Πανεπιστήμιο Πατρών

Σχολή Οικονομικών Επιστημών και Διοίκησης Επιχειρήσεων

Τμήμα Διοικητικής Επιστήμης και Τεχνολογίας

Πτυχιακή Εργασία

Η γλώσσα R στην επιστήμη δεδομένων

της

Γιαλελή Αγγελική

Επιβλέπουσα Καθηγήτρια:

Ρήγκου Μαρία

Μέλη Επιτροπής:

κος Παπαδόπουλος Δημήτριος

κος Γαρμπής Αριστογιάννης

Πάτρα Νοέμβριος 2020

University of Patras

School of Economics & Business

Department of Management Science and Technology

Thesis

The R Language in Data Science

Gialeli Angeliki

Supervising Professor:

Rigou Maria

Committee Members:

Papadopoulos Dimitrios

Garmpis Aristogiannis

Patra November 2020

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Διοικητικής Επιστήμης και Τεχνολογίας του Πανεπιστημίου Πατρών, δεν υποδηλώνει απαραιτήτως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

Ευχαριστίες

Ολοκληρώνοντας την πτυχιακή μου εργασία θα ήθελα αρχικά να ευχαριστήσω την αξιότιμη κα Ρήγκου Μαρία, Επίκουρο Καθηγήτρια του τμήματος, για την ευκαιρία που μου έδωσε να ασχοληθώ με το πολύ ενδιαφέρον πεδίο της Επιστήμης Δεδομένων μέσα από την Γλώσσα Προγραμματισμού R. Την ευχαριστώ για τις επιστημονικές της γνώσεις, τις συμβουλές της και για όλη την καθοδήγηση που μου παρείχε κατά την διάρκεια της συγγραφής της παρούσας πτυχιακής εργασίας. Ευχαριστώ επίσης τους καθηγητές κ. Παπαδόπουλο Δημήτριο και κ. Γαρμπή Αριστογιάννη, μέλη της τριμελούς Εξεταστικής Επιτροπής, οι οποίοι ευγενικά δέχθηκαν να αξιολογήσουν την πτυχιακή μου εργασία.

Ένα μεγάλο ευχαριστώ στους δικούς μου ανθρώπους, στον πατέρα μου Ανδρέα, στον σύζυγό μου Κώστα και στα δυο μας παιδιά, που με αγαπάνε, με ενθαρρύνουν και στηρίζουν την κάθε μου επιλογή. Και φυσικά ένα μεγάλο συγνώμη για τον χρόνο που τους στέρησα.

Τέλος θα ήθελα να ευχαριστήσω τον συμφοιτητή μου Παπανικολάου Δημήτρη, για τις γνώσεις που ανταλλάξαμε και για τον δημιουργικό χρόνο που μοιραστήκαμε κατά την διάρκεια των σπουδών μας.

Γιαλελή Αγγελική, Πάτρα 2020

Στα κοριτσάκια μου

Ευτυχία και Βασιλική

Πίνακας Περιεχομένων 6

Κατάλογος σχημάτων και εικόνων……..…9

Περίληψη……..……10

Abstract……..……11

Εισαγωγή……...…12

ΚΕΦΑΛΑΙΟ 1ο Επιστήμη και Επιστήμονες Δεδομένων……14

1.1 Η επιστήμη των Δεδομένων……...14

1.2 Επιστήμονας Δεδομένων……17

1.3 Ιστορική Αναδρομή του Επιστήμονα Δεδομένων……..…20

1.3.1 Ο Ρόλος του Επιστήμονα Δεδομένων……...…21

1.3.2 Η Καριέρα του Επιστήμονα Δεδομένων……25

1.3.3 Η εκπαίδευση του Επιστήμονα Δεδομένων……..……..26

1.3.4 Οι Δεξιότητες του Επιστήμονα Δεδομένων…….……27

ΚΕΦΑΛΑΙΟ 2ο Βασικές Τεχνολογίες της Επιστήμης Δεδομένων…….28

2.1 Η σημασία της μάθησης για τον άνθρωπο…….....28

2.2 Μηχανική Μάθηση……..…..…29

2.2.1 Είδη Μηχανικής Μάθησης…….…..30

2.3 Βασικές κατηγορίες αλγορίθμων……..31

2.3.1 Ο αλγόριθμος ID3……...….31

2.3.2 Δέντρα Απόφασης……...…...32

2.3.3 Τυχαία Δάση……..…33

2.3.4 Μάθηση κατά Bayes…….34

2.3.5 Παλινδρόμηση……...34

2.3.6 Ταξινόμηση……35

2.3.7 Ομαδοποίηση……36

2.3.8 Αλγόριθμος κ-Πλησιέστερου Γείτονα KNN……36

2.3.9 Αλγόριθμος LDA……37

2.4 Γλώσσες Προγραμματισμού και Επιστήμη Δεδομένων……..38

2.4.1 Η γλώσσα προγραμματισμού Python……40

2.4.2 Η γλώσσα προγραμματισμού R……40

2.5 Τεχνικές Αποθήκευσης δεδομένων σε cloud……...41

2.6 Τεχνικές Οπτικοποίησης δεδομένων ……...43

2.6.1 Πλεονεκτήματα και Μειονεκτήματα Οπτικοποίησης.....**...**43

ΚΕΦΑΛΑΙΟ 3ο Εισαγωγή στην Γλώσσα Προγραμματισμού R……45

3.1 Εισαγωγή……..…45

3.2 Το περιβάλλον εργασίας της R και του RStudio…….45

3.3 Γενική σύνταξη R…….47

3.3.1. Μεταβλητές……...48

3.3.1.1 Αντικείμενα και Κλάσεις……...49

3.3.2 Διανύσματα……50

3.3.2.1 Ορισμός διανύσματος……...51

3.3.2.2. Μετατροπή διανύσματος σε πίνακα……..53

3.3.2.3 Μητρώα…….....54

3.3.2.4 Λίστες……..54

3.3.3 Συναρτήσεις……..…...55

3.3.4 Πακέτα – Βιβλιοθήκες……..56

ΚΕΦΑΛΑΙΟ 4ο Μεθοδολογία Έρευνας με το πρόγραμμα R…….57

Case Study I: European Protein Consumption, εφαρμογή με clustering……..57

Case Study II: Social Network Clustering Analysis, εφαρμογή με clustering…...68

Case Study III: IRIS Flower Data Set εφαρμογή με classification……..81

Παράρτημα Κώδικα R……..102

Βιβλιογραφία……...107

Πνευματικά Δικαιώματα ……..115

Κατάλογος σχημάτων και εικόνων

Σχήμα 1.1 Το διάγραμμα Venn του Drew Conway

Σχήμα 1.2 Διάγραμμα ροής αδόμητων δεδομένων

Σχήμα 1.3 Τομείς εφαρμογής της επιστήμης δεδομένων πηγή

Σχήμα 1.4 Ο ορισμός ενός επιστήμονα δεδομένων

Σχήμα 1.5 Εννοιολογικό πρότυπο για το προφίλ Data Scientist – βάση γνώσεων

Σχήμα 1.6 Εννοιολογικό πρότυπο για το προφίλ Data Scientist – σύνολο δεξιοτήτων

Σχήμα 1.7 Οι δεξιότητες του Επιστήμονα Δεδομένων

Εικόνα 2.1 Δέντρα Απόφασης

Εικόνα 2.2 Δέντρα Απόφασης

Εικόνα 2.3 Αλγόριθμος Τυχαίου Δάσους

Εικόνα 2.4 Ταξινόμηση

Εικόνα 2.5 Clustering

Εικόνα 2.6 Αλγόριθμος LDA

Εικόνα 2.7 Δημοφιλείς γλώσσες προγραμματισμού

Εικόνα 3.1 Το περιβάλλον εργασίας R

Εικόνα 3.2 Παράδειγμα εφαρμογής εντολών σε R

Εικόνα 3.3 Προκαθορισμένες Συναρτήσεις R

Περίληψη

Η αλματώδης ανάπτυξη των πληροφοριακών συστημάτων, και η απεριόριστη χρήση ασύρματων και ενσύρματων δικτύων τα τελευταία χρόνια, έχουν ως συνέπεια την δημιουργία μεγάλων όγκων δεδομένων σε καθημερινή βάση. Τα δεδομένα συναντώνται με διάφορες μορφές, εικόνα, κείμενο κλπ, ή συχνά περιέχουν θόρυβο, και πολλές φορές είναι δύσκολο να αξιολογηθούν. Σε έναν κόσμο, όπου κάθε είδος εργασίας περιλαμβάνει την ενασχόληση με δεδομένα οποιουδήποτε είδους, σχεδόν όλοι χρειάζονται έναν επιστήμονα που να εξειδικεύεται στην εξαγωγή και ανάλυση των δεδομένων με στόχο την εύρεση ουσιαστικών λύσεων. Προβλέπεται άλλωστε ότι θα υπάρξει ζήτηση για εκατοντάδες χιλιάδες θέσεις εργασίας στο μέλλον, για ανθρώπους με δεξιότητες αναλυτικής και εμπειρία στη διαχείριση δεδομένων.

Και επειδή η εξέλιξη δεν σταματά ποτέ, νέες δεξιότητες απαιτούνται για το σχεδιασμό εκπαιδευτικών εργαλείων που θα προσφέρουν και θα αλλάξουν πολλά. Η Επιστήμη των Δεδομένων (Data Science) και η Μηχανική Μάθηση (Machine Learning), ως νέες αναδυόμενες τεχνολογίες που περιλαμβάνουν σύγχρονες μεθόδους, έχουν επιδείξει εντυπωσιακά αποτελέσματα σε πολλούς επιστημονικούς και επιχειρηματικούς κλάδους αλλά και στην καθημερινή ζωή γενικότερα. Είναι κατά βάση διεπιστημονικά πεδία με κύριο αντικείμενο τη διαχείριση, ανάλυση, επεξεργασία και εξαγωγή γνώσης από δεδομένα σε δομημένη ή σε αδόμητη μορφή. Η R από την άλλη, είναι μια γλώσσα προγραμματισμού ανοιχτού κώδικα που χρησιμοποιείται για τη στατιστική ανάλυση δεδομένων, η οποία έχει γίνει πολύ δημοφιλής τα τελευταία χρόνια. Ταυτόχρονα το λογισμικό της R διαθέτει έναν μεγάλο αριθμό γραφικών παραστάσεων για την οπτικοποίηση και παρατήρηση των δεδομένων.

Λέξεις κλειδιά : γλώσσα προγραμματισμού R, μηχανική μάθηση, επιστήμη δεδομένων, επιστήμονας δεδομένων

Abstract

The swift development of information systems and the unlimited access of wireless or ground internet, the last few years have as an outcome the creation of immense quantities of data on a daily basis. The data is recognized in various forms of image, text etc or it frequently contains noise and at times it is difficult to evaluate. In a world where any kind of task includes data of any sort, almost everyone needs a scientist who specializes in the recovery and the analysis of data with the aim of finding essential solutions. Evidently it is predicted that a demand for hundreds of employment positions will incur in the future for people with skills in analysis and experience in managing data.

Due to the never – ending evolution, new skills are in demand for the development of educational tools which will offer and will bring about a big change. Data Science and Machine Learning as new, emerging technologies which include modern techniques, not only have shown impressive results in numerous scientific and business fields, but in daily life as well. They are basically interdisciplinary fields as a primary objective the management, analysis, process and receiving knowledge from data in a structured or unstructured form. However, R is a programming language of an open code which is used for the analysis of data in statistics and has become very popular in recent years. At the same time the R software carries a vast number of graphic designs for the visualization and observation of data.

Key words: R programming language, machine learning, data science, data scientist

Εισαγωγή

Η παρούσα πτυχιακή εργασία πραγματεύεται τη μελέτη και την ανάλυση των τεχνικών και των αλγορίθμων που χρησιμοποιούνται στον κλάδο της Επιστήμης Δεδομένων (Data Science) και της Μηχανικής Μάθησης (Machine Learning). Στην συνέχεια θα πραγματοποιήσουμε εφαρμογή των αλγορίθμων σε γνωστά σύνολα δεδομένων, με χρήση της Γλώσσας Προγραμματισμού R με απώτερο στόχο τη πλήρη ανάλυσή τους.

Η εργασία είναι δομημένη σε τέσσερα κεφάλαια όπου στο πρώτο κεφάλαιο εισαγωγικές έννοιες και ορισμοί αποδίδονται τόσο για τον επιστήμονα όσο και για την επιστήμη των δεδομένων. Επιπλέον προσδιορίζεται ο ρόλος και σκιαγραφείται το προφίλ ενός επιστήμονα δεδομένων, μέσα από βιβλιογραφική ανασκόπηση και από τις δεξιότητες που αναζητούν οι επιχειρήσεις σε αυτόν.

Το δεύτερο κεφάλαιο της εργασίας, είναι αφιερωμένο στις βασικές τεχνολογίες που σχετίζονται με την επιστήμη των δεδομένων. Δίνεται περιγραφή της μηχανικής μάθησης και αναφέρονται τα είδη στα οποία αυτή διακρίνεται. Οι βασικές κατηγορίες αλγορίθμων και οι πιο δημοφιλείς γλώσσες προγραμματισμού που συναντάμε στην μηχανική μάθηση, αναφέρονται στο παρόν κεφάλαιο. Τέλος, γίνεται μία βασική εισαγωγή στις γλώσσες προγραμματισμού Python και R, ενώ περιγράφονται τεχνικές οπτικοποίησης και αποθήκευσης δεδομένων σε cloud.

Το τρίτο κεφάλαιο της εργασίας περιλαμβάνει την γλώσσα προγραμματισμού R. Το περιβάλλον εργασίας και η γενική σύνταξη της R περιγράφονται εδώ, ενώ δίνονται παραδείγματα για τον ορισμό των μεταβλητών, σταθερών, αντικειμένων κλπ.

Το τέταρτο και τελευταίο κεφάλαιο αποτελεί την εμπειρική μελέτη της παρούσας εργασίας και περιλαμβάνει τρεις εφαρμογές μελετών περίπτωσης, με το πρόγραμμα R. Στις δύο πρώτες μελέτες γίνεται εφαρμογή του αλγορίθμου συσταδοποίησης (clustering), ενώ στην τρίτη μελέτη ασχολούμαστε με το γνωστό

πρόβλημα λουλουδιών, Iris του Fisher, στο οποίο γίνεται εφαρμογή μερικών από τους πιο γνωστούς αλγόριθμους ταξινόμησης (classification) που χρησημοποιεί η μηχανική μάθηση. Κάθε μελέτη κλείνει, με το παράρτημά της σε κώδικα R.

Η πτυχιακή εργασία ολοκληρώνεται με την βιβλιογραφία και τις αναφορές που χρησιμοποιήθηκαν για την συγγραφή της.

ΚΕΦΑΛΑΙΟ 1ο Επιστήμη και Επιστήμονες Δεδομένων

1.1 Η επιστήμη των Δεδομένων

Ή Επιστήμη των Δεδομένων (Data Science), είναι ουσιαστικά μία καινούρια επιστήμη, που έχει ως αντικείμενο την εξαγωγή γνώσης και προβλέψεων από μεγάλους όγκους δεδομένων (Big Data). Χρησιμοποιείται όχι μόνο για να επιλύσει προβλήματα εμπορικού ή επιστημονικού ενδιαφέροντος, αλλά για να εντοπίσει την εμφάνιση ενός συγκεκριμένου γεγονότος στο μέλλον (πιθανότητα επανάληψης). Η περιοχή προέκυψε από το συνδυασμό σημαντικών εξελίξεων στην πληροφορική και από την συνεχή παραγωγή δεδομένων σε εικοσιτετράωρη βάση. Συγκεκριμένα, η πρόοδος που σημειώθηκε σε αλγορίθμους και τεχνικές μηχανικής μάθησης ή τεχνητής νοημοσύνης, και δεύτερον, οι ραγδαίες εξελίξεις στην περιοχή επεξεργασίας ετερογενών και συνεχώς μεταβαλλόμενων δεδομένων, οδήγησαν στην ανάπτυξή της, τις τελευταίες δεκαετίες.

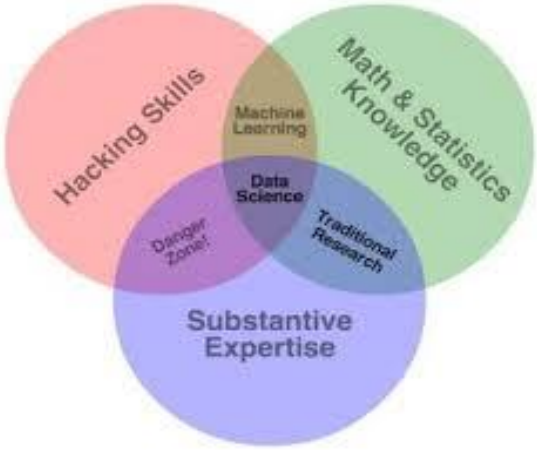
Ο Αμερικανός επιστήμονας δεδομένων Drew Conway (2013) στο διάγραμμά του, «Το διάγραμμα Venn του Drew Conway», υποστηρίζει ότι ο επαγγελματίας αυτός πρέπει να κατέχει γνώσεις σε τρεις σημαντικούς τομείς, όπως αναλύονται και παρουσιάζονται ακολούθως.

• Δεξιότητες πειρατείας (Hacking Skills): όλοι οι επιστήμονες δεδομένων πρέπει να έχουν γνώση αρκετών ψηφιακών δεξιοτήτων, όπως η εις βάθος γνώση διαφόρων λειτουργικών συστημάτων, γλώσσες προγραμματισμού, κωδικοποίηση κλπ.

• Γνώση στα στατιστικά και τα μαθηματικά (Math and Statistics Knowledge): είναι οι πιο συνηθισμένοι τομείς σπουδών και η γνώση αυτών των δύο κλάδων είναι απαραίτητη, για να επεξεργαστεί και να αναλύει μεγάλα δεδομένα ένας επαγγελματίας.

• Ουσιαστική εμπειρογνωμοσύνη (Substantive expertise): επειδή η επιστήμη είναι και ανακάλυψη αλλά και δημιουργία γνώσης, ένας επαγγελματίας πρέπει να έχει

τη δυνατότητα να κατανοεί και να χειρίζεται δεδομένα και πληροφορίες με την εφαρμογή των προαναφερθέντων κλάδων.



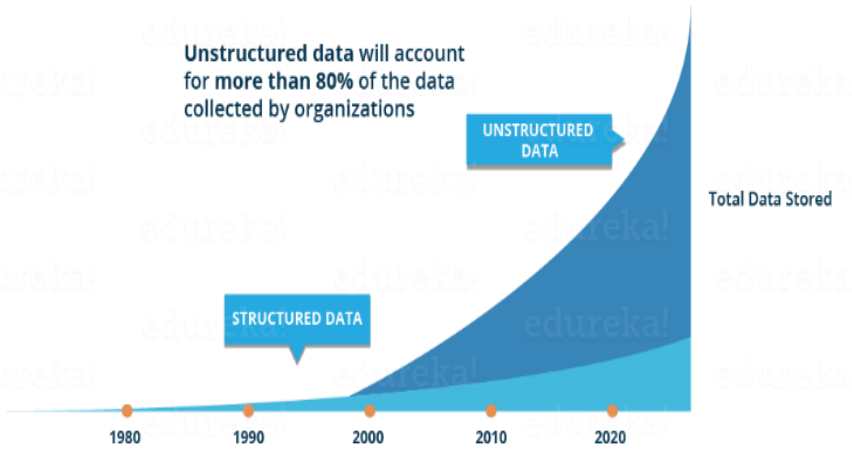
Σχήμα 1.1 Το διάγραμμα Venn του Drew Conway.

Αρχικά ο τίτλος που υιοθετήθηκε στη Δανία στα μέσα της δεκαετίας του 1960, από τον Peter Naur ήταν η επιστήμη της πληροφορικής να ασκείται με την ονομασία datalogy, ως έννοια της επιστήμης αξιολόγησης δεδομένων. Από την άλλη πλευρά, ο Tukey (1962) προέβλεψε την χρησιμότητα της στατιστικής ως μια εμπειρική επιστήμη για την μετάβαση από την ανάλυση δεδομένων στο συμπέρασμα. Ενώ μερικά χρόνια αργότερα, υποστήριξε ότι η διερευνητική ανάλυση δεδομένων και η ανάλυση των δεδομένων επιβεβαίωσης πρέπει να προχωρούν μαζί. Στα μέσα της δεκαετίας του '90, η επιστήμη δεδομένων άρχισε να θεωρείται μια νέα επιχειρηματική ευκαιρία όταν τα δεδομένα έγιναν το επίκεντρο του ενδιαφέροντος. Οι περισσότερες εταιρείες γνώριζαν ότι είχαν μεγάλο όγκο δεδομένων που δεν είχαν αναλυθεί σωστά (Berry, 1994). Αναγνώρισαν την επιστήμη δεδομένων ως το μέσο που τους δίνει τη δυνατότητα να δημιουργούν ηγέτες για τις επιχειρήσεις και να αποκτούν έξυπνη γνώση ώστε να δημιουργούν προϊόντα (δεδομένων) με επιχειρηματικό όφελος (Granville, 2013).

Σήμερα ο κόσμος μας αποτελείται από εφαρμογές ιστού που βασίζονται σε δεδομένα. Το Διαδίκτυο περισσότερο, βασίζεται σε βάσεις και υπηρεσίες δεδομένων και σε συνδυασμό με την ψηφιακή τεχνολογία, έχει δημιουργήσει μία

τεράστια αγορά γνώσεων/πληροφοριών. Στην πιο εξειδικευμένη μορφή του, επιτρέπει την συγκέντρωση γνώσης , με το μεγάλο ερώτημα να γεννάται: "*πώς θα ελέγξουμε την γνώση αυτή"*;

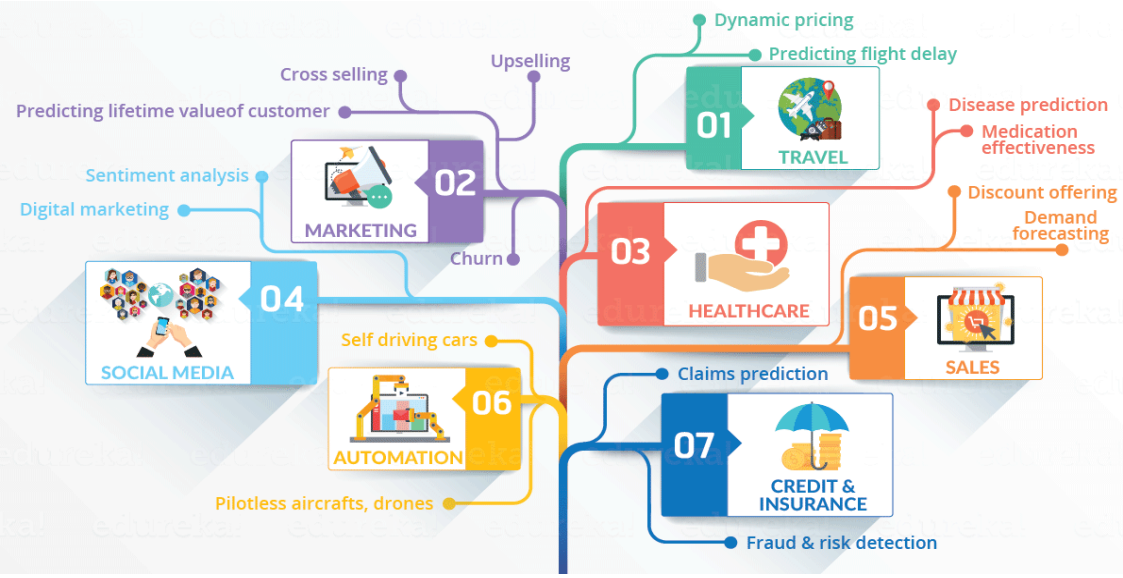
Παραδοσιακά, τα δεδομένα που είχαμε έως τώρα, ήταν κυρίως δομημένα και μικρού μεγέθους, τα οποία μπορούσαμε εύκολα να τα διαχειριστούμε χρησιμοποιώντας απλά εργαλεία. Στις μέρες μας τα περισσότερα δεδομένα είναι αδόμητα ή ημιδομημένα. Το παρακάτω γράφημα δείχνει την τάση στα δεδομένα από το 1980 έως το σήμερα όπου γίνεται ξεκάθαρο ότι περισσότερο από το 80% των δεδομένων είναι μη δομημένα.



Σχήμα 1.2 Διάγραμμα ροής αδόμητων δεδομένων (πηγή edureka.co)

Αυτό το κενό έρχεται να καλύψει η επιστήμη των δεδομένων σκοπός της οποίας είναι να δημιουργεί μοντέλα όχι μόνο για την περιγραφή αλλά και για την πρόβλεψη γεγονότων, και να τα παρουσιάζει με τρόπο κατανοητό στους ανθρώπους. Η πρόβλεψη μπορεί να αφορά π.χ. στην εύρεση ομάδων αντικειμένων με παρόμοια χαρακτηριστικά και γίνεται με την βοήθεια κάποιων εργαλείων. Αυτός όμως δεν είναι ο μοναδικός λόγος για τον οποίο η επιστήμη των δεδομένων έχει γίνει τόσο δημοφιλής. Συχνά εφαρμόζεται στις επιχειρήσεις και συγκεκριμένα στο μάρκετινγ (πρόβλεψη της αξίας ζωής του πελάτη, παράλληλη πώληση), ή στις πωλήσεις (εκπτωτικές προσφορές, πρόβλεψη ζήτησης). Επίσης εφαρμόζεται σε διάφορους άλλους τομείς όπως στην ιατρική

(πρόγνωση ασθενειών, αποτελεσματικότητα φαρμάκων), στις ασφάλειες (ανίχνευση κινδύνου, πρόβλεψη απαιτήσεων), στα ταξίδια (δυναμική τιμολόγηση, πρόβλεψη καθυστέρησης πτήσεων), στα κοινωνικά δίκτυα (ψηφιακό μάρκετινγκ, εξόρυξη γνώσης), στις περιβαλλοντικές επιστήμες και αλλού.



Σχήμα 1.3 Τομείς εφαρμογής της επιστήμης δεδομένων πηγή: (edureka.co)

1.2 Επιστήμονας Δεδομένων

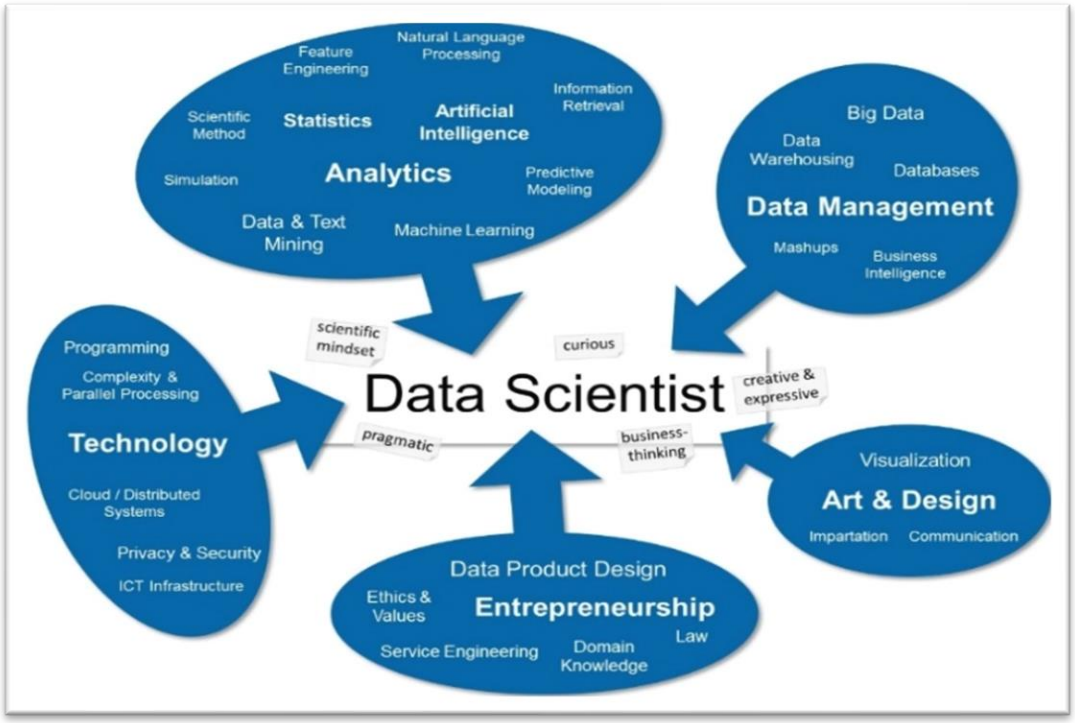
Τα τελευταία χρόνια, παρατηρούμε μία αυξανόμενη δημοτικότητα του όρου του Επιστήμονα Δεδομένων (Data Scientist), ο οποίος κερδίζει συνεχώς έδαφος έναντι άλλων, και βρίσκεται στο επίκεντρο πολλών επιχειρήσεων και οργανισμών. Ο ρόλος του δεν ήταν ποτέ τόσο σημαντικός, με τον σύγχρονο Τύπο να τον αναγνωρίζει ως ένα σπάνιο είδος (Harris and Eitel-Porter 2015), σχεδόν μυθικό, ικανό να σώσει τις επιχειρήσεις (Sicular 2012). Και αν ίσως στην Ελλάδα αργήσουν να βρουν τον δρόμο τους, οι επιστήμονες δεδομένων κατατάσσονται στην πρώτη θέση μεταξύ των πιο ελπιδοφόρων θέσεων εργασίας στις Ηνωμένες Πολιτείες για το 2019, σύμφωνα με την έκθεση του LinkedIn. Ενώ για την Ευρώπη ο αριθμός των ανθρώπων που εργάζονται σε δεδομένα,

πρόκειται να αυξηθεί σε 10,43 εκατομμύρια έως το 2020, σύμφωνα με την έκθεση της Ευρωπαϊκής Επιτροπής (2017).

Τι είναι όμως ο επιστήμονας δεδομένων; Τι δουλειά κάνει ακριβώς; Πώς μπορεί κάποιος να γίνει επιστήμονας δεδομένων; Πρέπει πρώτα να είναι επιστήμονας; Οι απαντήσεις θα μας βοηθήσουν να κατανοήσουμε τη βάση γνώσεων, τις ικανότητες και τις τεχνικές δεξιότητες που πρέπει να έχει ένας επιστήμονας δεδομένων, προκειμένου να περιγράψουμε το προφίλ του επαρκώς.

Ένας επιστήμονας δεδομένων είναι αυτός που ξέρει πώς να χειριστεί τα δομημένα και ημιδομημένα δεδομένα, πώς να εξαγάγει νόημα και μπορεί να «διηγηθεί» μια ιστορία γύρω από αυτά, ώστε να βοηθήσει την επιχείρηση να λάβει αποφάσεις. Έχει την ικανότητα να συλλέγει την κατάλληλη πληροφορία μέσα από άχρηστη ποσότητα δεδομένων που υπάρχει συσσωρευμένη στο διαδίκτυο και να εντοπίζει τάσεις. Το προς επίλυση πρόβλημα για αυτόν είναι ξεκάθαρο, και μπορεί να μεταφράζει με σαφήνεια τα τεχνικά ευρήματά του, αφού εξετάζει τα δεδομένα από πολλές οπτικές γωνίες. Πρέπει να είναι περίεργος και να δαπανά το 80% του χρόνου του για να ανακαλύψει και να κατανοήσει τις δυνατότητες των δεδομένων. (DalleMule & Davenport 2017). Όπως είπε κάποτε ο Albert Einstein, *“I have no special talent. I am only passionately curious.”*

Αυτό το μοναδικό σετ δεξιοτήτων συνδυάζει στατιστικές μεθόδους και απαιτεί τεχνογνωσία στον τομέα της τεχνολογίας λογισμικού, αλλά βασίζεται και σε μεγάλο βαθμό στην αναλυτική κριτική σκέψη. Ποιο είναι λοιπόν αυτό το μείγμα ικανοτήτων και δεξιοτήτων που πρέπει να έχει; Στο παρακάτω σχήμα, οι όροι που περιγράφουν τις ικανότητες ενός επιστήμονα, έχουν επιλεγεί λόγω της συχνότητας εμφάνισης στο έργο του.



Σχήμα 1.4 Ο ορισμός ενός επιστήμονα δεδομένων όπως αναθεωρήθηκε και επεκτάθηκε από τους Stadelmann et al.(2013)

Τεχνολογία και Διαχείριση Δεδομένων: είναι η βασική δουλειά του επιστήμονα δεδομένων. Η διαχείρισή τους απαιτεί χωρίς περιορισμούς υψηλή τεχνολογία, πολύπλοκες βάσεις δεδομένων και συγκεκριμένες γλώσσες προγραμματισμού. Το βασικό του υπόβαθρο περιλαμβάνει διάφορους τομείς από την επιστήμη των υπολογιστών καθώς και τεχνογνωσία στον τομέα της τεχνολογίας λογισμικού.

Ανάλυση, η ανάλυση δεδομένων και κυρίως η μηχανική μάθηση, είναι μία από τις βασικές ικανότητες ενός επιστήμονα δεδομένων για την εξαγωγή γνώσεων από δεδομένα. Η γνώσεις του, προέρχονται από την τομή των επιστημών της στατιστικής (Wasserman 2013) και της τεχνητής νοημοσύνης (Russell and Norvig 2010).

Στατιστική είναι η επιστήμη που ασχολείται με τη συλλογή, επεξεργασία, παρουσίαση και ανάλυση αριθμητικών δεδομένων, με σκοπό την εξαγωγή συμπερασμάτων χρήσιμων στη λήψη ορθών αποφάσεων (Παπακωνσταντίνου και Καΐτσα 1995).

Τεχνητή Νοημοσύνη, ένας από τους πρώτους ορισμούς που διατυπώθηκαν για την τεχνητή νοημοσύνη (Artificial Intelligence), δόθηκε από τους Barr και Feigenbaum και αναφέρει ότι: «είναι ο τομέας της επιστήμης των υπολογιστών, που ασχολείται με τη σχεδίαση ευφυών, (νοημόνων) υπολογιστικών συστημάτων, δηλαδή συστημάτων που επιδεικνύουν χαρακτηριστικά που σχετίζονται με τη νοημοσύνη στην ανθρώπινη συμπεριφορά» (The Handbook of Artificial Intelligence 1981).

Επιχειρηματικότητα: επιπλέον διαθέτει επιχειρηματικές δεξιότητες και ξέρει πως λειτουργούν οι επιχειρήσεις. Συγκεκριμένα απαιτείται μία ολοκληρωμένη εικόνα του κλάδου στο σύνολό του στον οποίο ανήκει η επιχείρηση, για να μπορεί να διακρίνει ποια προβλήματα είναι σημαντικά για αυτήν και πως θα την οδηγήσει προς τη σωστή κατεύθυνση.

Επικοινωνία: είναι ιδιαίτερα επικοινωνιακός, και μπορεί να παρουσιάσει με τρόπο απλό και κατανοητό τα συμπεράσματά του, σε μια μη τεχνική ομάδα, όπως είναι το τμήμα μάρκετινγκ ή πωλήσεων. Εκτελεί στο παρασκήνιο πλήρη αναλυτική εργασία, αλλά μπορεί να κοινοποιεί τα αποτελέσματα στην ανώτερη διοίκηση με συνοπτικούς τρόπους, (οπτικοποίηση πληροφοριών, δημιουργία γραφικών, στοχευμένη παρουσίαση κλπ). Η επιτυχία του έγκειται στην γνώση που εξάγει έγκυρα και έγκαιρα.

1.3 Ιστορική Αναδρομή του Επιστήμονα Δεδομένων

Και ενώ ο Jeff Wu (1997), χρησιμοποίησε τον όρο «Επιστήμονας Δεδομένων», ως αντικατάσταση του "στατιστικού", το επαγγελματικό τους προφίλ αναδύθηκε πίσω στο μακρινό 2008 όταν ο Hammerbacher και ο Patil επανεξέτασαν τον ρόλο τους στο Facebook και στο LinkedIn (Patil 2011). Μία από τις πρώτες επιστημονικές δημοσιεύσεις που περιγράφουν τι μπορεί να είναι ένας επιστήμονας δεδομένων, γίνεται το 2001 από τον Cleveland όταν παρουσιάζει ένα σχέδιο για να εντάξει την επιστήμη δεδομένων ως επέκταση των τεχνικών

πεδίων της στατιστικής. Παρόμοια οι Provost και Fawcett (2013), προσπάθησαν να συσχετίσουν θέματα όπως τα μεγάλα δεδομένα και τη λήψη αποφάσεων βάση δεδομένων, προσδιορίζοντας έτσι τις θεμελιώδεις αρχές της επιστήμης δεδομένων. Ο Dhar (2013), περιγράφει ένα σύνολο δεξιοτήτων που έχουν οι επιστήμονες δεδομένων και περιλαμβάνει από μαθηματικά, μηχανική μάθηση, τεχνητή νοημοσύνη, στατιστική, βάσεις δεδομένων κλπ. Η IBM (2014), στην ερώτηση τι είναι ο επιστήμονας δεδομένων, απάντα ότι είναι η εξέλιξη του αναλυτή επιχειρήσεων (business analyst) ή δεδομένων (data analyst). Η εκπαίδευση είναι παρόμοια γιατί εστιάζει στις: επιστήμη των υπολογιστών, εφαρμογές, μοντελοποίηση, στατιστική, και μαθηματικά. Τέλος η Power (2016), εξετάζει βασικές δεξιότητες των επιστημόνων δεδομένων, οι οποίοι χρησιμοποιούν αναλύσεις για να υποστηρίξουν τη λήψη αποφάσεων, και αναφέρεται στους ακαδημαϊκούς οι οποίοι πρέπει να προετοιμάσουν αυτούς τους επαγγελματίες.

1.3.1 Ο Ρόλος του Επιστήμονα Δεδομένων

Θα λέγαμε λοιπόν, ότι το προφίλ του προέρχεται από την τομή των δεξιοτήτων ενός επιστήμονα υπολογιστών (Cleveland, 2001) ή ενός στατιστικού (Cleveland 2001, Warden, 2011), ή ακόμα και το συνδυασμό αυτών των δύο. Αλλά για να καταλάβουμε τι δουλειά κάνει ακριβώς, πρέπει να εξετάσουμε τί ζητούν οι εταιρείες. Τα ακόλουθα είναι μία μικρή περίληψη των γνώσεων και των δεξιοτήτων που αναζητούνται από τις μεγάλες εταιρείες όπως το Google και το Facebook (Carlos Costa, Maribel Yasmina Santos 2017). Έτσι,

<u>ένας επιστήμονας δεδομένων θα πρέπει να γνωρίζει:</u>

- ✓ την επιστήμη των υπολογιστών, μηχανική μάθηση για την επεξεργασία δεδομένων και την απόκτηση γνώσεων από αυτά
- ✓ μεγάλα δεδομένα και λήψη αποφάσεων βάσει δεδομένων

- ✓ ποσοτική ανάλυση, στατιστικές μεθόδους και τεχνικές εξόρυξης γνώσης από δεδομένα
- ✓ εργαλεία απεικόνισης δεδομένων και αναπαράστασης/παρουσίασης αποτελεσμάτων
- ✓ βάσεις δεδομένων, ανάλυση δεδομένων και γλώσσες προγραμματισμού

<u>επίσης θα πρέπει να είναι σε θέση να :</u>

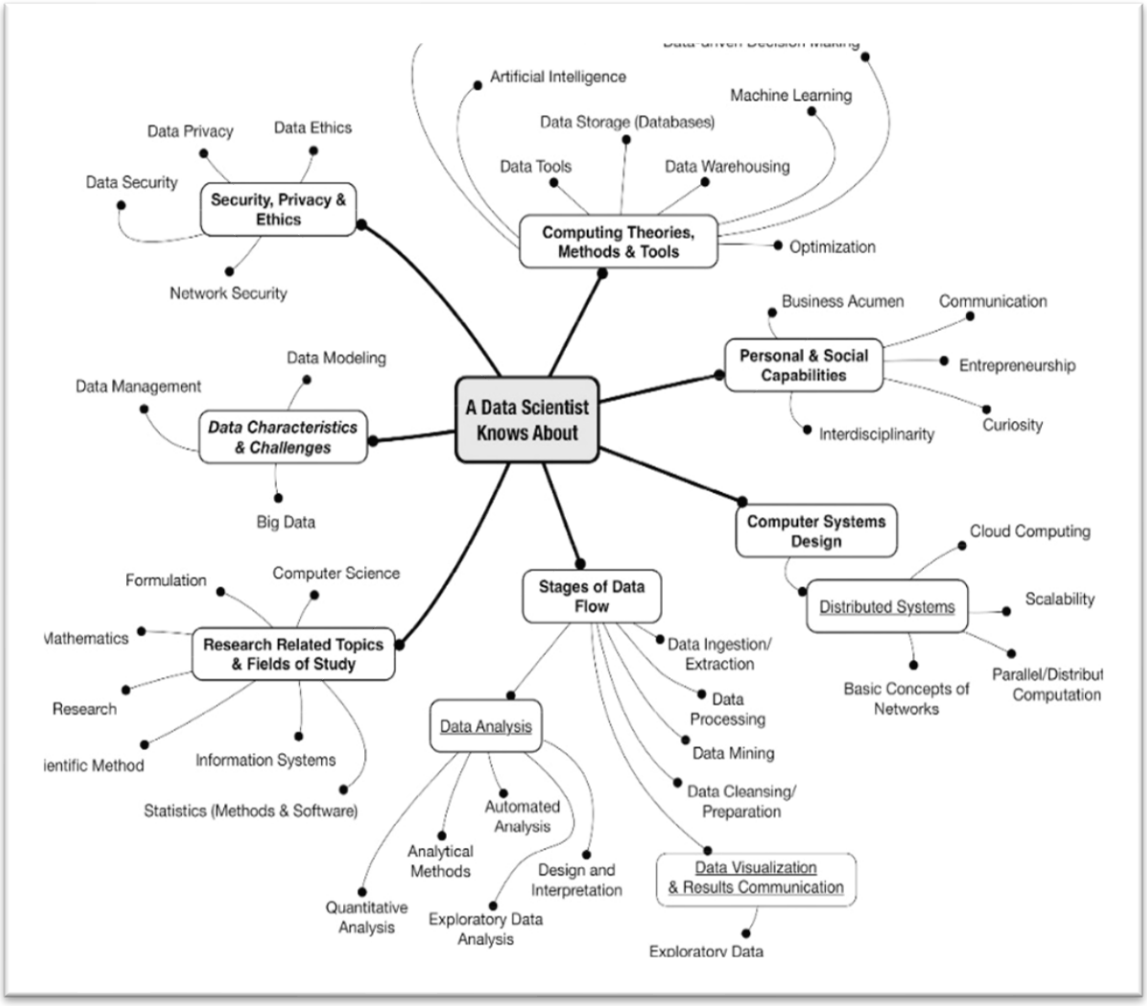
- ✓ κάνει χρήση αναλυτικών στοιχείων για την προώθηση, την ανάπτυξη και την επιτυχία ενός προϊόντος
- ✓ κατανοεί τον τρόπο αλληλεπίδρασης των χρηστών με τα προϊόντα των επιχειρήσεων
- ✓ συνεργάζεται με άλλες ομάδες για την επίλυση προβλημάτων και εντοπίζει τάσεις και ευκαιρίες
- ✓ θέτει στόχους και παρακολουθεί τις μετρήσεις του προϊόντος
- ✓ κάνει αναλύσεις των δεδομένων σχεδιασμού
- ✓ κατευθύνει στρατηγικές και προτείνει προϊόντα.

Παρόμοια, μεγάλες εταιρείες εύρεσης εργασίας, αναγνωρίζουν τον επιστήμονα δεδομένων ως τον επαγγελματία που:

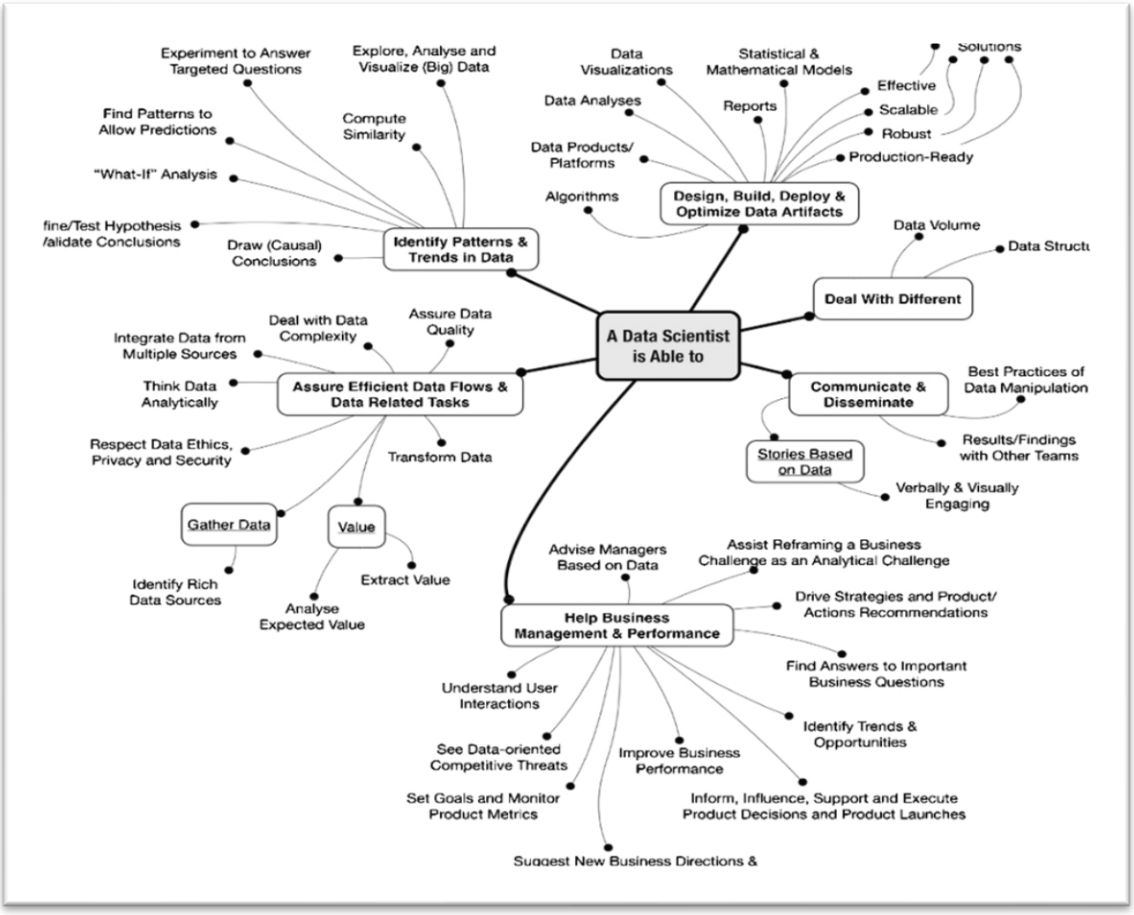
- ✓ αναζητά τρόπους για να αποκτήσει νέες πηγές δεδομένων,
- ✓ χρησιμοποιεί μοντέλα πρόβλεψης για να βελτιώσει την εμπειρία των πελατών, τη στόχευση διαφημίσεων, τη δημιουργία εσόδων κλπ,
- ✓ αναπτύσσει διαδικασίες και εργαλεία για την ανάλυση και την παρακολούθηση της απόδοσης του μοντέλου, εξασφαλίζοντας ταυτόχρονα την ακρίβεια των δεδομένων,
- ✓ αμφισβητεί τις υπάρχουσες υποθέσεις και διαδικασίες και υποβάλει ερωτήσεις του τύπου "τι γίνεται αν" κλπ.
- ✓ έχει γνώσεις προγραμματισμού: Java, JavaScript, C, C ++ , κλπ

- ✓ έχει γνώση / εμπειρία με μεγάλες υπηρεσίες ιστού, ανάλυσης κοινωνικών δικτύων, σε στατιστικές και τεχνικές εξόρυξης δεδομένων, τεχνικές παλινδρόμησης, σε πολύπλοκα προγραμματιστικά εργαλεία κλπ

Ένα εννοιολογικό πρότυπο για το επαγγελματικό προφίλ ενός επιστήμονα δεδομένων, προτείνουν στην εργασία τους με θέμα: “The data scientist profile and its representativeness in the European e-Competence framework and the skills framework for the information age”, οι Carlos Costa και Maribel Yasmina Santos (2017). Σύμφωνα με το οποίο, η αντιπροσωπευτικότητα αυτού του προφίλ αξιολογείται σε δύο πλαίσια, ικανοτήτων και δεξιοτήτων, στον τομέα της Τεχνολογίας Πληροφοριών και Επικοινωνιών (ΤΠΕ).



Σχήμα 1.5 Εννοιολογικό πρότυπο για το προφίλ Data Scientist – βάση γνώσεων (πηγή: ScienceDirect)



Σχήμα 1.6 Εννοιολογικό πρότυπο για το προφίλ Data Scientist – σύνολο δεξιοτήτων (πηγή: ScienceDirect).

Μία ενδιαφέρουσα συζήτηση για το τι αναμένεται ή όχι από τους σύγχρονους επιστήμονες δεδομένων, κάνουν οι Stadelman et al. (2013). Θεωρούν ότι το έργο ενός επιστήμονα δεδομένων περιορίζεται κάπως από τα εργαλεία επιχειρηματικής ευφυΐας. Επειδή ακριβώς χρησιμοποιούνται συχνά για την παρακολούθηση των κοινωνικών μέσων ενημέρωσης και λόγω της απλότητάς τους, έχουν αποτελέσματα πατώντας απλά ένα κουμπί. Όμως είναι σημαντικό να διατηρηθεί η επιστήμη ως κύριο συστατικό στον επιστήμονα των δεδομένων.

Επιπλέον κάνουν λόγο για επιστήμονες δύο εννοιών. Επιστήμονες τύπου “A” ή “B”, και επιστήμονες τύπου “I” ή “II”. Οι τύπου A, έχουν εκπαιδευτεί ως

στατιστικοί, αλλά έχουν διευρύνει το πεδίο τους στην επιστήμη δεδομένων, και οι τύπου Β, έχουν τις βάσεις τους στον προγραμματισμό αλλά επεμβαίνουν στον κώδικα αν χρειαστεί. Και οι δύο τύποι επιστημόνων θα πρέπει να σκέφτονται έξω από τα στεγανά της αρχικής τους επιστήμης, αυτής καθαυτής, προκειμένου να αντιμετωπίσουν προβλήματα χωρίς τυπικούς περιορισμούς.

Από την άλλη πλευρά, η έννοια των επιστημόνων τύπου Ι και ΙΙ, αναφέρεται στους επιστήμονες τύπου Ι ως τεχνικούς, δηλαδή αυτούς που κατέχουν παραδοσιακούς τίτλους εργασίας, στατιστικός, επιχειρησιακός αναλυτής, ειδικός επιχειρηματικής ευφυΐας, μηχανικός βάσεων δεδομένων, μηχανικός λογισμικού κ.α. Ενώ στους τύπου ΙΙ ως διευθυντές, που ενδιαφέρονται για την καθοδήγηση των επαγγελματιών του τομέα των δεδομένων και έχουν υψηλού επιπέδου άποψη για τις δυνατότητες της επιστήμης δεδομένων. Εάν θεωρήσουμε τους τίτλους αυτούς ως ρόλο, τότε περιγράφουν με ακρίβεια το σύνολο δεξιοτήτων του επιστήμονα δεδομένων.

1.3.2 Η Καριέρα του Επιστήμονα Δεδομένων

Ο Hal Varian, επικεφαλής οικονομολόγος της Google, το 2009, είχε πει ότι η πιο ελκυστική δουλειά τα επόμενα δέκα χρόνια θα είναι στατιστικοί, *(«I keep saying the sexy job in the next ten years will be statisticians. People think I'm joking, but who would've guessed that computer engineers would've been the sexy job of the 1990s»?).* Για να συμπληρώσουν τρία χρόνια αργότερα ο Davenport και ο Patil (2012), τον επιστήμονα δεδομένων ως την πιο σέξι εργασία του 21ου αιώνα.

Πώς μπορεί λοιπόν κάποιος να γίνει επιστήμονας δεδομένων; Τι χρειάζεται; Μια τυπική σταδιοδρομία μπορεί να αρχίσει με σπουδές προπτυχιακού επιπέδου σε τομείς όπως η στατιστική, τα μαθηματικά, την επιστήμη των υπολογιστών ή σε οποιαδήποτε άλλη επιστήμη με έμφαση στα δεδομένα. Από εκεί και πέρα οι δεξιότητες μπορούν να ενισχυθούν μέσα από εκπαίδευση και συνεργασία με

άλλους κλάδους. Ένα μεταπτυχιακό σε οποιοδήποτε τομέα της επιστήμης θα βοηθούσε μεν αλλά δεν θα ήταν αρκετό δε, γιατί αυτό που μετράει περισσότερο είναι η “απόδειξη” εμπειρίας που αποκτήθηκε μέσα από προσωπική ή ομαδική εργασία.

1.3.3 Η εκπαίδευση του Επιστήμονα Δεδομένων

Ένα πρόγραμμα σπουδών για τις επιστήμες των δεδομένων θα πρέπει να περιλαμβάνει του τρεις ακόλουθους τομείς και συγκεκριμένη εξειδίκευση στα εξής θεματικά πεδία (Stadelman et al. 2013)

I. Επιχειρήσεις:

- Οπτικοποίηση και επικοινωνία των αποτελεσμάτων
- Προστασία της ιδιωτικότητας, ασφάλεια και δεοντολογία
- Επιχειρηματικότητα και σχεδιασμός προϊόντων δεδομένων

II. Αλγόριθμοι:

- Εξόρυξη δεδομένων και στατιστική (Data mining and statistics)
- Μηχανική Μάθηση (Machine Learning)
- Ανάκτηση πληροφοριών και επεξεργασία φυσικής γλώσσας (Information Retrieval and natural language processing)
- Επιχειρησιακή νοημοσύνη και οπτική ανάλυση (Business intelligence and visual analytics)

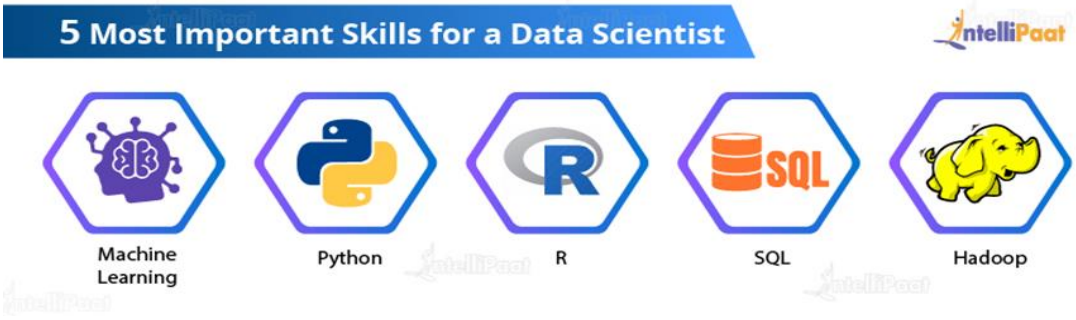
III. Υποδομές:

- Βάσεις δεδομένων, αποθήκες δεδομένων και συστήματα πληροφοριών

πολογισμός νέφους (Cloud computing) και τεχνολογία Μεγάλων Δεδομένων (Big Data)

1.3.4 Οι Δεξιότητες του Επιστήμονα Δεδομένων

Οι πέντε πιο σημαντικές δεξιότητες για έναν Επιστήμονα Δεδομένων όπως περιγράφονται από μεγάλες πύλες εργασίας παρουσιάζονται στο σχήμα που ακολουθεί.



Σχήμα 1.7 Οι δεξιότητες του Επιστήμονα Δεδομένων (πηγή: intellipaat.com)

Ως Μηχανική Μάθηση (Machine Learning), εννοούμε την ικανότητα των υπολογιστών συστημάτων να μαθαίνουν από δεδομένα και να κάνουν προβλέψεις σχετικά με αυτά.

Η Python, είναι η πιο κοινή γλώσσα προγραμματισμού. Το κύριο χαρακτηριστικό της είναι η αναγνωσιμότητα του κώδικά της και η ευκολία στη χρήσης της.

Η R είναι μια γλώσσα προγραμματισμού που χρησιμοποιείται κυρίως για στατιστικούς υπολογισμούς αλλά παρέχει επίσης τη δυνατότητα της παραγωγής γραφικών απεικονίσεων.

Η SQL είναι μία γλώσσα υπολογιστών, που σχεδιάστηκε για τη διαχείριση δεδομένων και με την οποία μπορούμε να εκτελέσουμε κάποιες λειτουργίες από μια βάση δεδομένων όπως προσθήκη, διαγραφή και εξαγωγή δεδομένων.

Το Hadoop είναι μία συλλογή ανοιχτού κώδικα που περιλαμβάνει βοηθητικά προγράμματα λογισμικού τα οποία διευκολύνουν τη χρήση ενός δικτύου πολλών υπολογιστών. Χρησιμοποιείται για την επίλυση προβλημάτων που αφορούν τεράστιες ποσότητες δεδομένων και για στατιστικές αναλύσεις.

ΚΕΦΑΛΑΙΟ 2^{Ο} Βασικές Τεχνολογίες της Επιστήμης Δεδομένων

2.1 Η σημασία της μάθησης για τον άνθρωπο

Η εξέλιξη είναι η κινητήρια δύναμη που καθορίζει τα ιδιαίτερα χαρακτηριστικά ενός οργανισμού και του δίνει την δυνατότητα να προσαρμόζεται και να επιβιώνει σε ένα συνεχώς μεταβαλλόμενο περιβάλλον. Για τον άνθρωπο την κινητήρια δύναμη αυτή αποτελεί ο εγκέφαλός του, που λειτουργεί ως μηχανισμός μάθησης μέσω του οποίου παρατηρεί, μαθαίνει και αποθηκεύει εμπειρική γνώση και την ανακαλεί όποτε χρειαστεί αν η ίδια κατάσταση προκύψει ξανά. Η Μάθηση (learning), είναι μία από τις βασικότερες λειτουργίες της ανθρώπινης συμπεριφοράς και πάντοτε οι άνθρωποι αναζητούσαν στο περιβάλλον τους παραστάσεις, (μοντέλα – πρότυπα), προσπαθώντας να εξηγήσουν οτιδήποτε συμβαίνει γύρω τους.

Και για τις μηχανές τι ισχύει; Χρειάζεται όλες αυτές οι ανθρώπινες λειτουργίες να ενσωματωθούν σε αυτοματοποιημένες διαδικασίες; Ναι γιατί τα μηχανήματα ξεπερνούν τους ανθρώπους σε δυνατότητες και οι επιστήμονες προσπαθούν να ερμηνεύσουν τα δεδομένα που η κοινωνία παράγει, και να ενσωματώσουν τον εμπειρικό τρόπο σκέψης των ανθρώπων σε υπολογιστικές οντότητες, με στόχο την εξαγωγή συμπερασμάτων μέσω της ανάλυσής τους. Ο Alan Turing (1950), στην εργασία του "Computing Machinery and Intelligence", διερωτάται «Οι μηχανές μπορούν να σκεφτούν»; Ενώ ο Stevan Harnad (2008) συμπληρώνει, "Μπορούν οι μηχανές να κάνουν ό,τι μπορούν οι στοχαστές σαν και εμάς να κάνουν , και αν ναι πώς"; Την απάντηση θα δώσει ο τομέας που ασχολείται με την συστηματική ανάλυση δεδομένων για την εξαγωγή χρήσιμης γνώσης, που ονομάζεται μηχανική μάθηση.

2.2 Μηχανική Μάθηση

Η ικανότητα ενός υπολογιστικού συστήματος να μελετά και να δημιουργεί αλγόριθμους από ένα σύνολο δεδομένων και να κάνει προβλέψεις σχετικά με αυτά ονομάζεται μηχανική μάθηση (Machine Learning).

Από τους πρώτους που καθιέρωσαν τον όρο ήταν ο Arthur Samuel (1959), ο οποίος όρισε τη μηχανική μάθηση ως το "πεδίο μελέτης που δίνει στους υπολογιστές την ικανότητα να μαθαίνουν, χωρίς να έχουν ρητά προγραμματιστεί". Ενώ έναν πιο λειτουργικό ορισμό που χρησιμοποιείται συχνότερα έδωσε Tom Mitchell (1997), "Ένα πρόγραμμα υπολογιστή λέμε ότι μαθαίνει από την εμπειρία E ως προς κάποια κλάση εργασιών T και μέτρο απόδοσης P, αν η απόδοσή του σε εργασίες από το T, όπως μετριέται από το P, βελτιώνεται μέσω της εμπειρίας E".

Σήμερα ζούμε σε μία εποχή ραγδαίας τεχνολογικής ανάπτυξης όπου οι επιχειρήσεις από την μία αναζητούν καινοτόμους τρόπους για να δημιουργήσουν ανταγωνιστικό πλεονέκτημα, και οι επιστήμονες από την άλλη προσπαθούν να ερμηνεύσουν μοντέλα που θα παράγουν αξιόπιστα αποτελέσματα. Η μηχανική μάθηση χρησιμοποιείται ευρέως σε εμπορικές και σε ερευνητικές εφαρμογές και η δυναμική της αναγνωρίζεται από πολλούς κλάδους. Μερικά πεδία εφαρμογής της είναι η ανάλυση συναισθήματος σε κοινωνικά δίκτυα, η παρακολούθηση των χρηματαγορών για την πορεία των μετοχών και η πρόβλεψη μελλοντικών οικονομικών καταστάσεων. Άλλα παραδείγματα εφαρμογών της είναι στα ηλεκτρονικά μηνύματα (spam filtering), στην οπτική αναγνώριση χαρακτήρων (OCR), στις μηχανές αναζήτησης και στην υπολογιστική όραση, σύμφωνα με το Computer Science Center (CSC, 2018). Η μηχανική μάθηση αλλάζει τον τρόπο με τον οποίο οι λιανοπωλητές δραστηριοποιούνται και πολλές εταιρείες την

χρησιμοποιούν για να βελτιστοποιήσουν τις διαδικασίες τους και για να δημιουργήσουν ισχυρότερες σχέσεις με τους πελάτες (Gottsegen, 2019).

Στον τραπεζικό κλάδο μπορεί να αποφέρει μείωση του κόστους έως και 25% σε λειτουργίες πληροφορικής, σε υποδομές και συντήρηση αλλά και αύξηση εσόδων, (ανάπτυξη νέων προϊόντων, διατήρηση παλαιών ή απόκτηση νέων πελατών) σύμφωνα με την έκθεση της Accenture (2018). Ενώ οι Rizzi et al (2018), υποστηρίζουν ότι μέσω της μηχανικής μάθησης και των τεχνικών της, παρέχεται στις εταιρείες η δύναμη να αντιστοιχίσουν τις τιμές των προϊόντων τους με την τρέχουσα αξία, λαμβάνοντας ταυτόχρονα υπόψη και το ανταγωνιστικό περιβάλλον.

2.2.1 Είδη Μηχανικής Μάθησης

Αν.άλογα με το είδος της γνώσης που είναι διαθέσιμη σε ένα σύστημα εκμάθησης, η μηχανική μάθηση διακρίνεται σε: επιβλεπόμενη μάθηση, μη επιβλεπόμενη μάθηση και ενισχυτική μάθηση. Συγκεκριμένα:

Επιβλεπόμενη Μάθηση (Supervised Learning), είναι η διαδικασία κατά την οποία ένα σύστημα δέχεται σύνολο δεδομένων που αποτελούνται από ζεύγη εισόδου και εξόδου με σκοπό να μάθει έναν γενικό κανόνα για να αντιστοιχεί τις εισόδους με τα αποτελέσματα. Χρησιμοποιείται σε προβλήματα ταξινόμησης (Classification) και πρόβλεψης (Prediction).

Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning), το σύστημα προσπαθεί να βρει την δομή των δεδομένων εισόδου χωρίς να γνωρίζει τις επιθυμητές εξόδους. Η μη επιβλεπόμενη μηχανική μάθηση προσπαθεί να εντοπίσει ομοιότητες και να ομαδοποιήσει τα στοιχεία για να αποκτήσει σημαντικές πληροφορίες. Χρησιμοποιείται σε προβλήματα ανάλυσης συσχετισμών (Association Analysis) και ομαδοποίησης (Clustering).

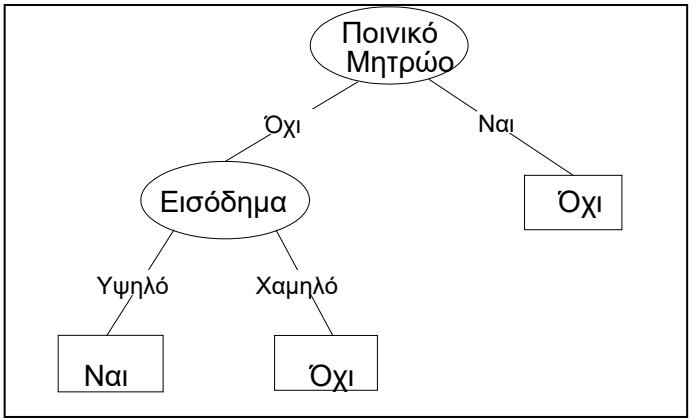
Ενισχυτική Μάθηση (Reinforcement Learning), όπου ένα σύστημα μαθαίνει μια στρατηγική και πώς να αλληλοεπιδρά με ένα δυναμικό περιβάλλον στο οποίο πρέπει να επιτευχθεί ένας συγκεκριμένος στόχος. Χρησιμοποιείται κυρίως σε προβλήματα σχεδιασμού (Planning), όπως για παράδειγμα να μάθει να παίζει ένα παιχνίδι εναντίον κάποιου αντιπάλου.

2.3 Βασικές κατηγορίες αλγορίθμων

Για κάθε ένα προς επίλυση πρόβλημα στη Μηχανική Μάθηση, υπάρχει ένας τουλάχιστον κατάλληλος τρόπος μάθησης, και για κάθε τρόπο μάθησης υπάρχει τουλάχιστον ένας κατάλληλος αλγόριθμος που μπορεί να χρησιμοποιηθεί. Οι αλγόριθμοι μηχανικής μάθησης, έχουν ως στόχο την δημιουργία μοντέλων όσο το δυνατόν πιο κοντά στα δεδομένα που εξετάζονται.

2.3.1 Ο αλγόριθμος ID3

Ο ID3 (Iterative Dichotomiser 3) είναι ένας αλγόριθμος, που χρησιμοποιείται για τη δημιουργία ενός δέντρου απόφασης. Σε κάθε κόμβο του δέντρου ο αλγόριθμος κάνει ερωτήσεις και επιλέγει αυτές των οποίων οι απαντήσεις παρέχουν περισσότερη πληροφορία. Ο αλγόριθμος προσπαθεί να ελαχιστοποιήσει τον αναμενόμενο αριθμό συγκρίσεων και σταματά εάν βρει το χαρακτηριστικό που διαχωρίζει πλήρως το δείγμα, αλλιώς συνεχίζει μέχρι να το εντοπίσει. Η μεγαλύτερη πρόκληση του αλγορίθμου ID3 είναι ο εντοπισμός του χαρακτηριστικού με το υψηλότερο κέρδος πληροφορίας. Η στατιστική μέθοδος που χρησιμοποιείται για να μετρηθεί η πληροφορία, καλείται εντροπία. Η εντροπία μετρά την αβεβαιότητα σε ένα σύνολο δεδομένων. Αν τα δεδομένα είναι ομοιογενή, η εντροπία θα είναι 0.

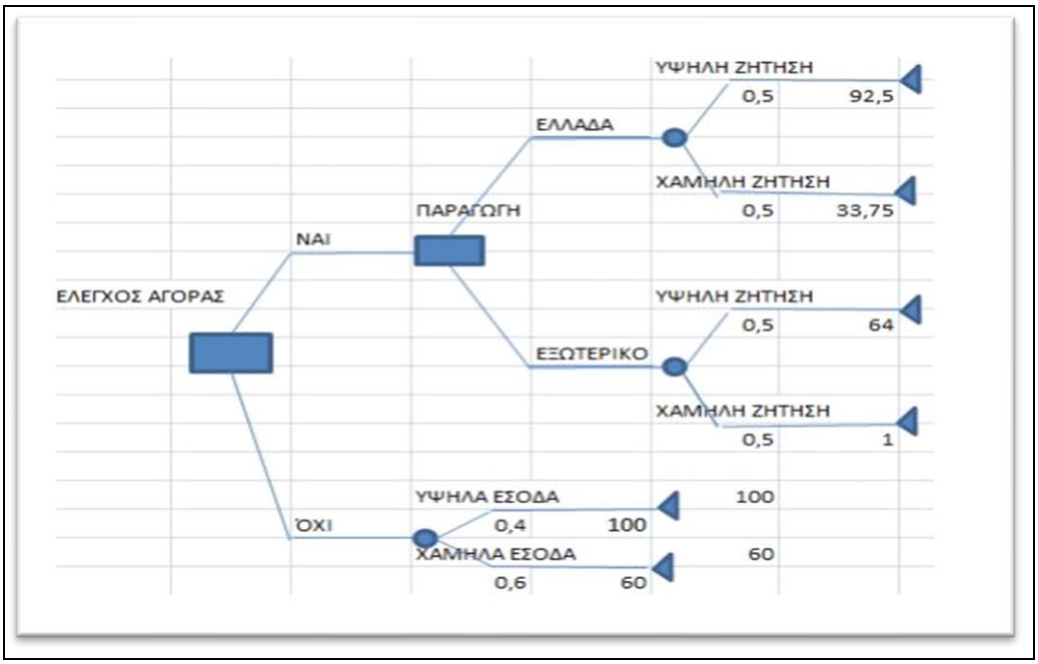


Εικόνα 2.1 Δέντρα Απόφασης (πηγή upatras eclass Πληροφοριακά Συστήματα Διοίκησης Βουτσινάς)

2.3.2 Δέντρα Απόφασης

Τα Δένδρα Απόφασης-ΔΑ (Decision Trees), είναι ο πιο γνωστός αλγόριθμος για προβλήματα κατηγοριοποίησης. Χρησιμοποιούνται για να προβλέψουν και εφαρμόζονται στους τομείς όπου απαιτείται ταξινόμηση. Η απεικόνιση γίνεται σε δενδροειδή μορφή όπου κάθε κόμβος απεικονίζει μία κατάσταση απόφασης και κάθε φύλλο (παιδί του κόμβου), απεικονίζει την πιθανή επιλογή που μπορεί να γίνει σε κάθε απόφαση.

Είναι μια μορφή εποπτευόμενης μάθησης, καθώς τα δέντρα μπορούν πρώτα να μάθουν χρησιμοποιώντας εκπαιδευτικές παρατηρήσεις και στη συνέχεια να χρησιμοποιηθούν για την πρόβλεψη στο σύνολο δοκιμών. Χρησιμοποιούνται ευρέως, ειδικά όταν ο αριθμός των μεταβλητών πρόβλεψης είναι μικρός, γιατί είναι εύκολα ερμηνεύσιμοι.

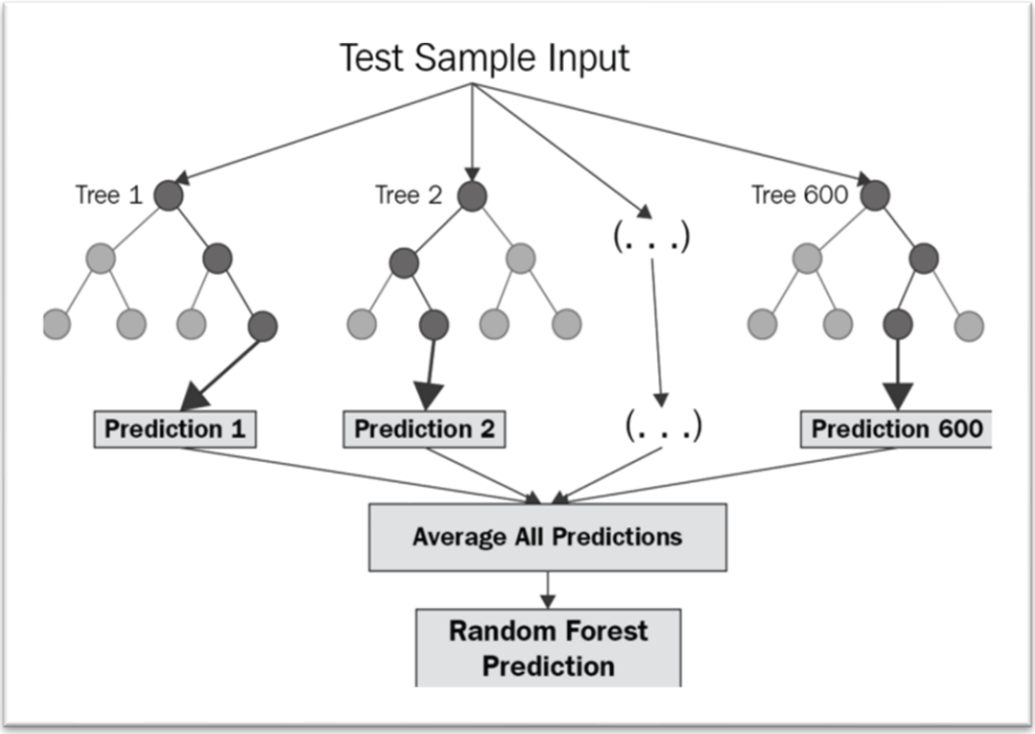


Εικόνα 2.2 Δέντρα Απόφασης (πηγή upatras eclass Λήψη Επιχειρηματικών Αποφάσεων Μητρόπουλος – Βάσιου)

2.3.3 Τυχαία Δάση

Τα τυχαία δάση (random forests), είναι ένας δημοφιλής αλγόριθμος ταξινόμησης που χρησιμοποιείται για πρόβλεψη, και λειτουργεί τόσο με κατηγορικές μεταβλητές (προβλήματα ταξινόμησης), όσο και με συνεχείς μεταβλητές (προβλήματα παλινδρόμησης). Είναι ένας εποπτευόμενος αλγόριθμος της μηχανικής μάθησης, που κατασκευάζει ένα μεγάλο σύνολο από μεμονωμένα δέντρα απόφασης, μη συσχετισμένα μεταξύ τους, προκειμένου να πραγματοποιήσει μία πιο ακριβής και πιο σταθερή πρόβλεψη. Η μείωση της συσχέτισης μεταξύ των δέντρων, επιτυγχάνεται μέσω της επιλογής τυχαίου αριθμού μεταβλητών σε κάθε εσωτερικό κόμβο διαχωρισμού. Όσο μεγαλύτερος είναι ο αριθμός των δέντρων (το δάσος), τόσο πιο ακριβές θα είναι το αποτέλεσμα. Το δάσος έχει την ίδια κατανομή για όλα τα δέντρα και το σφάλμα εξαρτάται από το κάθε ένα δέντρο αρκεί να μην έχουν όλα λάθος πάντα προς την ίδια κατεύθυνση. Έτσι, και αν ακόμα μερικά δέντρα μπορεί να είναι λάθος,

κάποια άλλα θα είναι σωστά και ως ομάδα μπορούν να επιτύχουν καλύτερη πρόβλεψη, από οποιοδήποτε μεμονωμένο μοντέλο.



Εικόνα 2.3 Αλγόριθμος Τυχαίου Δάσους (πηγή https://medium.com/@aaaanchakure/random-forest-and-its-implementation-71824ced454f)

2.3.4 Μάθηση κατά Bayes

Ο κανόνας του Bayes είναι ένα πιθανοθεωρητικό μοντέλο που υπολογίζει την πιθανότητα κάθε υπόθεσης με δεδομένη την τιμή κάποιου δεδομένου. Μας βοηθά στο να συνδυάσουμε την "a-priori" (εκ των προτέρων) γνώση με τα δεδομένα και να εξάγουμε συμπεράσματα, ή για λήψη αποφάσεων. Για παράδειγμα ένα δίκτυο Bayes μπορεί να αναπαραστήσει την πιθανοθεωρητική σχέση μεταξύ ασθενειών και συμπτωμάτων. Δεδομένων των συμπτωμάτων, το δίκτυο μπορεί να χρησιμοποιηθεί για να υπολογίσει τις πιθανότητες παρουσίας διαφόρων ασθενειών.

Δίνεται από την ακόλουθη σχέση:

$$P\ (Y|X)\quad \frac{P(X|Y)P(Y)}{P(X)}$$

Όπου X , Y είναι δεδομένα και αναλυτικά έχουμε:

- P(Y|X) είναι η δεσμευμένη πιθανότητα του Y δεδομένου του X
- P(X|Y) είναι η δεσμευμένη πιθανότητα του Y δεδομένου του X και
- P(Y) είναι η "a-priori" πιθανότητα.

2.3.5 Παλινδρόμηση

Η παλινδρόμηση (regression) είναι μια μέθοδος μοντελοποίησης που χρησιμοποιείται στην μηχανική μάθηση και έχει να κάνει με προβλέψεις. Ερευνά την σχέση μεταξύ μίας εξαρτημένης μεταβλητής y και μιας ή περισσότερων ανεξάρτητων μεταβλητών $x_1,x_2,...x_n$. Το πιο γνωστό μοντέλο είναι το γραμμικό, Απλή Γραμμική Παλινδρόμηση (Simple Linear Regression), όπου από ένα σύνολο τιμών {x,y}, προσπαθούμε να βρούμε ένα απλό μαθηματικό μοντέλο, που να περιγράφει την σχέση αυτών των δύο μεταβλητών. Δίνεται από τον τύπο:

$$\mathbf{y=b_1 x+b_0}$$

όπου:

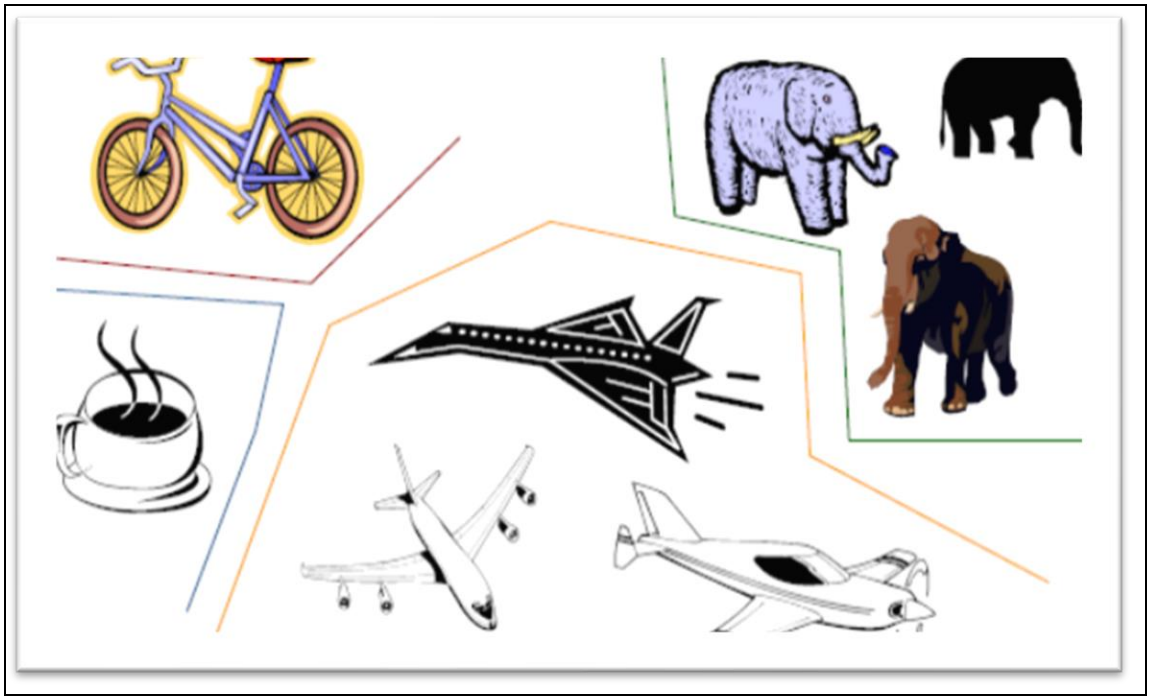
x είναι η ανεξάρτητη μεταβλητή, **y** είναι η εξαρτημένη μεταβλητή

$\mathbf{b_1}$ και $\mathbf{b_0}$ είναι οι τιμές που αναζητούμε για το συγκεκριμένο σύνολο τιμών.

2.3.6 Ταξινόμηση

Η ταξινόμηση (classification) είναι μία από τις βασικότερες τεχνικές μηχανικής μάθησης, η οποία κατατάσσει αντικείμενα του πραγματικού κόσμου (objects) σε προκαθορισμένα σύνολα (ομοειδών αντικειμένων) που ονομάζονται κλάσεις. 'Ολα

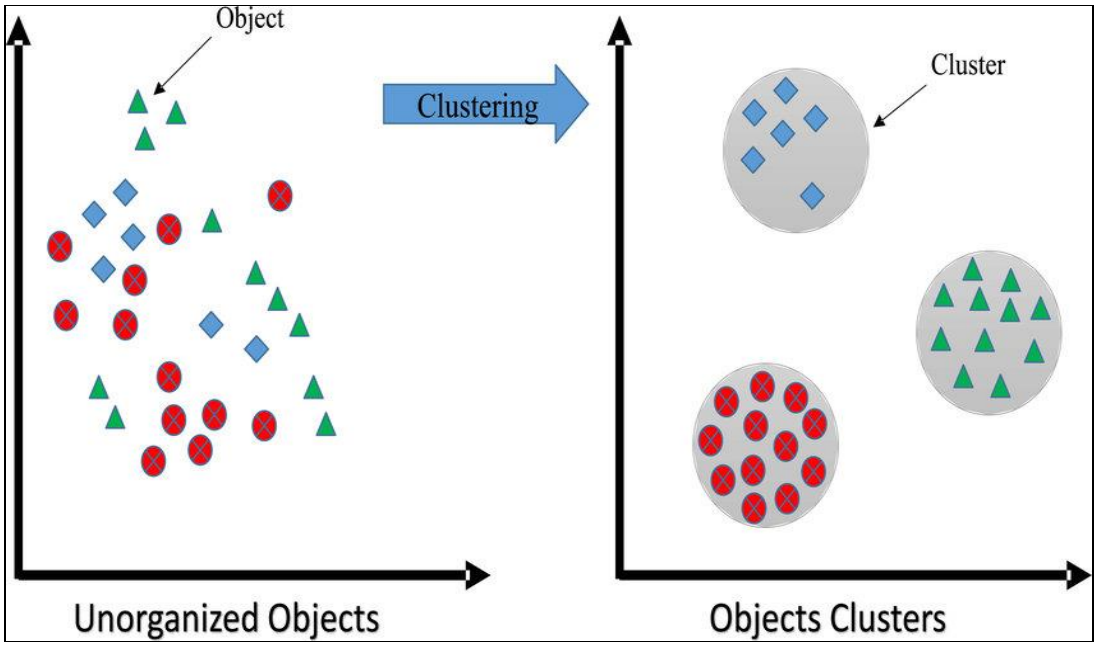
τα αντικείμενα μια κλάσης έχουν τα ίδια χαρακτηριστικά αλλά οι τιμές τους μπορεί να διαφέρουν.



Εικόνα 2.4 Ταξινόμηση (πηγή upatras eclass Αντικειμενοστραφής Προγραμματισμός, Πιερρακέας Χρ.)

2.3.7 Ομαδοποίηση

Η ομαδοποίηση ή συσταδοποίηση (Clustering), είναι μια διαδικασία καταμερισμού ενός συνόλου ετερογενών παρατηρήσεων σε υποσύνολα με βάση κάποιο μέτρο ομοιότητας. Τα υποσύνολα που δημιουργούνται (cluster), περιέχουν παρατήσεις από όμοια μεταξύ τους χαρακτηριστικά και από διαφορετικά με των άλλων υποσυνόλων χαρακτηριστικά.



Εικόνα 2.5 Clustering (πηγή https://www.researchgate.net/figure/An-example-of-the-document-clustering_fig1_322455242)

2.3.8 Αλγόριθμος κ-Πλησιέστερου Γείτονα KNN

Ο αλγόριθμος κ – Πλησιέστερου Γείτονα (k-Nearest Neighbors), είναι μία απλή αλλά επιτυχημένη μέθοδος κατηγοριοποίησης που χρησιμοποιεί η επιβλεπόμενη μηχανική μάθηση και βασίζεται στην απόσταση. Ο αλγόριθμος υποθέτει ότι όμοια πράγματα είναι το ένα κοντά στο άλλο, και η τιμή στόχου προσδιορίζεται αποκλειστικά και μόνο από τις αντίστοιχες τιμές των k πιο «κοντινών» γειτόνων της. Για τον υπολογισμό του αλγορίθμου αρκεί να προσδιορίσουμε την απόσταση μεταξύ δύο στιγμιότυπων, δηλαδή μιας τιμής πάνω στο χώρο των στιγμιότυπων, που θα εκφράζει την «ομοιότητα» μεταξύ τους, και φυσικά την τιμή του k.

Για στιγμιότυπα που ανήκουν στο n-διάστατο χώρο (R^n), προτιμάται η απλή ευκλείδεια απόσταση λόγω της απλότητας στην εφαρμογή της. Δίνεται από τον τύπο :

$$d\ (x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Όπου d είναι η ζητούμενη απόσταση μεταξύ των σημείων x_i και y_i .

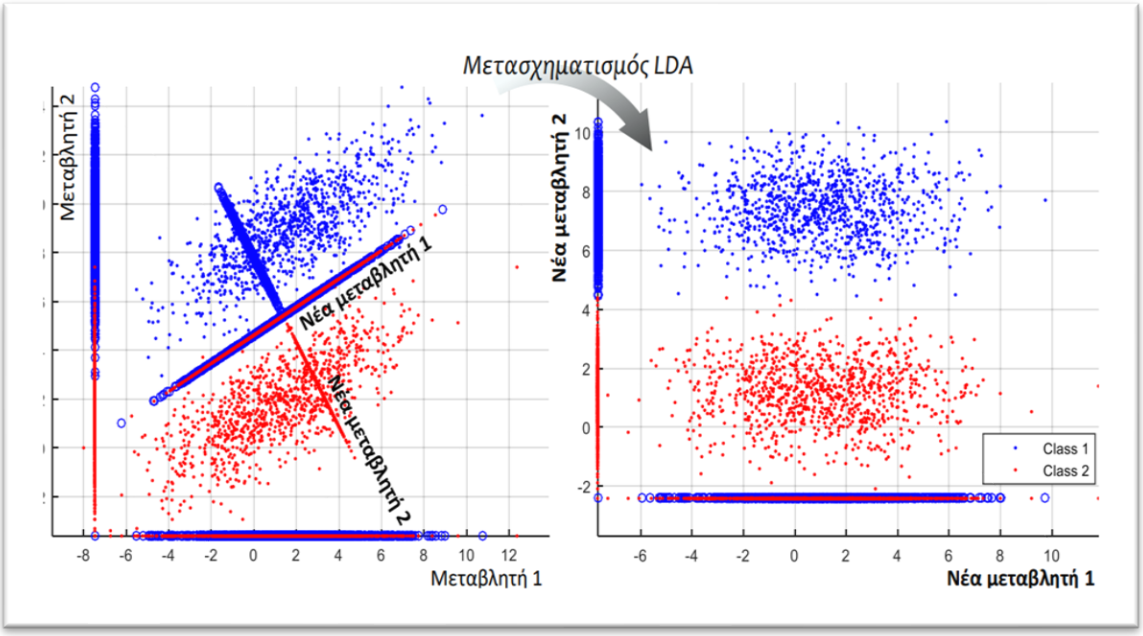
2.3.9 Αλγόριθμος LDA

Η Linear Discriminant Analysis, γραμμική ανάλυση διακρίσεων (LDA), είναι μια μέθοδος κατάταξης δεδομένων που ανήκουν σε κατηγορίες (κλάσεις). Η κεντρική ιδέα της LDA είναι ο μετασχηματισμός των δεδομένων με τέτοιο τρόπο ώστε να μεγιστοποιηθεί η απόσταση μεταξύ των κλάσεων, και ταυτόχρονα να ελαχιστοποιηθεί η διασπορά εντός των κλάσεων. Αποτελεί μια στατιστική μέθοδο που βρίσκει εφαρμογή σε πολλούς κλάδους. Εφαρμόζεται για διερεύνηση των διαφορών των κλάσεων, αλλά και για κατηγοριοποίηση. Ως μέθοδος κατηγοριοποίησης αναζητά τον γραμμικό συνδυασμό των χαρακτηριστικών, και θεωρεί ότι τα αντικείμενα των κλάσεων ακολουθούν την κανονική κατανομή.

Για να εφαρμοστεί η LDA, τα δεδομένα πρέπει:

α) να είναι αριθμητικά με συνεχείς τιμές, και

β) να ανήκουν σε δύο ή περισσότερες γνωστές εκ των προτέρων κατηγορίες (κλάσεις).



Εικόνα 2.6 Αλγόριθμος LDA (πηγή uniwa.gr/eclass Ανάλυση Γραμμικής Διάκρισης, Ιωάννης Καλατζής)

Στο παραπάνω παράδειγμα γίνεται γραφική απεικόνιση εφαρμογής του αλγόριθμου LDA σε σύνολο δεδομένων όπου παρατηρούμε ότι ο διαχωρισμός των κλάσεων είναι βέλτιστος.

2.4 Γλώσσες Προγραμματισμού και Επιστήμη Δεδομένων

Πολλές τεχνολογίες σχετίζονται με την Επιστήμη Δεδομένων καιfa υπάρχουν διάφορες γλώσσες προγραμματισμού που χρησιμοποιούνται ευρέως όπως για παράδειγμα η SQL, JavaScript, Java, C, C++, R, Python αλλά και πολλές άλλες. Ακολουθεί λίστα με τις πιο δημοφιλείς γλώσσες προγραμματισμού για το 2019 όπως δημοσιεύονται στο ιστολόγιο technode για τεχνολογικά θέματα από τον George S. Metallidis

Programming Language	Ratings	Change
Java	16.028%	-0.85%
C	15.154%	+0.19%
Python	10.020%	+3.03%
C++	6.057%	-1.41%
C#	3.842%	+0.30%
Visual Basic .NET	3.695%	-1.07%
JavaScript	2.258%	-0.15%
PHP	2.075%	-0.85%
Objective-C	1.690%	+0.33%
SQL	1.625%	-0.69%
Ruby	1.316%	+0.13%
MATLAB	1.274%	-0.09%
Groovy	1.225%	+1.04%
Delphi/Object Pascal	1.194%	-0.18%
Assembly language	1.114%	-0.30%
Visual Basic	1.025%	+0.10%
Go	0.973%	-0.02%
Swift	0.890%	-0.49%
Perl	0.860%	-0.31%
R	0.822%	-0.14%

Εικόνα 2.7 Δημοφιλείς γλώσσες προγραμματισμού 2019, πηγή: https://technode.gr/)

2.4.1 Η γλώσσα προγραμματισμού Python

Είναι γλώσσα προγραμματισμού ανοιχτού πηγαίου κώδικα (open source), υψηλού επιπέδου και γενικού προγραμματισμού. Είναι αντικειμενοστραφής γλώσσα, ευέλικτη, εύκολη στην εκμάθηση και την χρήση, για αυτό είναι η πρώτη επιλογή για προγραμματιστές αλλά και για επιστήμονες δεδομένων. Το συντακτικό της είναι περιεκτικό και επιτρέπει στους προγραμματιστές να γράψουν Project σε λιγότερες γραμμές κώδικα από ότι σε άλλες γλώσσες (C, C++, Java κλπ). Μπορεί να χρησιμοποιηθεί από την ανάπτυξη ιστού έως την ανάπτυξη λογισμικού, για επιστημονικές εφαρμογές ακόμα και για τον σχεδιασμό παιχνιδιών.

Δημιουργήθηκε από τον Guido van Rossum, στις αρχές της δεκαετίας '90 και ως σήμερα έχουν κυκλοφορήσει πολλές νέες εκδόσεις της. Είναι συμβατή με πολλά λειτουργικά συστήματα, συμπεριλαμβανομένων των Windows, Unix/Linux, Macintosh, iPod, μέχρι και το PlayStation. Περιέχει πλήθος βιβλιοθηκών που διευκολύνουν πολλές εργασίες, και είναι επεκτάσιμη αφού μπορούμε να ορίσουμε νέους τύπους ώστε να λειτουργούν σαν τους ήδη προκαθορισμένους τύπους, που είναι τμήμα της γλώσσας.

2.4.2 Η γλώσσα προγραμματισμού R

Η R δεν είναι απλά μια γλώσσα προγραμματισμού, αλλά και ένα περιβάλλον λογισμικού. Χρησιμοποιείται για αναλύσεις και για γραφικές απεικονίσεις στατιστικών δεδομένων στα μαθηματικά και την στατιστική, αλλά και σε άλλους

τομείς όπως στα οικονομικά, στην αστρονομία, στην χημεία, στην φαρμακευτική, στην ιατρική, στο μάρκετινγκ κτλ.

Είναι γλώσσα ανοιχτού κώδικα, εύκολη στην εκμάθησή της, συμβατή με τα πιο γνωστά λειτουργικά συστήματα Linux, Mac OS και Windows, και διανέμεται δωρεάν. Το γεγονός ότι όλοι έχουν πρόσβαση στον πηγαίο κώδικά της και ο καθένας μπορεί να κάνει διορθώσεις και να τις δημοσιεύει, έχει ως αποτέλεσμα να έχουν γίνει πολλές βελτιώσεις από τότε που δημιουργήθηκε. Διαθέτει έναν μεγάλο αριθμό έτοιμων πακέτων, (πάνω από 1500), με καλογραμμένα εγχειρίδια χρήσης, και το γεγονός ότι είναι εύκολη στην εκμάθησή της, έχει ως αποτέλεσμα να έχει γίνει πολύ δημοφιλής τα τελευταία χρόνια.

Το βασικό της μειονέκτημα της R είναι ότι χαρακτηρίζεται γενικά ως «αργή» γλώσσα και καταναλώνει πολύ μνήμη. Για αυτό άλλωστε και δεν προτιμάται για ανάλυση μεγάλων δεδομένων.

2.5 Τεχνικές Αποθήκευσης δεδομένων σε cloud

Σήμερα ζούμε σε μια εποχή ραγδαίας τεχνολογικής ανάπτυξης όπου σύγχρονες μέθοδοι και αναδυόμενες τεχνολογίες αντικαθιστούν τις παλιές μας συνήθειες και συνεχώς καλούμαστε να μάθουμε σε νέα πεδία προκειμένου να ακολουθήσουμε την πρόοδο.

Κάπως έτσι από την δισκέτα και τα cd φτάσαμε στο cloud «σύννεφο» που είναι υπηρεσία σύγχρονων αποθηκευτικών χώρων για την αυτόματη και online αποθήκευση δεδομένων και πληροφοριών. Η αποθήκευση δεν γίνεται πλέον στον τοπικό υπολογιστή αλλά σε διάφορους υπολογιστές και η πρόσβαση είναι δυνατή μόνο μέσω διαδικτύου. Είναι κάτι παρόμοιο με αυτό που κάνουν τα μέσα κοινωνικής δικτύωσης, στα οποία μπορούμε να συνδεθούμε από διάφορες συσκευές και να διαχειριστούμε το υλικό μας, φωτογραφίες, βίντεο, κλπ . Άλλοι τέτοιοι χώροι είναι η ταινιοθήκη του Netflix ή η λίστα τραγουδιών του Spotify,

στους οποίους υπάρχει ήδη έτοιμη αποθηκευμένη πληροφορία (σε Cloud) και είναι διαθέσιμη σε όποιον χρησιμοποιεί τις υπηρεσίες αυτές (Kotsibou 2019). Με απλά λόγια τα clouds είναι «σύννεφα» από servers στο περιβάλλον των οποίων μπορούμε να αποθηκεύουμε και να επεξεργαζόμαστε πληροφορίες αφού συνδεθούμε στο λογαριασμό μας.

Τελευταία έχουν αναπτυχθεί πολλές υπηρεσίες cloud τεχνολογίας, (cloud computing), αφού και τα πλεονεκτήματα από τέτοιες εφαρμογές είναι σημαντικά.

Ευκολία. Δεν χρειάζεται πλέον να έχουμε μαζί μας extra συσκευές , USB, σκληρό δίσκο κλπ, αρκεί μόνο πρόσβαση στο ίντερνετ για να διαχειριστούμε τα αρχεία μας οποιαδήποτε στιγμή (Χατζημιχαηλίδης 2019).

Κόστος. Για τον καθημερινό χρήστη τέτοιες υπηρεσίες μπορεί να παρέχονται δωρεάν, ωστόσο υπάρχουν συνδρομητικά πακέτα στα οποία ο χρήστης καλείται να πληρώσει αναλογικά μόνο για ό,τι χρησιμοποιεί.

Χωρητικότητα. Η χωρητικότητα είναι απεριόριστη, έτσι δεν χρειάζεται να «σπαταλάμε» την μνήμη του υπολογιστή μας.

Update και Backup. Όλες οι cloud εφαρμογές υποστηρίζουν τακτικά updates και αυτόματη ανάκτηση δεδομένων.

Ασφάλεια και Αξιοπιστία. Οι εταιρείες που ασχολούνται με Cloud υπηρεσίες χρησιμοποιούν πρωτόκολλα ασφάλειας και διαθέτουν την υποδομή ώστε να βεβαιώνουν για απόλυτη ασφάλεια των δεδομένων. Οι servers των εταιρειών αυτών λειτουργούν ομαλά στο 99,9% των περιπτώσεων.

2.6 Τεχνικές Οπτικοποίησης δεδομένων

Παρόμοια, άλλη τεχνολογική εξέλιξη που καλούμαστε να υιοθετήσουμε είναι οι οπτικοποίηση δεδομένων και πληροφοριών ως εργαλείο ερμηνείας των αποτελεσμάτων. Τα δεδομένα απεικονίζονται με άλλον τρόπο, διαφορετικό, που συχνά αποκαλύπτει νέες συσχετίσεις και έννοιες και εξάγονται συμπεράσματα που πριν μπορεί να μην γίνονταν ποτέ αντιληπτά. Αντίστροφα για την οπτικοποίηση πληροφοριών πρέπει πρώτα να αναλύσουμε τα δεδομένα διεξοδικά για να εξάγουμε συμπεράσματα και μετά αυτά να οπτικοποιηθούν.

Η οπτικοποίηση αναφέρεται στην αναπαράσταση δεδομένων και χρησιμοποιεί δυναμικές εφαρμογές όπως γραφικά, κίνηση, τρισδιάστατες απεικονίσεις και άλλα εργαλεία πολυμέσων. Τέτοιες εφαρμογές είναι τα bar charts, scatter graphs, pies, κλπ, ή και εικόνες τόσο στατικές αλλά και δυναμικές που επιτρέπουν αλληλεπίδραση με το χρήστη. Συνήθως η επιλογή του μέσου εξαρτάται από το σκοπό για τον οποίο προορίζεται η παρουσίαση. Έτσι η οπτικοποίηση δεδομένων μπορεί να έχει εκπαιδευτικό χαρακτήρα, επιχειρηματικό, επιστημονικό ή ακόμα να πρόκειται για αποτύπωση καλλιτεχνικής έκφρασης (Guernica, Picasso).

2.6.1 Πλεονεκτήματα και Μειονεκτήματα Οπτικοποίησης

Η οπτικοποίηση των δεδομένων είναι πολύτιμο εργαλείο για την εξαγωγή συμπερασμάτων αφού οι σύγχρονες τεχνικές της μας προσφέρουν σημαντικά πλεονεκτήματα:

- ✓ Απεικονίζουν πληροφορία με μια ματιά.
- ✓ Αποκαλύπτουν τάσεις, ακραίες τιμές και συστάδες δεδομένων.
- ✓ Μπορούν να χειρίζονται με ευκολία μεγάλους όγκους δεδομένων.

- ✓ Η πληροφορία αναπαρίσταται με αντικειμενικό τρόπο (συνήθως η περιγραφική αναφορά εμπεριέχει έντονα το στοιχείο της υποκειμενικότητας).
- ✓ Επιτρέπουν στον χρήστη να αλληλοεπιδρά και να κάνει διαφορετικές συσχετίσεις.
- ✓ Αποκαλύπτουν κρυμμένη πληροφορία, και τα αποτελέσματα μπορούν να αναλυθούν με άλλα μέσα περαιτέρω.

Από την άλλη πλευρά αν τα στοιχεία στα οποία βασίζεται η οπτικοποίηση δεν είναι αξιόπιστα, η πληροφορία που εξάγεται μπορεί να επιφέρει σύγχυση. Επιπλέον οι περισσότεροι χρήστες δεν είναι ακόμη εξοικειωμένοι με τις οπτικές αναπαραστάσεις και συχνά η οπτικοποίηση δεδομένων είναι «φτωχή» και οδηγεί σε εσφαλμένα συμπεράσματα.

Είναι γεγονός ότι οι δυνατότητες της οπτικοποίησης είναι σημαντικές και οι τεχνικές της αποτελούν χρήσιμο εργαλείο για τον εντοπισμό και την αναγνώριση δομών και ιδιοτήτων σε ένα σύνολο δεδομένων. Είναι βέβαιο ότι στο μέλλον η οπτικοποίηση θα γνωρίσει μεγάλη ανάπτυξη αφού οι τρέχουσες τάσεις απαιτούν εξοικείωση σε νέες μεθόδους μέσω των οποίων μαθαίνουμε εύκολα , γρήγορα και προπάντων ευχάριστα.

ΚΕΦΑΛΑΙΟ 3^{Ο} Εισαγωγή στην Γλώσσα Προγραμματισμού R

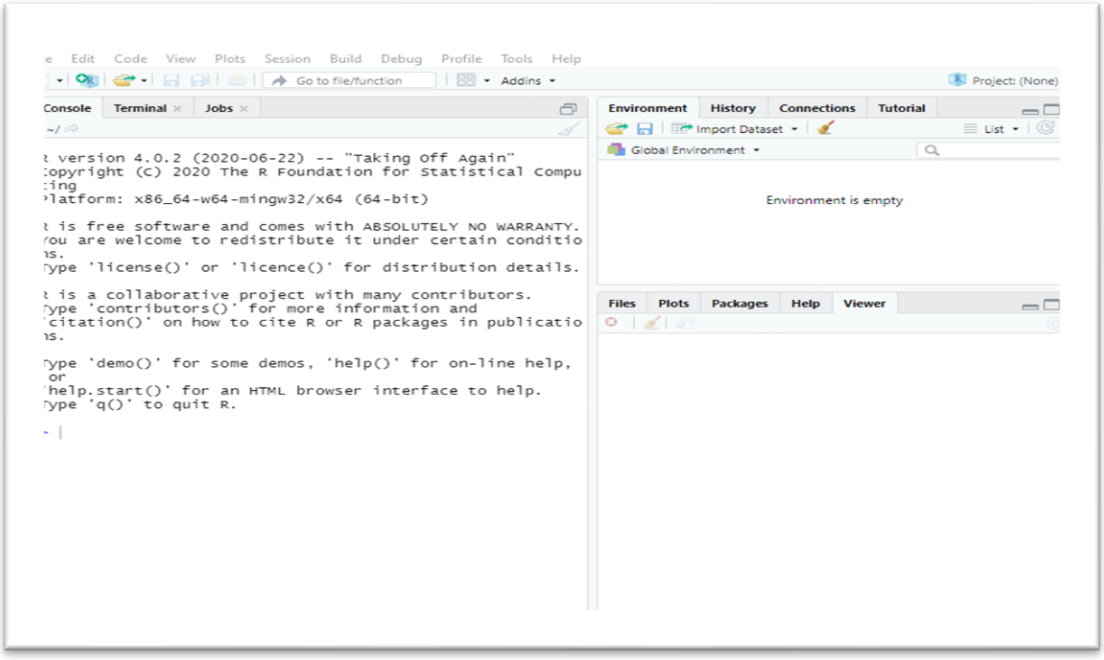
3.1 Εισαγωγή

Η R είναι μια διερμηνευόμενη γλώσσα προγραμματισμού και στατιστικής ανοιχτού κώδικα (εφαρμόζει διάλεκτο της γλώσσας S), και αποτελεί πρότυπο για τη Στατιστική και την Επιστήμη Δεδομένων. Αναπτύχθηκε από στατιστικολόγους και χρησιμοποιείται για την εφαρμογή στατιστικών αναλύσεων με διάφορες στατιστικές τεχνικές και αλγόριθμους μηχανικής μάθησης. Γνωρίζει ευρεία αποδοχή, και υιοθετείται διαρκώς από μεγάλες εταιρείες που ενδιαφέρονται για στατιστική μοντελοποίηση, για ανάλυση τάσεων και προτύπων, ή πραγματοποίηση προβλέψεων όπως το Facebook, η Google η Airbnb και πολλές άλλες.

3.2 Το περιβάλλον εργασίας της R και του RStudio

Μπορούμε να χρησιμοποιήσουμε την R αμέσως όπως είναι, αλλά προτιμάται συνήθως και χρησιμοποιείται η διεπαφή του RStudio, (περιβάλλον ανάπτυξης λογισμικού IDE – Integrated Development Environment), που ενσωματώνει το R με πρακτικό τρόπο και έχει οργανωμένη διάταξη αλλά και διάφορες πρόσθετες επιλογές.

Στο παρακάτω σχήμα φαίνεται το περιβάλλον εργασίας που περιλαμβάνει τις εξής 3 βασικές περιοχές :



Εικόνα 3.1 Το περιβάλλον εργασίας R

Αριστερά: **παράθυρο κονσόλας ή παράθυρο εντολών**. Είναι το πιο σημαντικό παράθυρο, επειδή εδώ τρέχει η R. Οι εντολές εισάγονται σε μία γραμμή κάθε φορά μετά το σύμβολο >. Στην συνέχεια η R θα εκτελέσει την εντολή και η έξοδος θα εκτυπωθεί στην επόμενη γραμμή.

Πάνω δεξιά: **χώρος εργασίας / ιστορικό.** Στο παράθυρο αυτό αποθηκεύεται το ιστορικό του κώδικα που έχουμε εκτελέσει και μπορούμε να δούμε ποια δεδομένα και ποιες τιμές έχει κρατήσει η R στη μνήμη της. Η εμφάνιση και η επεξεργασία των τιμών γίνεται κάνοντας «κλικ» πάνω στις μεταβλητές.

Κάτω δεξιά: **αρχεία / γραφικές παραστάσεις / πακέτα / βοήθεια**. Από εδώ μπορούμε να εξερευνήσουμε τα αρχεία μας, να εμφανίσουμε γραφικές παραστάσεις, να φορτώσουμε όλα τα εγκαταστημένα και διαθέσιμα πακέτα ή να χρησιμοποιήσουμε τη λειτουργία της βοήθειας.

3.3 Γενική σύνταξη R

Η γλώσσα προγραμματισμού R, στην πραγματικότητα είναι μία εύκολη γλώσσα έκφρασης που έχει πολύ απλή σύνταξη. Για την ονομασία των αντικειμένων μπορούμε να χρησιμοποιήσουμε :

τα κεφαλαία και πεζά λατινικά γράμματα A – z. Η γλώσσα είναι case sensitivity, δηλαδή γίνεται διάκριση μεταξύ πεζών και κεφαλαίων, (τα name και Name θεωρούνται διαφορετικά στη γλώσσα R), ενώ τα ονόματα μπορεί να έχουν απεριόριστο μήκος.

την τελεία «.» η οποία θεωρείται ως γράμμα στην αρχή του ονόματος, και

τα αριθμητικά ψηφία (0 – 9). τα οποία μπορούν να χρησιμοποιηθούν σε οποιαδήποτε θέση πλην της αρχικής, γιατί αν ένα όνομα ξεκινά με τελεία «.» και ο δεύτερος χαρακτήρας είναι ψηφίο, τότε θα θεωρηθεί ως υποδιαστολή δεκαδικού αριθμού (δηλαδή τα .35 και .1 νοούνται ως οι δεκαδικοί 0,36 και 0,1 αντίστοιχα).

Δεν επιτρέπεται το κενό ή άλλοι χαρακτήρες στα ονόματα των αντικειμένων, ενώ για τον διαχωρισμό των λέξεων μεταξύ τους χρησιμοποιούμε τα αλφαριθμητικά σύμβολα τελεία «.» , υπογράμμιση «_», ή χωρίς κενό. (Αν θέλουμε να χρησιμοποιήσουμε τον κενό χαρακτήρα η ονομασία πρέπει να εμπεριέχεται σε εισαγωγικά. Π.χ.

I_am_learning_programming

at.the.university.of.Patras

andIamExpertInR

" Welcome to R "

Επιπλέον, συγκεκριμένοι χαρακτήρες πρέπει να αποφεύγονται αφού χρησιμοποιούνται ήδη από το πρόγραμμα. Π.χ. t (ανάστροφο), F (False), T (True), diff (πρώτες διαφορές), range, for, function, if, in, next, repeat, return κλπ.

3.3.1. Μεταβλητές

Για να αποδώσουμε τιμές στις μεταβλητές και γενικότερα για να εισάγουμε δεδομένα στο πρόγραμμα, χρησιμοποιούμε τον τελεστή "<-", που "μοιάζει" σαν βέλος και έχει κατεύθυνση από δεξιά προς τα αριστερά. Η τιμή που δίνεται στα δεξιά αντιστοιχεί στη μεταβλητή που βρίσκεται στα αριστερά. πχ.

x <- 7.2 και διαβάζεται:

η μεταβλητή "**x** παίρνει (gets) την τιμή 7.2"

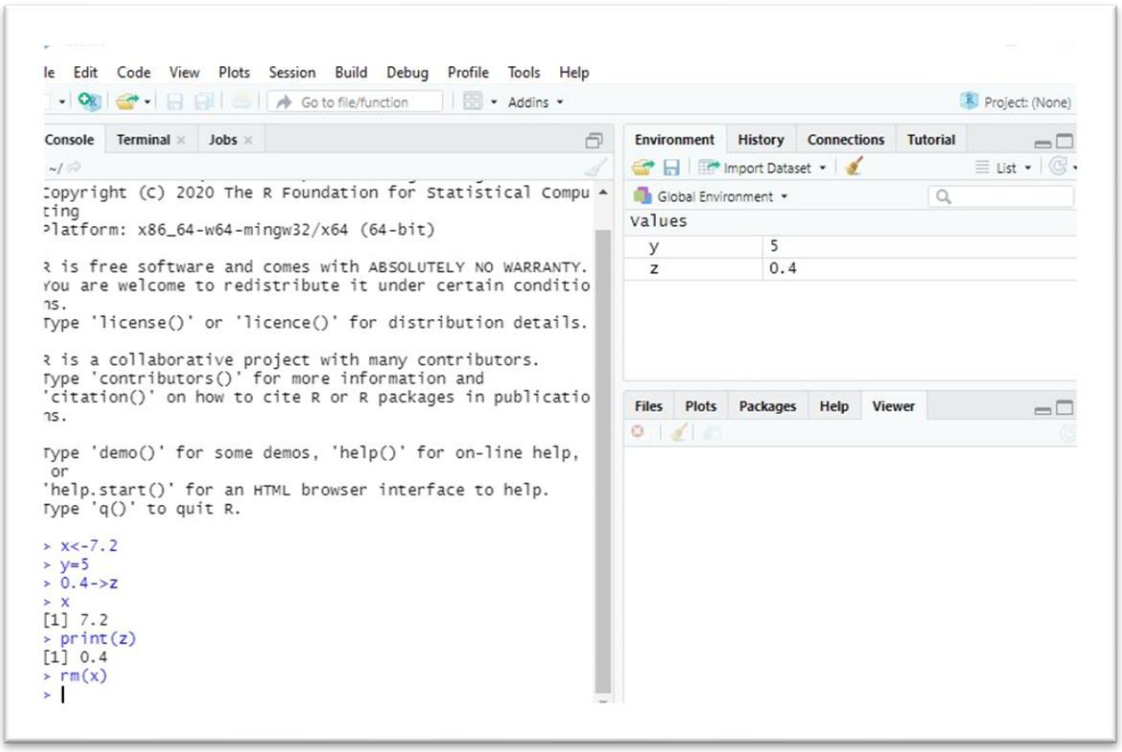
Άλλοι τελεστές ανάθεσης για την εισαγωγή τιμών σε μεταβλητές είναι το ίσον "=", το διπλό βέλος "<<-"ή το αντίστροφο βέλος "->" πχ.

x = 7.2

x <<- 7.2

7.2 -> x

Όλες οι παραπάνω εντολές είναι ισοδύναμες και οι τιμές αποδίδονται στις μεταβλητές χωρίς να τυπωθεί το αποτέλεσμα. Οι μεταβλητές με τις τιμές τους εμφανίζονται στο παράθυρο του χώρου εργασίας (επάνω δεξιά), ή μπορούμε να τις εμφανίσουμε καλώντας τις μεταβλητές με το όνομά τους ή με την εντολή print (). Κάθε μεταβλητή καταλαμβάνει χώρο στην μνήμη του υπολογιστή. Για να αφαιρέσουμε μια μεταβλητή από τον χώρο εργασίας, μπορούμε να χρησιμοποιήσουμε την συνάρτηση rm (). Σχετικά παραδείγματα στο στιγμιότυπο οθόνης που ακολουθεί.



Εικόνα 3.2 Παράδειγμα εφαρμογής εντολών σε R

3.3.1.1 Αντικείμενα και Κλάσεις

Όλες οι μεταβλητές στην R λογίζονται ως αντικείμενα (objects), τα οποία ανήκουν σε μια κλάση (class) που αντιπροσωπεύει τον τύπο τους. Η κλάση στην οποία ανήκουν τα αντικείμενα δεν χρειάζεται να δηλωθεί ρητά γιατί καθορίζεται αυτόματα όταν δώσουμε τιμή στο αντικείμενο. Η R υποστηρίζει τις εξής πέντε κλάσεις αντικειμένων:

- χαρακτήρας (character)

> a <- “Welcome to R!”

> class(a)

[1] “character”

● αριθμητικός – πραγματικοί αριθμοί (numeric)

```
> b <- 3.14

> class(b)

[1] “numeric”
```

● ακέραιος (integer)

```
> c <- 15L

(με το L το 15 αποθηκεύεται 15 ως ακέραιο)

> class(c)

[1] “integer”
```

● σύνθετος (complex)

```
> d <- 5 + 2i

> class(d)

[1] “complex”
```

● λογικός (logical)

```
> f <- FALSE

> class(f)

[1] “logical”
```

3.3.2 Διανύσματα

Όπως ήδη αναφέραμε, οι οντότητες που χρησιμοποιεί το R είναι γνωστές ως αντικείμενα. Όλα στο R είναι αντικείμενα. Η βασικότερη δομή δεδομένων που υποστηρίζει η R είναι το διάνυσμα (vector). Ένα διάνυσμα μπορεί να περιέχει αντικείμενα μόνο του ίδιου τύπου, αριθμητικές ή λογικές τιμές ή και συμβολοσειρές (strings), που να ανήκουν στην ίδια κλάση. Σε περίπτωση που

αυτό δεν ισχύει, τότε η R θα κάνει αυτόματα μετατροπή, ώστε όλα τα αντικείμενα να ανήκουν στην ίδια κλάση. Για να δημιουργήσουμε ένα διάνυσμα μπορούμε να χρησιμοποιήσουμε είτε τη συνάρτηση συνένωσης c, είτε τη συνάρτηση vector.

Κάθε αντικείμενο έχει συγκεκριμένες ιδιότητες/χαρακτηριστικά:

- name ()
- class () , τι τύπος είναι
- length (), πόσα στοιχεία περιέχει
- και άλλες ιδιότητες που ορίζει ο χρήστης

3.3.2.1 Ορισμός διανύσματος

Έστω ότι θέλουμε να δηλώσουμε τις ηλικίες των μαθητών ενός τμήματος. Χρησιμοποιούμε την συνάρτηση c και εν συνεχεία δίνουμε τα στοιχεία του διανύσματος. *Η απουσία τιμής σε ένα διάνυσμα δηλώνεται με το NA, ενώ ένα διάνυσμα με μήκος 0 δηλώνεται ως NULL.* Η εντολή:

```
> age <-c(18,25,30,32,45,63,70,85)
```

ορίζει το διάνυσμα age με στοιχεία τις οκτώ ηλικίες

Χρησιμοποιώντας στην συνέχεια την εντολή με το όνομα του διανύσματος, το R θα τυπώσει στην οθόνη το διάνυσμα. Δηλαδή:

```
> age
[1] 18 25 30 32 45 63 70 85
```

Χρησιμοποιώντας αγκύλες παίρνουμε επιμέρους συνιστώσες του διανύσματος:

> age [8]

[1] 85

Επιπλέον, σε ένα διάνυσμα μπορούμε να χρησιμοποιήσουμε λογικές εκφράσεις. Η έκφραση age>31 θα εμφανίσει το διάνυσμα με TRUE ή FALSE, ανάλογα με ποιες ηλικίες είναι μεγαλύτερες ή μικρότερες από 31 χρόνια.

[1] FALSE FALSE FALSE TRUE TRUE TRUE TRUE TRUE

Τέλος μπορούμε να αντιστοιχίσουμε τα στοιχεία ενός διανύσματος με χαρακτήρες και να αναφέρονται ονομαστικά. π.χ.

age <-c(18,25,30,32,45,63,70,85)

>students.names<-c("Angela","Kon","Bill","Jack","El","Meg","Sam","Lu")

>names(age)<-students.names

>age

Angela	Kon	Bill	Jack	El	Meg	Sam	Lu
18	25	30	32	45	63	70	85

Το μήκος του διανύσματος δίνεται από την εντολή:

> length(age) θα τυπώσει [1] 8

Το είδος του διανύσματος δίνεται από την εντολή:

> class(age) θα τυπώσει [1] "numeric"

3.3.2.2. Μετατροπή διανύσματος σε πίνακα

Μπορούμε εύκολα να μετατρέψουμε ένα διάνυσμα σε πίνακα με την εντολή dim. Για να μετατρέψουμε το διάνυσμα age του παραπάνω παραδείγματος σε πίνακα 2x4 γράφουμε:

```
age <-c(18,25,30,32,45,63,70,85)

> dim(age) <- c(2,4)

> age

     [,1]  [,2]  [,3]   [,4]

[1,]   18   30   45   70

[2,]   25   32   63   85
```

Αρχικά παρατηρούμε ότι η μετατροπή του διανύσματος σε πίνακα έγινε κατά στήλες. Δηλαδή τα πρώτα δύο στοιχεία του διανύσματος αποτελούν την πρώτη στήλη, τα επόμενα δύο τη δεύτερη στήλη κοκ. Επιπλέον μπορούμε να εμφανίσουμε τα αποτελέσματα ανά γραμμή ή ανά στήλη πληκτρολογώντας:

```
age [2, ]

[1] 25 32 63 85
```

και το R θα εμφανίσει τα στοιχεία της δεύτερης γραμμής του πίνακα, ή

```
> age [ ,4]

[1] 70 85
```

και θα εμφανίσει τα στοιχεία της τέταρτης στήλης του πίνακα κοκ.

Η επαναφορά στην αρχική κατάσταση του διανύσματος (ακύρωση του πίνακα), επιτυγχάνεται με την εντολή:`> dim(age) <- NULL`

3.3.2.3 Πίνακες

Ένας πίνακας (matrix) δεν είναι τίποτα άλλο παρά δισδιάστατα διανύσματα των οποίων τόσο οι γραμμές όσο και οι στήλες πρέπει να περιέχουν αντικείμενα της ίδιας κλάσης. Οι πίνακες είναι μία ειδική δομή, η οποία έχει ως επιπλέον ιδιότητα (attribute), τη διάσταση (dimension) και για τον ορισμό της θα χρησιμοποιήσουμε την εντολή matrix:

```
age=matrix(data=c(18,25,30,32,45,63,70,85),nrow=2)

age

     [,1] [,2] [,3] [,4]

[1,]   18   30   45   70

[2,]   25   32   63   85
```

Η χρήση του ορίσματος data καθορίζει τα νούμερα που θα εισαχθούν στον πίνακα. (*Εναλλακτικά μπορούμε να δημιουργήσουμε πρώτα το διάνυσμα age και στην συνέχεια να το μετατρέψουμε σε πίνακα όπως είδαμε προηγουμένως*). Για τον προσδιορισμό των στηλών χρησιμοποιούμε το χαρακτηριστικό ncol, ενώ για τις στήλες χρησιμοποιούμε το όρισμα nrow.

Άλλος τρόπος για να δημιουργήσουμε πίνακα είναι με την σύνδεση διανυσμάτων είτε κατά γραμμές με χρήση της συνάρτησης rbind είτε κατά στήλες με χρήση της συνάρτησης cbind.

3.3.2.4 Λίστες

Άλλη δομή δεδομένων που υποστηρίζει η R είναι η λίστα. Μια λίστα, όπως και το διάνυσμα, είναι ένα σύνολο από αντικείμενα, που όμως μπορούν να ανήκουν σε διαφορετική κλάση. Εδώ, η μετατροπή των αντικειμένων στην ίδια κλάση επιτυγχάνεται με ρητή δήλωση με τις συναρτήσεις as.integer, as.numeric, as.logical κλπ. Για τη δημιουργία λίστας χρησιμοποιούμε τη συνάρτηση list. π.χ.

```
x <- list("Angela", 1978, TRUE, 42)
```

Αν καλέσουμε την λίστα x θα εμφανιστούν τα επιμέρους στοιχεία της :

```
[[1]]          [[2]]          [[3]]          [[4]]

[1] "Angela"   [1] 1978       [1] TRUE       42
```

Με την εντολή class(x[[1,2 κλπ]]) εμφανίζονται οι κλάσεις που ανήκουν τα αντικείμενα της λίστας:

```
> class(x[[1]])      > class(x[[2]])      > class(x[[3]])      > class(x[[4]])

[1] "character"      [1] "numeric"        [1] "logical"        [1] "numeric"
```

3.3.3. Συναρτήσεις

Οι συναρτήσεις είναι αναπόσπαστο κομμάτι της R. Τις χρησιμοποιούμε για αυτοματοποιημένες διαδικασίες και παρέχουν μεγάλη ευκολία στον χρήστη. Εκτός των έτοιμων πακέτων που διαθέτει, η R επιτρέπει στον χρήστη να ορίσει τις δικές του συναρτήσεις. Για τον προγραμματισμό νέων συναρτήσεων χρησιμοποιείται η δεσμευμένη λέξη “function”. Οι συναρτήσεις που δημιουργούνται αποθηκεύονται ως αντικείμενα στα πακέτα της R. π.χ.

```
emvado= function(mikos,platos)    # καθορίζουμε τη συνάρτηση emvado
                                  με ορίσματα τα mikos,platos
{ emv=mikos*platos                # καθορίζουμε τι πρέπει να κάνει η συνάρτηση
return(emv) }                     # όταν την καλέσουμε
emvado(mikos=2,platos=3)          # καλούμε τη συνάρτηση με ορίσματα 2 και 3
[1] 6                             # εμφανίζεται το αποτέλεσμα
```

Μερικές από τις προκαθορισμένες συναρτήσεις που συναντάμε στην R είναι :

operator	operation	example
log2(x)	Logarithms base 2 of x	log2(10) [1] 3.321828
log10(x)	Logarithms base 10 of x	log10(34) [1] 1.53479
exp(x)	Exponential of x	exp(10) [1] 22026.7
cos(x)	Cosine of x	cos(pi*45) [1] -1
sin(x)	Sine of x	sin(90) [1] 0.8939967
tan(x)	Tangent of x	tan(180) [1] 1.33869
acos(x)	Arc-cosine of x	acos(-1) [1] 3.141593
asin(x)	Arc-sine of x	asin(-0.1) [1] -0.1001674
atan(x)	Arc-tangent of x	atan(90) [1] 1.559686
abs(x)	Absolute value of x	abs(-45) [1] 45
sqrt(x)	Squared root of x	sqrt(16)

Εικόνα 3.3 Προκαθορισμένες Συναρτήσεις R (πηγή: SEnDIng Online, Statistics for Data Science)

3.3.4. Πακέτα – Βιβλιοθήκες

Τα πακέτα της R είναι συλλογές από συναρτήσεις δεδομένων και κώδικα, οργανωμένα σε ένα κατάλογο που ονομάζεται βιβλιοθήκη και μας βοηθούν να κάνουμε στατιστικές αναλύσεις. Με την τυπική εγκατάσταση του προγράμματος, εγκαθίστανται και τα περισσότερα πακέτα της R. Για να δούμε μια λίστα με όλα τα εγκατεστημένα πακέτα αρκεί να πληκτρολογήσουμε την εντολή installed.packages (). Επιπλέον πακέτα μπορούν να εγκατασταθούν από διάφορα αποθετήρια ή από το CRAN, (https://cran.r-project.org/web/packages/), που διαθέτει έναν πλήρη κατάλογο πακέτων.

ΚΕΦΑΛΑΙΟ 4^{Ο} Μεθοδολογία Έρευνας με το πρόγραμμα R

Στο παρόν κεφάλαιο παρουσιάζεται εφαρμογή αλγορίθμων ταξινόμησης και ομαδοποίησης – συσταδοποίησης με το στατιστικό πακέτο της R σε έτοιμα δεδομένα μελετών.

Study case I: European Protein Consumption

Στην παρούσα μελέτη περίπτωσης θα εξετάσουμε το ποσοστό κατανάλωσης πρωτεϊνών 25 Ευρωπαϊκών χωρών (n = 25), από εννέα κύριες πηγές τροφίμων (p = 9), με χρήση ανάλυσης συστάδων (cluster analysis).

Εισαγωγή Δεδομένων στην R

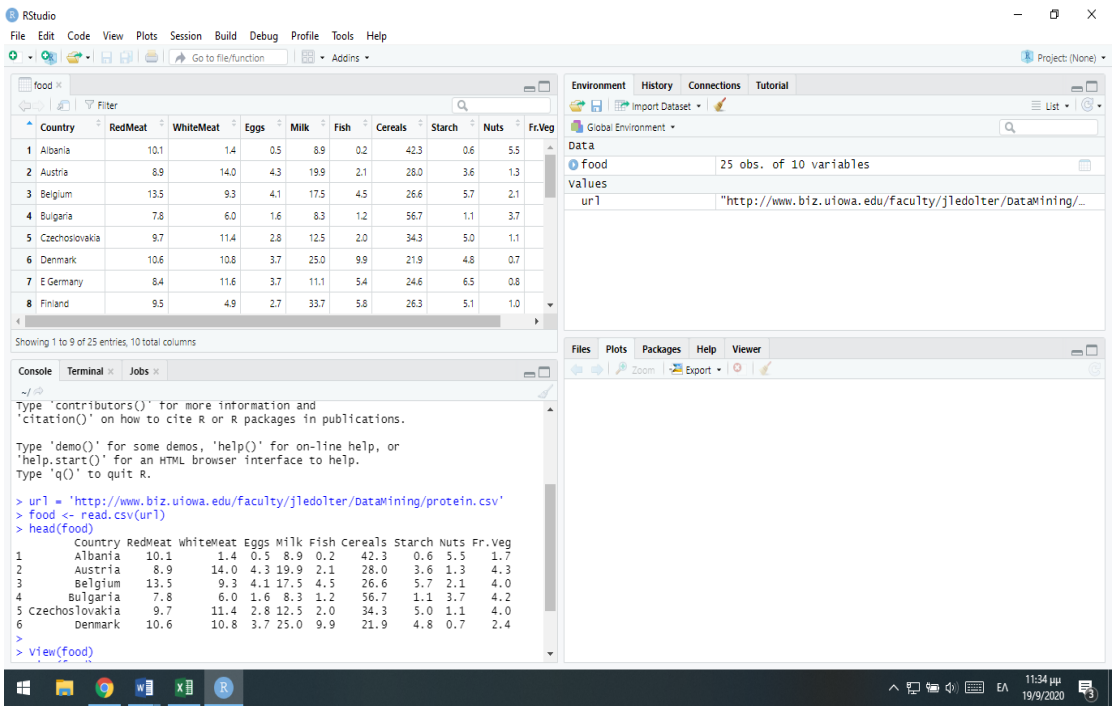
Η εισαγωγή των δεδομένων στο πρόγραμμα γίνεται με την εντολή read.csv(url), όπου το όρισμα url είναι η διεύθυνση στην οποία βρίσκεται το αρχείο. Το αρχείο είναι τύπου csv που σημαίνει ότι οι τιμές είναι αποθηκευμένες σε μία γραμμή οριζόντια και διαχωρισμένες με κόμμα.

```
url = 'http://www.biz.uiowa.edu/faculty/jledolter/DataMining/protein.csv'

food <- read.csv(url)

head(food)
```

Η head προβάλει τις πρώτες γραμμές των δεδομένων (εδώ μεταβλητή food) για μια συνοπτική επισκόπηση:

```
    Country RedMeat WhiteMeat Eggs Milk Fish Cereals Starch Nuts Fr.Veg
1   Albania    10.1      1.4  0.5  8.9  0.2    42.3    0.6  5.5    1.7
2   Austria     8.9     14.0  4.3 19.9  2.1    28.0    3.6  1.3    4.3
3   Belgium    13.5      9.3  4.1 17.5  4.5    26.6    5.7  2.1    4.0
4   Bulgaria    7.8      6.0  1.6  8.3  1.2    56.7    1.1  3.7    4.2
5   Czech/kia   9.7     11.4  2.8 12.5  2.0    34.3    5.0  1.1    4.0
6   Denmark    10.6     10.8  3.7 25.0  9.9    21.9    4.8  0.7    2.4
```

Στο παράθυρο του χώρου εργασίας (επάνω δεξιά), βλέπουμε ότι η R έχει κρατήσει στη μνήμη της την μεταβλητή food που περιέχει 25 παρατηρήσεις από 10 στοιχεία. Κάνοντας κλικ επάνω στην μεταβλητή εμφανίζεται ο πλήρης κατάλογος με όλα τα δεδομένα.



Ομαδοποίηση με K-Means

Στην συνέχεια, θα ομαδοποιήσουμε (clustering) τις ετερογενείς παρατηρήσεις σε k=3 υποσύνολα (clusters) με χρήση της μεθόδου K-means βάσει του μέτρου ομοιότητας των 2 μεταβλητών, δηλαδή κόκκινου και λευκού κρέατος (p = 2). Οπότε στη συνάρτηση kmeans δίνουμε ως όρισμα τα δεδομένα (μόνο τις δύο στήλες που αναφέραμε), ορίζουμε τον αριθμό ομάδων σε 3 στο όρισμα centers και θέτουμε σε 10 το όρισμα nstart (επαναλήψεις του τυχαίου αρχικού ορισμού των κέντρων). Χρησιμοποιούμε την εντολή set.seed για αρχικοποίηση της τυχαίας επιλογής αριθμών κατά την αρχική δημιουργία των συστάδων.

```
set.seed(123456789)

grpMeat <- kmeans(food[,c("WhiteMeat","RedMeat")], centers=3, nstart=10)

grpMeat
```

Στην συνέχεια παρατηρούμε ότι για την μεταβλητή grpMeat που είναι η έξοδος της συνάρτησης kmeans δημιουργούνται 3 συστάδες (οι 1, 2 και 3), με πλήθος παρατηρήσεων 8, 12 και 5 αντίστοιχα. Έτσι έχουμε τα κέντρα των συστάδων:

```
     WhiteMeat   RedMeat

1    12.062      8.838

2    4.658       8.258

3    9.000       15.180
```

Ενώ αναλυτικά το διατεταγμένο σύνολο τιμών (διάνυσμα) τιμών είναι:

```
[1] 2 1 3 2 1 1 1 2 3 2 1 3 2 1 2 1 2 2 2 2 3 3 2 1 2
```

Το άθροισμα τετραγώνων που προκύπτει ανά συστάδα είναι :

```
[1] 39.46    69.86    35.67
```

Ενώ τα μέσα τετράγωνα που προκύπτουν από το άθροισμα τετραγώνων είναι:

```
(between_SS / total_SS =  75.7 %)
```

Και η λίστα των διαθέσιμων συνδυασμών, αποτελείται από παρακάτω 9 στοιχεία:

```
[1] "cluster"      "centers"      "totss"        "withinss"
[5] "tot.withinss" "betweenss"    "size"         "iter"
[9] "ifault"
```

Στην συνέχεια θα χρησιμοποιήσω την εντολή data.frame με δύο ορίσματα, για να δημιουργήσουμε πίνακα matrix που θα αποτελείται από δύο στήλες, όπου κάθε στήλη θα αποτελεί και ένα διάνυσμα με τα δικά του χαρακτηριστικά και με δικό του όνομα.

```
o=order(grpMeat$cluster)

data.frame(food$Country[o],grpMeat$cluster[o])
```

Έπειτα, θα τρέξω το πρόγραμμα για να διαπιστώσω πώς τελικά ταξινομούνται τα υποσύνολα που δημιουργήθηκαν προηγουμένως κατά την ομαδοποίηση. Έτσι εμφανίζονται οι 25 χώρες και σε ποιο cluster εμπεριέχεται η κάθε μια.

```
    food.Country.o. grpMeat.cluster.o.
1          Austria                   1
2   Czechoslovakia                   1
3          Denmark                   1
4        E Germany                   1
5          Hungary                   1
6      Netherlands                   1
7           Poland                   1
8        W Germany                   1
```

9	Albania	2
10	Bulgaria	2
11	Finland	2
12	Greece	2
13	Italy	2
14	Norway	2
15	Portugal	2
16	Romania	2
17	Spain	2
18	Sweden	2
19	USSR	2
20	Yugoslavia	2
21	Belgium	3
22	France	3
23	Ireland	3
24	Switzerland	3
25	UK	3

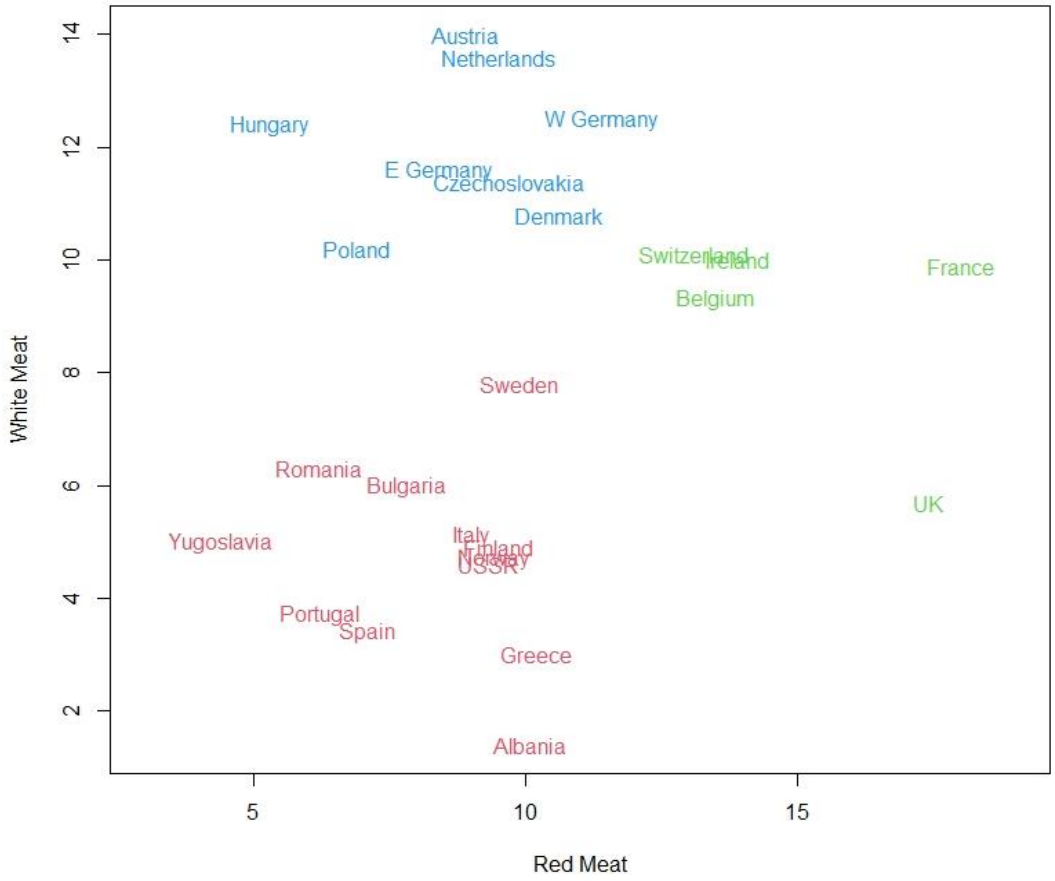
Παρατηρούμε ότι οι χώρες που βρίσκονται σε κοντινή απόσταση τείνουν να συγκεντρώνονται στην ίδια ομάδα.

Τέλος, για να δούμε την γραφική αναπαράσταση της προηγούμενης λύσης ομαδοποίησης, θα σχεδιάσουμε σχεδιάγραμμα διασποράς συστάδων σε κόκκινο και λευκό κρέας.

Για το διάγραμμα θα χρησιμοποιήσουμε την συνάρτηση plot, στην οποία εκτός από τα δεδομένα, δίνουμε ως όρισμα την τιμή type = "n", η οποία δημιουργεί αρχικά έναν κενό καμβά στον οποίο έπειτα μπορούμε να προσθέσουμε σημεία, γραμμές, κείμενο κλπ. Επίσης, ορίζουμε τα ονόματα των αξόνων με τα ορίσματα

xlab, ylab και με το όρισμα xlim=c(3,19) καθορίζουμε το κάτω και άνω όριο που θα εμφανίζεται στον άξονα x. Στη συνέχεια, με τη συνάρτηση text σχεδιάζουμε κείμενο στον καμβά, το οποίο ορίζεται στο όρισμα labels, στις συντεταγμένες που ορίζονται στα ορίσματα x, y, ενώ ορίζουμε και το χρώμα του κάθε σημείου με το όρισμα col (το οποίο αντιστοιχεί στην ομάδα του κάθε σημείου).

```
plot(food$Red, food$White, type="n", , xlab="Red Meat", ylab="White Meat",
xlim=c(3,19))
text(x=food$Red, y=food$White, labels=food$Country,col=grpMeat$cluster+1)
```



Στη συνέχεια, θα κάνουμε την ίδια ανάλυση, αλλά θα χρησιμοποιήσουμε και τις εννέα ομάδες πρωτεϊνών και θα χωρίσουμε τις χώρες σε 7 ομάδες. Σε σχέση με πριν, δίνουμε στην συνάρτηση kmeans όλα τα δεδομένα (εκτός από την πρώτη στήλη που είναι τα ονόματα των χωρών) και ορίζουμε τον αριθμό των ομάδων σε 7.

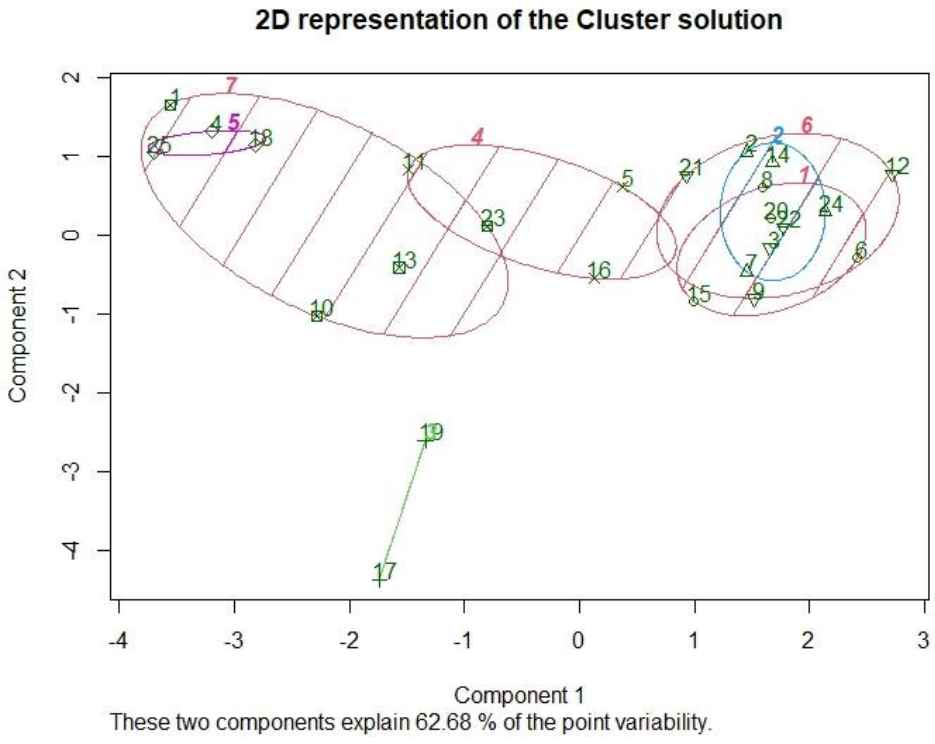
```
set.seed(123456789)
grpProtein <- kmeans(food[,-1], centers=7, nstart=10)
o=order(grpProtein$cluster)
data.frame(food$Country[o],grpProtein$cluster[o])
```

	food.Country.o	. grpProtein.cluster.o.
1	Denmark	1
2	Finland	1
3	Norway	1
4	Sweden	1
5	Austria	2
6	E Germany	2
7	Netherlands	2
8	W Germany	2
9	Portugal	3
10	Spain	3
11	Czechoslovakia	4
12	Hungary	4
13	Poland	4
14	Bulgaria	5
15	Romania	5
16	Yugoslavia	5
17	Belgium	6
18	France	6
19	Ireland	6

20	Switzerland	6
21	UK	6
22	Albania	7
23	Greece	7
24	Italy	7
25	USSR	7

Η γραφική αναπαράσταση της ανάλυσης των επτά ομάδων, αποτυπώνεται στο διάγραμμα συστάδων που ακολουθεί. Επειδή τα δεδομένα έχουν πάνω από δύο μεταβλητές, η προβολή γίνεται σε δύο άξονες που υπολογίζονται από την clusplot με τη μέθοδο PCA. Παρατηρούμε ότι όντως οι χώρες που είναι στην ίδια ομάδα είναι κοντά σε αυτό το διάγραμμα.

```
Library(cluster)

clusplot(food[,-1],grpProtein$cluster,main='2D representation of the Cluster
solution', color=TRUE, shade=TRUE, labels=2, lines=0)
```

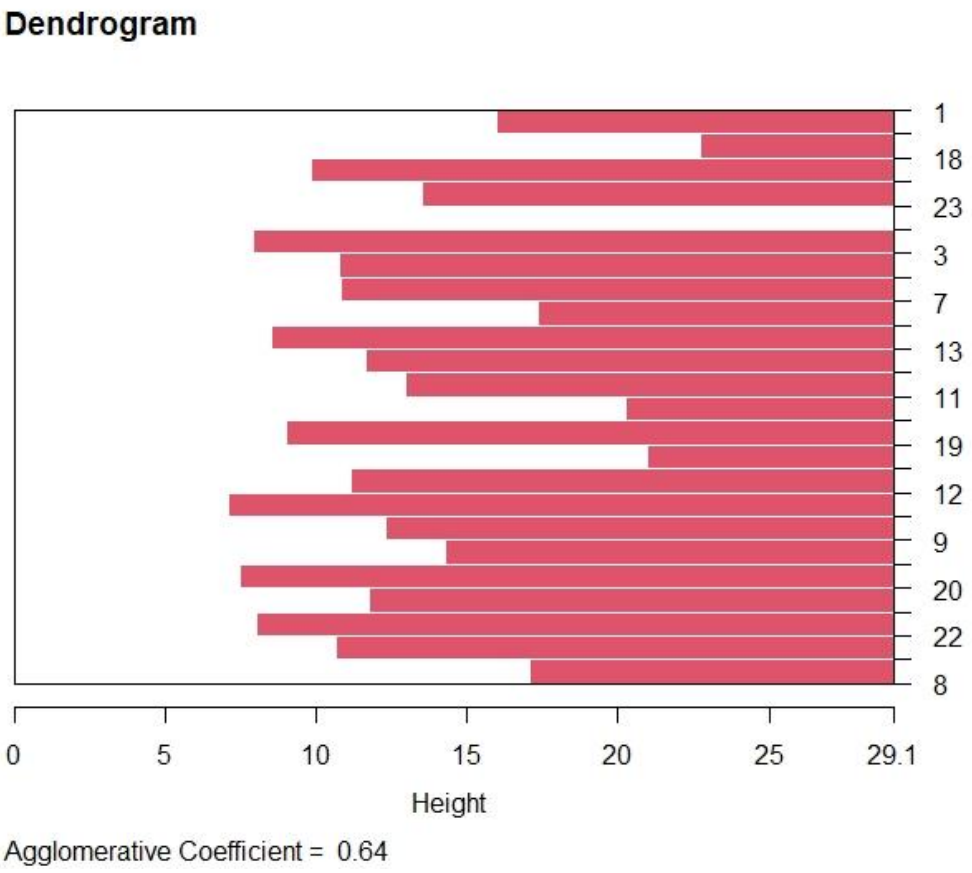


Ιεραρχική Ομαδοποίηση

Μία άλλη προσέγγιση του προβλήματος που μπορούμε να εφαρμόσουμε είναι η Ιεραρχική Ομαδοποίηση. Χρησιμοποιούμε τη συνάρτηση agnes στο σύνολο των δεδομένων, για συγκεντρωτική ιεραρχική ομαδοποίηση. Το όρισμα diss = FALSE υποδεικνύει ότι θα χρησιμοποιούμε έναν πίνακα παρατηρήσεων από μεταβλητές που υπολογίζεται από ανεπεξέργαστα δεδομένα. Το όρισμα metric = "euclidian" δηλώνει ότι θα χρησιμοποιήσουμε την Ευκλείδεια απόσταση.

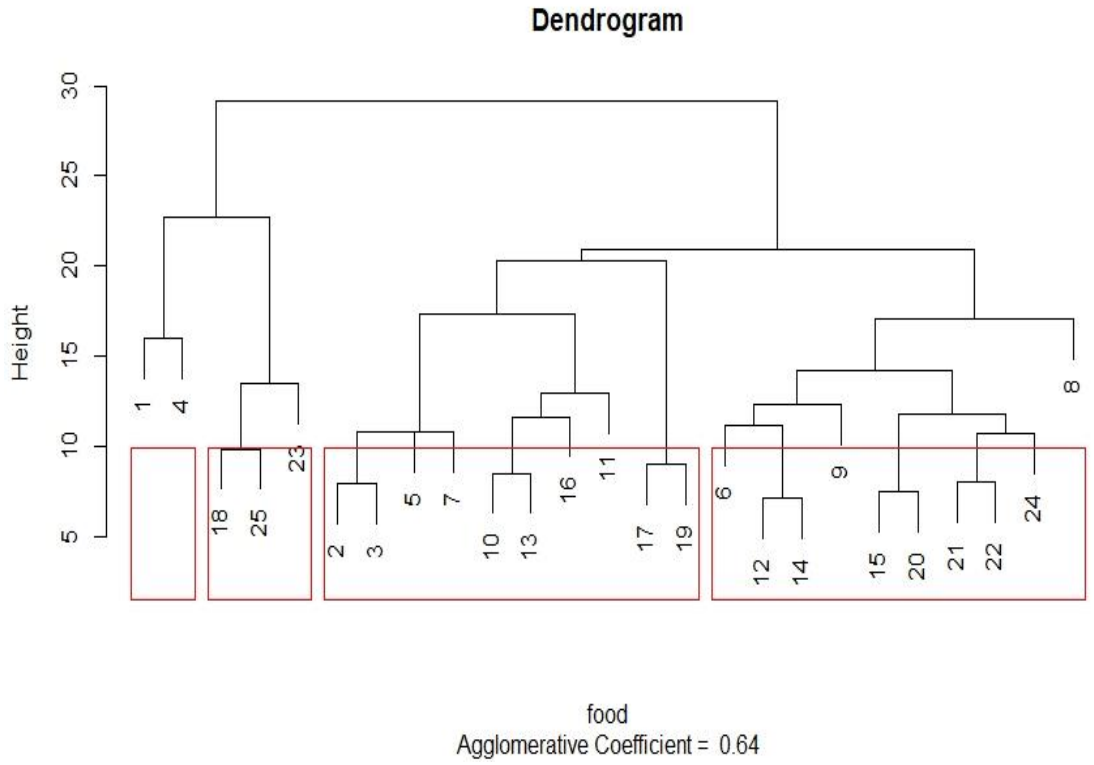
```
foodagg=agnes(food,diss=FALSE,metric="euclidian")

plot(foodagg, main='Dendrogram')
```

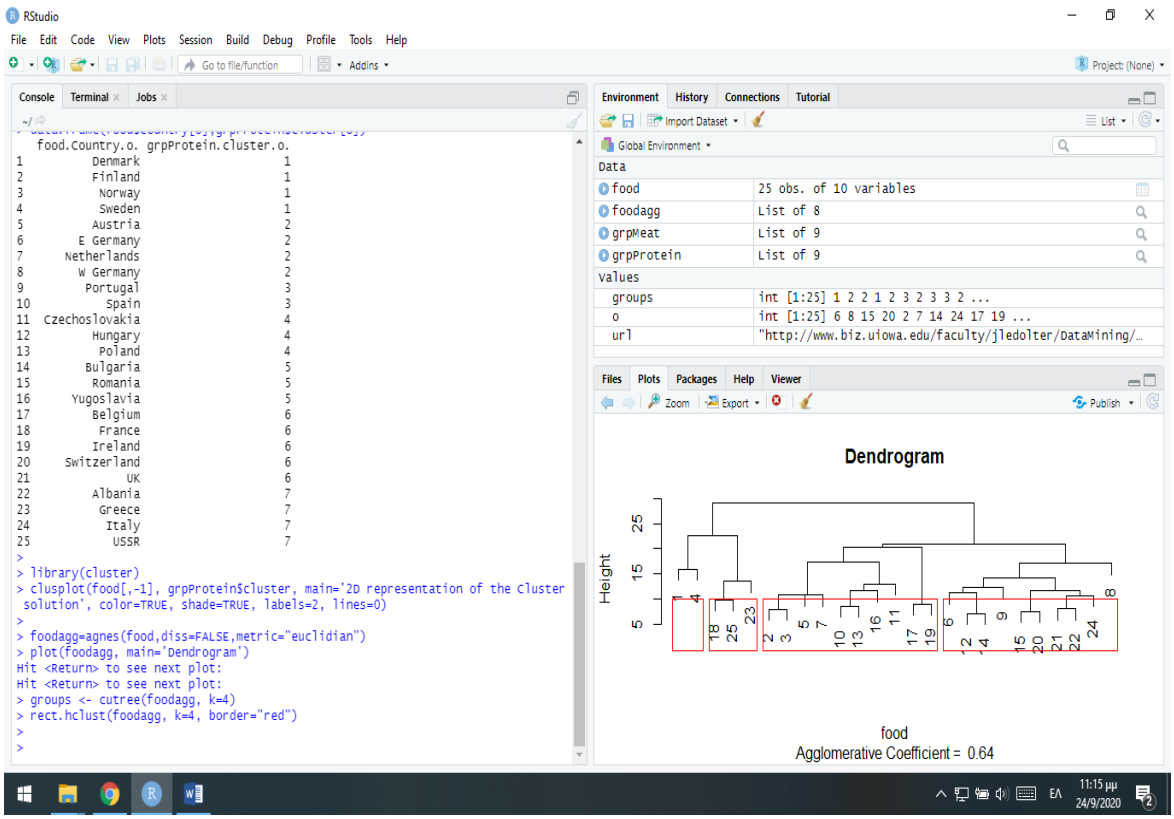
Το διάγραμμα που θα εμφανιστεί είναι δεντροδιάγραμμα της μορφής:



Για να «κόψουμε» το δέντρο σε 4 ομάδες, χρησιμοποιούμε την εντολή cutree, δίνοντας ως ορίσματα το αποτέλεσμα της συσταδοποίησης από την συνάρτηση agnes και ορίζοντας τον αριθμό των συστάδων σε k=4. Η εντολή rect.hclust χρησιμοποιείται για να σχεδιάσει ορθογώνια σχήματα γύρω από τα κλαδιά ενός δενδρογράμματος επισημαίνοντας μάλιστα και τις αντίστοιχες συστάδες. Έχει προηγηθεί το «κόψιμο» του δέντρου σε επίπεδα. Τέλος το όρισμα border καθορίζει το χρώμα του περιγράμματος των ορθογωνίων.

```
groups <- cutree(foodagg, k=4)

rect.hclust(foodagg, k=4, border="red")
```




RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Console Terminal Jobs
food.Country.o. grpProtein.cluster.o.
1 Denmark 1
2 Finland 1
3 Norway 1
4 Sweden 1
5 Austria 2
6 E Germany 2
7 Netherlands 2
8 W Germany 2
9 Portugal 3
10 Spain 3
11 Czechoslovakia 4
12 Hungary 4
13 Poland 4
14 Bulgaria 5
15 Romania 5
16 Yugoslavia 5
17 Belgium 6
18 France 6
19 Ireland 6
20 Switzerland 6
21 UK 6
22 Albania 7
23 Greece 7
24 Italy 7
25 USSR 7
> library(cluster)
> clusplot(food[,-1], grpProtein$cluster, main='2D representation of the Cluster solution', color=TRUE, shade=TRUE, labels=2, lines=0)
> foodagg=agnes(food,diss=FALSE,metric="euclidian")
> plot(foodagg, main='Dendrogram')
Hit <Return> to see next plot:
Hit <Return> to see next plot:
> groups <- cutree(foodagg, k=4)
> rect.hclust(foodagg, k=4, border="red")
Environment History Connections Tutorial
Import Dataset
Global Environment
Data
food 25 obs. of 10 variables
foodagg List of 8
grpMeat List of 9
grpProtein List of 9
Values
groups int [1:25] 1 2 2 1 2 3 2 3 3 2 ...
o int [1:25] 6 8 15 20 2 7 14 24 17 19 ...
url "http://www.biz.uiowa.edu/faculty/jledolter/DataMining/...
Files Plots Packages Help Viewer
Zoom Export Publish
Dendrogram
Height
25
15
5
food
Agglomerative Coefficient = 0.64
11:15 μμ
24/9/2020

Study case II: Social Network Clustering Analysis

Δεδομένα

Σε αυτή την μελέτη περίπτωσης θα χρησιμοποιήσουμε ένα σύνολο δεδομένων από τυχαίο δείγμα 30.000 μαθητών γυμνασίου των ΗΠΑ, οι οποίοι διέθεταν λογαριασμό σε γνωστό κοινωνικό δίκτυο μεταξύ 2006 και 2009.

Από τις 500 πιο συνηθισμένες λέξεις που εμφανίζονται σε όλες τις σελίδες, επιλέχθηκαν μόνο 36 οι οποίες αντιπροσωπεύουν πέντε κατηγορίες ενδιαφερόντων, όπως: εξωσχολικές δραστηριότητες, μόδα, θρησκεία, ειδύλλιο και αντικοινωνική συμπεριφορά. Αυτές οι 36 λέξεις αναφέρονται σε έννοιες όπως: μπάσκετ, ποδόσφαιρο, μουσική, Βίβλος, ψώνια, θάνατος και ναρκωτικά. Το τελικό σύνολο δεδομένων υποδεικνύει, για κάθε άτομο, πόσες φορές η κάθε λέξη εμφανίστηκε στο SNS (Social Network Service) προφίλ του.

```
url='https://raw.githubusercontent.com/brenden17/sklearnlab/master/facebook/sns
data.csv'

teens <- read.csv(url)

head(teens,3)
```

```
   gradyear  gender   age  friends basketball football soccer softball
1   2006       M     18.98     7        0        0       0        0
2   2006       F     18.80     0        0        1       0        0
3   2006       M     18.34     69       0        1       0        0
   volleyball swimming cheerleading baseball tennis sports  cute  sex  sexy
1       0        0           0        0        0      0      0     0    0
2       0        0           0        0        0      0      1     0    0
3       0        0           0        0        0      0      0     0    0
```

```
   hot kissed dance band marching music rock god church jesus bible hair
1   0      0     1     0       0      0    0    0      0      0     0    0
2   0      0     0     0       0      2    2    1      0      0     0    6
3   0      0     0     2       0      1    0    0      0      0     0    0

   dress blonde mall shopping clothes hollister abercrombie die death drunk
drugs
1      0      0     0       0       0          0          0       0     0    0     0
2      4      0     1       0       0          0          0       0     0    0     0
3      0      0     0       0       0          0          0       0     1    0     0
```

Για να δούμε την διάσταση του πίνακα, θα χρησιμοποιήσουμε την συνάρτηση dim. Ο πρώτος αριθμός αντιπροσωπεύει τον αριθμό των σειρών, και ο δεύτερος τον αριθμό των στηλών του πίνακα. Έτσι, τα παραπάνω δεδομένα αποτελούνται από 30000 σειρές και 40 στήλες.

```
dim(teens)
```

```
[1] 30000    40
```

Η συνάρτηση str σημαίνει δομή, και εμφανίζει την εσωτερική δομή ενός συγκεκριμένου αντικειμένου στο πρόγραμμα R. Έτσι, η δομή του αντικειμένου teens είναι:

```
str(teens)
```

```
## 'data.frame':    30000 obs. of  40 variables:
##  $ gradyear    : int  2006 2006 2006 2006 2006 2006 2006 2006 2006 2006 ...
##  $ gender      : Factor w/ 2 levels "F","M": 2 1 2 1 NA 1 1 2 1 1 ...
##  $ age         : num  19 18.8 18.3 18.9 19 ...
```

```
##  $ friends     : int  7 0 69 0 10 142 72 17 52 39 ...
##  $ basketball  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ football    : int  0 1 1 0 0 0 0 0 0 0 ...
##  $ soccer      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ softball    : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ volleyball  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ swimming    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ cheerleading: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ baseball    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ tennis      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ sports      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ cute        : int  0 1 0 1 0 0 0 0 0 1 ...
##  $ sex         : int  0 0 0 0 1 1 0 2 0 0 ...
##  $ sexy        : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ hot         : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ kissed      : int  0 0 0 0 5 0 0 0 0 0 ...
##  $ dance       : int  1 0 0 0 1 0 0 0 0 0 ...
##  $ band        : int  0 0 2 0 1 0 1 0 0 0 ...
##  $ marching    : int  0 0 0 0 0 1 1 0 0 0 ...
##  $ music       : int  0 2 1 0 3 2 0 1 0 1 ...
##  $ rock        : int  0 2 0 1 0 0 0 1 0 1 ...
##  $ god         : int  0 1 0 0 1 0 0 0 0 6 ...
##  $ church      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ jesus       : int  0 0 0 0 0 0 0 0 0 2 ...
##  $ bible       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hair        : int  0 6 0 0 1 0 0 0 0 1 ...
##  $ dress       : int  0 4 0 0 0 1 0 0 0 0 ...
##  $ blonde      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ mall        : int  0 1 0 0 0 0 2 0 0 0 ...
```

```
##  $ shopping    : int  0 0 0 0 2 1 0 0 0 1 ...
##  $ clothes     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hollister   : int  0 0 0 0 0 0 2 0 0 0 ...
##  $ abercrombie : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ die         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ death       : int  0 0 1 0 0 0 0 0 0 0 ...
##  $ drunk       : int  0 0 0 0 1 1 0 0 0 0 ...
##  $ drugs       : int  0 0 0 0 1 0 0 0 0 0 ...
```

Παρατηρούμε, όπως άλλωστε ήταν αναμενόμενο, ότι τα δεδομένα περιλαμβάνουν 30.000 αντικείμενα, δηλαδή εφήβους και 40 μεταβλητές. Οι 4 πρώτες μεταβλητές, ($dradyear, $gender,$age, $friends), δείχνουν τα προσωπικά τους χαρακτηριστικά, και οι 36 επόμενες, δείχνουν τα ενδιαφέροντά τους. Το NA σημαίνει ‘Not Available’ αφού δεν έχει τιμή η μεταβλητή φύλο.

Περίληψη δεδομένων

Μία πολύ χρήσιμη συνάρτηση στο R που χρησιμοποιείται για γρήγορη επισκόπηση δεδομένων, είναι η περίληψη summary (). Χρησιμοποιείται και είναι σημαντική κυρίως όταν έχουμε μεταβλητές με δεκάδες ή εκατοντάδες μετρήσεις. Τα αποτελέσματα που μας δίνει περιλαμβάνουν: την ελάχιστη (Min.) και μέγιστη (Max.) τιμή, το μέσο όρο (Mean), καθώς τη διάμεσο (Median), το πρώτο και τρίτο τεταρτημόριο και το πλήθος των τιμών NA.

Ο πιο σημαντικός δείκτης είναι η διάμεσος (Median), που μας δείχνει την τιμή που είναι μεγαλύτερη από τις μισές μετρήσεις και μικρότερη από τις άλλες μισές. Η διάμεσος του κάθε μισού, που χωρίζει το πρώτο και το τελευταίο τέταρτο των δεδομένων στη μέση, μας δίνει τα λεγόμενα τεταρτημόρια. Η διάμεσος είναι το δεύτερο τεταρτημόριο που χωρίζει το 50% του δείγματος, ενώ το πρώτο τεταρτημόριο (1st quartile), χωρίζει το χαμηλότερο 25% και το τρίτο τεταρτημόριο (3rd quartile) χωρίζει το υψηλότερο 25%.

Παρακάτω γίνεται η εφαρμογή της συνάρτησης και με τις τέσσερις μεταβλητές.

```
summary(teens$age)
```

```
Min.    1st Qu.    Median    Mean    3rd Qu.    Max.     NA's
3.086   16.312     17.287    17.994  18.259   106.927    5086
```

```
summary(teens$gender)
```

```
Length       Class        Mode
30000       character   character
```

```
summary(teens$grandyear)
```

```
Min.   1st Qu.  Median    Mean   3rd Qu.    Max.
2006    2007     2008     2008    2008      2009
```

```
summary(teens$friends)
```

```
Min.   1st Qu.  Median    Mean   3rd Qu.    Max.
0.00    3.00    20.00    30.18    44.00    830.00
```

Εφαρμόζοντας την περίληψη σε ολόκληρο το πλαίσιο δεδομένων, θα πάρουμε πληροφορίες για όλες τις μεταβλητές που περιλαμβάνονται σε αυτό, είτε είναι κατηγορικές είτε ποσοτικές. Έτσι έχουμε:

```
summary(teens)
```

```
 gradyear            gender                age                friends
 Min.   :2006     Length:30000        Min.   :  3.086        Min.   :  0.00
 1st Qu.:2007    Class :character     1st Qu.: 16.312        1st Qu.:  3.00
 Median :2008    Mode  :character     Median : 17.287        Median : 20.00
 Mean   :2008                         Mean   : 17.994        Mean   : 30.18
 3rd Qu.:2008                         3rd Qu.: 18.259        3rd Qu.: 44.00
 Max.   :2009                         Max.   :106.927        Max.   :830.00
                                      NA's   :5086
```

```
  basketball         football           soccer          softball
Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000
1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000
Median : 0.0000   Median : 0.0000   Median : 0.0000   Median : 0.0000
Mean   : 0.2673   Mean   : 0.2523   Mean   : 0.2228   Mean   : 0.1612
3rd Qu.: 0.0000   3rd Qu.: 0.0000   3rd Qu.: 0.0000   3rd Qu.: 0.0000
Max.   :24.0000   Max.   :15.0000   Max.   :27.0000   Max.   :17.0000

  volleyball         swimming        cheerleading       baseball
Min.   : 0.0000   Min.   : 0.0000   Min.   :0.0000   Min.   : 0.0000
1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.:0.0000   1st Qu.: 0.0000
Median : 0.0000   Median : 0.0000   Median :0.0000   Median : 0.0000
Mean   : 0.1431   Mean   : 0.1344   Mean   :0.1066   Mean   : 0.1049
3rd Qu.: 0.0000   3rd Qu.: 0.0000   3rd Qu.:0.0000   3rd Qu.: 0.0000
Max.   :14.0000   Max.   :31.0000   Max.   :9.0000   Max.   :16.0000

    tennis             sports           cute              sex
Min.   : 0.00000   Min.   : 0.00   Min.   : 0.0000   Min.   :  0.0000
1st Qu.: 0.00000   1st Qu.: 0.00   1st Qu.: 0.0000   1st Qu.:  0.0000
Median : 0.00000   Median : 0.00   Median : 0.0000   Median :  0.0000
Mean   : 0.08733   Mean   : 0.14   Mean   : 0.3229   Mean   :  0.2094
3rd Qu.: 0.00000   3rd Qu.: 0.00   3rd Qu.: 0.0000   3rd Qu.:  0.0000
Max.   :15.00000   Max.   :12.00   Max.   :18.0000   Max.   :114.0000

     sexy              hot              kissed           dance
Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000
1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000
Median : 0.0000   Median : 0.0000   Median : 0.0000   Median : 0.0000
Mean   : 0.1412   Mean   : 0.1266   Mean   : 0.1032   Mean   : 0.4252
```

```
 3rd Qu.: 0.0000   3rd Qu.: 0.0000   3rd Qu.: 0.0000   3rd Qu.: 0.0000
 Max.   :18.0000   Max.   :10.0000   Max.   :26.0000   Max.   :30.0000

      band            marching          music             rock
 Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000
 1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000
 Median : 0.0000   Median : 0.0000   Median : 0.0000   Median : 0.0000
 Mean   : 0.2996   Mean   : 0.0406   Mean   : 0.7378   Mean   : 0.2433
 3rd Qu.: 0.0000   3rd Qu.: 0.0000   3rd Qu.: 1.0000   3rd Qu.: 0.0000
 Max.   :66.0000   Max.   :11.0000   Max.   :64.0000   Max.   :21.0000

      god             church           jesus            bible
 Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.00000
 1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.00000
 Median : 0.0000   Median : 0.0000   Median : 0.0000   Median : 0.00000
 Mean   : 0.4653   Mean   : 0.2482   Mean   : 0.1121   Mean   : 0.02133
 3rd Qu.: 1.0000   3rd Qu.: 0.0000   3rd Qu.: 0.0000   3rd Qu.: 0.00000
 Max.   :79.0000   Max.   :44.0000   Max.   :30.0000   Max.   :11.00000
hair             dress           blonde             mall
 Min.   : 0.0000   Min.   :0.000   Min.   :  0.0000   Min.   : 0.0000
 1st Qu.: 0.0000   1st Qu.:0.000   1st Qu.:  0.0000   1st Qu.: 0.0000
 Median : 0.0000   Median :0.000   Median :  0.0000   Median : 0.0000
 Mean   : 0.4226   Mean   :0.111   Mean   :  0.0989   Mean   : 0.2574
 3rd Qu.: 0.0000   3rd Qu.:0.000   3rd Qu.:  0.0000   3rd Qu.: 0.0000
 Max.   :37.0000   Max.   :9.000   Max.   :327.0000   Max.   :12.0000

   shopping         clothes         hollister        abercrombie
 Min.   : 0.000   Min.   :0.0000   Min.   :0.00000   Min.   :0.00000
 1st Qu.: 0.000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.00000
```

```
 Median : 0.000   Median :0.0000   Median :0.00000   Median :0.00000
 Mean   : 0.353   Mean   :0.1485   Mean   :0.06987   Mean   :0.05117
 3rd Qu.: 1.000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.00000
 Max.   :11.000   Max.   :8.0000   Max.   :9.00000   Max.   :8.00000

      die              death             drunk             drugs
 Min.   : 0.0000   Min.   : 0.0000   Min.   :0.00000   Min.   : 0.00000
 1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.:0.00000   1st Qu.: 0.00000
 Median : 0.0000   Median : 0.0000   Median :0.00000   Median : 0.00000
 Mean   : 0.1841   Mean   : 0.1142   Mean   :0.08797   Mean   : 0.06043
 3rd Qu.: 0.0000   3rd Qu.: 0.0000   3rd Qu.:0.00000   3rd Qu.: 0.00000
 Max.   :22.0000   Max.   :14.0000   Max.   :8.00000   Max.   :16.00000
```

Στην συνέχεια θα εκτελέσουμε την εντολή na.omit (), με την χρήσης της οποίας θα παραλείψουμε όλες τις γραμμές δεδομένων που δεν έχουν κάποια τιμή:

```
teens = na.omit(teens)

dim(teens)
```

```
[1] 24005    40
```

Συσταδοποίηση

Θα προχωρήσουμε στην ανάλυση συστάδων (cluster analysis), κατηγοριοποιώντας τις 36 μεταβλητές που σχετίζονται με τα ενδιαφέροντα, και θα δημιουργήσουμε ένα νέο πλαίσιο δεδομένων μόνο με αυτές τις μεταβλητές.

```
interests <- teens[5:40]
```

Χρησιμοποιούμε την συνάρτηση lapply για να εφαρμόσουμε μια συνάρτηση σε κάθε στήλη (μεταβλητή) του πλαισίου δεδομένων. Εδώ εφαρμόζουμε την τυποποίηση βαθμολογίας z, η οποία κάνει κάθε στήλη να έχει μέσο όρο 0 και τυπική απόκλιση 1.

```
interests_z <- as.data.frame(lapply(interests, scale))
```

Για να κατηγοριοποιήσουμε τους εφήβους σε πέντε ομάδες, μπορούμε να χρησιμοποιήσουμε τη μέθοδο K-means:

```
teen_clusters <- kmeans(interests_z, 5)
```

Ενώ το μέγεθος του κάθε cluster είναι:

```
teen_clusters$size
```

```
[1]   403   17255   4783   717   847
```

και τα κέντρα των 5 ομάδων που παίρνουμε είναι:

```
teen_clusters$centers
```

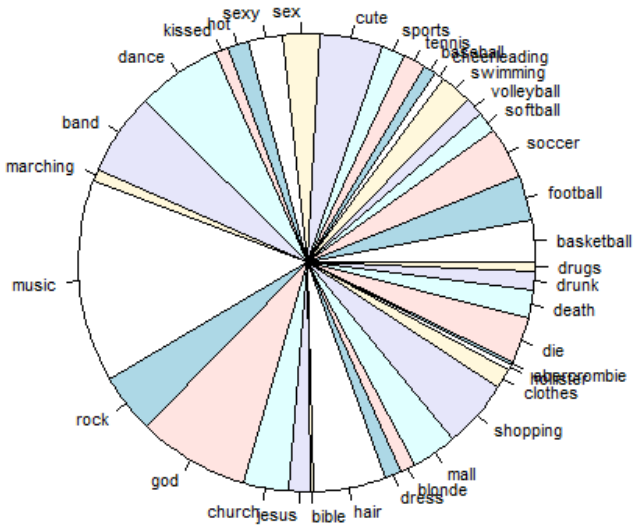
```
    basketball  football   soccer  softball volleyball  swimming cheerleading
1    0.1510    -0.004452   0.01377 -0.03832   0.004122   0.03706   0.001678
2    -0.1636   -0.171249  -0.09280 -0.11707  -0.116837  -0.09624   -0.116317
3     0.4979    0.521693   0.29185  0.38495   0.378986   0.27075    0.336997
4     0.1605    0.249815   0.12107  0.04462   0.200136   0.21498    0.380099
5     0.3143    0.333318   0.13348  0.19162   0.068687   0.23196    0.144012

    baseball   tennis    sports     cute      sex      sexy         hot
1   0.029672   0.04294   0.01238   0.02468   0.026179  -0.04248   0.07039
2  -0.109056  -0.05172  -0.12781  -0.18574  -0.096249  -0.08776  -0.13528
3   0.344627   0.14798   0.31350   0.52770  -0.008816   0.21064   0.36860
4   0.008058   0.09947   0.08630   0.40237   0.015712   0.13078   0.41142
5   0.254633   0.11345   0.75449   0.45156   1.984811   0.50795   0.29264
```

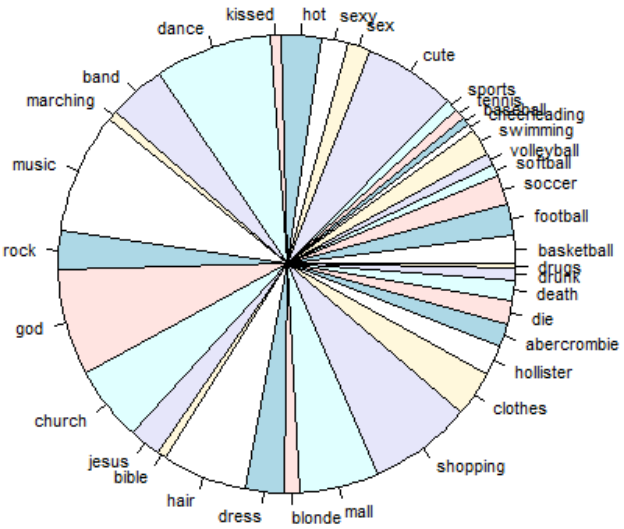
```
     kissed    dance     band marching   music     rock     god   church
1 -0.02317 -0.01394  0.14685  0.08007  0.2349  0.12825  2.2406  1.24023
2 -0.13438 -0.16607 -0.09301 -0.05726 -0.1539 -0.12691 -0.1063 -0.14412
3 -0.04044  0.49912  0.25190  0.19532  0.3165  0.23156  0.1348  0.39146
4  0.03950  0.20828 -0.10137 -0.09403  0.1072  0.05519 -0.0184 -0.02148
5  2.94352  0.39499  0.48827  0.10495  1.1448  1.17016  0.3546  0.15357

      jesus     bible       hair     dress    blonde       mall    shopping
1  2.332629  6.048477   0.05815   0.02987 -0.003915  -0.06875   -0.01053
2 -0.074454 -0.109423  -0.20514  -0.15210 -0.027665  -0.18923   -0.23212
3  0.059948 -0.105090   0.22836   0.43163  0.028245   0.49390    0.68176
4  0.006561 -0.073294   0.41466   0.12938  0.058491   0.63797    0.76795
5  0.062839  0.006795   2.51095   0.53741  0.356443   0.55865    0.23374
   clothes   hollister  abercrombie    die     death     drunk      drugs
1  0.04932   -0.09458   -0.08999     0.22400  0.28806   0.065740   0.08217
2 -0.19026   -0.15715   -0.15109    -0.09936 -0.08237  -0.088794  -0.11443
3  0.38665   -0.05650   -0.07429     0.02297  0.09233  -0.008421  -0.07876
4  0.54243    4.06769    3.90321     0.04736  0.08796   0.037217   0.02999
5  1.20992    0.12212    0.23620     1.74789  0.94516   1.793662   2.71150
```

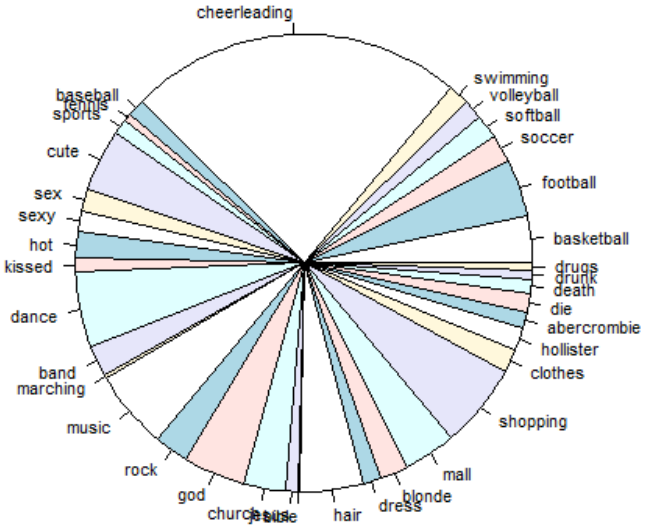
Τέλος, με την βοήθεια της εντολής pie, θα σχεδιάσουμε κυκλικά γραφήματα για κάθε ομάδα

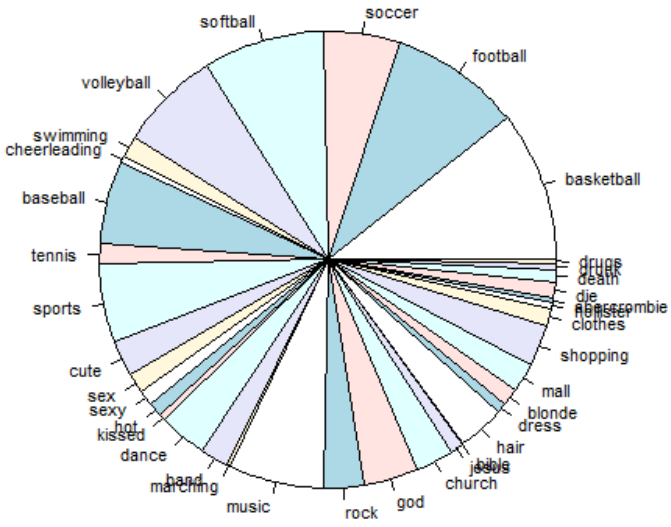
```
pie(colSums(interests[teen_clusters$cluster==1,]),cex=0.5)

pie(colSums(interests[teen_clusters$cluster==2,]),cex=0.5)

pie(colSums(interests[teen_clusters$cluster==3,]),cex=0.5)
```

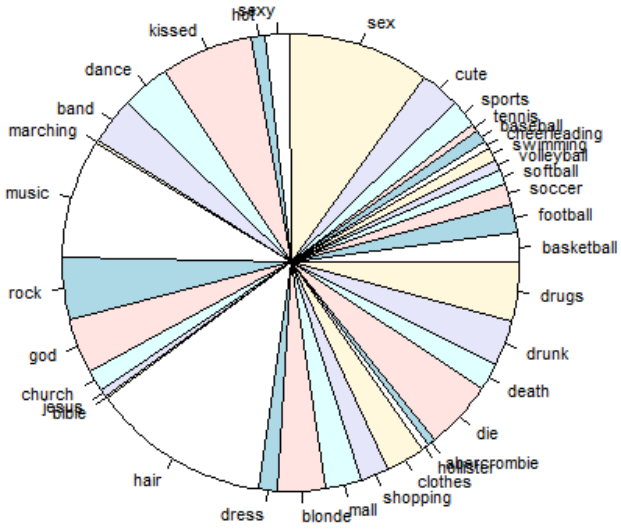
```
pie(colSums(interests[teen_clusters$cluster==4,]),cex=0.5)

pie(colSums(interests[teen_clusters$cluster==5,]),cex=0.5)
```






cheerleading
swimming
volleyball
softball
soccer
football
basketball
drugs
drunk
death
die
abercrombie
hollister
clothes
shopping
mall
blonde
dress
hair
church
god
rock
music
band
marching
dance
kissed
hot
sexy
sex
cute
sports
tennis
baseball


soccer
football
basketball
drugs
death
clothes
shopping
mall
blonde
dress
hair
church
god
rock
music
marching
dance
kissed
hot
sexy
sex
cute
sports
tennis
baseball
cheerleading
swimming
volleyball
softball



Παρατηρούμε ότι για την πρώτη ομάδα ξεχωρίζει η μουσική ως το μεγαλύτερο ενδιαφέρον, ενώ μετά έχουμε τη θρησκεία, το ροκ, τις μπάντες και το χορό. Για τη δεύτερη ομάδα δεν υπάρχει ξεκάθαρο ενδιαφέρον, αλλά σημαντικό μερίδιο καταλαμβάνουν η μουσική, ο χορός, και η θρησκεία. Για την τρίτη ομάδα κυριαρχεί το cheerleading με μεγάλη διαφορά από τα υπόλοιπα ενδιαφέροντα. Για την τέταρτη ομάδα το basketball, football, softball, soccer αλλά και η μουσική. Τέλος, στην πέμπτη ομάδα, υπάρχουν ενδιαφέροντα όπως μαλλιά, μουσική, σεξ και ναρκωτικά. Επομένως παρατηρούμε ότι η ομαδοποίηση όντως δείχνει ότι οι χρήστες που ανήκουν στην εκάστοτε ομάδα έχουν ενδιαφέροντα που σχετίζονται μεταξύ τους. Για παράδειγμα, η πρώτη ομάδα σχετίζεται με τη μουσική, ενώ η τέταρτη ομάδα σχετίζεται ξεκάθαρα με τον αθλητισμό, όπως πιθανόν και η τρίτη. Η δεύτερη έχει πιο συντηρητικά ενδιαφέροντα, σε αντίθεση με την πέμπτη που φαίνεται πιο απελευθερωμένη.

Case Study III: IRIS Classification

Εισαγωγή

Η παρούσα μελέτη περίπτωσης αφορά σε πρόβλημα ταξινόμησης, στο οποίο θα αναλύσουμε το γνωστό σύνολο δεδομένων λουλουδιών «Iris flower», του Fisher (1936). Μέσα από τη μελέτη αυτή μας δίνεται η ευκαιρία να εφαρμόσουμε πολλούς αλγόριθμους Μηχανικής Μάθησης και να εκπαιδεύσουμε διαφορετικούς τύπους μοντέλων που ταξινομούν τα λουλούδια βάσει των χαρακτηριστικών τους.

Το σύνολο δεδομένων Iris, είναι γνωστό στους κύκλους της επιστήμης δεδομένων και περιέχει 4 χαρακτηριστικά: μήκος και πλάτος σέπαλου σε εκατοστά, και μήκος και πλάτος πετάλου επίσης σε εκατοστά. Αποτελείται από 150 συνολικά παρατηρήσεις του λουλουδιού Iris, που κατανέμονται σε 50 δείγματα από κάθε είδος και συγκεκριμένα από τα τρία είδη: Iris Setosa, Iris Versicolour και Iris Virginica.

Φόρτωση των δεδομένων

Το σύνολο δεδομένων Iris είναι διαθέσιμο και ήδη προ-φορτωμένο στο πρόγραμμα R. Έτσι, χρησιμοποιούμε την συνάρτηση data, για να φορτώσουμε και να εμφανίσουμε τα δεδομένα.

```
data(iris)
```

Διαχωρισμός των δεδομένων σε σύνολα

Από τα πιο σημαντικά βήματα στη Μηχανική Εκμάθηση είναι ο διαχωρισμός των διαθέσιμων δεδομένων σε σύνολα. Έτσι, πριν ξεκινήσουμε την ανάλυσή μας χωρίζουμε τα δεδομένα μας σε 2 σύνολα, εκπαιδευτικό και σετ δοκιμής σε ποσοστό 80%-20%.

train set (εκπαιδευτικό σετ): Είναι τα δεδομένα εκείνα που χρησιμοποιούνται για την κατασκευή και την εκπαίδευση του μοντέλου. Σε ένα πρόβλημα ταξινόμησης όπως αυτό, εκπαιδεύουμε το μοντέλο χρησιμοποιώντας το ποσοστό σφάλματος ταξινόμησης, δηλαδή το ποσοστό των λανθασμένων ή σωστά ταξινομημένων

παρατηρήσεων. Αυτό μας βοηθά να κατανοήσουμε τα δεδομένα και να προσδιορίσουμε τις παραμέτρους του μοντέλου ώστε να επιλέξουμε το κατάλληλο.
test set (σετ δοκιμής): Πρόκειται για τα νέα δεδομένα. Δημιουργούμε ένα μοντέλο για να τα ταξινομήσουμε. Η απόδοση του μοντέλου αυτού (ποσοστό σφάλματος), είναι η πιο ρεαλιστική εκτίμηση της εφαρμογής του στον πραγματικό κόσμο.

Θέτουμε μια αρχική τιμή στη set.seed, ώστε ο τυχαίος χωρισμός των δεδομένων να είναι ο ίδιος κάθε φορά που εκτελούμε τον κώδικα.

```
library(caret)

set.seed(49)

index <- createDataPartition(iris$Species, p=0.80, list=FALSE)

test <- iris[-index,]

train <- iris[index,]
```

Στην συνέχεια θα εξετάσουμε τα δεδομένα εκπαίδευσης.
Αρχικά με την συνάρτηση dim τυπώνουμε τις διαστάσεις του πίνακα, και παρατηρούμε ότι πράγματι το εκπαιδευτικό σύνολο περιέχει 120 παρατηρήσεις από 5 μεταβλητές, (ποσοστό 80%). Για τη δομή του πίνακα χρησιμοποιούμε την συνάρτηση str, ενώ για συνοπτική επισκόπηση των δεδομένων χρησιμοποιούμε την συνάρτηση summary. Τα δεδομένα αποτελούνται από 4 μεταβλητές (Sepal.Length, Sepal.Width, Petal.Length, Petal.Width), ενώ η μεταβλητή Species είναι η κατηγορία είδους λουλουδιού, δηλαδή μία από τις τιμές Iris - setosa, Iris - versicolor, Iris – virginica.

```
dim(train)

str(train)

summary(train)

levels(train$Species)
```

```
> dim(train)
[1] 120   5
> # Structure of the data
> str(train)
'data.frame':    120 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 4.6 5 4.4 4.9 5.4 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.4 3.4 2.9 3.1 3.7 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.4 1.5 1.4 1.5 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.3 0.2 0.2 0.1 0.2 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1
1 ...
> # Summary of the data
> summary(train)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :40
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.575   1st Qu.:0.300   versicolor:40
 Median :5.800   Median :3.000   Median :4.400   Median :1.350   virginica :40
 Mean   :5.852   Mean   :3.057   Mean   :3.775   Mean   :1.207
 3rd Qu.:6.425   3rd Qu.:3.300   3rd Qu.:5.125   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
> # Levels of the prediction column
> levels(train$Species)
[1] "setosa"     "versicolor" "virginica"
```

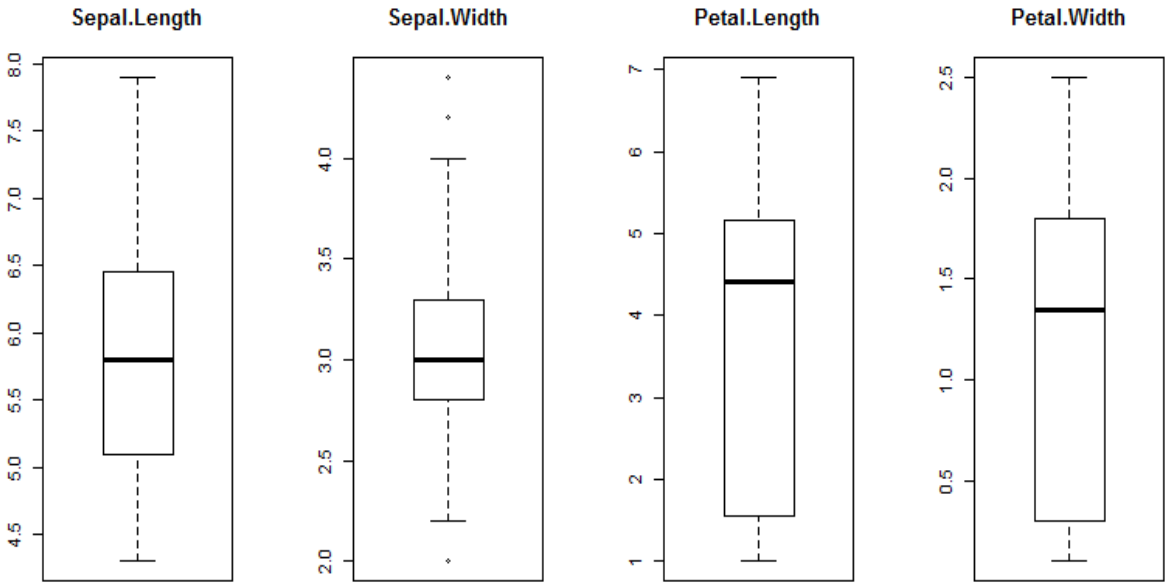
Απεικόνιση και κατανόηση δεδομένων

Το πρόγραμμα R, μας επιτρέπει να εξερευνήσουμε δημιουργικά και γρήγορα τα δεδομένα μέσω οπτικοποίησης. Θα χρησιμοποιήσουμε μερικά από τα εργαλεία του για να παράγουμε διάφορους τύπους διαγραμμάτων με σκοπό να εξερευνήσουμε τα δεδομένα και να κατανοήσουμε καλύτερα τις μεταβλητές και τη σχέση τους με την κάθε κατηγορία λουλουδιού.

Αρχικά παράγουμε ένα boxplot, το οποίο δείχνει την κατανομή των τιμών για κάθε μία από τις 4 μεταβλητές. Με την συνάρτηση par παράγουμε πολλαπλά διαγράμματα στο ίδιο γράφημα (φαίνεται από το όρισμα mfrwo = c(1,4)).

```
library(ggplot2)

par(mfrow=c(1,4))

for(i in 1:4) {

boxplot(train[,i], main=names(train)[i])
```

```
}
```

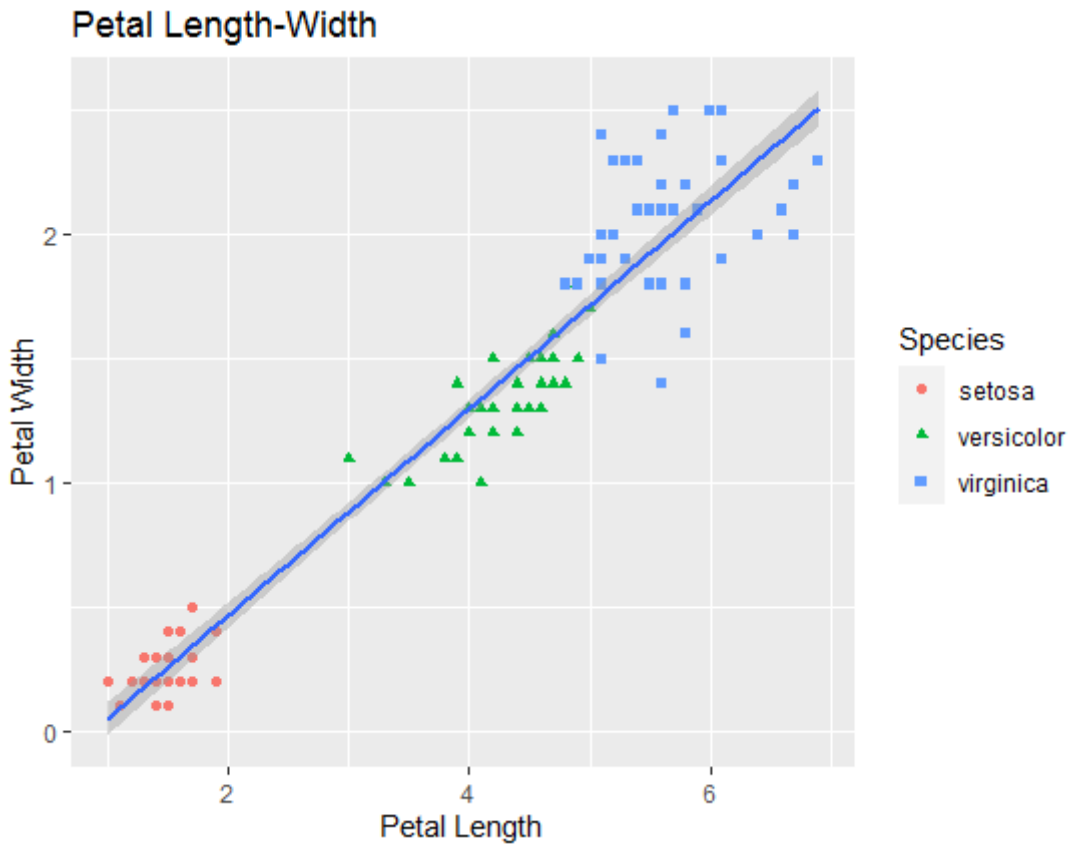


Συνεχίζουμε την ανάλυση με το διάγραμμα διασποράς. Συγκεκριμένα θα δημιουργήσουμε δύο scatterplot, ένα για κάθε δύο ζεύγη μεταβλητών (Petal Length–Width και Sepal Length–Width). Το κάθε σημείο αντιπροσωπεύει ένα δείγμα (γραμμή των δεδομένων) και έχει σχήμα και χρώμα ανάλογα με το είδος που ανήκει το συγκεκριμένο δείγμα. Παρατηρούμε ότι με τις μεταβλητές Petal.Length και Petal.Width τα δεδομένα χωρίζονται αρκετά καλά όσον αφορά τα τρία είδη λουλουδιών.

Πιο αναλυτικά, στην συνάρτηση ggplot δίνουμε ως όρισμα τα δεδομένα και καθορίζουμε ποιες μεταβλητές θα είναι στους άξονες x και y, (π.χ. για το πρώτο διάγραμμα x = Petal.Length και y = Petal.Width). Στη συνέχεια ορίζουμε ότι θέλουμε να απεικονίσουμε σημεία (geom_point), με χρώμα και σχήμα ανάλογα με τη στήλη Species. Με το ggtitle ορίζουμε τους τίτλους των δύο αξόνων και του γραφήματος, και τέλος, η τάση που έχουμε στα δεδομένα μας δίνεται με την συνάρτηση geom_smooth, μέθοδος lm = linear method.

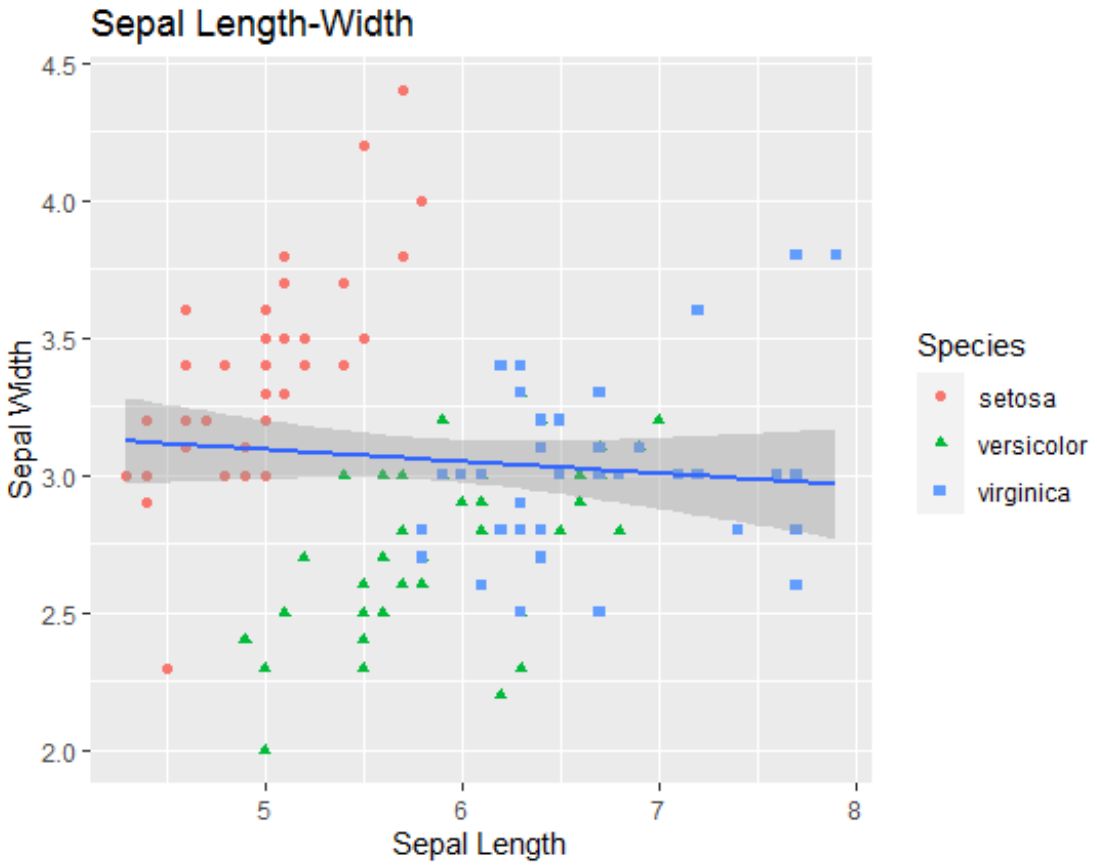
Διάγραμμα Petal Length-Width

```
g <- ggplot(data=train, aes(x = Petal.Length, y = Petal.Width))

g <-g +

  geom_point(aes(color=Species, shape=Species)) +

  xlab("Petal Length") +

  ylab("Petal Width") +

  ggtitle("Petal Length-Width")+

  geom_smooth(method="lm")

print(g)
```



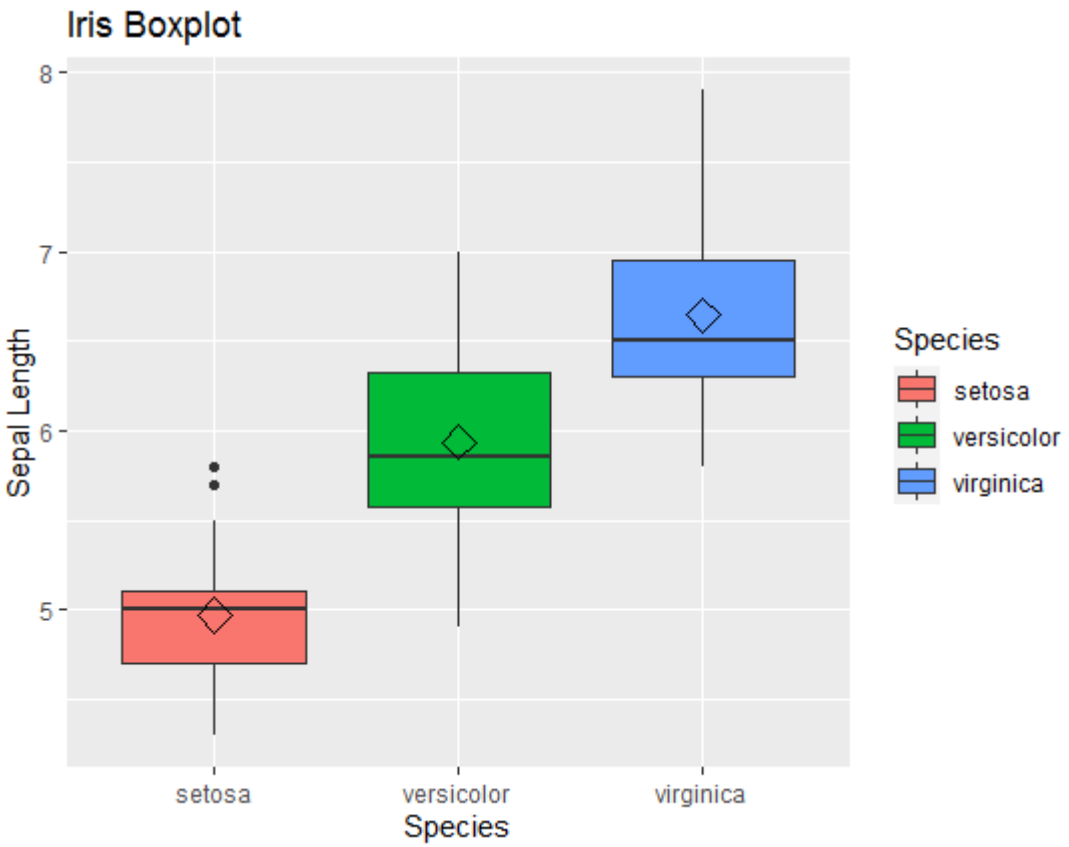
Διάγραμμα Sepal Length-Width

```
g <- ggplot(data=train, aes(x = Sepal.Length, y = Sepal.Width))

g <-g +

  geom_point(aes(color=Species, shape=Species)) +

  xlab("Sepal Length") +

  ylab("Sepal Width") +

  ggtitle("Sepal Length-Width")+

  geom_smooth(method="lm")

print(g)
```



Στην συνέχεια για κάθε κατηγορία θα απεικονίσουμε την τιμή της μεταβλητής Sepal.Length σε ένα Θηκόγραμμα – Boxplot. Ομοίως με πριν, δηλώνουμε στην συνάρτηση ggplot ότι στον άξονα x θα είναι το είδος και στον άξονα y η μεταβλητή Sepal.Length. Στη συνέχεια ορίζουμε ότι στο boxplot το χρώμα γεμίσματος θα είναι ανάλογο με το εκάστοτε είδος. Τέλος δίνουμε τίτλους στον άξονα y και γενικό τίτλο στο διάγραμμα, ενώ σε κάθε boxplot σημειώνουμε το μέσο όρο με ένα ρόμβο (stat_summary με fun.y = mean, geom=”point”, shape=5).

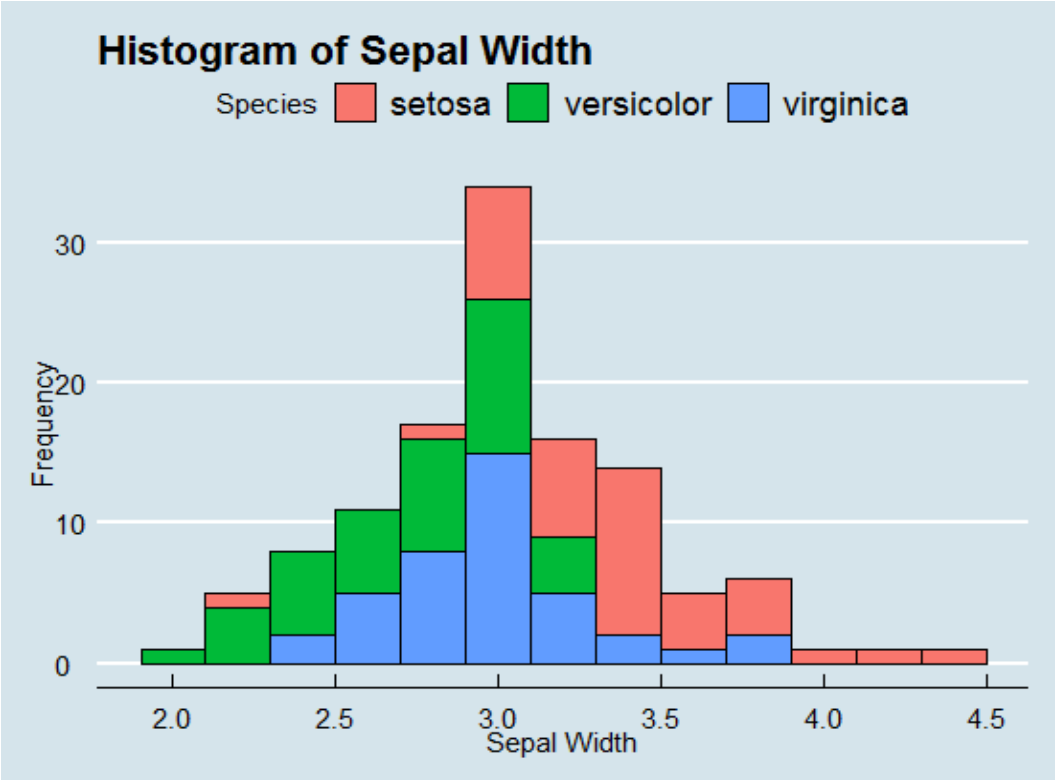
```
box <- ggplot(data=train, aes(x=Species, y=Sepal.Length)) +

  geom_boxplot(aes(fill=Species)) +

  ylab("Sepal Length") +

  ggtitle("Iris Boxplot") +

  stat_summary(fun.y=mean, geom="point", shape=5, size=4)

print(box)
```



Τέλος για τη μεταβλητή Sepal.Width (x= Sepal.Widht), θα απεικονίσουμε τις τιμές της ανά είδος σε ιστόγραμμα. Στο διάγραμμα θα ορίσουμε ότι θέλουμε ιστόγραμμα με την συνάρτηση geom_histogram. Στα ορίσματα θα καθορίσουμε συγκεκριμένο πλάτος για κάθε κουτί 0.2, και χρώμα γεμίσματος ανάλογα με το είδος του λουλουδιού, καθώς και μαύρο περίγραμμα. Ομοίως με την ggtitle ορίζουμε τίτλους αξόνων και γενικό τίτλο και θέτουμε θέμα theme_economist, το οποίο σχετίζεται με το background.

```
library(ggthemes)

histogram <- ggplot(data=train, aes(x=Sepal.Width)) +

  geom_histogram(binwidth=0.2, color="black", aes(fill=Species)) +

  xlab("Sepal Width") +

  ylab("Frequency") +

  ggtitle("Histogram of Sepal Width")+

  theme_economist()
```

```
print(histogram)
```



Ταξινόμηση

Το επόμενο βήμα είναι η δημιουργία του κατάλληλου μοντέλου (Model Building). Θα πραγματοποιήσουμε ταξινόμηση με διάφορους αλγορίθμους: LDA, δέντρο αποφάσεων, τυχαίο δάσος και KNN. Σε κάθε αλγόριθμο, θα δημιουργούμε ένα μοντέλο με τα δεδομένα εκπαίδευσης (train set), ενώ θα προβλέψουμε την ταξινόμηση με τα δεδομένα δοκιμής (test set). Στη συνέχεια θα εξετάσουμε κατά πόσο η πρόβλεψη στα test δεδομένα συμπίπτει με τις πραγματικές τιμές των test δεδομένων.

LDA

Θα ξεκινήσουμε με τη μέθοδο που ο ίδιος ο Fisher χρησιμοποίησε, και στη συνέχεια θα προσπαθήσουμε να βελτιώσουμε το μοντέλο χρησιμοποιώντας άλλους αλγόριθμους μηχανικής μάθησης. Το μοντέλο Linear Discriminant Analysis (LDA), προτιμάται γενικότερα για μικρά σύνολα δεδομένων και είναι

χρήσιμο για προβλήματα ταξινόμησης όπου υπάρχουν περισσότερες από δύο τιμές για τη μεταβλητή πρόβλεψης όπως εδώ.

Το μοντέλο LDA που θα δημιουργήσουμε, θα έχει όρισμα το Species ~ . που σημαίνει ότι θέλουμε να προβλέψουμε την μεταβλητή Species με βάση όλες τις υπόλοιπες μεταβλητές. Στη συνέχεια εκτυπώνουμε το μοντέλο που εκπαιδεύτηκε. Αυτό υπολογίζει διάφορους συντελεστές (LD1, LD2) για κάθε κατηγορία.

```
library(MASS)

model_LDA <- lda(Species ~ ., train)

model_LDA
```

```
> model_LDA
Call:
lda(Species ~ ., data = train)

Prior probabilities of groups:
    setosa versicolor  virginica
 0.3333333  0.3333333  0.3333333


Group means:
           Sepal.Length Sepal.Width Petal.Length Petal.Width
setosa           5.0625      3.4900       1.4775      0.2550
versicolor       5.9125      2.7675       4.2675      1.3275
virginica        6.4950      2.9675       5.4950      2.0725

Coefficients of linear discriminants:
                    LD1        LD2
Sepal.Length  0.7802014  0.1956841
Sepal.Width   1.8279837  1.6328905
Petal.Length -2.0395485 -1.7456434
Petal.Width  -2.9028309  4.3862929

Proportion of trace:
  LD1   LD2
0.986 0.014
```

Η αναλογία ίχνους δείχνει πόσο καλά γίνεται η διάκριση μεταξύ των ειδών. Δεδομένου του πολύ μεγάλου μεγέθους του πρώτου LD1 (0,986), βλέπουμε ότι εξηγεί το 99% της διακύμανσης, ενώ το LD2 (0,014), συμβάλλει πολύ λίγο στη διάκριση των ειδών.

Στην συνέχεια θα κάνουμε προβλέψεις των ειδών λουλουδιών Iris, χρησιμοποιώντας τη συνάρτηση predict στο σετ δοκιμής (test set). Με την συνάρτηση ConfusionMatrix παράγουμε τον πίνακα σύγχυσης που δείχνει αναλυτικά το πραγματικό είδος του λουλουδιού Iris (στις στήλες), και το προβλεπόμενο (στις γραμμές), για κάθε είδος. Παρατηρούμε ότι υπάρχει μόνο μια λανθασμένη πρόβλεψη στον πίνακα, virginica αντί versicolor. Επιπλέον παίρνουμε διάφορα στατιστικά στοιχεία με πιο σημαντικό την ακρίβεια (accuracy), η οποία ξεπερνά το 96%. Το μοντέλο έχει επιτυχία και αποτελεί ένα υψηλό σημείο αναφοράς.

```
predict_LDA <- predict(model_LDA, test)

confusion_LDA <- confusionMatrix(predict_LDA$class, test$Species)

confusion_LDA
```

```
Confusion Matrix and Statistics

            Reference
Prediction   setosa versicolor virginica
  setosa         10          0         0
  versicolor      0          9         0
  virginica       0          1        10


Overall Statistics

               Accuracy : 0.9667
                 95% CI : (0.8278, 0.9992)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : 2.963e-13

                  Kappa : 0.95

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: setosa Class: versicolor Class: virginica
Sensitivity                 1.0000            0.9000           1.0000
Specificity                 1.0000            1.0000           0.9500
Pos Pred Value              1.0000            1.0000           0.9091
```

Neg Pred Value	1.0000	0.9524	1.0000
Prevalence	0.3333	0.3333	0.3333
Detection Rate	0.3333	0.3000	0.3333
Detection Prevalence	0.3333	0.3000	0.3667
Balanced Accuracy	1.0000	0.9500	0.9750

Δέντρο απόφασης

Τα δέντρα απόφασης είναι ένα πολύ χρήσιμο εργαλείο που χρησιμοποιείται ευρέως, ειδικά όταν ο αριθμός των μεταβλητών πρόβλεψης είναι μικρός όπως εδώ, γιατί είναι εύκολα ερμηνεύσιμοι. Εδώ θα χρησιμοποιήσουμε τον απλό αλγόριθμο rpart για να ταξινομήσουμε το σύνολο δεδομένων μας και να κάνουμε προβλέψεις, ενώ για την απεικόνιση θα χρησιμοποιήσουμε την συνάρτηση rpart.plot.

Στη συνάρτηση rpart δίνουμε τον τύπο, τα δεδομένα, ορίζουμε κάποιες άλλες παραμέτρους (όπως minsplit = 5, minbucket = 2), και καθορίζουμε ότι θέλουμε ταξινόμηση (classification).

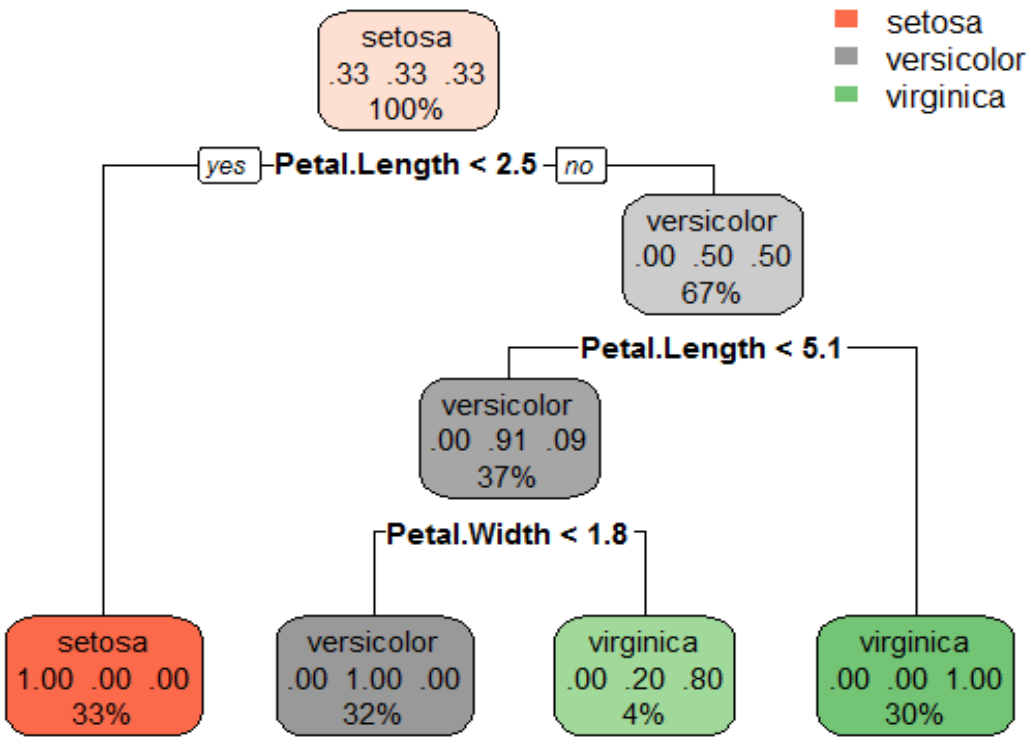
```
library(rpart)

library(rpart.plot)

iristree <- rpart(formula = Species ~ ., data = train, control =
rpart.control(minsplit = 5, minbucket = 2), method = "class")

print(iristree)

rpart.plot(iristree)
```

Το ίδιο το διάγραμμα λειτουργεί και ως ένα χρήσιμο σύνολο κανόνων που εφαρμόζονται για την πρόβλεψη, δηλαδή οι κόμβοι του παραγόμενου σχήματος υποδεικνύουν τα αντίστοιχα είδη εάν σταματήσουμε στον εκάστοτε κόμβο.
Έτσι:

εάν το πέταλο έχει μήκος Petal.Length < 2,5 εκατοστά τότε το είδος είναι Setosa

εάν το πέταλο έχει μήκος Petal.Length < 5,1 εκατοστά τότε το είδος είναι Virginica
εάν το πέταλο έχει μήκος Petal.Length μεταξύ 2,5 και 5,1 εκατοστά, και πλάτος Petal.Width < 1,8 εκατοστά τότε το είδος είναι Virginica
εάν το πέταλο έχει μήκος Petal.Length μεταξύ 2,5 και 5,1 εκατοστά, και πλάτος Petal.Width > 1,8 εκατοστά τότε το είδος είναι Versicolor.

```
n= 120
node), split, n, loss, yval, (yprob)
      * denotes terminal node
 1) root 120 80 setosa (0.33333333 0.33333333 0.33333333)
 2) Petal.Length< 2.45 40  0 setosa (1.00000000 0.00000000 0.00000000) *
 3) Petal.Length>=2.45 80 40 versicolor (0.00000000 0.50000000 0.50000000)
 6) Petal.Length< 5.05 44  4 versicolor (0.00000000 0.90909091 0.09090909)
 12) Petal.Width< 1.75 39  0 versicolor (0.00000000 1.00000000 0.00000000) *
 13) Petal.Width>=1.75 5  1 virginica (0.00000000 0.20000000 0.80000000) *
 7) Petal.Length>=5.05 36  0 virginica (0.00000000 0.00000000 1.00000000) *
```



Ομοίως με πριν, θα κάνουμε πρόβλεψη με τα δεδομένα του σετ δοκιμής (set test). Παρατηρούμε ότι έχουμε εσφαλμένη ταξινόμηση 3 λουλουδιών, 1 virginica αντί versicolor και 2 versicolor αντί virginica. Το ποσοστό ακρίβειας ανέρχεται σε 3/30 = 90%, επομένως το μοντέλο δεν αποδίδει τόσο καλά όσο το LDA. Το επόμενο λογικό βήμα είναι ο συνδυασμός πολλαπλών δέντρων, και ούτω καθεξής σε τυχαίο δάσος που ακολουθεί.

```
predict_tree <- predict(iristree, test, type="class")

confusion_tree <- confusionMatrix(predict_tree, test$Species)

confusion_tree
```

```
Confusion Matrix and Statistics
            Reference
Prediction   setosa versicolor virginica
  setosa         10          0         0
  versicolor      0          9         2
  virginica       0          1         8

Overall Statistics

               Accuracy : 0.9
                 95% CI : (0.7347, 0.9789)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : 1.665e-10

                  Kappa : 0.85

 Mcnemar's Test P-Value : NA

Statistics by Class:
                     Class: setosa Class: versicolor Class: virginica
Sensitivity                 1.0000            0.9000           0.8000
Specificity                 1.0000            0.9000           0.9500
Pos Pred Value              1.0000            0.8182           0.8889
Neg Pred Value              1.0000            0.9474           0.9048
Prevalence                  0.3333            0.3333           0.3333
Detection Rate              0.3333            0.3000           0.2667
Detection Prevalence        0.3333            0.3667           0.3000
Balanced Accuracy           1.0000            0.9000           0.8750
```

Τυχαίο Δάσος

Στην συνέχεια θα εφαρμόσουμε τον αλγόριθμο Τυχαίων Δασών ο οποίος είναι μια μέθοδος που εκπαιδεύει πολλά δέντρα απόφασης μαζί. Αρχικά εκπαιδεύουμε ένα μοντέλο, αφήνοντας τις διάφορες παραμέτρους στην προεπιλεγμένη τιμή, (αριθμός δέντρων 500 και να εξετάζονται 2 μεταβλητές για διαχωρισμό σε κάθε κόμβο). Επίσης, ορίζουμε να υπολογιστεί η τιμή της variable importance, η οποία δείχνει πόσο σημαντική είναι η κάθε μεταβλητή για την πρόβλεψη του αποτελέσματος.

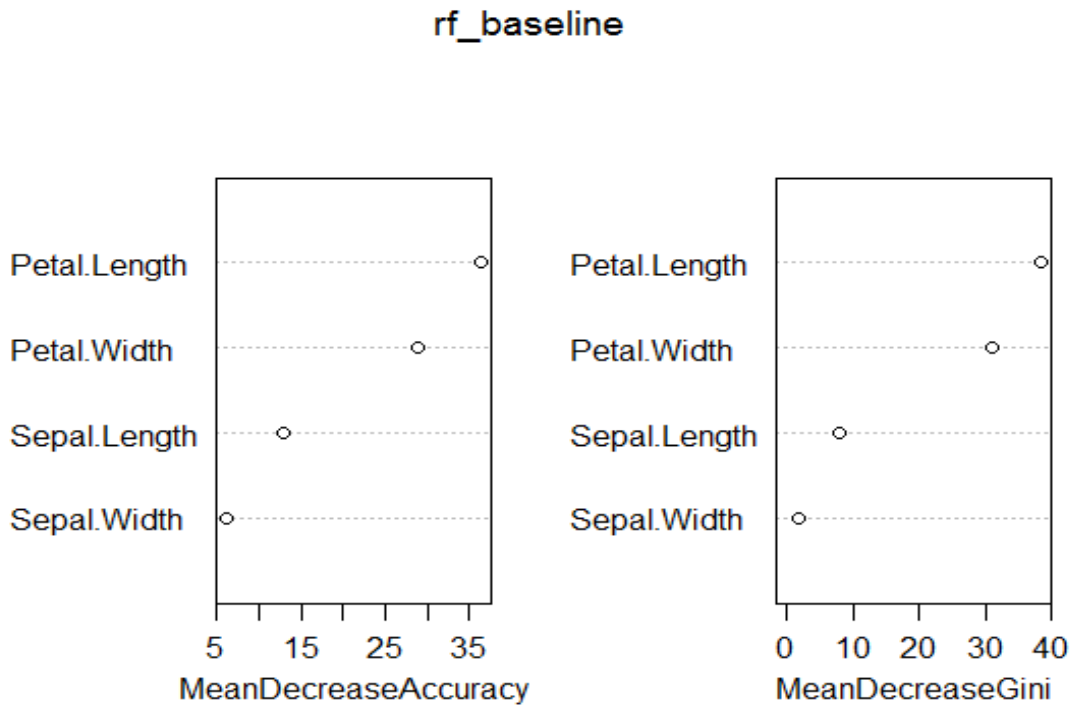
```
library(randomForest)

rf_baseline <- randomForest(Species ~ ., train, importance=TRUE)

rf_baseline
```

```
Call:
 randomForest(formula = Species ~ ., data = train, importance = TRUE)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 3.33%
Confusion matrix:
           setosa versicolor virginica class.error
setosa         40          0         0        0.00
versicolor      0         38         2        0.05
virginica       0          2        38        0.05
```

Το σφάλμα για αυτό το μοντέλο Out-of-bag (OOB) είναι 3,33%, το οποίο μας δίνει ακρίβεια 96,7%. Δύο εσφαλμένες ταξινομήσεις του versicolor και δύο του είδους virginica παρατηρούνται.

Η απεικόνιση της τιμής variable importance, δείχνει ότι έχει υπολογιστεί με δύο μεθόδους. Πιο σημαντική φαίνεται ότι είναι η μεταβλητή Petal.Length και μετά η Petal.Width.

```
varImpPlot(rf_baseline)
```



Στην συνέχεια πραγματοποιούμε πρόβλεψη στα δεδομένα δοκιμής test set. Η ακρίβεια είναι 90% και στον πίνακα βλέπουμε τρία λάθη, ίδια με τον αλγόριθμο του δέντρου αποφάσεων.

```
predict_rf_baseline <- predict(rf_baseline, test)

confusion_RF <- confusionMatrix(predict_rf_baseline, test$Species)

confusion_RF
```

```
Confusion Matrix and Statistics

            Reference
Prediction   setosa versicolor virginica
  setosa         10          0         0
  versicolor      0          9         2
  virginica       0          1         8

Overall Statistics

               Accuracy : 0.9
                 95% CI : (0.7347, 0.9789)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : 1.665e-10

                  Kappa : 0.85
```

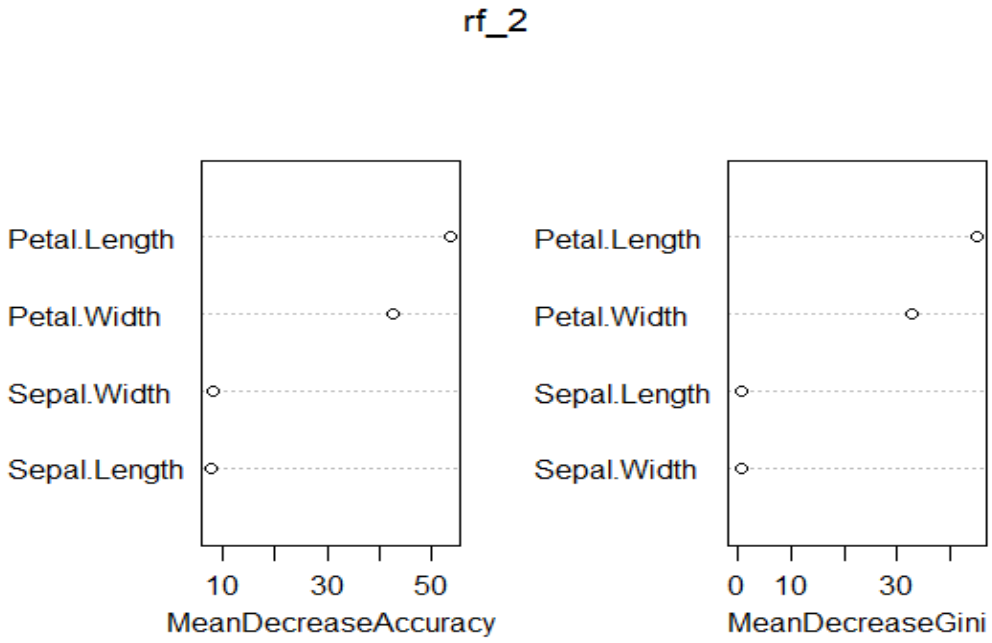
```
 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: setosa Class: versicolor Class: virginica
Sensitivity                 1.0000            0.9000           0.8000
Specificity                 1.0000            0.9000           0.9500
Pos Pred Value              1.0000            0.8182           0.8889
Neg Pred Value              1.0000            0.9474           0.9048
Prevalence                  0.3333            0.3333           0.3333
Detection Rate              0.3333            0.3000           0.2667
Detection Prevalence        0.3333            0.3667           0.3000
Balanced Accuracy           1.0000            0.9000           0.8750
```

Επαναλαμβάνουμε τη διαδικασία, αλλά ορίζουμε τον αριθμό των δέντρων σε 1000 και τον αριθμό των μεταβλητών που εξετάζονται για διαχωρισμό σε 4. Από την γραφική απεικόνιση της variable importance, βλέπουμε ότι αυξάνεται η σημασία των μεταβλητών Petal.Length και Petal.Width, ενώ οι άλλες δύο δεν αποτελούν σημαντικούς προγνωστικούς παράγοντες για το μοντέλο, αφού έχουν σχεδόν μηδενικό variable importance.

```
rf_2 <-randomForest(formula = Species ~ ., data = train, importance =
TRUE,ntree=1000, mtry=4)

rf_2
```

```
Call:
 randomForest(formula = Species ~ ., data = train, importance = TRUE,
ntree = 1000, mtry = 4)
               Type of random forest: classification
                     Number of trees: 1000
No. of variables tried at each split: 4

        OOB estimate of  error rate: 2.5%
Confusion matrix:
           setosa versicolor virginica class.error
setosa         40          0         0       0.000
versicolor      0         38         2       0.050
virginica       0          1        39       0.025
```

```
varImpPlot(rf_2)
```



Παρατηρούμε ότι παρόλο που η ακρίβεια βελτιώθηκε στα δεδομένα προπόνησης, δεν άλλαξε την ακρίβεια στο σύνολο δοκιμών. Ούτε στην πρόβλεψη το αποτέλεσμα βελτιώνεται. Παραμένουν τα τρία λάθη και η ακρίβεια στο 90%.

```
predict_rf_2 <- predict(rf_2, test)

confusion_rf2 <- confusionMatrix(predict_rf_2, test$Species)

Models_Accuracies <- add_row(Models_Accuracies, model = "Random Forest
complex", accuracy = confusion_rf2$overall['Accuracy'])

confusion_rf2
```

```
Confusion Matrix and Statistics

            Reference
Prediction   setosa versicolor virginica
  setosa         10          0         0
  versicolor      0          9         2
```

```
  virginica       0          1         8

Overall Statistics

               Accuracy : 0.9
                 95% CI : (0.7347, 0.9789)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : 1.665e-10

                  Kappa : 0.85

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: setosa Class: versicolor Class: virginica
Sensitivity                 1.0000            0.9000           0.8000
Specificity                 1.0000            0.9000           0.9500
Pos Pred Value              1.0000            0.8182           0.8889
Neg Pred Value              1.0000            0.9474           0.9048
Prevalence                  0.3333            0.3333           0.3333
Detection Rate              0.3333            0.3000           0.2667
Detection Prevalence        0.3333            0.3667           0.3000
Balanced Accuracy           1.0000            0.9000           0.8750
```

KNN

Τέλος θα χρησιμοποιήσουμε τον αλγόριθμο του κοντινότερου γείτονα KNN (K Nearest Neighbors), ο οποίος βασίζεται σε στιγμιότυπα και είναι ενδιαφέρον να δούμε πώς λειτουργεί εδώ. Η μέθοδος αυτή έχει μια παράμετρο K, για την οποία δοκιμάζουμε τιμές από το 1 ως το 10, και θα δημιουργήσουμε 10 μοντέλα. Ως ορίσματα δέχεται τα δεδομένα και τις κατηγορίες των δεδομένων του εκπαιδευτικού σετ. Υπολογίζουμε την ακρίβεια για κάθε K και βλέπουμε ότι βελτιώνεται για K=4 και πάνω, οπότε στη συνέχεια δημιουργούμε ένα μοντέλο με K=4.

```
library(class)

model_knn <- list()

accuracy_knn <- numeric()

for (i in 1:10) {

  model_knn[[i]] <- knn(train[,-5], test[,-5], train$Species, k=i, prob=TRUE)
```

```
  accuracy_knn[i] <- sum(model_knn[[i]]==test$Species)/length(test$Species)*100
}
accuracy_knn
```

```
[1] 90.00000 90.00000 90.00000 93.33333 93.33333 93.33333 93.33333
93.33333 93.33333 93.33333
```

Από τα στατιστικά στοιχεία παρατηρούμε ότι ο αλγόριθμος αυτός κάνει δύο λάθη και η ακρίβεια του μοντέλου φαίνεται να είναι στο 93 %.

```
model_knn4 <- knn(train[,-5], test[,-5], train$Species, k=4, prob=TRUE)

confusion_knn <- table(test$Species, model_knn4)

confusion_knn
```

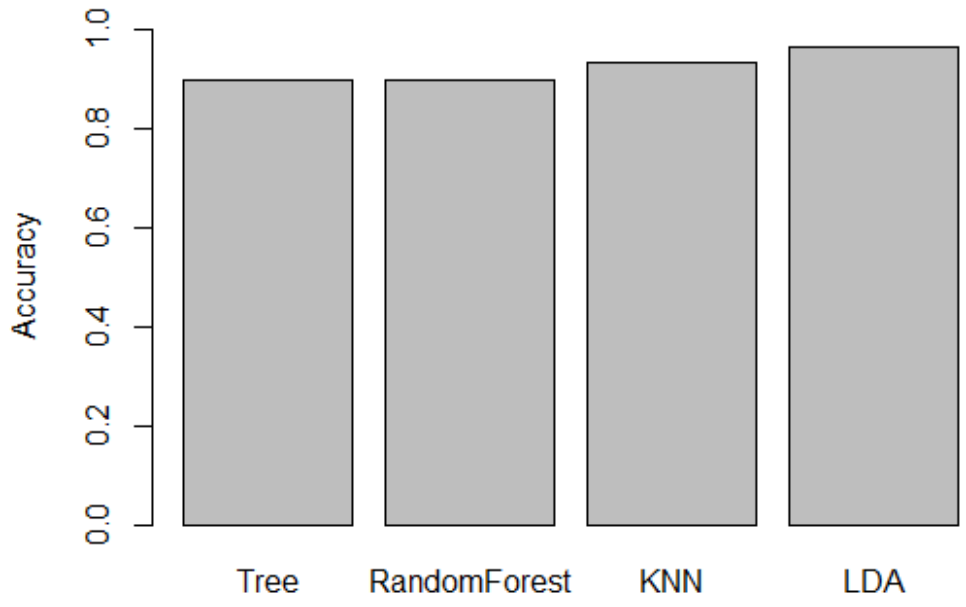
```
Confusion Matrix and Statistics

            Reference
Prediction   setosa versicolor virginica
  setosa         10          0         0
  versicolor      0          9         1
  virginica       0          1         9

Overall Statistics

               Accuracy : 0.9333
                 95% CI : (0.7793, 0.9918)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : 8.747e-12

                  Kappa : 0.9

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: setosa Class: versicolor Class: virginica
Sensitivity                 1.0000            0.9000           0.9000
Specificity                 1.0000            0.9500           0.9500
Pos Pred Value              1.0000            0.9000           0.9000
Neg Pred Value              1.0000            0.9500           0.9500
```

Prevalence	0.3333	0.3333	0.3333
Detection Rate	0.3333	0.3000	0.3000
Detection Prevalence	0.3333	0.3333	0.3333
Balanced Accuracy	1.0000	0.9250	0.9250

Σύγκριση αλγορίθμων ταξινόμησης – Συμπέρασμα

Τέλος, απεικονίζουμε την ακρίβεια όλων των αλγορίθμων μαζί σε ένα ραβδόγραμμα. Ο καλύτερος αλγόριθμος φαίνεται να είναι ο LDA με ακρίβεια 96%, μετά ο KNN με 93% και τέλος τα δέντρο αποφάσεων και τυχαίο δάσος με 90%. Παρατηρούμε ότι ο αλγόριθμος που χρησιμοποίησε ο ίδιος ο Fisher για να ταξινομήσει τα λουλούδια Iris λειτούργησε καλύτερα.

```
results <- c(Tree=confusion_tree$overall['Accuracy'],
RandomForest=confusion_rf2$overall['Accuracy'],
KNN=confusion_knn$overall['Accuracy'], LDA=confusion_LDA$overall['Accuracy'])

par(mfrow=c(1,1))

barplot(results, ylim=c(0,1), ylab='Accuracy',names.arg=c('Tree',
'RandomForest','KNN','LDA'))
```



Παράρτημα κώδικα R

Case I

```
#get data
url =
'http://www.biz.uiowa.edu/faculty/jledolter/DataMining/protein.csv'
food <- read.csv(url)
head(food)

#K-means clustering
#clustering with white and red meat
set.seed(123456789)
grpMeat <- kmeans(food[,c("WhiteMeat","RedMeat")], centers=3,
nstart=10)
grpMeat

#show results per country
o=order(grpMeat$cluster)
data.frame(food$Country[o],grpMeat$cluster[o])

#plot results
plot(food$Red, food$White, type="n", , xlab="Red Meat", ylab="White
Meat")
text(x=food$Red, y=food$White,
labels=food$Country,col=grpMeat$cluster+1)

#clustering with 7 food types
set.seed(123456789)
grpProtein <- kmeans(food[,-1], centers=7, nstart=10)

#show results per country
o=order(grpProtein$cluster)
data.frame(food$Country[o],grpProtein$cluster[o])

#plot results
library(cluster)
clusplot(food[,-1],grpProtein$cluster,main='2D representation of the
Cluster solution', color=TRUE, shade=TRUE, labels=2, lines=0)

#hierarchical clustering
foodagg=agnes(food,diss=FALSE,metric="euclidian")
plot(foodagg, main='Dendrogram')

#cut tree at k=4
groups <- cutree(foodagg, k=4)
rect.hclust(foodagg, k=4, border="red")
```

Case II

```
#get data
url='https://raw.githubusercontent.com/brenden17/sklearnlab/master/face
book/snsdata.csv'
teens <- read.csv(url)

#explore data
head(teens,3)

dim(teens)

str(teens)

summary(teens$age)
summary(teens$gender)
summary(teens$grandyear)
summary(teens$friends)
summary(teens)

teens = na.omit(teens)
dim(teens)

interests <- teens[5:40]

interests_z <- as.data.frame(lapply(interests, scale))

#clustering
teen_clusters <- kmeans(interests_z, 5)

#data for each cluster
teen_clusters$size
teen_clusters$centers

pie(colSums(interests[teen_clusters$cluster==1,]),cex=0.7)

pie(colSums(interests[teen_clusters$cluster==2,]),cex=0.7)

pie(colSums(interests[teen_clusters$cluster==3,]),cex=0.7)

pie(colSums(interests[teen_clusters$cluster==4,]),cex=0.7)

pie(colSums(interests[teen_clusters$cluster==5,]),cex=0.7)
```

Case III

```
# Load data
data(iris)

## split into train-test
library(caret)
set.seed(49) #for replication purpose
# We use the dataset to create a partition (80% training 20% testing)
index <- createDataPartition(iris$Species, p=0.80, list=FALSE)
# select 20% of the data for testing
test <- iris[-index,]
# select 80% of data to train the models
train <- iris[index,]

## explore data
# Dimensions of the data
dim(train)
# Structure of the data
str(train)
# Summary of the data
summary(train)
# Levels of the prediction column
levels(train$Species)

## visualize data
library(ggplot2)
# Box plot to understand how the distribution varies by class of flower
par(mfrow=c(1,4))
for(i in 1:4) {
  boxplot(train[,i], main=names(train)[i])
}

#plots using ggplot2 library
# Scatter plot
g <- ggplot(data=train, aes(x = Petal.Length, y = Petal.Width))
g <-g +
  geom_point(aes(color=Species, shape=Species)) +
  xlab("Petal Length") +
  ylab("Petal Width") +
  ggtitle("Petal Length-Width")+
  geom_smooth(method="lm")
print(g)

g <- ggplot(data=train, aes(x = Sepal.Length, y = Sepal.Width))
g <-g +
  geom_point(aes(color=Species, shape=Species)) +
  xlab("Sepal Length") +
  ylab("Sepal Width") +
  ggtitle("Sepal Length-Width")+
  geom_smooth(method="lm")
print(g)

## Box Plot
box <- ggplot(data=train, aes(x=Species, y=Sepal.Length)) +
  geom_boxplot(aes(fill=Species)) +
  ylab("Sepal Length") +
```

```
  ggtitle("Iris Boxplot") +
  stat_summary(fun.y=mean, geom="point", shape=5, size=4)
print(box)

library(ggthemes)
## Histogram
histogram <- ggplot(data=train, aes(x=Sepal.Width)) +
  geom_histogram(binwidth=0.2, color="black", aes(fill=Species)) +
  xlab("Sepal Width") +
  ylab("Frequency") +
  ggtitle("Histogram of Sepal Width")+
  theme_economist()
print(histogram)

#classification with LDA
library(MASS)
model_LDA <- lda(Species ~ ., train)
model_LDA
#prediction with testdata
predict_LDA <- predict(model_LDA, test)
confusion_LDA <- confusionMatrix(predict_LDA$class, test$Species)
confusion_LDA

#classification with decision tree
library(rpart)
library(rpart.plot)

iristree <- rpart(formula = Species ~ ., data = train, control =
rpart.control(minsplit = 5, minbucket = 2), method = "class") # method
= "class" directs the function to treat Species as a categorical
variable.
print(iristree)
rpart.plot(iristree)

predict_tree <- predict(iristree, test, type="class")
confusion_tree <- confusionMatrix(predict_tree, test$Species)
confusion_tree

#classification with random forest
library(randomForest)
rf_baseline <- randomForest(Species ~ ., train, importance=TRUE)
rf_baseline

#plot variable importance
varImpPlot(rf_baseline)

predict_rf_baseline <- predict(rf_baseline, test)
confusion_RF <- confusionMatrix(predict_rf_baseline, test$Species)
confusion_RF

#change parameters of random forest
rf_2 <- randomForest(formula = Species ~ ., data = train, importance =
TRUE, ntree=1000, mtry=4)
rf_2
```

```
#plot variable importance
varImpPlot(rf_2)
predict_rf_2 <- predict(rf_2, test)
confusion_rf2 <- confusionMatrix(predict_rf_2, test$Species)
confusion_rf2

#knn prediction
library(class)
model_knn <- list()
accuracy_knn <- numeric()
#try k 1-10
for (i in 1:10) {
  model_knn[[i]] <- knn(train[,-5], test[,-5], train$Species, k=i,
prob=TRUE)
  accuracy_knn[i] <-
sum(model_knn[[i]]==test$Species)/length(test$Species)*100
}
accuracy_knn
#best k=4:
model_knn4 <- knn(train[,-5], test[,-5], train$Species, k=4, prob=TRUE)
confusion_knn <- confusionMatrix(model_knn4, test$Species)
confusion_knn

# summarize accuracy of models
results <- c(Tree=confusion_tree$overall['Accuracy'],
RandomForest=confusion_rf2$overall['Accuracy'],
KNN=confusion_knn$overall['Accuracy'],
LDA=confusion_LDA$overall['Accuracy'])
par(mfrow=c(1,1))
barplot(results, ylim=c(0,1), ylab='Accuracy',names.arg=c('Tree',
'RandomForest','KNN','LDA'))
```

Βιβλιογραφία

Accenture (2018), 'Redefine Banking with Artificial Intelligence', [online], Available from: https://www.accenture.com/_acnmedia/pdf-68/accenture-redefine-banking.pdf (Accessed 13/06/2020)

Allan M. Turing (1950), Computing Machinery and Intelligence, Mind, [online], October pp. 433-460, Available from: https://www.csee.umbc.edu/courses/471/papers/turing.pdf (Accessed 03/06/2020)

Avron Barr and Edward A. Feigenbaum (1981), 'The Handbook of Artificial Intelligence' Volume 1, William Kaufmann, Inc, [online], pp.1-17, Available from: https://www.sciencedirect.com/book/9780865760899/the-handbook-of-artificial-intelligence (Accessed 09/05/2020)

Berry J, (1994), 'Database Marketing', Business Week, [online], 5 September 5, pp. 56-62, Available from: https://scholar.google.com/scholar_lookup?title=Database%20Marketing%2C%20Business%20Week%2C%20September%205&publication_year=1994&author=J.%20Berry (Accessed 28/03/2020)

Carlos Costa, Maribel Yasmina Santos, (2017), 'The data scientist profile and its representativeness in the European e-Competence framework and the skills framework for the information age', International Journal of Information Management, ScienceDirect, [online], vol 37, issue 6, December, pp.726-734, Available from: https://www.sciencedirect.com/science/article/pii/S026840121730600X#bib0130 (Accessed 28/03/2020)

Cleveland William S (2001), 'Data Science: an action plan for expanding the technical areas of the field of statistics", International Statistical Review, [online],

pp,21-26, Available from: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1751-5823.2001.tb00477.x (Accessed 24/03/2020)

Computer Science Center (2018), 'Machine Learning', CSC, [online], 18 November, Available from: https://www.csc.com.gr/machine-learning-%CE%BC%CE%B7%CF%87%CE%B1%CE%BD%CE%B9%CE%BA%CE%AE-%CE%BC%CE%AC%CE%B8%CE%B7%CF%83%CE%B7-%CF%84%CE%B9-%CE%B5%CE%AF%CE%BD%CE%B1%CE%B9/ (Accessed 08/06/2020)

Davenport Thomas H. and D.J. Patil, (2012), 'Data scientist', Harvard Business Review, [online], October, Available from: https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century (Accessed 24/03/2020)

Dhar Vasant, (2013), 'Data science and prediction', Communications of the ACM, pp, 64-73, [online], December, Available from: https://dl.acm.org/doi/abs/10.1145/2500499 (Accessed 24/03/2020)

European Commission (2017), 'Final results of the European Data Market study measuring the size and trends of the EU data economy' [online], 2 May, Available from: https://ec.europa.eu/digital-single-market/en/news/final-results-european-data-market-study-measuring-size-and-trends-eu-data-economy Accessed 10/05/2020)

Foster Provost, Tom Fawcett (2013), 'Data Science and its Relationship to Big Data and Data-Driven Decision Making', Big Data [online], 13 Feb, Available from: https://www.liebertpub.com/doi/10.1089/big.2013.1508 (Accessed 24/03/2020)

Gottsegen Gordon (2019), 'Machine Learning Is Changing The Way Retailers Do Business', Built In, [online], 17 April, Available from: https://builtin.com/artificial-intelligence/machine-learning-ecommerce-retail (Accessed 13/06/2020)

Hemant Sharma, (2019), 'What Is Data Science? A Beginner's Guide To Data Science' Edureka, [online], 20 June Available from: https://www.edureka.co/blog/what-is-data-science/ (Accessed 26/03/2020)

IBM (2014), What is a data scientist? IBM website, Available from: https://www.ibm.com/gr-en (Accessed 26/03/2020)

In Lee, Yong Jae Shin (2020) 'Machine learning for enterprises: Applications, algorithm selection, and challenges', ScienceDirect, [online] Vol 63, Issue 2, March–April Pages 157-170, Available from: https://www.sciencedirect.com/science/article/pii/S0007681319301521 (Accessed 04/06/2020)

Intellipaat, (2019) 'Data Scientist Roles and Responsibilities' Intellipaat, [online], 22 October, Available from: https://intellipaat.com/blog/data-scientist-roles-and-responsibilities/ (Accessed 26/03/2020)

Jeanne G. Harris and Ray Eitel-Porter, (2015), Data scientists: 'As rare as unicorns', The Guardian, [online], 12 February, Available from: https://www.theguardian.com/media-network/2015/feb/12/datascientists-as-rare-as-unicorns (Accessed 26/03/2020)

Karina Gibert, Jeffery S. Horsburgh, Ioannis N. Athanasiadis, Geoff Holmesd, (2018), 'Environmental Data Science' Environmental Modelling & Software, ScienceDirect, [online], vol 106, pp.4-12, August, Available from: https://www.sciencedirect.com/science/article/pii/S1364815218301269#bib70 (Accessed 28/03/2020)

KDnuggets, (2018)'9 Must-have skills you need to become a Data Scientist', KDnuggets, [online], 18 May, Available from: https://www.kdnuggets.com/2018/05/simplilearn-9-must-have-skills-data-scientist.html (Accessed 28/03/2020)

Kotsibou Loukia (2019), 'Τι είναι το Cloud και cloud computing', CompuTertechInfo.gr, [online], 27 March, Anailable from: https://computertechinfo.gr/cloud-storage-computing/ (Accessed 21/06/2020)

Kumaresh Pattabiraman, (2019), 'Most Promising Jobs of 2019', LinkedIn's, [online], 10 January, Available from: https://blog.linkedin.com/2019/january/10/linkedins-most-promising-jobs-of-2019 (Accessed 26/03/2020)

Leandro DalleMule and Thomas H. Davenport, (2017), 'What's Your Data Strategy?', Harvard Business Review, [online], May–June Issue, Available from: https://hbr.org/2017/05/whats-your-data-strategy (Accessed 28/03/2020)

Martijn Theuwissen, (2015) 'The different data science roles in the industry' KDnuggets, [online], 15 November, Available from: https://www.kdnuggets.com/2015/11/different-data-science-roles-industry.html Accessed 26/03/2020)

Metallidis George S. (2019), 'Μια ματιά στις 5 από τις πιο δημοφιλείς γλώσσες προγραμματισμού για το 2019' Technode.gr, [online], 6 December, Avialable from: https://technode.gr/2019/12/06/popular-programming-languages-2019/ (Accessed 21/06/2020)

Naur Peter (1966), 'The Science of Datalogy' Letters to the editor, Communications of the ACM [online], vol 9, no 7, July, p.485, Available from: https://dl.acm.org/doi/pdf/10.1145/365719.366510 (Accessed 28/03/2020)

Patil DJ, (2011), 'Building data science teams', Radar, [online],16 September, Available from: http://radar.oreilly.com/2011/09/building-data-science-teams.html (Accessed 22/03/2020)

Paul Torfs & Claudia Brauer (2014), “Μια (πολύ) σύντομη εισαγωγή στην R”, Πανεπιστήμιο Wageningen, Ολλανδία, [online], 4 November, Available from: https://www.accenture.com/_acnmedia/pdf-68/accenture-redefine-banking.pdf (Accessed 10/08/2020)

Power J Daniel, (2016), ‘Data science: supporting decision-making’, Journal of Decision Systems, Received 22 Nov 2015, Accepted 24 Mar 2016, Published [online], 25 Apr, Available from: https://orsociety.tandfonline.com/doi/full/10.1080/12460125.2016.1171610#.XnlCN4gzbIU (Accessed 24/03/2020)

Rizzi W, Wang Z. Maria, & Zielinski, K (2018), ‘How machine learning can improve pricing performance’, McKinsey & Company, [online], 20 September, Available from: https://www.mckinsey.com/industries/financial-services/our-insights/how-machine-learning-can-improve-pricing-performance (Accessed 03/06/2020)

Russell, S. J., & Norvig, P. (2010). Artificial intelligence: A modern approach (3rd ed.). New Jersey: Pearson Education.

Vincent Granville (2013), ‘What does a data scientist do?’ Data Science Central, [online], 28 March, Available from: https://www.datasciencecentral.com/profiles/blogs/what-does-a-data-scientist-do (Accessed 28/03/2020)

Samuel A.l., (1959), ‘Some Studies in Machine Learning Using the Game of Checkers’, IBM Journal of Research and Development, [online], July, pp. 210 – 229, Τόμος: 3, Τεύχος: 3, Available from: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5392560 (Accessed 14/06/2020)

SEnDIng Online, Ανοιχτά διαδικτυακά μαθήματα, “Statistics for Data Science”, Available from: http://mooc.sending-project.eu/, (Accessed 10/08/2020)

Stadelmann Thilo, Kurt Stockinger, Gundula Heinatz Bürki, Martin Braschler, (2019), [online], 14 June, pp.31-45, Available from: https://link.springer.com/chapter/10.1007%2F978-3-030-11821-1_3 (Accessed 28/03/2020)

Stevan Harnad (2008), ‘The Annotation Game: On Turing (1950) on Computing,Machinery, and Intelligence’, Université du Québec à Montréal, [online], January, pp. 1, Available from: file:///C:/Users/User/Downloads/The_Annotation_Game_On_Turing_1950_on_Computing_Ma.pdf (Accessed 03/06/2020)

Svetlana Sicular, (2012), ‘The quest for data scientists’ The Australian Business Review, [online], 11 July, Available from: https://www.theaustralian.com.au/business/business-spectator/the-quest-for-datascientists/news-story/eab27147e92d0011520f5adb32010e43 (Accessed 28/03/2020)

Tom Mitcel (1997), Machine Learning, McGraw-Hill Science/Engineering/Math, [online], 1 March, pp. 2, Available from: http://profsite.um.ac.ir/~monsefi/machine-learning/pdf/Machine-Learning-Tom-Mitchell.pdf (Accessed 03/06/2020)

Tukey John W.,(1962), ‘The future of data analysis’, The Annals of Mathematical Statistics, [online], vol 33, no 1, p.1-67, Available from: https://projecteuclid.org/euclid.aoms/1177704711 (Accessed 28/03/2020)

Warden Pete,(2011), 'Why the term data science is flawed but useful', Radar , [online], 9 May, Available from: http://radar.oreilly.com/2011/05/data-science-terminology.html (Accessed 24/03/2020)

Wasserman, L. (2013). All of statistics: A concise course in statistical inference. New York: Springer Science & Business Media.

Wu Jeff, (1997), Statistics = Data Science? Inaugural lecture at university of Michigan, Ann Arbor, [online], Available from: https://www2.isye.gatech.edu/~jeffwu/presentations/datascience (Accessed 22/03/2020)

"Εισαγωγή στην R", [online], Available from: https://repository.kallipos.gr/bitstream/11419/2967/1/02_chapter_02.pdf, (Accessed 10/08/2020)

Γεροθανάσης Ε. – Μπέκος Ε. (2012), 'Κατασκευή ταξινομητών weighted kNN με metric ball trees για εφαρμογές ανακάλυψης γνώσης από βάσεις δεδομένων' Oracle, ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΣΕΡΡΩΝ Available at: http://apothesis.teicm.gr/xmlui/bitstream/handle/123456789/833/gerothanasis.pdf?sequence=1&isAllowed=y (Accessed 17/06/2020)

Καλατζής Ιωάννης, (2017), 'Ανάλυση Γραμμικής Διάκρισης' , Πανεπιστήμιο Δυτικής Αττικής, [online], Available from: https://medisp.bme.uniwa.gr/eclass/modules/document/file.php/MTMBIT101/%CE%A5%CE%9B%CE%99%CE%9A%CE%9F%20%CE%99.%20%CE%9A%CE%91%CE%9B%CE%91%CE%A4%CE%96%CE%97%20%28DESCRIPTIVE%20STATISTICS%2C%20HYPOTHESIS%20TESTING%2C%20CLUSTERING%2C%20PCA%2C%20LDA%29/5.%20%CE%91%CE%BD%CE%AC%CE%BB%CF%85%CF%83%CE%B7%20%CE%93%CF%81%CE%B1%CE%BC%CE%BC%CE%B9%CE%BA%CE%AE%CF%82%20%CE%94%CE%B9%CE%AC%CE%BA%CF%81%CE%

B9%CF%83%CE%B7%CF%82%20%28LDA%29/LDA%20%28I.%20Kalatzis%202017%29.pdf (Accessed 04/11/2020)

“Οδηγός χρήσης της γλώσσας R”, [online], Available from:http://users.auth.gr/cmoi/Metaptyxiako/ThGraphs&DynamSyst/cmoi-R-guide.pdf, (Accessed 10/08/2020)

Παπακωνσταντίνου Ε. – Καΐτσα (1995), ‘Στατιστική’, Ίδρυμα Ευγενίδου, [online], pp.2, Available from: https://www.eef.edu.gr/media/2320/e_g00050.pdf (Accessed 22/03/2020)

Χατζηδάκης Στυλιανός (2014), Λήψη αποφάσεων κατά Bayes’,[online] Available at: https://www.projectrhea.org/rhea/images/6/69/Slecture_Derivation_of_Bayes_rule_in_Greek.pdf (Accessed 17/06/2020)

Χατζημιχαηλίδης Δ. (2019), ‘Τι είναι το Cloud’, Digital Challenge, [online], 7 February, Available from: https://www.dicha.gr/blog/ti-einai-to-cloud (Accessed 21/06/2020)

Οι μελέτες περίπτωσης (case studies) είναι διαθέσιμες στους παρακάτω διαδικτυακούς ιστότοπους:

https://rpubs.com/gabrielmartos/ClusterAnalysis

https://rpubs.com/rpadebet/269829

http://rstudio-pubs-static.s3.amazonaws.com/420656_c17c8444d32548eba6f894bcbdffcaab.html

Πνευματικά δικαιώματα

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1988 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα εργασία αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον.

Γιαλελή Αγγελική, [2020]