1. Κατεβάσαμε τα δεδομένα από το Yahoo Finance για τα τελευταία 5 έτη.
2. Βλέποντας τα δεδομένα, παρατηρήσαμε ότι η στήλη Close καθώς και η στήλη Adj Close έχουν τα ίδια δεδομένα. Ακόμα μετατρέψαμε την στήλη Date σε datetime type για να μπορούμε να χρησιμοποιούμε τις ιδιότητες των ημερομηνιών.  
   ΑΛΛΑΓΗ
3. Για το Preprocessing:  
   Χρησιμοποιήσαμε τον MInMaxScaler για να κάνουμε το normalization. Ο MinMax   
   για κάθε τιμή στο dataset, μετατρέπει όλες τις τιμές σε ένα εύρος [0,1]. Ο μαθηματικός τύπος που χρησιμοποιεί είναι ο παρακάτω:  
   Text

   Description automatically generated with medium confidence  
   Για το Learning:   
   Στην Linear Regression, αρχικά χρησιμοποιήσαμε την train\_test\_split για να χωρίσουμε τα δεδομένα σε train και test. Η μεταβλητή X περιέχει τις στήλες Open,High,Low,Volume μετά το scaling και στην Y την στήλη Close μετά το scaling. Στη συνέχεια χρησιμοποιήσαμε το cross validation για να βρούμε το R2 Score του training καθώς και το αντίστοιχο score για το testing. Μετέπειτα, χρησιμοποιήσαμε τον KFold για 10 φορές και βρήκαμε το R2 Score καθώς και το Mean Squared Error για κάθε επανάληψη.   
   ΣΥΝΕΧΙΣΕ

Για την Logistic Regression, αρχικά η μεταβλητή X περιέχει τις στήλες Open,High,Low,Volume και Close μετά το scaling και η Y την στήλη Up/Down. Η στήλη αυτή έχει 1 όταν η τιμή Open είναι μεγαλύτερη από την Close και 1 αλλιώς. Χρησιμοποιούμε αυτή τη στήλη γιατί θέλουμε να κάνουμε classification, κάτι που δεν γίνεται με την Linear Regression. Όμοια με πριν, χρησιμοποιήσαμε την train\_test\_split για να χωρίσουμε τα δεδομένα σε train και test. Σε αυτή την περίπτωση ψάχνουμε το ROC AUC score για το training και το testing για αυτό χρησιμοποιούμε το cross validation. Τέλος, βρίσκουμε το Accuracy score καθώς και το ROC AUC score.

1. Νευρωνικό Δίκτυο
2. Bonus – ARIMA  
     
   Το μοντέλο αυτό αντί να χρησιμοποιεί προηγούμενες τιμές της μεταβλητής πρόβλεψης σε μια παλινδρόμηση, χρησιμοποιεί προηγούμενα σφάλματα πρόβλεψης σε ένα μοντέλο που μοιάζει με παλινδρόμηση.   
   Για να χρησιμοποιήσουμε το ARIMA πρέπει τα δεδομένα μας να είναι σταθερά. Για αυτό τον λόγο χρησιμοποιήσαμε τα γνώσεις μας στην Στατιστική και εφαρμόσαμε τον κανόνα την Null Hypothesis. Σύμφωνα με αυτόν τον κανόνα, η Null Hypothesis γίνεται δεκτή μόνο όταν το pvalue είναι μεγαλύτερο από 0,05. Φτιάξαμε μία συνάρτηση που μας δίνει το pvaule. Οπότε αν θεωρήσουμε ότι η Null Hypothesis είναι ότι τα δεδομένα μας είναι σταθερά, η υπόθεση αυτή πρέπει να απορριφθεί. Η επόμενη επιλογή ήταν να χρησιμοποιήσαμε λογαρίθμους. Ακολουθούμε την ίδια διαδικασία αλλά το pvalue παραμένει υψηλό. Για αυτό τον λόγο, αφαιρέσαμε από τους λογαρίθμους ΘΑ ΤΟ ΒΡΩ. Τώρα το pvalue είναι μικρότερο από το 0.05, οπότε η Null Hypothesis είναι αποδεκτή και τα δεδομένα μας είναι σταθερά.   
   Στη συνέχεια, χωρίσαμε τα δεδομένα μας σε train και test σε αναλογία 9:1. Μετέπειτα φτιάξαμε το μοντέλο μας και για κάθε τιμή του testing dataset, εμφανίζουμε την τιμή που είναι σωστή, αυτή που προέβλεψε το μοντέλο μας καθώς και το error. Το τελευταίο υπολογίζεται από την διαφορά της πραγματικής τιμής από την τιμή που βρήκε το μοντέλο διαιρεμένο με την πραγματική τιμή.  
   Το ARIMA παίρνει τις τιμές p,d,q. Το p δείχνει τον αριθμό των autoregressive terms,το d τον αριθμό των nonseasonal differences που χρειάζονται για την σταθερότητα και το q είναι ο αριθμός των lagged forecast errors. Η εκφώνηση ζητούσε το πρώτο q να είναι 1 και το δεύτερο 0. Εμείς επιλέξαμε για τα p,d τα 4 και 2 αντίστοιχα καθώς είδαμε πως με αυτές τις τιμές είχαμε τα καλύτερα αποτελέσματα στο διάγραμμα. Στη συνέχεια κάνουμε fit και χρησιμοποιούμε την συνάρτηση forecast(), η οποία χρησιμοποιεί προηγούμενες τιμές για να προβλέψει τις επόμενες.