



UNIVERSITY OF
PATRAS
ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ

ΘΕΩΡΙΑ ΑΠΟΦΑΣΕΩΝ

ΕΡΓΑΣΙΑ 1

Πρόβλεψη Τιμών Μετοχών με Γραμμική Παλινδρόμηση

Ονοματεπώνυμο: Χρυσσαυγή Πατέλη

A.M.: 1084513

Εξάμηνο: 9^ο

mail: up1084513@ac.upatras.gr

ΔΙΔΑΣΚΩΝ : Δημήτριος Κοσμόπουλος

<https://github.com/chryssa-pat/Decision-Theory>

Τμήμα Μηχανικών Η/Υ και Πληροφορικής

Ακαδημαϊκό Έτος 2024-2025

ΠΕΡΙΕΧΟΜΕΝΑ

1. Βιβλιοθήκες.....	3
2. Προεπεξεργασία	4
3. Linear Regression.....	8
3.1 Training-Validation	8
3.2 Testing	11
4.Lasso Regression (L1 Κανονικοποίηση).....	12
4.1 Training-Validation	12
4.2 Testing	14
5.Ridge Regression (L2 Κανονικοποίηση)	15
5.1 Training-Validation.....	15
5.2 Testing	16
6.Συμπεράσματα.....	17

1. Βιβλιοθήκες

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 την συγκεκριμένη βιβλιοθήκη χρησιμοποιείται η συνάρτηση **gaussian_filter1d** για την εφαρμογή Gaussian φίλτρου στα δεδομένα ώστε να τα εξομαλύνει.
- **sklearn:** από την συγκεκριμένη 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 πολυωνμικά 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. Προεπεξεργασία

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. Τα 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.

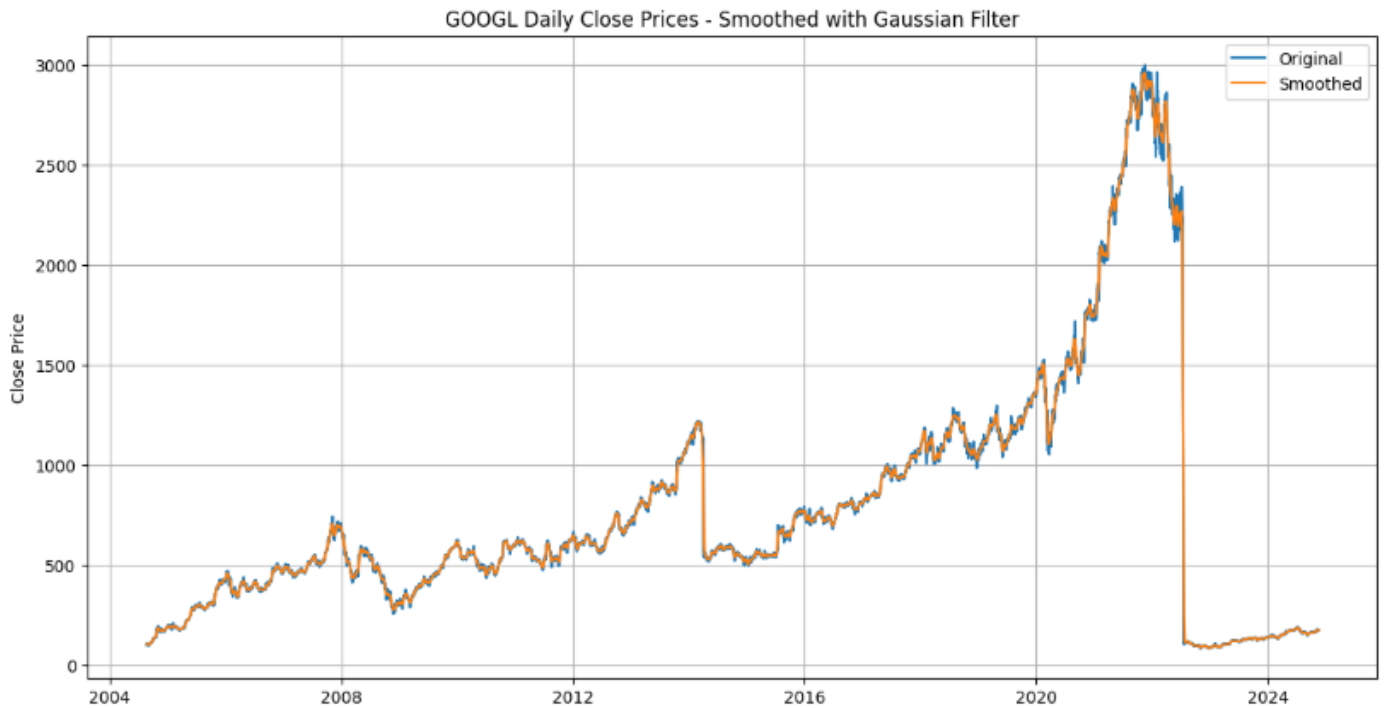
Προκειμένου να μελετηθεί η μετοχή εκτενέστερα και να κατανοηθούν διάφορα μοτίβα και συσχετίσεις που περιέχει έγινε η παρακάτω προεπεξεργασία:

Αρχικά εξομαλύνθηκαν τα δεδομένα χρησιμοποιώντας gaussian φίλτρο με $\sigma=3$, η επιλογή του σ 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 σ και τα σφάλματα κατά την αξιολόγηση των μοντέλων ήταν μεγαλύτερα.

	Close	Close_smoothed
Date		
2024-11-18	175.300	176.137720
2024-11-15	172.490	176.389924
2024-11-14	175.580	176.805351
2024-11-13	178.880	177.229916
2024-11-12	181.620	177.487574
...
2004-08-25	106.000	105.364413
2004-08-24	104.870	105.589222
2004-08-23	109.400	105.649125
2004-08-20	108.310	105.615239
2004-08-19	100.335	105.570576

Επίσης, αποτυπώνονται στην παρακάτω γραφική παράσταση οι τιμές κλεισίματος καθώς και οι εξομαλυσμένες 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.

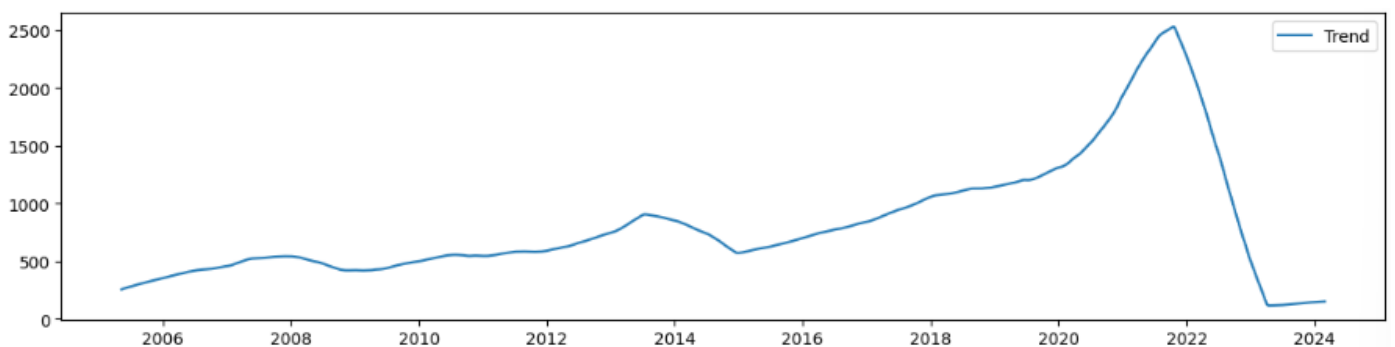


Επιλέχθηκε η στατιστική μέθοδος **Seasonal Decomposition** χρησιμοποιώντας το προσθετικό μοντέλο (additive), ώστε να διασπαστεί η χρονοσειρά στα βασικά της συστατικά (observed, trend, seasonal, residual). Με αυτόν τον τρόπο θα κατανοήσουμε καλύτερα την δομή και την τάση της μετοχής.

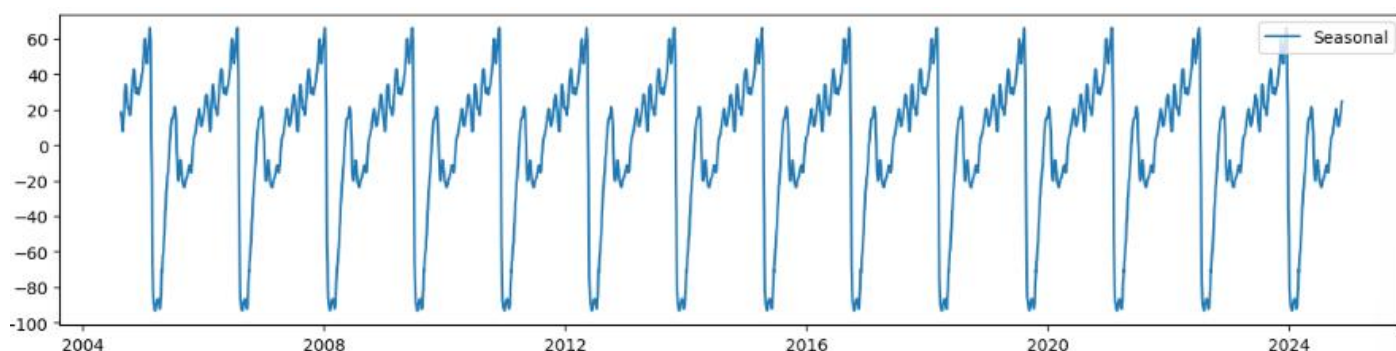
1. Αποτυπώνονται τα αρχικά δεδομένα (Observed), τα οποία αναλύθηκαν παραπάνω.



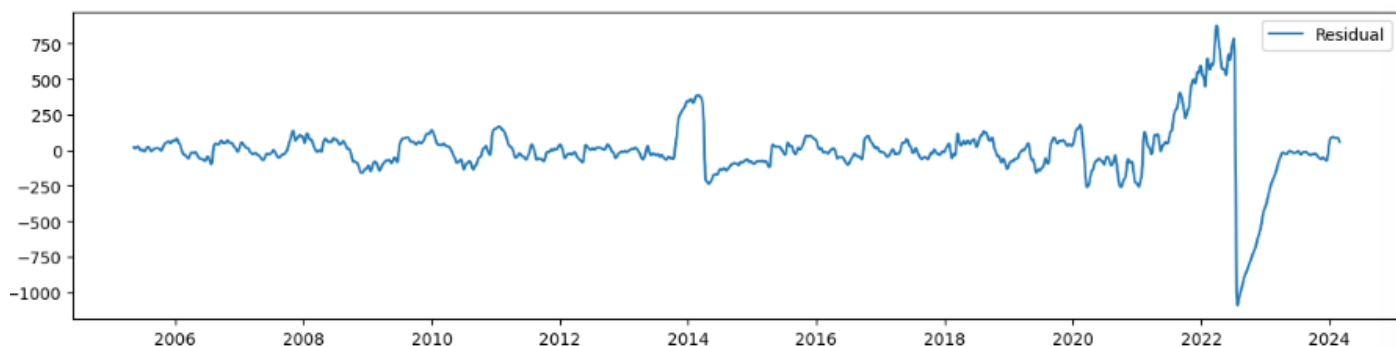
2. Στην επόμενη γραφική (Trend) αποτυπώνεται η μακροχρόνια συμπεριφορά της χρονοσειράς αφαιρώντας τι εποχικές διακυμάνσεις και τον θόρυβο. Παρατηρείται ότι μέχρι το 2021 η τάση είναι ανοδική και στην συνέχεια υπάρχει απότομη πτώση.



3. 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.



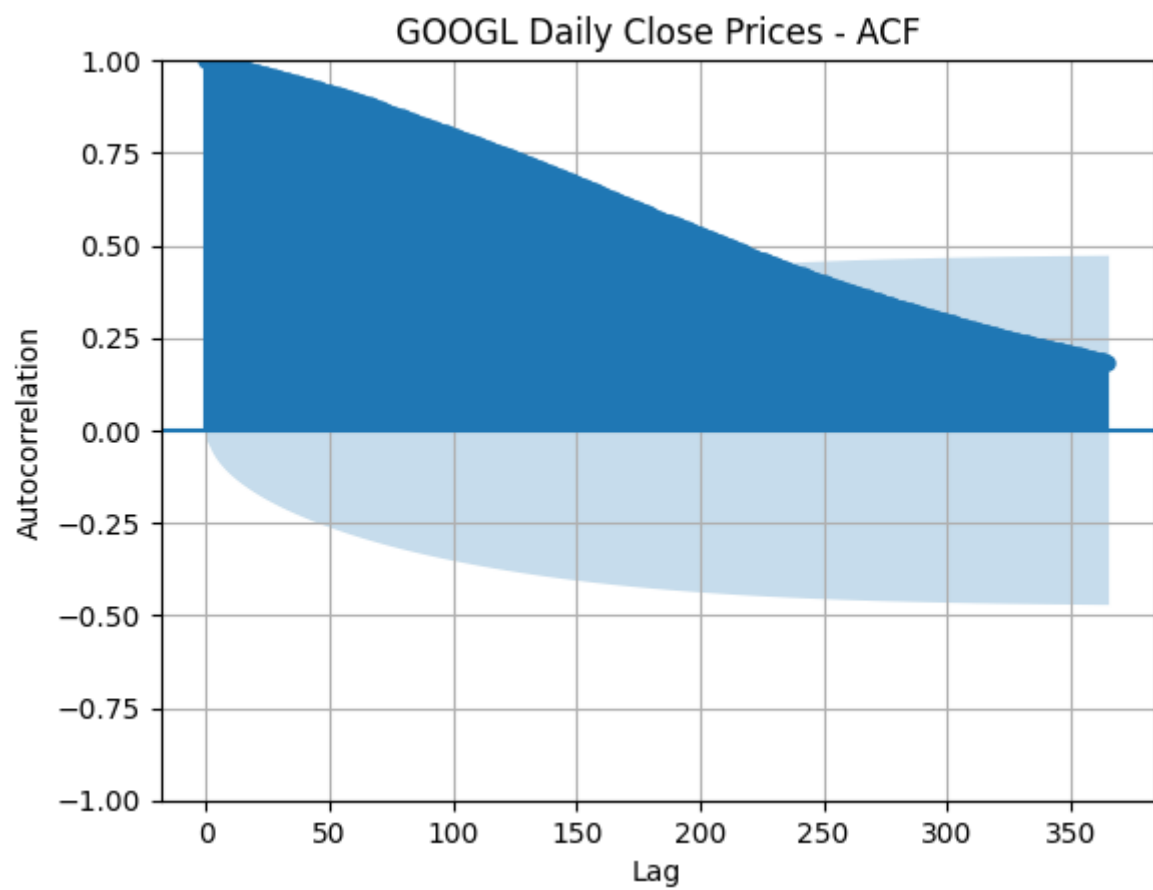
4. Τέλος η γραφική Residual αποτυπώνει τις τυχαίες διακυμάνσεις μετά την αφαίρεση της τάσης και της εποχικότητας. Μεγάλη αύξηση στις αποκλίσεις συμβαίνει κατά την περίοδο του 2022 που υπήρχε η μεγάλη πτώση, τα προηγούμενα χρόνια τα residuals είναι σχετικά σταθερά.



Ακόμη, δημιουργήθηκε η συνάρτηση αυτοσυσχέτισης (**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.



3. Linear Regression

3.1 Training-Validation

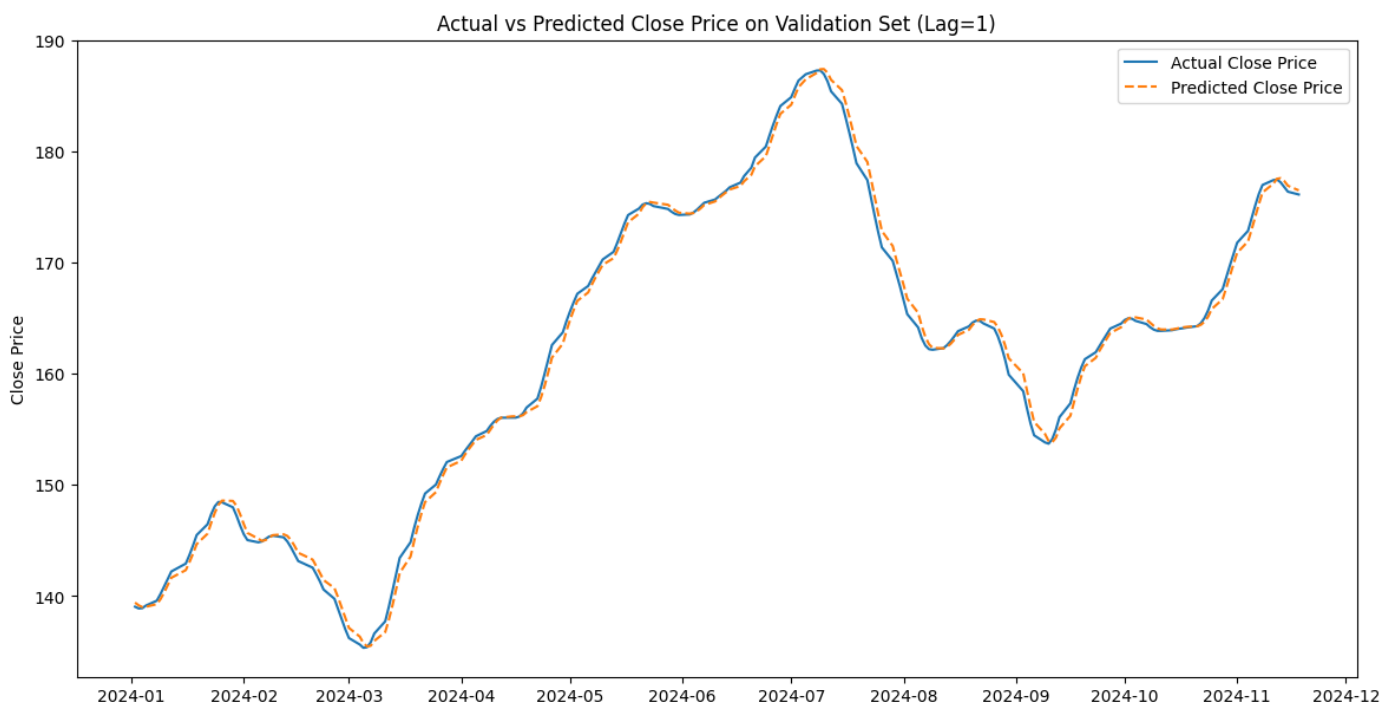
Αρχικά, εφαρμόζεται φιλτράρισμα στα δεδομένα χρησιμοποιώντας το gaussian filter με $\sigma=3$ και δημιουργείται στο DataFrame μια στήλη `closed_smoothed` που περιέχει τις τιμές που θα χρησιμοποιηθούν. Στην συνέχεια, στο DataFrame δημιουργούνται και στήλες που θα περιέχουν τα lagged features δηλαδή τις παρελθοντικές τιμές. Προκειμένου να βρεθεί το σωστό πλήθος lags και με βάση τις παρατηρήσεις που προέκυψαν από την προεπεξεργασία (για μικρό πλήθος lags υπάρχει αυτοσυσχέτιση μεταξύ των δεδομένων), 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 `X_train`, `X_validation` είναι τα lags και το `y_train`, `y_validation` είναι οι στόχοι, δηλαδή οι τιμές κλεισίματος της επόμενης μέρας.

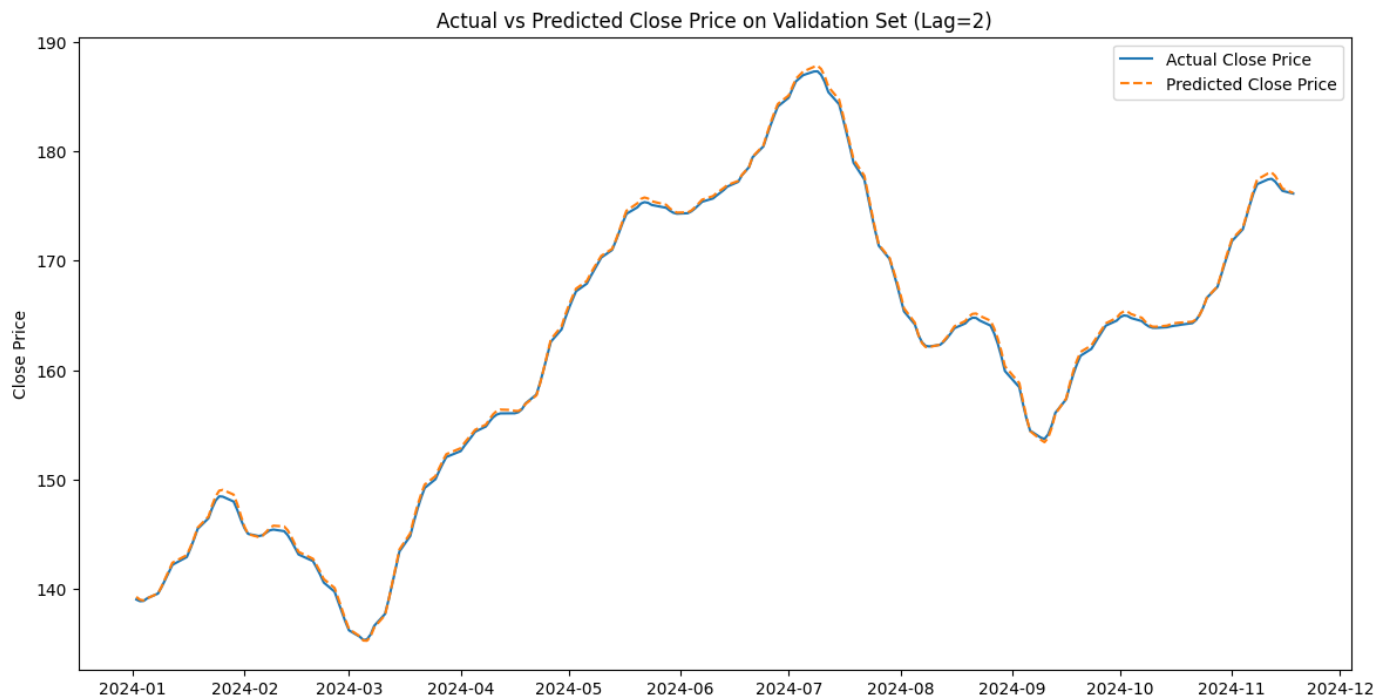
Υστερα, δημιουργείται το μοντέλο γραμμικής παλινδρόμησης με την συνάρτηση **LinearRegression**, εκπαιδεύεται στα δεδομένα εκπαίδευσης με την μέθοδο **fit()** και πραγματοποιούνται προβλέψεις στα δεδομένα επικύρωσης με την μέθοδο **predict()**. Το μοντέλο αξιολογείται χρησιμοποιώντας τις μετρικές Μέσο 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.

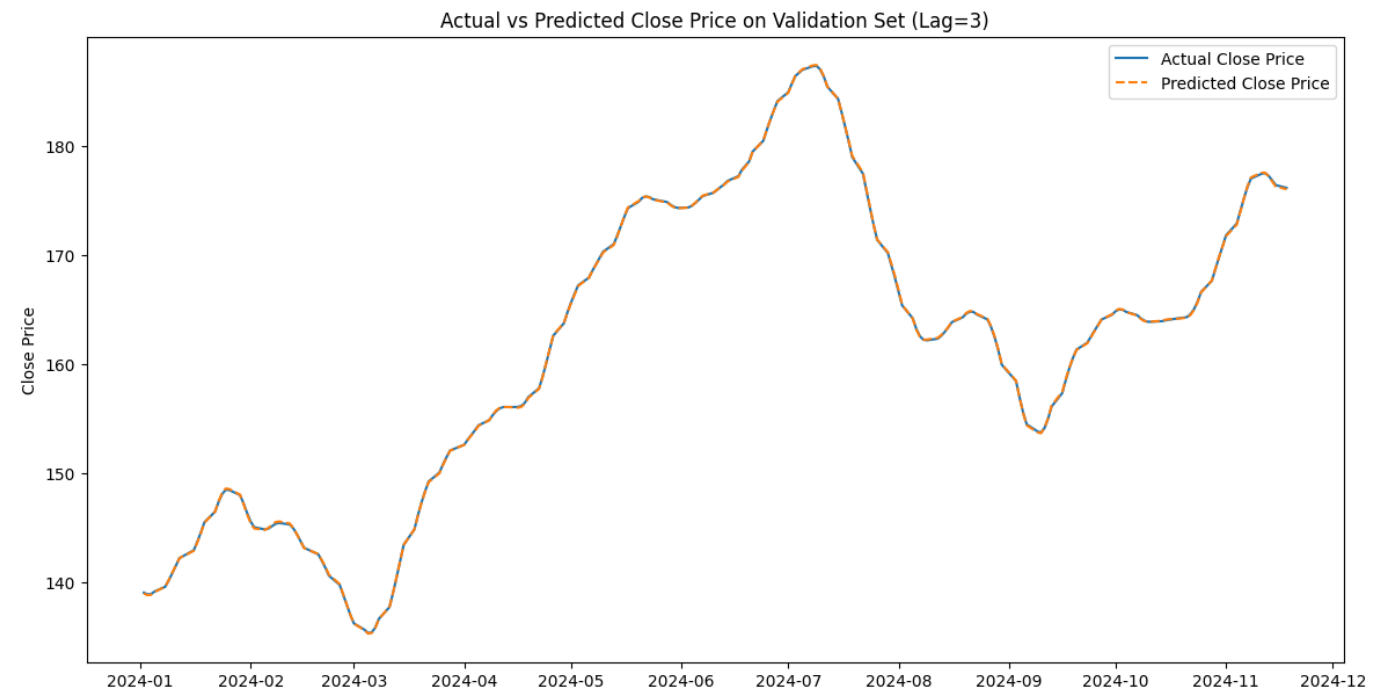
```
--- Lags: 1 ---  
Validation MSE: 0.5666315131761408  
Validation RMSE: 0.752749303006081  
Validation MAE: 0.6214900491711689  
Linear Regression Equation:  
y = 0.1503 + (0.9998) * close_t-1  
Predicted vs Actual Close Prices:
```



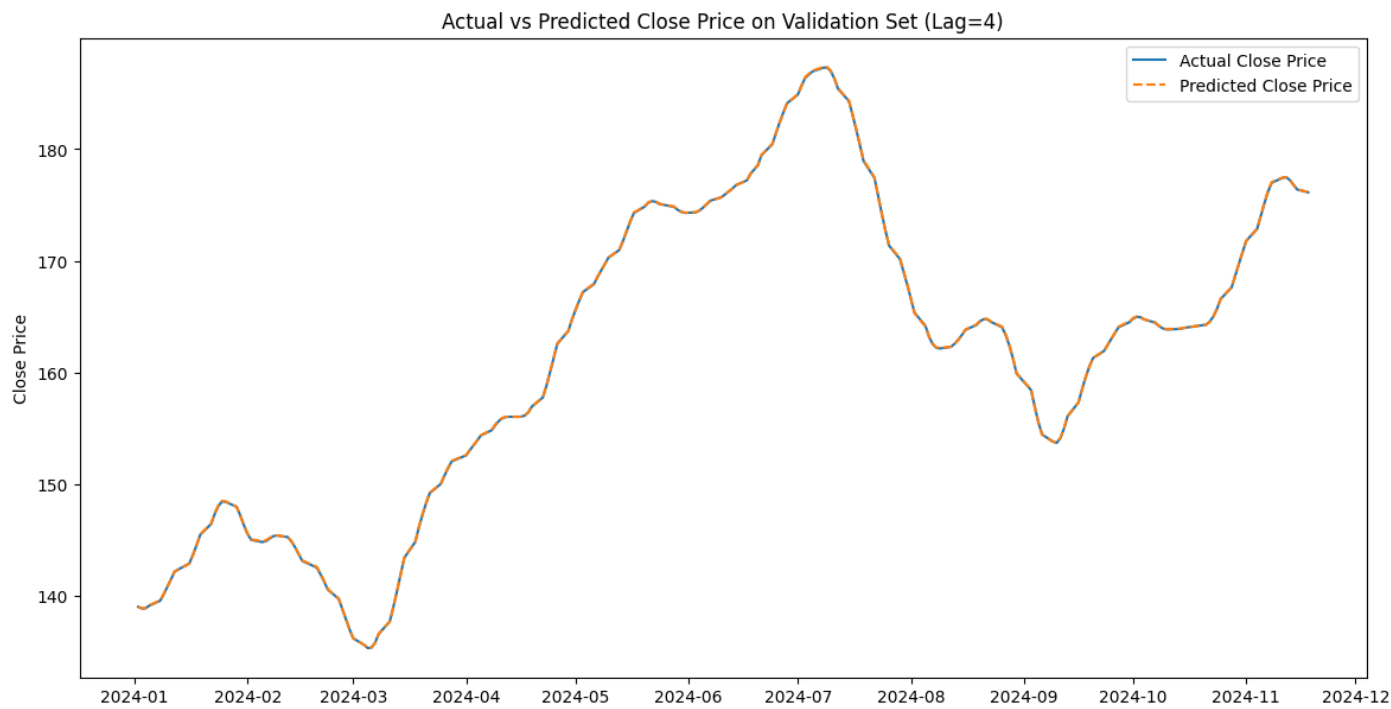
--- Lags: 2 ---
Validation MSE: 0.07001488312958021
Validation RMSE: 0.2646032560827251
Validation MAE: 0.22767450429252373
Linear Regression Equation:
 $y = 0.2630 + (1.9718) * \text{close_t-1} + (-0.9721) * \text{close_t-2}$
Predicted vs Actual Close Prices:



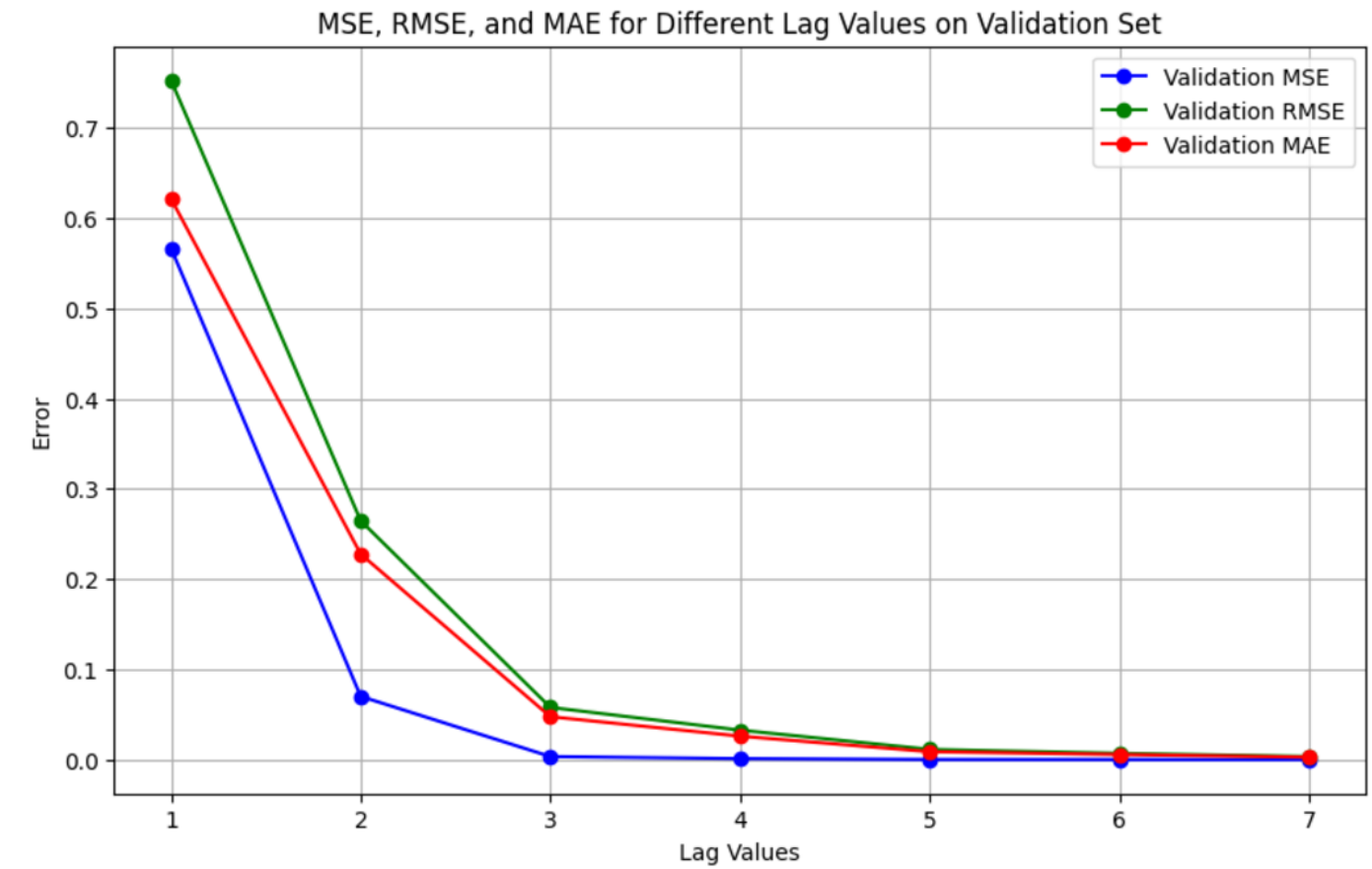
--- Lags: 3 ---
Validation MSE: 0.0033390303557124463
Validation RMSE: 0.057784343517188516
Validation MAE: 0.047518220172827925
Linear Regression Equation:
 $y = 0.0149 + (2.8924) * \text{close_t-1} + (-2.8394) * \text{close_t-2} + (0.9470) * \text{close_t-3}$
Predicted vs Actual Close Prices:



```
--- Lags: 4 ---
Validation MSE: 0.0010647297112459037
Validation RMSE: 0.03263019631025691
Validation MAE: 0.025952780376092346
Linear Regression Equation:
y = 0.0270 + (3.7691) * close_t-1 + (-5.4679) * close_t-2 + (3.6246) * close_t-3 + (-0.9258) * close_t-4
Predicted vs Actual Close Prices:
```



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



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 τιμές αν όχι ίδιες.

Μετά από 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 συνάρτηση **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:

Predicted Close Price for the next day: 177.54

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

11/19/2024

\$178.12

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.

4. Lasso Regression (L1 Κανονικοποίηση)

4.1 Training-Validation

Ακολουθείται η ίδια διαδικασία με το προηγούμενο μοντέλο, εφαρμόζεται φιλτράρισμα στα δεδομένα και διασπάται το DataFrame σε σύνολο εκπαίδευσης (train set) και σύνολο επικύρωσης (validation set).

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 πολυωνμικά χαρακτηριστικά, αυτό γίνεται δημιουργώντας ένα αντικείμενο της κλάσης **PolynomialFeatures**, το οποίο θα χρησιμοποιηθεί για να δημιουργήσει νέα χαρακτηριστικά που αντιστοιχούν στους πολυωνμικούς 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 πολυωνμικά 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_grid** 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 στην 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, δημιουργείται το μοντέλο με την συνάρτηση **Lasso** δίνοντας σαν παράμετρο την βέλτιστη τιμή **alpha** και τον αριθμό των επαναλήψεων **max_iter=1000**, εκπαιδεύεται στα δεδομένα εκπαίδευσης με την μέθοδο **fit()** και πραγματοποιούνται προβλέψεις στα δεδομένα επικύρωσης με την μέθοδο **predict()**. Το 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.

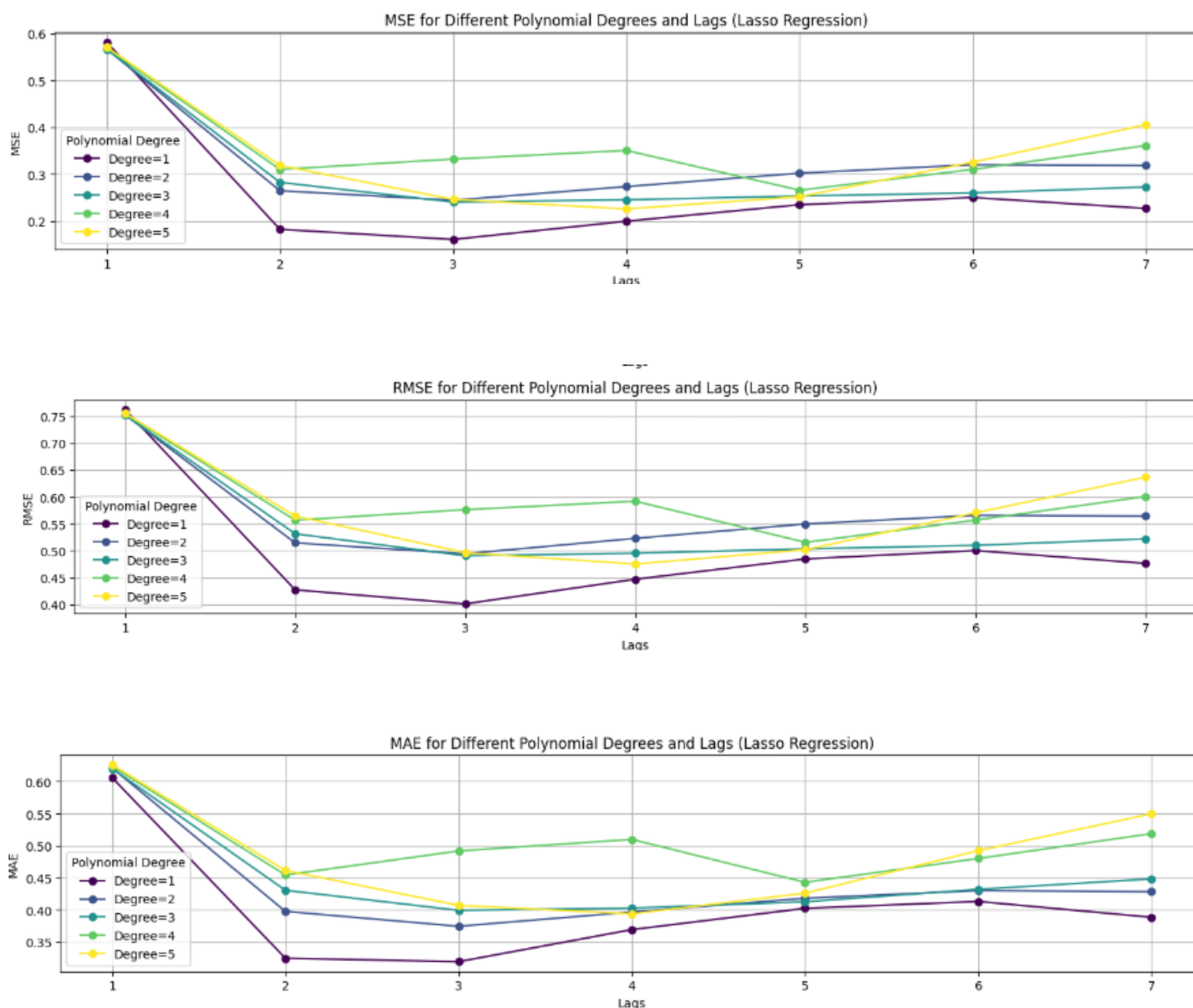
```
--- Lags: 1, Degree: 1, Best Alpha: 100 ---  
Validation MSE: 0.5811756228348759  
Validation RMSE: 0.7623487540718329  
Validation MAE: 0.6061288538631436
```

```
--- Lags: 3, Degree: 3, Best Alpha: 1 ---  
Validation MSE: 0.24059501647540682  
Validation RMSE: 0.4905048587683988  
Validation MAE: 0.399072887112303
```

--- Lags: 4, Degree: 4, Best Alpha: 10 ---
Validation MSE: 0.35036032572214526
Validation RMSE: 0.5919124307886643
Validation MAE: 0.5096877696508652

--- Lags: 6, Degree: 2, Best Alpha: 0.01 ---
Validation MSE: 0.32013150125418266
Validation RMSE: 0.565801644796286
Validation MAE: 0.42994595940013103

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.



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 overfitting. 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. Το σφάλμα (mse) σε αυτήν την περίπτωση είναι **Validation MSE: 0.24059501647540682**.

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 **degree=3**, **alpha=1** που αποθηκεύτηκε θα προβλεφθεί με την συνάρτηση **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:

Prediction for the next day: 177.07874003730063

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

11/19/2024	\$178.12
------------	----------

Το μοντέλο έχει μεγαλύτερη διαφορά με την πραγματική τιμή και το MSE είναι 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.

5. Ridge Regression (L2 Κανονικοποίηση)

5.1 Training-Validation

Ακολουθείται η ίδια διαδικασία με τα προηγούμενα μοντέλα, εφαρμόζεται φιλτράρισμα στα δεδομένα και διασπάται DataFrame σε σύνολο εκπαίδευσης (train set) και σύνολο επικύρωσης (validation set).

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 lasso. Όσον αφορά το cross validation πραγματοποιούνται πάλι τα ίδια βήματα με 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**, εκπαιδεύεται στα δεδομένα εκπαίδευσης με την μέθοδο **fit()** και πραγματοποιούνται προβλέψεις στα δεδομένα επικύρωσης με την μέθοδο **predict()**. Το 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 βαθμό 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.

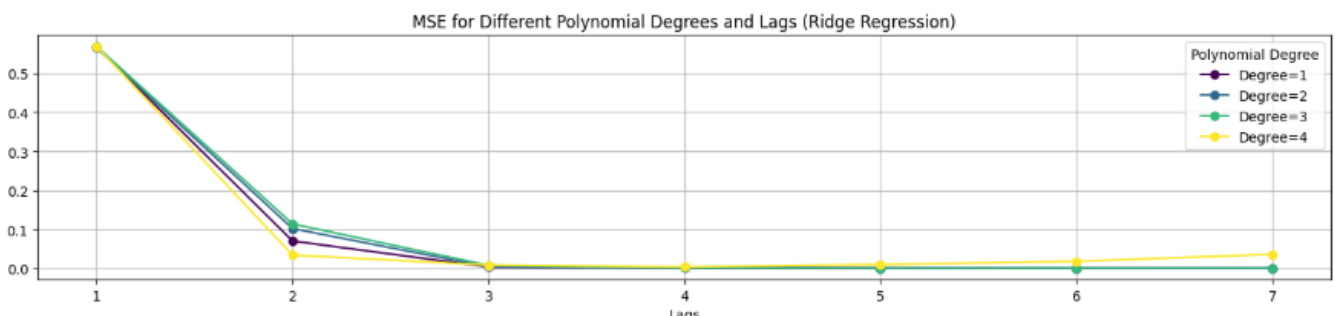
```
--- Lags: 1, Degree: 1, Best Alpha: 100 ---  
Validation MSE: 0.5666283199034802  
Validation RMSE: 0.7527471819299493  
Validation MAE: 0.6214844799800088
```

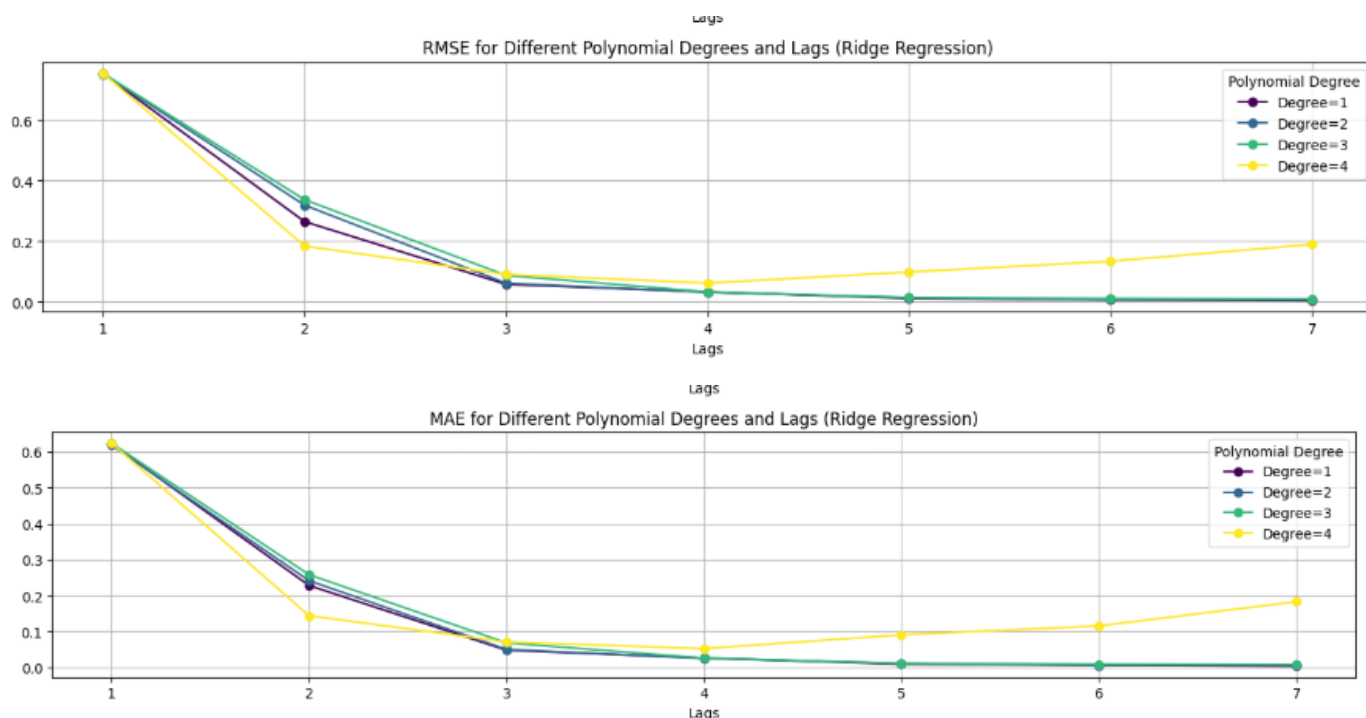
```
--- Lags: 4, Degree: 3, Best Alpha: 0.01 ---  
Validation MSE: 0.0010584331604625468  
Validation RMSE: 0.03253356974668699  
Validation MAE: 0.025430962809973954
```

```
--- Lags: 3, Degree: 3, Best Alpha: 10 ---  
Validation MSE: 0.007605802575855658  
Validation RMSE: 0.08721125257588988  
Validation MAE: 0.06799081280409595
```

```
--- Lags: 6, Degree: 4, Best Alpha: 1 ---  
Validation MSE: 0.017891510027931726  
Validation RMSE: 0.13375914932419286  
Validation MAE: 0.11512565106788683
```

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.





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 κάνει overfitting, καθώς εμφανίζει μεγαλύτερα σφάλματα για περισσότερα lags. Οπότε θα επιλέξουμε βαθμό πολυωνύμου 3. Το σφάλμα (mse) σε αυτήν την περίπτωση είναι πολύ μικρό και τείνει στο μηδέν **Validation MSE: 0.0010584331604625468**. Αυτό σημαίνει ότι το μοντέλο έχει εκπαιδευτεί σωστά και οι προβλέψεις που κάνει είναι πολύ κοντά στις πραγματικές τιμές αν όχι ίδιες.

Μετά από 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 **degree = 3** και **alpha = 0.01** που αποθηκεύτηκε θα προβλεφθεί με την συνάρτηση **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:

Prediction for the next day: 177.5717144706921

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

11/19/2024 \$178.12

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

6. 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, καταφέροντας 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 ικανοποιητικά 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.