σωστά και οι προβλέψεις που κάνει είναι πολύ κοντά στις πραγματικές τιμες αν όχι ίδιες.

Μετά απο αυτήν την παρατήρηση προστέθηκαν στον κώδικα οι εντολές οι οποίες αποθηκεύουν το μοντέλο με lag = 4 χρησιμοποιώντας την βιβλιοθήκη joblib, ώστε να μπορεί να χρησιμοποιηθεί στο testing.

    # Store the model for lag 4

    if lags == 4:

        best\_model = model

        joblib.dump(best\_model, 'linear\_regression\_model.pkl')

## **3.2 Testing**

Χρησιμοποιώντας το μοντέλο που αποθηκεύτηκε όπως αναφέρθηκε και παραπάνω θα προβλεφθεί με την μέθοδο predict και με βάση τις 4 προηγούμενες τιμές (13-11-2024 μέχρι 18-11-2024) η τιμή για τις 19-11-2024 η οποία δεν βρίσκεται στο σύνολο των δεδομένων. Η τιμή που πρόβλεψε το μοντέλο είναι:



Ενώ η πραγματική τιμή της μετοχής για εκείνη την μέρα είναι:



Το μοντέλο πέτυχε μια πρόβλεψη, η οποία είναι πολύ κοντά στην πραγματική τιμή. Το σφάλμα MSE κατά το στάδιο της επικύρωσης ήταν 0.0010647, υποδεικνύοντας υψηλή ακρίβεια και καλή γενίκευση. Αυτό σημαίνει ότι τα δεδομένα έχουν σχεδόν γραμμική συσχέτιση.

# **Lasso (L1 Κανονικοποίηση)**

## **4.1 Training-Validation**

Ακολουθείται η ίδια διαδικασία με τo προηγούμενο μοντέλο, εφαρμόζεται φιλτράρισμα στα δεδομένα και διασπάται το dataframe σε σύνολο εκπαίδευσης (train set) και σύνολο επικύρωσης (validation set), το train set θα περιέχει τιμές πριν το 2024 και το validation set θα περιέχει τις τιμές μετά το 2024 μέχρι 18-11-2024.

Το lasso μοντέλο εκτός από το διαφορετικό πλήθος lags , θα δοκιμαστεί και για διαφορετικό πλήθος πολυωνύμων (από 1 μέχρι 5), ώστε να βρεθεί το καταλληλότερο μοντέλο για την συγκεκριμένη μετοχή.

Τα δεδομένα εισόδου πρέπει να μετατραπούν σε πολυωνυμκά χαρακτηριστικά, αυτό γίνεται δημιουργώντας ένα αντικείμενο της κλάσης **PolynomialFeatures,** το οποίο θα χρησιμοποιηθεί για να δημιουργήσει νέα χαρακτηριστικά που αντιστοιχούν στους πολυωνυμικούς όρους της εισόδου με βάση τον βαθμό του πολυωνύμου. Επίσης, εφαρμόζοντας την μέθοδο **fit\_transform()** στα δεδομένα εκπαίδευσης υπολογίζονται οι όροι που πρέπει να δημιουργηθούν για το πολυώνυμο. Επιπλέον, εφαρμόζεται η μέθοδος **transform()** στα δεδομένα επικύρωσης για να εφαρμοστούν τα ίδια πολυωνυμικά χαρακτηριστικά που δημιουργήθηκαν από τα δεδομένα εκπαίδευσης.

Ακόμη, πρέπει να σημειωθεί ότι κάθε φορά για το διαφορετικό πλήθος lags και βαθμών πολυωνύμου για την επιλογή της παραμέτρου alpha (ελέγχει την κανονικοποίηση που εφαρμόζεται στο μοντέλο) του μοντέλου lasso πραγματοποιείται cross-validation για τιμές (0.01, 0.1, 1, 10, 100) με την χρήση Grid Search. Πιο συγκεκριμένα, το λεξικό **param\_gid** περιλαμβάνει τις τιμές του alpha και δίνεται ως όρισμα στην συνάρτηση **GridSearchCV** σαν ορίσματα δίνονται επιπλέον το μοντέλο **lasso**, το **cv=5** που υποδηλώνει ότι τα δεδομένα θα χωριστούν σε 5 υποσύνολα και το μοντέλο θα εκπαιδευτεί σε καθένα από αυτά και τέλος το **scoring='neg\_mean\_squared\_error’** που θα χρησιμοποιηθεί η μετρική MSE για να ελεγχθεί η απόδοση του μοντέλου για κάθε alpha. Αφού ολοκληρωθεί η διαδικασία Grid Search η **best\_params\_** παρέχει την βέλτιστη παράμετρο alpha (μικρότερο MSE) και αποθηκεύεται την μεταβλητή **best\_alpha**. Ύστερα, δημιουργείται το τελικό μοντέλο με την συνάρτηση **Lasso** δίνοντας σαν παράμετρο την βέλτιστη τιμή **alpha** και τον αριθμό των επαναλήψεων **max\_iter=1000**.

Τέλος, εκπαιδεύεται το μοντέλο στα δεδομένα εκπαίδευσης με την μέθοδο **fit()** και πραγματοποιούνται προβλέψεις στα δεδομένα επικύρωσης με την μέθοδο **predict().** Tο μοντέλο αξιολογείται χρησιμοποιώντας τις μετρικές Μέσο τετραγωνικό σφάλμα (**MSE**),Τετραγωνική ρίζα του MSE (**RMSE**) και Μέσο απόλυτο σφάλμα (**MAE**). Η διαδικασία επαναλαμβάνεται για πλήθος lags από 1 μέχρι και 7 και πλήθος πολυωνύμου από 1 μέχρι και 5 τα αποτελέσματα για το κάθε μοντέλο αποθηκεύονται στην λίστα  **results\_list.** Για κάθε μοντέλο εκτυπώνονται τα σφάλματα Ενδεικτικά παρατίθενται για 4 μοντέλα οι τιμές σφάλματος.

Εικόνα που περιέχει κείμενο, γραμματοσειρά, στιγμιότυπο οθόνης, λευκό

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα που περιέχει κείμενο, γραμματοσειρά, στιγμιότυπο οθόνης, λευκό

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, γραμματοσειρά, στιγμιότυπο οθόνης, λευκό

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, γραμματοσειρά, λευκό, στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα

Από την λίστα results\_list ανακτώνται τα σφάλματα (MSE, RMSE, MAE) για τις διάφορες τιμές των lags και βαθμών και δημιουργούνται τα παρακάτω 3 γραφήματα.

Εικόνα που περιέχει γραμμή, γράφημα, στιγμιότυπο οθόνης, διάγραμμα

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει γραμμή, γράφημα, στιγμιότυπο οθόνης, κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει γραμμή, γράφημα, διάγραμμα, στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα

Με βάση τα γραφήματα, φαίνεται ότι οι μετρικές σφάλματος (MSE, RMSE και MAE) μειώνονται σημαντικά από το lag 1 έως το lag 2 και στη συνέχεια φθάνουν σε χαμηλή τιμή για lag = 3, αυτό δείχνει την ικανότητα του μοντέλου να εκμεταλλεύεται χρήσιμες πληροφορίες από παρελθοντικά δεδομένα χωρίς να κάνει overfitting. Για τους βαθμούς 1,2,4,5 για μεγαλύτερο πλήθος lag αυξάνεται το σφάλμα λόγω οverfitting. Οπότε ο καταλληλότερος συνδυασμός είναι lag=3 , degree=3, καθώς έχει ένα από τα χαμηλότερα MSE και προσφέρει το βέλτιστο σημείο ισορροπίας μεταξύ ακρίβειας πρόβλεψης, γενίκευσης (επιτρέπει την καταγραφή μη γραμμικών σχέσεων) και πολυπλοκότητας.

Μετά απο αυτήν την παρατήρηση προστέθηκαν στον κώδικα οι εντολές οι οποίες αποθηκεύουν το μοντέλο με lag = 3 και degree = 3 χρησιμοποιώντας την βιβλιοθήκη joblib, ώστε να μπορεί να χρησιμοποιηθεί για το testing.

## **4.2 Testing**

Χρησιμοποιώντας το μοντέλο με βαθμό πολυωνύμου 3 που αποθηκεύτηκε θα προβλεφθεί με την μέθοδο **predict** και με βάση τις 3 προηγούμενες τιμές (14-11-2024 μέχρι 18-11-2024) η τιμή για τις 19-11-2024 η οποία δεν βρίσκεται στο σύνολο των δεδομένων. Η τιμή που πρόβλεψε το μοντέλο είναι:



Ενώ η πραγματική τιμή της μετοχής για εκείνη την μέρα είναι:



Το μοντέλο έχει μεγαλύτερη διαφορά με την πραγματική τιμή και το MSE στην επικύρωση ήταν επίσης αρκετά μεγαλύτερο, κάτι που δείχνει ότι το μοντέλο δεν κατάφερε να γενικεύσει σωστά. Η L1 κανονικοποίηση που εφαρμόζει το Lasso οδηγεί σε μηδενισμό συντελεστών για χαρακτηριστικά που θεωρεί λιγότερο σημαντικά. Άρα,όλα τα χαρακτηριστικά στα δεδομένα παίζουν σημαντικό ρόλο, οπότε η μείωση χαρακτηριστικών οδήγησε σε απώλεια κρίσιμων πληροφοριών και χειρότερη απόδοση.

# **Ridge (L2 Κανονικοποίηση)**

## **5.1 Training-Validation**

Ακολουθείται η ίδια διαδιασία με τα προηγούμενα μοντέλα, εφαρμόζεται φιλτράρισμα στα δεδομένα και διασπάται dataframe σε σύνολο εκπαίδευσης (train set) και σύνολο επικύρωσης (validation set), το train set θα περιέχει τιμές πριν το 2024 και το validation set θα περιέχει τις τιμές μετά το 2024 μέχρι 18-11-2024.

Το ridge μοντέλο εκτός από το διαφορετικό πλήθος lags , θα δοκιμαστεί και για διαφορετικό πλήθος πολυωνύμων (από 1 μέχρι 4), ώστε να βρεθεί το καταλληλότερο μοντέλο για την συγκεκριμένη μετοχή.

Τα δεδομένα εισόδου πρέπει να μετατραπούν σε πολυωνυμικά χαρακτηριστικά, η διαδικασία είναι ίδια με αυτή που ακολουθήθηκε και για το μοντέλο ridge

Όσον αφορά το cross validation πραγματοποιούνται τα ίδια βήματα όπως και για το μοντέλο lasso.

Τέλος, δημιουργείται το μοντέλο με την συνάρτηση **Ridge** δίνοντας σαν παράμετρο την βέλτιστη τιμή **alpha** και τον αριθμό των επαναλήψεων **max\_iter=1000** και εκπαιδεύεται το μοντέλο στα δεδομένα εκπαίδευσης με την μέθοδο **fit()** και πραγματοποιούνται προβλέψεις στα δεδομένα επικύρωσης με την μέθοδο **predict().** Tο μοντέλο αξιολογείται χρησιμοποιώντας τις μετρικές Μέσο τετραγωνικό σφάλμα (**MSE**),Τετραγωνική ρίζα του MSE (**RMSE**) και Μέσο απόλυτο σφάλμα (**MAE**). Η διαδικασία επαναλαμβάνεται για πλήθος lags από 1 μέχρι και 7 και πλήθος πολυωνύμου από 1 μέχρι και 4 τα αποτελέσματα για το κάθε μοντέλο αποθηκεύονται στην λίστα  **results\_list.** Για κάθε μοντέλο εκτυπώνονται τα σφάλματα Ενδεικτικά παρατίθενται για 4 μοντέλα οι τιμές σφάλματος.

Εικόνα που περιέχει κείμενο, γραμματοσειρά, λευκό, στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα που περιέχει κείμενο, γραμματοσειρά, στιγμιότυπο οθόνης, λευκό

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, γραμματοσειρά, λευκό, στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα Εικόνα που περιέχει κείμενο, γραμματοσειρά, στιγμιότυπο οθόνης, λευκό

Περιγραφή που δημιουργήθηκε αυτόματα

Από την λίστα results\_list ανακτώνται τα σφάλματα (MSE, RMSE, MAE) για τις διάφορες τιμές των lags και βαθμών και δημιουργούνται τα παρακάτω 3 γραφήματα.

Εικόνα που περιέχει γραμμή, κείμενο, γράφημα, απόδειξη

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει γραμμή, γράφημα, στιγμιότυπο οθόνης, κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει γραμμή, απόδειξη, γράφημα, κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Με βάση τα γραφήματα, φαίνεται ότι οι μετρικές σφάλματος (MSE, RMSE και MAE) μειώνονται σημαντικά από το lag 1 έως το lag 2 και στη συνέχεια φθάνουν σε χαμηλές σχεδόν σταθερές τιμές από το lag 3 και μετά αυτό δείχνει την ικανότητα του μοντέλου να εκμεταλλεύεται χρήσιμες πληροφορίες από παρελθοντικά δεδομένα χωρίς να κάνει overfitting.. Μετά το lag 4, τα σφάλματα μειώνονται ελάχιστα γεγονός που υποδηλώνει ότι τα πρόσθετα lag  πέραν αυτού του σημείου δεν παρέχουν ουσιαστική βελτίωση της ακρίβειας. Οπότε γι'αυτό και θα χρησιμοποιηθεί το μοντέλο που έχει 4 lag. Αυτά ισχύουν για βαθμό πολυωνύμου μέχρι 3 όταν ο βαθμός γίνει 4 φαίνεται να overfitting, καθώς εμφανίζει μεγαλύτερα σφάλματα για περισσότερα lags. Οπότε θα επιλέξουμε βαθμό πολυωνύμου 3.

Μετά απο αυτήν την παρατήρηση προστέθηκαν στον κώδικα οι εντολές οι οποίες αποθηκεύουν το μοντέλο με lag = 4 και degree = 3 χρησιμοποιώντας την βιβλιοθήκη joblib, ώστε να μπορεί να χρησιμοποιηθεί για το testing.

## **5.2 Testing**

Χρησιμοποιώντας το μοντέλο με βαθμό πολυωνύμου 3 που αποθηκεύτηκε θα προβλεφθεί με την μέθοδο **predict** και με βάση τις 4 προηγούμενες τιμές (14-11-2024 μέχρι 18-11-2024) η τιμή για τις 19-11-2024 η οποία δεν βρίσκεται στο σύνολο των δεδομένων. Η τιμή που πρόβλεψε το μοντέλο είναι:



Ενώ η πραγματική τιμή της μετοχής για εκείνη την μέρα είναι:



Το μοντέλο έχει την καλύτερη απόδοση από όλα τα μοντέλα με χαμηλότερη διαφορά με την πραγματική τιμή και καλύτερο MSE. Το Ridge παρουσιάζει μεγαλύτερη σταθερότητα και ανθεκτικότητα στον θόρυβο, χάρη στη L2 κανονικοποίησή του.

# **Συμπεράσματα**

Παρατηρείται ότι, το μοντέλο που χρησιμοποίησε L2 κανονικοποίηση (Ridge) πραγματοποίησε την καλύτερη πρόβλεψη. Αυτό οφείλεται στο ότι το Ridge μειώνει τον θόρυβο στα δεδομένα χωρίς να αφαιρεί σημαντικές πληροφορίες, καταφέρνοντας έτσι να βρει μια ισορροπία ανάμεσα στη γενίκευση και την αποφυγή overfitting. Ακολουθεί το γραμμικό μοντέλο (Linear Regression), με μικρή διαφορά στο σφάλμα και στην προβλεπόμενη τιμή (177.57 για το Ridge έναντι 177.54 για το Linear). Αυτό υποδηλώνει ότι τα δεδομένα έχουν μια σχετικά απλή γραμμική ή ημιγραμμική σχέση, που μπορεί να περιγραφεί με αποτελεσματικότητα από αυτά τα δύο μοντέλα. Αντίθετα, το Lasso( L1 κανονικοποίηση), τείνει να μηδενίζει συντελεστές που θεωρεί λιγότερο σημαντικούς πράγμα που μπορεί να οδηγήσει στην εξάλειψη σημαντικών πληροφοριών, γι’ αυτό και σφάλμα είναι σημαντικά μεγαλύτερο (MSE = 0.24059) και η πρόβλεψη (177.07) αποκλίνει περισσότερο από την πραγματική τιμή (178.12). Συνοψίζοντας, το Ridge Regression αναδεικνύεται ως η καλύτερη επιλογή για το συγκεκριμένο σύνολο δεδομένων, αφού συνδυάζει ακρίβεια, σταθερότητα και αντοχή στον θόρυβο, χωρίς να αφαιρεί σημαντικές πληροφορίες από τα δεδομένα.