ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΙΑΤΡΙΚΗ

Πληροφορική και Τεχνολογίες Πληροφορίας και Επικοινωνιών (Τ.Π.Ε.) στην Εκπαίδευση

Ανάλυση Μεγάλων Δεδομένων με τη χρήση εργαλείων της Python (Big Data analysis using Python tools)

ΣΠΑΝΤΩΝΗΣ ΑΝΑΣΤΑΣΙΟΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επιβλέπων

Τασουλής Σωτήριος

Λαμία, 2023

«Υπεύθυνη Δήλωση μη λογοκλοπής και ανάληψης προσωπικής ευθύνης»

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, και γνωρίζοντας τις συνέπειες της λογοκλοπής, δηλώνω υπεύθυνα και ενυπογράφως ότι η παρούσα εργασία με τίτλο ***«Ανάλυση Μεγάλων Δεδομένων με τη χρήση εργαλείων της Python – Big Data analysis using Python tools»*** αποτελεί προϊόν αυστηρά προσωπικής εργασίας και όλες οι πηγές από τις οποίες χρησιμοποίησα δεδομένα, ιδέες, φράσεις, προτάσεις ή λέξεις, είτε επακριβώς (όπως υπάρχουν στο πρωτότυπο ή μεταφρασμένες) είτε με παράφραση, έχουν δηλωθεί κατάλληλα και ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο ΔΗΛΩΝ

ΣΠΑΝΤΩΝΗΣ ΑΝΑΣΤΑΣΙΟΣ

Λαμία …….-2023

Τριμελής Επιτροπή:

Τασουλής Σωτήριος

……

……

Επιστημονικός Σύμβουλος:

Τασουλής Σωτήριος

ΠΕΡΙΛΗΨΗ

Κάθε δευτερόλεπτο παράγονται, αποθηκεύονται και αναλύονται τεράστιες ποσότητες δεδομένων που αυξάνονται εκθετικά. Τα ψηφιακά δεδομένα που παράγονται είναι εν μέρει αποτέλεσμα της χρήσης συνδεδεμένων συσκευών στο Διαδίκτυο. Έτσι, τα smartphones, τα tablets, οι αισθητήρες, οι υπολογιστές και πληθώρα άλλων ηλεκτρονικών συσκευών μεταδίδουν δεδομένα. Τα συνδεδεμένα έξυπνα αντικείμενα μεταφέρουν πληροφορίες σχετικά με τη χρήση καθημερινών αντικειμένων από τους ανθρώπους. Εκτός από τις συνδεδεμένες συσκευές, τα δεδομένα προέρχονται από ένα ευρύ φάσμα πηγών: δημογραφικά δεδομένα, κλιματικά δεδομένα, επιστημονικά και ιατρικά δεδομένα, δεδομένα κατανάλωσης ενέργειας κ.λπ. Όλα αυτά τα δεδομένα παρέχουν πληροφορίες σχετικά με την τοποθεσία των χρηστών των συσκευών, τις μετακινήσεις τους, τα ενδιαφέροντά τους, τις καταναλωτικές τους συνήθειες, τις δραστηριότητες αναψυχής τους, τα έργα τους κ.ο.κ. Αλλά και πληροφορίες σχετικά με τον τρόπο χρήσης των υποδομών, των μηχανημάτων και των συσκευών. Με τον ολοένα αυξανόμενο αριθμό χρηστών του Διαδικτύου και των κινητών τηλεφώνων, ο όγκος των ψηφιακών δεδομένων αυξάνεται ραγδαία. Για αυτό το λόγο έχει κάνει την εμφάνισή του ο όρος Μεγάλα Δεδομένα (Big Data).

Τα δεδομένα έχουν πολλές μορφές και μια διάσταση που πρέπει να εξετάσουμε είναι η δομή τους. Όσο πιο δομημένο είναι ένα σύνολο δεδομένων τόσο πιο εύχρηστο θα είναι στη μηχανική επεξεργασία του. Η ανάλυση μεγάλων δεδομένων είναι η υποπεριοχή των μεγάλων δεδομένων που ασχολείται με την προσθήκη δομής στα δεδομένα για την υποστήριξη της λήψης αποφάσεων, καθώς και με την υποστήριξη σεναρίων χρήσης συγκεκριμένων τομέων.

Η ανάλυση δεδομένων είναι ένας ευρύς όρος που περιλαμβάνει πολλά διαφορετικά είδη εξέτασης πληροφοριών. Οποιοδήποτε είδος δεδομένων μπορεί να εκτεθεί σε μεθόδους εξέτασης πληροφοριών για να αποκτηθεί η γνώση που μπορεί να χρησιμοποιηθεί για τη βελτίωση των πραγμάτων καθώς και για την λήψη σωστών αποφάσεων.

Στην παρούσα μεταπτυχιακή διατριβή αναλύω και περιγράφω τα εξής:

Στο Κεφάλαιο1 περιγράφω τι είναι τα μεγάλα δεδομένα, ποια τα χαρακτηριστικά τους και ποια τα οφέλη και οι εφαρμογές.

Στο Κεφάλαιο 2 αναλύω και περιγράφω τα βήματα και τις μεθόδους που ακολουθούνται για να γίνει η ανάλυση των δεδομένων.

Στο Κεφάλαιο 3 αναφέρονται τα εργαλεία που υπάρχουν για την ανάλυση των μεγάλων δεδομένων.

Στο Κεφάλαιο 4 περιγράφω τα βασικά χαρακτηριστικά καθώς και πλεονεκτήματα που έχει η γλώσσα Python ως εργαλείο ανάλυσης δεδομένων.

Στο Κεφάλαιο 5 παρουσιάζω αναλυτικά και βήμα προς βήμα μια μελέτη περίπτωσης χρήσης και εφαρμογής της Pyhton με τη βοήθεια του Jupyter Notebook όπου γίνεται ανάλυση δεδομένων από μετοχές του Χρηματιστηρίου Αξιών Αθηνών.

Λέξεις κλειδιά: Μεγάλα Δεδομένα, Ανάλυση Δεδομένων, Pyhton

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα αρχικά να ευχαριστήσω τον επιβλέποντα επίκουρο καθηγητή Δρ. Τασουλή Σωτήριο για την δυνατότητα την οποία μου έδωσε να ασχοληθώ με ένα τόσο σύγχρονο, ενδιαφέρον και σημαντικό θέμα όπως την Ανάλυση Μεγάλων Δεδομένων με τη χρήση της Python.

Επίσης ευχαριστώ θερμά τους δικούς μου ανθρώπους, την οικογένεια μου και τους φίλους μου, των οποίων η προσφορά όλα αυτά τα χρόνια είναι πιθανότατα και η σημαντικότερη όλων.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1

Τι είναι τα Big Data (Μεγάλα Δεδομένα)

1.1 Η Εξέλιξη των Δεδομένων

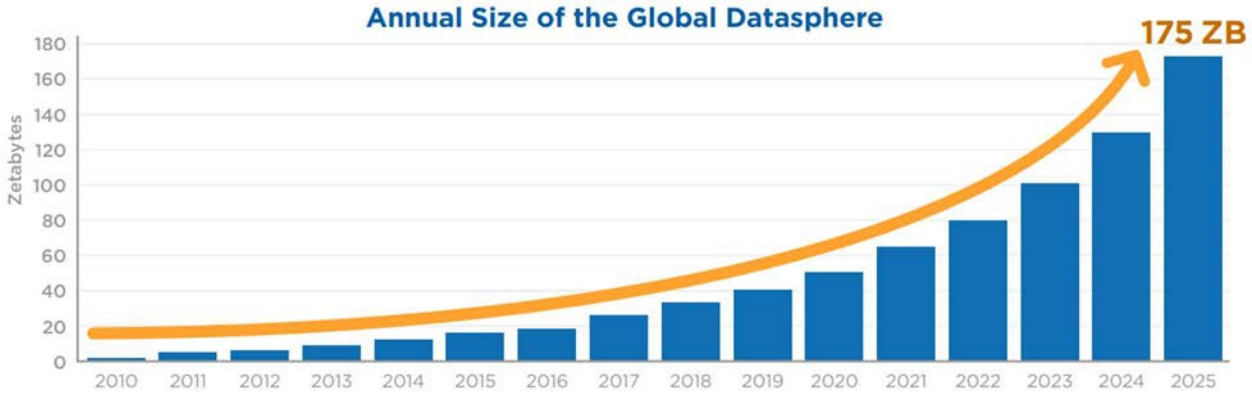
Για να κατανοήσουμε καλύτερα τι είναι τα Μεγάλα Δεδομένα και από πού προέρχονται, είναι ζωτικής σημασίας να κατανοήσουμε πρώτα κάποια παλαιότερη ιστορία της αποθήκευσης δεδομένων, των αποθετηρίων και των εργαλείων διαχείρισής τους. Όπως φαίνεται στο Σχήμα 1, υπήρξε τεράστια αύξηση του όγκου των δεδομένων κατά την τελευταία δεκαετία.

Τη δεκαετία του 1990 ο όγκος των δεδομένων μετριόταν σε terabytes. Οι βάσεις δεδομένων συσχετίσεων και οι αποθήκες δεδομένων που αναπαριστούσαν δομημένα δεδομένα σε γραμμές και στήλες ήταν οι τυπικές τεχνολογίες για την αποθήκευση και τη διαχείριση επιχειρηματικών πληροφοριών.

Τα δεδομένα της επόμενης δεκαετίας άρχισαν να ασχολούνται με διαφορετικά είδη πηγών δεδομένων που καθοδηγούνται από εργαλεία παραγωγικότητας και δημοσίευσης, όπως τα αποθετήρια που διαχειρίζονται περιεχόμενο και τα δικτυακά συστήματα αποθήκευσης. Κατά συνέπεια, ο όγκος των δεδομένων άρχισε να μετράται σε petabytes.

Η δεκαετία του 2010 είχε να αντιμετωπίσει τον εκθετικό όγκο δεδομένων που οφείλεται στην ποικιλία και τις πολλές πηγές ψηφιοποιημένων δεδομένων. Σχεδόν όλοι και όλα αφήνουν ένα ψηφιακό αποτύπωμα.

Σύμφωνα με μια έκθεση που κυκλοφόρησε από την IDC το 2018 και χρηματοδοτείται από τη Seagate, η κύρια πηγή δεδομένων θα μετατοπιστεί από την ψυχαγωγία στα δεδομένα "παραγωγικότητας" (στα οποία περιλαμβάνονται τα δεδομένα μάρκετινγκ) [1]. Σημαντική αύξηση θα σημειωθεί επίσης στα "ενσωματωμένα" δεδομένα, τα περισσότερα από τα οποία προέρχονται από συσκευές IoT.



Σχήμα 1-1. Η εξέλιξη των δεδομένων και η άνοδος των πηγών μεγάλων δεδομένων [1]

1.2 Πηγές δεδομένων

Οι πηγές αύξησης του όγκου των δεδομένων μπορούν να χωριστούν σε 3 κατηγορίες παραγωγής δεδομένων:

- Μηχανή,
- Ανθρώπινη αλληλεπίδραση, και
- Επεξεργασία δεδομένων.

Συνήθως, ο πρώτος τύπος συνδέεται με την εξάπλωση της ψηφιοποίησης των μηχανημάτων, η οποία σχετίζεται με την ενσωμάτωση αισθητήρων, την αύξηση της συνδεσιμότητας και των συσκευών που καταγράφουν ήχους, εικόνες ή βίντεο και με την επικοινωνία των μηχανημάτων μεταξύ τους. Ειδικότερα, συσκευές όπως κάμερες που καταγράφουν βίντεο, κινητά τηλέφωνα που συλλέγουν γεωχωρικά δεδομένα, μηχανές σε γραμμές παραγωγής βιομηχανικών συστημάτων, ανταλλάσσουν σημαντικές πληροφορίες κατά την επεξεργασία των δραστηριοτήτων τους. Περισσότερα παραδείγματα αναφέρονται παρακάτω:

- Ιατρικές πληροφορίες - μηχανήματα που καταγράφουν EEG (ηλεκτροεγκεφαλογραφία), καρδιακούς παλμούς, γονιδιωματική αλληλουχία,
- Πολυμέσα - φωτογραφίες και βίντεο που αναρτώνται στο Διαδίκτυο,
- Κινητές συσκευές - παρέχουν γεωχωρικά δεδομένα (θέση, γυροσκόπιο), καθώς και μεταδεδομένα σχετικά με τηλεφωνικές κλήσεις, μηνύματα, χρήση του διαδικτύου και δεδομένα που συλλέγονται από εφαρμογές κινητής τηλεφωνίας,
- Άλλες συσκευές - Συστήματα διαχείρισης αποθηκών (WMS) που παρέχουν εσωτερική θέση μέσω Wi-Fi, αναγνώριση των αποθηκευμένων προϊόντων ή υλικών μέσω BAR, κωδικών QR (Quick Response) ή τσιπ RFID (Radio- Frequency Identification). Υπάρχει παρουσία πολλών άλλων τεχνολογιών, όπως συστήματα πλοήγησης, σεισμική επεξεργασία κ.λπ.

Είναι σκόπιμο να εξεταστούν και άλλες πηγές που δημιουργούνται από δραστηριότητες στο Διαδίκτυο γενικά. Ας φανταστούμε ότι έχουμε ένα σύνολο διακομιστών που εκτελούν ιστοσελίδες που μπορούν να προσδιοριστούν για επιχειρήσεις λιανικής πώλησης. Οι διακομιστές μπορούν να συλλέγουν αρχεία όλων των δραστηριοτήτων των πελατών των ιστοσελίδων, των χρηστών, των συναλλαγών, των εφαρμογών και της ίδιας της δραστηριότητας και συμπεριφοράς των διακομιστών. Για παράδειγμα, υπάρχουν αρχεία καταγραφής τα οποία μπορούν να συλλεχθούν από [2]:

- Εφαρμογές - δραστηριότητες των χρηστών και απόδοση των εφαρμογών,
- Επιχειρηματικές διαδικασίες - αλλαγές λογαριασμών, αγορές και αναφορές προβλημάτων,
- Δεδομένα ροής κλικ - για την αποθήκευση των ενδιαφερόντων των πελατών,
- Αρχεία ρυθμίσεων - αποθηκευμένες ρυθμίσεις διακομιστών και εφαρμογών,
- Αρχεία καταγραφής βάσης δεδομένων - για να δείξετε ποιος έκανε αλλαγές στη βάση δεδομένων.

Προκειμένου να γίνει πιο συνειδητή η παραγωγή δεδομένων, είναι σκόπιμο να αναφέρουμε ένα άλλο παράδειγμα που σχετίζεται με μια συλλογή αστρονομικών δεδομένων. Το έτος 2000, ξεκίνησε ένα έργο που αφορά τη χαρτογράφηση του ουρανού, με την ονομασία Sloan Digital Sky Survey (SDSS). Κατά τη διάρκεια των πρώτων εβδομάδων του έργου, συλλέχθηκε από ένα τηλεσκόπιο ένας εκτεταμένος όγκος δεδομένων- περισσότερα δεδομένα από όσα έχουν συλλεχθεί ποτέ σε όλη την ιστορία της αστρονομίας.

Η επόμενη κατηγορία αφορά την ανταλλαγή πληροφοριών μεταξύ των ανθρώπων. Για παράδειγμα, μερικά παραδείγματα κοινωνικών δικτύων μπορεί να είναι το Facebook, το Twitter, το LinkedIn κ.λπ. Κάθε δευτερόλεπτο αυτά τα συστήματα παράγουν έναν τεράστιο όγκο δεδομένων που μοιράζονται εκατομμύρια άνθρωποι".

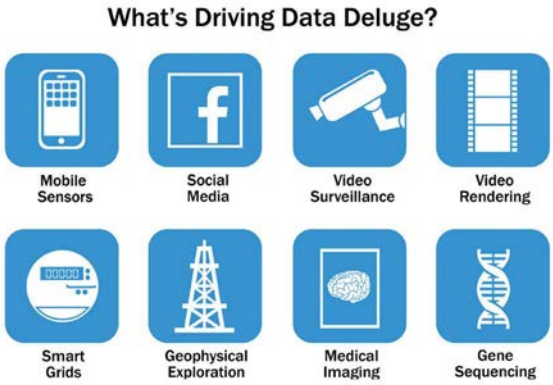
Η τριμηνιαία αύξηση της κίνησης δεδομένων σε δίκτυα κινητής τηλεφωνίας μεταξύ του τέταρτου τριμήνου του 2022 και του πρώτου τριμήνου του 2023 ήταν περίπου 7%. Η συνολική μηνιαία παγκόσμια κίνηση δεδομένων κινητών δικτύων έφτασε τα 126 EB. Σε απόλυτους αριθμούς, αυτό

σημαίνει ότι η κυκλοφορία δεδομένων κινητών δικτύων έχει σχεδόν διπλασιαστεί μέσα σε μόλις 2 χρόνια, από 66 EB ανά μήνα το 1ο τρίμηνο του 2021. Η μακροπρόθεσμη αύξηση της κίνησης1 οφείλεται τόσο στην αύξηση των συνδρομών smartphone όσο και στην αύξηση του μέσου όγκου δεδομένων ανά συνδρομή, που τροφοδοτείται κυρίως από την αυξημένη προβολή περιεχομένου βίντεο. Στο Σχήμα 18 παρουσιάζεται η καθαρή προσθήκη και η συνολική παγκόσμια μηνιαία κίνηση δεδομένων δικτύου από το 1ο τρίμηνο του 2016 έως το 1ο τρίμηνο του 2023, καθώς και η ετήσια ποσοστιαία αύξηση της κίνησης δεδομένων δικτύου κινητής τηλεφωνίας [3].

Σχήμα 1-2: Παγκόσμια κίνηση δεδομένων σε δίκτυα κινητής τηλεφωνίας και ετήσια αύξηση (EB ανά μήνα) [3]

Η επεξεργασία δεδομένων αφορά την επεξεργασία ακατέργαστων ή ήδη επεξεργασμένων δεδομένων για την επίτευξη μιας εξόδου ή για να φτάσουμε σε μια άλλη φάση διαδικασίας που εμπλέκεται σε ένα συγκεκριμένο έργο. Για να δώσουμε μια εικόνα των πεδίων που εξετάζονται, ας δούμε την περίπτωση του κλάδου της υγειονομικής περίθαλψης. Για παράδειγμα, η ανάλυση των δεδομένων που παράγονται από έναν καταγραφέα EEG μπορεί να παράγει νέα δεδομένα. Ειδικότερα, τα δεδομένα EEG θεωρούνται ως είσοδος η οποία επεξεργάζεται από έναν αλγόριθμο που παράγει νέο όγκο δεδομένων που αποθηκεύεται για περαιτέρω ανάλυση.

Η παραγωγή, η διαθεσιμότητα και η παρουσία μιας μεγάλης ποσότητας δεδομένων γύρω μας περιγράφεται και ως "Data Deluge" (Κατακλυσμός δεδομένων) [4]. Το Σχήμα 1-3 υπογραμμίζει διάφορες πηγές του Big Data Deluge.



Σχήμα 1-3: Τι οδηγεί στον κατακλυσμό δεδομένων [4]

Συνοψίζοντας, υπάρχουν πολλές πηγές που παράγουν πολλά δεδομένα τα οποία μπορούν να αποθηκευτούν για περαιτέρω αναλύσεις και εξερευνήσεις, οι οποίες, θα μπορούσαν να οδηγήσουν στην επιθυμητή γνώση.

1.3 Μεγάλα Δεδομένα - Όροι, Ορισμοί και Εφαρμογές

1.3.1 Τι είναι τα μεγάλα δεδομένα;

Τα μεγάλα δεδομένα είναι ένας ευρύς όρος που χρησιμοποιείται για να αναφερθεί στον τεράστιο όγκο ψηφιακών πληροφοριών που παράγονται από διάφορες επιχειρήσεις και οργανισμούς. Αυτά τα μεγάλα δεδομένα δεν παράγονται μόνο από την παραδοσιακή ανταλλαγή πληροφοριών και το λογισμικό, αλλά και από αισθητήρες διαφόρων τύπων που είναι ενσωματωμένοι σε διάφορα περιβάλλοντα: νοσοκομεία, σταθμούς του μετρό, αγορές και σχεδόν κάθε ηλεκτρονική συσκευή που παράγει δεδομένα.

Τα μεγάλα δεδομένα δίνουν υπερβολική έμφαση στο ζήτημα του όγκου των πληροφοριών. Ξεπερνούν την ικανότητα των παραδοσιακών τεχνολογιών διαχείρισης δεδομένων, δημιουργώντας την ανάγκη για νέα εργαλεία και τεχνολογίες για τον χειρισμό του εξαιρετικά μεγάλου όγκου. Δεν αποτελεί πρόκληση μόνο η αποθήκευση μεγάλου όγκου δεδομένων, αλλά και οι νέες δυνατότητες ανάλυσης αυτού του τεράστιου όγκου δεδομένων.

Ένας άλλος τρόπος ορισμού των μεγάλων δεδομένων είναι τα σύνολα δεδομένων των οποίων το μέγεθος ή ο τύπος υπερβαίνει την ικανότητα των παραδοσιακών σχεσιακών βάσεων δεδομένων να καταγράφουν, να διαχειρίζονται και να επεξεργάζονται τα δεδομένα με χαμηλή καθυστέρηση. Τα μεγάλα δεδομένα προέρχονται από βάσεις δεδομένων, αισθητήρες, συσκευές, ήχο/βίντεο, δίκτυα, αρχεία καταγραφής, εφαρμογές συναλλαγών, διαδίκτυο και μέσα κοινωνικής δικτύωσης που παράγονται κυρίως σε πραγματικό χρόνο και σε πολύ μεγάλη κλίμακα. [5]

Ο όρος "Big Data" έχει δεκάδες ορισμούς. Αν και εστιάζεται σε μεγάλο βαθμό στον όγκο των δεδομένων (**Volume)**, υπάρχουν και άλλα **V** - δηλαδή η ποικιλία (Variety) και η ταχύτητα (**Velocity**)- που έρχονται στο προσκήνιο. Οι ηγέτες της πληροφορικής έχουν αρχίσει να συνειδητοποιούν ότι υπάρχουν περισσότερες από μία προκλήσεις και διαστάσεις για τα δεδομένα εκτός από τις νέες δομές. [6]

Στις επόμενες ενότητες, θα περιγράψουμε τις διάφορες νέες δομές δεδομένων, τα τρία V που αποτελούν τη βάση των μεγάλων δεδομένων και τις νέες διαστάσεις που έχουν προκύψει ως προκλήσεις κατά τη συζήτηση των μεγάλων δεδομένων.

1.3.2 Δομές δεδομένων

Πρέπει πρώτα να κατανοήσουμε τους νέους τύπους δομών δεδομένων. Παραδοσιακά, έχουμε επικεντρωθεί σε δομημένα και αδόμητα δεδομένα.

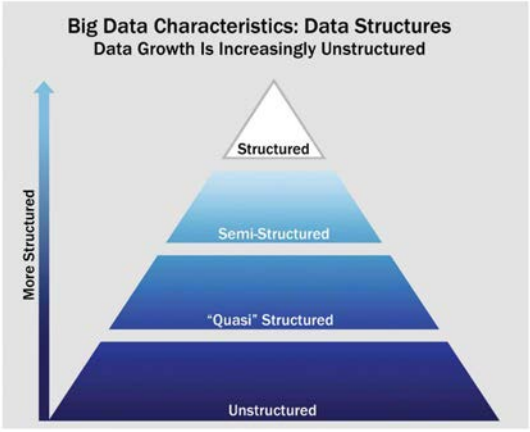
Τα δομημένα δεδομένα **(Structured data)** είναι αυτά που περιέχονται σε σχεσιακές βάσεις δεδομένων και λογιστικά φύλλα. Τα δομημένα δεδομένα συμμορφώνονται με ένα μοντέλο βάσης δεδομένων που έχει μια σταθερή δομή ή μορφή καταγραφής δεδομένων. Χρησιμοποιήθηκαν εργαλεία βάσεων δεδομένων και πρόσθετα εργαλεία αναφοράς και ανάλυσης για να βοηθήσουν στην ανάλυση αυτών των δεδομένων και στη δημιουργία ουσιαστικών αναφορών. [7]

Τα μη δομημένα δεδομένα **(Unstructured data)** δεν έχουν προκαθορισμένο μοντέλο δεδομένων ούτε είναι οργανωμένα με προκαθορισμένο τρόπο. Συνήθως είναι βαριά σε κείμενο και μπορεί να περιέχουν δεδομένα όπως ημερομηνίες, αριθμούς και γεγονότα και να περιλαμβάνουν δεδομένα χωρίς ετικέτες που αντιπροσωπεύουν φωτογραφίες και γραφικές εικόνες. Τα έγγραφα επεξεργασίας κειμένου, οι παρουσιάσεις και τα αρχεία PDF αποτελούν πρωταρχικά παραδείγματα μη δομημένων δεδομένων.

Οι νέες δομές δεδομένων που έχουν προκύψει είναι τα ημιδομημένα δεδομένα (semi-structured data) και τα οιονεί δομημένα δεδομένα (quasi-structured data)

Τα ημιδομημένα δεδομένα **(semi-structured data)** δεν είναι τα ακατέργαστα δεδομένα και δεν αποθηκεύονται σε ένα συμβατικό σύστημα βάσεων δεδομένων. Είναι δομημένα δεδομένα, αλλά δεν είναι οργανωμένα σε ένα ορθολογικό μοντέλο όπως ένας πίνακας ή ένας γράφος με βάση αντικείμενα. Τα ημιδομημένα δεδομένα περιέχουν ετικέτες ή δείκτες για τον διαχωρισμό σημασιολογικών στοιχείων και την επιβολή ιεραρχιών εγγραφών και πεδίων εντός των δεδομένων. Οι οντότητες που ανήκουν στην ίδια κλάση μπορεί να έχουν διαφορετικά χαρακτηριστικά, παρόλο που ομαδοποιούνται μαζί, ανεξάρτητα από τη σειρά των χαρακτηριστικών. Οι γλώσσες σήμανσης όπως η XML, το ηλεκτρονικό ταχυδρομείο και η Ηλεκτρονική Ανταλλαγή Δεδομένων (EDI - Electronic Data Interchange) αποτελούν μορφές ημιδομημένων δεδομένων. Αυτές υποστηρίζουν ένθετα ή ιεραρχικά δεδομένα απλοποιώντας τα μοντέλα δεδομένων που αναπαριστούν πολύπλοκες σχέσεις μεταξύ οντοτήτων. Υποστηρίζουν επίσης τις λίστες αντικειμένων που απλοποιούν τα μοντέλα δεδομένων αποφεύγοντας τις ακατάστατες μεταφράσεις των λιστών σε ένα σχεσιακό μοντέλο δεδομένων.

Τα οιονεί δομημένα δεδομένα **(quasi-structured data)** είναι περισσότερο δεδομένα κειμένου με ακανόνιστες μορφές δεδομένων. Μπορούν να μορφοποιηθούν με προσπάθεια, εργαλεία και χρόνο. Αυτός ο τύπος δεδομένων περιλαμβάνει δεδομένα ροής κλικ στο διαδίκτυο, όπως οι αναζητήσεις στο Google. Άλλα παραδείγματα είναι τα επικολλημένα κείμενα τα οποία αποδίδουν έναν χάρτη δικτύου με βάση την ομοιότητα της γλώσσας εντός του κειμένου, καθώς και την εγγύτητα των λέξεων μεταξύ τους εντός του κειμένου. Ωστόσο, δεν είναι "επισημειωμένα" (tagged) με τον τρόπο που το YouTube και το Flickr παρακολουθούν το περιεχόμενο των εικόνων. Γενικά, για να λάβετε δεδομένα κειμένου χωρίς ετικέτες ή με βάση εικόνες, δοκιμάστε έναν αλγόριθμο για την ανάλυσή τους και βελτιώστε τον με βάση τα αποτελέσματα που λαμβάνετε.



Σχήμα 1-4: Μορφές μεγάλων δεδομένων [7]

Πιστεύεται ότι τα πραγματικά δομημένα δεδομένα αποτελούν μόνο το 5% των συνολικών δεδομένων και επομένως χρειάζονται καλύτεροι τρόποι για την ανάλυση του υπόλοιπου 95%. Η παραδοσιακή ανάλυση βάσεων δεδομένων ή η αναζήτηση τυποποιημένων κειμένων δεν βοηθά στην ολοκλήρωση της συνολικής ανάλυσης. [7]

Ορισμένοι συνεχίζουν να ταξινομούν τα δεδομένα σε δομημένα και μη δομημένα δεδομένα μόνο με τα ημιδομημένα και τα οιονεί δομημένα ως υποτύπους των μη δομημένων για να αποφύγουν τη σύγχυση. Το πιο σημαντικό είναι ότι οι οργανισμοί συνειδητοποιούν τα οφέλη της ανάλυσης δεδομένων πέρα από τις βάσεις δεδομένων και, ως εκ τούτου, προχωρούν πέρα από τις αναφορές MIS (Management Information Systems).

1.3.3 Τα μεγάλα δεδομένα δεν είναι μόνο όγκος

Από την πλευρά της πληροφορικής, το πρώτο πράγμα που όλοι θέλουν να συζητήσουν όταν συζητούν για δεδομένα είναι το μέγεθος των δεδομένων. Πόσο μεγάλα είναι; Πόση φυσική αποθήκευση χρειάζεστε; Όταν πρόκειται για τη μετονομασία αυτών των δεδομένων σε Big Data, το ίδιο το όνομα υποδηλώνει ότι μιλάμε για πραγματικά μεγάλους όγκους δεδομένων.

Τα δεδομένα πρέπει να έχουν νόημα και να δημιουργούν ουσιαστικά αποτελέσματα για την επιχείρηση. Ως εκ τούτου, είναι σημαντικό να κατανοηθούν τα χαρακτηριστικά αυτών των μεγάλων δεδομένων, ώστε να αναπτυχθούν καλύτερα οι τρόποι και τα μέσα ανάλυσης αυτών των δεδομένων. Θα βοηθήσει επίσης στον καθορισμό του αποτελέσματος που πρέπει να περιμένει μια επιχείρηση από αυτά τα δεδομένα.

Οι ερευνητές όρισαν το πρώτο χαρακτηριστικό μοντέλο για την κατανόηση αυτού του γεγονότος χρησιμοποιώντας τα ακόλουθα **τρία V**.

- **Volume (Όγκος):** Ο όγκος καθορίζει το μέγεθος των δεδομένων που συλλέγονται και αποθηκεύονται. Η ανησυχία δεν αφορά μόνο τον αποθηκευτικό χώρο που απαιτείται για την αποθήκευση, αλλά και τους πόρους που απαιτούνται για την επεξεργασία αυτού του τεράστιου όγκου δεδομένων ανεξάρτητα από την πηγή των δεδομένων και τη δημιουργία ενός αποτελέσματος σε πραγματικό χρόνο από αυτά. Οι υποδομές και οι αρχιτέκτονες αποθήκευσης δεδομένων έχουν καταφέρει να επεξεργαστούν αυτό το όγκο δεδομένων πολλών terabyte με λογικό κόστος για ορισμένες μεγάλες επιχειρήσεις. Με την εμφάνιση εξαιρετικά κλιμακούμενων, χαμηλού κόστους συστημάτων αποθήκευσης και επεξεργασίας πολλαπλών πυρήνων, οι αναλύσεις γίνονται ευκολότερες και πιο προσιτές για τις μικρότερες επιχειρήσεις, ενώ αυξάνονται οι όγκοι σε petabytes και exabytes για τις μεγαλύτερες επιχειρήσεις. [8]

- **Velocity (Ταχύτητα)**: Η παραγωγή δεδομένων έχει αλλάξει από τις παραδοσιακές εφαρμογές, όπως η τιμολόγηση ή η παραγωγή, όπου τα δεδομένα παράγονται μόνο κατά τις ώρες παραγωγής και περιορίζονται σε πόσα τιμολόγια την ημέρα ή πόση παραγωγή την ημέρα. Οι νέες εφαρμογές, όπως η ειδοποίηση βάσει συμβάντων ή η παρακολούθηση της ροής ελέγχου, απαιτούν γρήγορη επεξεργασία δεδομένων. Οι επιχειρήσεις θέλουν πλέον να βλέπουν τα αποτελέσματα σε μια στιγμή και να βλέπουν τον αντίκτυπο κάθε νέας συναλλαγής. Το γεγονός αυτό έχει δημιουργήσει μια νέα διάσταση στην ανάλυση δεδομένων, γνωστή ως "ανάλυση ροής" (streaming analysis). [8]

- **Variety (Ποικιλία)**: Ο όγκος δεν προέρχεται μόνο από τα δομημένα δεδομένα ή τις εφαρμογές που βασίζονται σε βάσεις δεδομένων. Τα σύνολα δεδομένων έχουν πολλές νέες μορφές. Τα μέσα κοινωνικής δικτύωσης είναι μία από τις πιο σημαντικές νέες ποικιλίες που έχουν αναδυθεί. Οι επιχειρήσεις θέλουν να καταγράψουν πώς αισθάνονται οι χρήστες για τα προϊόντα τους στα μέσα κοινωνικής δικτύωσης και να σχεδιάσουν ανάλογα τις μελλοντικές στρατηγικές τους. Άλλα επιτεύγματα των μέσων κοινωνικής δικτύωσης περιλαμβάνουν τη μάθηση σχετικά με την εξάπλωση επιδημικών ασθενειών και τον ανάλογο σχεδιασμό της διανομής εμβολίων. Οι εφαρμογές RFID (Radio Frequency Identification), οι μεγάλες ποσότητες PDF, τα μηνύματα ηλεκτρονικού ταχυδρομείου, τα ηχογραφημένα φωνητικά μηνύματα και βίντεο κ.λπ. συμπληρώνουν την ποικιλία των δεδομένων. [8]

Η πολυπλοκότητα (Complexity) που προκαλείται από αυτές τις τρεις διαστάσεις γίνεται η τέταρτη διάσταση. Πολλαπλοί παράγοντες θα αλληλεπιδρούν για να κάνουν την πρόκληση της διαχείρισης δεδομένων πιο περίπλοκη.

Για παράδειγμα, η αύξηση της ταχύτητας παραγωγής νέων δεδομένων ωθεί στα άκρα την ταχύτητα (velocity). Από την άλλη πλευρά, μεγάλοι όγκοι που παράγονται με αυτή την ταχύτητα, ωθούν στο ακραίο σημείο τον όγκο (volume). Ως εκ τούτου, οι ηγέτες της πληροφορικής δεν μπορούν να επικεντρωθούν μόνο σε ένα άκρο, αλλά πρέπει να είναι προετοιμασμένοι να χειρίζονται και τα

τέσσερα ταυτόχρονα και να διαχειρίζονται την αλληλεπίδραση μεταξύ τους. Ένας τρόπος χειρισμού είναι η γρήγορη ανάλυση των δεδομένων φέρνοντας την εφαρμογή στα δεδομένα, σε αντίθεση με την παραδοσιακή πρακτική της μεταφοράς των δεδομένων στην εφαρμογή. Αυτό μπορεί να βοηθήσει στον ταυτόχρονο χειρισμό πολλαπλών σεναρίων.

Τα νέα εργαλεία, τόσο στη συλλογή δεδομένων όσο και στις εφαρμογές, βοηθούν στη συλλογή και επεξεργασία αυτού του μεγάλου όγκου δεδομένων που έρχονται με μεγάλη ταχύτητα από μια ποικιλία νέων πηγών δεδομένων. Πλατφόρμες όπως το κατανεμημένο σύστημα αρχείων Hadoop, το MapReduce και το Hive επιτρέπουν τη μηχανική ανάπτυξη νέων και καινοτόμων μεθόδων επεξεργασίας δεδομένων.

Ενώ τα τρία "V" που περιγράφονται παραπάνω βοηθούν στην κατανόηση των Big Data από τεχνική άποψη, υπάρχει ένα άλλο μοντέλο "V" που βοηθά στην κατανόηση του ίδιου από επιχειρηματική άποψη. Σε αυτά περιλαμβάνονται:

- **Variability (Μεταβλητότητα)**: Εξετάστε τα σύνολα δεδομένων πιο προσεκτικά. Διαπιστώστε αν η δομή του πλαισίου της ροής δεδομένων είναι τακτική και αξιόπιστη. Αποφασίστε πώς μπορεί να ερμηνευτεί η φύση και το πλαίσιο του περιεχομένου των δεδομένων κειμένου με τρόπο που να αποκτά νόημα για τα απαιτούμενα επιχειρηματικά μοντέλα που είναι έτοιμα για ανάλυση. [9]

- **Veracity (Αλήθεια)**: Επαληθεύει αν τα δεδομένα είναι κατάλληλα για το σκοπό τους και μπορούν να χρησιμοποιηθούν στο πλαίσιο του αναλυτικού μοντέλου. Αποφασίστε αν είναι αξιόπιστο να αναλυθούν τα δεδομένα αυτά με βάση ένα σύνολο καθορισμένων κριτηρίων. Καταγράψτε τις επιχειρησιακές διαδικασίες που επιτρέπουν τη σκιαγράφηση και την επικύρωση των δεδομένων. Εάν εντοπιστούν τα προβλήματα, αναλάβετε τις κατάλληλες δραστηριότητες για την αποκατάσταση των δεδομένων πριν από την εκτέλεση οποιασδήποτε ανάλυσης. [10]

- **Value (Αξία):** Προσδιορίστε τον σκοπό, το σενάριο ή το επιχειρηματικό όφελος που επιδιώκει να αντιμετωπίσει η αναλυτική λύση. Αναλύστε διεξοδικά ποιο είναι το πιθανό αποτέλεσμα και ποιο είναι το επιθυμητό αποτέλεσμα της ανάλυσης. Ποιες ενέργειες πρέπει να γίνουν για να επιτευχθούν τα επιθυμητά αποτελέσματα; Βεβαιωθείτε ότι η ανάλυση που πρόκειται να διεξαχθεί ανταποκρίνεται σε ηθικά ζητήματα χωρίς επιπτώσεις στη φήμη ή τη συμμόρφωση. [11]

Τα τρία χαρακτηριστικά των δεδομένων με επιχειρηματικό προσανατολισμό επιβάλλουν τη δημιουργία και την κοινοποίηση μιας σαφούς και κοινής κατανόησης του επιχειρηματικού πλαισίου. Ο ορισμός χρησιμοποιείται για να πλαισιώσει το νόημα και τον σκοπό του περιεχομένου των δεδομένων. Και τα τρία δείχνουν τις διαφορετικές πτυχές της καταλληλότητας για τον σκοπό των εν λόγω συνόλων δεδομένων και τον τρόπο αντιμετώπισής τους για την επίλυση ενός επιχειρηματικού προβλήματος.

1.3.4 Οι νέες διαστάσεις των δεδομένων

Τα μεγάλα δεδομένα συχνά ξεκινούν τη συζήτηση σχετικά με τις νέες διαστάσεις που ορίζονται για τα δεδομένα. Αυτές πρέπει να αντιμετωπιστούν με διαφορετικό τρόπο από τον απλό χειρισμό των μεγάλων δεδομένων. Αυτές οι νέες προκλήσεις είναι οι εξής:

- **Δεδομένα σε πραγματικό χρόνο (Real-time data):** Τα δεδομένα αυτά διαφέρουν από την παραδοσιακή μορφή δεδομένων που αποθηκεύουμε στους διακομιστές μας. Δεν έχει σημασία αν αυτά εμπίπτουν στον τύπο δομημένων ή μη δομημένων δεδομένων. Η βασική πτυχή είναι ότι πρόκειται για τα "τρέχοντα δεδομένα" και όχι για τα παλιά δεδομένα. Επιτρέπει την επίγνωση της κατάστασης σχετικά με το τι συμβαίνει τώρα. Τα δεδομένα πραγματικού χρόνου εγείρουν το ζήτημα των φθαρτών και ορφανών δεδομένων, τα οποία δεν έχουν πλέον έγκυρες περιπτώσεις χρήσης, αλλά συνεχίζουν να χρησιμοποιούνται παρ' όλα αυτά.

- **Κοινόχρηστα δεδομένων (Shared data):** Αυτό αφορά τις πληροφορίες που μοιράζονται σε ολόκληρο τον οργανισμό. Αυτό περιλαμβάνει την κοινή χρήση πληροφοριών μεταξύ διαφόρων εφαρμογών και πηγών δεδομένων. Για την αποτελεσματική κοινή χρήση πληροφοριών, οι επιχειρήσεις πρέπει να διασφαλίζουν ότι τα δεδομένα είναι συνεπή, χρησιμοποιήσιμα και επεκτάσιμα. Η σημαντική πτυχή εδώ είναι ότι η κοινή χρήση πληροφοριών περιπλέκει το έργο του καθορισμού της αυθεντίας των πληροφοριών.

- **Συνδεδεμένα δεδομένα (Linked data):** Αυτά προέρχονται από τις διάφορες πηγές δεδομένων που έχουν σχέσεις μεταξύ τους και διατηρούν αυτό το πλαίσιο ώστε να είναι χρήσιμα για τους ανθρώπους και τους υπολογιστές. Μόλις ένας χρήστης συνδέσει τα δεδομένα, η σχέση σε αυτά τα δεδομένα παραμένει από εκείνο το σημείο και μετά.

- **Δεδομένα υψηλής πιστότητας (High-fidelity data):** Τα δεδομένα αυτά διατηρούν το πλαίσιο, τις λεπτομέρειες, τις σχέσεις και τις ταυτότητες των σημαντικών επιχειρηματικών πληροφοριών. Αυτό επιτυγχάνεται σε μεγάλο βαθμό μέσω των ενσωματωμένων μεταδεδομένων. Τα δεδομένα υψηλής πιστότητας επιτρέπουν την προσθήκη νέου νοήματος χωρίς να καταστρέφεται το προηγούμενο νόημα των δεδομένων. [5]

Οι παραδοσιακές τεχνικές διαχείρισης πληροφοριών προϋποθέτουν συνεκτικό έλεγχο τόσο των διαδικασιών αποθήκευσης όσο και της ακεραιότητας των πληροφοριών. Στη συνέχεια, συνδέουν διαφορετικά σύνολα μέσω οδηγιών μεταδεδομένων, οι οποίες εκτελούνται σε έναν τύπο διακομιστή εφαρμογών. Με τα μεγάλα δεδομένα, η διεργασία πρέπει να μετακινείται προς τα δεδομένα, αντί να μετακινεί τα δεδομένα από την αποθηκευμένη θέση τους σε μια διεργασία και στη συνέχεια πίσω για να γράψει το αποτέλεσμα. Το μόνο πλεονέκτημα της παραδοσιακής προσέγγισης είναι ότι τυχαίνει να βρίσκεται στη θέση της. Ωστόσο, δεν έχει ακόμη προσφέρει κανένα πραγματικό πλεονέκτημα και στην πραγματικότητα αυξάνει τον αριθμό των ωρών που απαιτούνται για τη συντήρηση και την τροποποίησή της.

Με την είσοδο μεγάλων δεδομένων, οι παραδοσιακές προσεγγίσεις θα αποτύχουν. Οι αρχιτέκτονες δεδομένων και συστημάτων έχουν συνειδητοποιήσει ότι οι άμεσες επιχειρηματικές ανάγκες οδηγούν την αρχιτεκτονική πληροφοριών κατά μήκος μιας ή περισσότερων διαστάσεων της διαχείρισης δεδομένων, αποκλείοντας μερικές φορές την άλλη διάσταση. Ως εκ τούτου, τη στιγμή που οποιαδήποτε πληροφορία εγκαταλείπει την αρχική της διαδικασία, οι αποκλεισμένες διαστάσεις επιβεβαιώνονται εκ νέου.

1.3.5 Οφέλη από τα μεγάλα δεδομένα σε ορισμένες βιομηχανίες

Οι έννοιες των μεγάλων δεδομένων πρέπει να εφαρμοστούν σε πολλές βιομηχανίες, καθώς συλλέγουν δεδομένα από πολλαπλές πηγές και αποθηκεύουν μεγάλο όγκο αυτών των δεδομένων. Η συλλογή και η αποθήκευση μπορεί αρχικά να γίνεται για λόγους συμμόρφωσης. Ωστόσο, η επένδυση λίγο παραπάνω σε αυτό για να αποκτήσει κανείς χρήσιμη εικόνα των δεδομένων είναι ένα πρόσθετο όφελος. [12]

Οι οργανισμοί κατανοούν πλέον καλύτερα τι είναι τα Μεγάλα Δεδομένα και ποια επιχειρηματικά οφέλη μπορούν να προκύψουν από αυτά. Οι πρωτοβουλίες Big Data καθίστανται κρίσιμες για τους οργανισμούς, καθώς δεν είναι σε θέση να ανταποκριθούν στις επιχειρηματικές τους απαιτήσεις με τις παραδοσιακές πηγές δεδομένων, τεχνολογίες ή πρακτικές. Οι ηγέτες της πληροφορικής και των επιχειρήσεων αρχίζουν να αισθάνονται ότι έχουν μείνει πίσω στην αξιοποίηση των πλεονεκτημάτων αυτών των νέων τεχνολογιών.

Ωστόσο, οι περισσότεροι οργανισμοί που έχουν υιοθετήσει τα Big Data βρίσκονται στα αρχικά στάδια του σχεδιασμού των πρωτοβουλιών τους. Ακόμα και όσοι έχουν τεθεί σε λειτουργία τείνουν να αφορούν συγκεκριμένες λειτουργίες και διαφέρουν σημαντικά ως προς τους επιχειρηματικούς

στόχους και τις επιχειρηματικές και τεχνολογικές απαιτήσεις τους. Αυτό καθιστά δύσκολη την επιλογή της σωστής προσέγγισης και λύσης.

Μεγάλες επιχειρήσεις σε πολλούς κλάδους έχουν αρχίσει να εφαρμόζουν τις στρατηγικές τους για τα μεγάλα δεδομένα και έχουν αρχίσει να βλέπουν οφέλη. Οι συνάδελφοί τους στον ίδιο τομέα ή ακόμη και σε άλλον τομέα έχουν επίσης αρχίσει να μαθαίνουν από αυτούς και τους πειραματισμούς τους για να σχεδιάσουν ένα πλαίσιο για να εφαρμόσουν τις δικές τους στρατηγικές.

1.3.6 Μεγάλα δεδομένα στην εκπαίδευση

Διάφορες πρωτοβουλίες και εφαρμογές μεγάλων δεδομένων βοηθούν μεγάλα πανεπιστήμια και εκπαιδευτικά συστήματα να παρέχουν καλύτερες υπηρεσίες στους φοιτητές τους. Υπάρχουν διάφορα στάδια στα οποία αυτές οι πρωτοβουλίες βοηθούν το εκπαιδευτικό σύστημα. Για παράδειγμα, σε μια πρωτοβουλία πριν από την τάξη, τα μεγάλα δεδομένα βοηθούν τους μαθητές να ταιριάξουν με μια μαθησιακή ευκαιρία που ταιριάζει καλύτερα στις ανάγκες τους με το μαθησιακό προφίλ.

Έχουν χορηγηθεί επιχορηγήσεις σε εκπαιδευτικά συστήματα που στοχεύουν στη δημιουργία συστημάτων που επιτρέπουν στους κρατικούς και τοπικούς φορείς εκπαίδευσης να βελτιώσουν τη μάθηση και τα αποτελέσματα των μαθητών μέσω αποφάσεων που βασίζονται σε δεδομένα. Τέτοια δεδομένα μπορούν να χρησιμοποιηθούν για να βοηθήσουν τα σχολεία στην ανάθεση μαθητών σε κατάλληλα μαθησιακά περιβάλλοντα και μαθήματα σπουδών και είναι ιδιαίτερα χρήσιμα για την κατανόηση των αναγκών των μαθητών που μετακινούνται μεταξύ σχολείων και σχολικών περιφερειών μία ή περισσότερες φορές κατά τη διάρκεια ενός σχολικού έτους.

Μια βασική δραστηριότητα μετά την τάξη είναι η επίτευξη βελτιώσεων σε μακροκλίμακα σε τομείς όπως το θεσμικό πρόγραμμα σπουδών, ώστε να συμβάλει στην ικανοποίηση των αναγκών των φοιτητών και των εργοδοτών. Αυτό είναι κάτι παρόμοιο με όταν μια εταιρεία αλλάζει το επιχειρηματικό της μοντέλο για να είναι πιο επιτυχημένη σε μια αγορά στην οποία έχει επιλέξει να στοχεύσει. Το έργο Predictive Analytics Reporting Framework αντιπροσωπεύει πάνω από 600.000 εγγραφές φοιτητών και περισσότερες από 3 εκατομμύρια εγγραφές μαθημάτων από έξι μόνο ιδρύματα [13]. Ο στόχος του ήταν να αναζητήσει τα μοτίβα στις σχέσεις που δεν είναι ορατά, εκτός αν μπορεί κανείς να εξετάσει όλα τα ιδρυματικά δεδομένα από μια ενιαία βάση δεδομένων πολλαπλών ιδρυμάτων.

Η αρχική ανάλυση που έγινε στο τέλος των οκτώ μηνών του έργου παρήγαγε σημαντικά αποτελέσματα και επεκτάθηκε περαιτέρω ώστε να συμπεριλάβει άλλα 10 ινστιτούτα και να αντλήσει από μια πολύ μεγαλύτερη βάση δεδομένων.

Ένα άλλο παράδειγμα ήρθε από την ιταλική κυβέρνηση, η οποία συνέλεξε δεδομένα όπως το πόσο γρήγορα οι φοιτητές βρίσκουν δουλειά, τους μισθούς τους και τη σημασία των πτυχίων τους για την απόκτηση εργασίας. Αυτό μπορεί να χρησιμοποιηθεί για την απόκτηση κρίσιμων γνώσεων σχετικά με το πώς πρέπει να βελτιωθεί το συνολικό εκπαιδευτικό σύστημα, ώστε να παραδώσει στην αγορά μια καλύτερα εκπαιδευμένη γενιά.

1.3.7 Μεγάλα δεδομένα στην υγειονομική περίθαλψη

Η υγειονομική περίθαλψη διαθέτει μεγάλα σύνολα δεδομένων. Ακόμη και τα μικρότερα νοσοκομεία καταγράφουν πολλές ακτινογραφίες, σαρώσεις, αρχεία εξετάσεων κ.λπ. Υπάρχουν αρκετές περιπτώσεις χρήσης όπου τα νοσοκομεία μπορούν να αναλύσουν τα δεδομένα και να παράγουν σημαντικές πληροφορίες. Η καταγραφή δεδομένων δεν αποτελεί πρόβλημα για αυτά, καθώς λαμβάνουν δεδομένα ασθενών σε πραγματικό χρόνο, αποτελέσματα εξετάσεων ενώ ο ασθενής υποβάλλεται σε περίθαλψη και επιπτώσεις των φαρμάκων. Αυτά μπορούν να συσχετιστούν με τα δεδομένα των προσωπικών, περιβαλλοντικών και οικονομικών συνθηκών του ασθενούς. [15]

Ένα παράδειγμα υγειονομικής περίθαλψης που το χρησιμοποιεί αυτό είναι οι συσκευές που καταγράφουν σε πραγματικό χρόνο τις παραμέτρους υγείας του ασθενούς μέσω μιας φορητής συσκευής που καταγράφει βασικές παραμέτρους υγείας, όπως τα επίπεδα σακχάρου στο αίμα, την αρτηριακή πίεση, τον καρδιακό ρυθμό κ.λπ. από καιρό σε καιρό μαζί με τη σωματική δραστηριότητα και τις περιβαλλοντικές συνθήκες, δηλαδή τις καιρικές συνθήκες. Τα δεδομένα αυτά μεταφορτώνονται αυτόματα στους διακομιστές και οι γιατροί σε απομακρυσμένες τοποθεσίες μπορούν να βοηθήσουν τον ασθενή με φαρμακευτική αγωγή χωρίς να χρειάζεται να μεταβεί στο νοσοκομείο.

Τα ίδια σύνολα δεδομένων μπορούν να χρησιμοποιηθούν για να βρεθεί τι είδους ασθένεια αντιμετωπίζουν οι άνθρωποι σε διαφορετικές περιβαλλοντικές συνθήκες κατά καιρούς. Η συσχέτιση αυτών με περισσότερα δεδομένα από τα μέσα κοινωνικής δικτύωσης μπορεί να πιστοποιήσει καθώς και να επεκτείνει το πεδίο εφαρμογής για την επίτευξη καλύτερων αποτελεσμάτων.

Τα νοσοκομεία μπορούν να κατανοήσουν καλύτερα το αποτέλεσμα της φαρμακευτικής αγωγής στους ασθενείς τους, καθώς και παρόμοια αποτελέσματα από άλλους γιατρούς και νοσοκομεία, ώστε να λάβουν τεκμηριωμένη απόφαση σχετικά με το επόμενο επίπεδο φαρμακευτικής αγωγής και θεραπείας. Η καταγραφή των εμπειριών πολλών γιατρών και νοσοκομείων για τη λήψη τεκμηριωμένης απόφασης προσθέτει μια πιο επιστημονική προσέγγιση στη θεραπεία πέρα από την εμπειρία του γιατρού.

1.3.8 Τα μεγάλα δεδομένα στο χρηματοοικονομικό τομέα

Τα μεγάλα δεδομένα στα χρηματοοικονομικά περιγράφουν τον τεράστιο όγκο δεδομένων (δομημένα, μη δομημένα και ημιδομημένα δεδομένα) που παράγουν καθημερινά τα χρηματοπιστωτικά ιδρύματα. Οι πληροφορίες προέρχονται από διάφορες πηγές, συμπεριλαμβανομένων των χρηματιστηριακών πληροφοριών, των συναλλαγών των πελατών και των μέσων κοινωνικής δικτύωσης. Το μεγαλύτερο πρόβλημα για τις επιχειρήσεις είναι η συλλογή αυτών των δεδομένων, η ερμηνεία τους και η εξαγωγή διεισδυτικών συμπερασμάτων. Η διαδικασία αυτή ονομάζεται ανάλυση μεγάλων δεδομένων. Τα χρηματοπιστωτικά ιδρύματα βοηθούν στη λήψη αποφάσεων με βάση τα δεδομένα. [16]

Τα δεδομένα επικεντρώνονται στις τέσσερις κύριες κατηγορίες του χρηματοπιστωτικού κλάδου - χρηματοπιστωτικές αγορές, διαδικτυακές αγορές, εταιρείες δανεισμού και τράπεζες. Αυτές οι επιχειρήσεις παράγουν καθημερινά δισεκατομμύρια δεδομένα λόγω των τακτικών συναλλαγών τους, των λογαριασμών χρηστών, των ενημερώσεων δεδομένων, των τροποποιήσεων λογαριασμών και άλλων λειτουργιών.

Καλύπτουν διάφορες χρηματοπιστωτικές επιχειρήσεις, όπως ο διαδικτυακός δανεισμός από ομότιμους σε ομότιμους (peer to peer), η χρηματοδότηση μικρών και μεσαίων επιχειρήσεων (ΜΜΕ), οι πλατφόρμες διαχείρισης πλούτου και περιουσιακών στοιχείων, οι πλατφόρμες crowdfunding, η διαχείριση συναλλαγών, η μεταφορά χρημάτων ή εμβασμάτων, οι πλατφόρμες πληρωμών μέσω κινητών τηλεφώνων, τα κρυπτονομίσματα κ.λπ. Οι επιχειρήσεις αυτές αναλύουν μεγάλους όγκους μεταβλητών δεδομένων με μεγάλη ταχύτητα και τα αξιοποιούν για την πρόβλεψη των προτιμήσεων των καταναλωτών. Οι αναλύσεις αυτές βασίζονται στην προηγούμενη συμπεριφορά και στον βαθμό πιστωτικού κινδύνου που συνδέεται με κάθε χρήστη.

Άλλοι τρόποι με τους οποίους τα μεγάλα δεδομένα επηρεάζουν τη χρηματοδότηση, περιλαμβάνουν την προώθηση της διαφάνειας, την αξιολόγηση του κινδύνου, τις αλγοριθμικές συναλλαγές, τη χρήση των δεδομένων των καταναλωτών και την τροποποίηση της κουλτούρας. Επιπλέον, τα μεγάλα δεδομένα επηρεάζουν σε μεγάλο βαθμό την οικονομική μοντελοποίηση και έρευνα. Οι χρηματοπιστωτικές επιχειρήσεις χρησιμοποιούν τα μεγάλα δεδομένα για τη δημιουργία σύνθετων μοντέλων λήψης αποφάσεων με τη χρήση πολυάριθμων προγνωστικών αναλύσεων και την

παρακολούθηση των προτύπων δαπανών. Μέσω αυτού, οι βιομηχανίες μπορούν να επιλέξουν τα χρηματοοικονομικά προϊόντα που επιθυμούν να προσφέρουν.

Κάθε μέρα, χιλιάδες νέα δεδομένα παράγονται από όλες αυτές τις υπηρεσίες. Ως εκ τούτου, πιστεύεται επίσης ότι η διατήρηση αυτών των δεδομένων είναι το πιο κρίσιμο στοιχείο αυτών των υπηρεσιών. Ως εκ τούτου, οποιαδήποτε απώλεια δεδομένων θα μπορούσε να οδηγήσει σε σημαντικά ζητήματα για τον συγκεκριμένο χρηματοπιστωτικό τομέα.

Εφαρμογές στο χρηματοοικονομικό τομέα

Υπάρχει μια ποικιλία εφαρμογών για τη χρήση των μεγάλων δεδομένων και ορισμένες από αυτές παρατίθενται παρακάτω:

1 - Χρηματοοικονομικές αγορές

Οι πληροφορίες για τις χρηματοπιστωτικές αγορές μπορούν να αντληθούν από διάφορες πηγές, όπως τα μέσα κοινωνικής δικτύωσης (όπως το Facebook και το Twitter), οι εφημερίδες, οι διαφημίσεις, η τηλεόραση και τα στατιστικά στοιχεία της χρηματιστηριακής αγοράς (όπως οι τιμές των μετοχών, τα επιτόκια, ο όγκος συναλλαγών κ.λπ.). Τα δεδομένα αυτά διαδραματίζουν σημαντικό ρόλο στη χρηματοπιστωτική αγορά, όπως η πρόβλεψη της μεταβλητότητας της αγοράς, η πρόβλεψη των αποδόσεων της αγοράς, η αποτίμηση των θέσεων στην αγορά, ο εντοπισμός του υπερβολικού όγκου συναλλαγών, η ανάλυση του κινδύνου της αγοράς, η κίνηση των μετοχών, η τιμολόγηση των δικαιωμάτων προαίρεσης, η ιδιοσυγκρασιακή μεταβλητότητα, οι αλγοριθμικές συναλλαγές κ.λπ.

2 - Τραπεζικός τομέας

Ο τραπεζικός τομέας αποτελεί σημαντικό τμήμα του χρηματοπιστωτικού τομέα και βασικό χρήστη της ανάλυσης δεδομένων. Ο τραπεζικός τομέας χρησιμοποιεί την ανάλυση δεδομένων για την ανάλυση τάσεων, την εκτίμηση κινδύνων, την ανάλυση, την ανάλυση πελατών και την οικονομική πρόβλεψη.

3 - Επιχειρηματική απόδοση

Όταν οι επιχειρήσεις λαμβάνουν αποφάσεις σχετικά με τα προϊόντα τους, το μάρκετινγκ, το κοινό-στόχο, την παραγωγή και τις πωλήσεις τους χρησιμοποιώντας την ανάλυση δεδομένων, ανεξάρτητα από το μέγεθός τους, αποκτούν ανταγωνιστικό πλεονέκτημα, το οποίο βελτιώνει την απόδοσή τους.

4 - Χρηματοοικονομικά και Λογιστική

Οι επιχειρήσεις χρησιμοποιούν δεδομένα για λογιστικές, χρηματοοικονομικές και ελεγκτικές ανάγκες. Ως αποτέλεσμα, οι επιχειρήσεις έχουν αυξήσει την αποδοτικότητα και τις επιδόσεις τους για να έχουν τα καλύτερα δυνατά αποτελέσματα στις επιχειρησιακές τους δραστηριότητες με τη χρήση δεδομένων στα τμήματα ελέγχου και λογιστικής.

Οφέλη στο χρηματοοικονομικό τομέα

Παρακάτω παρατίθενται ορισμένα από τα οφέλη που προσφέρουν τα μεγάλα δεδομένα:

1 - Ανίχνευση απάτης

Στον ταχέως αναπτυσσόμενο ψηφιακό κόσμο, μία από τις σημαντικότερες ανησυχίες για τον τραπεζικό κλάδο είναι η αύξηση των κυβερνοεπιθέσεων που έχουν εκθέσει την ευπάθεια των ευαίσθητων δεδομένων των καταναλωτών. Ωστόσο, η ανάλυση της συμπεριφοράς των χρηστών και των μοτίβων δαπανών έχει επιτρέψει στις χρηματοπιστωτικές επιχειρήσεις να εντοπίζουν γρήγορα τις απάτες και τις απάτες. Αυτό καθίσταται εφικτό μέσω αλγορίθμων μηχανικής μάθησης και ανάλυσης δεδομένων.

2 - Διαχείριση κινδύνων

Η ανάλυση δεδομένων έχει καταστήσει εφικτό τον εντοπισμό των κινδύνων σε πραγματικό χρόνο και την προστασία των πελατών από την απάτη. Τα χρηματοπιστωτικά ιδρύματα μπορούν να το καταστήσουν αυτό εφικτό με τη δημιουργία ισχυρών συστημάτων διαχείρισης κινδύνων, κάτι που είναι και πάλι εφικτό μέσω της συλλογής των απαραίτητων πληροφοριών. Επιπλέον, οι χρηματοπιστωτικοί οργανισμοί χρησιμοποιούν την ανάλυση δεδομένων για να ενισχύσουν τα μοντέλα πρόβλεψης για τον προσδιορισμό των κινδύνων δανείων και την εκτίμηση του προβλεπόμενου κόστους. Τα ιδρύματα αυτά χρησιμοποιούν επίσης τα δεδομένα για να ανταποκρίνονται στις απαιτήσεις συμμόρφωσης και τις κανονιστικές απαιτήσεις, να μειώνουν τον λειτουργικό κίνδυνο, να αποτρέπουν την απάτη και να ξεπερνούν τα προβλήματα ασυμμετρίας πληροφοριών.

3 - Σχέσεις με τους πελάτες

Η συγκέντρωση και ανάλυση των δεδομένων των πελατών, η υποβολή χρήσιμων προσφορών και η εγγύηση της ασφάλειας των συναλλαγών συμβάλλουν στη διατήρηση ικανοποιημένων σχέσεων με τους πελάτες.

1.4 Μελέτες περιπτώσεων

Τα μεγάλα δεδομένα μπορούν να βοηθήσουν στην επίλυση προβλημάτων και στη βελτίωση των λειτουργιών, της παραγωγής και της συνολικής επιχειρηματικής δραστηριότητας σε όλους τους κλάδους. Σε αυτό το στάδιο, πολλές από αυτές θα μπορούσαν να είναι απλώς σπουδαίες ιδέες, αλλά θα υλοποιηθούν σύντομα για να βοηθήσουν στην άντληση αποτελεσμάτων από αυτές.

Ας εξετάσουμε εν συντομία μερικά ακόμη παραδείγματα από μερικούς ακόμη κλάδους:

- **Λιανικό εμπόριο**: Οι βελτιωμένες γνώσεις και η κατανόηση των συμπαθειών και αντιπαθειών, των επιρροών και των συμπεριφορών των πελατών μπορούν να οδηγήσουν σε αυξημένες πωλήσεις, απόδοση αποθεμάτων και επίγνωση του πλαισίου στο κατάστημα. Ένα από τα παραδείγματα περιλαμβάνει μια κατάσταση κατά την οποία βρίσκεστε σε ένα κατάστημα και εγκαταλείπετε ένα προϊόν αφού κοιτάξετε την τιμή του. Το κατάστημα αναγνωρίζει έξυπνα εσάς και το προϊόν και σας στέλνει γρήγορα ένα μήνυμα κειμένου για μια ειδική προσφορά για το προϊόν. Αυτό όχι μόνο έχει τη δυνατότητα να σας επηρεάσει να αγοράσετε το προϊόν και επομένως να αυξήσει τις πωλήσεις για το κατάστημα, αλλά και να νιώσετε ξεχωριστοί με αυτή την εξατομικευμένη εξυπηρέτηση που συμβαίνει αυτόματα.
- **Διαδικτυακά μέσα κοινωνικής δικτύωσης**: Πιθανόν να γνωρίζετε περιστατικά στο LinkedIn, όπου εμφανίζονται τα ονόματα ατόμων που επισκέφθηκαν άλλοι μετά την επίσκεψή σας στο ίδιο προφίλ. Πρόκειται για μια μικρή πρωτοβουλία που έχει τη δυνατότητα να σας βοηθήσει να συνδεθείτε με τους πιο γνωστούς πιο γρήγορα από το να τους αναζητήσετε.
- **Ηλεκτρονικά καταστήματα**: Τι θα λέγατε να πάρετε προσφορές για βιβλία των αγαπημένων σας συγγραφέων ενώ αγοράζετε ρούχα για τον εαυτό σας σε ένα ηλεκτρονικό κατάστημα; Για τη διευκόλυνσή σας, το κατάστημα προσθέτει αυτό που σας αρέσει να αγοράζετε περισσότερο. Και πάλι, με πιθανό αποτέλεσμα καλύτερες πωλήσεις και μεγαλύτερη ικανοποίηση των πελατών για επαναλαμβανόμενες επισκέψεις.
- **Δημόσιος τομέας**: Λεπτομερείς πληροφορίες σε πραγματικό χρόνο σχετικά με τις ροές κυκλοφορίας, τις θέσεις των οχημάτων και τη χρήση των πόρων υποστηρίζουν την παροχή υπηρεσιών και την αποτελεσματικότητα ενός οργανισμού σε ένα ευρύ φάσμα δημόσιων τομέων.
- **Υγειονομική περίθαλψη**: Η κατανόηση των προτύπων του τρόπου ζωής και του περιβάλλοντος μιας περιοχής θα μπορούσε να βοηθήσει τους Επιστήμονες Δεδομένων να προβλέψουν μια πιθανή επιδημία ή ασθένεια που εξαπλώνεται σε μια δεδομένη περιοχή. Αυτό θα μπορούσε

επίσης να βοηθήσει τις αρχές να προετοιμαστούν καλύτερα όταν η καταστροφή χτυπήσει και να διανείμουν ανάλογα τα εμβόλια.

1.5 Παρανοήσεις σχετικά με τα μεγάλα δεδομένα

Τα μεγάλα δεδομένα υπάρχουν εδώ και αρκετό καιρό. Ωστόσο, δεν γνωρίζουν όλοι τα πάντα γι' αυτά. Αρκετοί ηγέτες πληροφορικής έχουν τις ανησυχίες και τις αμφιβολίες τους σχετικά με την τεχνολογία Big Data.

Ας προσπαθήσουμε να κατανοήσουμε μερικά από αυτά και να δούμε αν πραγματικά μας απασχολούν:

- **Το 80% όλων των δεδομένων είναι αδόμητα:** Αυτή είναι μία από τις παλαιότερες παρανοήσεις σχετικά με τα δεδομένα και την ανάλυση δεδομένων. Δεδομένης της ποικιλίας των πηγών δεδομένων, αυτές φαίνονται αληθινές, αν και δεν είναι ακριβώς αληθινές. Δεν θα υπήρχαν μοτίβα προς ανακάλυψη στα δεδομένα αν δεν είχαν δομή. Η χρήση μη σχεσιακών βάσεων δεδομένων, όπως οι βάσεις δεδομένων NoSQL και οι βάσεις δεδομένων γράφων, συμβάλλει στη δημιουργία των δομών και των μοτίβων για τους περισσότερους τύπους δεδομένων.
- **Η προηγμένη ανάλυση είναι απλώς μια προηγμένη έκδοση της κανονικής ανάλυσης:** Πολλοί πιστεύουν ότι έχουν κατακτήσει την ανάλυση βάσεων δεδομένων και απλά χρειάζονται το επόμενο βήμα για να προχωρήσουν στην προηγμένη ανάλυση. Τα κανονικά analytics είναι κυρίως στατικές αναφορές από τις στατικές βάσεις δεδομένων. Τα προηγμένα analytics μας δίνουν τη δυνατότητα να καταλήγουμε σε έξυπνα συμπεράσματα και να επιλύουμε πραγματικά επιχειρηματικά προβλήματα από την ανάλυση των δεδομένων.
- **Η ενσωματωμένη ανάλυση λύνει όλα τα προβλήματα:** Ενσωματωμένα αναλυτικά στοιχεία είναι το τυποποιημένο σύνολο εργαλείων ή αναφορών που ενσωματώνονται πάνω στα δεδομένα και τα σύνολα δεδομένων. Όπως και τα κανονικά αναλυτικά εργαλεία, έτσι και αυτά παρέχουν μόνο αναφορές και δεν αναλύουν πραγματικά τα δεδομένα σχετικά με τα μοτίβα και τα αποτελέσματα ώστε να παρέχουν ουσιαστικά αποτελέσματα.
- **Τα βελτιωμένα εργαλεία θα αντικαταστήσουν τον επιστήμονα δεδομένων**: Ανεξάρτητα από τον τύπο του εργαλείου ή την εξέλιξη των εργαλείων, θα χρειαστείτε επιστήμονες/αναλυτές δεδομένων για να χρησιμοποιήσετε αυτά τα εργαλεία για να εκτελέσετε την ανάλυση και να λάβετε δυναμικές αναφορές. Σε κάθε περίπτωση, υπάρχει έλλειψη επιστημόνων δεδομένων, οπότε θα είναι πάντα απαραίτητοι.
- **Οι επιστήμονες δεδομένων χρειάζονται υψηλού επιπέδου εκπαίδευση:** Οι Επιστήμονες Δεδομένων χρειάζονται αναλυτικό, λογικό μυαλό για να καθορίσουν τους κανόνες και τις σχέσεις μεταξύ των πηγών δεδομένων για να έχουν θετικά αποτελέσματα. Πιστεύουμε ότι η εκπαίδευση αποκλειστικά και μόνο δεν βοηθά εάν η εφαρμογή του μυαλού και της λογικής δεν γίνεται σωστά. Είναι πιο σημαντικό να είστε πιο λογικοί και να κατανοείτε τις επιχειρηματικές ανάγκες.
- **Μπορούμε να προβλέψουμε τα πάντα με τα Μεγάλα Δεδομένα:** Ενώ μπορούμε να χρησιμοποιήσουμε τα μεγάλα δεδομένα για να σχηματίσουμε μοτίβα και να προβλέψουμε πολλά πράγματα, τα μεγάλα δεδομένα δεν μπορούν να προβλέψουν τα πάντα. Τα νοσοκομεία μπορούν να αναλύσουν ποιο είδος ανθρώπων διατρέχει μεγαλύτερο κίνδυνο καρδιακών παθήσεων, ώστε να ληφθούν προφυλάξεις, αλλά πολλά πράγματα σε πιο σύνθετους τομείς, όπως ο νόμος και η πολιτική, δεν μπορούν να προβλεφθούν.
- **Τα μεγάλα δεδομένα δεν είναι «προκατειλημμένα»:** Τα δεδομένα είναι πάντα προκατειλημμένα, ανεξάρτητα από τον όγκο ή την πηγή δεδομένων. Τα δεδομένα είναι αποτέλεσμα ορισμένων μετρήσεων και συλλέχθηκαν με κάποιο σκοπό. Πρέπει να τα προσεγγίσετε με προσοχή και να είστε προσεκτικοί ώστε να έχετε συλλέξει το σύνολο δεδομένων από τον εκπρόσωπο της ομάδας που μελετάτε, διαφορετικά θα λάβετε λάθος

δεδομένα.

1.6 Μεταβλητές πηγές δεδομένων

Οι σπουδαίες ιδέες σχετικά με τα μεγάλα δεδομένα προέρχονται από την κατανόηση του φάσματος των διαθέσιμων πηγών δεδομένων και των ερωτήσεων που μπορούν να απαντηθούν εάν αυτά ενσωματωθούν και συσχετιστούν. Οι πηγές δεδομένων είναι η πιο σημαντική πτυχή της στρατηγικής - τα αποτελέσματα επηρεάζονται από την επιλογή των δεδομένων και των πηγών δεδομένων [5]. Αυτές οι πηγές δεδομένων θα μπορούσαν να ταξινομηθούν ως εξής:

- **Επιχειρησιακά δεδομένα:** Τα δεδομένα αυτά αποτελούν βάσεις δεδομένων επεξεργασίας συναλλαγών ή/και αναλυτικής επεξεργασίας. Πρόκειται για τις παραδοσιακές βάσεις δεδομένων που διατηρούν στοιχεία πελατών, εργαζομένων, προμηθευτών κ.λπ. Μπορούν ακόμη και να συλλέγουν δεδομένα που βασίζονται σε αισθητήρες, όπως έξυπνοι μετρητές, συσκευές συνδεδεμένες στο Διαδίκτυο κ.λπ.
- **Σκοτεινά δεδομένα:** Αυτά αποτελούν τα παλιά δεδομένα που έχει συλλέξει ένας οργανισμός κατά τη διάρκεια της λειτουργίας του. Αυτά θα μπορούσαν να βρίσκονται σε αρχεία ή σε δεδομένα με τη λιγότερη δυνατή πρόσβαση. Η εξαγωγή αξιοποιήσιμων δεδομένων από αυτά τα δεδομένα μέσω ανάλυσης, επισήμανσης, σύνδεσης ή δόμησης θεωρείται η μεγαλύτερη ευκαιρία από τις περισσότερες επιχειρήσεις μεταξύ όλων των τύπων δεδομένων.
- **Εμπορικά δεδομένα:** Τα δεδομένα αυτά αποτελούν τα δεδομένα που θα είχαν συλλέξει οργανισμοί ερευνών ή έρευνας κατά τη διάρκεια μιας χρονικής περιόδου. Για παράδειγμα, έρευνα για μια νέα τεχνολογία μεταξύ της κοινότητας των CIO (Chief information officer) και στη συνέχεια συλλογή των πληροφοριών τους, οι οποίες αργότερα μοιράζονται/πωλούνται με άλλες προοπτικές.
- **Δημόσια δεδομένα:** Πολλές κυβερνήσεις έχουν αρχίσει να ανοίγουν τα «χρηματοκιβώτια» των δεδομένων τους. Αυτά γίνονται για να υποστηρίξουν την οικονομική ανάπτυξη και τις υπηρεσίες υγείας, πρόνοιας και εξυπηρέτησης των πολιτών σε διάφορα στάδια υλοποίησης. Αυτά έχουν σημαντική εμπορική αξία για την κατανόηση των τοπικών και παγκόσμιων συνθηκών της αγοράς, των πληθυσμιακών τάσεων, του καιρού κ.λπ.
- **Δεδομένα κοινωνικής δικτύωσης:** Η συμμετοχή στα μέσα κοινωνικής δικτύωσης γνωρίζει πρωτοφανή ανάπτυξη. Το Facebook και το LinkedIn ενημερώνουν την ταχέως αναπτυσσόμενη, ανεκτίμητη πηγή δεδομένων τους σχετικά με τις προτιμήσεις, τις τάσεις, τις στάσεις, τη συμπεριφορά, τα προϊόντα και τις εταιρείες. Οι αναρτήσεις, οι τάσεις, ακόμη και τα πρότυπα χρήσης χρησιμοποιούνται όλο και περισσότερο για τον εντοπισμό και την πρόβλεψη πελατών-στόχων και τμημάτων, ευκαιριών της αγοράς, ανταγωνιστικών απειλών, επιχειρηματικών κινδύνων, ακόμη και για την επιλογή ιδανικών υποψηφίων για πρόσληψη.

1.7 Τεχνολογική ανάπτυξη και περιορισμοί

Τα πλεονεκτήματα και οι εφαρμογές της ανάλυσης μεγάλων δεδομένων υλοποιούνται σε διάφορους τομείς. Η ανάπτυξη των κατανεμημένων συστημάτων αρχείων (π.χ. HDFS), η υπολογιστική νέφους (π.χ. Amazon EC2), η υπολογιστική σε συστοιχίες μνήμης (π.χ. Spark), η παράλληλη υπολογιστική (π.χ. Pig), η εμφάνιση των πλαισίων NoSQL, η πρόοδος των αλγορίθμων μηχανικής μάθησης (π.χ. Support Vector Machines, Deep Learning, Auto-Encoders, Random Forest) έκαναν πραγματικότητα την επεξεργασία μεγάλων δεδομένων.

Παρά την ανάπτυξη αυτών των τεχνολογιών και αλγορίθμων για τον χειρισμό μεγάλων δεδομένων, υπάρχουν ορισμένοι περιορισμοί, οι οποίοι περιγράφονται παρακάτω.

1. Ζητήματα επεκτασιμότητας και αποθήκευσης: Ο ρυθμός αύξησης των δεδομένων είναι πολύ ταχύτερος από τα υπάρχοντα συστήματα επεξεργασίας. Τα συστήματα αποθήκευσης δεν είναι

αρκετά ικανά να αποθηκεύσουν αυτά τα δεδομένα [17]. Υπάρχει ανάγκη να αναπτυχθεί ένα σύστημα επεξεργασίας που να ανταποκρίνεται όχι μόνο στις σημερινές αλλά και στις μελλοντικές ανάγκες.

2. Επικαιρότητα της ανάλυσης: Η αξία των δεδομένων μειώνεται με την πάροδο του χρόνου. Οι περισσότερες εφαρμογές, όπως η ανίχνευση απάτης στις τηλεπικοινωνίες, τις ασφάλειες και τις τράπεζες, απαιτούν ανάλυση των δεδομένων συναλλαγών σε πραγματικό χρόνο ή σχεδόν σε πραγματικό χρόνο [18].

3. Αναπαράσταση ετερογενών δεδομένων: Τα δεδομένα που λαμβάνονται από διάφορες πηγές είναι ετερογενή από τη φύση τους. Τα μη δομημένα δεδομένα, όπως οι εικόνες, τα βίντεο και τα δεδομένα των μέσων κοινωνικής δικτύωσης, δεν μπορούν να αποθηκευτούν και να υποστούν επεξεργασία με τη χρήση παραδοσιακών εργαλείων όπως η SQL. Τα έξυπνα κινητά τηλέφωνα καταγράφουν και μοιράζονται πλέον εικόνες, ήχους και βίντεο με απίστευτα αυξανόμενο ρυθμό, αναγκάζοντας τον εγκέφαλό μας να επεξεργάζεται περισσότερα. Ωστόσο, η διαδικασία αναπαράστασης εικόνων, ακουστικών και βίντεο στερείται αποτελεσματικής αποθήκευσης και επεξεργασίας [18].

4. Σύστημα ανάλυσης δεδομένων: Είναι κατάλληλα μόνο για δομημένα δεδομένα και στερούνται επεκτασιμότητας και επεκτασιμότητας. Αν και οι μη σχεσιακές βάσεις δεδομένων χρησιμοποιούνται για την επεξεργασία μη δομημένων δεδομένων, υπάρχουν όμως προβλήματα με τις επιδόσεις τους. Υπάρχει ανάγκη να σχεδιαστεί ένα σύστημα που να συνδυάζει τα πλεονεκτήματα τόσο των σχεσιακών όσο και των μη σχεσιακών συστημάτων βάσεων δεδομένων, ώστε να διασφαλίζεται η ευελιξία [18].

5. Έλλειψη δεξαμενής ταλέντων: Με την αύξηση του όγκου των (δομημένων και μη) δεδομένων που παράγονται, υπάρχει ανάγκη για ταλέντα. Η ζήτηση για άτομα με καλές αναλυτικές δεξιότητες στα μεγάλα δεδομένα αυξάνεται.

6. Απόρρητο και ασφάλεια: Νέες συσκευές και τεχνολογίες όπως το cloud computing παρέχουν μια πύλη πρόσβασης και αποθήκευσης πληροφοριών για ανάλυση. Αυτή η ενοποίηση αρχιτεκτονικών πληροφορικής θα δημιουργήσει μεγαλύτερους κινδύνους για την ασφάλεια των δεδομένων και την πνευματική ιδιοκτησία. Η πρόσβαση σε προσωπικές πληροφορίες όπως οι προτιμήσεις αγοράς και τα αρχεία λεπτομερειών κλήσεων θα οδηγήσει σε αύξηση των ανησυχιών για το απόρρητο [19]. Οι ερευνητές έχουν τεχνική υποδομή για πρόσβαση στα δεδομένα από οποιαδήποτε πηγή δεδομένων, συμπεριλαμβανομένων των τοποθεσιών κοινωνικής δικτύωσης, για μελλοντική χρήση, ενώ οι χρήστες δεν γνωρίζουν τα κέρδη που μπορούν να προκύψουν από τις πληροφορίες που δημοσιεύουν [20]. Οι ερευνητές των μεγάλων δεδομένων αποτυγχάνουν να κατανοήσουν τη διαφορά μεταξύ ιδιωτικότητας και ευκολίας.

7. Όχι πάντα καλύτερα δεδομένα: Η εξόρυξη μέσων κοινωνικής δικτύωσης προσέλκυσε ερευνητές. Το Twitter έχει γίνει μια δημοφιλής πηγή. Οι χρήστες του Twitter δεν αντιπροσωπεύουν τον παγκόσμιο πληθυσμό. Οι ερευνητές μεγάλων δεδομένων θα πρέπει να κατανοήσουν τη διαφορά μεταξύ μεγάλων δεδομένων και ολόκληρων δεδομένων. Τα tweets που περιέχουν αναφορές σε πορνογραφία και spam εξαλείφονται με αποτέλεσμα την ανακρίβεια της τοπικής συχνότητας. Υπάρχει πλεονασμός σε αριθμό χρηστών twitter και λογαριασμών twitter, ένας λογαριασμός στον οποίο έχουν πρόσβαση πολλά άτομα και πολλοί λογαριασμοί που δημιουργούνται από έναν χρήστη. Υπάρχουν ενεργοί χρήστες και παθητικοί χρήστες που απλώς συνδέονται για να ακούσουν. Υπάρχουν δύο τύποι λογαριασμών δημόσιοι και προστατευμένοι ή ιδιωτικοί. Τα δημόσια tweets είναι προσβάσιμα μέσω API. Λίγες εταιρείες και νεοφυείς επιχειρήσεις έχουν πρόσβαση στο firehose (που περιέχει όλα τα δημόσια tweets που δημοσιεύονται και εξαιρεί ιδιωτικά ή προστατευμένα tweets), ενώ άλλες έχουν πρόσβαση σε gardenhose (περίπου 10% δημόσια tweets) και spritzer (περίπου 1% των δημόσιων tweets). Επιπλέον, οι ερευνητές έχουν πολύ περιορισμένη πρόσβαση στο

firehose. Ως εκ τούτου, δεν παρέχει την ακρίβεια του μεγέθους του δείγματος του συνόλου δεδομένων και την ερμηνεία που προκύπτει από την ανάλυση [21].

8. Εκτός πλαισίου: Η μείωση δεδομένων είναι ένας από τους συνηθισμένους τρόπους προσαρμογής σε ένα μαθηματικό μοντέλο. Η διατήρηση του πλαισίου κατά την αφαίρεση δεδομένων είναι κρίσιμης σημασίας. Δεδομένα που είναι εκτός πλαισίου χάνουν νόημα και αξία. Υπάρχει μια εμμονή για το «κοινωνικό γράφημα» με την άνοδο των ιστότοπων κοινωνικής δικτύωσης. Τα μεγάλα δεδομένα εισάγουν δύο τύπους κοινωνικών δικτύων: «αρθρωτά δίκτυα» και «δίκτυα συμπεριφοράς». Τα αρθρωτά δίκτυα είναι εκείνα που προκύπτουν από τον καθορισμό επαφών μέσω της τεχνολογίας διαμεσολάβησης. Οι "Φίλοι", οι "Γνωστοί" στο Facebook, το "Follow" είναι το twitter και οι "Best Friends", "Friends" και άλλοι κύκλοι στο Google+ είναι παραδείγματα αρθρωτού δικτύου. Τα αρθρωτά δίκτυα δημιουργούνται για να έχουν ξεχωριστή ομάδα για φίλους, συναδέλφους, φίλους φίλων και να φιλτράρουν το περιεχόμενο που μπορεί να δει κάθε ομάδα. Τα δίκτυα συμπεριφοράς προέρχονται από τις αλληλεπιδράσεις των μέσων κοινωνικής δικτύωσης και τα πρότυπα επικοινωνίας. Αλλά τα πρότυπα επικοινωνίας δεν χρειάζεται απαραίτητα να αποκαλύπτουν τη δύναμη του δεσμού. Το αφεντικό και ένας υφιστάμενος έχουν μεγάλα μοτίβα επικοινωνίας σε σύγκριση με εκείνα με μέλη της οικογένειας. Αν και η ανάλυση τόσο των αρθρωτών όσο και των συμπεριφορικών δικτύων αποκαλύπτει σημαντικές ιδέες, αλλά δεν είναι ισοδύναμες με προσωπικά δίκτυα [20].

9. Ψηφιακό χάσμα: Η απόκτηση πρόσβασης σε μεγάλα δεδομένα είναι ένας από τους πιο σημαντικούς περιορισμούς. Οι εταιρείες δεδομένων και οι εταιρείες μέσων κοινωνικής δικτύωσης έχουν πρόσβαση σε μεγάλα κοινωνικά δεδομένα. Λίγες εταιρείες αποφασίζουν ποιος μπορεί να έχει πρόσβαση στα δεδομένα και σε ποιο βαθμό. Λίγοι πωλούν το δικαίωμα πρόσβασης έναντι υψηλών χρεώσεων, ενώ άλλοι προσφέρουν ένα μέρος των συνόλων δεδομένων σε ερευνητές. Αυτό έχει ως αποτέλεσμα το "Digital Divide" στη σφαίρα των μεγάλων δεδομένων: Big Data πλούσια και Big Data poor. Υπάρχουν τρεις τύποι ατόμων και οργανισμών σε αυτό το πεδίο μεγάλων δεδομένων. Πρώτον, οι άνθρωποι που δημιουργούν δεδομένα και αυτό περιλαμβάνει ολόκληρη την κοινότητα που χρησιμοποιεί web, mobile ή άλλες τεχνολογίες. Δεύτερον, άτομα που συλλέγουν δεδομένα και αυτή η ομάδα είναι μικρή. Τρίτον, άτομα που μπορούν να αναλύσουν τα δεδομένα. Αυτή η ομάδα είναι η μικρότερη και η πιο προνομιακή. Η περιορισμένη πρόσβαση σε μεγάλα κοινωνικά δεδομένα εξηγεί τη δυσκολία διεξαγωγής σύγχρονες κοινωνικές επιστήμες προσανατολισμένες στα δεδομένα και σύγχρονη έρευνα ανθρωπιστικών επιστημών με γνώμονα τα δεδομένα [22].

10. Σφάλματα δεδομένων: Με την αύξηση της ανάπτυξης της τεχνολογίας πληροφοριών, δημιουργείται τεράστιος όγκος δεδομένων. Με την έλευση του υπολογιστικού νέφους για αποθήκευση και ανάκτηση δεδομένων, υπάρχει ανάγκη να χρησιμοποιηθούν τα μεγάλα δεδομένα. Τα μεγάλα σύνολα δεδομένων από πηγές διαδικτύου είναι επιρρεπή σε σφάλματα και απώλειες, επομένως είναι αναξιόπιστα. Η πηγή των δεδομένων θα πρέπει να γίνει κατανοητή ότι ελαχιστοποιεί τα σφάλματα που προκαλούνται κατά τη χρήση πολλαπλών συνόλων δεδομένων. Οι ιδιότητες και τα όρια του συνόλου δεδομένων πρέπει να κατανοηθούν πριν από την ανάλυση για να αποφευχθεί ή να εξηγηθεί η μεροληψία στην ερμηνεία των δεδομένων [20].

1.8 Ερευνητικοί τομείς με τεχνολογικές βελτιώσεις

Η ανάλυση μεγάλων δεδομένων κερδίζει τόση προσοχή αυτές τις μέρες, αλλά υπάρχει μια σειρά από ερευνητικά προβλήματα που πρέπει ακόμη να αντιμετωπιστούν.

1. Αποθήκευση και ανάκτηση εικόνων, ήχου και βίντεο: Τα πολυδιάστατα δεδομένα θα πρέπει να ενσωματώνονται με αναλυτικά στοιχεία σε μεγάλα δεδομένα, επομένως μπορούν να διερευνηθούν μοντέλα αναπαράστασης στη μνήμη που βασίζονται σε πίνακα. Η ενσωμάτωση μοντέλων πολυδιάστατων δεδομένων σε μεγάλα δεδομένα απαιτεί τη βελτίωση της γλώσσας ερωτημάτων HiveQL με πολυδιάστατες επεκτάσεις [23]. Με τον πολλαπλασιασμό των έξυπνων τηλεφώνων, οι

εικόνες, οι ήχοι και τα βίντεο δημιουργούνται με ασυνήθιστο ρυθμό. Ωστόσο, η αποθήκευση, η ανάκτηση και η επεξεργασία αυτών των μη δομημένων δεδομένων απαιτεί τεράστια έρευνα σε κάθε διάσταση.

2. Κύκλος ζωής δεδομένων: Τα περισσότερα σενάρια εφαρμογών απαιτούν απόδοση σε πραγματικό χρόνο των αναλυτικών στοιχείων μεγάλων δεδομένων. Υπάρχει ανάγκη να καθοριστεί ο κύκλος ζωής των δεδομένων, η αξία που μπορεί να προσφέρει και η υπολογιστική διαδικασία για να γίνει η διαδικασία ανάλυσης σε πραγματικό χρόνο, αυξάνοντας έτσι την αξία της ανάλυσης. Τα μεγάλα δεδομένα δεν είναι πάντα καλύτερα, επομένως μπορούν να αναπτυχθούν κατάλληλες τεχνικές φιλτραρίσματος δεδομένων για να διασφαλιστεί η ορθότητα των δεδομένων. Ένα άλλο μεγάλο ζήτημα σχετίζεται με τη διαθεσιμότητα δεδομένων που είναι πλήρη και αξιόπιστα. Στις περισσότερες περιπτώσεις, τα δεδομένα είναι αραιά και δεν παρουσιάζουν σαφή κατανομή, οδηγώντας σε παραπλανητικά συμπεράσματα. Μια μέθοδος για να ξεπεραστούν αυτά τα προβλήματα χρειάζεται την κατάλληλη προσοχή και μερικές φορές ο χειρισμός μη ισορροπημένων συνόλων δεδομένων οδηγεί σε μεροληπτικά συμπεράσματα [6].

3. Υπολογισμοί μεγάλων δεδομένων: Εκτός από τα τρέχοντα παραδείγματα μεγάλων δεδομένων όπως το Map-Reduce, διερευνώνται και άλλα παραδείγματα όπως το YarcData (Big Data Graph Analytics) και το σύμπλεγμα Υπολογιστών Υψηλής Απόδοσης (HPCC εξερευνά εναλλακτικές λύσεις Hadoop) [24].

4. Οπτικοποίηση δεδομένων υψηλών διαστάσεων: Η οπτικοποίηση βοηθά στην ανάλυση αποφάσεων σε κάθε βήμα της ανάλυσης δεδομένων. Τα θέματα οπτικοποίησης εξακολουθούν να αποτελούν μέρος της έρευνας αποθήκευσης δεδομένων. Υπάρχει ένα πεδίο έρευνας για εργαλεία οπτικοποίησης για δεδομένα υψηλών διαστάσεων [23].

5. Ανάπτυξη Αλγορίθμων για το χειρισμό δεδομένων συγκεκριμένου τομέα: Οι αλγόριθμοι μηχανικής μάθησης αναπτύσσονται για να ικανοποιούν τις γενικές απαιτήσεις για την επεξεργασία δεδομένων. Ωστόσο, δεν μπορεί να αντικαταστήσει τις συγκεκριμένες απαιτήσεις του τομέα και θα χρειαστούν συγκεκριμένοι αλγόριθμοι για να αποκτήσετε πληροφορίες από τον επιθυμητό κλάδο.

6. Αλγόριθμοι επεξεργασίας σε πραγματικό χρόνο: Ο ρυθμός με τον οποίο παράγονται τα δεδομένα και οι προσδοκίες από αυτούς τους αλγόριθμους ενδέχεται να μην ικανοποιηθούν, εάν δεν επιτευχθεί η επιθυμητή χρονική καθυστέρηση.

7. Αποτελεσματικές συσκευές αποθήκευσης: Η ζήτηση για αποθήκευση ψηφιακών πληροφοριών αυξάνεται συνεχώς. Η αγορά και η χρήση διαθέσιμων συσκευών αποθήκευσης δεν μπορεί να καλύψει αυτή τη ζήτηση. Η έρευνα για την ανάπτυξη αποτελεσματικής συσκευής αποθήκευσης που μπορεί να αντικαταστήσει την ανάγκη για συστήματα HDFS που είναι ανεκτικά σε σφάλματα μπορεί να βελτιώσει τη δραστηριότητα επεξεργασίας δεδομένων και να αντικαταστήσει την ανάγκη για επίπεδο διαχείρισης λογισμικού [25].

8. Διαστάσεις κοινωνικών προοπτικών: Είναι σημαντικό να κατανοήσουμε ότι οποιαδήποτε τεχνολογία μπορεί να αποφέρει ταχύτερα αποτελέσματα, ωστόσο, εναπόκειται στους υπεύθυνους λήψης αποφάσεων να τη χρησιμοποιήσουν με σύνεση. Αυτά τα αποτελέσματα μπορεί να έχουν πολλές κοινωνικές και πολιτιστικές επιπτώσεις και μερικές φορές να οδηγούν σε κυνισμό προς τις διαδικτυακές πλατφόρμες. Υπάρχουν ερωτηματικά εάν τα δεδομένα αναζήτησης μεγάλης κλίμακας θα βοηθούσαν στη δημιουργία καλύτερων εργαλείων και υπηρεσιών ή θα οδηγήσουν σε εισβολές στο απόρρητο και επεμβατικό μάρκετινγκ. Εάν η ανάλυση δεδομένων θα βοηθούσε στην κατανόηση της διαδικτυακής συμπεριφοράς, των κοινοτήτων και των πολιτικών κινημάτων ή θα χρησιμοποιηθεί για την παρακολούθηση διαδηλωτών και την καταστολή της ελευθερίας του λόγου [26].

ΚΕΦΑΛΑΙΟ 2

Ανάλυση των δεδομένων

Σε αυτό το κεφάλαιο, θα περιγράψουμε όλες τις έννοιες και τις διαδικασίες που συνθέτουν τον κλάδο της ανάλυσης δεδομένων. Οι έννοιες που αναφέρονται σε αυτό το κεφάλαιο θα αποτελέσουν χρήσιμο υπόβαθρο για τα επόμενα κεφάλαια, όπου αυτές οι έννοιες και οι διαδικασίες θα εφαρμοστούν με τη μορφή κώδικα Python, μέσω της χρήσης διαφόρων βιβλιοθηκών.

2.1 Λίγα λόγια για την ανάλυση δεδομένων

Από τότε που δημιουργήθηκε η πληροφορική, τα δεδομένα χρειάζονταν πάντα μεθόδους για την ανάλυσή τους και την παραγωγή ουσιαστικών πληροφοριών από αυτά. Η παραγωγή μεγάλου όγκου δεδομένων και η δημιουργία αναφορών από αυτά ήταν η παραδοσιακή προσέγγιση για τη λειτουργία και την επέκταση των επιχειρήσεων.

Σε έναν κόσμο που συγκεντρώνεται όλο και περισσότερο γύρω από την τεχνολογία των πληροφοριών, παράγονται και αποθηκεύονται καθημερινά τεράστιες ποσότητες δεδομένων. Συχνά τα δεδομένα αυτά προέρχονται από συστήματα αυτόματης ανίχνευσης, αισθητήρες και επιστημονικά όργανα ή τα παράγετε καθημερινά και ασυνείδητα κάθε φορά που κάνετε ανάληψη από την τράπεζα ή αγορά, όταν καταγράφετε σε διάφορα ιστολόγια ή ακόμη και όταν αναρτάτε σε κοινωνικά δίκτυα.

Ποια είναι όμως τα δεδομένα; Τα δεδομένα στην πραγματικότητα δεν είναι πληροφορίες, τουλάχιστον όσον αφορά τη μορφή τους. Στην άμορφη ροή των bytes, με την πρώτη ματιά είναι δύσκολο να κατανοήσει κανείς την ουσία τους, αν δεν είναι αυστηρά ο αριθμός, η λέξη ή ο χρόνος που αναφέρουν. Οι πληροφορίες είναι στην πραγματικότητα το αποτέλεσμα της επεξεργασίας, η οποία λαμβάνοντας υπόψη ένα συγκεκριμένο σύνολο δεδομένων, εξάγει κάποια συμπεράσματα που μπορούν να χρησιμοποιηθούν με διάφορους τρόπους. Αυτή η διαδικασία εξαγωγής πληροφοριών από τα ακατέργαστα δεδομένα είναι ακριβώς η ανάλυση δεδομένων [27].

Ο σκοπός της ανάλυσης δεδομένων είναι ακριβώς η εξαγωγή πληροφοριών που δεν μπορούν να εξαχθούν εύκολα, αλλά που, όταν γίνουν κατανοητές, οδηγούν στη δυνατότητα διεξαγωγής μελετών σχετικά με τους μηχανισμούς των συστημάτων που τα παρήγαγαν, επιτρέποντας έτσι τη δυνατότητα πρόβλεψης των πιθανών αντιδράσεων αυτών των συστημάτων και της εξέλιξής τους στο χρόνο.

Ξεκινώντας από μια απλή μεθοδολογική προσέγγιση για την προστασία των δεδομένων, η ανάλυση δεδομένων έχει γίνει ένας πραγματικός κλάδος που οδηγεί στην ανάπτυξη πραγματικών μεθοδολογιών που δημιουργούν μοντέλα. Το μοντέλο είναι στην πραγματικότητα η μετάφραση σε μαθηματική μορφή ενός συστήματος που τίθεται υπό μελέτη. Μόλις υπάρξει μια μαθηματική ή λογική μορφή ικανή να περιγράψει τις αποκρίσεις του συστήματος υπό διαφορετικά επίπεδα ακρίβειας, μπορείτε στη συνέχεια να κάνετε προβλέψεις σχετικά με την εξέλιξη ή την απόκρισή του σε ορισμένες εισροές. Συνεπώς, ο στόχος της ανάλυσης δεδομένων δεν είναι το μοντέλο, αλλά η καλή προγνωστική του ικανότητα.

Η προγνωστική ισχύς ενός μοντέλου εξαρτάται όχι μόνο από την ποιότητα των τεχνικών μοντελοποίησης αλλά και από την ικανότητα επιλογής ενός καλού συνόλου δεδομένων πάνω στο οποίο θα βασιστεί ολόκληρη η ανάλυση των δεδομένων. Έτσι, η αναζήτηση δεδομένων, η εξαγωγή τους και η μετέπειτα προετοιμασία τους, ενώ αποτελούν προκαταρκτικές δραστηριότητες μιας ανάλυσης, ανήκουν επίσης στην ίδια την ανάλυση δεδομένων, λόγω της σημασίας τους για την επιτυχία των αποτελεσμάτων.

Μέχρι στιγμής έχουμε μιλήσει για τα δεδομένα, τον χειρισμό τους και την επεξεργασία τους μέσω διαδικασιών υπολογισμού. Παράλληλα με όλα τα στάδια επεξεργασίας της ανάλυσης των δεδομένων, έχουν αναπτυχθεί διάφορες μέθοδοι οπτικοποίησης των δεδομένων. Στην

πραγματικότητα, για την κατανόηση των δεδομένων, τόσο μεμονωμένα όσο και ως προς τον ρόλο που διαδραματίζουν στο σύνολο των δεδομένων, δεν υπάρχει καλύτερο σύστημα από την ανάπτυξη τεχνικών γραφικής αναπαράστασης ικανών να μετατρέπουν τις πληροφορίες, - μερικές φορές σιωπηρά κρυμμένων- σε αριθμούς, οι οποίοι σας βοηθούν να κατανοήσετε ευκολότερα το νόημά τους. Με την πάροδο των ετών έχουν αναπτυχθεί πολλοί διαφορετικοί τρόπους απεικόνισης δεδομένων.

Στο τέλος της ανάλυσης δεδομένων, θα έχετε ένα μοντέλο και ένα σύνολο γραφικών απεικονίσεων και στη συνέχεια θα είστε σε θέση να προβλέψετε τις αποκρίσεις του υπό μελέτη συστήματος και μετά από αυτό θα προχωρήσετε στη φάση της δοκιμής. Το μοντέλο θα δοκιμαστεί χρησιμοποιώντας ένα άλλο σύνολο δεδομένων για το οποίο γνωρίζουμε την απόκριση του συστήματος. Τα δεδομένα αυτά, ωστόσο, δεν χρησιμοποιούνται για τον ορισμό του μοντέλου πρόβλεψης. Ανάλογα με την ικανότητα του μοντέλου να αναπαράγει τις πραγματικές παρατηρούμενες αποκρίσεις, θα έχετε έναν υπολογισμό σφάλματος και μια γνώση της εγκυρότητας του μοντέλου και των ορίων λειτουργίας του.

Τα αποτελέσματα αυτά μπορούν να συγκριθούν με οποιαδήποτε άλλα μοντέλα για να γίνει κατανοητό αν το νέο μοντέλο που δημιουργήθηκε είναι πιο αποτελεσματικό από τα υπάρχοντα. Μόλις το αξιολογήσετε αυτό, μπορείτε να προχωρήσετε στην τελευταία φάση των δεδομένων ανάλυση - ή ανάπτυξη. Αυτή συνίσταται στην εφαρμογή των αποτελεσμάτων που παράγονται από την ανάλυση δεδομένων, δηλαδή στην εφαρμογή των αποφάσεων που πρέπει να ληφθούν με βάση τις προβλέψεις που παράγει το μοντέλο και τους κινδύνους που θα προβλέψει επίσης μια τέτοια απόφαση.

Η ανάλυση δεδομένων είναι ένας κλάδος που είναι κατάλληλος για πολλές επαγγελματικές δραστηριότητες. Έτσι, η γνώση του τι είναι και πώς μπορεί να εφαρμοστεί στην πράξη θα είναι σημαντική για την εδραίωση των αποφάσεων που πρέπει να ληφθούν. Θα μας επιτρέψει να ελέγξουμε υποθέσεις και να κατανοήσουμε βαθύτερα τα συστήματα που αναλύονται.

2.2 Αναλύσεις βάσεων δεδομένων

Ο πιο βασικός τύπος είναι η ανάλυση βάσεων δεδομένων, όπου τα δεδομένα αποθηκεύονται με τη σταθερή μορφή γραμμών και στηλών σε πίνακες της βάσης δεδομένων. Οι προγραμματιστές θα εκτελούσαν ερωτήματα σε αυτούς τους πίνακες για να λάβουν τα επιθυμητά αποτελέσματα και θα χρησιμοποιούσαν ένα άλλο εργαλείο για να παρουσιάσουν αυτά τα δεδομένα με πιο λογικό τρόπο. Τα εργαλεία αναφοράς θα βοηθούσαν επίσης με γραφήματα και ούτω καθεξής.

Η ανάλυση βάσεων δεδομένων συνεχίζει να έχει σημασία. Παρόλο που αναδύονται νέες αναλυτικές διαστάσεις, η ανάλυση βάσεων δεδομένων δεν θα χάσει ποτέ τη γοητεία της, καθώς είναι περισσότερο προς την κατεύθυνση της ανάλυσης των στατικών δεδομένων και της δημιουργίας αναφορών από αυτά.

Καθώς οι ανάγκες αυξάνονται, εμφανίζονται όλο και πιο κλιμακούμενες και υψηλών επιδόσεων βάσεις δεδομένων για να τις χειριστούν. Οι μεγάλες αναλυτικές βάσεις δεδομένων κατανέμονται σε πολλαπλούς κόμβους για να εξαλειφθούν και οι περιορισμοί των απαιτήσεων υπολογισμού.

2.3 Αναλύσεις σε πραγματικό χρόνο

Οι επιχειρήσεις δεν μπορούν να βασίζονται μόνο στις επιδόσεις του παρελθόντος για να σχεδιάσουν το μέλλον τους. Πρέπει να καταγράψουν τις τρέχουσες τάσεις και τις ανάγκες των καταναλωτών σήμερα. Οι αναλύσεις έχουν αλλάξει από την ανάλυση δεδομένων του παρελθόντος στην εκτέλεση αναλύσεων σε δεδομένα πραγματικού χρόνου, τα οποία παράγονται όχι μόνο από τις δομημένες εσωτερικές βάσεις δεδομένων αλλά και από μη δομημένα δεδομένα που παράγονται μέσω των μέσων κοινωνικής δικτύωσης και της συμπεριφοράς των καταναλωτών.

Σήμερα υπάρχουν διαθέσιμα εργαλεία για τη συλλογή δεδομένων από αυτές τις πηγές και τη συγκέντρωσή τους μέσω της λεγόμενης διαδικασίας "Data Conditioning" σε βάσεις δεδομένων που βοηθούν στην ανάλυσή τους. Όλα αυτά τα δεδομένα επεξεργάζονται σε πραγματικό χρόνο για να παράγουν σχεδόν άμεσα αποτελέσματα που βοηθούν τις επιχειρήσεις να εξυπηρετούν καλύτερα τους καταναλωτές τους.

Πολλοί ρωτούν αν υπάρχει πράγματι ανάγκη για ανάλυση σε πραγματικό χρόνο. Πώς θα τους βοηθούσε στην επιχείρησή τους και αξίζει το είδος της επένδυσης που χρειάζεται για να γίνει ανάλυση σε πραγματικό χρόνο;

Μπορούμε να αναφέρουμε μερικά μικρά παραδείγματα διαφορετικών ροών που έχουν χρησιμοποιήσει οι οργανισμοί για να δουν τα οφέλη. Ας ξεκινήσουμε με την υγειονομική περίθαλψη - μια μικρή συσκευή που φοριέται στη ζώνη της μέσης εγχέει τις επιθυμητές ποσότητες ινσουλίνης σε έναν διαβητικό ασθενή, ενώ παρακολουθείται το επίπεδο σακχάρου στο αίμα μέσω της συσκευής.

Φανταστείτε ότι μπαίνετε σε ένα ξενοδοχείο και το άτομο στη ρεσεψιόν γνωρίζει το όνομά σας, τα στοιχεία της κράτησης και τις προτιμήσεις σας. Ενώ το δεύτερο μέρος είναι ήδη υπό έλεγχο μέσω των συνδρομών σε κλαμπ, η κάμερα στην πόρτα τραβάει τη φωτογραφία σας, αναζητά το αρχείο και μέχρι να φτάσετε στη ρεσεψιόν, τα στοιχεία σας βρίσκονται στο γραφείο.

Κάποιος που περπατάει κοντά στο κατάστημα λαμβάνει ένα μήνυμα για πρόσθετες προσφορές στο αγαπημένο του προϊόν για τα επόμενα 30 λεπτά. Αν απορρίψετε ένα προϊόν από το ράφι, λαμβάνετε μήνυμα για πρόσθετες προσφορές στο προϊόν. Ένα win-win για το κατάστημα και τον καταναλωτή.

2.4 Τύποι Ανάλυσης Δεδομένων

Υπάρχουν τέσσερις κύριοι τύποι ανάλυσης μεγάλων δεδομένων που υποστηρίζουν και ενημερώνουν διαφορετικές επιχειρηματικές αποφάσεις.

2.4.1 Περιγραφική ανάλυση (Descriptive analytics)

Η περιγραφική ανάλυση αναφέρεται σε δεδομένα που μπορούν εύκολα να διαβαστούν και να ερμηνευτούν. Τα δεδομένα αυτά βοηθούν στη δημιουργία αναφορών και στην οπτικοποίηση πληροφοριών που μπορούν να περιγράφουν λεπτομερώς τα κέρδη και τις πωλήσεις της εταιρείας.

Παράδειγμα: Κατά τη διάρκεια της πανδημίας, μια κορυφαία φαρμακευτική εταιρεία διεξήγαγε ανάλυση δεδομένων στα γραφεία και τα ερευνητικά της εργαστήρια. Η περιγραφική ανάλυση τους βοήθησε να εντοπίσουν μη χρησιμοποιούμενους χώρους και τμήματα που ενοποιήθηκαν, εξοικονομώντας εκατομμύρια δολάρια στην εταιρεία.

2.4.2 Διαγνωστική ανάλυση (Diagnostics analytics)

Η διαγνωστική ανάλυση βοηθά τις εταιρείες να κατανοήσουν γιατί προέκυψε ένα πρόβλημα. Οι τεχνολογίες και τα εργαλεία μεγάλων δεδομένων επιτρέπουν στους χρήστες να εξορύξουν και να ανακτήσουν δεδομένα που βοηθούν στην ανάλυση ενός προβλήματος και στην αποτροπή του στο μέλλον. Παράδειγμα: Οι πωλήσεις μιας εταιρείας ένδυσης έχουν μειωθεί, παρόλο που οι πελάτες συνεχίζουν να προσθέτουν αντικείμενα στα καλάθια αγορών τους. Οι διαγνωστικές αναλύσεις βοήθησαν να γίνει κατανοητό ότι η σελίδα πληρωμής δεν λειτουργούσε σωστά για μερικές εβδομάδες.

2.4.3 Καθοδηγητική ανάλυση (Prescriptive analytics)

Η καθοδηγητική ανάλυση παρέχει λύση σε ένα πρόβλημα, βασιζόμενη στην ΤΝ και τη μηχανική μάθηση για τη συλλογή δεδομένων και τη χρήση τους για τη διαχείριση κινδύνων. Παράδειγμα: Στον τομέα της ενέργειας, οι εταιρείες κοινής ωφέλειας, οι παραγωγοί φυσικού αερίου και οι ιδιοκτήτες

αγωγών εντοπίζουν τους παράγοντες που επηρεάζουν την τιμή του πετρελαίου και του φυσικού αερίου προκειμένου να αντισταθμίσουν τους κινδύνους.

2.4.4 Η Προγνωστική ανάλυση (Predictive analytics)

Οι οργανισμοί δεν ζητούν πλέον μόνο την ανάλυση των δεδομένων τους. Τα παραδοσιακά εργαλεία επιχειρηματικής ευφυΐας (BI - Business Intelligence) το κάνουν αυτό εδώ και πολύ καιρό. Αυτό που εξερευνούν είναι να παίρνουν πιο χρήσιμες πληροφορίες από τα δεδομένα από εργαλεία οπτικοποίησης και προγνωστικής ανάλυσης για να εξερευνήσουν τα δεδομένα με νέους τρόπους και να ανακαλύψουν νέα μοτίβα [28].

Η ανάλυση πρόβλεψης είναι η πρακτική της εξαγωγής πληροφοριών από υπάρχοντα δεδομένα για τον προσδιορισμό προτύπων και την πρόβλεψη μελλοντικών αποτελεσμάτων και τάσεων. Βοηθά στην πρόβλεψη του τι μπορεί να συμβεί στο μέλλον με αποδεκτό επίπεδο αξιοπιστίας και περιλαμβάνει σενάρια "τι θα συμβεί αν" και αξιολόγηση κινδύνου.

Με τη εφαρμογή τους στις επιχειρήσεις, τα μοντέλα προγνωστικής ανάλυσης χρησιμοποιούνται για την ανάλυση τρεχόντων δεδομένων και ιστορικών γεγονότων για την καλύτερη κατανόηση των πελατών, των προϊόντων και των συνεργατών και για τον εντοπισμό πιθανών κινδύνων και ευκαιριών για μια εταιρεία. Χρησιμοποιεί διάφορες τεχνικές, συμπεριλαμβανομένης της εξόρυξης δεδομένων, της στατιστικής μοντελοποίησης και της μηχανικής μάθησης, για να βοηθήσει τους αναλυτές να κάνουν επιχειρηματικές προβλέψεις.

Όλες οι επιχειρήσεις λειτουργούν με κίνδυνο. Κίνδυνος του τρόπου διαχείρισης των επιχειρήσεων. Κάθε απόφαση που λαμβάνει ένας οργανισμός επηρεάζει τους κινδύνους που μπορεί να αντέξει μια επιχείρηση. Το μέσο που βασίζεται στα δεδομένα για τον υπολογισμό του κινδύνου οποιουδήποτε τύπου αρνητικού αποτελέσματος γενικά είναι η προγνωστική ανάλυση. Οι ασφαλιστικές εταιρείες το έχουν χρησιμοποιήσει πολύ καλά αυτό, ενισχύοντας τις πρακτικές τους με την ενσωμάτωση της προγνωστικής ανάλυσης προκειμένου να βελτιώσουν τις αποφάσεις τιμολόγησης και επιλογής.

Οι αναλογιστικές μέθοδοι που επιτρέπουν σε μια ασφαλιστική εταιρεία να διεξάγει την κύρια δραστηριότητά της εκτελούν την ίδια ακριβώς λειτουργία με τα μοντέλα πρόβλεψης, βαθμολογώντας τους πελάτες με βάση την πιθανότητα θετικών ή αρνητικών αποτελεσμάτων. Η προγνωστική μοντελοποίηση βελτιώνει τις συνήθεις αναλογιστικές μεθόδους ενσωματώνοντας πρόσθετους αναλυτικούς αυτοματισμούς και γενικεύοντας σε ένα ευρύτερο σύνολο μεταβλητών πελατών.

2.5 Ανάλυση μεγάλων δεδομένων

Έχουμε συζητήσει λεπτομερώς για τα μεγάλα δεδομένα που δεν είναι μόνο ο μεγάλος όγκος αλλά και η ποικιλία των συνόλων δεδομένων και η ταχύτητα με την οποία παράγονται. Τώρα, πρέπει να κατανοήσουμε τι σημαίνει **Big Data Analytics**.

Με απλά λόγια, είναι η διαδικασία με την οποία μπορούμε να εξετάσουμε αυτά τα μεγάλα σύνολα δεδομένων για να αποκαλύψουμε κρυμμένα μοτίβα, άγνωστες συσχετίσεις, τάσεις της αγοράς, προτιμήσεις πελατών κ.λπ. για να εξάγουμε χρήσιμες πληροφορίες. Είναι η χρήση προηγμένων αναλυτικών τεχνικών έναντι πολύ μεγάλων συνόλων δεδομένων που περιλαμβάνουν συλλογή δεδομένων σε πραγματικό χρόνο από διάφορες δομημένες/αδόμητες πηγές δεδομένων [29].

Οι πληροφορίες χωρίς διορατικότητα είναι ένας ανεκμετάλλευτος πόρος. Αντίθετα, η ανάλυση χωρίς στέρεη βάση πληροφοριών είναι πιθανό να οδηγήσει σε κακές αποφάσεις. Η ανάλυση μεγάλων δεδομένων είναι η εφαρμογή αναλυτικών δυνατοτήτων σε τεράστια, ποικίλα ή ταχέως μεταβαλλόμενα σύνολα δεδομένων. Είναι η εφαρμογή των αναλυτικών δυνατοτήτων σε συνδυασμό με το αυξημένο εύρος και το πλαίσιο των μεγάλων δεδομένων, ιδίως όταν συγχωνεύονται με παραδοσιακά δομημένα σύνολα δεδομένων.

Η ανάλυση μεγάλων δεδομένων επιτρέπει στους αναλυτές, τους ερευνητές και τους επιχειρηματικούς χρήστες να λαμβάνουν καλύτερες και ταχύτερες αποφάσεις χρησιμοποιώντας δεδομένα που προηγουμένως ήταν απρόσιτα ή άχρηστα. Χρησιμοποιώντας προηγμένες τεχνικές ανάλυσης, όπως η ανάλυση κειμένου, η μηχανική μάθηση, η προγνωστική ανάλυση, η εξόρυξη δεδομένων, η στατιστική και η επεξεργασία φυσικής γλώσσας, οι επιχειρήσεις μπορούν να αναλύουν προηγουμένως ανεκμετάλλευτες πηγές δεδομένων ανεξάρτητα ή μαζί με τα υπάρχοντα επιχειρησιακά δεδομένα τους για να αποκτήσουν νέες γνώσεις με αποτέλεσμα σημαντικά καλύτερες και ταχύτερες αποφάσεις.

Γνώσεις και δεξιότητες του αναλυτή δεδομένων

Η ανάλυση δεδομένων είναι βασικά ένας κλάδος κατάλληλος για τη μελέτη προβλημάτων που μπορεί να εμφανιστούν σε διάφορους τομείς εφαρμογών. Επιπλέον, στις διαδικασίες ανάλυσης δεδομένων έχετε πολλά εργαλεία και μεθοδολογίες που απαιτούν καλή γνώση της πληροφορικής, των μαθηματικών και στατιστικών εννοιών [30].

Έτσι, ένας καλός αναλυτής δεδομένων πρέπει να είναι σε θέση να κινείται και να ενεργεί σε πολλούς διαφορετικούς επιστημονικούς τομείς. Πολλοί από αυτούς τους κλάδους αποτελούν τη βάση των μεθόδων ανάλυσης δεδομένων και η επάρκεια σε αυτούς είναι σχεδόν απαραίτητη.

Η γνώση άλλων επιστημονικών κλάδων είναι απαραίτητη ανάλογα με τον τομέα εφαρμογής και μελέτης του συγκεκριμένου έργου ανάλυσης δεδομένων που πρόκειται να αναλάβετε και, γενικότερα, η επαρκής εμπειρία σε αυτούς τους τομείς μπορεί απλώς να σας βοηθήσει να κατανοήσετε καλύτερα τα ζητήματα και τον τύπο των δεδομένων που απαιτούνται για να ξεκινήσετε την ανάλυση.

Συχνά, όσον αφορά σημαντικά προβλήματα ανάλυσης δεδομένων, είναι απαραίτητη η ύπαρξη μιας διεπιστημονικής ομάδας εμπειρογνωμόνων που αποτελείται από μέλη τα οποία είναι όλα σε θέση να συμβάλουν με τον καλύτερο δυνατό τρόπο στους αντίστοιχους τομείς των αρμοδιοτήτων τους. Όσον αφορά τα μικρότερα προβλήματα, ένας καλός αναλυτής πρέπει να είναι σε θέση να αναγνωρίζει τα προβλήματα που προκύπτουν κατά την ανάλυση δεδομένων, να ερευνά για να διαπιστώσει ποιες ειδικότητες και δεξιότητες είναι απαραίτητες για την επίλυση του προβλήματος, να μελετά αυτές τις ειδικότητες και ίσως ακόμη και να ρωτά τους πιο γνώστες του τομέα. Εν ολίγοις, ο αναλυτής πρέπει να είναι σε θέση να γνωρίζει πώς να αναζητά όχι μόνο δεδομένα, αλλά και πληροφορίες για τον τρόπο αντιμετώπισής τους.

Επιστήμη υπολογιστών

Η γνώση της επιστήμης των υπολογιστών αποτελεί βασική προϋπόθεση για κάθε αναλυτή δεδομένων. Στην πραγματικότητα, μόνο όποιος έχει καλή γνώση και εμπειρία στην Επιστήμη των Υπολογιστών είναι σε θέση να διαχειριστεί αποτελεσματικά τα απαραίτητα εργαλεία για την ανάλυση δεδομένων. Στην πραγματικότητα, όλα τα βήματα που αφορούν την ανάλυση δεδομένων περιλαμβάνουν τη χρήση της τεχνολογίας των υπολογιστών, όπως λογισμικό υπολογισμού (όπως Matlab κ.λπ.) και γλώσσες προγραμματισμού (όπως C++, Java, Python).

Ο μεγάλος όγκος δεδομένων που είναι διαθέσιμος σήμερα χάρη στην τεχνολογία των πληροφοριών απαιτεί ειδικές δεξιότητες προκειμένου να διαχειριστεί όσο το δυνατόν πιο αποτελεσματικά. Πράγματι, η έρευνα και η εξαγωγή δεδομένων απαιτούν γνώση των διαφόρων μορφών. Τα δεδομένα είναι δομημένα και αποθηκεύονται σε αρχεία ή πίνακες βάσεων δεδομένων με συγκεκριμένες μορφές. Τα αρχεία XML, JSON ή απλώς XLS ή CSV είναι πλέον οι συνήθεις μορφές για την αποθήκευση και τη συλλογή δεδομένων, ενώ πολλές εφαρμογές επιτρέπουν επίσης την ανάγνωση και τη διαχείριση των δεδομένων που είναι αποθηκευμένα σε αυτά. Για την εξαγωγή των δεδομένων που περιέχονται σε μια βάση δεδομένων, τα πράγματα δεν είναι τόσο άμεσα, αλλά

χρειάζεται να γνωρίζετε τη γλώσσα ερωτημάτων SQL ή να χρησιμοποιήσετε λογισμικό που έχει αναπτυχθεί ειδικά για την εξαγωγή δεδομένων από μια συγκεκριμένη βάση δεδομένων.

Επιπλέον, για ορισμένους συγκεκριμένους τύπους έρευνας δεδομένων, τα δεδομένα δεν είναι διαθέσιμα σε προκατεργασμένη και σαφή μορφή, αλλά υπάρχουν σε αρχεία κειμένου (έγγραφα, αρχεία καταγραφής) ή σε ιστοσελίδες, που εμφανίζονται ως διαγράμματα, μετρήσεις, αριθμός επισκεπτών ή πίνακες HTML, οι οποίοι απαιτούν ειδική τεχνική εμπειρογνωμοσύνη για την ανάλυση και την ενδεχόμενη εξαγωγή αυτών των δεδομένων (Web Scraping).

Έτσι, η γνώση της πληροφορικής είναι απαραίτητη για να γνωρίζει κανείς πώς να χρησιμοποιεί τα διάφορα εργαλεία που διαθέτει η σύγχρονη επιστήμη της πληροφορικής, όπως οι εφαρμογές και οι γλώσσες προγραμματισμού. Τα εργαλεία αυτά, με τη σειρά τους, είναι απαραίτητα για την εκτέλεση της ανάλυσης δεδομένων και της οπτικοποίησης δεδομένων.

Σκοπός της παρούσας εργασίας είναι ακριβώς να περιγράψει όλες τις απαραίτητες ενέργειες που απαιτούνται, στο μέτρο του δυνατού, σχετικά με την ανάπτυξη μεθοδολογιών ανάλυσης δεδομένων με τη χρήση της Python ως γλώσσας προγραμματισμού και εξειδικευμένων βιβλιοθηκών που συμβάλλουν αποφασιστικά στην εκτέλεση όλων των βημάτων που συνθέτουν την ανάλυση δεδομένων, από την έρευνα δεδομένων έως την εξόρυξη δεδομένων, μέχρι τη δημοσίευση των αποτελεσμάτων του προγνωστικού μοντέλου.

Μαθηματικά και Στατιστική

Η ανάλυση δεδομένων απαιτεί πολλά μαθηματικά, τα οποία μπορεί να είναι αρκετά πολύπλοκα, κατά την επεξεργασία των δεδομένων. Επομένως, η επάρκεια σε όλα αυτά είναι απαραίτητη, τουλάχιστον για να καταλάβετε τι κάνετε. Κάποια εξοικείωση με τις βασικές στατιστικές έννοιες είναι επίσης απαραίτητη, επειδή όλες οι μέθοδοι που εφαρμόζονται κατά την ανάλυση και ερμηνεία των δεδομένων βασίζονται σε αυτές τις έννοιες. Όπως ακριβώς μπορείτε να πείτε ότι η επιστήμη των υπολογιστών σας παρέχει τα εργαλεία για την ανάλυση δεδομένων, έτσι μπορείτε να πείτε ότι η στατιστική παρέχει τις έννοιες που αποτελούν τη βάση της ανάλυσης δεδομένων.

Πολλά είναι τα εργαλεία και οι μέθοδοι που παρέχει αυτός ο κλάδος στον αναλυτή και η καλή γνώση του πώς να τα χρησιμοποιεί καλύτερα απαιτεί πολυετή εμπειρία. Μεταξύ των συνηθέστερα χρησιμοποιούμενων στατιστικών τεχνικών στην ανάλυση δεδομένων είναι:

- Μέθοδοι Bayesian
- Παλινδρόμηση
- Ομαδοποίηση

Έχοντας να αντιμετωπίσετε αυτές τις περιπτώσεις, θα ανακαλύψετε πώς τα μαθηματικά και η στατιστική συνδέονται στενά μεταξύ τους, αλλά χάρη στις ειδικές βιβλιοθήκες Python, θα έχετε τη δυνατότητα να τις διαχειρίζεστε και να τις χειρίζεστε.

Μηχανική μάθηση και τεχνητή νοημοσύνη

Ένα από τα πιο προηγμένα εργαλεία που εμπίπτουν στην ανάλυση δεδομένων είναι η μηχανική μάθηση. Στην πραγματικότητα, παρά την οπτικοποίηση δεδομένων και τις τεχνικές όπως η ομαδοποίηση και η παλινδρόμηση, οι οποίες θα πρέπει να μας βοηθήσουν σημαντικά στην εύρεση πληροφοριών σχετικά με το σύνολο δεδομένων μας, κατά τη διάρκεια αυτής της φάσης της έρευνας, μπορεί συχνά να προτιμάτε να χρησιμοποιείτε ειδικές διαδικασίες οι οποίες είναι ιδιαίτερα εξειδικευμένες στην αναζήτηση μοτίβων μέσα στο σύνολο δεδομένων.

Η μηχανική μάθηση είναι ένας κλάδος που χρησιμοποιεί μια ολόκληρη σειρά διαδικασιών και αλγορίθμων που αναλύουν τα δεδομένα προκειμένου να αναγνωρίσουν μοτίβα, συστάδες ή τάσεις

και στη συνέχεια να εξάγουν χρήσιμες πληροφορίες για την ανάλυση δεδομένων με έναν εντελώς αυτοματοποιημένο τρόπο.

Ο κλάδος αυτός καθίσταται όλο και περισσότερο θεμελιώδες εργαλείο της ανάλυσης δεδομένων και, ως εκ τούτου, η γνώση του, τουλάχιστον σε γενικές γραμμές, είναι θεμελιώδους σημασίας για τον αναλυτή δεδομένων.

Επαγγελματικά πεδία εφαρμογής

Ένα άλλο πολύ σημαντικό σημείο είναι επίσης ο τομέας αρμοδιότητας από τον οποίο προέρχονται τα δεδομένα (βιολογία, φυσική, χρηματοοικονομικά, δοκιμές υλικών, στατιστικές για τον πληθυσμό κ.λπ.). Στην πραγματικότητα, παρόλο που ο αναλυτής έχει εξειδικευμένη προετοιμασία στον τομέα της στατιστικής, πρέπει επίσης να είναι σε θέση να εμβαθύνει στον τομέα εφαρμογής και/ή να τεκμηριώσει την πηγή των δεδομένων, με στόχο την αντίληψη και την καλύτερη κατανόηση των μηχανισμών που παρήγαγαν τα δεδομένα. Στην πραγματικότητα, τα δεδομένα δεν είναι απλές συμβολοσειρές ή αριθμοί, αλλά είναι η έκφραση, ή μάλλον το μέτρο, κάθε παραμέτρου που παρατηρείται. Έτσι, η καλύτερη κατανόηση του πεδίου εφαρμογής από το οποίο προέρχονται τα δεδομένα μπορεί να βελτιώσει την ερμηνεία τους. Συχνά, ωστόσο, αυτό είναι πολύ δαπανηρό για έναν αναλυτή δεδομένων, ακόμη και με τις καλύτερες προθέσεις, και γι' αυτό είναι καλή πρακτική να βρείτε συμβούλους ή πρόσωπα-κλειδιά στα οποία μπορείτε να θέσετε τις σωστές ερωτήσεις.

Κατανόηση της φύσης των δεδομένων

Το αντικείμενο μελέτης της ανάλυσης δεδομένων είναι βασικά τα δεδομένα. Τα δεδομένα, λοιπόν, θα είναι οι βασικοί παράγοντες σε όλες τις διαδικασίες της ανάλυσης δεδομένων. Αποτελούν την πρώτη ύλη προς επεξεργασία και χάρη στην επεξεργασία και την ανάλυσή τους είναι δυνατόν να εξαχθούν ποικίλες πληροφορίες προκειμένου να αυξηθεί το επίπεδο γνώσης του υπό μελέτη συστήματος, δηλαδή εκείνου από το οποίο προήλθαν τα δεδομένα.

Όταν τα δεδομένα γίνονται πληροφορίες

Τα δεδομένα είναι τα γεγονότα που καταγράφονται στον κόσμο. Οτιδήποτε μπορεί να μετρηθεί ή ακόμη και να κατηγοριοποιηθεί μπορεί να μετατραπεί σε δεδομένα. Αφού συλλεχθούν, τα δεδομένα αυτά μπορούν να μελετηθούν και να αναλυθούν τόσο για να κατανοηθεί η φύση των γεγονότων όσο και πολύ συχνά για να γίνουν προβλέψεις ή τουλάχιστον για να ληφθούν τεκμηριωμένες αποφάσεις.

Όταν η πληροφορία γίνεται γνώση

Μπορείτε να μιλάτε για γνώση όταν οι πληροφορίες μετατρέπονται σε ένα σύνολο κανόνων που σας βοηθούν να κατανοήσετε καλύτερα ορισμένους μηχανισμούς και, κατά συνέπεια, να κάνετε προβλέψεις για την εξέλιξη ορισμένων γεγονότων.

Τύποι δεδομένων

Τα δεδομένα μπορούν να χωριστούν σε κατηγορίες:

- Κατηγορικά
- Ονομαστικά
- Ordinal
- Αριθμητικά
- Διακριτά
- Συνεχή

Τα κατηγορικά δεδομένα είναι τιμές ή παρατηρήσεις που μπορούν να χωριστούν σε ομάδες ή κατηγορίες. Υπάρχουν δύο τύποι κατηγορικών τιμών: οι ονομαστικές και οι τακτικές. Μια

ονομαστική μεταβλητή δεν έχει εγγενή τάξη που να προσδιορίζεται στην κατηγορία της. Μια τακτική μεταβλητή αντίθετα έχει μια προκαθορισμένη σειρά.

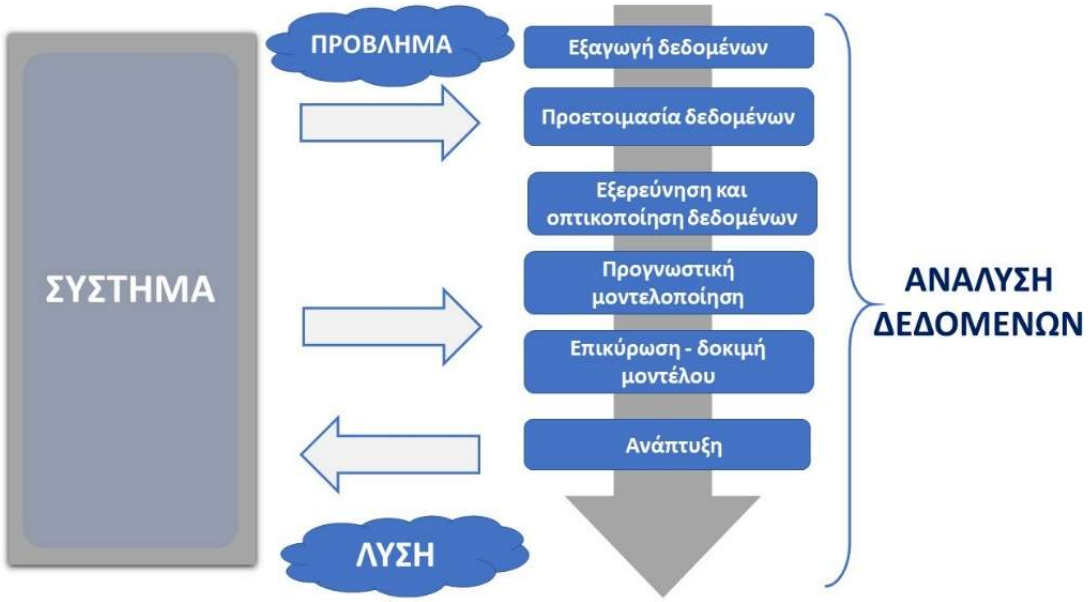
Τα αριθμητικά δεδομένα είναι τιμές ή παρατηρήσεις που προέρχονται από μετρήσεις. Υπάρχουν δύο τύποι διαφορετικών αριθμητικών τιμών: οι διακριτοί και οι συνεχείς αριθμοί. Οι διακριτές τιμές είναι τιμές που μπορούν να μετρηθούν και που είναι διακριτές και διαχωρισμένες μεταξύ τους. Οι συνεχείς τιμές, από την άλλη πλευρά, είναι τιμές που παράγονται από μετρήσεις ή παρατηρήσεις που λαμβάνουν οποιαδήποτε τιμή εντός ενός καθορισμένου εύρους.

2.6 Η διαδικασία ανάλυσης δεδομένων

Η ανάλυση δεδομένων μπορεί να περιγραφεί ως μια διαδικασία που αποτελείται από διάφορα βήματα κατά τα οποία τα ακατέργαστα δεδομένα μετασχηματίζονται και υποβάλλονται σε επεξεργασία προκειμένου να παραχθούν οπτικοποιήσεις δεδομένων και να μπορούν να γίνουν προβλέψεις χάρη σε ένα μαθηματικό μοντέλο που βασίζεται στα συλλεχθέντα δεδομένα. Στη συνέχεια, η ανάλυση δεδομένων δεν είναι τίποτα περισσότερο από μια ακολουθία βημάτων, καθένα από τα οποία διαδραματίζει βασικό ρόλο για τα επόμενα [31]. Έτσι, η ανάλυση δεδομένων σχεδόν σχηματοποιείται ως μια αλυσίδα διεργασιών που αποτελείται από την ακόλουθη ακολουθία σταδίων:

- Ορισμός του προβλήματος
- Εξαγωγή δεδομένων
- Καθαρισμός δεδομένων
- Μετασχηματισμός δεδομένων
- Εξερεύνηση δεδομένων
- Προβλεπτική μοντελοποίηση
- Επικύρωση/δοκιμή μοντέλου
- Οπτικοποίηση και ερμηνεία των αποτελεσμάτων
- Ανάπτυξη της λύσης

Το Σχήμα 2-1 είναι μια σχηματική αναπαράσταση όλων των διαδικασιών που εμπλέκονται στην ανάλυση δεδομένων.



Σχήμα 2-1. Η διαδικασία ανάλυσης δεδομένων

2.6.1 Ορισμός του προβλήματος

Η διαδικασία της ανάλυσης δεδομένων αρχίζει στην πραγματικότητα πολύ πριν από τη συλλογή των ακατέργαστων δεδομένων. Στην πραγματικότητα, μια ανάλυση δεδομένων ξεκινά πάντα με ένα πρόβλημα προς επίλυση, το οποίο πρέπει να οριστεί.

Το πρόβλημα ορίζεται μόνο αφού έχετε εστιάσει καλά το σύστημα που θέλετε να μελετήσετε. Αυτό μπορεί να είναι ένας μηχανισμός, μια εφαρμογή ή μια διαδικασία γενικά. Γενικά η μελέτη αυτή μπορεί να αποσκοπεί στην καλύτερη κατανόηση της λειτουργίας του, αλλά ειδικότερα η μελέτη θα σχεδιαστεί για την κατανόηση των αρχών της συμπεριφοράς του, ώστε να είναι σε θέση να κάνει προβλέψεις ή να κάνει επιλογές.

Το βήμα του ορισμού και η αντίστοιχη τεκμηρίωση του επιστημονικού προβλήματος ή της επιχείρησης είναι αμφότερα πολύ σημαντικά για να επικεντρωθεί η όλη ανάλυση αυστηρά στην επίτευξη αποτελεσμάτων. Στην πραγματικότητα, μια ολοκληρωμένη ή εξαντλητική μελέτη του συστήματος είναι μερικές φορές πολύπλοκη και δεν έχετε πάντα αρκετές πληροφορίες για να ξεκινήσουμε. Έτσι, ο ορισμός του προβλήματος και κυρίως ο σχεδιασμός του μπορεί να καθορίσει με μοναδικό τρόπο τις κατευθυντήριες γραμμές που πρέπει να ακολουθηθούν για ολόκληρο το έργο.

Αφού οριστεί και τεκμηριωθεί το πρόβλημα, μπορείτε να προχωρήσετε στον προγραμματισμό του έργου της ανάλυσης δεδομένων. Ο σχεδιασμός είναι απαραίτητος για να κατανοήσετε ποιοι επαγγελματίες και πόροι είναι απαραίτητοι για την ικανοποίηση των απαιτήσεων ώστε να πραγματοποιηθεί το έργο όσο το δυνατόν πιο αποτελεσματικά. Έτσι, θα εξετάσετε τα ζητήματα της περιοχής που αφορούν την επίλυση του προβλήματος. Θα αναζητήσετε ειδικούς σε διάφορους τομείς ενδιαφέροντος και, τέλος, θα εγκαταστήσετε το λογισμικό που απαιτείται για την εκτέλεση της ανάλυσης δεδομένων.

Έτσι, κατά τη φάση του σχεδιασμού, γίνεται η επιλογή μιας αποτελεσματικής ομάδας. Γενικά, οι ομάδες αυτές πρέπει να είναι διεπιστημονικές, ώστε να επιλύουν το πρόβλημα εξετάζοντας τα δεδομένα από διαφορετικές οπτικές γωνίες. Έτσι, η επιλογή μιας καλής ομάδας είναι σίγουρα ένας από τους βασικούς παράγοντες που οδηγούν στην επιτυχία στην ανάλυση δεδομένων.

2.6.2 Εξαγωγή δεδομένων

Αφού καθοριστεί το πρόβλημα, το επόμενο βήμα είναι η απόκτηση των δεδομένων για τη διενέργεια της ανάλυσης. Τα δεδομένα πρέπει να επιλεγούν με βασικό σκοπό την κατασκευή του προγνωστικού μοντέλου και έτσι η επιλογή τους είναι καθοριστική και για την επιτυχία της ανάλυσης. Τα δεδομένα του δείγματος που συλλέγονται πρέπει να αντικατοπτρίζουν όσο το δυνατόν περισσότερο τον πραγματικό κόσμο, δηλαδή τον τρόπο με τον οποίο το σύστημα ανταποκρίνεται σε ερεθίσματα από τον πραγματικό κόσμο. Στην πραγματικότητα, ακόμη και με τη χρήση τεράστιων συνόλων δεδομένων των ακατέργαστων δεδομένων, συχνά, εάν δεν συλλέγονται με επάρκεια, αυτά μπορεί να απεικονίζουν ψευδείς ή μη ισορροπημένες καταστάσεις σε σχέση με τις πραγματικές [31].

Έτσι, μια κακή επιλογή δεδομένων, ή ακόμη και η εκτέλεση ανάλυσης σε ένα σύνολο δεδομένων που δεν είναι απόλυτα αντιπροσωπευτικό του συστήματος, θα οδηγήσει σε μοντέλα που θα απομακρυνθούν από το υπό μελέτη σύστημα.

Η αναζήτηση και η ανάκτηση δεδομένων απαιτεί συχνά μια μορφή διαίσθησης που υπερβαίνει την απλή τεχνική έρευνα και την εξαγωγή δεδομένων. Απαιτεί επίσης προσεκτική κατανόηση της φύσης των δεδομένων και της μορφής τους, την οποία μόνο η καλή εμπειρία και η γνώση στον τομέα εφαρμογής του προβλήματος μπορούν να δώσουν.

Ανεξάρτητα από την ποιότητα και την ποσότητα των δεδομένων που απαιτούνται, ένα άλλο ζήτημα είναι η αναζήτηση και η σωστή επιλογή των πηγών δεδομένων. Εάν το περιβάλλον είναι ένα

εργαστήριο (τεχνικό ή επιστημονικό) και τα δεδομένα που παράγονται είναι πειραματικά, τότε στην περίπτωση αυτή η πηγή των δεδομένων είναι εύκολα αναγνωρίσιμη. Σε αυτή την περίπτωση, τα προβλήματα θα αφορούν μόνο την πειραματική διάταξη.

Ωστόσο, η ανάλυση δεδομένων δεν είναι δυνατόν να αναπαράγει συστήματα στα οποία τα δεδομένα συλλέγονται με αυστηρά πειραματικό τρόπο σε κάθε πεδίο εφαρμογής. Πολλά πεδία εφαρμογής απαιτούν την αναζήτηση δεδομένων από τον περιβάλλοντα κόσμο, στηριζόμενοι συχνά σε πειραματικά δεδομένα εξωτερικά, ή ακόμη συχνότερα τη συλλογή τους μέσω συνεντεύξεων ή ερευνών. Έτσι, σε αυτές τις περιπτώσεις, η αναζήτηση μιας καλής πηγής δεδομένων που είναι σε θέση να παρέχει όλες τις πληροφορίες που χρειάζεστε για την ανάλυση δεδομένων μπορεί να είναι αρκετά δύσκολη. Συχνά είναι απαραίτητο να ανακτήσουμε δεδομένα από πολλαπλές πηγές δεδομένων για να συμπληρώσουμε τυχόν ελλείψεις, να εντοπίσουμε τυχόν αποκλίσεις και να κάνουμε το σύνολο των δεδομένων μας όσο το δυνατόν πιο γενικό.

Όταν θέλετε να πάρετε τα δεδομένα, ένα καλό μέρος για να ξεκινήσετε είναι μόνο ο Παγκόσμιος Ιστός. Αλλά τα περισσότερα δεδομένα στον Ιστό μπορεί να είναι δύσκολο να συλλεχθούν - στην πραγματικότητα δεν είναι όλα τα δεδομένα διαθέσιμα σε ένα αρχείο ή μια βάση δεδομένων - αλλά μπορεί να είναι λίγο πολύ σιωπηρά περιεχόμενο που βρίσκεται μέσα σε σελίδες HTML σε πολλές διαφορετικές μορφές. Για το σκοπό αυτό, έχει αναπτυχθεί μια μεθοδολογία που ονομάζεται Web Scraping, η οποία επιτρέπει τη συλλογή δεδομένων μέσω της αναγνώρισης συγκεκριμένης εμφάνισης ετικετών HTML μέσα στις ιστοσελίδες. Υπάρχουν λογισμικά ειδικά σχεδιασμένα για το σκοπό αυτό, και μόλις βρεθεί μια εμφάνιση, εξάγουν τα επιθυμητά δεδομένα. Μόλις ολοκληρωθεί η αναζήτηση, θα λάβετε έναν κατάλογο δεδομένων έτοιμο να υποβληθεί στην ανάλυση δεδομένων.

2.6.3 Προετοιμασία δεδομένων

Μεταξύ όλων των βημάτων που περιλαμβάνει η ανάλυση δεδομένων, η προετοιμασία των δεδομένων, αν και φαινομενικά είναι λιγότερο προβληματική, είναι στην πραγματικότητα ένα από αυτά που απαιτούν περισσότερους πόρους και περισσότερο χρόνο για να ολοκληρωθούν. Τα δεδομένα που συλλέγονται συχνά συλλέγονται από διαφορετικές πηγές δεδομένων, καθεμία από τις οποίες θα έχει τα δεδομένα σε αυτήν με διαφορετική αναπαράσταση και μορφή. Έτσι, όλα αυτά τα δεδομένα θα πρέπει να προετοιμαστούν για τη διαδικασία της ανάλυσης δεδομένων. Η προετοιμασία των δεδομένων αφορά τη λήψη, τον καθαρισμό, την κανονικοποίηση και τη μετατροπή των δεδομένων σε ένα βελτιστοποιημένο σύνολο δεδομένων, δηλαδή σε μια προετοιμασμένη μορφή, συνήθως σε μορφή πίνακα, κατάλληλη για τις μεθόδους ανάλυσης που έχουν προγραμματιστεί κατά τη φάση του σχεδιασμού.

Πολλά είναι τα προβλήματα που πρέπει να αποφεύγονται, όπως οι άκυρες, διφορούμενες ή ελλιπείς τιμές, τα επαναλαμβανόμενα πεδία ή τα εκτός εύρους δεδομένα.

2.6.4 Εξερεύνηση δεδομένων - εικονικοποίηση

Η εξερεύνηση των δεδομένων είναι ουσιαστικά η αναζήτηση δεδομένων σε μια γραφική ή στατιστική παρουσίαση προκειμένου να βρεθούν μοτίβα, συνδέσεις και σχέσεις στα δεδομένα. Η οπτικοποίηση δεδομένων είναι το καλύτερο εργαλείο για την ανάδειξη πιθανών μοτίβων.

Τα τελευταία χρόνια, η οπτικοποίηση δεδομένων έχει αναπτυχθεί σε τέτοιο βαθμό που έχει γίνει ένας πραγματικός κλάδος. Στην πραγματικότητα, πολλές τεχνολογίες χρησιμοποιούνται αποκλειστικά για την απεικόνιση δεδομένων και εξίσου πολλοί είναι οι τύποι απεικόνισης που εφαρμόζονται για την εξαγωγή των καλύτερων δυνατών πληροφοριών από ένα σύνολο δεδομένων.

Η διερεύνηση των δεδομένων συνίσταται σε μια προκαταρκτική εξέταση των δεδομένων, η οποία είναι σημαντική για την κατανόηση του είδους των πληροφοριών που έχουν συλλεχθεί και της

σημασίας τους. Σε συνδυασμό με τις πληροφορίες που αποκτήθηκαν κατά τη διάρκεια του ορισμού του προβλήματος, αυτή η κατηγοριοποίηση θα καθορίσει ποια μέθοδος ανάλυσης δεδομένων θα είναι η καταλληλότερη για την κατασκευή ενός μοντέλου.

Γενικά, η φάση αυτή, εκτός από τη λεπτομερή μελέτη των διαγραμμάτων μέσω των δεδομένων οπτικοποίησης, μπορεί να αποτελείται από μία ή περισσότερες από τις ακόλουθες δραστηριότητες:

- Συγκέντρωση δεδομένων
- Ομαδοποίηση δεδομένων
- Διερεύνηση της σχέσης μεταξύ των διαφόρων χαρακτηριστικών
- Προσδιορισμός προτύπων και τάσεων
- Κατασκευή μοντέλων παλινδρόμησης
- Κατασκευή μοντέλων ταξινόμησης

Γενικά, η ανάλυση δεδομένων απαιτεί διαδικασίες σύνοψης σχετικά με τα δεδομένα που πρόκειται να μελετηθούν. Η σύνοψη είναι μια διαδικασία με την οποία τα δεδομένα περιορίζονται σε ερμηνεία χωρίς να θυσιάζονται σημαντικές πληροφορίες. Η ομαδοποίηση είναι μια μέθοδος ανάλυσης δεδομένων που χρησιμοποιείται για την εύρεση ομάδων που ενώνονται με κοινά χαρακτηριστικά (ομαδοποίηση).

Ένα άλλο σημαντικό βήμα της ανάλυσης επικεντρώνεται στον εντοπισμό σχέσεων, τάσεων και ανωμαλιών στα δεδομένα. Για να ανακαλύψει κανείς αυτού του είδους τις πληροφορίες, συχνά πρέπει να καταφύγει στα εργαλεία καθώς και να εκτελέσει έναν ακόμη γύρο ανάλυσης δεδομένων, αυτή τη φορά στην ίδια την οπτικοποίηση των δεδομένων.

Άλλες μέθοδοι εξόρυξης δεδομένων, όπως τα δέντρα αποφάσεων και οι κανόνες συσχέτισης, εξάγουν αυτόματα σημαντικά γεγονότα ή κανόνες από τα δεδομένα. Αυτές οι προσεγγίσεις μπορούν να χρησιμοποιηθούν παράλληλα με την οπτικοποίηση δεδομένων για την εύρεση πληροφοριών σχετικά με τις σχέσεις μεταξύ των δεδομένων.

2.6.5 Προγνωστική μοντελοποίηση

Η προγνωστική μοντελοποίηση είναι μια διαδικασία που χρησιμοποιείται στην ανάλυση δεδομένων για τη δημιουργία ή την επιλογή ενός κατάλληλου στατιστικού μοντέλου για την πρόβλεψη της πιθανότητας ενός αποτελέσματος.

Μετά τη διερεύνηση των δεδομένων έχετε όλες τις πληροφορίες που απαιτούνται για την ανάπτυξη του μαθηματικού μοντέλου που κωδικοποιεί τη σχέση μεταξύ των δεδομένων. Τα μοντέλα αυτά είναι χρήσιμα για την κατανόηση του υπό μελέτη συστήματος και με συγκεκριμένο τρόπο χρησιμοποιούνται για δύο κύριους σκοπούς. Ο πρώτος είναι να κάνετε προβλέψεις σχετικά με τις τιμές των δεδομένων που παράγει το σύστημα (σε αυτή την περίπτωση, θα ασχοληθείτε με μοντέλα παλινδρόμησης). Ο δεύτερος είναι η ταξινόμηση νέων παραγόμενων δεδομένων (και σε αυτή την περίπτωση θα χρησιμοποιήσετε μοντέλα ταξινόμησης ή μοντέλα ομαδοποίησης). Στην πραγματικότητα, είναι δυνατόν να διαχωρίσετε τα μοντέλα ανάλογα με τον τύπο του αποτελέσματος που παράγουν:

- Μοντέλα ταξινόμησης: Εάν το αποτέλεσμα που προκύπτει από τον τύπο του μοντέλου είναι κατηγοριοποιημένο.
- Μοντέλα παλινδρόμησης: Αν το αποτέλεσμα που προκύπτει από τον τύπο μοντέλου είναι αριθμητικό.
- Μοντέλα ομαδοποίησης: Εάν το αποτέλεσμα που προκύπτει από τον τύπο μοντέλου είναι περιγραφικό.

Οι απλές μέθοδοι για τη δημιουργία αυτών των μοντέλων περιλαμβάνουν τεχνικές όπως η γραμμική παλινδρόμηση, η λογιστική παλινδρόμηση, τα δέντρα ταξινόμησης και παλινδρόμησης και οι k-κοντινότεροι γείτονες (k-nearest neighbors). Όμως οι μέθοδοι ανάλυσης είναι πολυάριθμες και η καθεμία έχει συγκεκριμένα χαρακτηριστικά που την καθιστούν εξαιρετική για ορισμένους τύπους δεδομένων και αναλύσεων. Κάθε μία από αυτές τις μεθόδους θα παράγει ένα συγκεκριμένο μοντέλο και στη συνέχεια η επιλογή τους είναι σχετική με τη φύση του μοντέλου.

Ορισμένα από αυτά τα μοντέλα θα παρέχουν τιμές που αντιστοιχούν στο πραγματικό σύστημα και, επίσης, ανάλογα με τη δομή τους, θα εξηγούν ορισμένα χαρακτηριστικά του υπό μελέτη συστήματος με απλό και σαφή τρόπο. Άλλα μοντέλα θα συνεχίσουν να δίνουν καλές προβλέψεις, αλλά η δομή τους δεν θα είναι τίποτα περισσότερο από ένα "μαύρο κουτί" με περιορισμένη ικανότητα να εξηγεί ορισμένα χαρακτηριστικά του συστήματος.

2.6.6 Επικύρωση μοντέλου

Η επικύρωση του μοντέλου, δηλαδή η φάση δοκιμής, είναι μια σημαντική φάση που σας επιτρέπει να επικυρώσετε το μοντέλο που κατασκευάστηκε με βάση τα αρχικά δεδομένα. Αυτό είναι σημαντικό διότι σας επιτρέπει να αξιολογήσετε την εγκυρότητα των δεδομένων που παράγει το μοντέλο, συγκρίνοντας τα άμεσα με το πραγματικό σύστημα. Αλλά αυτή τη φορά, βγαίνετε από το σύνολο των αρχικών δεδομένων πάνω στα οποία έχει δημιουργηθεί ολόκληρη η ανάλυση.

Γενικά, θα αναφέρεστε στα δεδομένα ως σύνολο εκπαίδευσης, όταν τα χρησιμοποιείτε για τη δημιουργία του μοντέλου, και ως σύνολο επικύρωσης, όταν τα χρησιμοποιείτε για την επικύρωση του μοντέλου.

Έτσι, συγκρίνοντας τα δεδομένα που παράγονται από το μοντέλο με εκείνα που παράγει το σύστημα, θα είστε σε θέση να αξιολογήσετε το σφάλμα και χρησιμοποιώντας διαφορετικά σύνολα δοκιμαστικών δεδομένων, μπορείτε να εκτιμήσετε τα όρια εγκυρότητας του παραγόμενου μοντέλου. Στην πραγματικότητα, οι σωστά προβλεπόμενες τιμές θα μπορούσαν να είναι έγκυρες μόνο εντός ενός συγκεκριμένου εύρους ή να έχουν διαφορετικά επίπεδα ταύτισης ανάλογα με το εύρος των τιμών που λαμβάνονται υπόψη.

Η διαδικασία αυτή σας επιτρέπει όχι μόνο να αξιολογήσετε αριθμητικά την αποτελεσματικότητα του μοντέλου, αλλά και να το συγκρίνετε με άλλα υπάρχοντα μοντέλα. Υπάρχουν διάφορες τεχνικές από αυτή την άποψη. Η πιο διάσημη είναι η διασταυρούμενη επικύρωση. Η τεχνική αυτή βασίζεται στη διαίρεση του συνόλου εκπαίδευσης σε διαφορετικά μέρη. Κάθε ένα από αυτά τα μέρη, με τη σειρά του, θα χρησιμοποιηθεί ως σύνολο επικύρωσης και οποιοδήποτε άλλο ως σύνολο εκπαίδευσης. Με αυτόν τον επαναληπτικό τρόπο, θα έχετε ένα ολοένα και πιο τελειοποιημένο μοντέλο.

2.6.7 Ανάπτυξη

Πρόκειται για το τελικό στάδιο της διαδικασίας ανάλυσης, το οποίο αποσκοπεί στην παρουσίαση των αποτελεσμάτων, δηλαδή των συμπερασμάτων της ανάλυσης. Κατά τη διαδικασία ανάπτυξης, στο επιχειρηματικό περιβάλλον, η ανάλυση μεταφράζεται σε όφελος για τον πελάτη που την ανέθεσε. Σε τεχνικά ή επιστημονικά περιβάλλοντα, μεταφράζεται σε σχεδιαστικές λύσεις ή επιστημονικές δημοσιεύσεις. Δηλαδή, η ανάπτυξη συνίσταται βασικά στην εφαρμογή στην πράξη των αποτελεσμάτων που προκύπτουν από την ανάλυση δεδομένων.

Υπάρχουν διάφοροι τρόποι για την ανάπτυξη των αποτελεσμάτων μιας ανάλυσης δεδομένων ή εξόρυξης δεδομένων. Συνήθως, η ανάπτυξη ενός αναλυτή δεδομένων συνίσταται στη σύνταξη μιας έκθεσης για τη διοίκηση ή για τον πελάτη που ζήτησε την ανάλυση.

Για παράδειγμα, σε μια επιχείρηση, στην τεκμηρίωση που παρέχεται από τον αναλυτή, περιέχονται τα εξής τέσσερα θέματα:

- Αποτελέσματα ανάλυσης
- Ανάπτυξη απόφασης
- Ανάλυση κινδύνου
- Μέτρηση του επιχειρηματικού αντίκτυπου

Όταν τα αποτελέσματα του έργου περιλαμβάνουν τη δημιουργία προγνωστικών μοντέλων, τα μοντέλα αυτά μπορούν να αναπτυχθούν ως αυτόνομη εφαρμογή ή να ενσωματωθούν σε άλλο λογισμικό.

2.7 Ποσοτική και ποιοτική ανάλυση δεδομένων

Η ανάλυση δεδομένων είναι μια διαδικασία που επικεντρώνεται πλήρως στα δεδομένα και ανάλογα με τη φύση των δεδομένων είναι δυνατόν να γίνουν ορισμένες διακρίσεις. Όταν τα δεδομένα που αναλύονται έχουν αυστηρά αριθμητική ή έχουν κατηγορική δομή, τότε μιλάμε για ποσοτική ανάλυση, αλλά όταν έχουμε να κάνουμε με τιμές που εκφράζονται μέσω περιγραφών σε φυσική γλώσσα, τότε μιλάμε για ποιοτική ανάλυση. Ακριβώς λόγω της διαφορετικής φύσης των δεδομένων που επεξεργάζονται οι δύο τύποι αναλύσεων, μπορείτε να παρατηρήσετε ορισμένες διαφορές μεταξύ τους [31].

Η ποσοτική ανάλυση έχει να κάνει με δεδομένα που έχουν μια λογική σειρά μέσα τους ή που μπορούν να κατηγοριοποιηθούν με κάποιο τρόπο. Αυτό οδηγεί στο σχηματισμό δομών μέσα στα δεδομένα. Η σειρά, η κατηγοριοποίηση και οι δομές παρέχουν περισσότερες πληροφορίες και επιτρέπουν την περαιτέρω επεξεργασία των δεδομένων με έναν πιο αυστηρά μαθηματικό τρόπο. Αυτό οδηγεί στη δημιουργία μοντέλων που μπορούν να παρέχουν ποσοτικές προβλέψεις, επιτρέποντας έτσι στον αναλυτή δεδομένων να εξάγει πιο αντικειμενικά συμπεράσματα.

Η ποιοτική ανάλυση, αντίθετα, έχει να κάνει με δεδομένα που γενικά δεν έχουν δομή, τουλάχιστον όχι εμφανή και η φύση τους δεν είναι ούτε αριθμητική ούτε κατηγορική. Για παράδειγμα, τα δεδομένα για την ποιοτική μελέτη θα μπορούσαν να περιλαμβάνουν κείμενα, οπτικά ή ηχητικά δεδομένα. Αυτός ο τύπος ανάλυσης πρέπει επομένως να βασίζεται σε μεθοδολογίες, συχνά ad hoc, για την εξαγωγή πληροφοριών που θα οδηγήσουν γενικά σε μοντέλα ικανά να παρέχουν ποιοτικές προβλέψεις, με αποτέλεσμα τα συμπεράσματα στα οποία μπορεί να καταλήξει ο αναλυτής δεδομένων να περιλαμβάνουν και υποκειμενικές ερμηνείες. Από την άλλη πλευρά, η ποιοτική ανάλυση μπορεί να διερευνήσει πιο πολύπλοκα συστήματα και να εξάγει συμπεράσματα που δεν είναι δυνατά με μια αυστηρά μαθηματική προσέγγιση. Συχνά αυτός ο τύπος ανάλυσης αφορά τη μελέτη συστημάτων όπως τα κοινωνικά φαινόμενα ή πολύπλοκες δομές που δεν είναι εύκολα μετρήσιμες.

Το Σχήμα 2-2 δείχνει τις διαφορές μεταξύ των δύο τύπων ανάλυσης:

Σχήμα 2-2. Ποσοτικές και ποιοτικές αναλύσεις

ΚΕΦΑΛΑΙΟ 3

Εργαλεία για την ανάλυση μεγάλων δεδομένων

Υπάρχουν πολλά εργαλεία που βοηθούν στην επίτευξη αυτών των στόχων και βοηθούν τους επιστήμονες δεδομένων να επεξεργάζονται δεδομένα για την ανάλυσή τους. Έχουν εμφανιστεί πολλές νέες γλώσσες, πλαίσια και τεχνολογίες αποθήκευσης δεδομένων που υποστηρίζουν το χειρισμό μεγάλων δεδομένων.

Σε αυτό το κεφάλαιο θα αναφερθούμε σε ορισμένα από τα εργαλεία που μπορούν να χρησιμοποιηθούν για την εκτέλεση διαφόρων λειτουργιών που σχετίζονται με την ανάλυση μεγάλων δεδομένων για την εξαγωγή των επιθυμητών αποτελεσμάτων.

Περισσότερο βάρος θα δοθεί στο επόμενο κεφάλαιο στην καθοριστική σημασία που έχει η Python στην ανάλυση των δεδομένων.

3.1 MapReduce

Ένα πλαίσιο λογισμικού που επιτρέπει στους προγραμματιστές να γράφουν προγράμματα που επεξεργάζονται παράλληλα τεράστιες ποσότητες αδόμητων δεδομένων σε μια κατανεμημένη συστάδα επεξεργαστών ή σε μεμονωμένους υπολογιστές. Το MapReduce είναι σημαντικό επειδή επιτρέπει στους απλούς προγραμματιστές να χρησιμοποιούν τις ρουτίνες της βιβλιοθήκης MapReduce για τη δημιουργία παράλληλων προγραμμάτων χωρίς να χρειάζεται να ανησυχούν για τον προγραμματισμό της επικοινωνίας εντός του σμήνους, την παρακολούθηση των εργασιών ή τον χειρισμό αποτυχιών [32].

3.2 Hadoop

Το Hadoop είναι ένα πλαίσιο ανοικτού κώδικα για την κατανεμημένη αποθήκευση και επεξεργασία μεγάλων συνόλων δεδομένων σε υλικό βασικής χρήσης. Το Hadoop επιτρέπει στις επιχειρήσεις να αποκτούν γρήγορα πληροφορίες από τεράστιες ποσότητες δομημένων και αδόμητων δεδομένων. Έχει σχεδιαστεί για να κλιμακώνεται από έναν μόνο διακομιστή έως χιλιάδες μηχανές, με πολύ υψηλό βαθμό ανοχής σφαλμάτων. Αντί να βασίζεται σε υλικό υψηλών προδιαγραφών, η ανθεκτικότητα αυτών των συστάδων προέρχεται από την ικανότητα του λογισμικού να ανιχνεύει και να χειρίζεται τις αποτυχίες στο επίπεδο εφαρμογής [33].

Το Hadoop επιτρέπει μια υπολογιστική λύση που είναι:

Επεκτάσιμη: χωρίς να αλλάξουν οι μορφές δεδομένων, ο τρόπος φόρτωσης των δεδομένων, ο τρόπος εγγραφής των εργασιών ή οι εφαρμογές στην κορυφή.

Οικονομικά αποδοτική: Το Hadoop φέρνει μαζικά παράλληλους υπολογισμούς σε διακομιστές κοινής χρήσης. Το αποτέλεσμα είναι μια σημαντική μείωση του κόστους ανά terabyte αποθήκευσης, η οποία με τη σειρά της καθιστά προσιτή τη μοντελοποίηση όλων των δεδομένων.

Ευέλικτη: Το Hadoop δεν έχει σχήματα και μπορεί να απορροφήσει οποιονδήποτε τύπο δεδομένων από οποιονδήποτε αριθμό πηγών. Δεδομένα από πολλαπλές πηγές μπορούν να ενωθούν και να συγκεντρωθούν με αυθαίρετους τρόπους, επιτρέποντας βαθύτερες αναλύσεις από αυτές που μπορεί να προσφέρει ένα σύστημα.

Ανοχή σε σφάλματα: Όταν ένας κόμβος χάνεται, το σύστημα ανακατευθύνει την εργασία σε μια άλλη θέση τα δεδομένα και συνεχίζει την επεξεργασία χωρίς να χάνει ούτε ένα ρυθμό.

3.3 Apache Hive

Το λογισμικό αποθήκης δεδομένων Apache Hive διευκολύνει την αναζήτηση και τη διαχείριση

μεγάλων συνόλων δεδομένων που βρίσκονται σε κατανεμημένη αποθήκευση. Το Hive παρέχει έναν μηχανισμό για την προβολή δομής σε αυτά τα δεδομένα και την υποβολή ερωτημάτων στα δεδομένα με τη χρήση μιας γλώσσας που μοιάζει με SQL και ονομάζεται HiveQL. Ταυτόχρονα, η γλώσσα αυτή επιτρέπει στους παραδοσιακούς προγραμματιστές map/reduce να συνδέουν τους προσαρμοσμένους mappers και reducers τους, όταν είναι άβολο ή αναποτελεσματικό να εκφράσουν αυτή τη λογική στην HiveQL [34].

3.4 Apache Pig

Το Apache Pig είναι μια πλατφόρμα για την ανάλυση μεγάλων συνόλων δεδομένων που αποτελείται από μια γλώσσα υψηλού επιπέδου για την έκφραση προγραμμάτων ανάλυσης δεδομένων, σε συνδυασμό με υποδομές για την αξιολόγηση αυτών των προγραμμάτων. Η εξέχουσα ιδιότητα των προγραμμάτων Pig είναι ότι η δομή τους επιδέχεται σημαντικό παραλληλισμό, ο οποίος με τη σειρά του τους επιτρέπει να χειρίζονται πολύ μεγάλα σύνολα δεδομένων [35].

Προς το παρόν, το επίπεδο υποδομής του Pig αποτελείται από έναν μεταγλωττιστή που παράγει ακολουθίες προγραμμάτων MapReduce, για τα οποία υπάρχουν ήδη παράλληλες υλοποιήσεις μεγάλης κλίμακας (π.χ. το υποπρόγραμμα Hadoop). Το γλωσσικό στρώμα του Pig αποτελείται επί του παρόντος από μια γλώσσα κειμένου που ονομάζεται Pig Latin, η οποία έχει τις ακόλουθες βασικές ιδιότητες:

Ευκολία προγραμματισμού: Είναι τετριμμένο να επιτευχθεί η παράλληλη εκτέλεση απλών, "ενοχλητικά παράλληλων" εργασιών ανάλυσης δεδομένων. Οι σύνθετες εργασίες που αποτελούνται από πολλαπλούς αλληλένδετους μετασχηματισμούς δεδομένων κωδικοποιούνται ρητά ως ακολουθίες ροής δεδομένων, καθιστώντας εύκολη τη συγγραφή, την κατανόηση και τη συντήρησή τους.

Ευκαιρίες βελτιστοποίησης: Ο τρόπος κωδικοποίησης των εργασιών επιτρέπει στο σύστημα να βελτιστοποιεί αυτόματα την εκτέλεσή τους, επιτρέποντας στο χρήστη να εστιάζει στη σημασιολογία και όχι στην αποτελεσματικότητα.

Επεκτασιμότητα: Οι χρήστες μπορούν να δημιουργήσουν τις δικές τους συναρτήσεις για να κάνουν επεξεργασία ειδικού σκοπού.

3.5 Apache Spark

Είναι μια τεχνολογία υπολογισμού συστάδων στη μνήμη, σχεδιασμένη για γρήγορους υπολογισμούς, η οποία υλοποιείται στη Scala. Χρησιμοποιεί το Hadoop για σκοπούς αποθήκευσης, καθώς έχει τη δική του δυνατότητα διαχείρισης συστάδων. Παρέχει ενσωματωμένα API για Java, Scala και Python. Πρόσφατα, άρχισε επίσης να υποστηρίζει την R. Διαθέτει 80 τελεστές υψηλού επιπέδου για διαδραστικές ερωτήσεις. Ο υπολογισμός στη μνήμη υποστηρίζεται με το πλαίσιο Resilient Distributed Data (RDD), το οποίο διανέμει το πλαίσιο δεδομένων σε μικρότερα κομμάτια σε διαφορετικές μηχανές για ταχύτερους υπολογισμούς. Υποστηρίζει επίσης Map και Reduce για την επεξεργασία δεδομένων. Υποστηρίζει SQL, ροή δεδομένων, αλγορίθμους επεξεργασίας γράφων και αλγορίθμους μηχανικής μάθησης. Αν και η Spark μπορεί να προσπελαστεί με Python, Java και R, έχει ισχυρή υποστήριξη για τη Scala και είναι πιο σταθερή αυτή τη στιγμή [36].

3.6 MADlib

Η MADlib είναι μια βιβλιοθήκη ανοικτού κώδικα για κλιμακούμενες αναλύσεις σε βάσεις δεδομένων που μπορούν να βοηθήσουν στη βελτίωση της αποτελεσματικότητας και της ακρίβειας της ανάλυσης δεδομένων. Παρέχει παράλληλες υλοποιήσεις δεδομένων μαθηματικών, στατιστικών και μεθόδων μηχανικής μάθησης για δομημένα και αδόμητα δεδομένα. Αυτοί οι βασισμένοι σε SQL αλγόριθμοι για μηχανική μάθηση, εξόρυξη δεδομένων και στατιστική εκτελούνται με ταχύτητα και κλίμακα [37].

3.7 Amazon Elastic Compute Cloud (EC2)

Είναι μια διαδικτυακή υπηρεσία που παρέχει υπολογιστική χωρητικότητα μέσω του υπολογιστικού νέφους. Παρέχει πλήρη έλεγχο των υπολογιστικών πόρων και επιτρέπει στους προγραμματιστές να εκτελούν τους υπολογισμούς τους στο επιθυμητό υπολογιστικό περιβάλλον. Είναι μια από τις πιο επιτυχημένες πλατφόρμες υπολογιστικού νέφους. Λειτουργεί με βάση την αρχή του μοντέλου pay-as-you-go.

3.8 R

Είναι μια γλώσσα στατιστικών υπολογισμών ανοικτού κώδικα που παρέχει μια μεγάλη ποικιλία στατιστικών και γραφικών τεχνικών για την εξαγωγή συμπερασμάτων από τα δεδομένα. Διαθέτει μια αποτελεσματική δυνατότητα χειρισμού και αποθήκευσης δεδομένων και υποστηρίζει διανυσματικές πράξεις με μια σουίτα τελεστών για ταχύτερη επεξεργασία. Διαθέτει όλα τα χαρακτηριστικά μιας τυπικής γλώσσας προγραμματισμού και υποστηρίζει ορίσματα υπό όρους, βρόχους και συναρτήσεις που ορίζονται από τον χρήστη [38].
Η R υποστηρίζεται από έναν τεράστιο αριθμό πακέτων μέσω του Comprehensive R Archive Network (CRAN). Είναι διαθέσιμη σε πλατφόρμες Windows, Linux και Mac. Διαθέτει ισχυρή τεκμηρίωση για κάθε πακέτο. Διαθέτει ισχυρή υποστήριξη για αλγόριθμους επεξεργασίας δεδομένων, εξόρυξης δεδομένων και μηχανικής μάθησης μαζί με καλή υποστήριξη για ανάγνωση και εγγραφή σε κατανεμημένο περιβάλλον, γεγονός που το καθιστά κατάλληλο για το χειρισμό μεγάλων δεδομένων. Ωστόσο, η διαχείριση της μνήμης, η ταχύτητα και η αποδοτικότητα είναι ίσως η μεγαλύτερη πρόκληση που αντιμετωπίζει η R. Το R Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης που έχει αναπτυχθεί για τον προγραμματισμό στη γλώσσα R. Διανέμεται για αυτόνομες μηχανές Desktop καθώς και υποστηρίζει αρχιτεκτονική πελάτη-εξυπηρετητή, στην οποία μπορεί να έχει πρόσβαση από οποιοδήποτε πρόγραμμα περιήγησης.

3.9 Scala

Είναι μια αντικειμενοστραφής γλώσσα και έχει ακρωνύμιο για την "Επεκτάσιμη Γλώσσα" (“Scalable Language”). Το αντικείμενο και κάθε λειτουργία στη Scala είναι μια κλήση μεθόδου, όπως σε κάθε αντικειμενοστραφή γλώσσα. Απαιτεί περιβάλλον εικονικής μηχανής java. Το Spark, ένα πλαίσιο υπολογισμού σε συστοιχία μνήμης, είναι γραμμένο στη Scala. Η Scala γίνεται δημοφιλές εργαλείο προγραμματισμού για το χειρισμό προβλημάτων μεγάλων δεδομένων [39].

3.10 Python

Είναι μια από τις πιο δημοφιλείς γλώσσες προγραμματισμού, η οποία είναι ανοικτού κώδικα και υποστηρίζεται από τις πλατφόρμες Windows, Linux και Mac. Φιλοξενεί χιλιάδες πακέτα από τρίτους ή modules που συνεισφέρει η κοινότητα. Τα NumPy, Scikit και Pandas υποστηρίζουν μερικά από τα δημοφιλή πακέτα για μηχανική μάθηση και εξόρυξη δεδομένων για προεπεξεργασία δεδομένων, υπολογισμό και μοντελοποίηση. Το NumPy είναι το βασικό πακέτο για τον επιστημονικό υπολογισμό. Προσθέτει υποστήριξη για μεγάλους, πολυδιάστατους πίνακες και πίνακες με Python. Το Scikit υποστηρίζει ταξινόμηση, παλινδρόμηση, ομαδοποίηση, μείωση διαστάσεων, επιλογή χαρακτηριστικών, καθώς και αλγορίθμους προεπεξεργασίας και επιλογής μοντέλων. Το Pandas βοηθά στη συγκέντρωση δεδομένων και στην προετοιμασία για την ανάλυση δεδομένων και τη μοντελοποίηση. Διαθέτει ισχυρή υποστήριξη για την ανάλυση γραφημάτων με τη βιβλιοθήκη Matplotlib και Seaborn και επιπλέον με τις βιβλιοθήκες NetworkX και την nltk για την ανάλυση κειμένου και την επεξεργασία φυσικής γλώσσας. Η Python είναι πολύ φιλική προς το χρήστη και ιδανική για γρήγορη και ουσιαστική ανάλυση ενός προβλήματος. Ενσωματώνεται επίσης καλά με το spark μέσω της βιβλιοθήκης pyspark [40].

ΚΕΦΑΛΑΙΟ 4

Python και ανάλυση δεδομένων

Η Python είναι μια γλώσσα προγραμματισμού που χρησιμοποιείται ευρέως στους επιστημονικούς κύκλους λόγω του μεγάλου αριθμού βιβλιοθηκών που παρέχει ένα πλήρες σύνολο εργαλείων για την ανάλυση και τον χειρισμό δεδομένων.

Σε σύγκριση με άλλες γλώσσες προγραμματισμού που χρησιμοποιούνται γενικά για την ανάλυση δεδομένων, όπως η R και η Matlab, η Python όχι μόνο παρέχει μια πλατφόρμα για την επεξεργασία δεδομένων, αλλά διαθέτει και ορισμένα χαρακτηριστικά που την καθιστούν μοναδική σε σύγκριση με άλλες γλώσσες και εξειδικευμένες εφαρμογές. Η ανάπτυξη ενός συνεχώς αυξανόμενου αριθμού υποστηρικτικών βιβλιοθηκών, η υλοποίηση αλγορίθμων πιο καινοτόμων μεθοδολογιών και η δυνατότητα διασύνδεσης με άλλες γλώσσες προγραμματισμού (C και Fortran) καθιστούν την Python μοναδική στο είδος της [31].

Επιπλέον, η Python δεν είναι μόνο εξειδικευμένη για την ανάλυση δεδομένων, αλλά έχει και πολλές άλλες εφαρμογές, όπως ο γενικός προγραμματισμός, η δημιουργία σεναρίων, η διασύνδεση με βάσεις δεδομένων και πρόσφατα και η ανάπτυξη ιστοσελίδων, χάρη σε διαδικτυακά πλαίσια όπως το Django. Έτσι, είναι δυνατή η ανάπτυξη έργων ανάλυσης δεδομένων που είναι απολύτως συμβατά με τον εξυπηρετητή ιστού με τη δυνατότητα ενσωμάτωσής τους σε αυτόν.

Έτσι, για όσους θέλουν να κάνουν ανάλυση δεδομένων, η Python, με όλα τα πακέτα της, μπορεί να θεωρηθεί η καλύτερη επιλογή για το προσεχές μέλλον.

4.1 Η Python ως εργαλείο ανάλυσης δεδομένων

Στον κόσμο της ανάλυσης δεδομένων, η Python έχει αναδειχθεί ως μια εξέχουσα γλώσσα προγραμματισμού. Σύμφωνα με τον δείκτη TIOBE Index, ο οποίος μετρά τη δημοτικότητα των γλωσσών προγραμματισμού, η Python είναι η πιο δημοφιλής γλώσσα προγραμματισμού στον κόσμο. Η δημοτικότητά της μεταξύ των αναλυτών δεδομένων πηγάζει από την ευελιξία της, τις εκτεταμένες βιβλιοθήκες της και το διαισθητικό συντακτικό της.

Σε αυτή την ενότητα εξηγούμε τους βασικούς λόγους για τους οποίους οι επαγγελματίες και οι ερασιτέχνες αναλυτές δεδομένων επιλέγουν την Python ως εργαλείο για την εξερεύνηση, τον χειρισμό, την οπτικοποίηση και τη μοντελοποίηση δεδομένων.

4.1.1 Πώς χρησιμοποιούν οι αναλυτές δεδομένων την Python;

Η γλώσσα προγραμματισμού υψηλού επιπέδου χρησιμοποιείται για την τεχνητή νοημοσύνη (AI), την ανάπτυξη API, το Διαδίκτυο των πραγμάτων (IoT) και την ανάπτυξη ιστοσελίδων.

Η ευκολία χρήσης της και η εντυπωσιακή γκάμα βιβλιοθηκών την καθιστούν δημοφιλή μεταξύ των αναλυτών δεδομένων. Η γλώσσα βοηθά τους αναλυτές δεδομένων σε κάθε ένα από τα βήματα της ανάλυσης δεδομένων [41].

4.1.2 Συλλογή δεδομένων

Οι επαγγελματίες των δεδομένων χρησιμοποιούν βιβλιοθήκες Python όπως η BeautifulSoup και η Scrapy για να εξορύξουν - ή να συλλέξουν - δεδομένα. Το Scrapy, για παράδειγμα, βοηθά στη δημιουργία προγραμμάτων για τη συλλογή οργανωμένων και δομημένων δεδομένων από τον ιστό.

Το BeautifulSoup, ακόμη περισσότερο από το Scrapy, επικεντρώνεται στο web scraping [27].

4.1.3 Επεξεργασία δεδομένων

Οι αναλυτές δεδομένων μπορούν επίσης να χρησιμοποιούν βιβλιοθήκες Python για να δομούν μεγάλα σύνολα δεδομένων και να κάνουν τις μαθηματικές πράξεις πιο εύχρηστες.

Η Pandas, μια βιβλιοθήκη της Python, προσφέρει μια δομή δεδομένων που ονομάζεται πλαίσιο δεδομένων (dataframe) για την αποτελεσματική εργασία με μεγάλους πίνακες δεδομένων.

Το Pandas σας επιτρέπει να φορτώσετε τα δεδομένα στο πλαίσιο δεδομένων και στη συνέχεια να προσθέσετε νέες υπολογιζόμενες στήλες με βάση διάφορες απλές αλλά ισχυρές λειτουργίες. Συνολικά, βοηθά στην εκτέλεση διαφόρων λειτουργιών ανάλυσης δεδομένων για τη βελτίωση της αποδοτικότητας της εργασίας.

4.1.4 Οπτικοποίηση δεδομένων

Οι αναλυτές δεδομένων πρέπει να παρουσιάζουν σημαντικές πληροφορίες σε κατανοητή μορφή. Αυτό βοηθά τους υπεύθυνους λήψης αποφάσεων της εταιρείας να εντοπίζουν τις τάσεις και να λαμβάνουν καλύτερες επιχειρηματικές αποφάσεις.

Οι βιβλιοθήκες της Python, όπως η Matplotlib, επιτρέπουν στους αναλυτές δεδομένων να μετατρέπουν αριθμούς σε διαγράμματα πίτας, γραφικά, ιστογράμματα κ.λπ.

Αυτό διευκολύνει τους αναλυτές δεδομένων να κάνουν τα δεδομένα που οδηγούν οπτικά ελκυστικά και κατανοητά.

4.2 Πλεονεκτήματα της Python

Η διαλειτουργικότητα της γλώσσας προσφέρει πολλά πλεονεκτήματα στους χρήστες της. Βοηθά τους αναλυτές δεδομένων να κατανοήσουν πολύπλοκα σύνολα δεδομένων και να τα κάνουν πιο εύκολα κατανοητά.

Ένα άλλο πλεονέκτημα της χρήσης της Python είναι η υψηλή αναγνωσιμότητά της. Ο κώδικας Python είναι ευκολότερος για τη συνεργασία με άλλους αναλυτές, για την επικοινωνία με άλλους τεχνικούς ενδιαφερόμενους και τον καθιστά πιο συντηρήσιμο όταν έρθει η ώρα να προσαρμοστεί για νέες πηγές δεδομένων και ανάγκες.

Η γλώσσα είναι κατάλληλη για επαγγελματίες αναλυτές δεδομένων, καθώς παρέχει μεγάλη υποστήριξη και προσφέρει ένα ευρύ φάσμα βιβλιοθηκών για διάφορες εργασίες [31].

4.2.1 Εύκολη στην εκμάθηση

Η Python είναι γνωστή για την απλή σύνταξη και την αναγνωσιμότητά της, γεγονός που αποτελεί σημαντικό πλεονέκτημα. Μειώνει το χρόνο που οι αναλυτές δεδομένων θα ξόδευαν διαφορετικά για να εξοικειωθούν με μια γλώσσα προγραμματισμού.

Η ήπια καμπύλη εκμάθησης την κάνει να ξεχωρίζει ανάμεσα στις παλιές γλώσσες προγραμματισμού με περίπλοκο συντακτικό.

Για παράδειγμα, γλώσσες όπως η Java, η C++ και η Ruby απαιτούν μια απότομη καμπύλη εκμάθησης, ειδικά για αρχάριους αναλυτές δεδομένων. Από την άλλη πλευρά, η απλότητα της Python βοηθά τους αναλυτές δεδομένων να εκτελούν ταυτόχρονα διάφορες εργασίες που σχετίζονται με δεδομένα.

Αποτελεί μια απλή αλλά αποτελεσματική γλώσσα προγραμματισμού προσφέροντας λύσεις για τις περισσότερες πολυπλοκότητες που αντιμετωπίζονται κατά το χειρισμό δεδομένων.

4.2.2 Τεράστια συλλογή βιβλιοθηκών

Η Python προσφέρει στους χρήστες της έναν εκτενή κατάλογο δωρεάν βιβλιοθηκών. Το πιο

δελεαστικό με τις βιβλιοθήκες είναι ότι αναπτύσσονται συνεχώς, προσφέροντας ισχυρές λύσεις. Οι Numpy, Scikit-leam, Pandas και Matplotlib είναι μερικές δημοφιλείς βιβλιοθήκες που βοηθούν στην επιτάχυνση των εργασιών ανάλυσης δεδομένων.

4.2.3 Καλά υποστηριζόμενη

Παρά την απλότητά της, θα βρεθείτε σε καταστάσεις όπου θα χρειαστείτε βοήθεια με τη γλώσσα προγραμματισμού. Ευτυχώς, προσφέρει μια σειρά από χρήσιμες βιβλιοθήκες με χρήσιμο υποστηρικτικό υλικό. Επιπλέον, μπορείτε να τις χρησιμοποιήσετε δωρεάν. Εκτός αυτού, μπορείτε να έχετε πρόσβαση σε κώδικες που συνεισφέρουν οι χρήστες, από λίστες αλληλογραφίας μέχρι τεκμηρίωση και πολλά άλλα. Επιπλέον, οι χρήστες παγκοσμίως μπορούν να απευθύνονται σε εξειδικευμένους προγραμματιστές για να ζητήσουν βοήθεια και συμβουλές όταν χρειάζεται.

4.2.4 Εργαλεία οπτικοποίησης

Ο εγκέφαλός μας επεξεργάζεται τα οπτικά στοιχεία καλύτερα από το κείμενο. Ως εκ τούτου, οι αναλυτές δεδομένων παρουσιάζουν σημαντικές πληροφορίες σε γραφήματα, διαγράμματα ή γραφικά για να τις κάνουν πιο κατανοητές. Ευτυχώς, δεν χρειάζεται να είστε ειδικός στην οπτικοποίηση δεδομένων ως αρχάριος. Η Python σας καλύπτει με μια ποικιλία από εργαλεία απεικόνισης δεδομένων που διαθέτει. Μπορείτε να μετατρέψετε τα περίπλοκα σύνολα δεδομένων σας σε γραφικά, διαγράμματα ή διαδραστικά γραφήματα.

Όσο καλύτερα παρουσιάσετε τα δεδομένα στους υπεύθυνους λήψης αποφάσεων, τόσο καλύτερα θα κατανοήσουν τι καταφέρατε να αποσπάσετε από τον ιστό. Αυτό θα τους βοηθήσει να εντοπίσουν τις τάσεις και να δουν τι μπορούν να κάνουν για να αλλάξουν τις υπάρχουσες επιχειρηματικές στρατηγικές.

4.2.5 Εργαλεία εκτεταμένης ανάλυσης δεδομένων

Αν και η συλλογή δεδομένων είναι απαραίτητη για την ανάλυση δεδομένων, πρέπει επίσης να χειρίζεστε αποτελεσματικά τα εξαγόμενα δεδομένα. Η Python διαθέτει αρκετά ενσωματωμένα εργαλεία ανάλυσης που σας βοηθούν να εντοπίζετε μοτίβα και να εντοπίζετε χρήσιμες πληροφορίες για καλύτερες γνώσεις. Μπορείτε να χρησιμοποιήσετε εύχρηστα εργαλεία για να αξιολογήσετε την απόδοση μιας εταιρείας με βάση διάφορες κρίσιμες μετρήσεις.

4.3 Η Python ως εργαλείο του αναλυτή δεδομένων

Οι αναλυτές δεδομένων χρειάζονται μια αξιόπιστη πλατφόρμα για να γράφουν και να εκτελούν τον κώδικά τους. Η Python προσφέρει πολλά χαρακτηριστικά για την αποτελεσματική εκτέλεση τακτικών εργασιών ως αναλυτής δεδομένων. Θα μοιραστούμε μερικές συμβουλές για αρχάριους αναλυτές δεδομένων για να τους βοηθήσουμε να κατανοήσουν πώς να εργάζονται με την Python.

4.3.1 Οι βασικές βιβλιοθήκες της Python για ανάλυση δεδομένων

Οι βιβλιοθήκες της Python είναι γνωστό ότι απλοποιούν το έργο των αναλυτών δεδομένων. Ως εκ τούτου, είναι ζωτικής σημασίας η κατανόηση των πρωταρχικών χαρακτηριστικών τους [42].
Ορισμένες βιβλιοθήκες που μπορούν να σας βοηθήσουν στην ανάλυση δεδομένων είναι οι ακόλουθες.

- **Numpy.** Αυτή η βιβλιοθήκη της Python προσφέρει υπολογιστικά εργαλεία που σας βοηθούν να βελτιστοποιήσετε τις στατιστικές και μαθηματικές σας συναρτήσεις. Οι πίνακες Numpy (είτε πολυδιάστατοι είτε μονοδιάστατοι) σας επιτρέπουν να εφαρμόζετε αποτελεσματικά μια πράξη σε ολόκληρο τον πίνακα ταυτόχρονα, καθιστώντας τον κώδικα τόσο συντομότερο όσο και αποτελεσματικότερο. Επιπλέον, η Numpy διευκολύνει την εκτέλεση πράξεων πινάκων, οι οποίες είναι κεντρικές σε πολλούς αλγορίθμους μηχανικής μάθησης.

- **Matplotlib.** Η βιβλιοθήκη Matplotlib σας επιτρέπει να κάνετε οπτικοποιήσεις δεδομένων προσφέροντας αναπαραστάσεις βασισμένες σε γραφήματα. Σας επιτρέπει επίσης να ελέγχετε τις γραφικές παραστάσεις. Ως εκ τούτου, μπορείτε να αλλάξετε το σχήμα, το πάχος, τα χρώματα και το στυλ σύμφωνα με τις προτιμήσεις σας [43].
- **Η Pandas** είναι μία από τις σημαντικότερες βιβλιοθήκες της Python που αφορούν την ανάλυση δεδομένων. Περιλαμβάνει εργαλεία και δομές επεξεργασίας δεδομένων υψηλού επιπέδου που βοηθούν κατά τη διάρκεια των διαδικασιών καθαρισμού και επεξεργασίας δεδομένων. Μπορείτε να καθαρίσετε τα γεμάτα λάθη δεδομένα σας και να αφαιρέσετε ό,τι δεν χρειάζεται. Η βιβλιοθήκη επιτρέπει την αποθήκευση δεδομένων σε μορφή πίνακα με τη χρήση Data Frame [44].
- **Scikit-learn.** Αυτή η βιβλιοθήκη της Python σας βοηθά να εκτελέσετε μοντέλα ML (Machine Learning). Αυτοματοποιεί τη διαδικασία παραγωγής χρήσιμων πληροφοριών από τεράστιο όγκο δεδομένων.

Συνήθως θα βρείτε αυτές τις βιβλιοθήκες προεγκατεστημένες στο σύστημά σας, ειδικά αν έχετε χρησιμοποιήσει το Anaconda για να εγκαταστήσετε το Jupyter. Ωστόσο, αν χρειαστεί, μπορείτε πάντα να εγκαταστήσετε τις βιβλιοθήκες χρησιμοποιώντας την εντολή pip. Περισσότερα τεχνικά θέματα για την εγκατάσταση των απαραίτητων εργαλείων για την ανάλυση δεδομένων (Anaconda, Jupyter), αναλύονται και περιγράφονται σε επόμενη ενότητα.

4.3.2 Εξάσκηση με διαφορετικά σύνολα δεδομένων

Είναι σημαντικό να εξασκηθείτε με διαφορετικά σύνολα δεδομένων και να δείτε αν μπορείτε να διαχειριστείτε την ανάλυση δεδομένων. Θα βρείτε διάφορα σύνολα δεδομένων στη βιβλιοθήκη StatsModel της Python [45].

Μπορείτε να τα κατεβάσετε και να εφαρμόσετε θεμελιώδεις αναλυτικές και στατιστικές πράξεις. Αυτό θα σας βοηθήσει να παρακολουθείτε την πρόοδό σας ως αναλυτής δεδομένων και να εντοπίζετε τις αδυναμίες σας [46].
Ακολουθούν μερικές διαδικασίες για να εξασκηθείτε:

- **Καθαρισμός δεδομένων.** Πρόκειται για τον καθαρισμό των δεδομένων για την αφαίρεση σφαλμάτων και τη διόρθωση ανακριβειών που ενδέχεται να επηρεάσουν τα αποτελέσματα.
- **Προεπεξεργασία.** Εδώ πρέπει να μετατρέψετε τα δεδομένα σε μορφές κατάλληλες για την εκτέλεση εργασιών ανάλυσης δεδομένων με την Python.
- **Χειραγώγηση.** Πρόκειται για την εφαρμογή μοντέλων ML για την επίτευξη των επιθυμητών αποτελεσμάτων.
- **Οπτικοποίηση δεδομένων.** Εδώ θα μετατρέψετε τις πολύτιμες πληροφορίες σας σε κατανοητά διαγράμματα, γραφικά, χάρτες και άλλα.

4.4 Ρύθμιση του περιβάλλοντος Python

Αρχικά, πρέπει να δημιουργήσετε ένα περιβάλλον που να σας επιτρέπει να εργάζεστε εύκολα στην Python. Αρκετά προγράμματα σας προσφέρουν το προγραμματιστικό περιβάλλον για να κάνουν την εργασία σας εύκολη. Για παράδειγμα, η πλατφόρμα Python της Anaconda ανταποκρίνεται στις ανάγκες σας παρέχοντας διάφορες βιβλιοθήκες, όπως οι Numpy, IPython, Pandas, Matplotlib και άλλες.

Μπορείτε να κατεβάσετε και να εγκαταστήσετε το Anaconda στο σύστημά σας. Περιλαμβάνει διάφορα ενσωματωμένα προγράμματα, που σας επιτρέπουν να συγκεντρώνετε και να εκτελείτε τον

κώδικά σας χωρίς προβλήματα.

Για παράδειγμα, το Jupyter Notebook είναι ένα από τα πιο διαδεδομένα προγράμματα που μπορείτε να ανοίξετε στο πρόγραμμα περιήγησής σας και να χρησιμοποιήσετε.

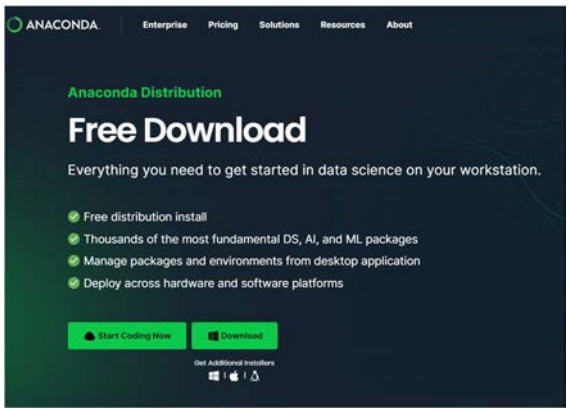
Σε αυτό την ενότητα, θα περιγράψουμε πως γίνεται η εγκατάσταση του Anaconda Navigator στον υπολογιστή μας, πως ρυθμίζουμε ένα περιβάλλον Python που περιλαμβάνει αρκετές κοινές βιβλιοθήκες και να ρυθμίσουμε το Jupyter Notebook.

4.4.1 Εγκαθιστώντας τη διανομή Anaconda

Το Anaconda είναι μια διανομή Python (προκατασκευασμένη και προκαθορισμένη συλλογή πακέτων) που χρησιμοποιείται συνήθως για την επιστήμη των δεδομένων. Η διανομή Anaconda περιλαμβάνει τον διαχειριστή πακέτων Conda εκτός από τα προρυθμισμένα πακέτα Python και άλλα εργαλεία. Ο διαχειριστής πακέτων Conda μπορεί να χρησιμοποιηθεί από τη γραμμή εντολών για τη δημιουργία περιβαλλόντων Python και την εγκατάσταση πρόσθετων πακέτων που συνοδεύουν την προεπιλεγμένη διανομή Anaconda.

Το Anaconda Navigator είναι ένα εργαλείο GUI που περιλαμβάνεται στη διανομή Anaconda και διευκολύνει τη ρύθμιση παραμέτρων, την εγκατάσταση και την εκκίνηση εργαλείων όπως το Jupyter Notebook.

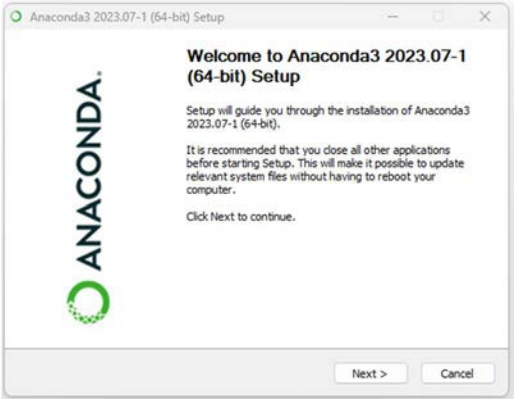
Για να ξεκινήσετε τη διαδικασία εγκατάστασης του Anaconda Navigator, επισκεφθείτε τη διεύθυνση https://www.anaconda.com/download και κάντε κλικ στο σύνδεσμο Download (Λήψεις) (βλ. Σχήμα 4-1).



Σχήμα 4-1. Σελίδα https://www.anaconda.com/download

Το αρχείο που θα κάνετε download είναι μεγέθους περίπου 900 MB.

Εντοπίστε το πρόγραμμα εγκατάστασης στο φάκελο λήψης του υπολογιστή σας και εκκινήστε το για να ξεκινήσει η διαδικασία εγκατάστασης (βλ. Σχήμα 4-2).



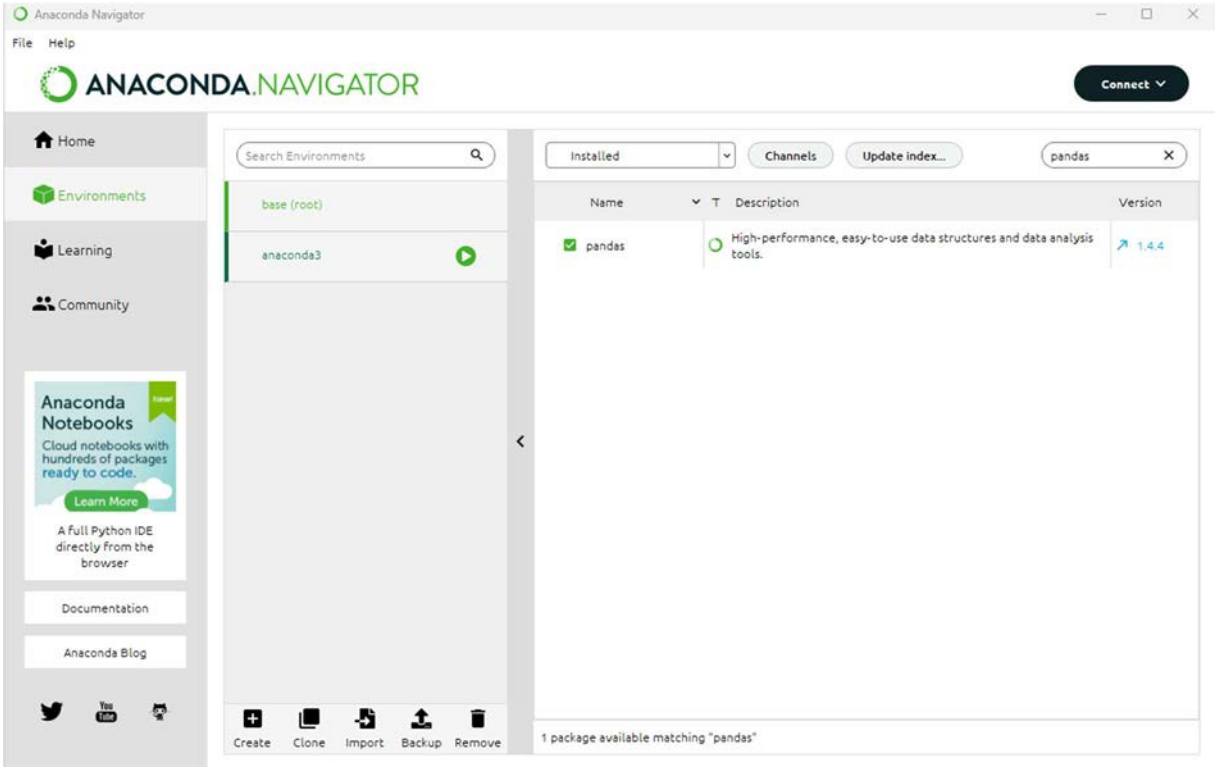
Σχήμα 4-2. Screenshot από την εγκατάσταση του Anaconda

Η διανομή Anaconda θα εγκατασταθεί στον υπολογιστή σας όταν το πρόγραμμα εγκατάστασης ολοκληρωθεί επιτυχώς.

Δημιουργία περιβάλλοντος Conda Python

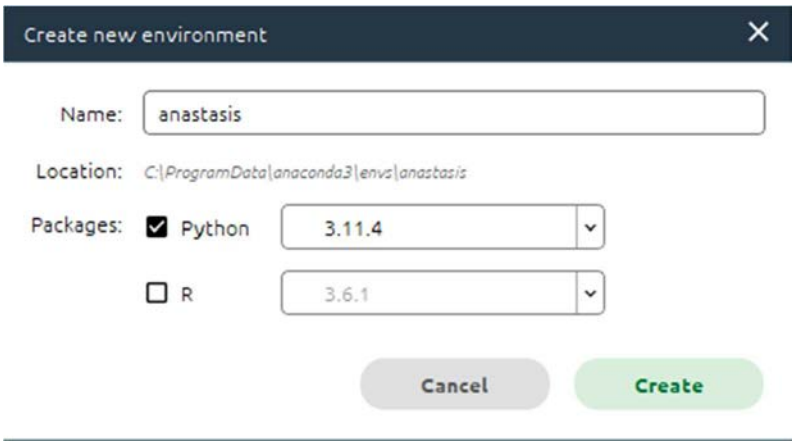
Μια νέα εγκατάσταση του Anaconda έρχεται με ένα μόνο περιβάλλον Conda Python που ονομάζεται base (root) και περιλαμβάνει ένα σύνολο προεγκατεστημένων πακέτων. Σε αυτή την ενότητα, θα δημιουργήσετε ένα νέο περιβάλλον Conda Python και θα εγκαταστήσετε ένα σύνολο πακέτων. Ένα περιβάλλον Conda Python είναι ένα απομονωμένο περιβάλλον και σας επιτρέπει να εγκαθιστάτε πακέτα χωρίς να τροποποιείτε την εγκατάσταση Python του συστήματός σας.
Ο επίσημα εγκεκριμένος διαχειριστής πακέτων της Python ονομάζεται PiP (Pip Installs Packages). Παίρνει τα πακέτα του από ένα διαδικτυακό αποθετήριο που ονομάζεται Python Package Index (PyPI). Το Conda είναι ένας διαχειριστής πακέτων χωρίς γλωσσική ανεξαρτησία και χρησιμοποιείται για την εγκατάσταση πακέτων μέσα σε περιβάλλοντα Conda. Αυτή είναι μια σημαντική διάκριση - το Conda δεν μπορεί να σας βοηθήσει να εγκαταστήσετε πακέτα στην εγκατάσταση Python του συστήματός σας. Ορισμένα εργαλεία που περιλαμβάνονται στο Anaconda Navigator, όπως το Jupyter Notebook και το Spyder IDE, θα εντοπίσουν τα περιβάλλοντα Conda που είναι διαθέσιμα στον υπολογιστή σας και θα σας επιτρέψουν την εναλλαγή μεταξύ διαφορετικών περιβαλλόντων.

Για να ξεκινήσετε τη δημιουργία ενός νέου περιβάλλοντος, εκκινήστε το Anaconda Navigator και μεταβείτε στην ενότητα Environments (Περιβάλλοντα) της διεπαφής χρήστη. Στη συνέχεια θα δείτε το βασικό (root) περιβάλλον που δημιουργήθηκε από τον εγκαταστάτη του Anaconda μαζί με όλα τα πακέτα που περιέχονται στο περιβάλλον. Κάντε κλικ στο κουμπί Create (Δημιουργία) για να δημιουργήσετε ένα νέο περιβάλλον Conda. (βλ. Σχήμα 4-3).



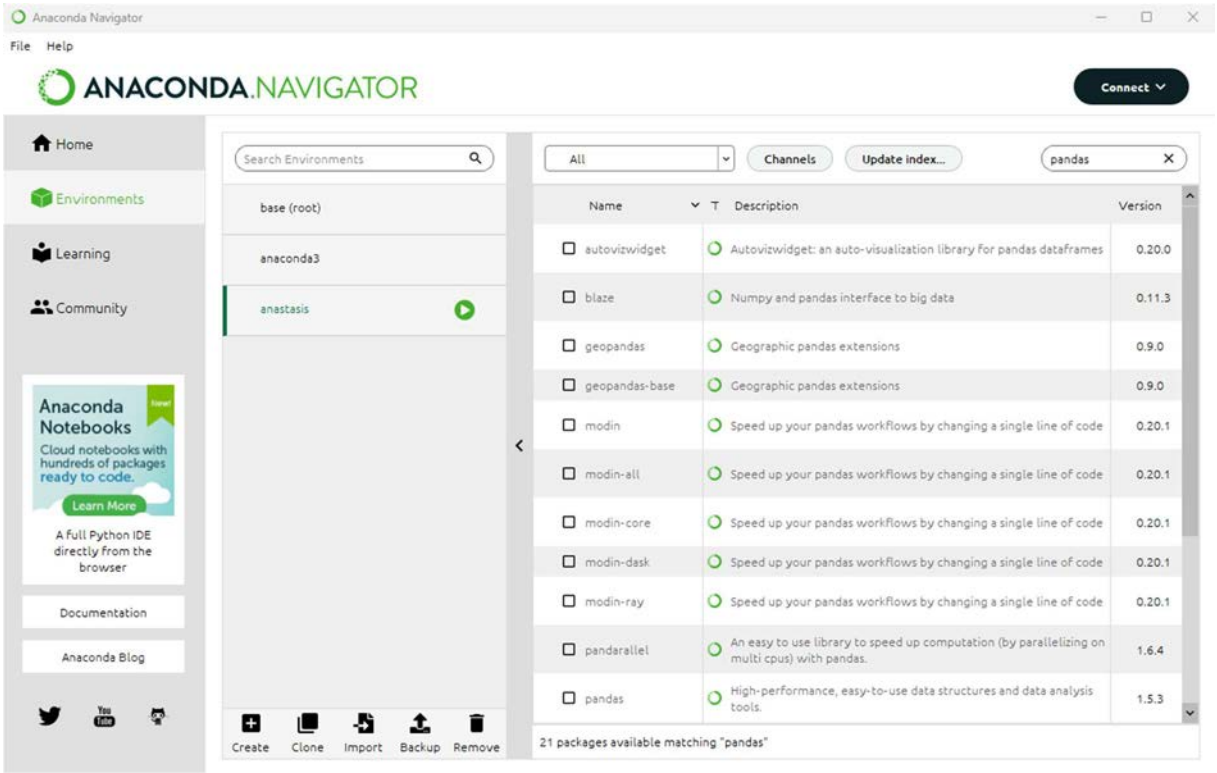
Σχήμα 4-3. Screenshot από τον Anaconda Navigator

Δώστε ένα όνομα για το νέο περιβάλλον Conda. Βεβαιωθείτε ότι η γλώσσα Python είναι η επιλεγμένη. Βεβαιωθείτε ότι το πλαίσιο ελέγχου της γλώσσας R είναι απενεργοποιημένο. Κάντε κλικ στο κουμπί Create (Δημιουργία) στο παράθυρο διαλόγου Create New Environment (Δημιουργία νέου περιβάλλοντος) για να ολοκληρώσετε τη δημιουργία του περιβάλλοντος. (βλ. Σχήμα 4-4).



Σχήμα 4-4. Screenshot από την δημιουργία νέου περιβάλλοντος στο Anaconda Navigator

Μετά από λίγα λεπτά, ένα νέο περιβάλλον Conda θα δημιουργηθεί στον υπολογιστή σας και θα το δείτε να εμφανίζεται στο Anaconda Navigator (βλ. Σχήμα 4-5).



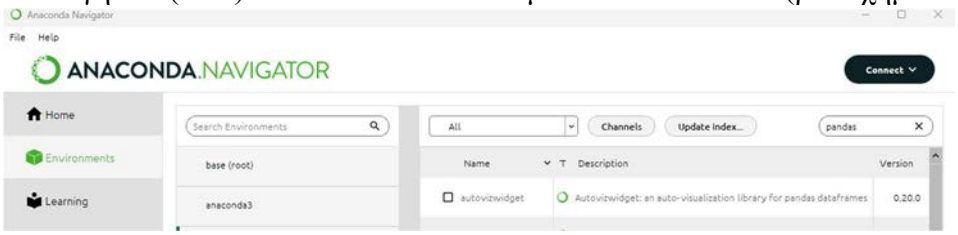
Σχήμα 4-5. Screenshot από το νέο περιβάλλον στο Anaconda Navigator

Κάνοντας κλικ στο νέο περιβάλλον θα εμφανιστεί μια λίστα με τα πακέτα σε αυτό το περιβάλλον. Από προεπιλογή, όλα τα νέα περιβάλλοντα Conda δημιουργούνται με ένα ελάχιστο σύνολο πακέτων.

Εγκατάσταση πακέτων Python

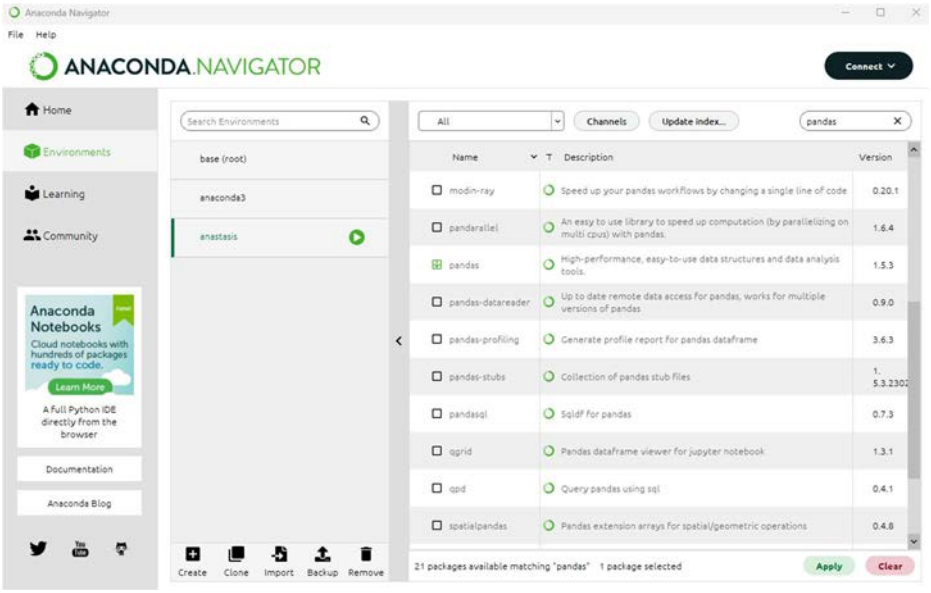
Σε αυτή την ενότητα, θα εγκαταστήσετε ορισμένα πακέτα Python στο νέο σας περιβάλλον Conda. Αρχικά, βεβαιωθείτε ότι το περιβάλλον Conda είναι επιλεγμένο στην εφαρμογή Anaconda Navigator.

Επιλέξτε την επιλογή All (Όλα) στο πλαίσιο συνδυασμού τύπου πακέτου (βλ. Σχήμα 4-6).



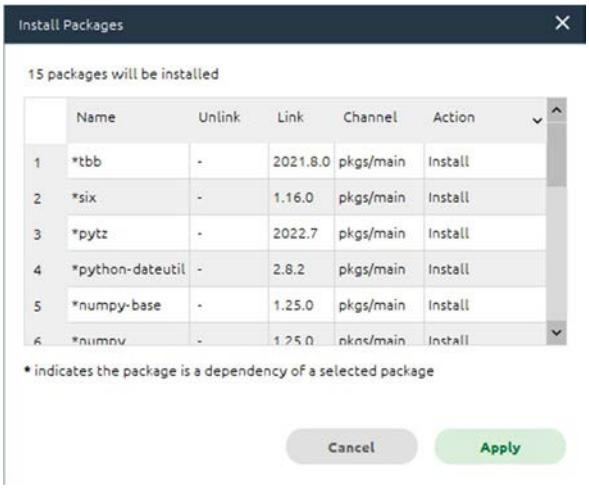
Σχήμα 4-6. Screenshot από το νέο περιβάλλον στο Anaconda Navigator

Χρησιμοποιήστε το πλαίσιο κειμένου αναζήτησης για να αναζητήσετε το πακέτο Pandas. Εντοπίστε το πακέτο Pandas στο αποτέλεσμα της αναζήτησης, επιλέξτε το και κάντε κλικ στο κουμπί Apply (Εφαρμογή) (βλ. Σχήμα 4-7).



Σχήμα 4-7. Screenshot από την εγκατάσταση της βιβλιοθήκης Pandas

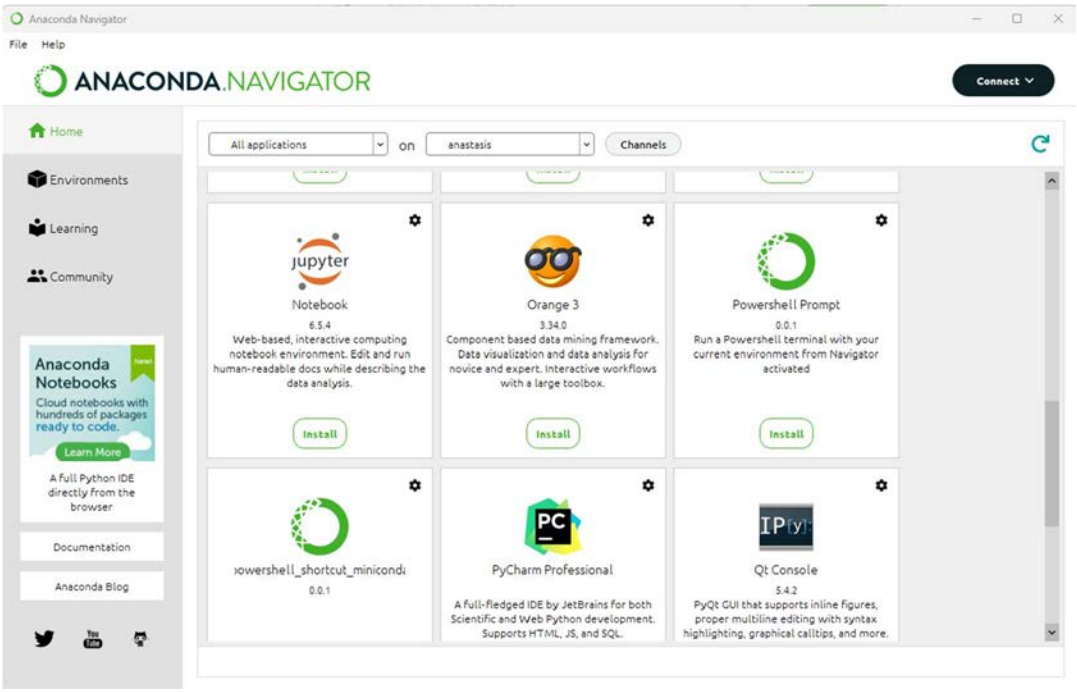
Το Anaconda Navigator θα παρουσιάσει ένα παράθυρο διαλόγου που παραθέτει το πακέτο Pandas και όλες τις εξαρτήσεις που θα εγκατασταθούν (βλ. Σχήμα 4-8). Κάντε κλικ στο κουμπί Apply (Εφαρμογή) για να ολοκληρώσετε την εγκατάσταση του πακέτου Pandas και των εξαρτήσεών του. Χρησιμοποιήστε την ίδια τεχνική για να εγκαταστήσετε και το πακέτο matplotlib μαζί με τις αντίστοιχες εξαρτήσεις του.



Σχήμα 4-8. Screenshot από την εγκατάσταση της βιβλιοθήκης Pandas

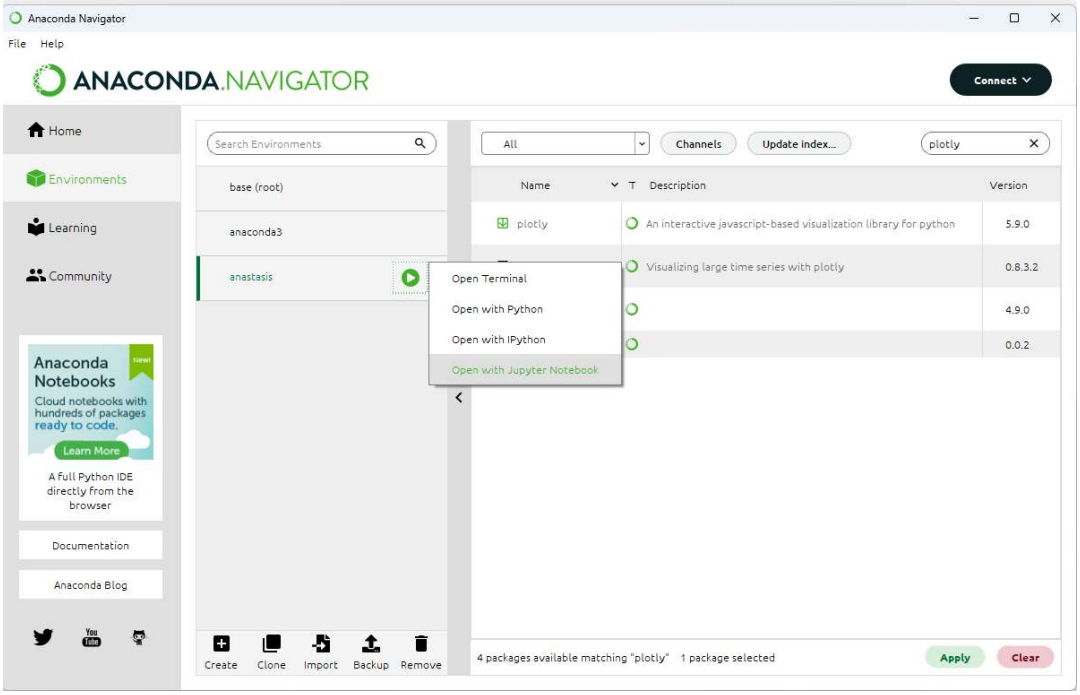
Εγκατάσταση του Jupyter Notebook

Σε αυτή την ενότητα, θα εγκαταστήσετε το Jupyter Notebook στο περιβάλλον Conda που δημιουργήσατε νωρίτερα σε αυτό το παράρτημα. Εκκινήστε το Anaconda Navigator και μεταβείτε στην καρτέλα Home. Εντοπίστε το εικονίδιο Jupyter Notebook και κάντε κλικ στο κουμπί Install (Εγκατάσταση) (βλ. Σχήμα 4-9). Αυτή η ενέργεια θα εγκαταστήσει το Jupyter Notebook στο προεπιλεγμένο βασικό (root) περιβάλλον.



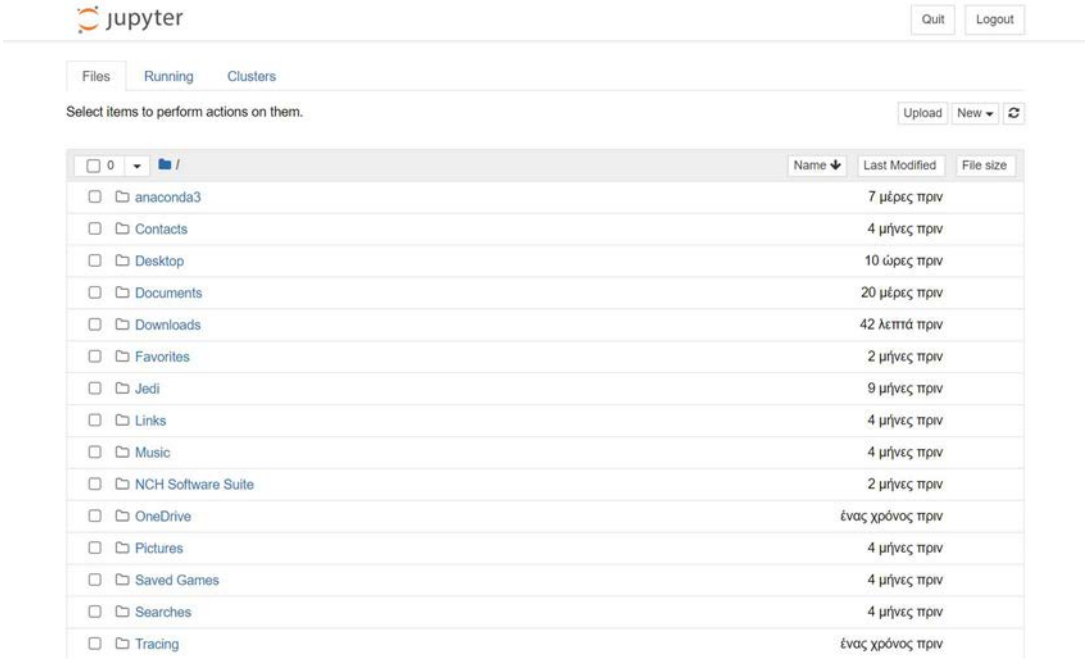
Σχήμα 4-9. Screenshot από την εγκατάσταση του Jupyter Notebook

Αφού εγκατασταθεί το Jupyter Notebook, ανοίξτε με την επιλογή Open with Jupyter Notebook.



Σχήμα 4-10. Screenshot από την εκκίνηση του Jupyter Notebook

Το σημειωματάριο εκκινείται στο προεπιλεγμένο πρόγραμμα περιήγησης στο διαδίκτυο και ρυθμίζεται έτσι ώστε να εμφανίζει τα περιεχόμενα του καταλόγου χρήστη σας (βλ. Σχήμα 4-11

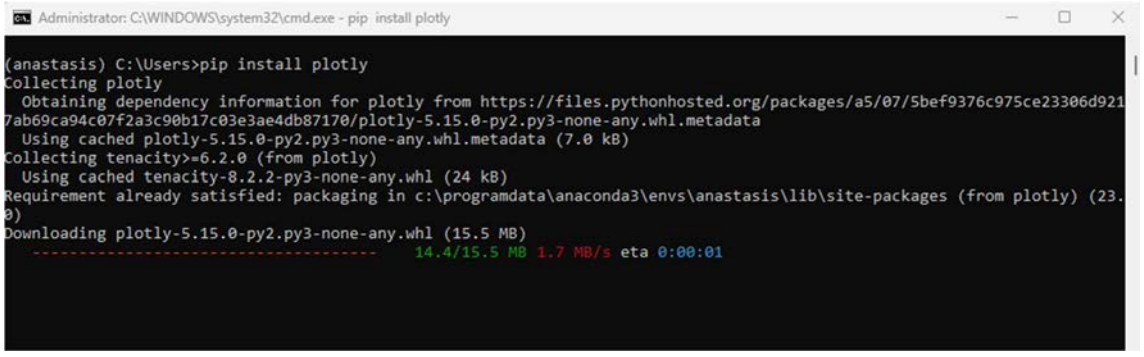


Σχήμα 4-11. Screenshot από το περιβάλλον του Jupyter Notebook

Κάνοντας κλικ στο κουμπί New (Νέο) από τη σελίδα Jupyter Notebook μέσα στο πρόγραμμα περιήγησής σας, θα εμφανιστεί ένα αναπτυσσόμενο μενού που θα σας επιτρέψει να δημιουργήσετε ένα νέο σημειωματάριο χρησιμοποιώντας έναν από τους πυρήνες που είναι εγκατεστημένοι στον υπολογιστή σας. Στο αναπτυσσόμενο μενού θα πρέπει να εμφανίζεται ένας πυρήνας που αντιστοιχεί στο νέο περιβάλλον Conda που δημιουργήσατε νωρίτερα σε αυτό το παράρτημα.

Μπορείτε επίσης να εκκινήσετε το Jupyter Notebook χρησιμοποιώντας τη γραμμή εντολών πληκτρολογώντας την ακόλουθη εντολή σε ένα παράθυρο Terminal (ή Command Prompt) και πατώντας Enter:
$ jupyter notebook

Αν οποιαδήποτε στιγμή θελήσουμε να κάνουμε εγκατάσταση ένα οποιοδήποτε νέο πακέτο αρκεί να είμαστε στο περιβάλλον που επιθυμούμε (Open Terminal) και με την εντολή pip install ακολουθούμενη από το όνομα του πακέτου επιτυγχάνεται η εγκατάστασή του.



Σχήμα 4-10. Screenshot από την εγκατάσταση του πακέτου plotly

ΚΕΦΑΛΑΙΟ 5

Μελέτη περίπτωσης

Η μελέτη περίπτωσης που παρουσιάζεται παρακάτω ασχολείται με την ανάλυση των χρηματιστηριακών δεδομένων 4 εισηγμένων εταιρειών στο Χρηματιστήριο Αξιών Αθηνών.

Αυτές είναι:

1. Η Εθνική Τράπεζα της Ελλάδος (ΕΤΕ – Τραπεζικός κλάδος)
2. Ο Οργανισμός Τηλεπικοινωνιών Ελλάδας (ΟΤΕ – Κλάδος Τηλεπικοινωνιών)
3. Η Δημόσια Επιχείρηση Ηλεκτρισμού (ΔΕΗ – Κλάδος Ενέργειας)
4. Ο Μυτιληναίος (ΜΥΤΙΛ – Κλάδος Ενέργειας)

Τα δεδομένα τα αντλήσαμε από το capital.gr [47] ένα από τα πιο έγκυρα και αξιόπιστα οικονομικά websites της Ελλάδας. Πρέπει να επισημάνουμε ότι τα δεδομένα από το συγκεκριμένο website είναι διαθέσιμα για μη εμπορική χρήση και μόνο για ακαδημαϊκούς σκοπούς.

Τα αρχεία είναι σε μορφή csv και ενδεικτικά έχουν την παρακάτω μορφή:

HistoryClosesETEall.csv

```
- ΕΤΕ -
Ημ/νία,Κλείσιμο,ΜΕΤ.%,Άνοιγμα,Υψηλό,Χαμηλό,Όγκος,Τζίρος
21/7/2023,"6,2100","-1,5848%","6,31","6,322","6,21",526.717,"3.303.527,97"
20/7/2023,"6,3100","1,4469%","6,24","6,42","6,16",772.694,"4.864.979,74"
19/7/2023,"6,2200","0,2902%","6,2","6,244","6,152",1.675.378,"10.417.698,25"
18/7/2023,"6,2020","3,3667%",6,"6,216","5,986",1.510.715,"9.293.977,41"
17/7/2023,"6,0000","-2,9126%","6,216","6,218","5,988",987.507,"6.005.352,73"
14/7/2023,"6,1800","-1,4354%","6,35","6,35","6,178",1.266.302,"7.889.724,03"
13/7/2023,"6,2700","-1,0417%","6,348","6,364","6,25",1.116.800,"7.043.899,20"
12/7/2023,"6,3360","-0,2205%","6,36","6,394","6,274",1.435.123,"9.112.104,70"
11/7/2023,"6,3500","4,0984%","6,12","6,382","6,1",2.715.495,"17.143.184,94"
10/7/2023,"6,1000","2,5210%","5,99","6,1","5,962",630.370,"3.820.003,90"
07/7/2023,"5,9500","0,0336%","5,948","5,984","5,896",705.728,"4.197.320,26"
06/7/2023,"5,9480","-1,5232%","6,024","6,024","5,874",733.692,"4.366.528,33"
05/7/2023,"6,0400","-0,3300%","6,06","6,08","5,978",1.825.049,"10.975.728,86"
04/7/2023,"6,0600","-0,9480%","6,12","6,128","5,978",818.792,"4.975.253,29"
03/7/2023,"6,1180","2,7545%","6,048","6,118","5,986",950.121,"5.771.127,48"
```

Πίνακας 5.1 Screenshot από το csv αρχείο με τα δεδομένα της ΕΤΕ

Όλα τα σύνολα δεδομένων είναι σε επίπεδο ημέρας και περιέχουν τα εξής πεδία:

- Ημερομηνία
- Τιμή Κλεισίματος
- Ημερήσια μεταβολή %
- Τιμή Ανοίγματος
- Υψηλότερη τιμή ημέρας
- Χαμηλότερη τιμή ημέρας
- Όγκος (σε τεμάχια)
- Τζίρος

Τα δεδομένα εκτείνονται σε βάθος 35ετίας. Η τελευταία ημέρα συναλλαγών που εργαστήκαμε ήταν η 21-7-2023.

Οι βιβλιοθήκες που χρησιμοποιούνται σε αυτό το έργο καθιστούν την ανάλυση και την οπτικοποίηση δεδομένων αρκετά απλή. Χρησιμοποιήθηκε η βιβλιοθήκη Pandas, για τον χειρισμό και

την ανάλυση δεδομένων καθώς και η βιβλιοθήκη Matplotlib για την οπτικοποίηση των δεδομένων (σχεδίαση γραφημάτων).

Αρχικά δημιουργήσαμε 4 dataframes από τα αντίστοιχα τέσσερα csv αρχεία. Το πρώτο θέμα που αντιμετωπίσαμε ήταν να συμπληρωθούν σωστά και να καθαριστούν τα δεδομένα των dataframes και γενικά να γίνει σωστή διαχείριση των ελλιπών τιμών (missing values – missing data). Επίσης έπρεπε να γίνουν οι απαραίτητες μετατροπές ώστε τα δεδομένα μας να παρουσιάζονται και να περιγράφονται από τους σωστούς τύπους δεδομένων.

Μετά από μία γενική επισκόπηση των δεδομένων μας σε όλο αυτό το χρονικό βάθος επιλέξαμε να εργαστούμε για τη χρονική περίοδο από τις αρχές του 2017 μέχρι και το τέλος του πρώτου εξαμήνου του 2023 (καθώς τα δεδομένα των τιμών των μετοχών είναι ένα είδος TimeSeries, τα δεδομένα από το πρόσφατο παρελθόν έχουν πολύ μεγαλύτερη σημασία σε σύγκριση με όλα τα ιστορικά δεδομένα).

Θελήσαμε να εξετάσουμε μία στρατηγική δημιουργίας ενός χαρτοφυλακίου που θα ακολουθούσε τον εξής κανόνα: θα γίνεται αγορά ενός τεμαχίου από κάθε μία μετοχή την τελευταία εργάσιμη ημέρα κάθε μήνα.

Με βάση αυτή τη στρατηγική, αναλύσαμε και ερευνήσαμε με σκοπό να εντοπίσουμε τυχόν ενδιαφέροντα μοτίβα των μετοχών, εξετάσαμε τη συμπεριφορά που είχαν οι μετοχές την περίοδο της κρίσης της πανδημίας, της κρίσης του πολέμου Ρωσίας Ουκρανίας, και εν γένει αποτιμήσαμε αν αυτή η στρατηγική ήταν αποδοτική ή όχι.

Επίσης να αναφέρουμε ότι σε κάθε στάδιο της ανάλυσης των δεδομένων μας οι χρήση των συναρτήσεων από τη βιβλιοθήκη matplotlib αποτέλεσε βασικό εργαλείο οπτικοποίησης των αποτελεσμάτων της ανάλυσης μας

Τέλος, αυτό το έργο είναι μόνο ένα μικρό μέρος της ανάλυσης δεδομένων της χρηματιστηριακής αγοράς με τη βοήθεια της Python, καθώς η ανάλυση μετοχών περιλαμβάνει τόσο την τεχνική όσο και τη θεμελιώδη ανάλυση, η οποία είναι ένας ευρύς τομέας.

Ο κώδικας

Αρχικά κάνουμε import τις βιβλιοθήκες (libraries) pandas, matplotlib καθώς και τη συνάρτηση datetime από τη βιβλιοθήκη datetime.

```
In [1]: import pandas as pd
        import matplotlib.pyplot as plt
        from datetime import datetime
```

Στη συνέχεια δημιουργούμε 4 dataframes από ισάριθμα csv αρχεία που περιέχουν τα δεδομένα των τεσσάρων μετοχών (ΕΤΕ, ΟΤΕ, ΔΕΗ, ΜΥΤΙΛΗΝΑΙΟΣ) αντίστοιχα. Το σύμβολο (,) το αντιστοιχούμε στην υποδιαστολή και προσπερνάμε την πρώτη γραμμή του αρχείου που περιέχει το όνομα της μετοχής.

```
In [2]: df1=pd.read_csv('HistoryClosesETEall.csv',thousands='.', decimal=',', skiprows=1)
        df2=pd.read_csv('HistoryClosesOTEall.csv',thousands='.', decimal=',', skiprows=1)
        df3=pd.read_csv('HistoryClosesΔΕΗall.csv',thousands='.', decimal=',', skiprows=1)
        df4=pd.read_csv('HistoryClosesΜΥΤΙΛall.csv',thousands='.', decimal=',', skiprows=1)
```

Εμφανίζουμε το καθένα από τα dataframes για να έχουμε μια πρώτη εικόνα αυτών. Γράφοντας μόνο το dataframe εμφανίζονται οι 5 πρώτες και 5 τελευταίες εγγραφές σε αυτό.

```
In [3]: df1
```

Out[3]:

	Ημ/νία	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
0	21/7/2023	6.210	-1,5848%	6.310	6.322	6.210	526717.0	3303527.97
1	20/7/2023	6.310	1,4469%	6.240	6.420	6.160	772694.0	4864979.74
2	19/7/2023	6.220	0,2902%	6.200	6.244	6.152	1675378.0	10417698.25
3	18/7/2023	6.202	3,3667%	6.000	6.216	5.986	1510715.0	9293977.41
4	17/7/2023	6.000	-2,9126%	6.216	6.218	5.988	987507.0	6005352.73
...	...	...	...	...	...	...	...	...
9317	09/1/1986	2225.608	-0,3365%	2233.122	2233.122	2225.608	NaN	0.00
9318	08/1/1986	2233.122	0,3376%	2225.608	2233.122	2225.608	NaN	0.00
9319	07/1/1986	2225.608	0,0000%	2225.608	2225.608	2225.608	NaN	0.00
9320	03/1/1986	2225.608	0,0000%	2225.608	2225.608	2225.608	NaN	0.00
9321	02/1/1986	2225.608	0,0000%	2225.608	2225.608	2225.608	NaN	0.00

9322 rows × 8 columns

Εναλλακτικά αν θέλουμε να εμφανίσουμε μόνο τις 5 πρώτες εγγραφές χρησιμοποιούμε τη συνάρτηση .head()

```
In [4]: df2.head()
```

Out[4]:

	Ημ/νία	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
0	21/7/2023	14.46	-1,3643%	14.56	14.78	14.46	302714	4418499.54
1	20/7/2023	14.66	0,4110%	14.60	14.91	14.60	349915	5158690.77
2	19/7/2023	14.60	3,1073%	14.22	14.83	14.18	416552	6057419.93
3	18/7/2023	14.16	-1,9391%	14.45	14.50	14.13	616644	8785942.12
4	17/7/2023	14.44	-1,1636%	14.61	14.69	14.44	350657	5113042.16

Αντίθετα αν θέλουμε να εμφανίσουμε μόνο τις 5 τελευταίες εγγραφές χρησιμοποιούμε τη συνάρτηση .tail()

```
In [5]: df2.tail()
```

Out[5]:

	Ημ/νία	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
6762	25/4/1996	12.09	-2,1053%	12.35	12.35	12.05	223572	0.0
6763	24/4/1996	12.35	-0,7235%	12.44	12.48	12.19	402983	0.0
6764	23/4/1996	12.44	-0,7183%	12.53	12.53	12.09	677540	0.0
6765	22/4/1996	12.53	0,0000%	12.53	13.16	12.53	1096191	0.0
6766	19/4/1996	12.53	0,0000%	12.53	12.53	12.22	1515533	0.0

Συνεχίζουμε με την εμφάνιση του dataframe της μετοχής της ΔΕΗ.

```
In [6]: df3
```

Out[6]:

	Ημ/νία	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
0	21/7/2023	11.12	3,0584%	10.76	11.12	10.76	718299.0	7.898804e+06
1	20/7/2023	10.79	0,3721%	10.75	10.89	10.66	294236.0	3.166867e+06
2	19/7/2023	10.75	3,3654%	10.38	10.82	10.38	641175.0	6.835485e+06
3	18/7/2023	10.40	1,4634%	10.30	10.46	10.12	352796.0	3.633993e+06
4	17/7/2023	10.25	-1,8199%	10.38	10.52	10.20	276116.0	2.859946e+06
...	...	...	...	...	...	...	...	...
5347	18/12/2001	12.20	2,6936%	11.94	12.30	11.88	435262.0	5.249204e+06
5348	17/12/2001	11.88	0,6780%	11.80	12.00	11.76	412250.0	4.892755e+06
5349	14/12/2001	11.80	0,5111%	11.60	11.94	11.60	439880.0	5.202276e+06
5350	13/12/2001	11.74	-2,6534%	12.00	12.00	11.66	1491111.0	1.758379e+07
5351	12/12/2001	12.06	0,0000%	12.32	12.46	12.00	20905755.0	2.597361e+08

5352 rows × 8 columns

Τέλος εμφανίζουμε το dataframe της μετοχής του ΜΥΤΙΛΗΝΑΙΟΥ.

```
In [7]: df4
```

Out[7]:

	Ημ/νία	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
0	21/7/2023	35.820	-1,3223%	36.300	36.400	35.820	145221	5235476.58
1	20/7/2023	36.300	1,7377%	36.000	36.300	35.820	216367	7820437.02
2	19/7/2023	35.680	3,3005%	35.100	36.120	34.880	293517	10527500.36
3	18/7/2023	34.540	3,7237%	33.340	35.000	33.340	412590	14278135.60
4	17/7/2023	33.300	1,4625%	32.980	33.300	32.700	171926	5683682.32
...	...	...	...	...	...	...	...	...
6934	04/8/1995	0.656	1,5480%	0.646	0.656	0.646	182	0.00
6935	03/8/1995	0.646	1,4129%	0.637	0.646	0.637	261	0.00
6936	02/8/1995	0.637	1,5949%	0.627	0.637	0.627	87	0.00
6937	01/8/1995	0.627	1,6207%	0.617	0.627	0.617	105	0.00
6938	31/7/1995	0.617	0,0000%	0.617	0.617	0.617	87	0.00

6939 rows × 8 columns

Με τη συνάρτηση .info() εμφανίζουμε όλα τα πεδία (fields), πόσες είναι οι μη κενές τιμές σε καθένα από αυτά και τον τύπο δεδομένων για αυτά.

```
In [8]: df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9322 entries, 0 to 9321
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ημ/νία    9322 non-null   object
 1   Κλείσιμο  9322 non-null   float64
 2   ΜΕΤ.%     9322 non-null   object
 3   Άνοιγμα   9322 non-null   float64
 4   Υψηλό     9322 non-null   float64
 5   Χαμηλό    9322 non-null   float64
 6   Όγκος     9010 non-null   float64
 7   Τζίρος    9322 non-null   float64
dtypes: float64(6), object(2)
memory usage: 582.8+ KB
```

Όπως παρατηρούμε από τη παραπάνω έξοδο το πεδίο MET.% που εκφράζει την ποσοστιαία ημερησία μεταβολή της τιμής της μετοχής (%), είναι τύπου object δηλαδή string. Επειδή θέλουμε το συγκεκριμένο πεδίο να είναι τύπου float θα χρειαστεί πρώτα να κάνουμε αντικατάσταση τον χαρακτήρα % σε 0 και το χαρακτήρα ‘,’ σε ‘.’ Αν δεν προηγηθούν αυτές οι τροποποιήσεις δεν μπορούμε να μετατρέψουμε ένα string σε float επειδή περιέχεται κάποιος χαρακτήρας (%) και η υποδιαστολή δεν αναγνωρίζεται με το απλό κόμμα (,) αλλά με την τελεία (.)

```
In [9]: df1['MET.%'] = df1['MET.%'].str.replace('%','0')
        df1['MET.%'] = df1['MET.%'].str.replace(',','.')
```

Στη συνέχεια μπορούμε να μετατρέψουμε το πεδίο από string σε float.

```
In [10]: df1['MET.%'] = df1['MET.%'].astype('float')
```

Εμφανίζοντας τις πληροφορίες των πεδίων του dataframe παρατηρούμε ότι το πεδίο MET.% έγινε τύπου float.

```
In [11]: df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9322 entries, 0 to 9321
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ημ/νία    9322 non-null   object
 1   Κλείσιμο  9322 non-null   float64
 2   MET.%     9322 non-null   float64
 3   Άνοιγμα   9322 non-null   float64
 4   Υψηλό     9322 non-null   float64
 5   Χαμηλό    9322 non-null   float64
 6   Όγκος     9010 non-null   float64
 7   Τζίρος    9322 non-null   float64
dtypes: float64(7), object(1)
memory usage: 582.8+ KB
```

Αντίστοιχες ενέργειες θα κάνουμε και για τα υπόλοιπα dataframes.

```
In [12]: df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6767 entries, 0 to 6766
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ημ/νία    6767 non-null   object
 1   Κλείσιμο  6767 non-null   float64
 2   MET.%     6767 non-null   object
 3   Άνοιγμα   6767 non-null   float64
 4   Υψηλό     6767 non-null   float64
 5   Χαμηλό    6767 non-null   float64
 6   Όγκος     6767 non-null   int64
 7   Τζίρος    6767 non-null   float64
dtypes: float64(5), int64(1), object(2)
memory usage: 423.1+ KB
```

```
In [13]: df2['MET.%'] = df2['MET.%'].str.replace('%','0')
         df2['MET.%'] = df2['MET.%'].str.replace(',','.')
         df2['MET.%'] = df2['MET.%'].astype('float')
```

In [14]:
```
df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6767 entries, 0 to 6766
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ημ/νία    6767 non-null   object
 1   Κλείσιμο  6767 non-null   float64
 2   MET.%     6767 non-null   object
 3   Άνοιγμα   6767 non-null   float64
 4   Υψηλό     6767 non-null   float64
 5   Χαμηλό    6767 non-null   float64
 6   Όγκος     6767 non-null   int64
 7   Τζίρος    6767 non-null   float64
dtypes: float64(5), int64(1), object(2)
memory usage: 423.1+ KB
```

In [15]:
```
df3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5352 entries, 0 to 5351
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ημ/νία    5352 non-null   object
 1   Κλείσιμο  5352 non-null   float64
 2   MET.%     5352 non-null   object
 3   Άνοιγμα   5352 non-null   float64
 4   Υψηλό     5352 non-null   float64
 5   Χαμηλό    5352 non-null   float64
 6   Όγκος     5350 non-null   float64
 7   Τζίρος    5352 non-null   float64
dtypes: float64(6), object(2)
memory usage: 334.6+ KB
```

In [16]:
```
df3['MET.%'] = df3['MET.%'].str.replace('%','0')
df3['MET.%'] = df3['MET.%'].str.replace(',','.')
df3['MET.%'] = df3['MET.%'].astype('float')
```

In [17]:
```
df3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5352 entries, 0 to 5351
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ημ/νία    5352 non-null   object
 1   Κλείσιμο  5352 non-null   float64
 2   MET.%     5352 non-null   float64
 3   Άνοιγμα   5352 non-null   float64
 4   Υψηλό     5352 non-null   float64
 5   Χαμηλό    5352 non-null   float64
 6   Όγκος     5350 non-null   float64
 7   Τζίρος    5352 non-null   float64
dtypes: float64(7), object(1)
memory usage: 334.6+ KB
```

```
In [18]: df4.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6939 entries, 0 to 6938
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ημ/νία    6939 non-null   object
 1   Κλείσιμο  6939 non-null   float64
 2   MET.%     6939 non-null   object
 3   Άνοιγμα   6939 non-null   float64
 4   Υψηλό     6939 non-null   float64
 5   Χαμηλό    6939 non-null   float64
 6   Όγκος     6939 non-null   int64
 7   Τζίρος    6939 non-null   float64
dtypes: float64(5), int64(1), object(2)
memory usage: 433.8+ KB
```

```
In [19]: df4['MET.%'] = df4['MET.%'].str.replace('%','0')
         df4['MET.%'] = df4['MET.%'].str.replace(',','.')
         df4['MET.%'] = df4['MET.%'].astype('float')
```

```
In [20]: df4.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6939 entries, 0 to 6938
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ημ/νία    6939 non-null   object
 1   Κλείσιμο  6939 non-null   float64
 2   MET.%     6939 non-null   float64
 3   Άνοιγμα   6939 non-null   float64
 4   Υψηλό     6939 non-null   float64
 5   Χαμηλό    6939 non-null   float64
 6   Όγκος     6939 non-null   int64
 7   Τζίρος    6939 non-null   float64
dtypes: float64(6), int64(1), object(1)
memory usage: 433.8+ KB
```

Για να μπορέσουμε να δούμε την έκταση που έχει ένα dataframe, δηλαδή πόσες γραμμές και πόσες στήλες έχει χρησιμοποιούμε την εντολή dataframe.shape η οποία μας επιστρέφει μια πλειάδα (tuple) που αντιπροσωπεύει τη διάσταση του dataFrame.

```
In [21]: df1.shape
Out[21]: (9322, 8)

In [22]: df2.shape
Out[22]: (6767, 8)

In [23]: df3.shape
Out[23]: (5352, 8)

In [24]: df4.shape
Out[24]: (6939, 8)
```

Αρκετές φορές από τα δεδομένα που λαμβάνουμε από τα αρχεία δεδομένων (στην προκειμένη περίπτωση από τα .csv αρχεία) χρειάζεται να μετονομάσουμε κάποια πεδία. Έτσι για τη δική μας ανάλυση μετονομάζουμε το πεδίο ‘Ημ/νία’ σε ‘Ημερομηνια’.

```
In [25]: df1=df1.rename(columns={'Ημ/νία':"Ημερομηνια"})
         df2=df2.rename(columns={'Ημ/νία':"Ημερομηνια"})
         df3=df3.rename(columns={'Ημ/νία':"Ημερομηνια"})
         df4=df4.rename(columns={'Ημ/νία':"Ημερομηνια"})
```

Στη συνέχεια μετατρέπουμε το πεδίο ‘Ημερομηνια’ να είναι τύπου datetime ώστε η συμπεριφορά των δεδομένων μας να είναι αντικειμενικά σαν ημερομηνίες.

```
In [26]: df1['Ημερομηνια'] = pd.to_datetime(df1['Ημερομηνια'], format='%d/%m/%Y')
         df2['Ημερομηνια'] = pd.to_datetime(df2['Ημερομηνια'], format='%d/%m/%Y')
         df3['Ημερομηνια'] = pd.to_datetime(df3['Ημερομηνια'], format='%d/%m/%Y')
         df4['Ημερομηνια'] = pd.to_datetime(df4['Ημερομηνια'], format='%d/%m/%Y')
```

Εμφανίζοντας τους τύπους δεδομένων των πεδίων των dataframes θα δούμε ότι μετατράπηκαν οι ημερομηνίες σε τύπο datetime64.

```
In [27]: print (df1.dtypes)
```

```
Ημερομηνια    datetime64[ns]
Κλείσιμο             float64
MET.%                float64
Άνοιγμα              float64
Υψηλό                float64
Χαμηλό               float64
Όγκος                float64
Τζίρος               float64
dtype: object
```

```
In [28]: print (df2.dtypes)
```

```
Ημερομηνια    datetime64[ns]
Κλείσιμο             float64
MET.%                 object
Άνοιγμα              float64
Υψηλό                float64
Χαμηλό               float64
Όγκος                  int64
Τζίρος               float64
dtype: object
```

```
In [29]: print (df3.dtypes)
```

```
Ημερομηνια    datetime64[ns]
Κλείσιμο             float64
MET.%                float64
Άνοιγμα              float64
Υψηλό                float64
Χαμηλό               float64
Όγκος                float64
Τζίρος               float64
dtype: object
```

```
In [30]: print (df4.dtypes)
```

```
Ημερομηνια    datetime64[ns]
Κλείσιμο             float64
MET.%                float64
Άνοιγμα              float64
Υψηλό                float64
Χαμηλό               float64
Όγκος                  int64
Τζίρος               float64
dtype: object
```

Ταξινομούμε τα dataframes με βάση την ημερομηνία σε αύξουσα σειρά δηλαδή από την παλιότερη προς τη νεότερη.

```
In [31]: df1=df1.sort_values(by=['Ημερομηνια'])
         df2=df2.sort_values(by=['Ημερομηνια'])
         df3=df3.sort_values(by=['Ημερομηνια'])
         df4=df4.sort_values(by=['Ημερομηνια'])
```

Εμφανίζουμε ενδεικτικά το 1ο dataframe (της μετοχής της ΕΤΕ) για να διαπιστώσουμε ότι οι προηγούμενες ενέργειές μας έχουν εφαρμοστεί.

```
In [32]: df1
```

Out[32]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
9321	1986-01-02	2225.608	0.0000	2225.608	2225.608	2225.608	NaN	0.00
9320	1986-01-03	2225.608	0.0000	2225.608	2225.608	2225.608	NaN	0.00
9319	1986-01-07	2225.608	0.0000	2225.608	2225.608	2225.608	NaN	0.00
9318	1986-01-08	2233.122	0.3376	2225.608	2233.122	2225.608	NaN	0.00
9317	1986-01-09	2225.608	-0.3365	2233.122	2233.122	2225.608	NaN	0.00
...	...	...	...	...	...	...	...	...
4	2023-07-17	6.000	-2.9126	6.216	6.218	5.988	987507.0	6005352.73
3	2023-07-18	6.202	3.3667	6.000	6.216	5.986	1510715.0	9293977.41
2	2023-07-19	6.220	0.2902	6.200	6.244	6.152	1675378.0	10417698.25
1	2023-07-20	6.310	1.4469	6.240	6.420	6.160	772694.0	4864979.74
0	2023-07-21	6.210	-1.5848	6.310	6.322	6.210	526717.0	3303527.97

9322 rows × 8 columns

Παρατηρούμε ότι στις πρώτες ημερομηνίες (αρχές του έτους 1996) στις τιμές του πεδίου ‘Όγκος’ υπάρχουν οι τιμές NaN. Αυτός ο συμβολισμός σημαίνει ότι λείπουν τιμές σε αυτές τις περιπτώσεις. Για να εντοπίσουμε αυτές τις ελλιπείς τιμές (missing values – missing data) αρχικά μπορούμε να δούμε αυτές με τη συνάρτηση .isna()

```
In [33]: df1.isna()
```

Out[33]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
9321	False	False	False	False	False	False	True	False
9320	False	False	False	False	False	False	True	False
9319	False	False	False	False	False	False	True	False
9318	False	False	False	False	False	False	True	False
9317	False	False	False	False	False	False	True	False
...	...	...	...	...	...	...	...	...
4	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False
0	False	False	False	False	False	False	False	False

9322 rows × 8 columns

Όπως παρατηρούμε στο παραπάνω dataframe εκεί που λείπουν οι τιμές έχουμε την τιμή True ενώ όταν υπάρχει τιμή έχουμε το χαρακτηρισμό False. Για να καταμετρήσουμε πόσες είναι χαμένες τιμές σε όλο το dataframe χρησιμοποιούμε το συνδυασμό των συναρτήσεων .isna().sum()

```
In [34]: df1.isna().sum()

Out[34]: Ημερομηνια      0
         Κλείσιμο        0
         MET.%           0
         Άνοιγμα         0
         Υψηλό           0
         Χαμηλό          0
         Όγκος         312
         Τζίρος          0
         dtype: int64
```

Επομένως παρατηρούμε ότι υπάρχουν 312 missing values στο πεδίο 'Όγκος'. Για λόγους επίδειξης των τεχνικών διαχείρισης των missing values αρχικά δημιουργούμε ένα νέο dataframe με όνομα df1_test ως πιστό αντίγραφο του df1.

```
In [35]: df1_test=df1
```

Κατόπιν, ταξινομούμε το df1_test με βάση την 'Ημερομηνια' σε αύξουσα σειρά (default η ταξινόμηση όταν γίνεται είναι σε αύξουσα σειρά. Αν χρειάζεται φθίνουσα διάταξη των δεδομένων μας επιλέγουμε την παράμετρο ascending=False).

```
In [36]: df1_test=df1_test.sort_values(by=['Ημερομηνια'])
```

Εμφανίζουμε το dataframe df1_test για να διαπιστώσουμε την ορθότητα των ενεργειών μας.

```
In [37]: df1_test
```

Out[37]:

	Ημερομηνια	Κλείσιμο	MET.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
9321	1986-01-02	2225.608	0.0000	2225.608	2225.608	2225.608	NaN	0.00
9320	1986-01-03	2225.608	0.0000	2225.608	2225.608	2225.608	NaN	0.00
9319	1986-01-07	2225.608	0.0000	2225.608	2225.608	2225.608	NaN	0.00
9318	1986-01-08	2233.122	0.3376	2225.608	2233.122	2225.608	NaN	0.00
9317	1986-01-09	2225.608	-0.3365	2233.122	2233.122	2225.608	NaN	0.00
...	...	...	...	...	...	...	...	...
4	2023-07-17	6.000	-2.9126	6.216	6.218	5.988	987507.0	6005352.73
3	2023-07-18	6.202	3.3667	6.000	6.216	5.986	1510715.0	9293977.41
2	2023-07-19	6.220	0.2902	6.200	6.244	6.152	1675378.0	10417698.25
1	2023-07-20	6.310	1.4469	6.240	6.420	6.160	772694.0	4864979.74
0	2023-07-21	6.210	-1.5848	6.310	6.322	6.210	526717.0	3303527.97

9322 rows × 8 columns

Η μέθοδος pandas series.first_valid_index() χρησιμοποιείται για να ληφθεί ο δείκτης των πρώτων έγκυρων δεδομένων. Αυτό σημαίνει ότι η μέθοδος first_valid_index() επιστρέφει το δείκτη του πρώτου μη μηδενικού στοιχείου της σειράς.

```
In [38]: df1_test.first_valid_index()

Out[38]: 9321
```

Για να γίνει πιο σαφές το εγχείρημα της διαχείρισης των ελλιπών τιμών (NaN) επιλέγουμε ένα ημερολογιακό διάστημα τιμών όπου μεταξύ κάποιων σωστών τιμών εμφανίζεται και μια ελλιπής τιμή.

```
In [39]: df1_test.loc[(df1_test['Ημερομηνια'] >= '1987/11/24')
                      & (df1_test['Ημερομηνια'] <= '1987/11/30')]
```

Out[39]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
8854	1987-11-24	2481.247	-1.1974	2511.318	2511.318	2481.247	13.0	0.0
8853	1987-11-25	2481.247	0.0000	2481.247	2481.247	2481.247	42.0	0.0
8852	1987-11-26	2466.204	-0.6063	2481.247	2481.247	2458.690	NaN	0.0
8851	1987-11-27	2451.176	-0.6094	2466.204	2466.204	2443.647	78.0	0.0
8850	1987-11-30	2451.176	0.0000	2451.176	2458.690	2443.647	85.0	0.0

Όπως βλέπουμε παραπάνω στις 26-11-1987 στην στήλη του Όγκου υπάρχει μια ελλιπής τιμή. Υπάρχουν διάφορες μέθοδοι συμπλήρωσης των τιμών αυτών. Η μέθοδος 'ffill' όταν εφαρμόζεται συμπληρώνει τα missing values με τις αμέσως προηγούμενες σωστές τιμές που διαθέτει. Με την παράμετρο inplace=True καταφέρνουμε οι αλλαγές αυτές να εφαρμοστούν και να ενσωματωθούν στο dataframe που χειριζόμαστε.

```
In [40]: df1_test["Όγκος"].fillna( method='ffill', inplace = True)
```

Επιλέγουμε να εμφανίσουμε στο ίδιο διάστημα που αναφερθήκαμε τα περιεχόμενα του df1_test.

```
In [41]: df1_test.loc[(df1_test['Ημερομηνια'] >= '1987/11/24')
                      & (df1_test['Ημερομηνια'] <= '1987/11/30')]
```

Out[41]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
8854	1987-11-24	2481.247	-1.1974	2511.318	2511.318	2481.247	13.0	0.0
8853	1987-11-25	2481.247	0.0000	2481.247	2481.247	2481.247	42.0	0.0
8852	1987-11-26	2466.204	-0.6063	2481.247	2481.247	2458.690	42.0	0.0
8851	1987-11-27	2451.176	-0.6094	2466.204	2466.204	2443.647	78.0	0.0
8850	1987-11-30	2451.176	0.0000	2451.176	2458.690	2443.647	85.0	0.0

Παρατηρούμε ότι ο Όγκος στις 26-11-1987 συμπληρώθηκε με την τιμή 42 η οποία ήταν η αμέσως προηγούμενη έγκυρη τιμή (δηλ. της ημερομηνίας 25-11-1987).

Στη συνέχεια μετατρέπουμε το πεδίο 'Όγκος' σε τύπου δεδομένων ακεραίων αριθμών (int64). Η παράμετρος errors='ignore' αγνοεί τυχόν λάθη που μπορεί να εμφανιστούν κατά τη μετατροπή του τύπου δεδομένων επειδή μπορεί να υπάρχουν ακόμη ελλιπείς τιμές μέσα στο dataframe. Παρακάτω θα εξηγήσουμε γιατί μπορεί να συμβαίνει κάτι τέτοιο.

```
In [42]: df1_test['Όγκος']=df1_test['Όγκος'].astype('Int64', errors='ignore')
```

Με την παρακάτω εντολή df1_test.dtypes μπορούμε να δούμε τον τύπο δεδομένων κάθε πεδίου (στήλης). Έτσι επιβεβαιώνουμε ότι η μετατροπή του Όγκου έγινε σωστά σε τύπο ακεραίων (int64).

```
In [43]: print(df1_test.dtypes)
```

```
Ημερομηνια    datetime64[ns]
Κλείσιμο             float64
ΜΕΤ.%                float64
Άνοιγμα              float64
Υψηλό                float64
Χαμηλό               float64
Όγκος                  Int64
Τζίρος               float64
dtype: object
```

Εμφανίζοντας τα περιεχόμενα του dataframe (τις 5 πρώτες και τις 5 τελευταίες εγγραφές) παρατηρούμε ότι οι αρχικές τιμές στον Όγκο (αρχές του έτους 1986) δεν περιέχουν τιμές. Όπως προαναφέραμε, με τη μέθοδο 'ffill' που εφαρμόσαμε παραπάνω συμπληρώθηκαν οι ελλιπείς τιμές με τις αμέσως προηγούμενες σωστές τιμές. Όταν όμως τα missing values είναι από την αρχή είναι

αντιληπτό ότι δεν υπάρχουν προηγούμενες σωστές τιμές και για αυτό η μέθοδος ffill δεν εφαρμόστηκε σε αυτές.

```
In [44]: df1_test
```

Out[44]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
9321	1986-01-02	2225.608	0.0000	2225.608	2225.608	2225.608	<NA>	0.00
9320	1986-01-03	2225.608	0.0000	2225.608	2225.608	2225.608	<NA>	0.00
9319	1986-01-07	2225.608	0.0000	2225.608	2225.608	2225.608	<NA>	0.00
9318	1986-01-08	2233.122	0.3376	2225.608	2233.122	2225.608	<NA>	0.00
9317	1986-01-09	2225.608	-0.3365	2233.122	2233.122	2225.608	<NA>	0.00
...	...	...	...	...	...	...	...	...
4	2023-07-17	6.000	-2.9126	6.216	6.218	5.988	987507	6005352.73
3	2023-07-18	6.202	3.3667	6.000	6.216	5.986	1510715	9293977.41
2	2023-07-19	6.220	0.2902	6.200	6.244	6.152	1675378	10417698.25
1	2023-07-20	6.310	1.4469	6.240	6.420	6.160	772694	4864979.74
0	2023-07-21	6.210	-1.5848	6.310	6.322	6.210	526717	3303527.97

9322 rows × 8 columns

Για να δούμε πόσες είναι οι εναπομείναντες ελλιπείς τιμές εκτελούμε την παρακάτω εντολή.

```
In [45]: df1_test.isna().sum()
```

```
Out[45]: Ημερομηνια       0
         Κλείσιμο         0
         ΜΕΤ.%            0
         Άνοιγμα          0
         Υψηλό            0
         Χαμηλό           0
         Όγκος          248
         Τζίρος           0
         dtype: int64
```

Διαπιστώνουμε ότι αυτές βρίσκονται στη στήλη Όγκος και είναι στο σύνολο 248 και επίσης είμαστε σίγουροι ότι αυτές βρίσκονται στην αρχή του dataframe. Για αυτό το λόγο έχουμε τη μέθοδο bfill με την οποία συμπληρώνονται οι ελλιπείς τιμές με την αμέσως επόμενη σωστή τιμή που υπάρχει στο συγκεκριμένο πεδίο. Εφαρμόζουμε τη μέθοδο συμπλήρωσης bfill.

```
In [46]: df1_test["Όγκος"].fillna( method='bfill', inplace = True)
```

Στη συνέχεια ελέγχουμε πόσα missing values έχουμε στο dataframe.

```
In [47]: df1_test.isna().sum()
```

```
Out[47]: Ημερομηνια     0
         Κλείσιμο       0
         ΜΕΤ.%          0
         Άνοιγμα        0
         Υψηλό          0
         Χαμηλό         0
         Όγκος          0
         Τζίρος         0
         dtype: int64
```

Διαπιστώνουμε ότι πλέον δεν υπάρχουν ελλιπείς τιμές. Εμφανίζοντας μάλιστα και τα περιεχόμενα του dataframe παρατηρούμε ότι οι τιμές του Όγκου στις αρχές του 1986 έχουν συμπληρωθεί με την τιμή 1 που στην συγκεκριμένη περίπτωση ήταν και η πρώτη σωστά συμπληρωμένη τιμή που υπήρχε στη συγκεκριμένη στήλη σε κάποια ημερομηνία.

```
In [48]: df1_test
```

Out[48]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
9321	1986-01-02	2225.608	0.0000	2225.608	2225.608	2225.608	1	0.00
9320	1986-01-03	2225.608	0.0000	2225.608	2225.608	2225.608	1	0.00
9319	1986-01-07	2225.608	0.0000	2225.608	2225.608	2225.608	1	0.00
9318	1986-01-08	2233.122	0.3376	2225.608	2233.122	2225.608	1	0.00
9317	1986-01-09	2225.608	-0.3365	2233.122	2233.122	2225.608	1	0.00
...	...	...	...	...	...	...	...	...
4	2023-07-17	6.000	-2.9126	6.216	6.218	5.988	987507	6005352.73
3	2023-07-18	6.202	3.3667	6.000	6.216	5.986	1510715	9293977.41
2	2023-07-19	6.220	0.2902	6.200	6.244	6.152	1675378	10417698.25
1	2023-07-20	6.310	1.4469	6.240	6.420	6.160	772694	4864979.74
0	2023-07-21	6.210	-1.5848	6.310	6.322	6.210	526717	3303527.97

9322 rows × 8 columns

Παραπάνω δείξαμε πως μπορούμε να χειριστούμε τις ελλιπείς τιμές (missing values) μέσα σε ένα dataframe συμπληρώνοντάς τες με τις μεθόδους ffill και bfill.

Στη δική μας ανάλυση και για λόγους μεγαλύτερης αξιοπιστίας και γνωρίζοντας ότι οι ελλιπείς τιμές υπάρχουν στους Όγκους των ημερήσιων συναλλαγών και μάλιστα ως επι το πλείστον σε ημερομηνίες που μας είναι σχετικά αδιάφορες (τη δεκαετία του ’80), επιλέγουμε να τις εξαιρέσουμε με τη μέθοδο dropna.

```
In [49]: df1.dropna(inplace=True)
```

Ελέγχουμε για missing values στο df1 (μετοχή της ΕΤΕ).

```
In [50]: df1.isna().sum()
```

```
Out[50]: Ημερομηνια    0
         Κλείσιμο      0
         ΜΕΤ.%         0
         Άνοιγμα       0
         Υψηλό         0
         Χαμηλό        0
         Όγκος         0
         Τζίρος        0
         dtype: int64
```

Ελέγχουμε για missing values στο df2 (μετοχή του ΟΤΕ).

```
In [51]: df2.isna().sum()
```

```
Out[51]: Ημερομηνια    0
         Κλείσιμο      0
         ΜΕΤ.%         0
         Άνοιγμα       0
         Υψηλό         0
         Χαμηλό        0
         Όγκος         0
         Τζίρος        0
         dtype: int64
```

Ελέγχουμε για missing values στο df3 (μετοχή της ΔΕΗ).

```
In [52]: df3.isna().sum()
```

```
Out[52]: Ημερομηνια    0
         Κλείσιμο      0
         MET.%         0
         Άνοιγμα       0
         Υψηλό         0
         Χαμηλό        0
         Όγκος         2
         Τζίρος        0
         dtype: int64
```

Παρατηρούμε ότι υπάρχουν 2 και κατόπιν τις εξαιρούμε.

```
In [53]: df3.dropna(inplace=True)
```

Επιβεβαιώνουμε την ενέργειά μας.

```
In [54]: df3.isna().sum()
```

```
Out[54]: Ημερομηνια    0
         Κλείσιμο      0
         MET.%         0
         Άνοιγμα       0
         Υψηλό         0
         Χαμηλό        0
         Όγκος         0
         Τζίρος        0
         dtype: int64
```

Ελέγχουμε για missing values στο df3 (μετοχή του ΜΥΤΙΛΗΝΑΙΟΥ).

```
In [55]: df4.isna().sum()
```

```
Out[55]: Ημερομηνια    0
         Κλείσιμο      0
         MET.%         0
         Άνοιγμα       0
         Υψηλό         0
         Χαμηλό        0
         Όγκος         0
         Τζίρος        0
         dtype: int64
```

Εμφανίζουμε τη στήλη του Όγκου για να επιβεβαιώσουμε τις αλλάγες.

```
In [56]: df1.loc[:,'Όγκος']
```

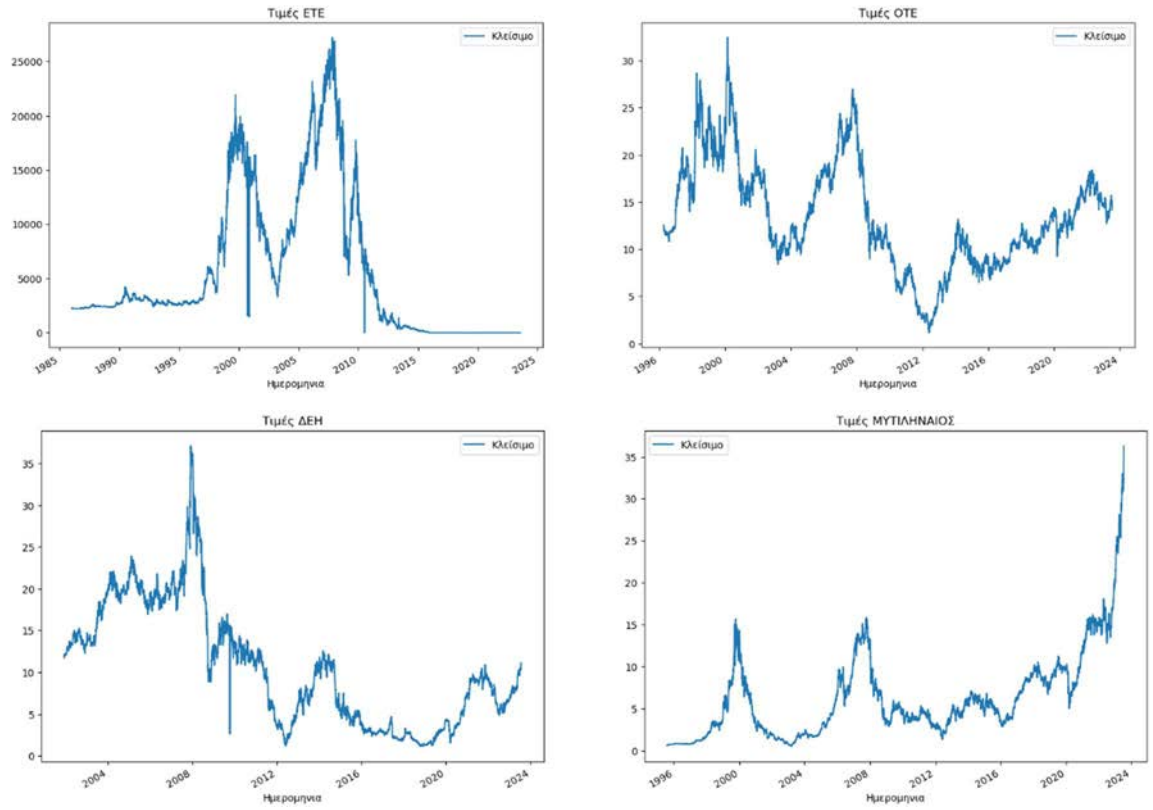
```
Out[56]: 9073          1.0
         9072          2.0
         9071          4.0
         9070          4.0
         9069          6.0
                      ...
         4        987507.0
         3       1510715.0
         2       1675378.0
         1        772694.0
         0        526717.0
         Name: Όγκος, Length: 9010, dtype: float64
```

Πλέον έχουμε όλα τα dataframes «καθαρά» και χωρίς ελλιπείς τιμές και επομένως μπορούμε να προχωρήσουμε στην ανάλυσή μας.

Δημιουργούμε ένα σύνολο από 4 διαγράμματα με τις τιμές κλεισιμάτων των μετοχών μας για να έχουμε μια πρώτη συγκεντρωτική εικόνα.

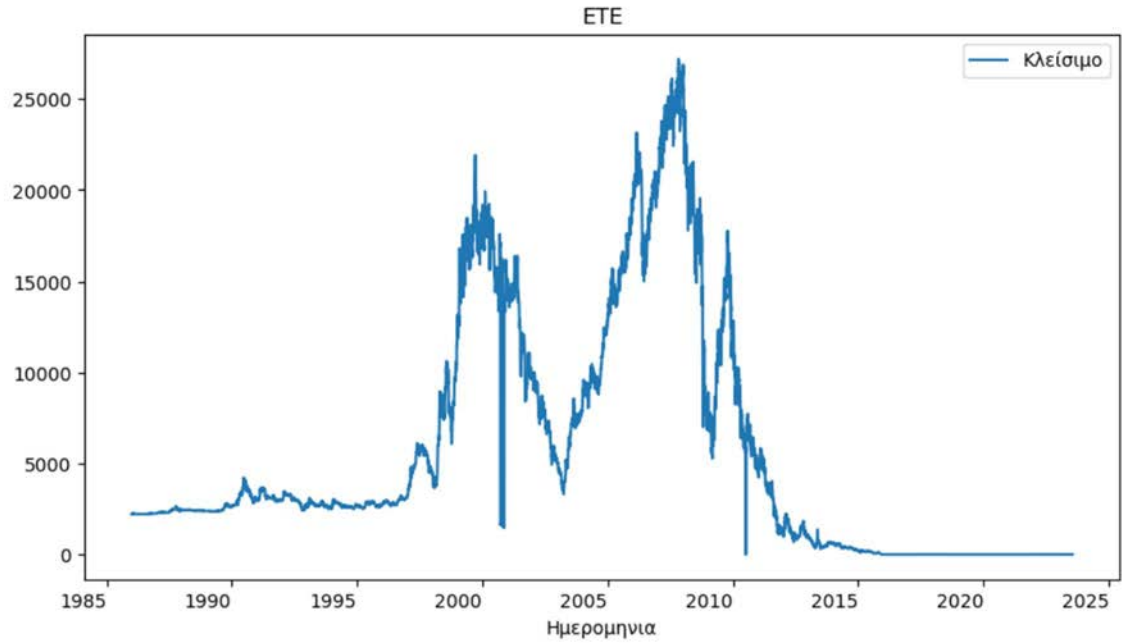
In [57]:
```
fig=plt.figure(figsize=(20,15))

ax1=fig.add_subplot(2,2,1)
df1.plot(kind='line',x='Ημερομηνια',y='Κλείσιμο',ax=ax1)
ax1.set_title("Τιμές ΕΤΕ")

ax2=fig.add_subplot(2,2,2)
df2.plot(kind='line',x='Ημερομηνια',y='Κλείσιμο',ax=ax2)
ax2.set_title("Τιμές ΟΤΕ")

ax3=fig.add_subplot(2,2,3)
df3.plot(kind='line',x='Ημερομηνια',y='Κλείσιμο',ax=ax3)
ax3.set_title("Τιμές ΔΕΗ")

ax4=fig.add_subplot(2,2,4)
df4.plot(kind='line',x='Ημερομηνια',y='Κλείσιμο',ax=ax4)
ax4.set_title("Τιμές ΜΥΤΙΛΗΝΑΙΟΣ")
```

Out[57]: Text(0.5, 1.0, 'Τιμές ΜΥΤΙΛΗΝΑΙΟΣ')



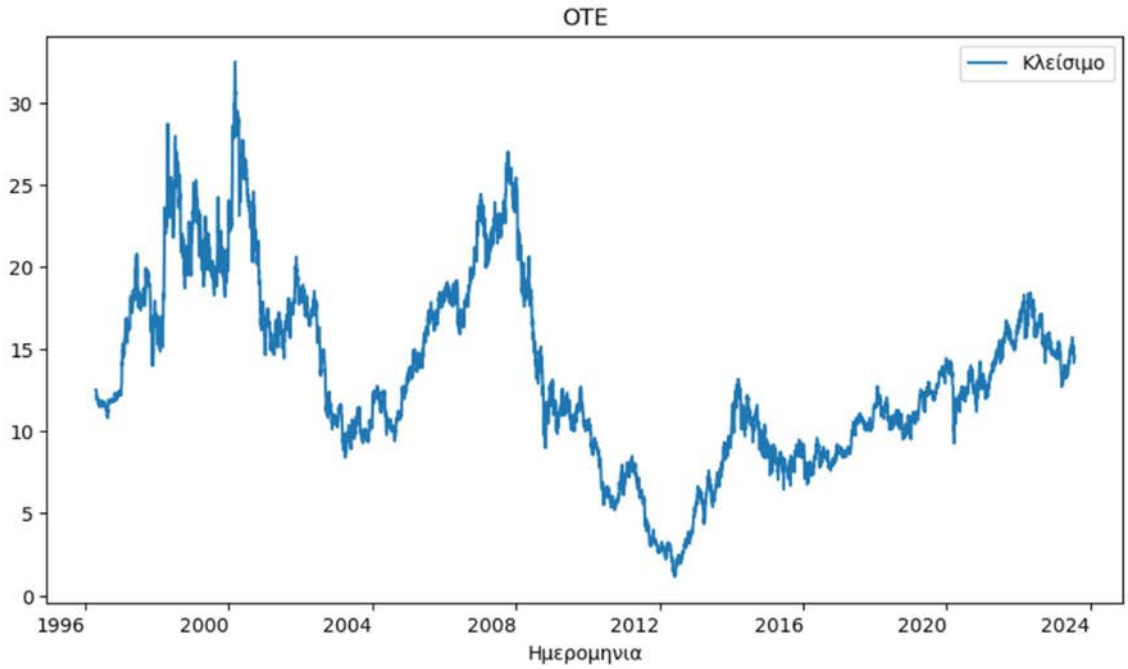
Δημιουργούμε ένα ξεχωριστό διάγραμμα για όλο το χρονικό διάστημα που έχουμε δεδομένα για τη μετοχή της ΕΤΕ.

```
In [58]: df1.plot(x='Ημερομηνια', y='Κλείσιμο', rot=0, figsize=(10, 6), title='ETE')
Out[58]: <AxesSubplot:title={'center':'ETE'}, xlabel='Ημερομηνια'>
```



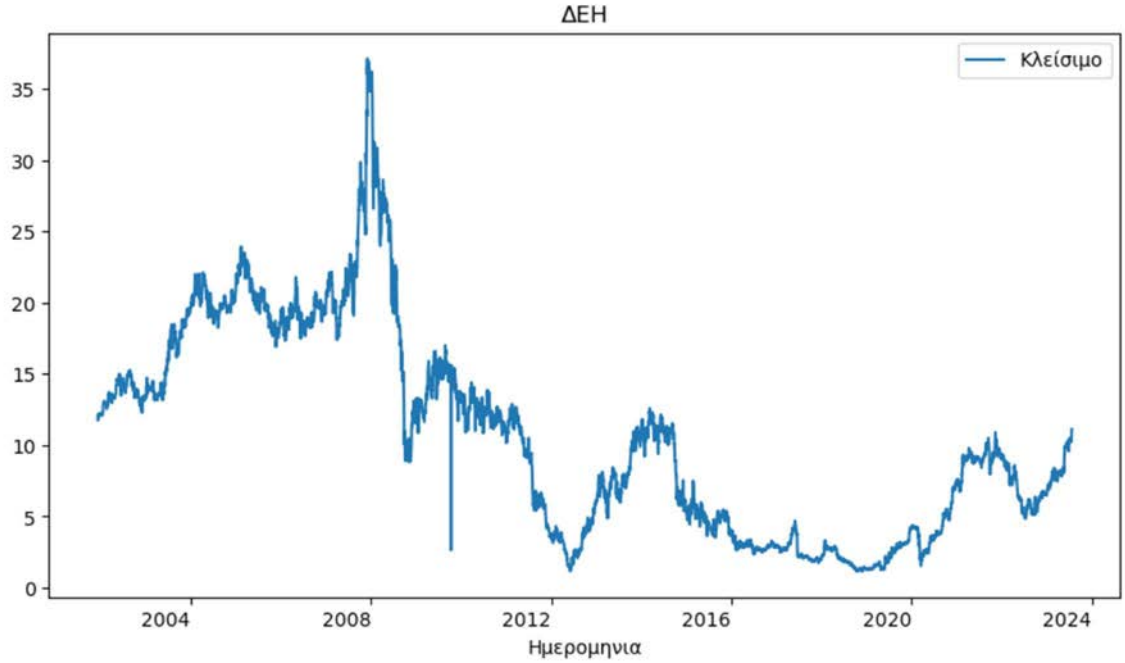
Δημιουργούμε ένα ξεχωριστό διάγραμμα για όλο το χρονικό διάστημα που έχουμε δεδομένα για τη μετοχή του ΟΤΕ.

```
In [59]: df2.plot(x='Ημερομηνια', y='Κλείσιμο', rot=0, figsize=(10, 6), title='OTE')
Out[59]: <AxesSubplot:title={'center':'OTE'}, xlabel='Ημερομηνια'>
```



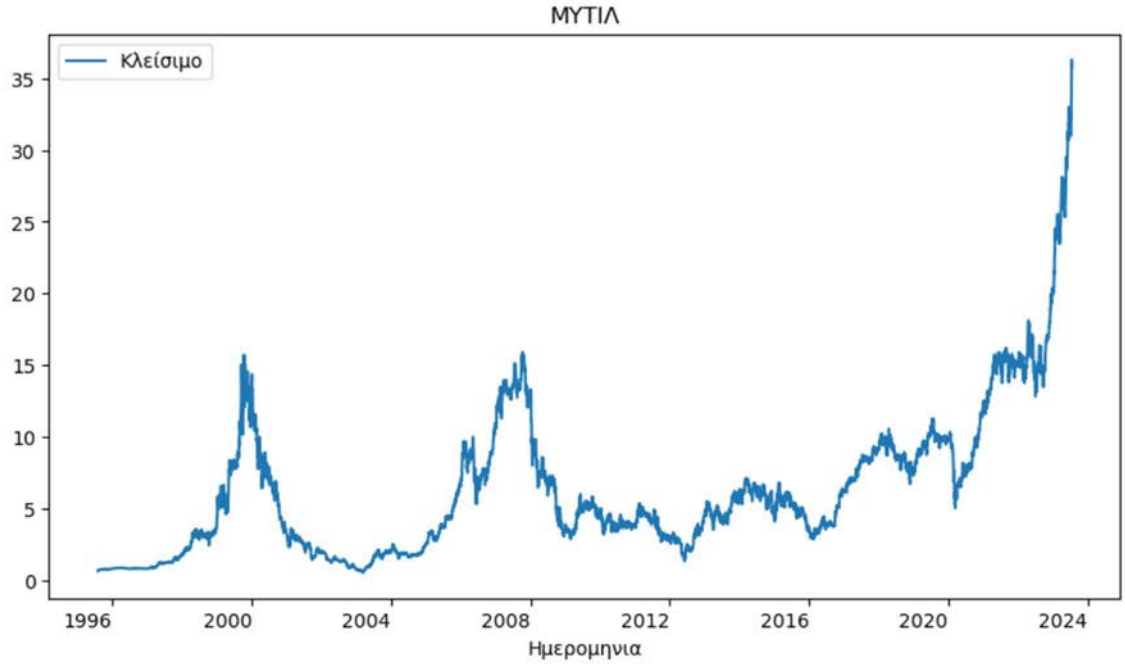
Δημιουργούμε ένα ξεχωριστό διάγραμμα για όλο το χρονικό διάστημα που έχουμε δεδομένα για τη μετοχή της ΔΕΗ.

```
In [60]: df3.plot(x='Ημερομηνια', y='Κλείσιμο', rot=0, figsize=(10, 6), title='ΔΕΗ')
Out[60]: <AxesSubplot:title={'center':'ΔΕΗ'}, xlabel='Ημερομηνια'>
```



Δημιουργούμε ένα ξεχωριστό διάγραμμα για όλο το χρονικό διάστημα που έχουμε δεδομένα για τη μετοχή του ΜΥΤΙΛΗΝΑΙΟΥ.

```
In [61]: df4.plot(x='Ημερομηνια', y='Κλείσιμο', rot=0, figsize=(10, 6), title='ΜΥΤΙΛ')
Out[61]: <AxesSubplot:title={'center':'ΜΥΤΙΛ'}, xlabel='Ημερομηνια'>
```



Ένα επιπλέον δεδομένο που μπορούμε να εντοπίσουμε είναι οι μέγιστες τιμές σε κάθε μια στήλη του dataframe.

```
In [62]: maxValues = df3.max(skipna=True)
         print(maxValues)
```

```
Ημερομηνια    2023-07-21 00:00:00
Κλείσιμο                     37.1
ΜΕΤ.%                     491.2879
Άνοιγμα                      37.1
Υψηλό                        37.4
Χαμηλό                       36.7
Όγκος                    25830963.0
Τζίρος                 2874429788.0
dtype: object
```

Εναλλακτικά μπορούμε να δούμε τη μέγιστη τιμή κλεισίματος που έχει σημειωθεί στη μετοχή της ΕΤΕ.

```
In [63]: df1['Κλείσιμο'].max()
```

```
Out[63]: 27203.489
```

Συμπληρωματικά και για περισσότερη ενημέρωση μπορούμε να δούμε πότε σημειώθηκε αυτή η μέγιστη τιμή κλεισίματος της μετοχής της ΕΤΕ.

```
In [64]: df1[df1['Κλείσιμο'] == df1['Κλείσιμο'].max()]
```

Out[64]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
3884	2007-10-31	27203.489	0.5838	27173.418	27203.489	26887.694	1918.0	56342828.74

Κατόπιν πότε σημειώθηκε η ελάχιστη τιμή κλεισίματος της μετοχής της ΕΤΕ.

```
In [65]: df1[df1['Κλείσιμο'] == df1['Κλείσιμο'].min()]
```

Out[65]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
828	2020-03-23	0.85	-14.6929	0.9302	0.9696	0.8404	5123284.0	4589399.19

Σε αυτό το σημείο σκόπιμο είναι να αναφερθούμε στις τιμές της ΕΤΕ σύμφωνα με τα παραπάνω. Παρατηρούμε ότι η μέγιστη τιμή ήταν 27203.489 και σημειώθηκε στις 31-10-2007 και η ελάχιστη τιμή ήταν 0.85 και σημειώθηκε στις 23-3-2020. Οι τιμές αυτές με μια πρώτη ματιά μπορεί να μας ξαφνιάζουν και να νομίζουμε ότι κάτι λάθος μπορεί να έχουν τα δεδομένα μας αλλά κάτι τέτοιο δεν ισχύει. Όντως με σημερινές τιμές οι μετοχή της ΕΤΕ από τα περίπου 27000 € κατρακύλησε το 2020 (και εν μέσω της πανδημίας COVID) κάτω από το 1€. Για να εξηγηθεί μια τέτοια πτώση αρκεί να συνυπολογίσουμε την οικονομική κρίση της δεκαετίας του 2010-2020 που κυρίως έπληξε τον τραπεζικό κλάδο και ο οποίος χρειάστηκε σχεδόν 3 ανακεφαλαιοποίησεις για να σταθεί όρθιος. Κοινώς με απλά λόγια μέσα στην οικονομική κρίση της Ελλάδας οι τιμές των τραπεζικών μετοχών σχεδόν «μηδένισαν».

Στη συνέχεια εντοπίζουμε πότε σημειώθηκε η μέγιστη και η ελάχιστη τιμή κλεισίματος της μετοχής του ΟΤΕ.

```
In [66]: df2[df2['Κλείσιμο'] == df2['Κλείσιμο'].max()]
```

Out[66]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
5797	2000-03-06	32.47	4,7757%	31.4	32.71	31.4	1691882	0.0

```
In [67]: df2[df2['Κλείσιμο'] == df2['Κλείσιμο'].min()]
```

Out[67]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
2741	2012-06-05	1.13	-5,0420%	1.2	1.23	1.09	4028127	4592583.24

Βρίσκουμε πότε σημειώθηκε η μέγιστη και η ελάχιστη τιμή κλεισίματος της μετοχής της ΔΕΗ.

```
In [68]: df3[df3['Κλείσιμο'] == df3['Κλείσιμο'].max()]
```

Out[68]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
3859	2007-12-05	37.1	11.8143	33.5	37.1	33.5	3146796.0	1.120319e+08

```
In [69]: df3[df3['Κλείσιμο'] == df3['Κλείσιμο'].min()]
```

Out[69]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
1151	2018-11-27	1.152	-4.0	1.218	1.23	1.152	246238.0	289564.46

Τέλος εντοπίζουμε πότε σημειώθηκε η μέγιστη και η ελάχιστη τιμή κλεισίματος της μετοχής του ΜΥΤΙΛΗΝΑΙΟΥ.

```
In [70]: df4[df4['Κλείσιμο'] == df4['Κλείσιμο'].max()]
```

Out[70]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
1	2023-07-20	36.3	1.7377	36.0	36.3	35.82	216367	7820437.02

```
In [71]: df4[df4['Κλείσιμο'] == df4['Κλείσιμο'].min()]
```

Out[71]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
5046	2003-03-12	0.519	-3.7106	0.539	0.539	0.509	39353	116974.65

Αφού πήραμε μια πρώτη εικόνα για τις μετοχές που έχουμε επιλέξει να μελετήσουμε, επιλέγουμε για την ανάλυσή μας το χρονικό διάστημα από τις αρχές του 2017 μέχρι και σήμερα (21-7-2023). Για αυτό το λόγο μετασχηματίζουμε τα datframes (df1, df2, df3, df4) να περιέχουν τα δεδομένα για το παραπάνω χρονικό διάστημα. Για το λόγο αυτό εφαρμόζουμε τις παρακάτω εντολές.

```
In [72]: df1 = df1.loc[df1['Ημερομηνια'] >= '2017-01-01']
         df2 = df2.loc[df2['Ημερομηνια'] >= '2017-01-01']
         df3 = df3.loc[df3['Ημερομηνια'] >= '2017-01-01']
         df4 = df4.loc[df4['Ημερομηνια'] >= '2017-01-01']
```

Στη συνέχεια, επειδή τα dataframes περιέχουν τμήμα από τα αρχικά, κάνουμε επαναφορά των δεικτών (ευρετήριο - index) με τη μέθοδο .reset_index εφαρμόζοντάς τα εντός των dataframes και στην ίδια ακριβώς στήλη index (drop=True).

```
In [73]: df1.reset_index(drop=True)
         df2.reset_index(drop=True)
         df3.reset_index(drop=True)
         df4.reset_index(drop=True)
```

Out[73]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
0	2017-01-02	6.135	-0.2439	6.095	6.165	6.095	14037	86234.97
1	2017-01-03	6.315	2.9340	6.135	6.385	6.135	87554	550382.02
2	2017-01-04	6.315	0.0000	6.315	6.365	6.315	84689	536761.19
3	2017-01-05	6.295	-0.3167	6.295	6.355	6.275	57620	363360.18
4	2017-01-09	6.295	0.0000	6.255	6.315	6.215	49885	312608.30
...	...	...	...	...	...	...	...	...
1625	2023-07-17	33.300	1.4625	32.980	33.300	32.700	171926	5683682.32
1626	2023-07-18	34.540	3.7237	33.340	35.000	33.340	412590	14278135.60
1627	2023-07-19	35.680	3.3005	35.100	36.120	34.880	293517	10527500.36
1628	2023-07-20	36.300	1.7377	36.000	36.300	35.820	216367	7820437.02
1629	2023-07-21	35.820	-1.3223	36.300	36.400	35.820	145221	5235476.58

1630 rows × 8 columns

Σύμφωνα με τα παραπάνω παρατηρούμε ενδεικτικά για το df4 ότι έχει γίνει επαναφορά και επανυπολογισμός του ευρετηρίου (index).

Στη συνέχεια δημιουργούμε ένα νέο dataframe, df11 πιστό αντίγραφο του df1 το οποίο θα μας χρειαστεί προς το τέλος της ανάλυσής μας. (βλέπε 138)

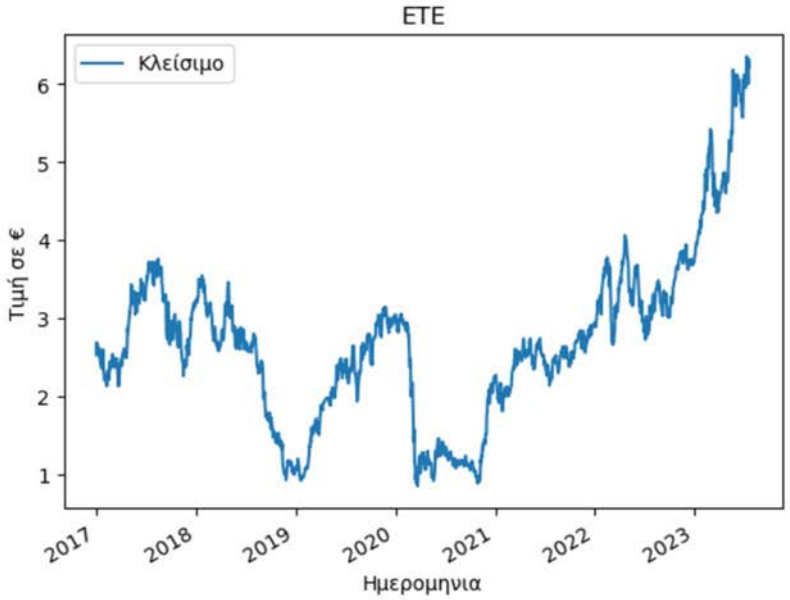
```
In [74]: df11=df1 # Το df11 θα μας χρειαστεί σε παρακάτω ανάλυση
```

Επειδή τα dataframes είναι ταξινομημένα με βάση την Ημερομηνία μπορούμε να δούμε την τιμή κλεισίματος της μετοχής (π.χ. του ΜΥΤΙΛΗΝΑΙΟΥ, df4) με τη χρήση της παρακάτω εντολής.

```
In [75]: df4.iloc[0][1]
Out[75]: 6.135
```

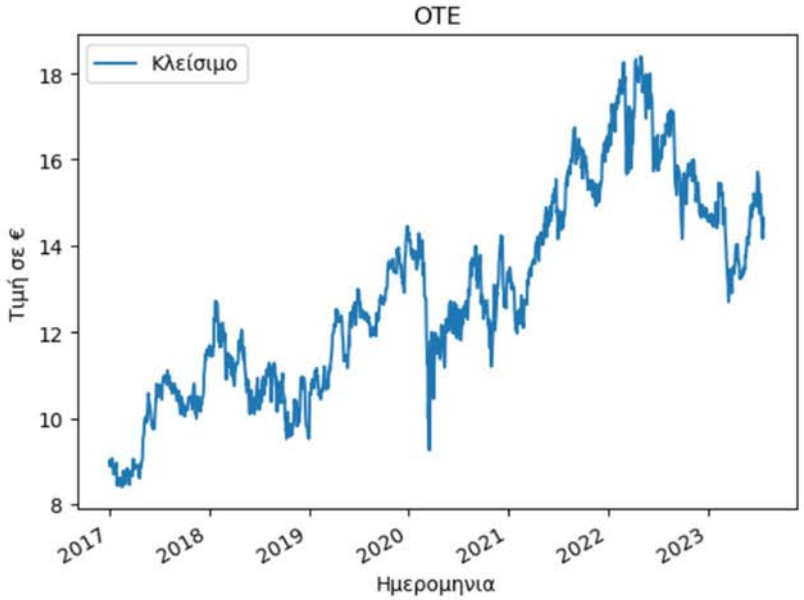
Δημιουργούμε το γράφημα των τιμών κλεισίματος της μετοχής της ΕΤΕ για το χρονικό διάστημα μετά από το 2017 και μετά.

```
In [76]: df1.plot(x='Ημερομηνια', y='Κλείσιμο', title='ETE',ylabel='Τιμή σε €')
Out[76]: <AxesSubplot:title={'center':'ETE'}, xlabel='Ημερομηνια', ylabel='Τιμή σε €'>
```



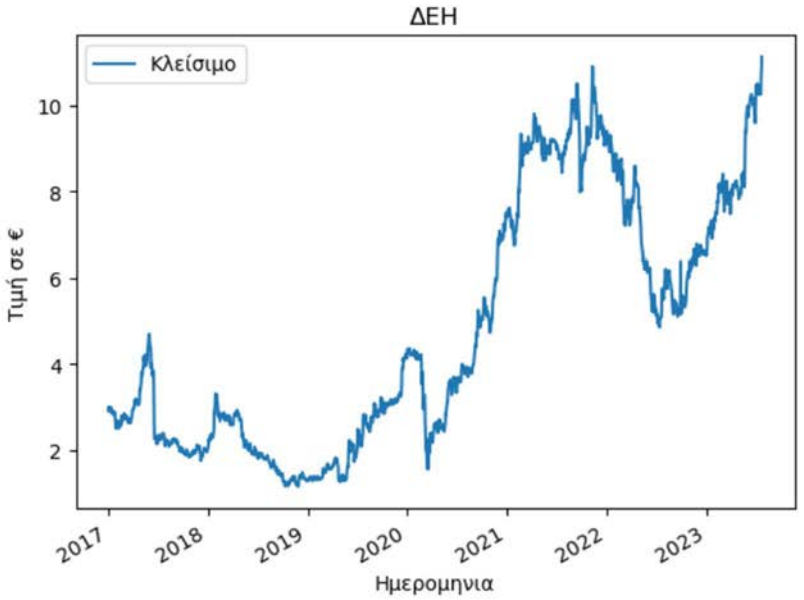
Δημιουργούμε το γράφημα των τιμών κλεισίματος της μετοχής του ΟΤΕ για το χρονικό διάστημα μετά από το 2017 και μετά.

```
In [77]: df2.plot(x='Ημερομηνια', y='Κλείσιμο', title='OTE',ylabel='Τιμή σε €')
Out[77]: <AxesSubplot:title={'center':'OTE'}, xlabel='Ημερομηνια', ylabel='Τιμή σε €'>
```



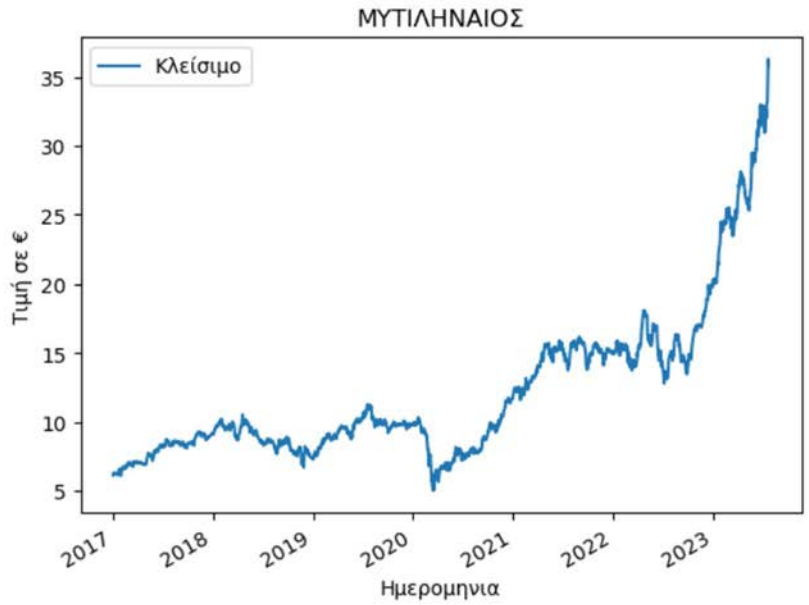
Δημιουργούμε το γράφημα των τιμών κλεισίματος της μετοχής της ΔΕΗ για το χρονικό διάστημα μετά από το 2017 και μετά.

```
In [78]: df3.plot(x='Ημερομηνια', y='Κλείσιμο', title='ΔΕΗ',ylabel='Τιμή σε €')
Out[78]: <AxesSubplot:title={'center':'ΔΕΗ'}, xlabel='Ημερομηνια', ylabel='Τιμή σε €'>
```



Δημιουργούμε το γράφημα των τιμών κλεισίματος της μετοχής του ΜΥΤΙΛΗΝΑΙΟΥ για το χρονικό διάστημα μετά από το 2017 και μετά.

```
In [79]: df4.plot(x='Ημερομηνια', y='Κλείσιμο', title='ΜΥΤΙΛΗΝΑΙΟΣ',ylabel='Τιμή σε €')
Out[79]: <AxesSubplot:title={'center':'ΜΥΤΙΛΗΝΑΙΟΣ'}, xlabel='Ημερομηνια', ylabel='Τιμή σε €'>
```



Για να μπορέσουμε να δούμε ένα σύνολο περιγραφικών στατιστικών στοιχείων, χρησιμοποιούμε τη μέθοδο .describe (με τη συνάρτηση round(2) όλα τα αποτελέσματα στρογγυλοποιούνται σε 2 δεκαδικά ψηφία).

Οι περιγραφικές στατιστικές περιλαμβάνουν εκείνες που συνοψίζουν την κεντρική τάση, τη διασπορά και το σχήμα της κατανομής ενός συνόλου δεδομένων, εξαιρουμένων των τιμών NaN. Αναλύει τόσο αριθμητικές σειρές όσο και σειρές αντικειμένων, καθώς και σύνολα στηλών dataframe με μικτούς τύπους δεδομένων.

```
In [80]: df1.describe().round(2)
Out[80]:
```

	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
count	1630.00	1630.00	1630.00	1630.00	1630.00	1627.00	1.630000e+03
mean	2.70	0.12	2.70	2.75	2.65	3033997.31	6.746798e+06
std	1.07	3.46	1.06	1.08	1.05	3575894.82	8.288003e+06
min	0.85	-20.98	0.88	0.90	0.82	136238.00	0.000000e+00
25%	2.10	-1.62	2.10	2.14	2.05	1443869.50	2.856465e+06
50%	2.65	0.13	2.65	2.70	2.60	2272140.00	5.094255e+06
75%	3.20	1.82	3.20	3.26	3.15	3519894.00	8.633641e+06
max	6.35	29.24	6.36	6.42	6.27	60095372.00	1.759474e+08

Όπως παρατηρούμε από τον παραπάνω πίνακα τα αποτελέσματα της .describe() μας δείχνουν: το πλήθος των τιμών (count), τη μέση τιμή (mean), την τυπική απόκλιση (std - standard deviation), την ελάχιστη (min) και τη μέγιστη τιμή (max). Επίσης παρατηρούμε ότι η συνάρτηση describe() υπολογίζει το 25^{ο} , 50^{ο} και 75^{ο} εκατοστημόριο για κάθε μεταβλητή από προεπιλογή.

Εφαρμόζουμε τη μέθοδο .describe() για το df2.

```
In [81]: df2.describe().round(2)
```

Out[81]:

	Κλείσιμο	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
count	1630.00	1630.00	1630.00	1630.00	1630.00	1.630000e+03
mean	12.92	12.91	13.07	12.79	593137.28	6.823492e+06
std	2.36	2.37	2.39	2.35	748327.97	9.080033e+06
min	8.40	8.40	8.50	8.33	49299.00	0.000000e+00
25%	10.91	10.88	11.00	10.78	339320.25	3.585823e+06
50%	12.64	12.63	12.76	12.50	483417.50	5.732713e+06
75%	14.76	14.79	14.91	14.64	686481.25	8.344967e+06
max	18.40	18.45	18.50	18.28	26240532.00	3.019256e+08

Εφαρμόζουμε τη μέθοδο .describe() για το df3.

```
In [82]: df3.describe().round(2)
```

Out[82]:

	Κλείσιμο	MET.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
count	1630.00	1630.00	1630.00	1630.00	1630.00	1628.00	1.630000e+03
mean	4.79	0.14	4.79	4.87	4.71	564682.44	2.606580e+06
std	2.89	3.39	2.89	2.92	2.86	739326.94	6.646223e+06
min	1.15	-40.99	1.16	1.19	1.11	35662.00	0.000000e+00
25%	2.22	-1.35	2.21	2.26	2.17	246143.00	4.472247e+05
50%	3.83	0.00	3.81	3.87	3.74	408585.00	1.472738e+06
75%	7.50	1.55	7.50	7.62	7.38	670384.00	3.234981e+06
max	11.12	28.41	10.85	11.12	10.76	22743681.00	2.343193e+08

Εφαρμόζουμε τη μέθοδο .describe() για το df4.

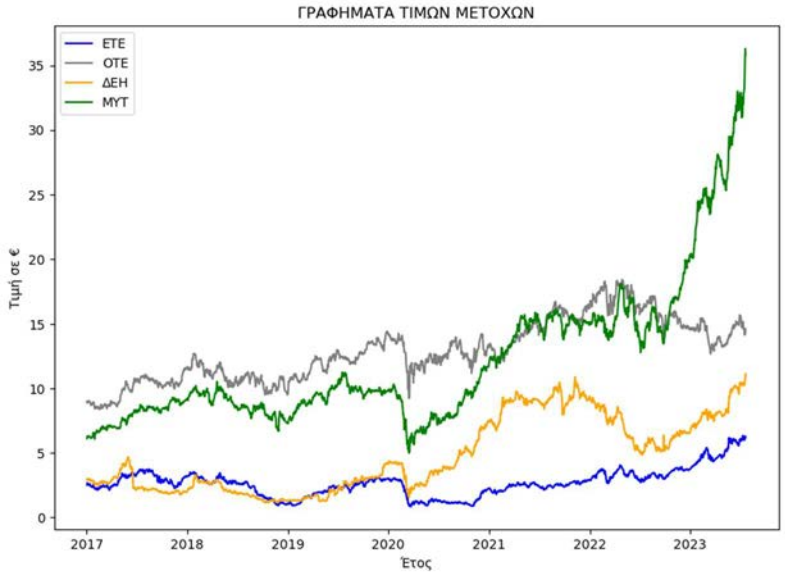
```
In [83]: df4.describe().round(2)
```

Out[83]:

	Κλείσιμο	MET.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος
count	1630.00	1630.00	1630.00	1630.00	1630.00	1630.00	1.630000e+03
mean	12.19	0.13	12.17	12.32	12.03	268000.12	3.097604e+06
std	5.59	1.99	5.57	5.64	5.51	304913.50	5.151414e+06
min	5.01	-17.15	5.00	5.04	4.90	14037.00	0.000000e+00
25%	8.45	-0.91	8.45	8.55	8.37	148348.00	1.219516e+06
50%	9.82	0.12	9.84	9.93	9.75	215537.00	2.185745e+06
75%	15.04	1.19	15.02	15.19	14.90	316991.25	3.735721e+06
max	36.30	11.76	36.30	36.40	35.82	8207462.00	1.363938e+08

Στη συνέχεια της ανάλυσής μας, δημιουργούμε ένα γράφημα για κάθε μία από της μετοχές που μελετάμε. Τα γραφήματα αυτά θα είναι αποτυπωμένα στο ίδιο πλαίσιο και άξονες για να μπορέσουμε να έχουμε μια συγκριτική εικόνα μεταξύ των μετοχών αυτών.

```
In [84]: plt.figure(figsize=(10,7))

         plt.plot(df1['Ημερομηνια'],df1['Κλείσιμο'],color='blue',label='ETE')
         plt.plot(df2['Ημερομηνια'],df2['Κλείσιμο'],color='grey',label='OTE')
         plt.plot(df3['Ημερομηνια'],df3['Κλείσιμο'],color='orange',label='ΔΕΗ')
         plt.plot(df4['Ημερομηνια'],df4['Κλείσιμο'],color='green',label='MYT')

         plt.title("ΓΡΑΦΗΜΑΤΑ ΤΙΜΩΝ ΜΕΤΟΧΩΝ")
         plt.xlabel("Έτος")
         plt.ylabel("Τιμή σε €")
         plt.legend(title="")
         plt.show()
```

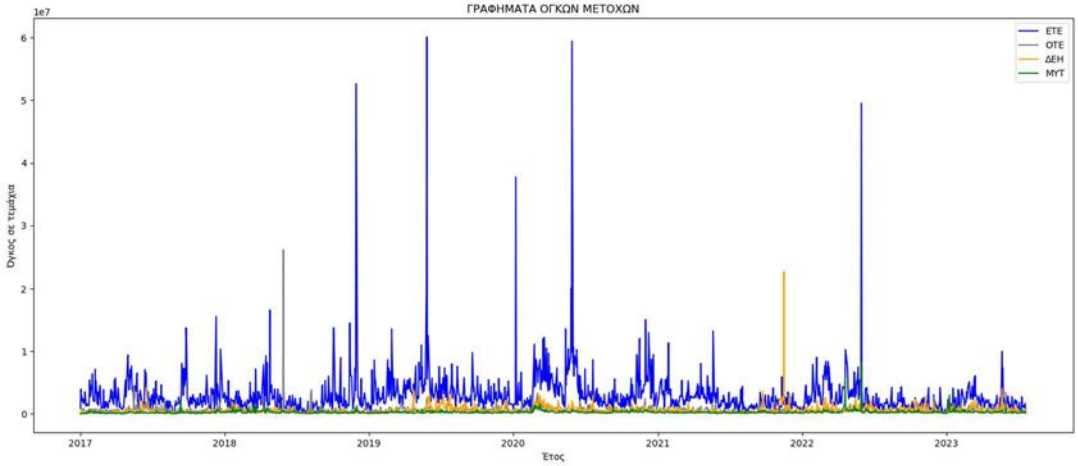


Σύμφωνα με το παραπάνω γράφημα παρατηρούμε τα εξής:

- Όλες οι μετοχές κατά την έναρξη της πανδημίας την Άνοιξη του 2020 κινήθηκαν έντονα καθοδικά.
- Πιο γρήγορα συνήλθαν από την πανδημική κρίση οι μετοχές της ενέργειας (ΔΕΗ, ΜΥΤΙΛ)
- Η μετοχή της ΔΕΗ ταλαιπωρήθηκε με μια πτώση από το 2ο εξάμηνο του 2021 έως και τα μέσα του 2022 που άρχισε εκ νέου να ανακάμπτει.
- Η μετοχή του ΟΤΕ παρόλο που ανέκαμψε μετά από ένα χρόνο από την έναρξη της πανδημίας συνεχίζει και βρίσκεται περίπου στα ίδια επίπεδα όπως και πριν από αυτή.
- Η ΕΤΕ ανέκαμψε με αργό ρυθμό μετά την πανδημία, αλλά μετά από τα μέσα του 2022 έχει σημαντική βελτίωση στην τιμή της.
- Η μετοχή του ΜΥΤΙΛΗΝΑΙΟΥ πραγματικά εκπλήσσει ευχάριστα για την ορμή που επιδεικνύει όπου από τα χαμηλά περίπου των 5 ευρώ στην πανδημία έχει βρεθεί τελευταία πάνω από τα 35 ευρώ, 7πλασιάζοντας σχεδόν την τιμή της.

Στη συνέχεια δημιουργούμε ένα γράφημα με τους ημερήσιους Όγκους συναλλαγών των μετοχών.

```
In [85]: plt.figure(figsize=(20,8))

         plt.plot(df1['Ημερομηνια'],df1['Όγκος'],color='blue',label='ΕΤΕ')
         plt.plot(df2['Ημερομηνια'],df2['Όγκος'],color='grey',label='ΟΤΕ')
         plt.plot(df3['Ημερομηνια'],df3['Όγκος'],color='orange',label='ΔΕΗ')
         plt.plot(df4['Ημερομηνια'],df4['Όγκος'],color='green',label='ΜΥΤ')

         plt.title("ΓΡΑΦΗΜΑΤΑ ΟΓΚΩΝ ΜΕΤΟΧΩΝ")
         plt.xlabel("Έτος")
         plt.ylabel("Όγκος σε τεμάχια")
         plt.legend(title="")
         plt.show()
```



Σύμφωνα με το παραπάνω γράφημα δεν μας εκπλήσσει το γεγονός της μεγαλύτερης συναλλακτικής δραστηριότητας στη μετοχή της ΕΤΕ λόγων και της χαμηλότερης τιμής μεταξύ των υπολοίπων 3 μετοχών.

Κατόπιν δημιουργούμε μια νέα στήλη ‘Day’ η οποία θα περιέχει ως τιμή τον αριθμό ημέρας μιας ημερομηνίας π.χ. για την ημερομηνία 21-7-2023 θα έχει την τιμή 21.

```
In [86]: from datetime import date
         df1['Day']=df1['Ημερομηνια'].dt.day
         df2['Day']=df2['Ημερομηνια'].dt.day
         df3['Day']=df3['Ημερομηνια'].dt.day
         df4['Day']=df4['Ημερομηνια'].dt.day
```

Τα αποτελέσματα μπορούμε να τα δούμε παρακάτω, με τη δημιουργία αυτής της νέας στήλης Day.

```
In [87]: df1
```

Out[87]:

	Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος	Day
1629	2017-01-02	2.530	2.0161	2.500	2.550	2.480	858927.0	2168559.82	2
1628	2017-01-03	2.690	6.3241	2.530	2.730	2.520	3940597.0	10367046.15	3
1627	2017-01-04	2.620	-2.6022	2.680	2.700	2.610	2073396.0	5465725.52	4
1626	2017-01-05	2.550	-2.6718	2.590	2.600	2.530	2859241.0	7311028.08	5
1625	2017-01-09	2.530	-0.7843	2.550	2.580	2.520	1670785.0	4247937.85	9
...	...	...	...	...	...	...	...	...	...
4	2023-07-17	6.000	-2.9126	6.216	6.218	5.988	987507.0	6005352.73	17
3	2023-07-18	6.202	3.3667	6.000	6.216	5.986	1510715.0	9293977.41	18
2	2023-07-19	6.220	0.2902	6.200	6.244	6.152	1675378.0	10417698.25	19
1	2023-07-20	6.310	1.4469	6.240	6.420	6.160	772694.0	4864979.74	20
0	2023-07-21	6.210	-1.5848	6.310	6.322	6.210	526717.0	3303527.97	21

1630 rows × 9 columns

Δημιουργούμε ένα νέο dataframe από το df1 και ομαδοποιούμε με βάση τη στήλη Day και μετράμε το πλήθος αυτών των ημερών πόσες φορές εμφανίζονται μέσα στο dataframe.

Θέλουμε με αυτό τον τρόπο να δούμε ποια ημέρα του μήνα μπορούμε να χρησιμοποιήσουμε ώστε να την επιλέξουμε για τις αγορές μας.

```
In [88]: #Υπολογισμός ποια ημέρα του μήνα έμφανίζεται τις περισσότερς φορές
         df69 = df1.groupby(['Day'])['Day'].count()
         df69
```

```
Out[88]: Day
         1     47
         2     52
         3     56
         4     54
         5     53
         6     52
         7     56
         8     55
         9     55
         10    55
         11    54
         12    56
         13    56
         14    55
         15    50
         16    56
         17    53
         18    54
         19    54
         20    57
         21    56
         22    53
         23    56
         24    52
         25    46
         26    50
         27    55
         28    51
         29    49
         30    49
         31    33
         Name: Day, dtype: int64
```

Σύμφωνα με τα παραπάνω παρατηρούμε ότι η 20η ημέρα εμφανίζεται τις περισσότερες φορές (57 φορές) μέσα στο χρονικό διάστημα που μελετάμε (από τις αρχές του 2017 και μετά).

Ενδιαφέρον όμως θα είχε να δούμε πόσες αλλά και ποιες είναι οι τελευταίες εργάσιμες ημέρες κάθε μήνα. Για αυτό το λόγο δημιουργήσαμε μια λίστα ddays η οποία κρατάει τα δεδομένα της στήλης Day και μια λίστα ddates η οποία κρατάει τα στοιχεία της στήλης Ημερομηνία. Διασχίζοντας τη λίστα ddays όταν υπάρχει αριθμός μικρότερος από τον προηγούμενο τότε αποθηκεύουμε στη λίστα ddates_new την προηγούμενη ακριβώς ημερομηνία που είναι και η τελευταία ημερομηνία του μήνα.

```
In [89]: ddays = df1['Day'].values.tolist()
         ddates=df1['Ημερομηνια'].dt.date.tolist()
         ddates_new=[]
         for i in range(1,len(ddays)):
             if ddays[i]<ddays[i-1]:
                 ddates_new.append(ddates[i-1])
```

Εμφανίζουμε τα περιεχόμενα της λίστας ddates_new που περιέχει τις ημερομηνίες με την τελευταία εργάσιμη ημέρα του μήνα καθώς και το πλήθος τους.

```
In [90]: #Εμφάνιση της λίστας με ημερ. τελευταίας Παρασκευής κάθε μήνα
         # στο διάστημα 2017 έως Ιούνιο 2023
         print ('Συνολικό πλήθος ημερομηνιών:',len(datesFriday))
         for x in datesFriday:
             print(x)
```

```
Συνολικό πλήθος ημερομηνιών: 78
2017-01-31
2017-02-28
2017-03-31
2017-04-28
2017-05-31
2017-06-30
2017-07-31
2017-08-31
2017-09-29
2017-10-31
2017-11-30
2017-12-29
2018-01-31
2018-02-28
2018-03-29
2018-04-30
2018-05-31
2018-06-29
2018-07-31
2018-08-31
2018-09-28
2018-10-31
2018-11-30
2018-12-31
2019-01-31
2019-02-28
2019-03-29
2019-04-30
2019-05-31
2019-06-28
2019-07-31
2019-08-30
2019-09-30
2019-10-31
2019-11-29
2019-12-31
2020-01-30
2020-02-28
2020-03-31
2020-04-30
2020-05-29
2020-06-30
2020-07-31
2020-08-31
2020-09-30
2020-10-30
2020-11-30
2020-12-31
2021-01-29
2021-02-26
2021-03-31
2021-04-29
2021-05-31
2021-06-30
2021-07-30
2021-08-31
2021-09-30
2021-10-29
2021-11-30
2021-12-31
2022-01-31
2022-02-28
2022-03-31
2022-04-29
2022-05-31
2022-06-30
2022-07-29
2022-08-31
2022-09-30
2022-10-31
2022-11-30
2022-12-30
2023-01-31
2023-02-28
2023-03-31
2023-04-28
2023-05-31
2023-06-30
```

Παρατηρούμε ότι το πλήθος των ημερομηνιών που αντιπροσωπεύουν την τελευταία εργάσιμη κάθε μήνα από την αρχή του 2017 έως και το τέλος του 1ου εξαμήνου του 2023 είναι 78 και είναι προτιμότερη τακτική να ακολουθήσουμε για τις αγορές μας.

Απόδοση επένδυσης στην ΕΤΕ

Όπως και από την αρχή της ενότητας έχουμε αναφέρει, η στρατηγική που ακολουθήσαμε για να κάνουμε την ανάλυσή μας ήταν η εξής: Κάθε τελευταία εργάσιμη ημέρα κάθε μήνα (από την αρχή του 2017 έως και το τέλος του Ιουνίου 2023) θα αγοράζουμε ένα τεμάχιο από καθεμία από τις 4 μετοχές που μελετάμε. Για παράδειγμα ξεκινάμε από 31-1-2017 με αγορά ενός τεμαχίου της μετοχής της ΕΤΕ, ενός τεμαχίου της μετοχής του ΟΤΕ, ενός τεμαχίου της μετοχής της ΔΕΗ και ενός τεμαχίου της μετοχής του ΜΥΤΙΛΗΝΑΙΟΥ. Ακολούθως την τελευταία εργάσιμη ημέρα του Φεβρουαρίου του 2027 (28-2-2017) κάνουμε αγορά από ένα επιπλέον τεμάχιο από καθεμία μετοχή κ.ο.κ

Για να μπορούμε να υλοποιήσουμε αυτή τη στρατηγική επένδυσης για κάθε μια μετοχή εκτελούμε τα ακόλουθα.

Συγκεκριμένα για αρχή για την μετοχή της ΕΤΕ και ομοίως για τις υπόλοιπες:

Δημιουργούμε μια λίστα buy_date1 η οποία θα περιέχει τις ημερομηνίες που γίνονται οι αγορές και μια λίστα lt1 με τις αντίστοιχες τιμές κλεισίματος της μετοχής εκείνη την ημέρα αγοράς. Επιπλέον δημιουργούμε μια λίστα sum_stock1 η οποία περιέχει το σύνολο των τεμαχίων της ΕΤΕ κάθε τελευταία εργάσιμη ημέρα κάθε μήνα καθώς και μια λίστα money_invest1 με το συνολικό ποσό που έχουμε επενδύσει για κάθε αντίστοιχη ημερομηνία.

Η μεταβλητή sum1 αθροίζει σωρευτικά τις τιμές αγοράς (και επομένως το ποσό επένδυσης), και μεταβλητή s1 υπολογίζει σωρευτικά τα τεμάχια που έχουμε στην κατοχή μας κάθε φορά.

Στο τέλος εμφανίζουμε τα αποτελέσματα.

```
In [91]: lt1=[] #  λίστα με τιμές αγοράς κάθε τελευταία εργάσιμη ημέρα κάθε μήνα
         buy_date1=[] #  λίστα με ημερομηνίες αγορών
         sum_stock1=[] #  λίστα με το σύνολο μετοχών κάθε τέλευταία εργάσιμη ημέρα ανά μήνα
         money_invest1=[] #  λίστα με συνολικό ποσό επένδυσης κάθε τέλος του μήνα
         sum1=0 #συνολικό ποσό επένδυσης ΕΤΕ
         s1=0 #συνολικος αριθμός μετοχών ΕΤΕ

         #υπολογισμός του συνολικού ποσού επένδυσης και του αριθμού των μετοχών ΕΤΕ
         for x in ddates_new:
             for i in range(len(df1)):
                 if df1.loc[i,'Ημερομηνια']==x:
                     sum1+=df1.loc[i,'Κλείσιμο']
                     s1+=1
                     lt1.append((df1.loc[i,'Κλείσιμο']))
                     buy_date1.append((df1.loc[i,'Ημερομηνια']))
                     sum_stock1.append(s1)
                     money_invest1.append(sum1)

         #εμφάνιση των βασικών αποτελεσμάτων
         print("Σύνολο επενδύσεων στην ΕΤΕ =",round(sum1,2),"€")
         print("Σύνολο μετοχών =",s1)
         print("Μέση τιμή αγοράς μετοχής",round((sum1/s1),2),"€")

         Σύνολο επενδύσεων στην ΕΤΕ = 210.37 €
         Σύνολο μετοχών = 78
         Μέση τιμή αγοράς μετοχής 2.7 €
```

Όπως παρατηρούμε με βάση την στρατηγική αγορών ενός τεμαχίου της μετοχής της ΕΤΕ την τελευταία εργάσιμη ημέρα κάθε μήνα από τις αρχές του 2017 και μέχρι το τέλος Ιουνίου 2023, έχουμε επενδύσει συνολικά 210.37 € αποκτώντας 78 τεμάχια μετοχών με μέση τιμή κτήσης τα 2.7 €/μετοχή ΕΤΕ.

Στη συνέχεια εντοπίζουμε με τη τιμή της μετοχής στις 21-7-2023 με τη βοήθεια των παρακάτω εντολών.

```
In [92]: timi1=df1.loc[0][1] # η τιμή της μετοχής στις 21-7-2023
         print("Η τιμή της μετοχής της ΕΤΕ στις 21-7-2023:",timi1,'€')

         Η τιμή της μετοχής της ΕΤΕ στις 21-7-2023: 6.21 €
```

Όπως παρατηρούμε η τιμή κλεισίματος της μετοχής της ΕΤΕ στις 21-7-2023 ήταν 6.21 €.

Για να μπορέσουμε να υπολογίσουμε την απόδοση της επένδυσής μας εκτελούμε τα ακόλουθα:

Στη μεταβλητή result1 υπολογίζουμε το καθαρό μη πραγματοποιηθέν κέρδος ως τη διαφορά μεταξύ της τρέχουσας αποτίμησης του χαρτοφυλακίου μας (στις 21-7-2023) πολλαπλασιάζοντας την τιμή κλεισίματος εκείνη την ημερομηνία (timi1=6.21 €) με το πλήθος των τεμαχίων (sum1=72) και αφαιρώντας το συνολικό ποσό επένδυσης (sum1=193.97)

Κατόπιν η μεταβλητή roi θα υπολογίζει την απόδοση της επένδυσης στην μετοχή της ΕΤΕ ως το λόγο καθαρό μη πραγματοποιηθέν κέρδος(result1) προς το συνολικό ποσό επένδυσης (sum1).

```
In [93]: #υπολογισμός των αποτελεσμάτων της επένδυσης

         # κέρδος=αξία επενδυσης στις 21-7-2023 μειον το ποσο που επενδύθηκε
         result1=round((timi1*s1)-sum1,2)

         # απόδοση της επένδυσης= κέρδος/ποσό της επένδυσης
         roi1=round((result1/sum1)*100,2)

         print("Αποτελέσματα της επένδυσης στην ΕΤΕ")

         if result1<0:
             print("Καθαρή μη πραγματοποιηθείσα ζημία = ",result1,'€')
         else:
             print("Καθαρό μη πραγματοποιηθέν κέρδος = ",result1,'€')

         print("Απόδοση επένδυσης στην ΕΤΕ (από 1-1-2017 έως 21-7-2023) =",roi1,"%")
```

```
Αποτελέσματα της επένδυσης στην ΕΤΕ
Καθαρό μη πραγματοποιηθέν κέρδος =  274.01 €
Απόδοση επένδυσης στην ΕΤΕ (από 1-1-2017 έως 21-7-2023) = 130.25 %
```

Σύμφωνα με τα παραπάνω αποτελέσματα έχουμε για τη μετοχή της ΕΤΕ καθαρό μη πραγματοποιηθέν κέρδος 274.01 € και η απόδοση της επένδυσής μας ανήλθε στο 130.25 %.

Γνωρίζοντας ότι η λίστα buy_date1 περιέχει τις ημερομηνίες αγορών και η λίστα lt1 τις τιμές κλεισίματος που χρησιμοποιήσαμε ως τιμές για τις αγορές μας, εμφανίζουμε τα περιεχόμενά τους κατά αντιστοιχία.

```
In [94]: print("Ημερομηνία","Τιμή")
         for i in range(len(buy_date1)):
             print (buy_date1[i].date(),lt1[i])
```

```
Ημερομηνία Τιμή
2017-01-31 2.21
2017-02-28 2.4
2017-03-31 2.41
2017-04-28 2.86
2017-05-31 3.11
2017-06-30 3.33
2017-07-31 3.42
2017-08-31 3.41
2017-09-29 2.87
2017-10-31 2.84
2017-11-30 2.6
2017-12-29 3.19
2018-01-31 3.42
2018-02-28 3.052
2018-03-29 2.612
2018-04-30 3.46
2018-05-31 2.694
2018-06-29 2.63
2018-07-31 2.782
2018-08-31 2.448
2018-09-28 1.75
2018-10-31 1.53
2018-11-30 1.024
2018-12-31 1.1
2019-01-31 0.9775
2019-02-28 1.59
2019-03-29 1.556
2019-04-30 1.954
2019-05-31 2.368
2019-06-28 2.41
2019-07-31 2.65
2019-08-30 2.638
2019-09-30 2.793
2019-10-31 3.04
2019-11-29 3.074
2019-12-31 3.02
2020-01-30 2.917
2020-02-28 2.0
2020-03-31 1.239
2020-04-30 1.239
2020-05-29 1.209
2020-06-30 1.248
2020-07-31 1.112
2020-08-31 1.141
2020-09-30 1.08
2020-10-30 0.8966
2020-11-30 1.517
2020-12-31 2.261
2021-01-29 1.951
2021-02-26 2.1
2021-03-31 2.48
2021-04-29 2.585
2021-05-31 2.584
2021-06-30 2.4
2021-07-30 2.39
2021-08-31 2.562
2021-09-30 2.42
2021-10-29 2.72
2021-11-30 2.637
2021-12-31 2.932
2022-01-31 3.49
2022-02-28 3.3
2022-03-31 3.354
2022-04-29 3.825
2022-05-31 3.543
2022-06-30 2.823
2022-07-29 3.048
2022-08-31 3.224
2022-09-30 3.026
2022-10-31 3.67
2022-11-30 3.87
2022-12-30 3.747
2023-01-31 4.346
2023-02-28 5.3
2023-03-31 4.47
2023-04-28 4.74
2023-05-31 5.8
2023-06-30 5.954
```

Όπως ακριβώς δουλέψαμε για την μετοχή της ΕΤΕ υλοποιούμε τα αντίστοιχα για τις υπόλοιπες μετοχές με τις απαραίτητες προσθήκες λιστών και μεταβλητών. Η φιλοσοφία μας παραμένει η ίδια.

Απόδοση επένδυσης στον ΟΤΕ

In [95]:
```
lt2=[] #  λίστα με τιμές αγοράς κάθε τελευταία εργάσιμη ημέρα κάθε μήνα
buy_date2=[] #  λίστα με ημερομηνίες αγορών
sum_stock2=[] #  λίστα με το σύνολο μετοχών κάθε τέλευταία εργάσιμη ημέρα ανά μήνα
money_invest2=[] #  λίστα με συνολικό ποσό επένδυσης κάθε τέλος του μήνα
sum2=0 #συνολικό ποσό επένδυσης ΟΤΕ
s2=0 #συνολικος αριθμός μετοχών ΟΤΕ

#υπολογισμός του συνολικού ποσού επένδυσης και του αριθμού των μετοχών ΟΤΕ
for x in ddates_new:
    for i in range(len(df2)):
        if df2.loc[i,'Ημερομηνια']==x:
            sum2+=df2.loc[i,'Κλείσιμο']
            s2+=1
            lt2.append((df2.loc[i,'Κλείσιμο']))
            buy_date2.append((df2.loc[i,'Ημερομηνια']))
            sum_stock2.append(s2)
            money_invest2.append(sum2)

#εμφάνιση των βασικών αποτελεσμάτων
print("Σύνολο επενδύσεων στον ΟΤΕ =",round(sum2,2),"€")
print("Σύνολο μετοχών =",s2)
print("Μέση τιμή αγοράς μετοχής",round((sum2/s2),2),"€")
```

```
Σύνολο επενδύσεων στον ΟΤΕ = 1010.29 €
Σύνολο μετοχών = 78
Μέση τιμή αγοράς μετοχής 12.95 €
```

In [96]:
```
timi2=df2.loc[0][1] # η τιμή της μετοχής στις 21-7-2023
print("Η τιμή της μετοχής του ΟΤΕ στις 21-7-2023:",timi2,'€')
```

```
Η τιμή της μετοχής του ΟΤΕ στις 21-7-2023: 14.46 €
```

In [97]:
```
#υπολογισμός των αποτελεσμάτων της επένδυσης

# κέρδος=αξία επενδυσης στις 21-7-2023 μειον το ποσο που επενδύθηκε
result2=round((timi2*s2)-sum2,2)

# απόδοση της επένδυσης= κέρδος/ποσό της επένδυσης
roi2=round((result2/sum2)*100,2)


print("Αποτελέσματα της επένδυσης στον ΟΤΕ")

if result2<0:
    print("Καθαρή μη πραγματοποιηθείσα ζημία =",result2,"€")
else:
    print("Καθαρό μη πραγματοποιηθέν κέρδος = ",result2,"€")

print("Απόδοση επένδυσης στον ΟΤΕ (από 1-1-2017 έως 21-7-2023) =",roi2,"%")
```

```
Αποτελέσματα της επένδυσης στον ΟΤΕ
Καθαρό μη πραγματοποιηθέν κέρδος =  117.59 €
Απόδοση επένδυσης στον ΟΤΕ (από 1-1-2017 έως 21-7-2023) = 11.64 %
```

Απόδοση επένδυσης στη ΔΕΗ

In [98]:
```
lt3=[] #  λίστα με τιμές αγοράς κάθε τελευταία εργάσιμη ημέρα κάθε μήνα
buy_date3=[] #  λίστα με ημερομηνίες αγορών
sum_stock3=[] #  λίστα με το σύνολο μετοχών κάθε τέλευταία εργάσιμη ημέρα ανά μήνα
money_invest3=[] #  λίστα με συνολικό ποσό επένδυσης κάθε τέλος του μήνα
sum3=0 #συνολικό ποσό επένδυσης ΔΕΗ
s3=0 #συνολικος αριθμός μετοχών ΔΕΗ

#υπολογισμός του συνολικού ποσού επένδυσης και του αριθμού των μετοχών ΔΕΗ
for x in ddates_new:
    for i in range(len(df3)):
        if df3.loc[i,'Ημερομηνια']==x:
            sum3+=df3.loc[i,'Κλείσιμο']
            s3+=1
            lt3.append((df3.loc[i,'Κλείσιμο']))
            buy_date3.append((df3.loc[i,'Ημερομηνια']))
            sum_stock3.append(s3)
            money_invest3.append(sum3)

#εμφάνιση των βασικών αποτελεσμάτων
print("Σύνολο επενδύσεων στη ΔΕΗ =",round(sum3,2),"€")
print("Σύνολο μετοχών =",s3)
print("Μέση τιμή αγοράς μετοχής",round((sum3/s3),2),"€")
```

```
Σύνολο επενδύσεων στη ΔΕΗ = 373.86 €
Σύνολο μετοχών = 78
Μέση τιμή αγοράς μετοχής 4.79 €
```

In [99]:
```
timi3=df3.loc[0][1] # η τιμή της μετοχής στις 21-7-2023
print("Η τιμή της μετοχής της ΔΕΗ στις 21-7-2023:",timi3,'€')
```

```
Η τιμή της μετοχής της ΔΕΗ στις 21-7-2023: 11.12 €
```

In [100]:
```
#υπολογισμός των αποτελεσμάτων της επένδυσης

# κέρδος=αξία επενδυσης στις 21-7-2023 μειον το ποσο που επενδύθηκε
result3=round((timi3*s3)-sum3,2)

# απόδοση της επένδυσης= κέρδος/ποσό της επένδυσης
roi3=round((result3/sum3)*100,2)


print("Αποτελέσματα της επένδυσης στη ΔΕΗ")

if result3<0:
    print("Καθαρή μη πραγματοποιηθείσα ζημία =",result3,"€")
else:
    print("Καθαρό μη πραγματοποιηθέν κέρδος = ",result3,"€")

print("Απόδοση επένδυσης στη ΔΕΗ (από 1-1-2017 έως 21-7-2023) =",roi3,"%")
```

```
Αποτελέσματα της επένδυσης στη ΔΕΗ
Καθαρό μη πραγματοποιηθέν κέρδος =  493.5 €
Απόδοση επένδυσης στη ΔΕΗ (από 1-1-2017 έως 21-7-2023) = 132.0 %
```

Απόδοση επένδυσης στο ΜΥΤΙΛΗΝΑΙΟ

In [101]:
```
lt4=[] #  λίστα με τιμές αγοράς κάθε τελευταία εργάσιμη ημέρα κάθε μήνα
buy_date4=[] #  λίστα με ημερομηνίες αγορών
sum_stock4=[] #  λίστα με το σύνολο μετοχών κάθε τέλευταία εργάσιμη ημέρα ανά μήνα
money_invest4=[] #  λίστα με συνολικό ποσό επένδυσης κάθε τέλος του μήνα
sum4=0 #συνολικό ποσό επένδυσης ΜΥΤΙΛ
s4=0 #συνολικος αριθμός μετοχών ΜΥΤΙΛ

#υπολογισμός του συνολικού ποσού επένδυσης και του αριθμού των μετοχών ΜΥΤΙΛ
for x in ddates_new:
    for i in range(len(df4)):
        if df4.loc[i,'Ημερομηνια']==x:
            sum4+=df4.loc[i,'Κλείσιμο']
            s4+=1
            lt4.append((df4.loc[i,'Κλείσιμο']))
            buy_date4.append((df4.loc[i,'Ημερομηνια']))
            sum_stock4.append(s4)
            money_invest4.append(sum4)

#εμφάνιση των βασικών αποτελεσμάτων
print("Σύνολο επενδύσεων στο ΜΥΤΙΛ =",round(sum4,2),"€")
print("Σύνολο μετοχών =",s4)
print("Μέση τιμή αγοράς μετοχής",round((sum4/s4),2),"€")
```

```
Σύνολο επενδύσεων στο ΜΥΤΙΛ = 950.1 €
Σύνολο μετοχών = 78
Μέση τιμή αγοράς μετοχής 12.18 €
```

In [102]:
```
timi4=df4.loc[0][1] # η τιμή της μετοχής στις 21-7-2023
print("Η τιμή της μετοχής του ΜΥΤΙΛ στις 21-7-2023:",timi4,'€')
```

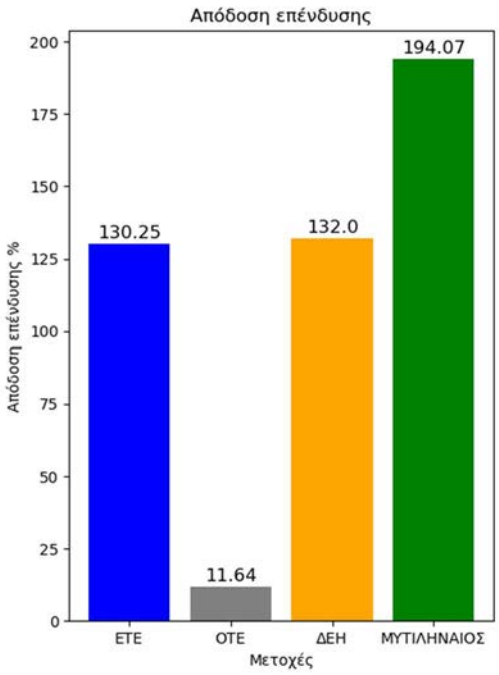
```
Η τιμή της μετοχής του ΜΥΤΙΛ στις 21-7-2023: 35.82 €
```

In [103]:
```
#υπολογισμός των αποτελεσμάτων της επένδυσης

# κέρδος=αξία επενδυσης στις 21-7-2023 μειον το ποσο που επενδύθηκε
result4=round((timi4*s4)-sum4,2)

# απόδοση της επένδυσης= κέρδος/ποσό της επένδυσης
roi4=round((result4/sum4)*100,2)


print("Αποτελέσματα της επένδυσης στο ΜΥΤΙΛ")

if result4<0:
    print("Καθαρή μη πραγματοποιηθείσα ζημία =",result4,"€")
else:
    print("Καθαρό μη πραγματοποιηθέν κέρδος = ",result4,"€")

print("Απόδοση επένδυσης στο ΜΥΤΙΛ (από 1-1-2017 έως 21-7-2023) =",roi4,"%")
```

```
Αποτελέσματα της επένδυσης στο ΜΥΤΙΛ
Καθαρό μη πραγματοποιηθέν κέρδος =  1843.86 €
Απόδοση επένδυσης στο ΜΥΤΙΛ (από 1-1-2017 έως 21-7-2023) = 194.07 %
```

Στη συνέχεια για να έχουμε μια πιο ολοκληρωμένη αποτύπωση των αποδόσεων που είχαμε ανά μετοχή με βάση τη στρατηγική που ακολουθήσαμε, δημιουργούμε ένα διάγραμμα με ιστογράμματα των αντίστοιχων αποδόσεων (η λίστα ROI περιέχει ως στοιχεία τις αποδόσεις κάθε μετοχής και η λίστα stock τα ονόματα των μετοχών)

```
In [104]: plt.figure(figsize=(5,7))

stock=['ΕΤΕ','ΟΤΕ','ΔΕΗ','ΜΥΤΙΛΗΝΑΙΟΣ']

ROI=[roi1,roi2,roi3,roi4]

col=['Blue','Grey','Orange', 'Green']

plt.bar(stock,ROI,color=col)
plt.title("Απόδοση επένδυσης")
plt.xlabel("Μετοχές")
plt.ylabel("Απόδοση επένδυσης %")
for index,data_s in enumerate(ROI):
    plt.text(x=index, y=data_s+2, s=f"{data_s}", ha="center", fontsize=12)
```



Υπολογισμός απόδοσης χαρτοφυλακίου

Αφού μελετήσαμε και αναλύσαμε τις αποδόσεις κάθε μιας μετοχής, επόμενος στόχος είναι να μελετήσουμε την απόδοση συνολικά όλου του χαρτοφυλακίου μας. Για το σκοπό αυτό δημιουργούμε μια μεταβλητή sum_all που αντιστοιχεί στο συνολικό ποσό επένδυσης.

```
In [105]: sum_all=round((sum1+sum2+sum3+sum4),2)
          print('Συνολική αξία επένδυσης:',sum_all,'€')

          Συνολική αξία επένδυσης: 2544.62 €
```

Σύμφωνα με το παραπάνω, το συνολικό ποσό επένδυσης ήταν 2544.62 €.

Στη συνέχεια δημιουργούμε μια μεταβλητή με όνομα result_all που αντιστοιχεί στα συνολικά δυνητικά κέρδη που είχαμε με βάση τη στρατηγική μας.

```
In [106]: result_all=round((result1+result2+result3+result4),2)
          print ('Δυνητικά κέρδη:',result_all,'€')

          Δυνητικά κέρδη: 2728.96 €
```

Τα συνολικά δυνητικά κέρδη του χαρτοφυλακίου μας ήταν 2728.96 €.

Με τη μεταβλητή ap υπολογίζουμε την απόδοση του χαρτοφυλακίου μας διαιρώντας τα συνολικά δυνητικά κέρδη (result_all) με το συνολικό ποσό επένδυσης (sum_all)

```
In [107]: ap=result_all/sum_all
          print ('Απόδοση χαρτοφυλακίου:',round((ap*100),2),'%')

          Απόδοση χαρτοφυλακίου: 107.24 %
```

Όπως παρατηρούμε η συνολική απόδοση του χαρτοφυλακίου μας ήταν 107.24 %.

Τέλος υπολογίζουμε το μέσο ποσό επένδυσης κάθε τελευταία εργάσιμη ημέρα κάθε μήνα.

```
In [108]: average_price_all=round((sum_all/s1),2)
          print('Μέση αξία επένδυσης 1 φορά το μήνα σε σύνολο',s1,' μηνών:',average_price_all,'€')

          Μέση αξία επένδυσης 1 φορά το μήνα σε σύνολο 78  μηνών: 32.62 €
```

Έτσι βλέπουμε ότι με ένα ποσό κατά μέσο όρο 32.62 € κάθε τελευταία εργάσιμη ημέρα κάθε μήνα από τις αρχές του 2017 έως και το τέλος του 1ου εξαμήνου του 2023 θα είχαμε δαπανήσει για επένδυση στις 4 μετοχές που μελετάμε το συνολικό ποσό των 2544.62 € έχοντας συνολικά δυνητικά κέρδη 2728.96 € και μια απόδοση της τάξης 107.24 %.

ΥΠΟΛΟΓΙΣΜΟΣ ΚΟΣΤΟΣ ΚΕΡΔΟΣ ΑΝΑ ΜΗΝΑ ΜΕ ΑΥΤΗ ΤΗΝ ΚΙΝΗΣΗ

Ακολούθως δημιουργούμε μια λίστα money_invest_all που θα περιέχει το συνολικό ποσό επένδυσης σε κάθε ημερομηνία αγοράς.

```
In [109]: sum_stock1.sort(reverse=True) #  λίστα με το σύνολο μετοχών κάθε τέλος του μήνα

          money_invest_all=[]
          for i in range(len(money_invest1)):
              money_invest_all.append(money_invest1[i]
                                      +money_invest2[i]
                                      +money_invest3[i]
                                      +money_invest4[i])

          sum_stock1.sort()
          money_invest_all.sort()
```

Δημιουργούμε ένα λεξικό (dictionary) με όνομα dict_all που περιέχει ως στοιχεία:

Την ημερομηνία αγορών, τις τιμές κλεισίματος τις αντίστοιχες ημερομηνίες για καθεμιά μετοχή, τον αριθμό μετοχών που έχουμε ανά μετοχή και το συνολικό ποσό που έχουμε επενδύσει πάλι αντίστοιχα στις συγκεκριμένες ημερομηνίες αγορών.

```
In [110]: dict_all = {'Ημερομηνια':buy_date1 ,
                      'ETE': lt1,
                      'OTE': lt2,
                      'ΔΕΗ': lt3,
                      'MYT': lt4,
                      'Αρ. μετοχών/μετοχή': sum_stock1,
                      'Ποσό επένδυσης': money_invest_all}
```

Κατόπιν μετατρέπουμε το λεξικό αυτό σε ένα dataframe με όνομα df20 και εμφανίζουμε τις 5 πρώτες εγγραφές.

```
In [111]: df20 = pd.DataFrame(dict_all)
          df20.head()
```

Out[111]:

	Ημερομηνια	ETE	OTE	ΔΕΗ	MYT	Αρ. μετοχών/μετοχή	Ποσό επένδυσης
0	2017-01-31	2.21	8.43	2.59	6.424	1	19.654
1	2017-02-28	2.40	8.47	2.75	7.083	2	40.357
2	2017-03-31	2.41	8.80	2.93	7.083	3	61.580
3	2017-04-28	2.86	8.93	3.38	6.893	4	83.643
4	2017-05-31	3.11	10.15	4.70	7.761	5	109.364

Στη συνέχεια εμφανίζουμε τις 5 τελευταίες εγγραφές.

```
In [112]: df20.tail()
```

Out[112]:

	Ημερομηνια	ΕΤΕ	ΟΤΕ	ΔΕΗ	ΜΥΤ	Αρ. μετοχών/μετοχή	Ποσό επένδυσης
73	2023-02-28	5.300	14.52	8.300	25.52	74	2316.4091
74	2023-03-31	4.470	13.49	7.980	26.20	75	2368.5491
75	2023-04-28	4.740	13.25	7.820	26.30	76	2420.6591
76	2023-05-31	5.800	14.32	9.905	29.48	77	2480.1641
77	2023-06-30	5.954	15.71	10.450	32.34	78	2544.6181

Παράλληλα επιβεβαιώνουμε με αυτό τον τρόπο ότι τελικά κατέχουμε 78 τεμάχια από κάθε μετοχή και το συνολικό ποσό επένδυσης στο τέλος του 1ου εξαμήνου του 2023 είναι 2544.62 €.

Στη συνέχεια δημιουργούμε μια νέα στήλη (πεδίο) με όνομα axia-xart που θα περιέχει την αντίστοιχη αξία του χαρτοφυλακίου μας στις ημερομηνίες των αγορών μας.

```
In [113]: df20['axia-xart'] = (df20['Αρ. μετοχών/μετοχή']
                               *(df20['ΕΤΕ']+df20['ΟΤΕ']+df20['ΔΕΗ']+df20['ΜΥΤ']))
          df20.head()
```

Out[113]:

	Ημερομηνια	ΕΤΕ	ΟΤΕ	ΔΕΗ	ΜΥΤ	Αρ. μετοχών/μετοχή	Ποσό επένδυσης	axia-xart
0	2017-01-31	2.21	8.43	2.59	6.424	1	19.654	19.654
1	2017-02-28	2.40	8.47	2.75	7.083	2	40.357	41.406
2	2017-03-31	2.41	8.80	2.93	7.083	3	61.580	63.669
3	2017-04-28	2.86	8.93	3.38	6.893	4	83.643	88.252
4	2017-05-31	3.11	10.15	4.70	7.761	5	109.364	128.605

```
In [114]: df20.tail()
```

Out[114]:

	Ημερομηνια	ΕΤΕ	ΟΤΕ	ΔΕΗ	ΜΥΤ	Αρ. μετοχών/μετοχή	Ποσό επένδυσης	axia-xart
73	2023-02-28	5.300	14.52	8.300	25.52	74	2316.4091	3969.360
74	2023-03-31	4.470	13.49	7.980	26.20	75	2368.5491	3910.500
75	2023-04-28	4.740	13.25	7.820	26.30	76	2420.6591	3960.360
76	2023-05-31	5.800	14.32	9.905	29.48	77	2480.1641	4581.885
77	2023-06-30	5.954	15.71	10.450	32.34	78	2544.6181	5027.412

Δημιουργούμε ένα νέο dataframe με όνομα df22 το οποίο θα περιέχει μόνο τις ημερομηνίες αγορών, και τα αντίστοιχα ποσά επένδυσης και την αξία του χαρτοφυλακίου μας.

```
In [115]: df22= df20.iloc[: , [0, 6, 7]].copy()
          df22
```

Out[115]:

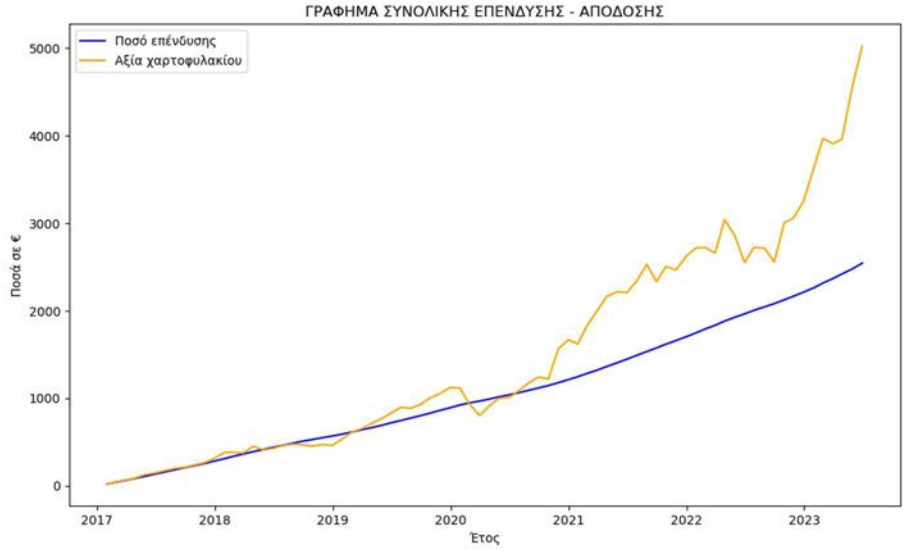
	Ημερομηνια	Ποσό επένδυσης	axia-xart
0	2017-01-31	19.6540	19.654
1	2017-02-28	40.3570	41.406
2	2017-03-31	61.5800	63.669
3	2017-04-28	83.6430	88.252
4	2017-05-31	109.3640	128.605
...	...	...	...
73	2023-02-28	2316.4091	3969.360
74	2023-03-31	2368.5491	3910.500
75	2023-04-28	2420.6591	3960.360
76	2023-05-31	2480.1641	4581.885
77	2023-06-30	2544.6181	5027.412

78 rows × 3 columns

Όπως παρατηρούμε από τον παραπάνω πίνακα το df22 έχει σχεδιαστεί σωστά με 78 έγγραφές και τις σωστές στήλες που επιλέξαμε παραπάνω.

Στη συνέχεια θέλουμε να αποτυπώσουμε γραφικά την πορεία της επένδυσής μας ως εξής. Μια γραμμή (μπλε) θα αποτυπώνει το ποσό που έχουμε συνολικά επενδύσει και μια γραμμή (πορτοκαλί) την αξία του χατοφυλακίου.

```
In [116]: plt.figure(figsize=(12,7))

          plt.plot(df22['Ημερομηνια'],df22['Ποσό επένδυσης'],color='blue',label='Ποσό επένδυσης')
          plt.plot(df22['Ημερομηνια'],df22['axia-xart'],color='orange',label='Αξία χαρτοφυλακίου')

          plt.title("ΓΡΑΦΗΜΑ ΣΥΝΟΛΙΚΗΣ ΕΠΕΝΔΥΣΗΣ - ΑΠΟΔΟΣΗΣ")
          plt.xlabel("Έτος")
          plt.ylabel("Ποσά σε €")
          plt.legend(title="")
          plt.show()
```



Σύμφωνα με το παραπάνω γράφημα παρατηρούμε τα εξής:

- Το χαρτοφυλάκιό μας άρχισε να αποδίδει με καλούς ρυθμούς μετά από 2,5 περίπου χρόνια δηλαδή από τα μέσα του 2019.
- Η έναρξη της πανδημίας την άνοιξη του 2020 διατάραξε την απόδοση του χαρτοφυλακίου μας.
- Αυτή η αρνητική εξέλιξη όμως δεν κράτησε σχεδόν καθόλου αφού από τα μέσα της ίδια χρονιάς (μετά από περίπου 3 μήνες) ξεκίνησε να αποδίδει με ταχείς ρυθμούς.
- Μια επιπλέον πρόσκαιρη όπως αποτυπώνεται επίσης αναταραχή υπήρξε και την άνοιξη έως το φθινόπωρο του 2022 λόγω του πολέμου Ρωσίας – Ουκρανίας. Από το τέλος του φθινοπώρου όμως του 2022 έως και τέλη του 1^{ου} εξαμήνου του 2023 βλέπουμε μια ραγδαία άνοδο της απόδοσης της επένδυσής μας.

Στη συνέχεια της ανάλυσής μας ταξινομούμε τις λίστες με τα συνολικά ποσά που έχουμε επενδύσει σε κάθε μετοχή για κάθε αντίστοιχη ημερομηνία.

```
In [117]: money_invest1.sort()
          money_invest2.sort()
          money_invest3.sort()
          money_invest4.sort()
```

Κατόπιν δημιουργούμε 4 λεξικά (ένα για κάθε μετοχή) όπου το κάθε ένα να περιλαμβάνει:

- Την Ημερομηνία αγοράς
- Την τιμή αγοράς
- Τον αριθμό τεμαχίων που έχουμε σε κάθε μια Ημερομηνία
- Το συνολικό ποσό επένδυσης σε κάθε μια Ημερομηνία

```
In [118]: dict_1 = {'Ημερομηνια':buy_date1 ,
                    'ΕΤΕ': lt1,
                    'Αρ. μετοχών/μετοχή': sum_stock1,
                    'Ποσό επένδυσης': money_invest1}

          dict_2 = {'Ημερομηνια':buy_date2 ,
                    'ΟΤΕ': lt2,
                    'Αρ. μετοχών/μετοχή': sum_stock2,
                    'Ποσό επένδυσης': money_invest2}

          dict_3 = {'Ημερομηνια':buy_date3 ,
                    'ΔΕΗ': lt3,
                    'Αρ. μετοχών/μετοχή': sum_stock3,
                    'Ποσό επένδυσης': money_invest3}

          dict_4 = {'Ημερομηνια':buy_date4 ,
                    'ΜΥΤ': lt4,
                    'Αρ. μετοχών/μετοχή': sum_stock4,
                    'Ποσό επένδυσης': money_invest4}
```

Μετατρέπουμε τα αντίστοιχα λεξικά σε dataframes.

```
In [119]: df91 = pd.DataFrame(dict_1)
          df92 = pd.DataFrame(dict_2)
          df93 = pd.DataFrame(dict_3)
          df94 = pd.DataFrame(dict_4)
```

Δημιουργούμε σε κάθε ένα dataframe μια νέα στήλη (axia-xart) η οποία θα περιέχει για κάθε ημερομηνία αγοράς την εκάστοτε τρέχουσα αξία της επένδυσής μας (σε κάθε μετοχή) στο χαρτοφυλάκιό μας.

```
In [120]: df91['axia-xart'] = (df91['Αρ. μετοχών/μετοχή']*df91['ΕΤΕ'])
          df92['axia-xart'] = (df92['Αρ. μετοχών/μετοχή']*df92['ΟΤΕ'])
          df93['axia-xart'] = (df93['Αρ. μετοχών/μετοχή']*df93['ΔΕΗ'])
          df94['axia-xart'] = (df94['Αρ. μετοχών/μετοχή']*df94['ΜΥΤ'])
```

Ενδεικτικά παρουσιάζουμε τι περιέχει το dataframe df91 που αφορά τη μετοχή της ΕΤΕ.

```
In [121]: df91
```

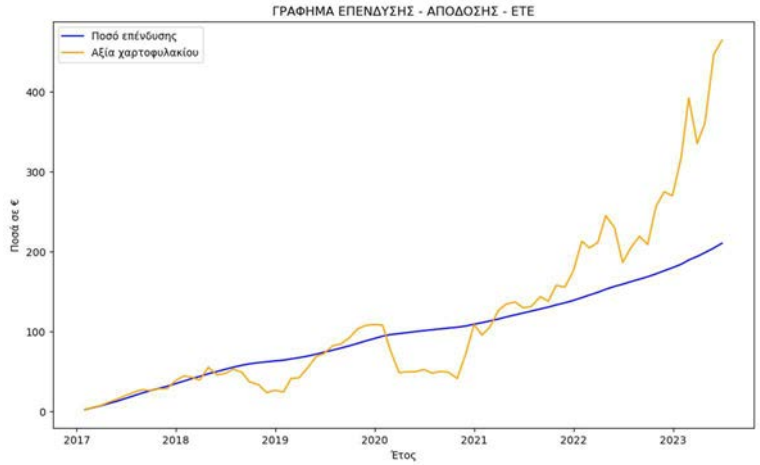
Out[121]:

	Ημερομηνια	ΕΤΕ	Αρ. μετοχών/μετοχή	Ποσό επένδυσης	axia-xart
0	2017-01-31	2.210	1	2.2100	2.210
1	2017-02-28	2.400	2	4.6100	4.800
2	2017-03-31	2.410	3	7.0200	7.230
3	2017-04-28	2.860	4	9.8800	11.440
4	2017-05-31	3.110	5	12.9900	15.550
...	...	...	...	...	...
73	2023-02-28	5.300	74	189.4091	392.200
74	2023-03-31	4.470	75	193.8791	335.250
75	2023-04-28	4.740	76	198.6191	360.240
76	2023-05-31	5.800	77	204.4191	446.600
77	2023-06-30	5.954	78	210.3731	464.412

78 rows × 5 columns

Δημιουργούμε γράφημα που αποτυπώνει την αξία της επένδυσής μας (μπλε γραμμή) και την εκάστοτε τρέχουσα αξία της επένδυσής μας σε κάθε μια ημερομηνία της μετοχής της ΕΤΕ.

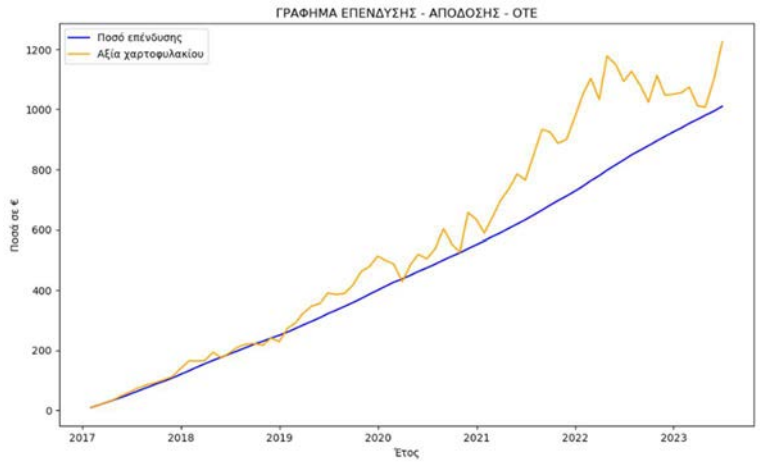
```
In [122]: plt.figure(figsize=(12,7))

          plt.plot(df91['Ημερομηνια'],df91['Ποσό επένδυσης'],color='blue',label='Ποσό επένδυσης')
          plt.plot(df91['Ημερομηνια'],df91['axia-xart'],color='orange',label='Αξία χαρτοφυλακίου')

          plt.title("ΓΡΑΦΗΜΑ ΕΠΕΝΔΥΣΗΣ - ΑΠΟΔΟΣΗΣ - ΕΤΕ")
          plt.xlabel("Έτος")
          plt.ylabel("Ποσά σε €")
          plt.legend(title="")
          plt.show()
```



Όπως παρατηρούμε από το παραπάνω γράφημα η μετοχή της ΕΤΕ ξεκίνησε να μας αποδίδει από τα μέσα του 2019 αλλά έναρξη της πανδημίας την ταλαιπώρησε για ένα περίπου χρόνο μέχρι την άνοιξη του 2021. Από εκείνο το σημείο και μετά παρουσιάζει θετική απόδοση ενώ παρατηρούμε ακόμη μεγαλύτερη απόδοση από το φθινόπωρο του 2022 μέχρι και το τα τέλη του 1^{ου} εξαμήνου του 2023.

Στη συνέχεια, δημιουργούμε γράφημα που αποτυπώνει την αξία της επένδυσής μας (μπλε γραμμή) και την εκάστοτε τρέχουσα αξία της επένδυσής μας σε κάθε μια ημερομηνία της μετοχής του ΟΤΕ.

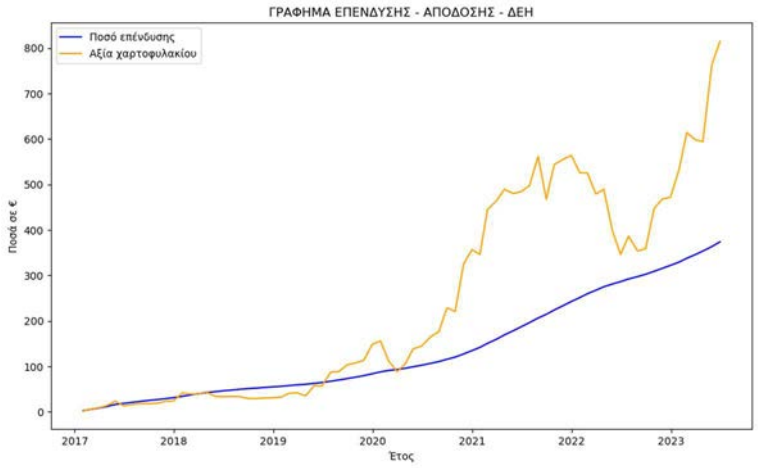
```
In [123]: plt.figure(figsize=(12,7))

          plt.plot(df92['Ημερομηνια'],df92['Ποσό επένδυσης'],color='blue',label='Ποσό επένδυσης')
          plt.plot(df92['Ημερομηνια'],df92['axia-xart'],color='orange',label='Αξία χαρτοφυλακίου')

          plt.title("ΓΡΑΦΗΜΑ ΕΠΕΝΔΥΣΗΣ - ΑΠΟΔΟΣΗΣ - ΟΤΕ")
          plt.xlabel("Έτος")
          plt.ylabel("Ποσά σε €")
          plt.legend(title="")
          plt.show()
```



Όπως παρατηρούμε από το παραπάνω γράφημα η μετοχή του ΟΤΕ πάντα παρέμενε σε θετικό έδαφος χωρίς ποτέ να υποαποδόσει της επένδυσής μας. Χαρακτηριστικό παράδειγμα είναι ότι ακόμη και στην έναρξη της πανδημίας την άνοιξη του 2020 και για ακόμη 2 φορές μετά από ένα χρόνο απλά άγγιξε το ποσό που είχαμε επενδύσει στη μετοχή. Από τις αρχές του 2021 και για ένα χρόνο μέχρι τις αρχές του 2022 είχε καλή απόδοση ενώ όλο το 2022 είχε μια πλαγιοκαθοδική κίνηση σε σχέση με την επένδυσή μας. Το 2ο τρίμηνο του 2023 άρχισε πάλι να αποδίδει.

Στη συνέχεια, δημιουργούμε γράφημα που αποτυπώνει την αξία της επένδυσής μας (μπλε γραμμή) και την εκάστοτε τρέχουσα αξία της επένδυσής μας σε κάθε μια ημερομηνία της μετοχής της ΔΕΗ.

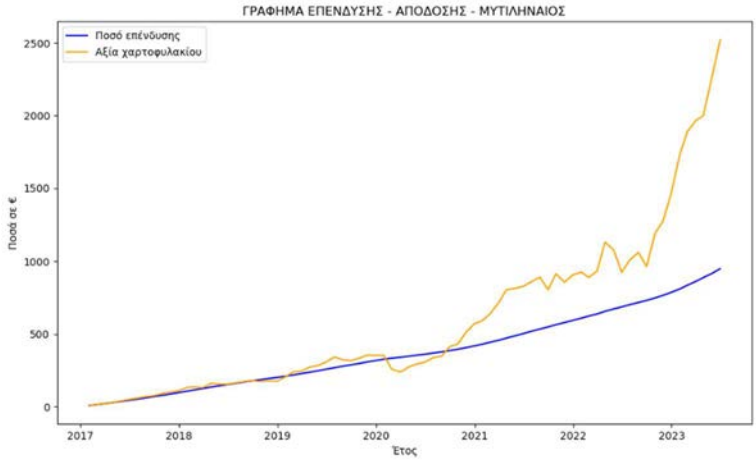
```
In [124]: plt.figure(figsize=(12,7))

          plt.plot(df93['Ημερομηνια'],df93['Ποσό επένδυσης'],color='blue',label='Ποσό επένδυσης')
          plt.plot(df93['Ημερομηνια'],df93['axia-xart'],color='orange',label='Αξία χαρτοφυλακίου')

          plt.title("ΓΡΑΦΗΜΑ ΕΠΕΝΔΥΣΗΣ - ΑΠΟΔΟΣΗΣ - ΔΕΗ")
          plt.xlabel("Έτος")
          plt.ylabel("Ποσά σε €")
          plt.legend(title="")
          plt.show()
```



Όπως παρατηρούμε από το παραπάνω γράφημα η μετοχή της ΔΕΗ είχε μια υποαπόδοση της επένδυσής μας από τα μέσα του 2028 έως τα μέσα του 2029 όπου και πέρασε σε θετικό έδαφος. Η έναρξη της πανδημίας όμως ανέκοψε αυτή την πορεία χωρίς ωστόσο να περάσει σε αρνητικό έδαφος. Πάρα πολύ γρήγορα ανέκαμψε αλλά με την έναρξη του πολέμου Ρωσίας – Ουκρανίας ΄στις αρχές του 2022 ξεκίνησε μια καθοδική κίνησε μέχρι και το φθινόπωρο του ίδιου έτους, ξεκινώντας και πάλι μια εντυπωσιακή ανοδική πορεία.

Στη συνέχεια δημιουργούμε γράφημα που αποτυπώνει την αξία της επένδυσής μας (μπλε γραμμή) και την εκάστοτε τρέχουσα αξία της επένδυσής μας σε κάθε μια ημερομηνία της μετοχής του ΜΥΤΙΛΗΝΑΙΟΥ.

```
In [125]: plt.figure(figsize=(12,7))

          plt.plot(df94['Ημερομηνια'],df94['Ποσό επένδυσης'],color='blue',label='Ποσό επένδυσης')
          plt.plot(df94['Ημερομηνια'],df94['axia-xart'],color='orange',label='Αξία χαρτοφυλακίου')

          plt.title("ΓΡΑΦΗΜΑ ΕΠΕΝΔΥΣΗΣ - ΑΠΟΔΟΣΗΣ - ΜΥΤΙΛΗΝΑΙΟΣ")
          plt.xlabel("Έτος")
          plt.ylabel("Ποσά σε €")
          plt.legend(title="")
          plt.show()
```

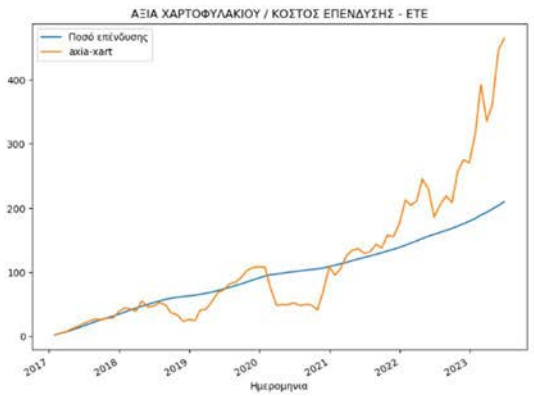


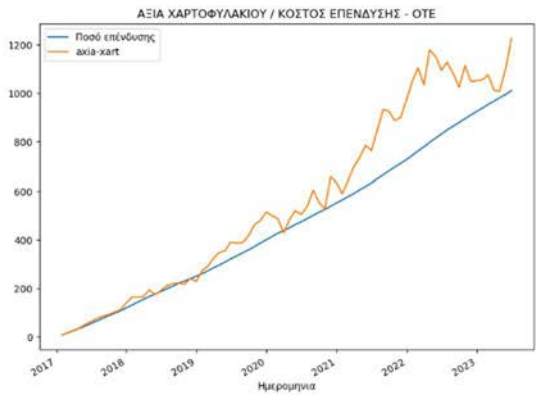
Όπως παρατηρούμε από το παραπάνω γράφημα η μετοχή του ΜΥΤΙΛΗΝΑΙΟΥ μέχρι και την έναρξη της πανδημίας κινούνταν σχεδόν παράλληλα με την επένδυσή μας χωρίς σχεδόν καμία απόδοση. Με την έναρξη της πανδημίας πέρασε σε αρνητικό έδαφος, όχι όμως να δημιουργεί αίσθηση μεγάλων απωλειών. Από το φθινόπωρο του 2020 ξεκίνησε μια ανοδική πορεία η οποία πήρε τα χαρακτηριστικά «έκρηξης» από το φθινόπωρο του 2022 μέχρι και τα τέλη του 1ου εξαμήνου του 2023 παρουσιάζοντας υπεραπόδοση στο χαρτοφυλάκιό μας.

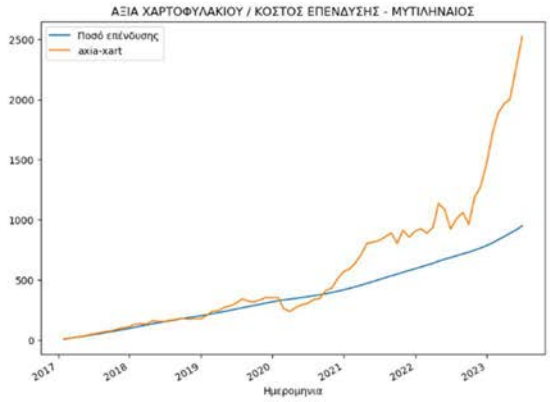
Στη συνέχεια παρουσιάζουμε και τα 4 παραπάνω γραφήματα σε ένα γράφημα.

```
In [126]: fig=plt.figure(figsize=(20,15))

          ax1=fig.add_subplot(2,2,1)
          df91.plot(kind='line',x='Ημερομηνια',y=['Ποσό επένδυσης','axia-xart'],ax=ax1)
          ax1.set_title("ΑΞΙΑ ΧΑΡΤΟΦΥΛΑΚΙΟΥ / ΚΟΣΤΟΣ ΕΠΕΝΔΥΣΗΣ - ΕΤΕ")

          ax2=fig.add_subplot(2,2,2)
          df92.plot(kind='line',x='Ημερομηνια',y=['Ποσό επένδυσης','axia-xart'],ax=ax2)
          ax2.set_title("ΑΞΙΑ ΧΑΡΤΟΦΥΛΑΚΙΟΥ / ΚΟΣΤΟΣ ΕΠΕΝΔΥΣΗΣ - ΟΤΕ")
          ax3=fig.add_subplot(2,2,3)
          df93.plot(kind='line',x='Ημερομηνια',y=['Ποσό επένδυσης','axia-xart'],ax=ax3)
          ax3.set_title("ΑΞΙΑ ΧΑΡΤΟΦΥΛΑΚΙΟΥ / ΚΟΣΤΟΣ ΕΠΕΝΔΥΣΗΣ - ΔΕΗ")
          ax4=fig.add_subplot(2,2,4)
          df94.plot(kind='line',x='Ημερομηνια',y=['Ποσό επένδυσης','axia-xart'],ax=ax4)
          ax4.set_title("ΑΞΙΑ ΧΑΡΤΟΦΥΛΑΚΙΟΥ / ΚΟΣΤΟΣ ΕΠΕΝΔΥΣΗΣ - ΜΥΤΙΛΗΝΑΙΟΣ")
```

```
Out[126]: Text(0.5, 1.0, 'ΑΞΙΑ ΧΑΡΤΟΦΥΛΑΚΙΟΥ / ΚΟΣΤΟΣ ΕΠΕΝΔΥΣΗΣ - ΜΥΤΙΛΗΝΑΙΟΣ')
```







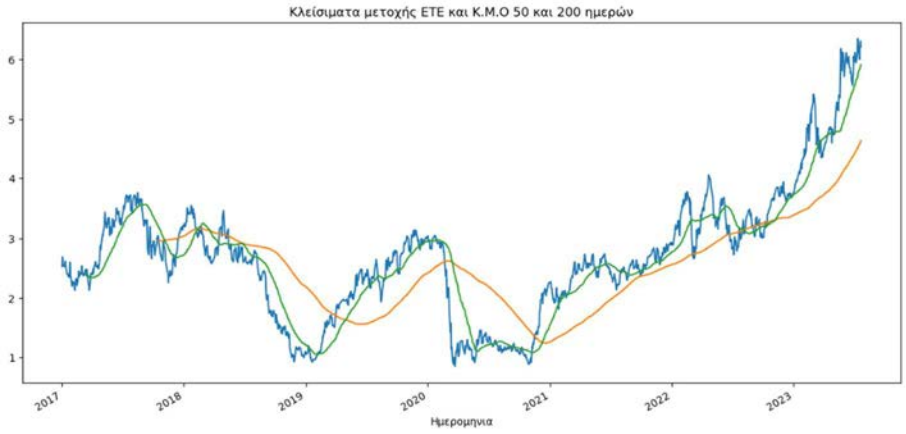
Σύμφωνα με τα παραπάνω γραφήματα μπορούμε να εξάγουμε κάποια συμπεράσματα συγκριτικά για τις μετοχές:

- Η μετοχή της ΔΕΗ ήταν αυτή που ταλαιπώρησε περισσότερο την απόδοση στο χαρτοφυλάκιό μας εξαιτίας της πανδημίας.
- Ο ΟΤΕ δεν πέρασε ποτέ σε αρνητική απόδοση της επένδυσής μας.
- Η μετοχή της ΔΕΗ ήταν αυτή που συνήλθε πιο γρήγορα από όλες από την πανδημία έχοντας υπεραπόδοση για ένα χρόνο (μέσα του 2020 έως τα μέσα του 2021) αλλά στην πορεία έχασε αρκετά από τα κέρδη ξεκινώντας μια δυνατή άνοδο αποδόσεων από τα τέλη του 2022.
- Η μετοχή του ΜΥΤΙΛΗΝΑΙΟΥ ξεκίνησε ουσιαστικά να αποδίδει μετά από 4 χρόνια από την έναρξη του επενδυτικού μας πλάνου, αλλά μπορεί τελικά δικαίως να χαρακτηριστεί ως η πιο αποτελεσματική επιλογή έχοντας χαρίσει πάρα πολύ σημαντικές αποδόσεις στο χαρτοφυλάκιό μας.
- Τέλος όλες οι μετοχές έχουν χαράξει ανοδική πορεία το 1ο εξάμηνο του 2023.

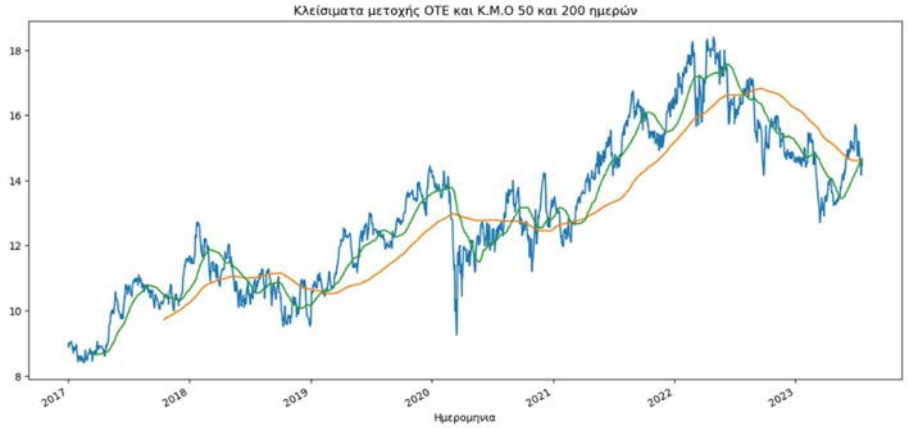
ΚΜΟ 50-200

Στη συνέχεια της ανάλυσής μας παρουσιάζουμε τα γραφήματα των μετοχών εμπλουτισμένα με τους κινητούς μέσους όρους 50 και 200 ημερών.

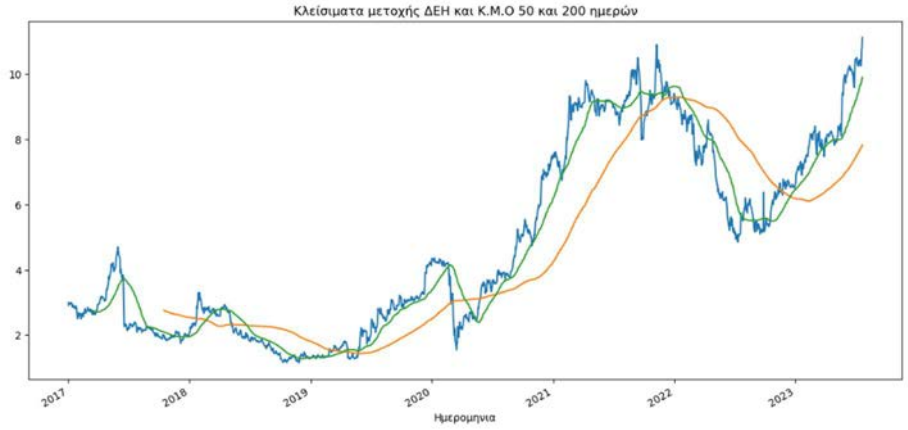
```
In [127]: df1=df1.set_index("Ημερομηνια", drop= True)

          df1['KMO50'] = df1['Κλείσιμο'].rolling(50).mean()
          df1['KMO200'] = df1['Κλείσιμο'].rolling(200).mean()
          df1['Κλείσιμο'].plot(figsize = (15,7))
          df1['KMO200'].plot()
          df1['KMO50'].plot()
          plt.title("Κλείσιματα μετοχής ΕΤΕ και Κ.Μ.Ο 50 και 200 ημερών")
```



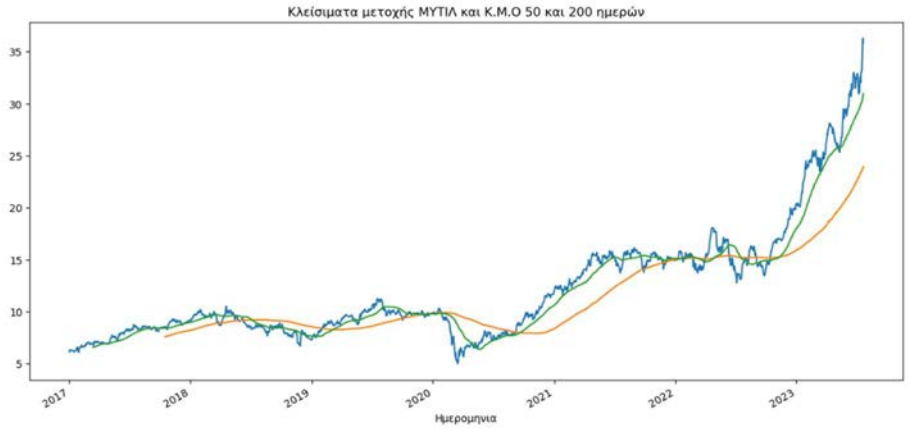
In [128]:
```
df2=df2.set_index("Ημερομηνια", drop= True)
df2['ΚΜΟ50'] = df2['Κλείσιμο'].rolling(50).mean()
df2['ΚΜΟ200'] = df2['Κλείσιμο'].rolling(200).mean()
df2['Κλείσιμο'].plot(figsize = (15,7))
df2['ΚΜΟ200'].plot()
df2['ΚΜΟ50'].plot()
plt.title("Κλείσιματα μετοχής ΟΤΕ και Κ.Μ.Ο 50 και 200 ημερών")
```



In [129]:
```
df3=df3.set_index("Ημερομηνια", drop= True)
df3['ΚΜΟ50'] = df3['Κλείσιμο'].rolling(50).mean()
df3['ΚΜΟ200'] = df3['Κλείσιμο'].rolling(200).mean()
df3['Κλείσιμο'].plot(figsize = (15,7))
df3['ΚΜΟ200'].plot()
df3['ΚΜΟ50'].plot()
plt.title("Κλείσιματα μετοχής ΔΕΗ και Κ.Μ.Ο 50 και 200 ημερών")
```



In [130]:
```
df4=df4.set_index("Ημερομηνια", drop= True)
df4['ΚΜΟ50'] = df4['Κλείσιμο'].rolling(50).mean()
df4['ΚΜΟ200'] = df4['Κλείσιμο'].rolling(200).mean()
df4['Κλείσιμο'].plot(figsize = (15,7))
df4['ΚΜΟ200'].plot()
df4['ΚΜΟ50'].plot()
plt.title("Κλείσιματα μετοχής ΜΥΤΙΛ και Κ.Μ.Ο 50 και 200 ημερών")
```



Εμφανίζουμε ενδεικτικά το dataframe που αφορά τη μετοχή του ΜΥΤΙΛΗΝΑΙΟΥ για να δούμε ότι έχουν προστεθεί 2 στήλες με τους κινητούς μέσους όρους 50 και 200 ημερών. Εντύπωση δεν πρέπει να μας προξενεί ότι οι αρχικές τιμές των Κ.Μ.Ο. δεν είναι ορισμένες (NaN) αφού για αυτές τις ημερομηνίες δεν μπορεί να υπολογιστεί Κ.Μ.Ο.

```
In [131]: df4
```

Out[131]:

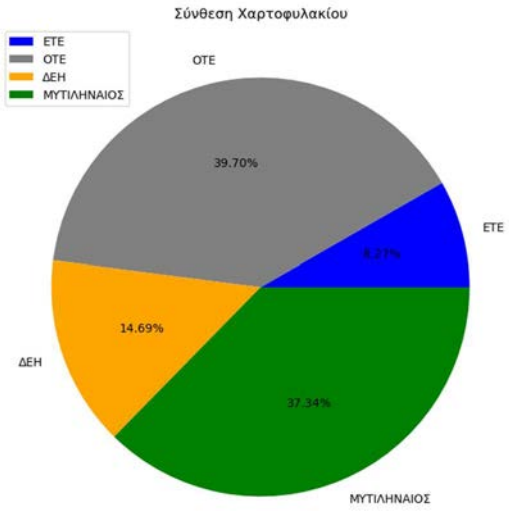
Ημερομηνια	Κλείσιμο	ΜΕΤ.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος	Day	KMO50	KMO200
2017-01-02	6.135	-0.2439	6.095	6.165	6.095	14037	86234.97	2	NaN	NaN
2017-01-03	6.315	2.9340	6.135	6.385	6.135	87554	550382.02	3	NaN	NaN
2017-01-04	6.315	0.0000	6.315	6.365	6.315	84689	536761.19	4	NaN	NaN
2017-01-05	6.295	-0.3167	6.295	6.355	6.275	57620	363360.18	5	NaN	NaN
2017-01-09	6.295	0.0000	6.255	6.315	6.215	49885	312608.30	9	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...
2023-07-17	33.300	1.4625	32.980	33.300	32.700	171926	5683682.32	17	30.1512	23.49830
2023-07-18	34.540	3.7237	33.340	35.000	33.340	412590	14278135.60	18	30.3268	23.60320
2023-07-19	35.680	3.3005	35.100	36.120	34.880	293517	10527500.36	19	30.5292	23.71210
2023-07-20	36.300	1.7377	36.000	36.300	35.820	216367	7820437.02	20	30.7464	23.82365
2023-07-21	35.820	-1.3223	36.300	36.400	35.820	145221	5235476.58	21	30.9564	23.92975

1630 rows × 10 columns

Στη συνέχεια δημιουργούμε ένα γράφημα πίτας το οποίο σύμφωνα με τη στρατηγική που ακολουθήσαμε για την αγορά μετοχών μας δείχνει τη σύνθεση του χαρτοφυλακίου μας με βάση την αξία της επένδυσής μας σε κάθε μια μετοχή.

```
In [132]: plt.figure(figsize=(15,8))

          stock=['ΕΤΕ','ΟΤΕ','ΔΕΗ','ΜΥΤΙΛΗΝΑΙΟΣ']

          shares=[sum1,sum2,sum3,sum4]

          col=['Blue','Grey','Orange','Green']

          plt.pie(shares,labels=stock,autopct="%1.2f%%",colors=col)

          plt.legend(title="",loc="upper left")

          plt.title("Σύνθεση Χαρτοφυλακίου")
```

```
Out[132]: Text(0.5, 1.0, 'Σύνθεση Χαρτοφυλακίου')
```

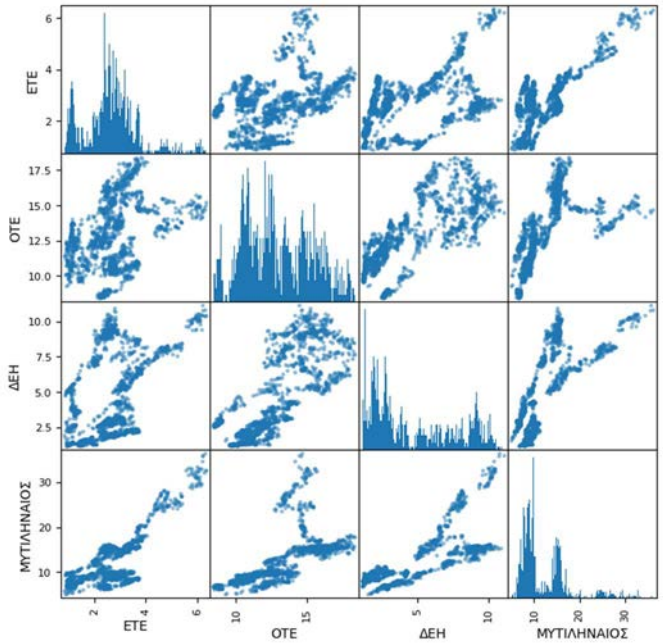


Κατόπιν δημιουργούμε ένα γράφημα διασποράς με συνδυασμό όλων των μετοχών μεταξύ τους.

Για την υλοποίηση δημιουργήσαμε ένα νέο dataframe με όνομα data το οποίο προκύπτει από την ένωση των τιμών κλεισίματος κάθε μετοχής.

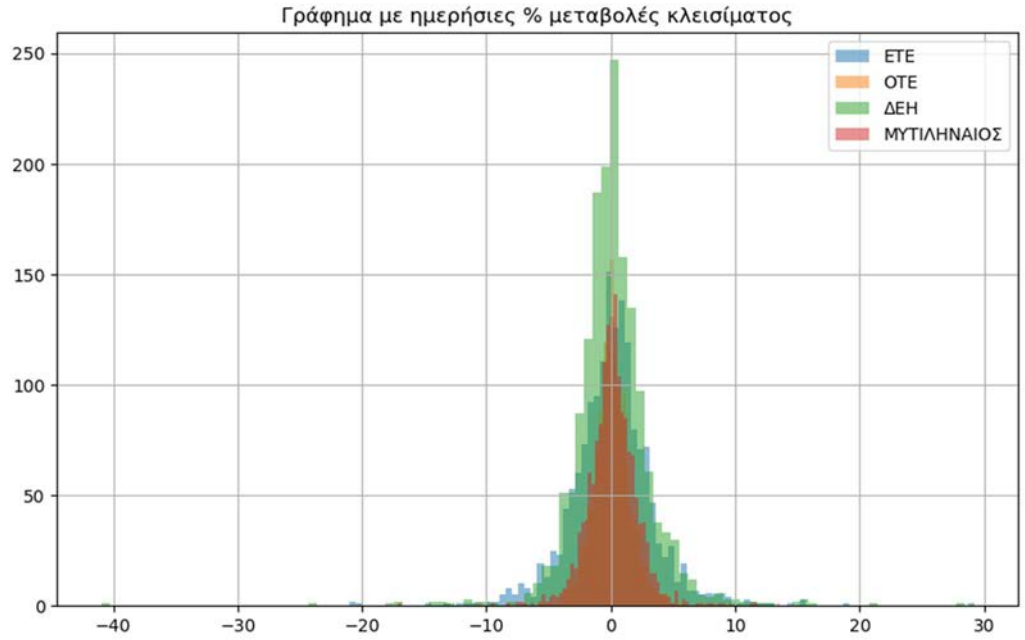
```
In [133]: from pandas.plotting import scatter_matrix

          data = pd.concat([df1['Κλείσιμο'],df2['Κλείσιμο'],df3['Κλείσιμο'],df4['Κλείσιμο']],axis = 1)
          data.columns = ['ETE','OTE','ΔΕΗ','ΜΥΤΙΛΗΝΑΙΟΣ']
          scatter_matrix(data, figsize = (8,8), hist_kwds= {'bins':250})
```

```
Out[133]: array([[<AxesSubplot:xlabel='ETE', ylabel='ETE'>,
                  <AxesSubplot:xlabel='OTE', ylabel='ETE'>,
                  <AxesSubplot:xlabel='ΔΕΗ', ylabel='ETE'>,
                  <AxesSubplot:xlabel='ΜΥΤΙΛΗΝΑΙΟΣ', ylabel='ETE'>],
                 [<AxesSubplot:xlabel='ETE', ylabel='OTE'>,
                  <AxesSubplot:xlabel='OTE', ylabel='OTE'>,
                  <AxesSubplot:xlabel='ΔΕΗ', ylabel='OTE'>,
                  <AxesSubplot:xlabel='ΜΥΤΙΛΗΝΑΙΟΣ', ylabel='OTE'>],
                 [<AxesSubplot:xlabel='ETE', ylabel='ΔΕΗ'>,
                  <AxesSubplot:xlabel='OTE', ylabel='ΔΕΗ'>,
                  <AxesSubplot:xlabel='ΔΕΗ', ylabel='ΔΕΗ'>,
                  <AxesSubplot:xlabel='ΜΥΤΙΛΗΝΑΙΟΣ', ylabel='ΔΕΗ'>],
                 [<AxesSubplot:xlabel='ETE', ylabel='ΜΥΤΙΛΗΝΑΙΟΣ'>,
                  <AxesSubplot:xlabel='OTE', ylabel='ΜΥΤΙΛΗΝΑΙΟΣ'>,
                  <AxesSubplot:xlabel='ΔΕΗ', ylabel='ΜΥΤΙΛΗΝΑΙΟΣ'>,
                  <AxesSubplot:xlabel='ΜΥΤΙΛΗΝΑΙΟΣ', ylabel='ΜΥΤΙΛΗΝΑΙΟΣ'>]],
                dtype=object)
```



Δημιουργούμε μια στήλη HM_MET η οποία αποτυπώνει την ποσοστιαία ημερήσια μεταβολή κάθε μιας μετοχής και δημιουργούμε ένα γράφημα με ιστογράμματα.

```
In [134]: #Μεταβλητότητα μετοχών
          df1['HM_MET'] = ((df1['Κλείσιμο']/df1['Κλείσιμο'].shift(1)) -1)*100
          df2['HM_MET'] = ((df2['Κλείσιμο']/df2['Κλείσιμο'].shift(1))-1)*100
          df3['HM_MET'] = ((df3['Κλείσιμο']/df3['Κλείσιμο'].shift(1)) - 1)*100
          df4['HM_MET'] = ((df4['Κλείσιμο']/df4['Κλείσιμο'].shift(1)) - 1)*100

          df1['HM_MET'].hist(bins = 100, label = 'ΕΤΕ', alpha = 0.5, figsize = (10,6))
          df2['HM_MET'].hist(bins = 100, label = 'ΟΤΕ', alpha = 0.5)
          df3['HM_MET'].hist(bins = 100, label = 'ΔΕΗ', alpha = 0.5)
          df4['HM_MET'].hist(bins = 100, label = 'ΜΥΤΙΛΗΝΑΙΟΣ', alpha = 0.5)
          plt.title("Γράφημα με ημερήσιες % μεταβολές κλεισίματος")
          plt.legend()

Out[134]: <matplotlib.legend.Legend at 0x25a84ece100>
```



Από το παραπάνω γράφημα διαπιστώνουμε ότι όλες οι μετοχές δεν χαρακτηρίζονται από μεγάλη και έντονη μεταβλητότητα. Η πλειοψηφία των μεταβολών είναι κοντά στο 0.

Παρακάτω δείχνουμε ενδεικτικά τα περιεχόμενα του df1 (μετοχή της ΔΕΗ) για να συγκρίνουμε τη στήλη HM_MET (ημερήσιες μεταβολές) με τη στήλη MET.% και παρατηρούμε ότι ταυτίζονται.

```
In [135]: df1
```

Out[135]:

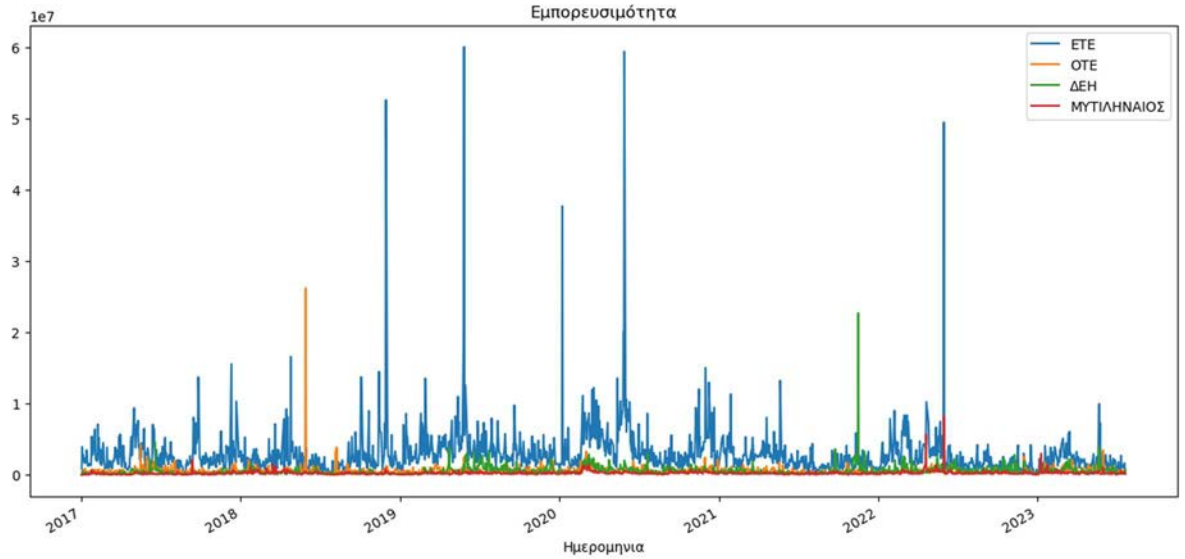
Ημερομηνια	Κλείσιμο	MET.%	Άνοιγμα	Υψηλό	Χαμηλό	Όγκος	Τζίρος	Day	KMO50	KMO200	HM_MET
2017-01-02	2.530	2.0161	2.500	2.550	2.480	858927.0	2168559.82	2	NaN	NaN	NaN
2017-01-03	2.690	6.3241	2.530	2.730	2.520	3940597.0	10367046.15	3	NaN	NaN	6.324111
2017-01-04	2.620	-2.6022	2.680	2.700	2.610	2073396.0	5465725.52	4	NaN	NaN	-2.602230
2017-01-05	2.550	-2.6718	2.590	2.600	2.530	2859241.0	7311028.08	5	NaN	NaN	-2.671756
2017-01-09	2.530	-0.7843	2.550	2.580	2.520	1670785.0	4247937.85	9	NaN	NaN	-0.784314
...	...	...	...	...	...	...	...	...	...	...	...
2023-07-17	6.000	-2.9126	6.216	6.218	5.988	987507.0	6005352.73	17	5.82808	4.570605	-2.912621
2023-07-18	6.202	3.3667	6.000	6.216	5.986	1510715.0	9293977.41	18	5.84800	4.586575	3.366667
2023-07-19	6.220	0.2902	6.200	6.244	6.152	1675378.0	10417698.25	19	5.86740	4.602525	0.290229
2023-07-20	6.310	1.4469	6.240	6.420	6.160	772694.0	4864979.74	20	5.88796	4.618945	1.446945
2023-07-21	6.210	-1.5848	6.310	6.322	6.210	526717.0	3303527.97	21	5.90868	4.634895	-1.584786

1630 rows × 11 columns

Στη συνέχεια δημιουργούμε ένα γράφημα με την ημερήσια εμπορευσιμότητα των μετοχών που στηρίζεται στους Όγκους των ημερήσιων συναλλαγών.

```
In [136]: df1['Όγκος'].plot(label = 'ETE', figsize = (15,7))
          df2['Όγκος'].plot(label = 'OTE')
          df3['Όγκος'].plot(label = 'ΔΕΗ')
          df4['Όγκος'].plot(label = 'ΜΥΤΙΛΗΝΑΙΟΣ')
          plt.title('Εμπορευσιμότητα')
          plt.legend()
```

```
Out[136]: <matplotlib.legend.Legend at 0x1990efa9ee0>
```



Παρουσιάζουμε τα δεδομένα που περιέχει το dataframe data που δημιουργήσαμε παραπάνω (βλέπε γράφημα διασποράς).

```
In [137]: data
```

```
Out[137]:
```

Ημερομηνια	ETE	OTE	ΔΕΗ	ΜΥΤΙΛΗΝΑΙΟΣ
2017-01-02	2.530	9.00	2.91	6.135
2017-01-03	2.690	8.88	3.00	6.315
2017-01-04	2.620	9.00	2.95	6.315
2017-01-05	2.550	8.98	2.98	6.295
2017-01-09	2.530	9.00	2.95	6.295
...	...	...	...	...
2023-07-17	6.000	14.44	10.25	33.300
2023-07-18	6.202	14.16	10.40	34.540
2023-07-19	6.220	14.60	10.75	35.680
2023-07-20	6.310	14.66	10.79	36.300
2023-07-21	6.210	14.46	11.12	35.820

1630 rows × 4 columns

Τέλος δημιουργούμε ένα διαδραστικό γράφημα για την περίπτωση της μετοχής της ΕΤΕ και για το χρονικό διάστημα μετά από 1-11-2022. Το γράφημα είναι τύπου candlestick όπου σε κάθε μια ημέρα παρουσιάζεται η τιμή ανοίγματος, η τιμή κλεισίματος, το υψηλό και χαμηλό ημέρας.

```
In [138]: import plotly.graph_objects as go
```

```
In [139]: df11 = df11.loc[df11['Ημερομηνια'] > '2022-11-01']
          df11.reset_index(drop=True)

          fig = go.Figure(data=[go.Candlestick(x=df11['Ημερομηνια'],
                          open=df11['Ανοιγμα'], high=df11['Υψηλό'],
                          low=df11['Χαμηλό'], close=df11['Κλείσιμο'])
                               ])

          fig.update_layout(xaxis_rangeslider_visible=False, title='Γράφημα Candlestick ETE')
          fig.show()
```

Όπως βλέπουμε στο παραπάνω γράφημα με πράσινο χρώμα είναι οι ημέρες όπου η τιμή κλεισίματος ήταν μεγαλύτερη από την τιμή ανοίγματος της μετοχής ενώ με κόκκινο χρώμα αποτυπώνεται η αντίθετη περίπτωση.

Μάλιστα το γράφημα ενσωματώνει την πληροφορία αυτή όπου όταν με τον δείκτη του mouse πηγαίνουμε σε μια συγκεκριμένη ημέρα μας εμφανίζει αναλυτικά όλες αυτές τις πληροφορίες όπως φαίνεται και στο παρακάτω σχήμα.

Συμπεράσματα

Με τον όρο "Μεγάλα Δεδομένα" (Big Data) αναφερόμαστε σε οποιοδήποτε σύνολο δεδομένων που είναι τόσο μεγάλο ή τόσο πολύπλοκο που οι συμβατικές εφαρμογές δεν επαρκούν για την επεξεργασία τους. Η ενσωμάτωση αισθητήρων, η αύξηση της συνδεσιμότητα των συσκευών που καταγράφουν ήχους, εικόνες ή βίντεο, η επικοινωνία των μηχανημάτων μεταξύ τους, τα έξυπνα κινητά τηλέφωνα που συλλέγουν γεωχωρικά δεδομένα, μηχανές σε γραμμές παραγωγής βιομηχανικών συστημάτων, ανταλλάσσουν σημαντικές πληροφορίες κατά την επεξεργασία των δραστηριοτήτων τους, δημιουργούν αυτό το πλαίσιο που ονομάζουμε Big Data.

Ο όρος αναφέρεται επίσης στα εργαλεία και τις τεχνολογίες που χρησιμοποιούνται για τον χειρισμό των "μεγάλων δεδομένων". Αυτό το νέο περιβάλλον που δημιουργείται αυξάνει την ανάγκη για τεχνολογίες Big Data, απαιτεί νέα εργαλεία για το χειρισμό τους και για την ανάλυση αυτού του τεράστιου όγκου δεδομένων ώστε να είναι εφικτή η λήψη πολύτιμων αποφάσεων για το άτομο τον οργανισμό ή την επιχείρηση.

Τα μεγάλα δεδομένα έχουν τρία χαρακτηριστικά V, δηλαδή τον μεγάλο όγκο (Volume), την μεγάλη ποικιλία (Variety) και την μεγάλη ταχύτητα που παράγονται (Velocity). Πιο πρόσφατα, διάφορα άλλα V έχουν προστεθεί σε διάφορες περιγραφές των μεγάλων δεδομένων, όπως η αληθοφάνεια (Veracity), η αξία (Value) και η μεταβλητότητα (Variability).

Τα μεγάλα δεδομένα είναι ένας συνδυασμός δομημένων, ημιδομημένων και αδόμητων δεδομένων τα οποία πρέπει να οργανωθούν για να μετατραπούν τα αμέτρητα bits και bytes σε αξιοποιήσιμες πληροφορίες. Η καθαρή ποσότητα των δεδομένων δεν θα ήταν χρήσιμη αν δεν είχαμε τρόπους να τα αξιοποιήσουμε. Για αυτό το λόγο έχει ραγδαία αναπτυχθεί τα τελευταία χρόνια και προβλέπεται ακόμη μεγαλύτερη ανάπτυξη στο μέλλον ο τομέας της Ανάλυσης των Δεδομένών.

Η ανάλυση μεγάλων δεδομένων είναι η διαδικασία συλλογής, εξέτασης και ανάλυσης μεγάλου όγκου δεδομένων για την ανακάλυψη τάσεων της αγοράς, γνώσεων και προτύπων που μπορούν να βοηθήσουν τις επιχειρήσεις ή τους οργανισμούς να λάβουν καλύτερες αποφάσεις. Οι πληροφορίες αυτές είναι διαθέσιμες γρήγορα και αποτελεσματικά.

Για παράδειγμα, η ανάλυση μεγάλων δεδομένων είναι αναπόσπαστο στοιχείο του σύγχρονου κλάδου της υγειονομικής περίθαλψης. Όπως μπορείτε να φανταστείτε, πρέπει να διαχειριστείτε χιλιάδες αρχεία ασθενών, ασφαλιστικά προγράμματα, συνταγές και πληροφορίες εμβολίων. Πρόκειται για τεράστιες ποσότητες δομημένων και μη δομημένων δεδομένων, τα οποία μπορούν να προσφέρουν σημαντικές πληροφορίες όταν εφαρμόζονται αναλύσεις. Η ανάλυση μεγάλων δεδομένων το κάνει αυτό γρήγορα και αποτελεσματικά, ώστε οι πάροχοι υγειονομικής περίθαλψης να μπορούν να χρησιμοποιούν τις πληροφορίες για να κάνουν τεκμηριωμένες, σωτήριες για τη ζωή διαγνώσεις.

Επίσης η ανάλυση δεδομένων, έχει σημαντικό πεδίο εφαρμογής και στον χρηματοπιστωτικό τομέα. Τα δεδομένα αυτά διαδραματίζουν σημαντικό ρόλο στη χρηματοπιστωτική αγορά, όπως η πρόβλεψη της μεταβλητότητας της αγοράς, η πρόβλεψη των αποδόσεων της αγοράς, η αποτίμηση των θέσεων στην αγορά, ο εντοπισμός του υπερβολικού όγκου συναλλαγών, η ανάλυση του κινδύνου της αγοράς, η κίνηση των μετοχών, η τιμολόγηση των δικαιωμάτων προαίρεσης, η μεταβλητότητα, οι αλγοριθμικές συναλλαγές κ.λπ.

Άλλα πεδία εφαρμογής της ανάλυσης δεδομένων είναι στις συγκοινωνίες, στο εμπόριο, στην εκπαίδευση, στα διαδικτυακά μέσα κοινωνικής δικτύωσης, στα ηλεκτρονικά καταστήματα, στο δημόσιο τομέα κ.λ.π.

Υπάρχουν τέσσερις κύριοι τύποι ανάλυσης μεγάλων δεδομένων που υποστηρίζουν και ενημερώνουν διαφορετικές αποφάσεις. Η περιγραφική ανάλυση η οποία αναφέρεται σε δεδομένα

που μπορούν εύκολα να διαβαστούν και να ερμηνευτούν. Τα δεδομένα αυτά βοηθούν στη δημιουργία αναφορών και στην οπτικοποίηση πληροφοριών που μπορούν να περιγράφουν λεπτομερώς τα κέρδη και τις πωλήσεις της εταιρείας. Η διαγνωστική ανάλυση που βοηθά τις εταιρείες να κατανοήσουν γιατί προέκυψε ένα πρόβλημα. Οι τεχνολογίες και τα εργαλεία μεγάλων δεδομένων επιτρέπουν στους χρήστες να εξορύξουν και να ανακτήσουν δεδομένα που βοηθούν στην ανάλυση ενός προβλήματος και στην αποτροπή του στο μέλλον. Η προγνωστική ανάλυση με τη βοήθεια της οποίας εξετάζονται δεδομένα του παρελθόντος και του παρόντος για να κάνει προβλέψεις. Η προδιαγραφική ανάλυση που παρέχει λύσεις σε ένα πρόβλημα, βασιζόμενη στην Τεχνητή Νοημοσύνη και τη μηχανική μάθηση για τη συλλογή δεδομένων και τη χρήση τους για τη διαχείριση κινδύνων.

Υπάρχουν πολλά εργαλεία που βοηθούν τους επιστήμονες δεδομένων να επεξεργάζονται και να αναλύουν τα δεδομένα για την επίτευξη των σκοπών τους. Έχουν εμφανιστεί πολλές νέες γλώσσες, πλαίσια και τεχνολογίες αποθήκευσης δεδομένων που υποστηρίζουν το χειρισμό μεγάλων δεδομένων. Τέτοια είναι: MapReduce, Hadoop, Apache Hive, Apache Pig, Apache Spark, MADlib, Amazon Elastic Compute Cloud (EC2), R, Scala, και η Python.

Η Python είναι μια διαδραστική γλώσσα προγραμματισμού γενικού σκοπού και υψηλού επιπέδου. Αποκτά κορυφαία προτίμηση μεταξύ των προγραμματιστών και των αναλυτών των μεγάλων δεδομένων λόγω του απλού συντακτικού της και του συστήματος διερμηνείας. Το ενδιαφέρον στοιχείο είναι ότι η Python έχει λιγότερες συντακτικές κατασκευές, γεγονός που την καθιστά ιδιαίτερα ευανάγνωστη. Περιέχει μια δεξαμενή ενσωματωμένων λειτουργικών δυνατοτήτων - βιβλιοθήκες, δομές δεδομένων και πλαίσια. Συν τοις άλλοις, ο καθένας (ανεξάρτητα από το υπόβαθρο κωδικοποίησης) μπορεί να μάθει και να χρησιμοποιήσει την Python χωρίς κόστος. Έτσι, από τους αρχάριους έως τους ειδικούς, λειτουργεί για όλους.

Από τα πιο βασικά της χαρακτηριστικά που την κάνουν δημοφιλή στον κόσμο της ανάλυσης δεδομένων είναι:

1. Η Python είναι μια εξαιρετικά ευέλικτη γλώσσα προγραμματισμού με ένα τεράστιο οικοσύστημα εργαλείων και βιβλιοθηκών που έχουν σχεδιαστεί ρητά για την ανάλυση και τον χειρισμό δεδομένων. Δημοφιλείς βιβλιοθήκες όπως η NumPy, η pandas, η matplotlib και η scikit-learn παρέχουν ισχυρές δυνατότητες χειρισμού δεδομένων, έρευνας και μηχανικής μάθησης, καθιστώντας την Python ιδανική επιλογή για μεγάλα δεδομένα.
2. Η Python διαθέτει μια μεγάλη κοινότητα επιστημόνων δεδομένων, μηχανικών και προγραμματιστών που συμβάλλουν στην ανάπτυξη βιβλιοθηκών, εργαλείων και πλαισίων για την ανάλυση μεγάλων δεδομένων. Αυτό σημαίνει ότι οι χρήστες της Python έχουν πρόσβαση σε πληθώρα πόρων, σεμιναρίων και υποστήριξης από την κοινότητα, διευκολύνοντας την εκμάθηση και την εργασία με την Python για την ανάλυση μεγάλων δεδομένων.
3. Η Python προσφέρει ισχυρές βιβλιοθήκες οπτικοποίησης δεδομένων, όπως οι Matplotlib, Seaborn και Plotly, οι οποίες παρέχουν ολοκληρωμένες δυνατότητες σχεδίασης και οπτικοποίησης. Η αποτελεσματική οπτικοποίηση δεδομένων είναι ζωτικής σημασίας για την κατανόηση μεγάλων συνόλων δεδομένων, τον εντοπισμό μοτίβων και την απόκτηση πληροφοριών, καθιστώντας την Python ένα πολύτιμο εργαλείο για την ανάλυση μεγάλων δεδομένων.
4. Η Python μπορεί εύκολα να ενσωματωθεί με δημοφιλείς τεχνολογίες μεγάλων δεδομένων, όπως το Spark, το Hadoop και το Hive, επιτρέποντας την απρόσκοπτη ενσωμάτωση σε υπάρχουσες ροές εργασίας μεγάλων δεδομένων. Αυτό καθιστά την Python βολική για την ανάλυση μεγάλων δεδομένων, καθώς μπορεί εύκολα να ενσωματωθεί σε υπάρχουσες σωληνώσεις και ροές εργασίας επεξεργασίας δεδομένων.

5. Το συντακτικό της Python είναι εύκολο στην εκμάθηση και την ανάγνωση, γεγονός που την καθιστά δημοφιλή επιλογή για ταχεία πρωτοτυποποίηση και επαναληπτική ανάπτυξη. Αυτό είναι ιδιαίτερα πολύτιμο στην ανάλυση μεγάλων δεδομένων, όπου η δυνατότητα γρήγορης πρωτοτυποποίησης και δοκιμής αλγορίθμων ανάλυσης δεδομένων είναι απαραίτητη για τον εντοπισμό αποτελεσματικών στρατηγικών και συμπερασμάτων από μεγάλα σύνολα δεδομένων.

Για τους παραπάνω λόγους επιλέξαμε να χρησιμοποιήσουμε την Python για την μελέτη περίπτωσης που αναπτύξαμε στο τελευταίο κεφάλαιο. Η ανάλυση των δεδομένων μας έγινε σε περιβάλλον Anaconda και Jupyter Notebook με τη βοήθεια των βιβλιοθηκών της Python όπως η Pandas και η Matplotlib. Η ανάλυση των δεδομένων μας έγινε σε δεδομένα τεσσάρων μετοχών του Χρηματιστηρίου Αξιών Αθηνών: της Εθνικής Τράπεζας (ΕΤΕ), του Οργανισμού Τηλεπικοινωνιών Ελλάδος (ΟΤΕ), της Δημόσιας Επιχείρησης Ηλεκτρισμού (ΔΕΗ) και της μετοχής του Ομίλου Μυτιληναίου (ΜΥΤΙΛ). Στόχος μας ήταν η δημιουργία μιας στρατηγικής δημιουργίας ενός χαρτοφυλακίου με τη σύνθεση των παραπάνω μετοχών και η μελέτη των αποδόσεών του. Η χρονική περίοδος που εστιάσαμε περισσότερο ήταν από τις αρχές του 2017 έως και τα τέλη του 1ου εξαμήνου του 2023. Εντοπίσαμε, μελετήσαμε και περιγράψαμε τα μοτίβα και τη συμπεριφορά των μετοχών σε κρίσιμες χρονικές στιγμές όπως η πανδημία και ο πόλεμος Ρωσίας – Ουκρανίας. Δημιουργήσαμε ένα πλήθος συναρτήσεων και dataframes που σκοπό είχαν την καλύτερη και όσο το δυνατό πληρέστερη ανάλυση των δεδομένων μας. Θα πρέπει να επισημάνουμε ότι η συγκεκριμένη ανάλυση των δεδομένων μας μπορεί να τροποποιηθεί τόσο στο πλήθος και τους τίτλους των μετοχών, όσο και στις χρονικές περιόδους που μελετήσαμε.

Βιβλιογραφία

[1] David Reinsel, John Gantz, John Rydning, The Digitization of the World From Edge to Core, IDC White Paper US44413318, International Data Corporation (IDC), Framingham, Massachusetts, November 2018 https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf

[2] Nathan Marz, James Warren, Big Data, Principles and best practices of scalable realtime data systems, Manning Publications Co, 2015

[3] Ericsson Mobility Report, June 2023, https://www.ericsson.com/49dd9d/assets/local/reports-papers/mobility-report/documents/2023/ericsson-mobility-report-june-2023.pdf

[4] EMC Education Services, Data Science & Big Data Analytics. Indianapolis : John Wiley & Sons, 2015

[5] Anuj Mediratta, Big Data: Terms, Definitions and Applications, EMC, 2015

[6] Chen, M., Mao, S., Liu, Y., Big data: A survey. Mobile Networks and Applications 19 (2), pp 171-209, 2014

[7] Venkat Ankam, Big Data Analytics, Packt Publishing, 2016

[8] Krishnan, Krish. Data Warehousing in the Age of Big Data, San Francisco: Morgan Kaufmann Publishers, 2013

[9] Saurabh Kamble, Heman Pawar, An Analytical Study On Big Data And Hadoop, International Journal of Research Publication and Reviews, s, Vol 3, no 6, pp 3884-3885, 2022

[10] T. Nguyen, "A Framework for Five Big V's of Big Data and Organizational Culture in Firms", 2018 IEEE International Conference on Big Data (Big Data), 2018

[11] M. Ali-ud-din Khan, Muhammad Fahim Uddin, Navarun Gupta, Seven V's of Big Data Understanding Big Data to extract Value, Proceedings of 2014 Zone 1 Conference of the American Society for Engineering Education (ASEE Zone 1), 2014

[12] Manyika, James and Chui, Michael. Big data: The next frontier for innovation, competition, and productivity. s.l. : McKinsey Global Institute, 2011.

[13] Michelle Burke, Amelia Parnell, Alexis Wesaw, Kevin Kruger, Predictive Analysis of Student Data, National Association of Student Personnel Administrators (NASPA), 2017

[14] Dutta Pramanik, Pijush & Pal, Saurabh & Mukherjee, Moutan, Healthcare Big Data: A Comprehensive Overview. 10.4018/978-1-5225-7071-4.ch004., 2018

[15] Wang, Y., Kung, L. ,Byrd, T.A., "Big data analytics: understanding its capabilities and potential benefits for healthcare organizations", Technological Forecasting and Social Change, Vol. 126, pp. 3-13, 2018

[16] Yongfang Nie, Qingguo Nie, Application of Big Data Analysis Technology in Financial Investment Risk Management, Proceedings of the 2022 International Conference on Bigdata Blockchain and Economy Management (ICBBEM 2022), 2022

[17] Stephen Kaisler, Frank Armour, J. Alberto Espinosa, William Money, Big data: Issues and challenges moving forward, 46th Hawaii International Conference on System Sciences (HICSS), Hawaii. pp. 995-1004, 2014

[18] Li, H. and Lu, X. (2014) Challenges and Trends of Big Data Analytics, 9th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), pp 566-567, Guangzhou, 2014

[19] Benjamins, V. Richard. (2014). Big Data: from Hype to Reality?, Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14). ACM, New York, NY, USA, pp. 2:1-2:2, 2014

[20] Boyd, D., Crawford, K., Six provocations for big data. In: A Decade in Internet Time: Symposium on the Dynamics of the Internet and Society, 2011

[21] Fisher, D., DeLine, R., Czerwinski, M., Drucker, S., Interactions with big data analytics. Interactions 19 (3), 50-59, 2012

[22] Manovich, L., Trending: the promises and the challenges of big social data, Gold, M. K. (Ed.), Debates in the Digital Humanities. The University of Minnesota Press, Minneapolis, MN, 2011

[23] Cuzzocrea, A., Song, I.-Y., Davis, K. C., Analytics over large-scale multidimensional data: the big data revolution, Proceedings of the ACM 14th International workshop on Data Warehousing and OLAP (DOLAP '11). ACM, New York, NY, USA, pp. 101-104, 2011

[24] Agneeswaran, V., Big-data - Theoretical, engineering and analytics perspective, Srinivasa, S., Bhatnagar, V. (Eds.), Big Data Analytics. Vol. 7678 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 8-15, 2012

[25] Khan, N., Yaqoob, I., Hashem, Big Data: Survey, Technologies, Opportunities, and Challenges. The Scientific World Journal, vol. 2014, Article ID 712826, 18 pages, 2014

[26] Jothimani, D., Bhadani, A.K., Shankar, R., Towards Understanding the Cynicism of Social Networking Sites: An Operations Management Perspective, Procedia - Social and Behavioral Sciences, vol. 189, pp 117–132, 2015

[27] Wes McKinney, Python for Data Analysis, O'Reilly Media, Inc, 2017

[28] Frisk, J.E. and Bannister, F., "Improving the use of analytics and big data by changing the decision-making culture", Management Decision, Vol. 55 No. 10, pp. 2074-2088., 2017

[29] Bhadani, A., Jothimani, D., Big data: Challenges, opportunities and realities, Singh, M.K., & Kumar, D.G. (Eds.), Effective Big Data Management and Opportunities for Implementation (pp. 1-24), Pennsylvania, USA, IGI Global, 2016

[30] Gupta, M. and George, J.F., "Toward the development of a big data analytics capability", Information and Management, Vol. 53 No. 8, pp. 1049-1064., 2016

[31] Fabio Nelli, Python Data Analytics, With Pandas, NumPy, and Matplotlib Second Edition, Springer, 2018

[32] https://www.ibm.com/topics/mapreduce

[33] https://hadoop.apache.org/

[34] https://hive.apache.org/

[35] https://pig.apache.org/

[36] https://spark.apache.org/

[37] https://madlib.apache.org/

[38] https://www.r-project.org/about.html

[39] https://www.scala-lang.org/

[40] https://www.python.org/

[41] Gayathri Rajagopalan, A Python Data Analyst's Toolkit, Springer, 2021

[42] Bernd Klein, Data Analysis Numpy, Matplotlib and Pandas, http://www.python-course.eu ,2021

[43] Ossama Embarak, Data Analysis and Visualization Using Python - Analyze Data to Create Visualizations for BI Systems, Springer, 2018

[44] Wes McKinney and the Pandas Development Team, pandas: powerful Python data analysis toolkit Release 1.4.4, 2022 https://pandas.pydata.org/pandas-docs/version/1.4/pandas.pdf

[45] Czygan, M., Getting Started with Python Data Analysis. 1st edn. Packt Publishing, 2015

[46] Stefanie Molin, Hands-On Data Analysis with Pandas, Packt Publishing, 2019

[47] www.capital.gr