Διπλωματική Εργασία

Ανάλυση Μεγάλων Δεδομένων και Δυναμική Οπτικοποίηση Αποτελεσμάτων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

UNIVERSITY OF PIRAEUS

Ψηφιακά Συστήματα
Πρόγραμμα Μεταπτυχιακών Σπουδών
Μεγάλα Δεδομένα και Αναλυτική

Παναγιώτης Τσάκαλης

Υπεύθυνος Καθηγητής
Δημοσθένης Κυριαζής

Πίνακας Περιεχομένων

Πίνακας Περιεχομένων 2

1 Εισαγωγή 6

1.1 Προσδιορισμός προβλήματος 8

2 Serverless Architectures 11

2.1 Serverless Solutions 13

2.1.1 AWS Lambda 14

2.1.2 Microsoft Azure Functions 14

2.1.3 Google Cloud Functions 15

2.1.4 IBM Bluemix OpenWhisk 15

2.1.5 Iron.io Ironworker 16

2.1.6 Auth0 Webtask 16

2.1.7 Galactic Fog Gestal Laser 17

2.1.8 Apache OpenWhisk 17

2.2 Big Data Analytics 18

2.3 Python 21

2.3.1 Numpy 22

2.3.2 Pandas 22

2.3.3 Matplotlib 23

2.3.4 IPython 23

2.3.5 SciPy 23

2.3.6 Cython 23

2.3.7 R Language 23

2.3.8 Jupyter 24

2.3.9 Scikit-learn 27

2.3.10 Regular Expression (RE) 28

2.3.11 Seaborn 29
2.3.12 NTLK (Word_Tokenize) 30
2.3.13 Counter 31
2.3.14 Stop Words 32
2.4 Αλγόριθμοι Analytics 33
2.4.1 Δέντρα ταξινόμησης και παλινδρόμησης 33
2.4.2 Ομαδοποίηση (Clustering) 34
2.4.3 Apriori 36
2.4.4 Ανάλυση RFM 37
2.4.5 Kmeans 39
3 Business Intelligence 42
3.1 Οργανωτική απόδοση 48
3.2 Επιπτώσεις του Business Intelligence 48
3.3 Ανταγωνιστικό Πλεονέκτημα του Business Intelligence 49
4 Tableau 51
5 Περιγραφή Προσέγγισης 53
5.1 Τεχνικές και Αλγόριθμοι 53
5.2 Περιγραφή Συνόλου Δεδομένων 55
6 Αποτελέσματα 57
6.1 Τα αποτελέσματα των πελατών με μοντέλο RFM 57
6.2 Αποτελέσματα υλοποίησης του αλγόριθμου Kmeans 58
6.3 Αποτελέσματα του Αλγορίθμου Apriori 59
6.4 Αποτελέσματα από Tableau 61
7 Συμπεράσματα 72
7.1 Προοπτική μετά την εφαρμογή της ομαδοποίησης k-means: 72
7.2 Συμπεράσματα οπτικοποίησης μέσω του Tableau 72

Βιβλιογραφία 74

Abstract

Η παρούσα διπλωματική εργασία ερευνά τις χωρίς διακομηστές αρχιτεκτονικές που υπάρχουν στις μέρες μας, τα πλεονεκτήματα και τις δυνατότητες που προσφέρουν στην ανάλυση μεγάλου όγκου δεδομένων. Πιο συγκεκριμένα πραγματεύεται την υλοποίηση αλγορίθμων analytics πάνω σε ένα σύνολο δεδομένων που αναφέρεται σε συναλλαγές καταστήματος με βάση το Ηνωμένο Βασίλειο με σκοπό την εξαγωγή συμπερασμάτων σχετικά με τους πελάτες, τις χώρες προέλευσης, τα κέρδη της επιχείρησης και προσπαθέι να αναδείξει ενδιαφέροντα αποτελέσματα καλυτέρευσης της παραγωγικής διαδικασίας. Τρέχοντας τις διεργασίες πάνω στο Apache OpenWhisk πραγματοποιείται η λεγόμενη ανάλυση καλαθιού του καταναλωτή, η ομαδοποίηση με βάση τις προτιμήσεις τους και το πως μπορεί η εταιρεία να διαχειριστεί το πελατολόγιο της με σκοπό την αύξηση των εσόδων. Τέλος γίνεται προσπάθεια πρόβλεψης του αριθμού παραγγελιών για τους επόμενους μήνες με σκοπό την καλύτερη διαχείρηση και την κατάστρωση στρατηγικών από το επιχειρησιακό κομμάτι της επιχείρησης

1 Εισαγωγή

Μετά από την καθοδήγηση της AWS Lambda (2020), αναδύθηκαν και εισήχθησαν υπηρεσίες χωρίς διακομιστές όπως οι Apache OpenWhisk (2020), Azure Functions (2020), Google Cloud Functions (2020), Iron.io IronFunctions (2020) και OpenLambda (2020), υπηρεσίες νέφους (εφεξής cloud) όπου η λογική εφαρμογής χωρίζεται σε λειτουργίες και εκτελείται ως ανταπόκριση σε γεγονότα. Αυτά τα συμβάντα μπορούν να ενεργοποιηθούν από πηγές εξωτερικές της πλατφόρμας cloud, αλλά επίσης να εμφανίζονται συνήθως εσωτερικά μεταξύ των προσφερόμενων υπηρεσιών της πλατφόρμας cloud, επιτρέποντας στους προγραμματιστές να συνθέτουν εύκολα εφαρμογές που διανέμονται σε πολλές υπηρεσίες εντός ενός cloud.

Ο υπολογιστής χωρίς διακομιστές είναι μια μερική υλοποίηση ενός ιδεατόμου, το οποίο καθορίζει τις εφαρμογές με δράσεις και τα γεγονότα που τα ενεργοποιούν. Αυτή η γλώσσα θυμίζει ενεργά συστήματα βάσεων δεδομένων και η βιβλιογραφία που βασίζεται σε γεγονότα έχει θεωρήσει για κάποιο χρονικό διάστημα γενικά συστήματα υπολογιστών στα οποία οι πράξεις επεξεργάζονται αντιδραστικά σε ροές συμβάντων (Schmidt et al., 2008). Οι πλατφόρμες λειτουργιών χωρίς διακομιστές αγκαλιάζουν πλήρως αυτές τις ιδέες, ορίζοντας ενέργειες μέσω απλών αφαίρεσης λειτουργιών και δημιουργώντας λογική επεξεργασίας συμβάντων στα σύννεφα. Η IBM αντανακλά έντονα αυτές τις έννοιες στην πλατφόρμα OpenWhisk (τώρα Apache OpenWhisk), στην οποία οι λειτουργίες καθορίζονται ρητά από την άποψη του γεγονότος, της ενεργοποίησης και της δράσης (Baldini et al, 2016).

Πέρα από την ιδέα που βασίζεται στην εκδήλωση, οι συζητήσεις σχεδίασης μετατοπίζονται προς τη διαχείριση περιεχόντων και τις στρατηγικές ανάπτυξης λογισμικού που χρησιμοποιούνται για τη μόχλευση κεντρικής υποδομής λειτουργίας. Το Iron.io χρησιμοποιεί το Docker για να αποθηκεύει τα δοχεία λειτουργίας σε ιδιωτικά μητρώα, τραβώντας και τρέχοντας τα δοχεία όταν απαιτείται εκτέλεση (Dwyer, 2016). Η ομαδική εργασία στην πλατφόρμα OpenLambda παρουσιάζει μια ανάλυση των πλεονεκτημάτων κλιμάκωσης των υπολογιστών χωρίς διακομιστές, καθώς και μια ανάλυση απόδοσης των διαφόρων μεταβάσεων περιεχόντων (Hendrickson et al, 2016). Άλλες αναλύσεις απόδοσης έχουν μελετήσει την επίδραση του χρόνου εκτέλεσης

γλώσσας και του αντίκτυπου του VPC στους χρόνους έναρξης του AWS Lambda (Vojta, 2016) και μέτρησαν το δυναμικό της AWS Lambda για παράλληλες επιστημονικές υπολογιστικές εφαρμογές υψηλής απόδοσης (Jonas, 2016).

Η χρήση υπολογιστή χωρίς διακομιστές έχει αποδειχθεί κατάλληλη για εφαρμογές IoT, οι οποίες διασταυρώνονται με τη συνομιλία υποδομής υπολογιστών για edge / fog. Καταβάλλονται συνεχείς προσπάθειες για την ενσωμάτωση της πληροφορικής χωρίς διακομιστές σε μια "ιεραρχία των κέντρων δεδομένων" για την ενίσχυση της προβλεπόμενης διάδοσης συσκευών διαδικτύου (Lara et al, 2016). Η AWS προσχώρησε πρόσφατα στον τομέα αυτό με το προϊόν Lambda @ Edge (AWS, 2020), το οποίο επιτρέπει στους προγραμματιστές εφαρμογών να τοποθετούν περιορισμένες λειτουργίες Lambda στις άκρες κόμβων. Η AWS έχει επιδιώξει και άλλες επεκτάσεις υπολογιστών χωρίς διακομιστές, συμπεριλαμβανομένης της Greengrass (2020), η οποία παρέχει ένα ενιαίο μοντέλο προγραμματισμού σε διάφορες λειτουργίες IoT και Lambda. Ο υπολογιστής χωρίς διακομιστές επιτρέπει στους προγραμματιστές εφαρμογών να αποσυνθέτουν μεγάλες εφαρμογές σε μικρές λειτουργίες, επιτρέποντας στα εξαρτήματα των εφαρμογών να κλιμακώνονται μεμονωμένα, αλλά αυτό παρουσιάζει ένα νέο πρόβλημα στη συνεκτική διαχείριση μιας μεγάλης σειράς λειτουργιών. Η AWS εισήγαγε πρόσφατα τις λειτουργίες βημάτων (AWS, 2020), που διευκολύνουν την οργάνωση και την απεικόνιση της αλληλεπίδρασης των λειτουργιών. Η εφαρμογή της υπολογιστικής χωρίς διακομιστή είναι ένας ενεργός τομέας ανάπτυξης. Οι προηγούμενες εργασίες για υπολογιστές χωρίς διακομιστές μελέτησαν παραδείγματα προγραμματισμού χωρίς διακομιστές, όπως οι καταρράκτες λειτουργιών, και πειραματίστηκαν με την ανάπτυξη μονολιθικών εφαρμογών σε πλατφόρμες χωρίς διακομιστές (McGrath et al, 2016). Άλλες εργασίες έχουν μελετήσει την αρχιτεκτονική των κλιμακωτών ρομπότ συνομιλίας («scalable chatbots») σε πλατφόρμες χωρίς διακομιστές (Yan et al, 2016). Υπάρχουν πολλά έργα που αποσκοπούν στην επέκταση της λειτουργικότητας των υφιστάμενων πλατφορμών χωρίς διακομιστές. Το Lambdash (Hammond, 2017) είναι ένα εργαλείο που επιτρέπει την εύκολη εκτέλεση εντολών κελύφους σε δοχεία AWS Lambda, επιτρέποντας στους προγραμματιστές να εξερευνήσουν το περιβάλλον λειτουργίας Lambda. Άλλες προσπάθειες όπως το Apex (Apex, 2020) και το Sparta (Sparta, 2020) επιτρέπουν στους

χρήστες να αναπτύξουν λειτουργίες στο AWS Lambda σε μη υποστηριζόμενες γλώσσες, όπως είναι η Go.

Η υπολογιστική χωρίς διακομιστή υποστηρίζεται συχνά ως εργαλείο εξοικονόμησης κόστους και υπάρχουν πολλά έργα που αναφέρουν ευκαιρίες εξοικονόμησης κόστους στην ανάπτυξη μικροεπιχειρήσεων σε πλατφόρμες χωρίς διακομιστές και όχι στην κατασκευή παραδοσιακών εφαρμογών (Villamizar et al, 2016, Wagner & Sood, 2016). Άλλοι προσπάθησαν να υπολογίσουν τα σημεία με τα οποία οι εφαρμογές χωρίς διακομιστές ή εικονικές μηχανές γίνονται πιο οικονομικά αποδοτικές (Warzon, 2016). Η υπολογιστική χωρίς διακομιστές αποκτά όλο και μεγαλύτερη σημασία, με τον Gartner να αναφέρει ότι «η αξία του (serverless computing) έχει αποδειχθεί σαφώς, χαρτογραφεί φυσικά την αρχιτεκτονική λογισμικού μικροεπιχειρήσεων και βρίσκεται σε μια τροχιά αυξημένης ανάπτυξης και υιοθεσίας» (Lowery, 2016). Ο Forrester υποστηρίζει ότι «οι σημερινές επενδύσεις στο PaaS οδηγούν σε υπολογιστική χωρίς διακομιστές», βλέποντας την υπολογιστική χωρίς διακομιστές ως την επόμενη γενιά των παρεκκλίσεων από υπηρεσίες cloud (Hammond et al, 2016). Ο υπολογιστής χωρίς διακομιστές αναπτύσσεται γρήγορα σε πολλούς παρόχους σύννεφων και τροφοδοτεί όλο και περισσότερους εφαρμογές κινητής τηλεφωνίας και IoT. Καθώς το εύρος και η δημοτικότητά του επεκτείνεται, είναι σημαντικό να διασφαλιστεί ότι τα βασικά χαρακτηριστικά απόδοσης των πλατφόρμων χωρίς διακομιστές είναι υγιή.

1.1 Προσδιορισμός προβλήματος

Στις εταιρείες ηλεκτρονικού εμπορίου, όπως οι ηλεκτρονικές λιανικές πωλήσεις, ο κατακερματισμός των πελατών είναι απαραίτητος για να κατανοήσουμε τις συμπεριφορές των πελατών. Επωφελείται από τα αποκτηθέντα δεδομένα πελατών, όπως αυτό που έχουμε στην περίπτωσή μας, τα δεδομένα συναλλαγών προκειμένου να διαιρέσουμε τους πελάτες σε ομάδες.

Στόχος μας σε αυτή τη διατριβή είναι να συγκεντρώσουμε τους πελάτες μας για να αποκτήσουν πληροφορίες σχετικά με:

• Αύξηση εσόδων (Γνωρίζοντας πελάτες που παρουσιάζουν το μεγαλύτερο μέρος των εσόδων μας)
• Αύξηση της διατήρησης των πελατών
• Ανακαλύπτοντας τάσεις και μοτίβα
• Καθορισμός των πελατών σε κίνδυνο

Επιπλέον, στην παρούσα διατριβή χρησιμοποιήθηκε η ανάλυση του καλαθιού αγοράς, προκειμένου να προσδιοριστούν οι κανόνες σύνδεσης που ανακαλύφθηκαν στα δεδομένα συναλλαγών χρησιμοποιώντας μέτρα ενδιαφέροντος, βασισμένα στην έννοια των ισχυρών κανόνων.
Η ανάλυση της αγοράς καλαθιού είναι μία από τις βασικές τεχνικές που χρησιμοποιούν οι μεγάλοι έμποροι λιανικής πώλησης για να αποκαλύψουν συσχετισμούς μεταξύ αντικειμένων. Λειτουργεί αναζητώντας συνδυασμούς στοιχείων που συμβαίνουν συχνά στις συναλλαγές. Για να το θέσουμε με άλλο τρόπο, επιτρέπει στους εμπόρους λιανικής πώλησης να εντοπίζουν τις σχέσεις μεταξύ των αντικειμένων που αγοράζουν οι άνθρωποι.
Οι έμποροι λιανικής πώλησης μπορούν να χρησιμοποιήσουν τις γνώσεις που αποκτήθηκαν από το MBA με πολλούς τρόπους, όπως:
• Ομαδοποίηση προϊόντων που συνυπάρχουν στο σχεδιασμό της διάταξης ενός καταστήματος για να αυξήσει την πιθανότητα σταυροειδών πωλήσεων.
• Οδηγώντας μηχανές ηλεκτρονικών συμβουλών ("οι πελάτες που αγόρασαν αυτό το προϊόν είδαν επίσης αυτό το προϊόν.
• Στόχευση καμπανιών μάρκετινγκ στέλνοντας διαφημιστικά κουπόνια στους πελάτες για προϊόντα που σχετίζονται με αντικείμενα που αγόρασαν πρόσφατα.

Στη συνέχεια θέλοντας να εξετάσουμε το business κομμάτι έγινε ανάλυση του συνόλου δεδομένων με βάση τις παραγγελίες που γίνονται ανά χώρα και μέρα προσπαθώντας να βγουν διάφορα συμπεράσματα για προτιμήσεις καταναλωτών πάνω σε

συγκεκριμένα προιόντα, για το ποιες μέρες προτιμούν να κάνουν τις παραγγελίες τους, για το κέρδος της εταιρείας απο τις ολοκληρωμένες παραγγελίες αλλά και πρόβλεψη για τον αριθμό παραγγελιών τους επόμενους μήνες

2 Serverless Architectures

Η αρχιτεκτονική χωρίς διακομιστές, γνωστή και ως «function as a service» (FAAS) ή Back-end As A Service (BAAS) χρησιμοποιείται κυρίως για μικρές συναλλαγές από εταιρείες (Baldini et al, 2017). Το όνομα "serverless", είναι αρκετά παραπλανητικό, καθώς δεν σημαίνει ότι δεν εμπλέκονται διακομιστές (Wagner, 2015). Για τις επιχειρήσεις, αντί να χρειάζεται να δημιουργήσουν το δικό τους δίκτυο πελατών-εξυπηρετητή, το οποίο είναι πολύ δαπανηρό, μπορούν πλέον να προσλάβουν διακομιστές για χρήση κατόπιν ζήτησης από εταιρείες όπως το Amazon. Το Amazon προσφέρει μια υπηρεσία που ονομάζεται Lambda και χρεώνει τους χρήστες για τα χιλιοστά του δευτερολέπτου. Η τιμολόγησή του είναι περίπου 0.0000002 δολάρια ανά χιλιοστό του δευτερολέπτου (Jones, 2016). Συνήθως μια επιχείρηση θα έχει μια εφαρμογή για έναν χρήστη, όπως μια εφαρμογή πρόβλεψης καιρού. Όταν το αίτημα αυτό αποσταλεί στην πύλη API (Interface Programming Interface) της υπηρεσίας Lambda, θα ξεκινήσει ένα διακομιστή, θα εκτελέσει τον κώδικα και θα επιστρέψει την επιθυμητή τιμή (Wagner, 2015). Οι περισσότεροι διακομιστές θα υπάρχουν μόνο για περίπου 40 χιλιοστά του δευτερολέπτου με 4,5 δευτερόλεπτα που είναι η μεγαλύτερη διάρκεια ζωής του διακομιστή (Jones, 2016).

Η διαχείριση ταυτότητας και πρόσβασης AWS (IAM) επιτρέπει σε μια οργάνωση να καθορίζει δικαιώματα για ομάδες και άτομα (Cui, 2017). Μια λειτουργία Lambda θα αντιστοιχεί σε ένα σύνολο διαπιστευτηρίων (Cui, 2017). Το IAM διαθέτει πολλές λειτουργίες που μπορούν να χρησιμοποιηθούν όπως έλεγχος ταυτότητας πολλαπλών παραμέτρων, ώρα της ημέρας στον οποίο μπορεί να αποκτηθεί πρόσβαση ο διακομιστής και ποιες διευθύνσεις IP επιτρέπεται να χρησιμοποιούν το AWS (Pirtle, 2017). Το IAM διαμορφώνεται και συντηρείται από τον πελάτη, όχι από το Amazon (Jones, 2016). Μια πύλη διασύνδεσης προγραμματισμού εφαρμογών ενεργεί ουσιαστικά ως "μπροστινή πόρτα" για όλες τις εφαρμογές και τις υπηρεσίες back end (Pirtle, 2017). Η πύλη API του Amazon λειτουργεί ως διαχειριστής για όλη την κυκλοφορία API. Όλες οι κλήσεις API που παράγουν οι επιχειρήσεις, συμπεριλαμβανομένων όλων των ελέγχων πιστοποίησης από το IAM, διακινούνται μέσω αυτής της πύλης (Pirtle, 2017).

Οι περισσότερες εταιρείες πιστεύουν τώρα ότι πρέπει μόνο να επικεντρωθούν στην ασφάλεια στο επίπεδο εφαρμογής (Podjarny, 2017). Μία από τις κορυφαίες ανησυχίες στο επίπεδο της εφαρμογής είναι η "Ενσωμάτωση Δεδομένων Ενεργοποίησης Συμβάντων", η οποία είναι η έγχυση κώδικα από οποιοδήποτε αριθμό συμβάντων όπως το IoT, η αποθήκευση σε νέφος κλπ. (Segal, 2018; Sheridan, 2018). Σε γενικές γραμμές, τα συστήματα χωρίς διακομιστές είναι πιο δύσκολο να διεισδύσουν από τους παραδοσιακούς διακομιστές (Jones, 2016). Αυτό οφείλεται στο ότι υπάρχει λιγότερο κοινός κώδικας και οι κακοί παραγωγοί δεν μπορούν να χρησιμοποιήσουν γνωστά πλαίσια (Jones, 2016). Τόσο οι χρήστες όσο και οι λειτουργίες είναι απομονωμένες και δεν υπάρχει κανένας διαχειριστής συστήματος που να κλιμακώνεται για να αποκτήσει τον πλήρη έλεγχο (Jones, 2016). Λόγω της μικρής διάρκειας ζωής των εξυπηρετητών αυτών, δεν χρειάζεται πλέον να ανησυχούν για τους μακροχρόνιους διακομιστές με συνεχή εξάτμιση δεδομένων (Jones, 2016). Η Lambda επίσης δεν έχει συνεχή σύνδεση στο διαδίκτυο και κάθε λειτουργία έχει αυστηρή άδεια σε ό, τι επιτρέπεται να έχει πρόσβαση (Jones, 2016)

Η αρχιτεκτονική χωρίς διακομιστές είναι εξίσου ασφαλής με την τρίτη εταιρεία που χρησιμοποιείτε. Αν και εξακολουθεί να είναι ευάλωτη σε πολλές κοινές εκμεταλλεύσεις, όπως αδρανοποιημένες εισροές, ατέλειες και σφάλματα, η πραγματική απειλή για τις επιχειρήσεις που χρησιμοποιούν serverless τεχνολογίες βρίσκεται στους ανθρώπους που τη διαμορφώνουν και τη διαχειρίζονται. Όπως φαίνεται στις γνωστές ευπάθειες από τη δοκιμή διείσδυσης, ο καλύτερος και ευκολότερος τρόπος πρόσβασης είναι μέσω της λανθασμένης διαμόρφωσης του IAM. Τα πρόσφατα γεγονότα δείχνουν ότι είναι πολύ εύκολο για τους ανθρώπους να κακοποιούν απλές βάσεις δεδομένων όπως η διαρροή του Πενταγώνου AWS (Muncaster, 2017). Στην περίπτωση του Lambda, εάν μια επιχείρηση έχει άλλες υπηρεσίες AWS, αυτό θα μπορούσε να σημαίνει την έκθεση ολόκληρου του εταιρικού δικτύου. Σχεδόν κάθε τρίτο μέρος χωρίς διακομιστές θα σας καθοδηγήσει με ανασφαλείς προεπιλογές (Jones, 2016). Αυτό σημαίνει ότι οι επιχειρήσεις πρέπει να έχουν εκπαιδευμένο προσωπικό για τη διαμόρφωση του συστήματος. Πρέπει να γνωρίζουν πώς λειτουργεί το σύστημα, όχι μόνο να ακολουθεί την τεκμηρίωση. Όπως και στην περίπτωση ανθρώπων που χρησιμοποιούν παλιά τεκμηρίωση που άφησε στο

φόρουμ του Amazon, το οποίο οδήγησε σε πολλά τρωτά σημεία στα συστήματά τους (Jones, 2016). Αυτό σημαίνει ότι η ασφάλεια των επιχειρήσεων εξαρτάται κυρίως από την ικανότητα και την κατάρτιση των υπαλλήλων της.

2.1 Serverless Solutions

Η αξιολόγηση των επιδόσεων τεχνολογίας νέφους, συμπεριλαμβανομένων των ετερογενών υποδομών, αποτέλεσε αντικείμενο προηγούμενης έρευνας. Ένα εξαιρετικό παράδειγμα είναι το (Isoup et al, 2011), όπου συγκρίνονται πολλαπλές τεχνολογίες νέφους από την οπτική γωνία εφαρμογών υπολογιστικής πολλών εργασιών. Αρκετές υποθέσεις σχετικά με την απόδοση δημόσιων σύννεφων συζητούνται σε μια εκτενή μελέτη που παρουσιάζεται στο (Leitner & Cito, 2016). Πιο πρόσφατες μελέτες επικεντρώνονται π.χ. σε περιπτώσεις "burstable" (Leitner & Scheuner, 2015), οι οποίες είναι φθηνότερες από τις συνήθεις περιπτώσεις αλλά έχουν διαφορετικά χαρακτηριστικά απόδοσης και αξιοπιστίας. Έχει επίσης αναλυθεί η απόδοση εναλλακτικών λύσεων cloud, όπως το Platform-as-a-Service (PaaS). Π.χ., οι Malawski και συνεργάτες (2013) και οι Prodan και συνεργάτες (2012) επικεντρώθηκαν στο Google App Engine από την οπτική των επιστημονικών εφαρμογών έντασης CPU. Μια λεπτομερής απόδοση και σύγκριση κόστους των παραδοσιακών σύννεφων με μικροεπιχειρήσεις και της αρχιτεκτονικής AWS Lambda χωρίς διακομιστές παρουσιάζεται στο Villamizaer και συνεργάτες (2016), χρησιμοποιώντας μια επιχειρησιακή εφαρμογή. Ομοίως, στο Wagner & Sood (2016) οι συγγραφείς συζητούν τα πλεονεκτήματα της χρήσης υπηρεσιών cloud και AWS Lambda για συστήματα που απαιτούν μεγαλύτερη ανθεκτικότητα. Μια ενδιαφέρουσα συζήτηση για το μοντέλο χωρίς διακομιστές δίνεται από τους McGrath και συνεργάτες (2016), όπου οι μελέτες περιπτώσεων είναι η εφαρμογή ιστολογίων (blogs) και διαχείρισης μέσων. Ένα παράδειγμα τιμής και απόδοσης των λειτουργιών νέφους παρέχεται επίσης στο Spillner & Snafu (2017), που περιγράφει το Snafu, μια νέα εφαρμογή του μοντέλου FaaS, που μπορεί να αναπτυχθεί σε ένα cluster Docker στο AWS. Η απόδοσή του και το κόστος του συγκρίνεται με το AWS Lambda με τη χρήση αναδρομικού δείκτη αναφοράς Fibonacci.

Για τους σκοπούς του παρόντος εγγράφου εντοπίστηκαν επτά πλατφόρμες υπολογιστικής cloud computing για εταιρίες. Συγκεκριμένα: AWS Lambda, Google Cloud Functions, Microsoft Azure Functions, IBM Bluemix OpenWhisk, Iron.io Ironworker, Auth0 Webtask, Galactic Fog Gestal Laser και Apache OpenWhisk.

2.1.1 AWS Lambda

Το AWS Lambda είναι υπηρεσία ατελούς επεξεργασίας υπολογιστών της Amazon Web Service που βασίζεται σε διάφορες προσφορές σύννεφων της AWS. Τα αναφερθέντα συμβάντα ενεργοποίησης περιελάμβαναν μεταφορτώσεις εικόνων, δραστηριότητες εντός εφαρμογής, κλικ στο δικτυακό τόπο, εξόδους από συνδεδεμένες συσκευές ή άλλα προσαρμοσμένα αιτήματα. Αρχικά, η Amazon Web Services υπογράμμισε ότι η AWS Lambda θα εξαλείψει την ανάγκη παροχής ή διαχείρισης εικονικών διακομιστών και αυτόματης κλιμάκωσης σε πολλαπλές δικαιοδοσίες (Ζώνες Διαθεσιμότητας). Οι υπηρεσίες Amazon Web Services ανέφεραν μια σειρά περιπτώσεων χρήσης, όπως επεξεργασίας δεδομένων (επεξεργασία αρχείων σε πραγματικό χρόνο, επεξεργασία ροής σε πραγματικό χρόνο, Extract Trasform and Load - ETL) και backend χωρίς διακομιστές (IoT, Mobile και Web). Οι συνδεδεμένοι χρήστες υψηλού προφίλ περιλαμβάνουν το Netflix (ανακατασκευή, παρακολούθηση, αποκατάσταση καταστροφών και συμμόρφωση), το SPS Commerce (επεξεργασία πληροφοριών), το Earth Network (ανίχνευση δεδομένων αισθητήρων, παρακολούθηση και πρόβλεψη), το Vidroll (διαφήμιση σε πραγματικό χρόνο)), το Seattle Times (αλλαγή μεγέθους εικόνων), το Zillow (κινητές μετρήσεις σε πραγματικό χρόνο), το Bustle (κινητό και διαδικτυακό backend) και το Major Media Baseball Advanced Media (ανάλυση δεδομένων, μετρήσεις παικτών και παιχνιδιών).

2.1.2 Microsoft Azure Functions

Το Microsoft Azure Functions διατέθηκε σε περιορισμένη κυκλοφορία τον Μάρτιο του 2016 και στη γενική έκδοση τον Νοέμβριο του 2016. Σχεδιάστηκε για να επεκτείνει την υπάρχουσα πλατφόρμα εφαρμογών Azure με δυνατότητες για την εφαρμογή κώδικα που ενεργοποιείται από συμβάντα που συμβαίνουν σε υπηρεσίες Azure ή συστήματα τρίτου μέρους. Η Microsoft παραθέτει τις ακόλουθες περιπτώσεις χρήσης

για τις λειτουργίες Azure - επεξεργασία με βάση το χρόνο (Cron workloads), ενεργοποιήσεις Azure, επεξεργασία συμβάντων Software-as-a-Service (SaaS), backends κινητών, επεξεργασία σε πραγματικό χρόνο (IoT) αυτόματα μηνύματα ηλεκτρονικού ταχυδρομείου. Οι αναφερόμενοι χρήστες υψηλού προφίλ περιλαμβάνουν τους Accuweather (φόρτου εργασίας Cron) και το Plexure (αυτόματη κλιμάκωση λογισμικού).

2.1.3 Google Cloud Functions

Η Google κυκλοφόρησε το Google Cloud Functions σχετικά ήσυχα το Φεβρουάριο του 2016 και παραμένει σε beta κατά τη σύνταξη. Σχεδιασμένο κυρίως για τις υπηρεσίες Google Cloud, η Google παραθέτει ορισμένες συγκεκριμένες περιπτώσεις χρήσης για τις λειτουργίες Google Cloud, όπως το backend για κινητά, τα API και την ανάπτυξη μικροεπιχειρήσεων, την επεξεργασία δεδομένων / ETL, τα webhooks (για απαντήσεις σε τρίτους) και το IoT. Μόνο ένας αναφερόμενος χρήστης υψηλού προφίλ μπορούσε να αναγνωριστεί - Meetup (ολοκλήρωση υπηρεσίας).

2.1.4 IBM Bluemix OpenWhisk

Το IBM Bluemix OpenWhisk είναι η πλατφόρμα υπολογιστικής cloud computing της IBM που προέρχεται από το Open Source Open Source Project του Apache, στο οποίο η IBM ήταν ο κύριος συντελεστής. Δόθηκε για γενική χρήση τον Δεκέμβριο του 2016. Οι συνήθεις περιπτώσεις χρήσης στην τεκμηρίωση του OpenWhisk περιλαμβάνουν μικροεπεξεργασίες, web, κινητά και API backends, IoT και επεξεργασία δεδομένων. Προτείνει επίσης ότι μπορεί να χρησιμοποιηθεί σε συνδυασμό με γνωστικές τεχνολογίες (π.χ. Alchemy και Watson) και συστήματα ανταλλαγής μηνυμάτων (π.χ. Kafka και IBM Messaging Hub). Παρόλο που η IBM παρέχει συνδέσμους σε πολλές αποδείξεις ιδεών, δεν μπορούσαν να εντοπιστούν χρήστες υψηλού προφίλ. Για παράδειγμα, το Skylink είναι μια απόδειξη-αντίκρουσης που χρησιμοποιεί το IBM Bluemix OpenWhisk με άλλες υπηρεσίες για την ανάλυση και την ετικέτα σε εικόνες σε πραγματικό χρόνο που συλλαμβάνονται από ένα drone. Η IBM υπογραμμίζει την ολοκλήρωση του δοχείου Docker ως σημείο διαφοροποίησης από τις λειτουργίες AWS Lambda και Google Cloud.

2.1.5 Iron.io Ironworker

Το Iron.io περιγράφει το Ironworker ως ένα επιχειρησιακό σύστημα επεξεργασίας εργασιών για την κατασκευή ασύγχρονου λογισμικού βασισμένου σε εργασία. Στον πυρήνα της, είναι μια πλατφόρμα εφαρμογών χωρίς διακομιστές για την πρωταρχική κλιμάκωση των φόρτων εργασίας που βασίζονται σε Docker σε οποιοδήποτε σύννεφο. Το Ironworker είναι διαθέσιμο σε διάφορες μορφές από το 2014. Το Iron.io είναι αξιοσημείωτο καθώς είναι μια πλατφόρμα ανεξάρτητη από το cloud και έτσι υπάρχει πάνω από άλλα σύννεφα. Το Iron.io αναφέρει τρεις κύριες περιπτώσεις χρήσης - επεξεργασία δεδομένων, επεξεργασία αρχείων και ETL. Σε ένα πιο κορεσμένο επίπεδο, προτείνει διακομιστές χωρίς διακομιστές ως λύση για τον προγραμματισμό των θέσεων εργασίας (ως αντικατάσταση του Cron), τις προτεραιότητες εργασίας, τις ιστοσελίδες, τις ουρές αναμονής, τις ουρές ώθησης και τη μακρά ψηφοφορία και τον χειρισμό των αποτυχιών. Οι χρήστες με υψηλό προφίλ περιλαμβάνουν την έκθεση λευκαντικών (κοντά σε ειδοποιήσεις ειδήσεων σε πραγματικό χρόνο), Untappd (παρτίδα επεξεργασίας) και HotelTonight (ETL). Το Iron.io επίσης απαριθμεί το Twitter, το Google, το Whole Foods Market, μεταξύ άλλων ως πελάτες χωρίς να δίνει λεπτομέρειες για το πώς το λογισμικό τους χρησιμοποιείται από αυτούς τους πελάτες και σε ποιο βαθμό. Το Ironworker είναι επίσης διαθέσιμο σε προπληρωμή, το οποίο παρέχει πρόσθετες ευκαιρίες στους ερευνητές.

2.1.6 Auth0 Webtask

Το Webtask, το webtask.io και το Auth0 Extend είναι υπολογιστικές υπηρεσίες χωρίς διακομιστές που προσφέρονται από την Auth0, κορυφαίος πάροχος λύσεων ελέγχου ταυτότητας. Τον Μάρτιο του 2017, κυκλοφόρησε ένας επεξεργαστής Webtask από την Auth0 για να υποστηρίξει εφαρμογές χωρίς διακομιστές με κύριο άξονα την αγορά Node.js. Εκτός από τους φόρτους εργασίας του Cron, το Auth0 παρέχει μια σειρά προτύπων για την ενσωμάτωση του Webtask με κοινές υπηρεσίες όπως Stripe, Slack, Sendgrid, Github, Facebook κ.λπ. Οι κύριες περιπτώσεις χρήσης που αναφέρονται περιλαμβάνουν το Webtask ως ένα κώδικα sandbox και ως webhook. Το Webtask ακολουθήθηκε από την έκδοση του Auth0 Extend, το Μάιο του 2017, μια πλατφόρμα εκτάκτου ανάγκης χωρίς διακομιστές για το SaaS, με έμφαση στους γάντζους ιστού και

τον έλεγχο ταυτότητας και διαχείρισης ταυτότητας. Οι αναφερόμενοι χρήστες του Auth0 Extend περιλαμβάνουν το Stamplay, το Meteor Development Group και το Graphcool, αν και δεν προσδιορίζονται ειδικές περιπτώσεις χρήσης για αυτούς τους πελάτες.

2.1.7 Galactic Fog Gestal Laser

Το Λασερ, συντομογραφία για το Lambda Application Server, είναι η υπηρεσία διακομιστή Galactic Fog. Η υπηρεσία κυκλοφόρησε τον Απρίλιο του 2017, ως μέρος της τέταρτης έκδοσης της πλατφόρμας Gestal, η οποία περιλαμβάνει επίσης ένα επίπεδο αφαίρεσης container-as-a-service (CaaS) και ένα πλαίσιο ενοποίησης επιχειρήσεων με δυνατότητες διαχείρισης πολιτικής και ασφάλειας. Το Laser είναι ένας μηχανισμός υψηλής απόδοσης, χαμηλής καθυστέρησης, χωρίς διακομιστές ο οποίος υποστηρίζει τις περισσότερες κοινές γλώσσες προγραμματισμού και χαρακτηρίζεται από υψηλή επεκτασιμότητα. Δεν έχουν εντοπιστεί συγκεκριμένες περιπτώσεις χρήσης.

2.1.8 Apache OpenWhisk

Το Apache OpenWhisk (2020) είναι μια εξέχουσα πλατφόρμα ανοιχτού κώδικα χωρίς διακομιστές που κατασκευάστηκε χρησιμοποιώντας το πλαίσιο Akka (Lightbend, 2020). Οι χρήστες έχουν τη δυνατότητα να υποβάλλουν λειτουργίες γραμμένες σε πολλές υποστηριζόμενες γλώσσες, οι οποίες θα εκτελούνται κατ 'απαίτηση ή σύμφωνα με τους παράγοντες ενεργοποίησης. Αν η προεπιλεγμένη γλώσσα τους δεν υποστηρίζεται, μπορούν να καθορίσουν τις δικές τους προσαρμοσμένες εικόνες Docker για να εκτελέσουν τον κώδικα τους. Οι λειτουργίες δεν έχουν εκχωρημένους πόρους εκ των προτέρων, οπότε δημιουργείται ένα δοχείο με τις απαιτούμενες εξαρτήσεις κατά την παραλαβή του πρώτου αιτήματος. Τα δοχεία διακόπτονται μετά από επεξεργασία μιας αίτησης και θα διαγραφούν μετά από ένα ορισμένο χρονικό όριο. Αν το αίτημα επανάληψης γίνει εντός αυτού του χρονικού ορίου, το δοχείο παραμένει μη επαναλαμβανόμενο και επαναχρησιμοποιείται. Αν γίνονται πιο συχνά αιτήματα, θα δημιουργηθούν πολλαπλά κοντέινερ για την εκτέλεση της ίδιας λειτουργίας. Όταν το σύστημα είναι υπερφορτωμένο, τα κοντέινερ που θα σταματήσουν θα καταστραφούν για να δημιουργήσουν περιθώρια για νέα. Αυτή η προσέγγιση είναι παρόμοια με αυτή

του OpenLambda η οποία συζητείται λεπτομερώς στο Oakes και συνεργάτες (2017). Ο προγραμματισμός είναι συγκεντρωμένος και χειρίζεται από έναν ή περισσότερους κόμβους "ελεγκτή" οι οποίοι διαβιβάζουν αιτήματα σε κόμβους "invoker" όπου εκτελούνται. Οι ελεγκτές θα προσπαθήσουν πάντα να δρομολογήσουν τις αιτήσεις κάθε λειτουργιών στον ίδιο κόμβο invoker για να μεγιστοποιήσουν την πιθανότητα μιας "θερμής εκκίνησης". Εάν ο απαιτούμενος invoker είναι υπερφορτωμένος, ο ελεγκτής θα επιλέξει έναν άλλο invoker που θα ξεκινήσει να δημιουργεί περισσότερα δοχεία για να εκτελέσει τη λειτουργία.

2.2 Big Data Analytics

Η ανάλυση μεγάλων δεδομένων (big data analytics) είναι εκεί όπου οι προηγμένες αναλυτικές τεχνικές λειτουργούν σε μεγάλα σύνολα δεδομένων. Ως εκ τούτου, οι μεγάλες αναλύσεις δεδομένων αφορούν πραγματικά δύο πράγματα -τα μεγάλα δεδομένα και τα αναλυτικά στοιχεία- καθώς και ο τρόπος με τον οποίο τα δύο συνεργάστηκαν για να δημιουργήσουν μια από τις πιο βαθιές τάσεις της επιχειρηματικής ευφυΐας σήμερα. Παρόλο που σήμερα είναι πανταχού παρόν, τα "μεγάλα δεδομένα" ως έννοια είναι γεννημένα και έχουν αβέβαιη προέλευση. Το Ίδρυμα TechAmerica ορίζει μεγάλα δεδομένα ως εξής: «Τα μεγάλα δεδομένα είναι ένας όρος που περιγράφει μεγάλους όγκους υψηλής ταχύτητας, σύνθετα και μεταβλητά δεδομένα που απαιτούν προηγμένες τεχνικές και τεχνολογίες για να καταστεί δυνατή η συλλογή, αποθήκευση, διανομή, διαχείριση και ανάλυση των πληροφοριών». (TechAmerica Foundation's Federal Big Data Commission, 2012). Ο Laney (2001) πρότεινε ότι "ο όγκος, η ποικιλία και η ταχύτητα" (“Volume, Variety, Velocity” ή «τρία V») είναι οι τρεις διαστάσεις των προκλήσεων στη διαχείριση δεδομένων. Τα τρία V έχουν αναδειχθεί ως ένα κοινό πλαίσιο για την περιγραφή μεγάλων δεδομένων (Chen et al., 2012, Kwon et al., 2014).

Ο όγκος αναφέρεται στο μέγεθος των δεδομένων. Μεγάλα μεγέθη δεδομένων αναφέρονται σε πολλαπλά terabyte και petabytes. Μια έρευνα που διεξήχθη από την IBM στα μέσα του 2012 έδειξε ότι πάνω από το ήμισυ των 1144 ερωτηθέντων θεωρούσαν σύνολα δεδομένων πάνω από ένα terabyte ως μεγάλα δεδομένα (Schroeck, Shockley, Smart, Romero-Morales, & Tufano, 2012). Ένα terabyte αποθηκεύει τόσα

δεδομένα που θα ταιριάζουν σε 1500 CD ή 220 DVD, αρκετά για να αποθηκεύουν περίπου 16 εκατομμύρια φωτογραφίες στο Facebook. Οι Beaver, Kumar, Li, Sobel και Vajgel (2010) αναφέρουν ότι το Facebook επεξεργάζεται έως και ένα εκατομμύριο φωτογραφίες ανά δευτερόλεπτο. Ένα petabyte ισούται με 1024 terabytes. Σύμφωνα με προηγούμενες εκτιμήσεις, το Facebook αποθηκεύει 260 δισεκατομμύρια φωτογραφίες χρησιμοποιώντας χώρο αποθήκευσης άνω των 20 petabytes.

Οι ορισμοί των μεγάλων όγκων δεδομένων είναι σχετικά και ποικίλλουν ανάλογα με παράγοντες, όπως ο χρόνος και ο τύπος δεδομένων. Αυτό που μπορεί σήμερα να θεωρηθεί ότι είναι μεγάλα δεδομένα ενδέχεται να μην ανταποκρίνεται στο όριο στο μέλλον, επειδή οι δυνατότητες αποθήκευσης θα αυξηθούν, επιτρέποντας την καταγραφή ακόμα μεγαλύτερων συνόλων δεδομένων. Επιπλέον, ο τύπος δεδομένων, που συζητείται υπό ποικιλία, ορίζει τι σημαίνει «μεγάλο». Δύο σύνολα δεδομένων του ίδιου μεγέθους μπορεί να απαιτούν διαφορετικές τεχνολογίες διαχείρισης δεδομένων βάσει του τύπου τους, π.χ. δεδομένα από πίνακα σε σχέση με βίντεο. Έτσι, οι ορισμοί των μεγάλων δεδομένων εξαρτώνται επίσης από τη βιομηχανία. Συνεπώς, οι παρατηρήσεις αυτές καθιστούν ανεπαρκή τον καθορισμό συγκεκριμένου ορίου για τους μεγάλους όγκους δεδομένων.

Η ποικιλία αναφέρεται στη δομική ετερογένεια σε ένα σύνολο δεδομένων. Οι τεχνολογικές εξελίξεις επιτρέπουν στις επιχειρήσεις να χρησιμοποιούν διάφορους τύπους δομημένων, ημιδομημένων και αδόμητων δεδομένων. Τα δομημένα δεδομένα, που αποτελούν μόνο το 5% όλων των υφισταμένων δεδομένων (Cukier, 2010), αναφέρονται στα πίνακες που βρίσκονται σε υπολογιστικά φύλλα ή σε σχεσιακές βάσεις δεδομένων. Το κείμενο, οι εικόνες, ο ήχος και το βίντεο είναι παραδείγματα μη δομημένων δεδομένων, τα οποία μερικές φορές στερούνται της δομικής οργάνωσης που απαιτείται από τις μηχανές για ανάλυση. Η μορφή των ημι-δομημένων δεδομένων, που καλύπτει μια συνέχεια μεταξύ πλήρως δομημένων και μη δομημένων δεδομένων, δεν συμμορφώνεται με αυστηρά πρότυπα. Η εκτεταμένη γλώσσα σήμανσης (XML), μια γλώσσα κειμένου για την ανταλλαγή δεδομένων στον Ιστό, είναι ένα τυπικό παράδειγμα ημιδομημένων δεδομένων. Τα έγγραφα XML περιέχουν ετικέτες δεδομένων καθορισμένες από το χρήστη, οι οποίες τις καθιστούν αναγνώσιμες από το μηχάνημα.

Ένα υψηλό επίπεδο ποικιλίας, ένα καθοριστικό χαρακτηριστικό των μεγάλων δεδομένων, δεν είναι απαραίτητα νέο. Οι οργανισμοί έχουν αποθηκεύσει μη δομημένα δεδομένα από εσωτερικές πηγές (π.χ. δεδομένα αισθητήρων) και εξωτερικές πηγές (π.χ. κοινωνικά μέσα). Ωστόσο, η εμφάνιση νέων τεχνολογιών διαχείρισης δεδομένων και αναλυτικών στοιχείων, που επιτρέπουν στους οργανισμούς να αξιοποιούν δεδομένα στις επιχειρηματικές τους διαδικασίες, είναι η καινοτόμος πτυχή. Για παράδειγμα, οι τεχνολογίες αναγνώρισης προσώπου εξουσιοδοτούν τους λιανοπωλητές να αποκτήσουν νοημοσύνη σχετικά με την κυκλοφορία των καταστημάτων, την ηλικία ή τη σύνθεση των φύλων των πελατών τους, καθώς και τα μοντέλα κίνησης των καταστημάτων τους. Αυτές οι ανεκτίμητες πληροφορίες αξιοποιούν τις αποφάσεις που σχετίζονται με τις προωθήσεις προϊόντων, την τοποθέτηση και τη στελέχωση. Τα δεδομένα Clickstream παρέχουν πληθώρα πληροφοριών σχετικά με τη συμπεριφορά των πελατών και τα πρότυπα περιήγησης σε διαδικτυακούς εμπόρους λιανικής πώλησης. Το Clickstream παρέχει συμβουλές σχετικά με το χρόνο και τη σειρά των σελίδων που προβάλλει ο πελάτης. Με τη χρήση μεγάλων αναλυτικών στοιχείων, ακόμη και οι μικρομεσαίες επιχειρήσεις (ΜΜΕ) μπορούν να προωθήσουν τεράστιους όγκους ημιδομημένων δεδομένων για να βελτιώσουν τα σχέδια ιστοσελίδων και να εφαρμόσουν αποτελεσματικά συστήματα cross-selling και εξατομικευμένων συστάσεων για τα προϊόντα.

Η ταχύτητα αναφέρεται στον ρυθμό με τον οποίο παράγονται τα δεδομένα και στην ταχύτητα με την οποία θα πρέπει να αναλυθεί και να ενεργήσει. Ο πολλαπλασιασμός των ψηφιακών συσκευών, όπως τα smartphones και οι αισθητήρες, έχει οδηγήσει σε έναν πρωτοφανή ρυθμό δημιουργίας δεδομένων και οδηγεί σε αυξανόμενη ανάγκη για αναλύσεις σε πραγματικό χρόνο και σχεδιασμό βάσει στοιχείων. Ακόμη και οι συμβατικοί λιανοπωλητές δημιουργούν δεδομένα υψηλής συχνότητας. Η Wal-Mart, για παράδειγμα, επεξεργάζεται περισσότερα από ένα εκατομμύριο συναλλαγές ανά ώρα (Cukier, 2010). Τα δεδομένα που προέρχονται από κινητές συσκευές και διακινούνται μέσω κινητών εφαρμογών παράγουν χείμαρρους πληροφοριών που μπορούν να χρησιμοποιηθούν για τη δημιουργία εξατομικευμένων προσφορών σε πραγματικό χρόνο για τους καθημερινούς πελάτες. Αυτά τα δεδομένα παρέχουν σωστές πληροφορίες σχετικά με τους πελάτες, όπως η γεωχωρική τοποθεσία, τα

δημογραφικά στοιχεία και τα πρότυπα αγορών του παρελθόντος, τα οποία μπορούν να αναλυθούν σε πραγματικό χρόνο για να δημιουργήσουν πραγματική αξία για τους πελάτες.

Τα μεγάλα δεδομένα είναι άχρηστα στο κενό. Η δυνητική αξία του ξεκλειδώνεται μόνο όταν χρησιμοποιείται για τη λήψη αποφάσεων. Για να καταστεί δυνατή η λήψη τέτοιων αποφάσεων βάσει τεκμηριωμένων στοιχείων, οι οργανώσεις χρειάζονται αποτελεσματικές διαδικασίες για να μετατρέψουν μεγάλους όγκους ταχέως κινούμενων και ποικίλων δεδομένων σε ουσιαστικές γνώσεις. Η συνολική διαδικασία εξαγωγής πληροφοριών από μεγάλα δεδομένα μπορεί να αναλυθεί σε πέντε στάδια (Labrinidis & Jagadish, 2012):

1) Διαχείριση δεδομένων
    a. Απόκτηση και εγγραφή
    b. Εξόρυξη, καθαρισμός και σχολιασμός
    c. Ενσωμάτωση, Συγκέντρωση και Αντιπροσώπευση
2) Analytics
    a. Μοντελοποίηση και Ανάλυση
    b. Ερμηνεία

Αυτά τα πέντε στάδια αποτελούν τις δύο βασικές υπο-διαδικασίες: τη διαχείριση δεδομένων και την ανάλυση. Η διαχείριση δεδομένων περιλαμβάνει διαδικασίες και υποστηρικτικές τεχνολογίες για την απόκτηση και αποθήκευση δεδομένων και την προετοιμασία και ανάκτηση για ανάλυση. Το Analytics, από την άλλη πλευρά, αναφέρεται σε τεχνικές που χρησιμοποιούνται για την ανάλυση και την απόκτηση πληροφοριών από μεγάλα δεδομένα. Έτσι, οι μεγάλες αναλύσεις δεδομένων μπορούν να θεωρηθούν ως μια υποδιαδικασία στη συνολική διαδικασία της «εξαγωγής γνώσεων» από μεγάλα δεδομένα.

2.3 Python

Η γλώσσα Python είναι ένα από τα πιο γνωστά δεδομένα που αναλύουν τη γλώσσα την οποία χρησιμοποιούν οι επιστήμονες δεδομένων για να επικεντρωθούν στις έρευνές τους. Ο διαδραστικός χαρακτήρας υψηλού επιπέδου αυτής της γλώσσας και των επιστημονικών βιβλιοθηκών του οικοσυστήματος την καθιστούν την προτιμώμενη

επιλογή για την ανάπτυξη αναλυτικών αλγορίθμων και τη διερεύνηση των κρυφών δεδομένων στα δεδομένα (Chen & Zhang, 2014). Εστιάζοντας στην επιστημονική κοινότητα υπολογιστών, είναι εύκολο να παρατηρήσουμε πώς αυξάνεται η χρήση της γλώσσας Python σε αυτήν την κοινότητα (ξεκινώντας από τις αρχές του 2000), τόσο σε βιομηχανικές λύσεις (εφαρμογές) όσο και σε ακαδημαϊκές έρευνες (Pedregosa et al, 2011). Η Python έχει το επιστημονικό της οικοσύστημα καθώς και πολλές χρήσιμες βιβλιοθήκες, οι οποίες είναι:

2.3.1 Numpy

Είναι συντομογραφία για το "Numerical Python", είναι η βασική δομή δεδομένων και το βασικό πακέτο σε γλώσσα Python. Γνωρίζοντας ότι όλα τα δεδομένα εισόδου στην Python αντιπροσωπεύονται ως numpy array. καθιστούν εύκολο το συμπέρασμα ότι όλες οι βιβλιοθήκες αυτής της γλώσσας είναι χτισμένες πάνω σε αυτό το πακέτο (Chen & Zhang, 2014). Το Numpy παρέχει αυτά τα χαρακτηριστικά (Pedregosa et al, 2011): 1) Ndarray: Ένα αποτελεσματικό και γρήγορο αντικείμενο πολυδιάστατου πίνακα. 2) Μακρύς κατάλογος λειτουργιών για χειρισμό με συστοιχίες μέσω υπολογισμών στοιχείων με αυτούς ή με παροχή μαθηματικών φορέων μεταξύ συστοιχιών. 3) Εργαλεία για την ανάγνωση και τη σύνταξη συστοιχιών δεδομένων που βασίζονται σε συστοιχίες στο δίσκο. 4) Μετασχηματισμός Fourier, λειτουργίες γραμμικής άλγεβρας και δημιουργία τυχαίων αριθμών. 5) Εργαλεία για την ενσωμάτωση άλλων γλωσσικών κωδικών (C, C ++ και FORTRAN) στην Python.

2.3.2 Pandas

Αυτή η δέσμη υποστηρίζει τους επιστήμονες και συνδέει τα καθήκοντά τους με τα δομημένα δεδομένα με τις προετοιμασμένες και προκαθορισμένες λειτουργίες τους και τις πλούσιες δομές δεδομένων και καθιστά αυτά τα καθήκοντα εύκολη και σημαντική (Pedregosa et al, 2011).

2.3.3 Matplotlib

Για δεδομένα απεικόνισης και σχεδίαση εκφραστικών οικότοπων, αυτό το εργαλείο είναι πολύ δημοφιλές και αποτελεσματικό για αυτά τα καθήκοντα ειδικά για 2D οικόπεδα (Pedregosa et al, 2011).

2.3.4 IPython

Πρόκειται για ένα περιβάλλον αλληλεπίδρασης υπολογιστών και ανάπτυξης, που χρησιμοποιείται για τη μεγιστοποίηση της παραγωγικότητάς σας σε αμφίδρομη υπολογιστική και ανάπτυξη λογισμικού. Περιλαμβάνει επίσης μια πλούσια κονσόλα GUI με ενσωματωμένες γραφικές παραστάσεις, μια διαδραστική διαδραστική μορφή φορητού υπολογιστή και έναν ελαφρύ, γρήγορο παράλληλο υπολογιστικό κινητήρα. Είναι μια ενοποίηση στο κέλυφος Python για να επισπεύσει τη γραφή, τον έλεγχο και την αποσφαλμάτωση του κώδικα Python (Pedregosa et al, 2011).

2.3.5 SciPy

Πρόκειται για μια συλλογή πακέτων αποτελεσματικών αλγορίθμων για γραμμική άλγεβρα, αραιή αναπαράσταση πίνακα, ειδικές λειτουργίες και βασικές στατιστικές λειτουργίες (Chen & Zhang, 2014).

2.3.6 Cython

Οι επιστήμονες δεδομένων μπορούν να χρησιμοποιήσουν τη σύνταξη Python και τις λειτουργίες υψηλού επιπέδου και να αυξήσουν την απόδοση σύνταξης για να επιτύχουν την απόδοση των μεταγλωττισμένων γλωσσών, χρησιμοποιώντας αυτό το πακέτο επειδή συνδυάζουν το C σε Python (Chen & Zhang, 2014).

2.3.7 R Language

Το R είναι μια εξαιρετικά ευέλικτη γλώσσα προγραμματισμού ανοιχτού κώδικα για τις στατιστικές και την επιστήμη των δεδομένων (McKinney, 2013). Στο σύστημα R μπορεί κανείς να κάνει κάθε είδους στατιστικό υπολογισμό χρησιμοποιώντας λειτουργικό βασισμένο σε σύνταξη κώδικα ή κώδικα βασισμένο σε πρόγραμμα με πολύ ισχυρές δυνατότητες εντοπισμού σφαλμάτων και αυτή η γλώσσα έχει πολλές

διασυνδέσεις με άλλες γλώσσες προγραμματισμού. Στη συνέχεια, τα στατιστικά που προκύπτουν μπορούν να εμφανιστούν χρησιμοποιώντας το γραφικό εργαλείο υψηλού επιπέδου στο R (Rotolo & Leydesdorff, 2015).

2.3.8 Jupyter

Το πρόγραμμα Jupyter υπάρχει για την ανάπτυξη λογισμικού ανοιχτού κώδικα, ανοικτών προτύπων και υπηρεσιών για διαδραστική χρήση υπολογιστών σε δεκάδες γλώσσες προγραμματισμού. Το JupyterLab είναι ένα διαδραστικό περιβάλλον ανάπτυξης για το Jupyter για φορητούς υπολογιστές, κώδικες και δεδομένα. Το JupyterLab είναι ευέλικτο: διαμορφώνει και ρυθμίζει τη διεπαφή χρήστη για να υποστηρίξει ένα ευρύ φάσμα ροών εργασίας στην επιστήμη των δεδομένων, την επιστημονική πληροφορική και τη μηχανική μάθηση. Το JupyterLab είναι επεκτάσιμο και αρθρωτό: γράφει plugins που προσθέτουν νέα εξαρτήματα και ενσωματώνονται με υπάρχοντα.

Το JupyterLab δίνει τη δυνατότητα επεξεργασίας με έγγραφα και δραστηριότητες όπως σημειωματάρια Jupyter, επεξεργαστές κειμένων, τερματικά και προσαρμοσμένα στοιχεία, με ευέλικτο, ολοκληρωμένο και επεκτάσιμο τρόπο. Ο χρήστης μπορεί να οργανώσει πολλά έγγραφα και δραστηριότητες δίπλα-δίπλα στην περιοχή εργασίας χρησιμοποιώντας καρτέλες και διαχωριστές. Τα έγγραφα και οι δραστηριότητες ενσωματώνονται μεταξύ τους, επιτρέποντας τη δημιουργία νέων ροών εργασίας για διαδραστικούς υπολογιστές, για παράδειγμα: 1) Κονσόλες κώδικα παρέχουν παροδικά scratchpads για την εκτέλεση κώδικα διαδραστικά, με πλήρη υποστήριξη για την πλούσια απόδοση. Μια κονσόλα κώδικα μπορεί να συνδεθεί με έναν πυρήνα φορητού υπολογιστή ως αρχείο καταγραφής υπολογισμών από τον φορητό υπολογιστή, για παράδειγμα. 2) Τα έγγραφα που υποστηρίζονται από πυρήνα επιτρέπουν την εκτέλεση κώδικα σε οποιοδήποτε αρχείο κειμένου (Markdown, Python, R, LaTeX κ.λπ.) σε οποιοδήποτε πυρήνα Jupyter. 3) Οι εξόδους των φορητών υπολογιστών μπορούν να αντικατοπτρίζονται στη δική τους καρτέλα, δίπλα-δίπλα στο φορητό υπολογιστή, επιτρέποντας απλούς πίνακες ελέγχου με διαδραστικούς ελέγχους που υποστηρίζονται από έναν πυρήνα και 4) Πολλαπλές προβολές εγγράφων με διαφορετικούς συντάκτες ή θεατές επιτρέπουν ζωντανή επεξεργασία εγγράφων που αντανακλώνται σε άλλους

θεατές. Για παράδειγμα, είναι εύκολο να έχετε ζωντανή προεπισκόπηση των Markdown, Delimiter-separated Values ή Vega / Vega-Lite.

Το JupyterLab προσφέρει επίσης ένα ενοποιημένο μοντέλο για την προβολή και το χειρισμό μορφών δεδομένων. Το JupyterLab κατανοεί πολλές μορφές αρχείων (εικόνες, CSV, JSON, Markdown, PDF, Vega, Vega-Lite κλπ.) Και μπορεί επίσης να εμφανίσει έξοδο πλούσιου πυρήνα σε αυτές τις μορφές. Για περισσότερες πληροφορίες, ανατρέξτε στην ενότητα Μορφές αρχείου και εξόδου. Για να πλοηγηθεί ο χρήστης στο περιβάλλον εργασίας χρήστη, το JupyterLab προσφέρει προσαρμόσιμες συντομεύσεις πληκτρολογίου και τη δυνατότητα χρήσης χαρτών κλειδιών από vim, emacs και Sublime Text στον επεξεργαστή κειμένου. Οι επεκτάσεις JupyterLab μπορούν να προσαρμόσουν ή να βελτιώσουν οποιοδήποτε τμήμα του JupyterLab, συμπεριλαμβανομένων νέων θεμάτων, επεξεργαστών αρχείων και προσαρμοσμένων στοιχείων. Το JupyterLab εξυπηρετείται από τον ίδιο διακομιστή και χρησιμοποιεί την ίδια μορφή φορητού εγγράφου με το κλασικό Notebook Jupyter.

Τα Notebooks έχουν σχεδιαστεί για να υποστηρίζουν τη ροή εργασιών της επιστημονικής πληροφορικής, από τη διαδραστική εξερεύνηση έως τη δημοσίευση λεπτομερούς αρχείου υπολογισμών. Ο κώδικας σε ένα σημειωματάριο είναι οργανωμένος σε κελιά, κομμάτια τα οποία μπορούν να τροποποιηθούν και να εκτελεστούν μεμονωμένα. Η έξοδος από κάθε κελί εμφανίζεται ακριβώς κάτω από αυτό και αποθηκεύεται ως μέρος του εγγράφου. Πρόκειται για μια εξέλιξη του αλληλεπιδραστικού κελύφους ή REPL (read-evaluate-print loop), που εδώ και πολύ καιρό αποτελεί τη βάση του διαδραστικού προγραμματισμού (Iverson, 1962; Spence, 1975). Ωστόσο, ενώ η άμεση έξοδος στα περισσότερα κελύφη μπορεί να είναι μόνο κείμενο, τα σημειωματάρια μπορούν να περιλαμβάνουν πλούσιο αποτέλεσμα όπως οικόπεδα, μορφοποιημένες μαθηματικές εξισώσεις, ακόμα και διαδραστικούς ελέγχους και γραφικά. Το κείμενο Prose μπορεί να παρεμβληθεί με τον κώδικα και την έξοδο σε ένα σημειωματάριο για να εξηγήσει και να τονίσει συγκεκριμένα μέρη, σχηματίζοντας μια πλούσια υπολογιστική αφήγηση.

Η διασύνδεση του φορητού υπολογιστή έγινε πιο δημοφιλής μεταξύ των μαθηματικών. Τα ιδιόκτητα συστήματα αλγεβρικής πληροφορικής Mathematica και Maple διαθέτουν αμφίδρομες διεπαφές, όπως και το - ανοικτού κώδικα- SageMath. Ο Jupyter στοχεύει

να φέρει φορητούς υπολογιστές σε ένα ευρύτερο κοινό. Το Jupyter είναι ένα έργο ανοιχτού κώδικα, το οποίο μπορεί να λειτουργήσει με κώδικα σε πολλές διαφορετικές γλώσσες προγραμματισμού. Οι διαφορετικές γλώσσες, που ονομάζονται πυρήνες, επικοινωνούν με το Jupyter χρησιμοποιώντας ένα κοινό, τεκμηριωμένο πρωτόκολλο. πάνω από 50 τέτοια backends έχουν ήδη γραφτεί, για γλώσσες που κυμαίνονται από C ++ έως Bash. Το Jupyter εξελίχθηκε από το έργο IPython (Pérez & Granger, 2007), που αρχικά παρείχε αυτή τη διασύνδεση μόνο για τη γλώσσα Python. Το IPython συνεχίζει να παρέχει τον κανονικό πυρήνα Python για το Jupyter.

Το Jupyter Notebook είναι προσβάσιμο μέσω ενός σύγχρονου web browser. Αυτό καθιστά πρακτική την χρήση της ίδιας διεπαφής που εκτελείται τοπικά όπως μια εφαρμογή επιφάνειας εργασίας ή εκτελείται σε έναν απομακρυσμένο διακομιστή. Στην τελευταία περίπτωση, το μόνο λογισμικό που χρειάζεται ο τοπικός χρήστης είναι ένα πρόγραμμα περιήγησης ιστού. Έτσι, για παράδειγμα, ένας δάσκαλος μπορεί να ρυθμίσει το λογισμικό σε ένα διακομιστή και να δώσει εύκολα πρόσβαση στους μαθητές. Τα αρχεία σημειωματάριων που δημιουργεί είναι μια απλή, τεκμηριωμένη μορφή JSON, με την επέκταση '.ipynb'. Είναι απλό να γράφετε άλλα εργαλεία λογισμικού τα οποία έχουν πρόσβαση και χειρίζονται αυτά τα αρχεία.

Τα Notebook καταγράφουν έναν υπολογισμό για να τα εξηγήσουν λεπτομερώς σε άλλους και μια ποικιλία εργαλείων βοηθούν τους χρήστες να μοιράζονται εύκολα σημειωματάρια. Το πρόγραμμα Jupyter περιλαμβάνει το nbconvert, το οποίο μετατρέπει τα αρχεία των φορητών υπολογιστών σε διάφορες μορφές αρχείων, συμπεριλαμβανομένων των HTML, LaTeX και PDF, έτσι ώστε να είναι προσβάσιμα χωρίς να χρειάζεται να έχετε εγκατεστημένο λογισμικό Jupyter. Το Nbconvert χρησιμοποιεί έναν ισχυρό μηχανισμό (Jinja), οπότε η διαδικασία μετατροπής μπορεί να προσαρμοστεί πλήρως για να παράγει διαφορετικά είδη παραγωγής. Ένα άλλο έργο Jupyter, nbviewer, είναι μια φιλοξενούμενη υπηρεσία web που βασίζεται στο nbconvert. Το nbviewer παρέχει μια προβολή HTML των αρχείων σημειωματάριων που δημοσιεύονται οπουδήποτε στον ιστό. Το πρωτεύον παράδειγμα εμφανίζεται στη διεύθυνση https://nbviewer.jupyter.org/, αλλά επειδή είναι ανοικτού κώδικα, ο καθένας μπορεί να τρέχει το δικό του παράδειγμα - για παράδειγμα σε εσωτερικό δίκτυο, για να βλέπει φορητούς υπολογιστές που δεν πρέπει να δημοσιοποιούνται. Αυτές οι προβολές

HTML έχουν ένα σημαντικό πλεονέκτημα σε σχέση με τη δημοσίευση μετατρεπόμενων HTML απευθείας: συνδέονται πίσω στο αρχείο του φορητού υπολογιστή, έτσι οι ενδιαφερόμενοι αναγνώστες μπορούν να το κατεβάσουν, να το εκτελέσουν και να το τροποποιήσουν οι ίδιοι.

Ενώ το nbconvert και το nbviewer διευκολύνουν την κοινή χρήση στατικών επεξεργασμένων φορητών υπολογιστών, ένα νέο έργο που ονομάζεται Binder (http://mybinder.org/) επιτρέπει την ανταλλαγή ζωντανών φορητών υπολογιστών, συμπεριλαμβανομένου ενός υπολογιστικού περιβάλλοντος στο οποίο οι χρήστες μπορούν να εκτελέσουν τον κώδικα. Οι συντάκτες μπορούν να δημοσιεύουν σημειωματάρια στο GitHub μαζί με μια προδιαγραφή περιβάλλοντος σε μία από τις μερικές κοινές μορφές. Με τον εντοπισμό της υπηρεσίας ιστού Binder στον αποθετήριο, δημιουργείται αυτόματα ένα προσωρινό περιβάλλον με τους φορητούς υπολογιστές και τις βιβλιοθήκες και τα δεδομένα που απαιτούνται για την εκτέλεση τους. Αυτό επιτρέπει στους δημιουργούς να δημοσιεύουν τον κώδικα τους σε μια αλληλεπιδραστική και άμεσα επαληθεύσιμη μορφή. Μαζί, αυτά τα εργαλεία επιτρέπουν τη διατήρηση και επαναχρησιμοποίηση του επιστημονικού κώδικα, του υπολογιστικού περιβάλλοντος για την εκτέλεση αυτού του κώδικα και των δεδομένων εντός των περιορισμών μεγέθους ενός αποθετηρίου git. Τα εργαλεία τρίτων, όπως το noWorkflow, μπορούν να ενσωματωθούν με αυτό για να παρακολουθήσουν την προέλευση: πώς οι εισροές, ο κώδικας και τα δημιουργούμενα αρχεία σχετίζονται μεταξύ τους. Το noWorkflow καταγράφει την εκτέλεση ενός επισημανθέντος φορητού υπολογιστή ή μιας δέσμης ενεργειών που εκτελείται μέσω του εργαλείου γραμμής εντολών ως «δοκιμή», καταγράφοντας σε μια βάση δεδομένων τον κώδικα που χρησιμοποιήθηκε, το περιβάλλον στο οποίο έτρεχε, τις εκδόσεις των ενοτήτων που χρησιμοποιήθηκαν , και τα αρχεία ανάγνωσης και γραφής.

2.3.9 Scikit-learn

Η γλώσσα προγραμματισμού Python καθιερώνεται ως μια από τις πιο δημοφιλείς γλώσσες για την επιστημονική πληροφορική. Χάρη στη διαδραστική φύση υψηλού επιπέδου και το ωρίμανο οικοσύστημα επιστημονικών βιβλιοθηκών, αποτελεί μια ελκυστική επιλογή για την αλγοριθμική ανάπτυξη και την διερευνητική ανάλυση των

δεδομένων. Ωστόσο, ως γλώσσα γενικού σκοπού, χρησιμοποιείται όλο και περισσότερο όχι μόνο σε ακαδημαϊκά περιβάλλοντα αλλά και στη βιομηχανία. Το Scikit-learn εκμεταλλεύεται αυτό το πλούσιο περιβάλλον για να παρέχει εφαρμογές τελευταίας τεχνολογίας σε πολλούς γνωστούς αλγόριθμους μηχανικής μάθησης, διατηρώντας παράλληλα μια εύχρηστη διασύνδεση σφικτά ενσωματωμένη στη γλώσσα Python. Αυτό ανταποκρίνεται στην αυξανόμενη ανάγκη ανάλυσης στατιστικών δεδομένων από μη ειδικούς σε βιομηχανίες λογισμικού και διαδικτύου, καθώς και σε τομείς εκτός της επιστήμης των υπολογιστών, όπως η βιολογία ή η φυσική. Το Scikit-learn διαφέρει από τις άλλες εργαλειοθήκες μηχανικής μάθησης της Python για διάφορους λόγους: i) διανέμεται υπό την άδεια BSD ii) ενσωματώνει κώδικα για αποδοτικότητα, αντίθετα με MDP και pybrain, iii) εξαρτάται μόνο από πολλούς και scipy για διευκόλυνση εύκολης διανομή, σε αντίθεση με το pymvpa που έχει προαιρετικές εξαρτήσεις όπως το R και το shogun, και iv) επικεντρώνεται στον επιτακτικό προγραμματισμό, σε αντίθεση με το pybrain που χρησιμοποιεί ένα πλαίσιο ροής δεδομένων. Ενώ το πακέτο είναι κυρίως γραμμένο σε Python, ενσωματώνει τις βιβλιοθήκες C ++ LibSVM και LibLinear που παρέχουν υλοποιήσεις αναφοράς SVM και γενικευμένων γραμμικών μοντέλων με συμβατές άδειες χρήσης. Τα δυαδικά πακέτα διατίθενται σε ένα πλούσιο σύνολο πλατφορμών, συμπεριλαμβανομένων των Windows και των πλατφορμών POSIX.

Επιπλέον, έχει διανεμηθεί ευρέως ως μέρος μεγάλων διανομών ελεύθερου λογισμικού, όπως το Ubuntu, το Debian, το Mandriva, το NetBSD και το Macports, καθώς και σε εμπορικές διανομές όπως η "Enthought Python Distribution".

2.3.10 Regular Expression (RE)

Μια κανονική έκφραση (ή RE) καθορίζει ένα σύνολο συμβολοσειρών που ταιριάζει με αυτό. οι λειτουργίες σε αυτήν την ενότητα επιτρέπουν στον χρήστη να ελέγξει εάν μια συγκεκριμένη συμβολοσειρά αντιστοιχεί σε μια δεδομένη κανονική έκφραση (ή αν μια δεδομένη κανονική έκφραση ταιριάζει με μια συγκεκριμένη συμβολοσειρά, η οποία έρχεται στο ίδιο πράγμα).

Οι κανονικές εκφράσεις μπορούν να συνενωθούν για να σχηματίσουν νέες κανονικές εκφράσεις. εάν οι A και B είναι και οι δύο κανονικές εκφράσεις, τότε το AB είναι

επίσης μια κανονική έκφραση. Γενικά, αν μια συμβολοσειρά p ταιριάζει με το A και μια άλλη συμβολοσειρά q ταιριάζει με το B, η συμβολοσειρά pq θα ταιριάζει με το AB Αυτό ισχύει εκτός εάν οι A ή B περιέχουν λειτουργίες χαμηλής προτεραιότητας. οριακές συνθήκες μεταξύ A και B · ή έχουν αριθμημένες αναφορές ομάδων. Έτσι, πολύπλοκες εκφράσεις μπορούν εύκολα να κατασκευαστούν από απλούστερες πρωτόγονες εκφράσεις όπως αυτές που περιγράφονται εδώ. Για λεπτομέρειες σχετικά με τη θεωρία και την εφαρμογή των κανονικών εκφράσεων, συμβουλευτείτε το βιβλίο Friedl (Frie09), ή σχεδόν οποιοδήποτε βιβλίο σχετικά με την κατασκευή μεταγλωττιστή.

Οι κανονικές εκφράσεις μπορούν να περιέχουν τόσο ειδικούς όσο και συνήθεις χαρακτήρες. Οι περισσότεροι συνηθισμένοι χαρακτήρες, όπως «A», «α» ή «0», είναι οι απλούστερες κανονικές εκφράσεις. απλώς ταιριάζουν. Ο χρήστης μπορεί να συνδυάσει συνηθισμένους χαρακτήρες, οπότε η τελευταία αντιστοιχεί στη συμβολοσειρά 'last'. Μερικοί χαρακτήρες, όπως '|' ή '(', είναι ειδικοί.) Ειδικοί χαρακτήρες είτε ισχύουν για τάξεις συνήθων χαρακτήρων είτε επηρεάζουν τον τρόπο με τον οποίο ερμηνεύονται οι κανονικές εκφράσεις γύρω τους.

2.3.11 Seaborn

Το Seaborn είναι μια βιβλιοθήκη οπτικοποίησης δεδομένων Python που βασίζεται στο matplotlib. Παρέχει μια διεπαφή υψηλού επιπέδου για την κατάρτιση ελκυστικών και ενημερωτικών στατιστικών γραφικών. Εδώ ακολουθείτε μερικές από τις λειτουργίες που προσφέρει το θαλάσσιο σκάφος:

Ένα API προσανατολισμένο στο σύνολο δεδομένων για την εξέταση σχέσεων μεταξύ πολλαπλών μεταβλητών: 1) Εξειδικευμένη υποστήριξη για τη χρήση κατηγορηματικών μεταβλητών για την εμφάνιση παρατηρήσεων ή συγκεντρωτικών στατιστικών. 2) Επιλογές για την απεικόνιση των μονοκαναλικών ή διμερών διανομών και για τη σύγκρισή τους μεταξύ υποσυνόλων δεδομένων. 3) Αυτόματη εκτίμηση και σχεδίαση μοντέλων γραμμικής παλινδρόμησης για διαφορετικές μεταβλητές που εξαρτώνται από το είδος. 4) Βολικές προβολές στη συνολική δομή σύνθετων συνόλων δεδομένων. 5) Αφαίρεση υψηλού επιπέδου για τη διαμόρφωση πλέγματος πολλαπλών γραμμών που επιτρέπουν στον χρήστη να κατασκευάζει εύκολα πολύπλοκες απεικονίσεις. 6)

Συνοπτικός έλεγχος του styling με σχήμα matplotlib με πολλά ενσωματωμένα θέματα.
7) Εργαλεία για την επιλογή των παλετών χρωμάτων που αποκαλύπτουν πιστά τα πρότυπα στα δεδομένα του χρήστη.

Το Seaborn στοχεύει να κάνει την απεικόνιση κεντρικό μέρος της διερεύνησης και κατανόησης των δεδομένων. Οι λειτουργίες σχεδίασης με βάση το σύνολο δεδομένων λειτουργούν σε πλαίσια δεδομένων και συστοιχίες που περιέχουν σύνολα δεδομένων και εσωτερικά εκτελούν την απαραίτητη σημασιολογική χαρτογράφηση και στατιστική συσσωμάτωση για να παράγουν ενημερωτικά οικόπεδα.

Πίσω από τις σκηνές, ο θαλασσοπόρος χρησιμοποιεί matplotlib για να σχεδιάσει οικόπεδα. Πολλές εργασίες μπορούν να εκτελεστούν μόνο με λειτουργίες θαλάσσιου θαλάμου, αλλά η περαιτέρω προσαρμογή μπορεί να απαιτεί τη χρήση του matplotlib άμεσα.

2.3.12 NTLK (Word_Tokenize)

Το NLTK είναι μια κορυφαία πλατφόρμα για την ανάπτυξη προγραμμάτων Python για την επεξεργασία δεδομένων με την ανθρώπινη γλώσσα. Παρέχει εύχρηστες διασυνδέσεις σε πάνω από 50 κορμούς και λεξικά μέσα, όπως το WordNet, μαζί με μια σουίτα βιβλιοθηκών επεξεργασίας κειμένου για ταξινόμηση, tokenization, starking, tagging, parsing και σημασιολογική συλλογιστική, περιτυλίγματα για βιβλιοθήκες NLP βιομηχανικής αντοχής, και ένα ενεργό φόρουμ συζήτησης.

Χάρη σε έναν πρακτικό οδηγό που εισάγει θεμελιώδη στοιχεία προγραμματισμού παράλληλα με τα θέματα της υπολογιστικής γλωσσολογίας, καθώς και τη συνολική τεκμηρίωση API, το NLTK είναι κατάλληλο για γλωσσολόγους, μηχανικούς, φοιτητές, εκπαιδευτικούς, ερευνητές και χρήστες του κλάδου. Το NLTK είναι διαθέσιμο για Windows, Mac OS X και Linux. Το NLTK είναι ένα ελεύθερο, ανοικτού κώδικα, έργο που βασίζεται στην κοινότητα.

Το NLTK έχει ονομαστεί "ένα θαυμάσιο εργαλείο για τη διδασκαλία και την εργασία στην υπολογιστική γλωσσολογία χρησιμοποιώντας τη Python" και "μια εκπληκτική βιβλιοθήκη για να παίξει με τη φυσική γλώσσα".

Η επεξεργασία της φυσικής γλώσσας με την Python παρέχει μια πρακτική εισαγωγή στον προγραμματισμό της επεξεργασίας γλώσσας. Γράφτηκε από τους δημιουργούς

του NLTK, καθοδηγεί τον αναγνώστη από τις βασικές αρχές της γραφής των προγραμμάτων Python, που εργάζονται με τα σωματίδια, κατηγοριοποιούν το κείμενο, αναλύουν τη γλωσσική δομή και πολλά άλλα. Το Tokenization είναι η διαδικασία του διαχωρισμού μιας συμβολοσειράς, ενός κειμένου σε μια λίστα με tokens. Μπορούμε να σκεφτούμε το συμβολικό, καθώς μέρη όπως μια λέξη είναι ένα συμβολικό στοιχείο σε μια πρόταση, και μια πρόταση είναι ένα συμβολικό στοιχείο σε μια παράγραφο. Πώς λειτουργεί το send_tokenize; Η συνάρτηση sent_tokenize χρησιμοποιεί μια παρουσία του PunktSentenceTokenizer από το nltk.

Με τη βοήθεια της μεθόδου nltk.tokenize.word_tokenize (), ο χρήστης είναι σε θέση να εξάγει τα μάρκες από σειρά χαρακτήρων χρησιμοποιώντας τη μέθοδο tokenize.word_tokenize (). Στην πραγματικότητα επιστρέφει τις συλλαβές από μια μόνο λέξη. Μια μόνο λέξη μπορεί να περιέχει μία ή δύο συλλαβές.

2.3.13 Counter

Παρέχεται ένα αντίθετο εργαλείο (counter) για την υποστήριξη βολικών και γρήγορων λογαριασμών. Ο μετρητής είναι μια υποκατηγορία dict για την καταμέτρηση των αντικειμένων που έχουν υποστεί ζημιά. Πρόκειται για μια μη προσαρμοσμένη συλλογή όπου τα στοιχεία αποθηκεύονται ως πλήκτρα λεξικών και οι μετρήσεις τους αποθηκεύονται ως τιμές λεξικού. Οι αριθμοί επιτρέπονται να είναι οποιαδήποτε ακέραιη τιμή, συμπεριλαμβανομένων μηδενικών ή αρνητικών αριθμών. Η κλάση Counter είναι παρόμοια με τις τσάντες ή τα πολλαπλάσια σε άλλες γλώσσες.

Τα στοιχεία υπολογίζονται από μια επαναληπτική ή αρχικοποιημένη από άλλη χαρτογράφηση (ή μετρητή). Τα αντικείμενα Counter έχουν μια λεξικολογική διεπαφή εκτός από το ότι επιστρέφουν μια μηδενική μέτρηση για τα στοιχεία που λείπουν αντί για την αύξηση ενός KeyError. Ο ορισμός μέτρησης στο μηδέν δεν αφαιρεί ένα στοιχείο από έναν μετρητή. Τα αντικείμενα Counter υποστηρίζουν τρεις μεθόδους πέραν αυτών που είναι διαθέσιμες για όλα τα λεξικά: 1) elements(): Επιστροφή ενός iterator πάνω από τα στοιχεία που επαναλαμβάνουν το καθένα όσες φορές είναι η μέτρησή του. Τα στοιχεία επιστρέφονται σε αυθαίρετη σειρά. Εάν η μέτρηση ενός στοιχείου είναι μικρότερη από ένα, τα στοιχεία () θα το αγνοήσουν. 2) most_common((n)): Επιστροφή μιας λίστας με τα πιο κοινά στοιχεία και τις μετρήσεις

τους από τα πιο συνηθισμένα στα ελάχιστα. Εάν το n παραλείπεται ή Κανένα, το most_common () επιστρέφει όλα τα στοιχεία στον μετρητή. Τα στοιχεία με ίσες μετρήσεις παραγγέλλονται αυθαίρετα 3) subtract((iterable-or-mapping)): Τα στοιχεία αφαιρούνται από μια επαναληπτική ή άλλη χαρτογράφηση (ή μετρητή). Όπως το dict.update () αλλά αφαιρεί τις μετρήσεις αντί να τις αντικαθιστά. Και οι δύο εισόδους και εξόδους μπορεί να είναι μηδενικές ή αρνητικές. Οι συνήθεις μέθοδοι λεξικού είναι διαθέσιμες για αντικείμενα Counter εκτός από δύο που λειτουργούν διαφορετικά για μετρητές.4) fromkeys(iterable): Αυτή η μέθοδος κλάσης δεν εφαρμόζεται για αντικείμενα Counter. 5) update((iterable-or-mapping)): Τα στοιχεία υπολογίζονται από ένα επαναληπτικό ή προστιθέμενο από μια άλλη χαρτογράφηση (ή μετρητή). Όπως το dict.update () αλλά προσθέτει μετρήσεις αντί να τις αντικαθιστά. Επίσης, το επαναληπτικό αναμένεται να είναι μια ακολουθία στοιχείων, όχι μια ακολουθία ζευγών (κλειδί, τιμή).

Διάφορες μαθηματικές λειτουργίες παρέχονται για το συνδυασμό Αντικειμένων αντικειμένων για την παραγωγή multisets (μετρητές που έχουν μετρήσεις μεγαλύτερες από το μηδέν). Η προσθήκη και η αφαίρεση συνδυάζουν μετρητές προσθέτοντας ή αφαιρώντας τις μετρήσεις των αντίστοιχων στοιχείων. Η διασταύρωση και η ένωση επιστρέφουν το ελάχιστο και το μέγιστο των αντίστοιχων μετρήσεων. Κάθε λειτουργία μπορεί να δεχθεί εισόδους με υπογεγραμμένες μετρήσεις, αλλά η έξοδος θα αποκλείσει τα αποτελέσματα με μετρήσεις μηδέν ή μικρότερες.

Οι μετρητές σχεδιάστηκαν κατά κύριο λόγο για να δουλεύουν με θετικούς ακέραιους αριθμούς για να αντιπροσωπεύουν τις μετρήσεις λειτουργίας. Εντούτοις, ελήφθη μέριμνα ώστε να μην εμποδίζονται χωρίς λόγο οι περιπτώσεις χρήσης που χρειάζονται άλλους τύπους ή αρνητικές τιμές.

2.3.14 Stop Words

Τα Stopwords είναι συνηθισμένες λέξεις που γενικά δεν συμβάλλουν στην έννοια μιας φράσης, τουλάχιστον για τους σκοπούς της ανάκτησης πληροφοριών και της επεξεργασίας φυσικής γλώσσας. Οι περισσότερες μηχανές αναζήτησης θα φιλτράρουν τα αποσπάσματα από τα ερωτήματα αναζήτησης και τα έγγραφα, προκειμένου να εξοικονομήσουν χώρο στο ευρετήριό τους.

Λέξεις όπως "οι περισσότεροι" εμφανίζονται πολύ συχνά σε όλα τα διαφορετικά πλαίσια και λέξεις όπως αυτό ονομάζονται λέξεις σταματήματος. Δεν φέρνουν τόσα πολλά στοιχεία και επομένως δεν πρέπει να ζυγίζονται τόσο όσο λέξεις όπως «εικόνες» που δεν συμβαίνουν συχνά σε διαφορετικά πλαίσια. Η καλύτερη επιλογή θα ήταν να αφαιρέσετε όλες τις λέξεις που είναι τόσο συχνές ώστε να μην βοηθούν στη διάκριση μεταξύ διαφορετικών κειμένων. Αυτές οι λέξεις ονομάζονται stop words.

2.4 Αλγόριθμοι Analytics

2.4.1 Δέντρα ταξινόμησης και παλινδρόμησης

Τα δέντρα ταξινόμησης και παλινδρόμησης χρησιμοποιούν μια απόφαση για την κατηγοριοποίηση δεδομένων. Κάθε απόφαση βασίζεται σε μια ερώτηση που σχετίζεται με μία από τις μεταβλητές εισόδου. Με κάθε ερώτηση και αντίστοιχη απάντηση, η παρουσία δεδομένων μεταφέρεται πιο κοντά στην κατηγοριοποίηση με συγκεκριμένο τρόπο. Αυτό το σύνολο ερωτήσεων και απαντήσεων και οι επακόλουθες διαιρέσεις δεδομένων δημιουργούν μια δομή που μοιάζει με δέντρο. Στο τέλος κάθε σειράς ερωτήσεων υπάρχει μια κατηγορία. Αυτό ονομάζεται κόμβος φύλλων του δέντρου ταξινόμησης (Kelleher et al, 2015).

Αυτά τα δέντρα ταξινόμησης μπορούν να γίνουν αρκετά μεγάλα και πολύπλοκα. Μία μέθοδος για τον έλεγχο της πολυπλοκότητας είναι το κλάδεμα του δέντρου ή η σκόπιμη αφαίρεση των επιπέδων αμφισβήτησης για την εξισορρόπηση μεταξύ ακριβούς εφαρμογής και αφαίρεσης. Ένα μοντέλο που λειτουργεί καλά με όλες τις περιπτώσεις τιμών εισόδου, τόσο αυτές που είναι γνωστές στην εκπαίδευση όσο και αυτές που δεν είναι, είναι πρωταρχικής σημασίας. Η πρόληψη της υπερθέρμανσης αυτού του μοντέλου απαιτεί μια ευαίσθητη ισορροπία ανάμεσα στην ακριβή εφαρμογή και την αφαίρεση (Cevher et al, 2014).

Μια παραλλαγή των δέντρων ταξινόμησης και παλινδρόμησης ονομάζεται τυχαία δάση. Αντί να κατασκευάσουμε ένα ενιαίο δέντρο με πολλούς κλάδους λογικής, ένα τυχαίο δάσος είναι το αποκορύφωμα πολλών μικρών και απλών δέντρων που κάθε ένα αξιολογεί τις περιπτώσεις δεδομένων και καθορίζει μια κατηγοριοποίηση. Αφού όλα αυτά τα απλά δέντρα ολοκληρώσουν την αξιολόγηση των δεδομένων τους, η διαδικασία συγχωνεύει τα επιμέρους αποτελέσματα για να δημιουργήσει μια τελική

πρόβλεψη της κατηγορίας με βάση το σύνθετο των μικρότερων κατηγοριοποιήσεων. Αυτό συνήθως αναφέρεται ως μέθοδος συνόλου. Αυτά τα τυχαία δάση συχνά λειτουργούν καλά στην εξισορρόπηση της ακριβούς εφαρμογής και της αφαίρεσης και έχουν εφαρμοστεί επιτυχώς σε πολλές επιχειρηματικές περιπτώσεις (Kelleher et al, 2015).

Σε αντίθεση με τη λογική παλινδρόμηση, η οποία επικεντρώνεται σε μια κατηγοριοποίηση ναι ή όχι, μπορούν να χρησιμοποιηθούν ταξινομήσεις και παλινδρόμηση για την πρόβλεψη κατηγοριοποιήσεων πολλαπλών τιμών. Είναι επίσης ευκολότερο να απεικονίσουν και να δουν την οριστική διαδρομή που οδήγησε τον αλγόριθμο σε μια συγκεκριμένη κατηγοριοποίηση (Cevher et al, 2014).

2.4.2 Ομαδοποίηση (Clustering)

Η ομαδοποίηση είναι το καθήκον της διαίρεσης του πληθυσμού ή των σημείων δεδομένων σε διάφορες ομάδες έτσι ώστε τα σημεία δεδομένων στις ίδιες ομάδες να είναι περισσότερο παρόμοια με άλλα σημεία δεδομένων στην ίδια ομάδα από εκείνα σε άλλες ομάδες. Με απλά λόγια, ο στόχος είναι να διαχωρίσουν ομάδες με παρόμοια χαρακτηριστικά και να τους αναθέσουν σε ομάδες. Σε γενικές γραμμές, η ομαδοποίηση μπορεί να χωριστεί σε δύο υποομάδες. 1) Σκληρή ομαδοποίηση: Στη σκληρή ομαδοποίηση, κάθε σημείο δεδομένων είτε ανήκει πλήρως είτε όχι σε ένα σύμπλεγμα. 2) Μαλακή ομαδοποίηση: Στην μαλακή συσσωμάτωση, αντί να τοποθετείται κάθε σημείο δεδομένων σε ξεχωριστό σύμπλεγμα, εκχωρείται μια πιθανότητα ή πιθανότητα να υπάρχει αυτό το σημείο δεδομένων σε αυτά τα συμπλέγματα (Cevher et al, 2014).

Δεδομένου ότι το καθήκον της ομαδοποίησης είναι υποκειμενικό, τα μέσα που μπορούν να χρησιμοποιηθούν για την επίτευξη αυτού του στόχου είναι άφθονα. Κάθε μεθοδολογία ακολουθεί ένα διαφορετικό σύνολο κανόνων για τον ορισμό της «ομοιότητας» μεταξύ των σημείων δεδομένων. Στην πραγματικότητα, είναι γνωστοί περισσότεροι από 100 αλγόριθμοι ομαδοποίησης (Cevher et al, 2014). Αλλά λίγοι από τους αλγόριθμους χρησιμοποιούνται ευρέως, όπως φαίνεται λεπτομερώς παρακάτω:

Μοντέλα συνδεσιμότητας: Όπως υποδηλώνει το όνομα, αυτά τα μοντέλα βασίζονται στην ιδέα ότι τα σημεία δεδομένων πλησιέστερα στο χώρο δεδομένων παρουσιάζουν μεγαλύτερη ομοιότητα μεταξύ τους από τα σημεία δεδομένων που βρίσκονται πιο μακριά. Αυτά τα μοντέλα μπορούν να ακολουθήσουν δύο προσεγγίσεις. Στην πρώτη

προσέγγιση, αρχίζουν με την ταξινόμηση όλων των σημείων δεδομένων σε ξεχωριστά συμπλέγματα και στη συνέχεια τη συγκέντρωσή τους καθώς η απόσταση μειώνεται. Στη δεύτερη προσέγγιση, όλα τα σημεία δεδομένων ταξινομούνται ως ένα ενιαίο σύμπλεγμα και στη συνέχεια διαχωρίζονται καθώς αυξάνεται η απόσταση. Επίσης, η επιλογή της λειτουργίας απόστασης είναι υποκειμενική. Αυτά τα μοντέλα είναι πολύ εύκολο να ερμηνευτούν αλλά δεν διαθέτουν δυνατότητα κλιμάκωσης για το χειρισμό μεγάλων συνόλων δεδομένων. Παραδείγματα αυτών των μοντέλων είναι ο ιεραρχικός αλγόριθμος ομαδοποίησης και οι παραλλαγές του (Kelleher et al, 2015).

Κεντροειδή Μοντέλα: Αυτοί είναι αλγόριθμοι επαναληπτικής ομαδοποίησης, στους οποίους η έννοια της ομοιότητας προέρχεται από την εγγύτητα ενός σημείου δεδομένων στο κεντροειδές των συστάδων. Ο αλγόριθμος ομαδοποίησης K-Means είναι ένας δημοφιλής αλγόριθμος που εμπίπτει σε αυτήν την κατηγορία. Σε αυτά τα μοντέλα, ο αριθμός των συστοιχιών που απαιτούνται στο τέλος πρέπει να αναφερθεί εκ των προτέρων, πράγμα που καθιστά σημαντικό να έχουμε προηγούμενη γνώση του συνόλου δεδομένων. Αυτά τα μοντέλα τρέχουν επαναληπτικά για να βρουν την τοπική optima (Kelleher et al, 2015).

Μοντέλα διανομής: Αυτά τα μοντέλα συσσώρευσης βασίζονται στην έννοια του πόσο πιθανό είναι ότι όλα τα σημεία δεδομένων στο σύμπλεγμα ανήκουν στην ίδια διανομή (Για παράδειγμα: Κανονική, Gaussian). Αυτά τα μοντέλα συχνά υποφέρουν από υπερφόρτωση. Ένα δημοφιλές παράδειγμα αυτών των μοντέλων είναι ο αλγόριθμος μεγιστοποίησης προσδοκιών που χρησιμοποιεί πολλαπλές κανονικές κατανομές (Kelleher et al, 2015).

Μοντέλα Πυκνότητας: Αυτά τα μοντέλα αναζητούν το χώρο δεδομένων για περιοχές με ποικίλη πυκνότητα σημείων δεδομένων στον χώρο δεδομένων. Απομονώνει διάφορες περιοχές διαφορετικής πυκνότητας και εκχωρεί τα σημεία δεδομένων σε αυτές τις περιοχές στο ίδιο σύμπλεγμα. Δημοφιλή παραδείγματα μοντέλων πυκνότητας είναι το DBSCAN και το OPTICS (Kelleher et al, 2015).

Αλγόριθμος K-Means: Ο απλούστερος αλγόριθμος μάθησης χωρίς επίβλεψη. Αυτό λειτουργεί με βάση την αρχή της ομαδοποίησης k-means. Αυτό στην πραγματικότητα σημαίνει ότι οι συγκεντρωμένες ομάδες (ομάδες) για ένα δεδομένο σύνολο δεδομένων αντιπροσωπεύονται από μια μεταβλητή «k». Για κάθε σύμπλεγμα ορίζεται ένα

κεντροειδές. Το κεντροειδές είναι ένα σημείο δεδομένων που βρίσκεται στο κέντρο κάθε συστάδας (λαμβάνοντας υπόψη την απόσταση Euclidean). Τα κεντροειδή πρέπει να ορίζονται μακριά από το ένα το άλλο έτσι ώστε η διακύμανση να είναι μικρότερη. Μετά από αυτό, κάθε σημείο δεδομένων στο σύμπλεγμα αντιστοιχεί στο πλησιέστερο κέντρο (Kelleher et al, 2015).

2.4.3 Apriori

Αυτός ο αλγόριθμος, που εισήχθη από τους R Agrawal και R Srikant το 1994, έχει μεγάλη σημασία στην εξόρυξη δεδομένων. Ο αλγόριθμος Apriori, ένας κλασικός αλγόριθμος, είναι χρήσιμος σε συχνές αντικείμενα εξόρυξης και σε σχετικούς κανόνες σύνδεσης. Συνήθως, ο χρήστης χρησιμοποιεί αυτόν τον αλγόριθμο σε βάση δεδομένων που περιέχει μεγάλο αριθμό συναλλαγών. Ένα τέτοιο παράδειγμα είναι τα στοιχεία που οι πελάτες αγοράζουν σε ένα σούπερ μάρκετ.

Βοηθάει τους πελάτες να αγοράζουν τα αντικείμενα τους με ευκολία και βελτιώνει την απόδοση των πωλήσεων του καταστήματος.

Αυτός ο αλγόριθμος έχει χρησιμότητα στον τομέα της υγειονομικής περίθαλψης, καθώς μπορεί να βοηθήσει στην ανίχνευση των ανεπιθύμητων ενεργειών (ADR) με την παραγωγή κανόνων σύνδεσης για να υποδείξει τον συνδυασμό φαρμάκων και τα χαρακτηριστικά του ασθενούς που θα μπορούσαν να οδηγήσουν σε ADRs.

Μαθηματική Προσέγγιση

Στήριξη (Support)

Αυτό το μέτρο δίνει μια ιδέα για το πόσο συχνή είναι ένα σύνολο στοιχείων σε όλες τις συναλλαγές.

Μαθηματικά, η υποστήριξη είναι το κλάσμα του συνολικού αριθμού των συναλλαγών στις οποίες πραγματοποιείται το σύνολο στοιχείων.

$$Support(\{X\} \rightarrow \{Y\}) = \frac{Transactions\ containing\ both\ X\ and\ Y}{Total\ number\ of\ transactions}$$

Αυτοπεπίθηση (Confidence)

Αυτό το μέτρο καθορίζει την πιθανότητα εμφάνισης συνέπειαε επί του καροτσιού δεδομένου ότι το καλάθι έχει ήδη τα προηγούμενα.

Από τεχνική άποψη, η εμπιστοσύνη είναι η υπό όρους πιθανότητα εμφάνισης επακόλουθου δεδομένου του προηγούμενου.

$$Confidence(\{X\} \rightarrow \{Y\}) = \frac{Transactions\ containing\ both\ X\ and\ Y}{Transactions\ containing\ X}$$

Βάρος (Lift)

Το lift ελέγχει τη στήριξη (συχνότητα) των επακόλουθων ενώ υπολογίζει την υποθετική πιθανότητα εμφάνισης {Y} δεδομένης {X}. Το lift είναι ένας πολύ κυριολεκτικός όρος που δίνεται σε αυτό το μέτρο. Σκεφτείτε το ως τον ανελκυστήρα που {X} παρέχει στην εμπιστοσύνη μας ότι έχετε {Y} στο καλάθι. Για να αναδιατυπώσετε, το lift είναι η αύξηση της πιθανότητας να {Y} στο καλάθι με γνώση του {X} να είναι παρούσα σχετικά με την πιθανότητα να {Y} στο καλάθι χωρίς καμία γνώση σχετικά με την παρουσία {X}.

$$Lift(\{X\} \rightarrow \{Y\}) = \frac{(Transactions\ containing\ both\ X\ and\ Y)/(Transactions\ containing\ X)}{Fraction\ of\ transactions\ containing\ Y}$$

2.4.4 Ανάλυση RFM

Η ανάλυση RFM (πρόσφατη, συχνότητα, νομισματική) είναι μια τεχνική μάρκετινγκ που χρησιμοποιείται για να προσδιοριστεί ποσοτικά ποιοι πελάτες είναι οι καλύτεροι, εξετάζοντας πόσο πρόσφατα ο πελάτης αγόρασε (πρόσφατη), πόσο συχνά αγοράζει (συχνότητα) και πόσο ξοδεύει ο πελάτης νομισματικός). Η ανάλυση RFM βασίζεται στο αξίωμα του μάρκετινγκ ότι "το 80% των επιχειρήσεων του χρήστη προέρχεται από το 20% των πελατών του χρήστη".

Για περισσότερα από 30 χρόνια, οι διαφημιστές για μη κερδοσκοπικούς οργανισμούς χρησιμοποίησαν μια ανεπίσημη ανάλυση RFM για να στοχεύσουν τις αποστολές τους στους πελάτες που είναι πιθανότερο να κάνουν δωρεές. Ο συλλογισμός πίσω από το RFM ήταν απλός: οι άνθρωποι που δώρισαν μια φορά ήταν πιο πιθανό να δώσουν ξανά. Με την εμφάνιση καμπανιών μάρκετινγκ ηλεκτρονικού ταχυδρομείου και λογισμικού διαχείρισης σχέσεων με πελάτες, οι αξιολογήσεις RFM έχουν καταστεί ένα σημαντικό εργαλείο. Χρησιμοποιώντας την ανάλυση RFM, στους πελάτες αποδίδεται ένας

αριθμός κατάταξης 1,2,3,4 ή 5 (με το 5 υψηλότερο) για κάθε παράμετρο RFM. Οι τρεις βαθμολογίες μαζί αναφέρονται ως "κύτταρο" RFM. Η βάση δεδομένων ταξινομείται για να προσδιορίσει ποιοι πελάτες ήταν οι "καλύτεροι πελάτες" στο παρελθόν, με την κατάταξη κυττάρων "555" να είναι ιδανική.

Αν και η ανάλυση RFM είναι ένα χρήσιμο εργαλείο, έχει τους περιορισμούς της. Μια εταιρεία πρέπει να προσέχει ώστε να μην υπερνικά τους πελάτες με τις υψηλότερες βαθμολογίες. Οι ειδικοί προειδοποιούν επίσης τους εμπόρους να θυμούνται ότι οι πελάτες με χαμηλή κατάταξη κυττάρων δεν πρέπει να παραμεληθούν, αλλά πρέπει να καλλιεργηθούν για να γίνουν καλύτεροι πελάτες.

Η τμηματοποίηση της συχνότητας της πρόσφατης κρίσης-συχνότητας-νομισματικής αξίας βρίσκεται εδώ και αρκετό καιρό και παρέχει έναν αρκετά απλό αλλά αποτελεσματικό τρόπο για την κατάτμηση των πελατών. Ένα μοντέλο RFM μπορεί να χρησιμοποιηθεί σε συνδυασμό με ορισμένα προγνωστικά μοντέλα για να αποκτήσει ακόμα μεγαλύτερη εικόνα της συμπεριφοράς του πελάτη. Σε αυτήν την ανάρτηση θα συζητήσουμε τρία μοντέλα πρόβλεψης - K-ομαδοποίηση, Logistic Regression και Recommendation και θα δούμε πώς βελτιώνουν τα αποτελέσματα από την ανάλυση RFM.

Το παρακάτω είναι μια ροή υψηλού επιπέδου της ανάλυσης. 1) Υπολογίστε τις παραμέτρους R, F και M. 2) Εφαρμόστε τον αλγόριθμο ομαδοποίησης k-mean σε αυτές τις παραμέτρους για να ομαδοποιήσετε παρόμοιους πελάτες. 3) Σημειώστε ότι οι τιμές εισόδου σε αυτόν τον αλγόριθμο πρέπει να είναι συνεχείς μεταβλητές. 4) Τα K-means και τα offshoots είναι μια δημοφιλής προσέγγιση για την ταξινόμηση λόγω της απλότητας της εφαρμογής και έχουν χρησιμοποιηθεί ευρέως στην κατάτμηση της αγοράς. 5) Ο αριθμός των συστάδων μπορεί να προσδιοριστεί χρησιμοποιώντας τη μέθοδο elbow. 6) Εφαρμόστε αλγορίθμους ταξινόμησης, όπως Logistic Regression και Decision Trees, για να προβλέψετε τη μελλοντική συμπεριφορά των πελατών. 7) Αυτό θα είναι ένα πρόβλημα ταξινόμησης πολλών τάξεων με τον αριθμό των κλάσεων που αντιστοιχούν στον αριθμό των συστάδων από το προηγούμενο βήμα. 8) Χρησιμοποιήστε οποιαδήποτε χαρακτηριστικά πελάτη, όπως ηλικία, φύλο, περιοχή κτλ. ως ανεξάρτητες μεταβλητές στο μοντέλο. 9) Υπολογίστε παράγοντες μεταβλητού ενδιαφέροντος για να καταλάβετε ποια μεταβλητή έχει τον μεγαλύτερο αντίκτυπο στο

αποτέλεσμα. 10) Τέλος, εφαρμόστε αλγορίθμους συστάσεων, όπως φιλτραρίσματα ή περιεχόμενο που βασίζεται στο φιλτράρισμα και τους κανόνες σύνδεσης. Η μέθοδος των κανόνων σύνδεσης συνδέει τις σχέσεις μεταξύ των προϊόντων

Αυτά τα βήματα προσδιορίζουν τις συσχετίσεις μεταξύ των τμημάτων πελατών και των προϊόντων που αγοράζονται από κοινού. Η παραπάνω διαδικασία παρέχει τη δυνατότητα πρόβλεψης της συμπεριφοράς των πελατών έτσι ώστε οι επιχειρήσεις να μπορούν να λάβουν τα κατάλληλα μέτρα για να προσελκύσουν νέους πελάτες ή να διατηρήσουν και να προωθήσουν τους υφιστάμενους πελάτες.

2.4.5 Kmeans

Ο αλγόριθμος Kmeans είναι ένας επαναληπτικός αλγόριθμος που προσπαθεί να χωρίσει το σύνολο δεδομένων σε ξεχωριστές μη αλληλεπικαλυπτόμενες υποομάδες (clusters) που ορίζονται από το Kpre, όπου κάθε σημείο δεδομένων ανήκει σε μία μόνο ομάδα. Προσπαθεί να καταστήσει τα σημεία δεδομένων μεταξύ των συμπλεγμάτων όσο το δυνατόν πιο ομοιόμορφα, διατηρώντας παράλληλα τις ομάδες όσο το δυνατόν πιο διαφορετικές. Αναθέτει σημεία δεδομένων σε ένα σύμπλεγμα έτσι ώστε το άθροισμα της τετραγωνικής απόστασης μεταξύ των σημείων δεδομένων και του κεντροειδούς του συμπλέγματος (ο αριθμητικός μέσος όρος όλων των σημείων δεδομένων που ανήκουν σε αυτό το σύμπλεγμα) είναι το ελάχιστο. Όσο μικρότερη είναι η διακύμανση που έχουμε στα πλαίσια συμπλεγμάτων, τόσο πιο ομοιογενή (παρόμοια) σημεία δεδομένων βρίσκονται μέσα στο ίδιο σύμπλεγμα.

Ο τρόπος με τον οποίο λειτουργεί ο αλγόριθμος kmeans είναι ο εξής: 1) Καθορίστε τον αριθμό των ομάδων K. 2) Αρχικοποιήστε τα κεντροειδή ξεκινώντας πρώτα την αναδιάταξη του συνόλου δεδομένων και στη συνέχεια επιλέγοντας τυχαία τα σημεία δεδομένων K για τα κεντροειδή χωρίς αντικατάσταση. 3) Συνεχίστε να ρωτάτε μέχρι να μην υπάρξει αλλαγή στα κεντροειδή. δηλ. η αλλαγή των σημείων δεδομένων σε ομάδες δεν αλλάζει. 4) Υπολογίστε το άθροισμα της τετραγωνικής απόστασης μεταξύ των σημείων δεδομένων και όλων των κεντροειδών. 5) Προσθέστε κάθε σημείο δεδομένων στο πλησιέστερο σύμπλεγμα (centroid). 6) Υπολογίστε τα centroids για τα clusters λαμβάνοντας τον μέσο όρο όλων των σημείων δεδομένων που ανήκουν σε κάθε σύμπλεγμα.

Η προσέγγιση kmeans ακολουθεί για να λύσει το πρόβλημα ονομάζεται Προσδοκία-μεγιστοποίηση. Το βήμα Ε αντιστοιχεί τα σημεία δεδομένων στο πλησιέστερο σύμπλεγμα. Το βήμα Μ υπολογίζει το κεντρικό στοιχείο κάθε ομάδας.

Ο αλγόριθμος kmeans είναι πολύ δημοφιλής και χρησιμοποιείται σε ποικίλες εφαρμογές όπως τμηματοποίηση της αγοράς, ομαδοποίηση εγγράφων, τμηματοποίηση εικόνας και συμπίεση εικόνας κλπ. Ο στόχος συνήθως όταν υποβάλλονται σε μια ανάλυση συμπλέγματος είναι είτε: 1) Πάρτε μια ουσιαστική διαίσθηση της δομής τα δεδομένα που έχουμε να κάνουμε. 2) Συγκεντρώστε τότε -προβλέψτε πού θα κατασκευαστούν διαφορετικά μοντέλα για διαφορετικές υποομάδες εάν πιστεύουμε ότι υπάρχει μεγάλη ποικιλία στις συμπεριφορές διαφορετικών υποομάδων. Ένα παράδειγμα είναι η συσσώρευση ασθενών σε διαφορετικές υποομάδες και η δημιουργία ενός μοντέλου για κάθε υποομάδα για την πρόβλεψη της πιθανότητας εμφάνισης καρδιακής προσβολής.

Σε αντίθεση με την εποπτευόμενη μάθηση όπου έχουμε την αλήθεια για να αξιολογήσουμε την απόδοση του μοντέλου, η ανάλυση ομαδοποίησης δεν έχει μια σταθερή μετρική αξιολόγησης που μπορούμε να χρησιμοποιήσουμε για να αξιολογήσουμε το αποτέλεσμα διαφορετικών αλγορίθμων ομαδοποίησης. Επιπλέον, δεδομένου ότι τα kmeans απαιτούν το k ως είσοδο και δεν το μαθαίνουν από τα δεδομένα, δεν υπάρχει σωστή απάντηση όσον αφορά τον αριθμό των συστάδων που πρέπει να έχουμε σε οποιοδήποτε πρόβλημα. Μερικές φορές η γνώση και η διαίσθηση του τομέα μπορεί να βοηθήσει αλλά συνήθως δεν συμβαίνει αυτό. Στη μεθοδολογία πρόβλεψης συμπλέγματος, μπορούμε να αξιολογήσουμε πόσο καλά εκτελούνται τα μοντέλα με βάση διαφορετικά συμπλέγματα Κ από τη χρήση συμπλεγμάτων στη μεταγενέστερη μοντελοποίηση.

Ο αλγόριθμος Kmeans είναι καλός στη σύλληψη της δομής των δεδομένων αν τα σμήνη έχουν σφαιρικό σχήμα. Προσπαθεί πάντα να κατασκευάσει ένα ωραίο σφαιρικό σχήμα γύρω από το κέντρο. Αυτό σημαίνει ότι, όταν τα clusters έχουν περίπλοκα γεωμετρικά σχήματα, τα kmeans κάνουν κακή δουλειά στη συγκέντρωση των δεδομένων.

Η ομαδοποίηση Kmeans είναι ένας από τους πιο δημοφιλείς αλγόριθμους συσσώρευσης και συνήθως τα πρώτα πράγματα που εφαρμόζουν οι ασκούντες την

πρακτική όταν επιλύουν τα καθήκοντα συγκέντρωσης για να πάρουν μια ιδέα της δομής του συνόλου δεδομένων. Ο στόχος των kmeans είναι η ομαδοποίηση των σημείων δεδομένων σε ξεχωριστές υπο-ομάδες που δεν αλληλεπικαλύπτονται. Κάνει πολύ καλή δουλειά όταν τα σμήνη έχουν ένα είδος σφαιρικών σχημάτων. Παρόλα αυτά, υποφέρει καθώς τα γεωμετρικά σχήματα των συστάδων αποκλίνουν από τα σφαιρικά σχήματα. Επιπλέον, δεν μαθαίνει επίσης τον αριθμό των συσπειρώσεων από τα δεδομένα και απαιτεί την προκαθορισμό τους. Για να είστε καλός επαγγελματίας, είναι καλό να γνωρίζετε τις υποθέσεις πίσω από αλγόριθμους / μεθόδους, έτσι ώστε ο χρήστης να έχει μια πολύ καλή ιδέα για τη δύναμη και την αδυναμία κάθε μεθόδου. Αυτό θα βοηθήσει τον χρήστη να αποφασίσει πότε θα χρησιμοποιήσει κάθε μέθοδο και υπό ποιες συνθήκες.

3 Business Intelligence

Η «επιχειρησιακή νοημοσύνη» (Business Intelligence - εφεξής BI) έχει καταστεί μια όλο και πιο σημαντική έννοια με τη διαθεσιμότητα «μεγάλων δεδομένων» και την πρόοδο στη μηχανική ευφυΐα (Agarwal & Dgarm 2014). Λαμβάνοντας ευρύτατο ενδιαφέρον τόσο στον ακαδημαϊκό χώρο όσο και στον κλάδο (Shollo & Kautz, 2010), τα συστήματα BI χρησιμοποιούνται πλέον εκτεταμένα σε πολλούς τομείς δραστηριοτήτων που συνεπάγονται τη λήψη αποφάσεων για τη δημιουργία αξίας. Ωστόσο, για να βοηθηθεί η BI να επιτύχει το πλήρες δυναμικό της, οι επαγγελματίες και οι ερευνητές πρέπει να κατανοήσουν καλύτερα τις διαδικασίες μέσω των οποίων οι οργανισμοί μπορούν να αποκτήσουν αξία από τη BI. Μέχρι σήμερα, οι ερευνητές έχουν εξετάσει το BI χρησιμοποιώντας ποικίλες θεωρίες, ερευνητικούς φακούς και εμπειρικές προσεγγίσεις. Ενώ αυτά τα διάφορα ερευνητικά ρεύματα παρέχουν διαφορετικές απόψεις για τη BI, μπορούν επίσης να δυσχεράνουν την οικοδόμηση μιας ολιστικής και ολοκληρωμένης άποψης για την αξία της επιχειρηματικής δραστηριότητας και τη διατήρηση μιας αθροιστικής ερευνητικής παράδοσης. Ενώ πολλοί συγγραφείς ασχολούνται μάλλον με συγκεκριμένα ερευνητικά ερωτήματα σχετικά με το πώς η BI δημιουργεί επιχειρηματική αξία, δεν έχει αναπτυχθεί ολοκληρωμένη ερευνητική ατζέντα για να κατανοήσει τη διαδικασία των οργανώσεων που αποκτούν επιχειρηματική αξία από τη BI. Επομένως, το ερευνητικό ερώτημα που εξετάζεται σε αυτό το έγγραφο είναι: Τι γνωρίζουμε, πόσο καλά γνωρίζουμε και τι πρέπει να γνωρίζουμε για τις διαδικασίες των οργανισμών που αποκτούν επιχειρηματική αξία από τα συστήματα BI; Στόχος αυτής της ανασκόπησης της βιβλιογραφίας είναι να μάθει κανείς σε ποιο βαθμό μπορούμε να απαντήσουμε σε αυτό το ερώτημα βάσει της υπάρχουσας βιβλιογραφίας, να εντοπίσουμε ποια μέρη της απάντησης χρειάζονται περισσότερο έρευνα και να αποκαλύψουν βασικές ερευνητικές ερωτήσεις για μελλοντικές εργασίες.

Αντί να έχει έναν καλά αποδεκτό και συγκεκριμένο ορισμό (Gibson et al, 2004), η BI χρησιμοποιείται συνήθως ως όρο "ομπρέλα" για να περιγράψει μια διαδικασία (Shollo & Kautz, 2010) ή έννοιες και μεθόδους (Sabherwal et al, 2011) που βελτιώνουν τη λήψη αποφάσεων χρησιμοποιώντας τη στήριξη που βασίζεται σε γεγονότα συστήματα. Πολλοί όροι (όπως «επιχειρησιακή ευφυΐα», «επιχειρηματικές αναλύσεις», «μεγάλα

δεδομένα», «εξόρυξη δεδομένων» και «αποθήκευση δεδομένων») χρησιμοποιούνται συχνά στη βιβλιογραφία και οι συγγραφείς περιγράφουν διαφορετικά τη BI ως «διαδικασία και προϊόν "(Jourdan et al, 2008, σελ. 121), "μια διαδικασία, ένα προϊόν και ένα σύνολο τεχνολογιών ή ένας συνδυασμός αυτών" (Shollo & Kautz, 2010 σελ. 87), ή μόνο ένα προϊόν (Seddon et al, 2012). Ως αποτέλεσμα αυτών των διαφορετικών ορισμών και προοπτικών και του αυξανόμενου ενδιαφέροντος για την BI στον ακαδημαϊκό χώρο και της σημασίας για τη βιομηχανία, είναι σημαντικό να συνθέσουμε τη βιβλιογραφία για να καθορίσουμε αυτό που ήδη γνωρίζουμε σχετικά με τη διαδικασία παραγωγής επιχειρηματικής αξίας από τη BI, να γνωρίζουμε και πώς μπορούμε να φτάσουμε εκεί. Υπάρχουν διάφορες μελέτες που συμβάλλουν, με διάφορους τρόπους, σε αυτή τη γνώση. Οι Seddon και συνεργάτες (2012), για παράδειγμα, ανέπτυξαν ένα μοντέλο επιτυχίας BI, αλλά δεν παρουσίασε κενά στη βιβλιογραφία ούτε πρότεινε μελλοντικές κατευθύνσεις. Ομοίως, ενώ οι Arnott και Pervan (2005) ανέλυσαν τις μελέτες BI από το 1990-2003 και οι Jourdan el al. (2008) ανέλυσε τις μελέτες BI από το 1997-2006, ούτε χαρτί επικεντρώθηκε στη διαδικασία μέσω της οποίας η BI συνέβαλε στην επιχειρηματική αξία. Συνεπώς, παραμένει η ανάγκη για μια βαθύτερη ανάλυση των διαδικασιών των οργανισμών που αποκομίζουν αξία από την BI (Sharma et al, 2014).

Τα πρότυπα χρήσης στη λήψη αποφάσεων αφορούν συνήθως τον τύπο της απόφασης που θα υποστηριχθεί και τον τύπο του διαχειριστή που λαμβάνει την απόφαση. Ο λόγος αυτής της εστίασης είναι ότι ο τύπος εργασίας και ο τύπος του χρήστη στο DSS διαφέρουν θεμελιωδώς από τους χρήστες και τις εργασίες που υποστηρίζονται από εταιρικές συναλλαγές, με βάση το διαδίκτυο, κινητά, κοινωνικά συστήματα και άλλα IS. Η εστίαση απόφασης / διαχειριστή είναι μοναδική για το DSS και είναι κεντρική για την κατανόηση των συστημάτων BI. Μια ανασκόπηση της έρευνας περιπτώσεων BI σε όλα τα περιοδικά και στις τέσσερις μεγάλες διασκέψεις AIS (ICIS, ECIS, PACIS, AMCIS) από το 2000 έως το 2016 βρήκε 68 άρθρα. Από αυτά, 13 συστήματα BI που χρησιμοποιούνται με κάποιο τρόπο. Δεν υπάρχουν διευθυντές λήψης αποφάσεων και τύποι χρήσης τύπων αποφάσεων. Αυτό σημαίνει ότι τα πρότυπα χρήσης BI είναι ένα κενό στη βιβλιογραφία της έρευνας του BI.

Από τη σκοπιά της χρήσης BI σε διοικητικό επίπεδο, ο Negash (στο Park et al, 2001) ανέφερε ότι "η BI βοηθά στη λήψη στρατηγικών και επιχειρησιακών αποφάσεων" (σελ. 179) και ότι "η επιχειρησιακή νοημοσύνη χρησιμοποιείται από τους υπεύθυνους λήψης αποφάσεων σε όλη την επιχείρηση. Στα ανώτερα διευθυντικά επίπεδα, είναι η συμβολή στις στρατηγικές και τακτικές αποφάσεις. Σε χαμηλότερα επίπεδα διοίκησης, βοηθά τα άτομα να κάνουν την καθημερινή τους εργασία. " (σελ. 189). Οι Audzeyeva και Hudson (στο Sharma et al, 2014) υποστήριξαν στη μελέτη τους για τα οφέλη του BI ότι "τα βασικά οργανωτικά οφέλη της BI ... περιλαμβάνουν καλύτερες αποφάσεις διαχείρισης τόσο σε μεσαία διοίκηση όσο και σε στρατηγικά επίπεδα και υποστήριξη για την επίτευξη στρατηγικών επιχειρηματικών στόχων". Οι Arnott και Pervan (2005) ως μέρος μιας κριτικής ανάλυσης 25 ετών γενικής έρευνας DSS εξέτασαν το επίπεδο των καθηκόντων λήψης αποφάσεων που εξετάστηκαν στην έρευνα BI. Διαπίστωσαν ότι το 22,5% της έρευνας BI αφορούσε καθήκοντα στρατηγικής λήψης αποφάσεων. Οι Isik και συνεργάτες (στο Clark et al, 2007) ανέφεραν ότι "πολλές εταιρείες χρησιμοποιούν σήμερα BI κυρίως για δομημένη λήψη αποφάσεων με βάση εσωτερικά δεδομένα" (σελ. 14). Συγκεκριμένα, αυτό σημαίνει ότι, σε κάποιο βαθμό, η BI στοχεύει στην αντιμετώπιση πολλών τύπων λήψης αποφάσεων σε οργανισμούς.

Για την διερεύνηση των προτύπων των συστημάτων BI, χρησιμοποιούνται δύο ομάδες θεωρίας. Το πρώτο είναι το πλαίσιο του Gorry και του Scott Morton. Το πλαίσιο οδήγησε στην ανάπτυξη του πεδίου DSS και εξακολουθεί να επηρεάζει την έρευνα DSS και BI. Το δεύτερο θεωρητικό υπόβαθρο είναι η κυρίαρχη σύγχρονη προσέγγιση για την κατανόηση της λήψης αποφάσεων από τη συμπεριφορά των οικονομικών. Αυτό ακολουθείται από ένα σημείωμα σχετικά με τη μεταφορά θεωρίας μεταξύ των τύπων DSS και τη φύση των πλαισίων στη θεωρία IS.

Ο καθορισμός των διαδικασιών διαχείρισης και των καθηκόντων λήψης αποφάσεων σε τυπολογίες τριών επιπέδων υπήρξε επίμονο θέμα στην επιχειρηματική έρευνα από τη δεκαετία του 1960. Αυτές οι τυπολογίες έχουν επιτύχει την κατάσταση του παραδείγματος και συχνά χρησιμοποιούνται χωρίς αναφορά (για παράδειγμα, (Agarwal & Dhar, 2014)). Η πιο συνηθισμένη τυπολογία της διαδικασίας διαχείρισης είναι ο στρατηγικός σχεδιασμός / έλεγχος διαχείρισης / έλεγχος λειτουργίας του Anthony (Gibson et al, 2004). Σύμφωνα με τον Anthony και Dearden (στο Sabherwal

et al, 2011), ο στρατηγικός σχεδιασμός είναι η διαδικασία λήψης αποφάσεων σχετικά με τους στόχους της οργάνωσης, τους πόρους που απαιτούνται για την επίτευξη αυτών των στόχων και τις πολιτικές απόκτησης και χρήσης αυτών των πόρων. ο έλεγχος της διαχείρισης είναι η διαδικασία με την οποία οι διευθυντές διαβεβαιώνουν ότι οι πόροι αποκτώνται και χρησιμοποιούνται αποτελεσματικά και αποτελεσματικά για την επίτευξη των στόχων του οργανισμού. και ο επιχειρησιακός έλεγχος είναι η διαδικασία διασφάλισης ότι τα συγκεκριμένα καθήκοντα εκτελούνται αποτελεσματικά και αποτελεσματικά. Η τυπολογία της διαδικασίας δεν είναι ισομορφική με τα επίπεδα διαχείρισης, αλλά σχετίζεται κατά κάποιο τρόπο. Για παράδειγμα, ένα εκτελεστικό που βρίσκεται στο ανώτατο επίπεδο ενός οργανισμού μπορεί να αντιμετωπίσει στρατηγικά και τακτικά καθήκοντα και να χρησιμοποιήσει μια σειρά λειτουργικών και διαχειριστικών διαδικασιών ελέγχου. Ωστόσο, το γενικό επιχείρημα είναι ότι όσο υψηλότερο είναι ότι ένας διαχειριστής είναι σε έναν οργανισμό, τόσο πιο πιθανό είναι να χρησιμοποιήσει διαδικασίες στρατηγικού σχεδιασμού και να πάρει στρατηγικές αποφάσεις. Η τυπολογία του Anthony είναι ευρέως αποδεκτή στις επιχειρηματικές έρευνες και οι κριτικές είναι σπάνιες. Μια εξαίρεση είναι ο Langfield-Smith (στο Agarwal et al, 2014) ο οποίος ισχυρίστηκε ότι από την άποψη της λογιστικής διαχείρισης «τα τεχνητά όρια μεταξύ του επιχειρησιακού, διαχειριστικού και στρατηγικού ελέγχου, όπως περιγράφεται αρχικά από τον Anthony (Gibson et al, 2004), μπορεί να μην κατέχουν πλέον». (σελ. 209). Οι περισσότεροι ερευνητές θεωρούν την τυπολογία του Anthony ως συνέχεια και όχι ως διακριτές κατηγορίες.

Η τυπολογία των τριών επιπέδων των καθηκόντων λήψης αποφάσεων που έχει φτάσει στο πρότυπο είναι το μοντέλο λήψης αποφάσεων του νικητή του βραβείου Νόμπελ Herbert Simon (Agarwal et al, 2014). Το μοντέλο φάσης θεωρεί ότι η λήψη αποφάσεων λαμβάνει χώρα σε τρεις διαδοχικές, επαναληπτικές και επαναληπτικές διαδικασίες πληροφοριών (συλλογή δεδομένων), σχεδίαση (φθάνοντας σε εναλλακτικές λύσεις) και επιλογή (επιλέγοντας την καλύτερη εναλλακτική λύση). Ένα σημαντικό μέρος του μοντέλου φάσης είναι η έννοια της δομημένης απόφασης. Μια τελείως δομημένη απόφαση είναι εκείνη όπου μπορούν να προσδιοριστούν όλες οι φάσεις λήψης αποφάσεων. μια εντελώς μη δομημένη απόφαση είναι εκείνη που δεν μπορεί να διατυπωθεί καμιά πτυχή των φάσεων απόφασης. Η ύπαρξη μιας συνέχειας μεταξύ

δομημένων και μη δομημένων αποφάσεων είναι ημιδομημένα καθήκοντα λήψης αποφάσεων που παρουσιάζουν διάφορους βαθμούς δομής ή σαφήνειας ορισμού και κατανόησης.

Το βασικό άρθρο της γενικής πειθαρχίας του DSS είναι το έγγραφο του 1971 για το πλαίσιο των πληροφοριακών συστημάτων διαχείρισης από τους Anthony Gorry και Michael Scott Morton. Το πλαίσιο τους βασίστηκε σε ένα συνδυασμό της διαδικασίας διαχείρισης του Αντώνιου και των τυπολογιών δομής της απόφασης του Simon. Τα καθήκοντα έχουν μειωμένα επίπεδα δομής και ο Gorry και ο Scott Morton ονομάζονται IS που μπορούν να υποστηρίξουν αυτά τα καθήκοντα "συστήματα υποστήριξης αποφάσεων". Πάνω από τη γραμμή χαρακτήριζαν την τεχνολογία πληροφορικής ως δομημένη λειτουργική IS. σήμερα πολλά από αυτά θα θεωρούνται DSS. Η σημαντική συνέπεια είναι ότι το DSS μπορεί να υποστηρίξει τα περισσότερα από τα κύτταρα στο πλαίσιο. Περαιτέρω, υποστήριξαν ότι με την πάροδο του χρόνου, με την αύξηση της έρευνας και της πρακτικής, η γραμμή θα μετατοπίζονταν κάτω από το σχήμα καθώς τα ημι-δομημένα καθήκοντα δομούνται. Τα δομημένα καθήκοντα επιχειρησιακού ελέγχου είναι τα πιο εύκολα για έναν επαγγελματία πληροφορικής να αντιληφθεί και στη συνέχεια να αναπτύξει συστήματα που υποστηρίζουν. Ο Keen και ο Scott Morton (Shollo & Kautz, 2010) πρότειναν ότι τα αδόμητα καθήκοντα υποστηρίζονται κυρίως από την ανθρώπινη διαίσθηση. Οι Kirs και συνεργάτες. (Sharma et al, 2014) παρείχαν μια πειραματική επικύρωση του πλαισίου Gorry και Scott Morton, που κατά το χρόνο αυτό δικαιολόγησε τη σφαιρική θέση του πλαισίου. Το πλαίσιο του Gorry και του Scott Morton είναι μία από τις σημαντικότερες συμβολές στην έρευνα του DSS και με 2233 αναφορές είναι ένα από τα πιο αναφερόμενα έγγραφα σε όλες τις έρευνες του IS.

Το κύριο ζήτημα με το πλαίσιο Gorry και Scott Morton είναι η εγκυρότητα του μοντέλου φάσης λήψης αποφάσεων του Simon - η πηγή του κάθετου άξονα του πλαισίου. Το μοντέλο φάσης του Simon αναπτύχθηκε στη δεκαετία του 1940 και το Simon's είναι ένα διαφορετικό είδος υποτροφίας στην τρέχουσα επιχειρηματική έρευνα. οι περισσότερες εκδόσεις του Simon θα ταξινομούνται τώρα ως εννοιολογικές μελέτες. Η φύση της έρευνας των επιχειρήσεων και της συμπεριφορικής επιστήμης είναι ριζικά διαφορετική σήμερα και τα πρότυπα αυστηρότητας και εγκυρότητας και οι στατιστικές τεχνικές που χρησιμοποιούνται σήμερα δεν υπήρχαν όταν ο Simon

ανέπτυξε τη θεωρία λήψης αποφάσεων. Το πρόβλημα είναι όπως ο Lipschitz και ο Bar-Ilan (στο Sharma et al, 2014) αναφέρονται: "Λαμβάνοντας υπόψη την ποικιλία και την πανταχού παρούσα κατάσταση των μοντέλων φάσης, είναι εκπληκτικό να διαπιστώσουμε ότι οι εμπειρικές ενδείξεις για την περιγραφική και συντακτική εγκυρότητά τους είναι πολύ περιορισμένες". (σελ. 48). Οι Lipschitz και Bar-Ilan διεξήγαγαν πειραματικές έρευνες που διαπίστωσαν αποδεικτικά στοιχεία για την κανονιστική ισχύ του μοντέλου φάσης και μόνο αδύναμη υποστήριξη για την περιγραφική ισχύ του. Το συμπέρασμα από την εμπειρική δοκιμή του μοντέλου φάσης είναι ότι δεν διαθέτει την απαραίτητη επιστημονική ισχύ για να αποτελέσει μέρος ενός σημαντικού και σημαντικού πλαισίου όπως το Gorry και ο Scott Morton's. Ένα άλλο ζήτημα με το πλαίσιο των Gorry και Scott Morton είναι ότι, όπως και η έρευνα του Simon σχετικά με τη λήψη αποφάσεων, είναι μια εννοιολογική μελέτη και η ανάθεση των καθηκόντων και των συστημάτων λήψης αποφάσεων στο πλαίσιο βασίστηκε σε άποψη και όχι σε εμπειρική έρευνα.

Οι τεχνολογίες που σχετίζονται με την τεχνολογία BI (ακόμα) κατατάσσονται στις κορυφαίες τεχνολογικές προτεραιότητες πολλών επικεφαλής πληροφοριακών λειτουργιών, ενώ οι συνολικές δαπάνες λογισμικού BI αναμένεται να αυξηθούν κατά 7% σε σχέση με το προηγούμενο έτος (Gartner, 2013a). Ένας τέτοιος ενθουσιασμός μπορεί να αποδοθεί στην αυξανόμενη σημασία των συστημάτων BI, τα οποία θεωρούνται τακτικά ως "μια ευρεία κατηγορία τεχνολογιών, εφαρμογών και διαδικασιών για τη συλλογή, αποθήκευση, πρόσβαση και ανάλυση δεδομένων ώστε οι χρήστες να λαμβάνουν καλύτερες αποφάσεις" (Wixom & Watson, 2010, σελ. 14). Ωστόσο, η εφαρμογή ενός συστήματος BI δεν συνεπάγεται μόνο την αγορά ενός συνδυασμού λογισμικού και υλικού. Πρόκειται μάλλον για μια σύνθετη επιχείρηση που απαιτεί κατάλληλη υποδομή και πόρους για μεγάλο χρονικό διάστημα (Yeoh & Koronios, 2010). Στην πραγματικότητα, έχουν αναφερθεί περιπτώσεις όπου οι μεγάλες επενδύσεις σε διάφορες πρωτοβουλίες BI σε μεγαλύτερες περιόδους είχαν ελάχιστα ή καθόλου οφέλη για τους οργανισμούς που τις υλοποίησαν (Williams & Williams, 2007).

Τα παρακάτω περιγράφουν λεπτομερέστερα τα αποτελέσματα της ανασκόπησης της βιβλιογραφίας και προτείνουν ευκαιρίες για έρευνα.

3.1 Οργανωτική απόδοση

Σύμφωνα με τους Soh και Markus (Sharma et al, 2014), οι εννοιολογικοποιήσεις της οργανωτικής απόδοσης εξαρτώνται από τον τρόπο με τον οποίο αντιμετωπίζονται οι οργανισμοί. Περιγράφουν τρεις προσεγγίσεις (Sharma et al, 2014). Πρώτον, οι οργανώσεις μπορεί να θεωρηθούν λογικές. με μέτρα απόδοσης που αντικατοπτρίζουν την επιτυχή ολοκλήρωση του στόχου. Δεύτερον, οι οργανώσεις μπορεί να θεωρηθούν ως συμμαχίες εκλογικών περιφερειών. με την απόδοση να μετράται με την ικανοποίηση των συστατικών στοιχείων όπως οι εργαζόμενοι και οι πελάτες. Τέλος, μπορούν να θεωρηθούν ως οντότητες που «συμμετέχουν στη διαπραγμάτευση σχέσεων με το περιβάλλον τους, εισάγοντας διάφορους σπάνιους πόρους που πρέπει να επιστραφούν ως αξιόλογη παραγωγή» (Sharma et al, 2014, σελ.36). τα κατάλληλα μέτρα μέτρησης επιδόσεων περιλαμβάνουν την ικανότητα του οργανισμού να αποκτά λιγοστά μέσα και να τα μετατρέπει παραγωγικά σε αποτιμημένα αποτελέσματα (Sharma et al, 2014). Δεδομένου ότι και οι τρεις κύριες προοπτικές για τις οργανώσεις είναι ταυτόχρονα έγκυρες στις περισσότερες οργανώσεις, οι διερευνήσεις της οργανωτικής απόδοσης πρέπει να λαμβάνουν υπόψη όλα τα μέτρα απόδοσης που αντικατοπτρίζουν τις διαφορετικές προοπτικές των οργανισμών.

3.2 Επιπτώσεις του Business Intelligence

Οι επιπτώσεις της BI είναι η πρώτη απαραίτητη προϋπόθεση για τη βελτίωση των οργανωτικών επιδόσεων (Sharma et al, 2014). BI Επιπτώσεις αφορούν μια κατάσταση όταν οι οργανισμοί έχουν επιτύχει ένα ή περισσότερα από τα ακόλουθα αποτελέσματα: βελτίωση της λειτουργικής αποτελεσματικότητας των διαδικασιών. νέα / βελτιωμένα προϊόντα ή υπηρεσίες. και / ή ενισχυμένη οργανωτική νοημοσύνη και δυναμική οργανωτική δομή (Sharma et al, 2014).

BI Impacts αποτέλεσαν το κύριο μέλημα των μελετών BI για τα τελευταία 15 χρόνια. Οι ερευνητές έχουν δείξει ειδικότερα ότι η BI μπορεί να χρησιμοποιηθεί για τη βελτίωση της επιχειρησιακής αποτελεσματικότητας μιας επιχείρησης, ελαχιστοποιώντας τους πελάτες με εσφαλμένη στόχευση (Agarwal & Dhar, 2014), μετατρέποντας τις επιχειρηματικές διαδικασίες (Agarwal & Dhar, 2014), εμπλουτίζοντας την οργανωτική νοημοσύνη (Agarwal & Dhar, 2014) και την ανάπτυξη

νέων ή βελτιωτικών προϊόντων ή υπηρεσιών (Agarwal & Dhar, 2014). Ωστόσο, η BI λογοτεχνία δεν μιλούσε για το πώς αυτές οι επιπτώσεις BI συμπληρώνουν άλλους εσωτερικούς και εξωτερικούς παράγοντες για τη δημιουργία επιχειρηματικής αξίας.

3.3 Ανταγωνιστικό Πλεονέκτημα του Business Intelligence

Οι επιπτώσεις των τεχνολογιών πληροφορικής είναι σημαντικές και αναγκαίες, αλλά δεν επαρκούν για να βελτιώσουν τις οργανωτικές τους επιδόσεις εάν οι επιχειρηματικές συνθήκες δεν είναι ευνοϊκές. Με βάση τα μοντέλα αξίας των τεχνολογιών πληροφορικής (Agarwal & Dhar, 2014), οι απαραίτητες συνθήκες και οι πιθανολογικοί παράγοντες που τα μοντέλα αυτά υποδεικνύουν ότι έχουν ζωτική σημασία για τις επιπτώσεις της BI στη βελτίωση των οργανωτικών επιδόσεων είναι η ανταγωνιστική θέση ενός οργανισμού, η ανταγωνιστική δυναμική, .

Μια ισχυρή αρχική ανταγωνιστική θέση στο ανταγωνιστικό τοπίο στην οποία δραστηριοποιείται μια επιχείρηση είναι μια ευνοϊκή επιχειρηματική συνθήκη που συμβάλλει στη θετική σύνδεση μεταξύ των επιπτώσεων της BI και της βελτίωσης της οργανωτικής απόδοσης. Με ένα ισχυρό ανταγωνιστικό πλεονέκτημα, η επιχείρηση εστίασης θα πρέπει να είναι σε θέση να μετατρέπει τις ευνοϊκές επιπτώσεις της BI στη βελτίωση της οργανωτικής απόδοσης.

Ανταγωνιστική δυναμική ως καθοριστικοί παράγοντες στις οργανωτικές επιδόσεις (Agarwal & Dhar, 2014). Οι Soh και Markus (στο Gibson et al, 2004) υποστηρίζουν ότι η ευνοϊκή ανταγωνιστική δυναμική (δηλαδή η μη ανταπόκριση ή η βραδεία ανταπόκριση από τους ανταγωνιστές) είναι μία από τις πιθανολογούμενες συνθήκες υποστήριξης των επιπτώσεων της BI που έχουν ως αποτέλεσμα οργανωτική απόδοση. Εάν οι επιχειρήσεις επωφεληθούν από την πλούσια οργανωτική νοημοσύνη και τα νέα και βελτιωμένα προϊόντα και υπηρεσίες από τη BI (δηλ. Τις επιπτώσεις BI), ο βαθμός ανταγωνιστικής πίεσης από τους ανταγωνιστές στις επιχειρήσεις θα μειωθεί (Bidan et al, 2012). Η μελέτη της ανταγωνιστικής δυναμικής στο περιβάλλον BI θα βοηθήσει στην καλύτερη κατανόηση του τρόπου με τον οποίο τα BI Impacts μπορούν να μετατραπούν σε βελτίωση της οργανωτικής απόδοσης.

Τα χαρακτηριστικά της βιομηχανίας βασίζονται στον τρόπο με τον οποίο εφαρμόζεται η BI στο πλαίσιο μιας κεντρικής επιχείρησης για την παραγωγή επιχειρηματικής αξίας

και περιλαμβάνουν την ανταγωνιστικότητα, τη ρύθμιση και την ταχύτητα της αλλαγής (Bidan et al, 2012). Τα αποτελέσματα της εξέτασης των παραγόντων της βιομηχανίας στις μελέτες BI δείχνουν ότι ένα πλήρως διαμορφωμένο σύστημα BI παρέχει διαφοροποιημένη αξία με βάση τους τύπους της βιομηχανίας στην οποία λειτουργεί μια επιχείρηση (Bidan et al, 2012). Για παράδειγμα, οι Elbashir et al. (στο Agarwal et al, 2014) εξηγούν ότι οι βιομηχανίες που δεν υπηρετούν δείχνουν ισχυρότερες σχέσεις μεταξύ των επιπτώσεων της BI και της οργανωτικής απόδοσης από ό, τι οι βιομηχανίες υπηρεσιών, ισχυριζόμενοι ότι οι τομείς που δεν εξυπηρετούν φαίνεται να είναι σε θέση να μετατρέπουν αποτελεσματικότερα τις επιπτώσεις της BI σε βελτιώσεις στην απόδοση των οργανώσεων. Η υιοθέτηση ενός ισχυρού μετριασμού του κινδύνου της BI για την πρόβλεψη των μεταδοτικών τραπεζικών αποτυχιών και ο καθορισμός των προτεραιοτήτων της εισροής κεφαλαίων μετά τις κρίσεις βοηθά στην επιβίωση των τραπεζών. Όσον αφορά τις διαφορές στη ρύθμιση του κλάδου, οι Abbasi et al. (2016) αναφέρουν ότι ο κανονισμός είναι βαρύς, για παράδειγμα, στον τομέα του ελέγχου, δεδομένου ότι οι κανονισμοί περιορίζουν τους ελεγκτές από την πρόσβαση σε εσωτερικά δεδομένα έως ότου πραγματοποιηθεί έλεγχος ή έρευνα. Υποδεικνύουν ότι οι επιχειρήσεις θα μπορούσαν να χρησιμοποιούν τα εργαλεία BI μαζί με τις διαθέσιμες στο κοινό πληροφορίες για τη βελτίωση της λήψης αποφάσεων και την ιεράρχηση των ερευνητικών πόρων.

4 Tableau

Το λογισμικό Tableau είναι ένα εργαλείο για την εξερεύνηση, την ανάλυση και την παρουσίαση δεδομένων σε οπτική, διαδραστική μορφή. Χρησιμοποιούμενη από χιλιάδες εταιρείες, δημοσιογράφους και μη κερδοσκοπικούς οργανισμούς, η αποστολή του Tableau είναι να "βοηθήσει τους ανθρώπους να δουν και να κατανοήσουν τα δεδομένα" (Deardorff, 2016). Ένας από τους λόγους για τους οποίους ο Tableau έχει γίνει τόσο δημοφιλής είναι ότι η διασύνδεσή του είναι σχετικά απλή στη χρήση. Επομένως, οι χρήστες χωρίς καμία γνώση προγραμματισμού μπορούν εύκολα να χειριστούν τα δεδομένα για να δημιουργήσουν μια μεγάλη ποικιλία διαδραστικών απεικονίσεων.

Οι βιβλιοθηκονόμοι μπορούν να χρησιμοποιήσουν το Tableau για μια ευρεία ποικιλία εργασιών ανάλυσης και παρουσίασης. Οι βιβλιοθηκονόμοι θα μπορούσαν να μοιράζονται τα αποτελέσματα των ερευνών μέσω ενός διαδραστικού, οπτικά επιτακτικού ταμπλό με μια ποικιλία γραφημάτων και διαγραμμάτων, τα οποία θα μπορούσαν να ενσωματώσουν στις ιστοσελίδες βιβλιοθηκών τους για να διερευνήσουν τους προστάτες. Οι διαχειριστές θα μπορούσαν να παρακολουθήσουν τον αριθμό των κυκλοφοριών με την πάροδο του χρόνου με ένα απλό διάγραμμα Excel ή θα μπορούσαν να κάνουν ένα πιο ισχυρό γράφημα στο Tableau που θα τους επέτρεπε να εφαρμόσουν μια σειρά φίλτρων (όπως υποκατάστημα βιβλιοθήκης, ώρα της ημέρας, περιοχή συλλογής) για βαθύτερη ανάλυση.

Με το Tableau, οι βιβλιοθηκονόμοι μπορούν να απεικονίσουν μια μεγάλη ποικιλία τύπων δεδομένων, συμπεριλαμβανομένων των χρονικών, χωρικών, τοπικών και δεδομένων δικτύου. Οι διαθέσιμοι τύποι οπτικοποίησης περιλαμβάνουν πίνακες, γραφήματα, χάρτες θερμότητας, χάρτες δέντρων, ιστογράμματα, διαγράμματα φυσαλίδων, παγκόσμιους χάρτες και πολλά άλλα. Επειδή το εργαλείο υποδεικνύει ποιες μεταβλητές απαιτούνται για κάθε τύπο απεικόνισης, είναι σχετικά εύκολο να εξερευνήσετε μια ποικιλία από διαφορετικές απόψεις των δεδομένων κάποιου. Μόλις οι χρήστες επιλέξουν μια βασική μορφή ή γράφημα, μπορούν να προσθέσουν περισσότερες διαστάσεις στις απεικονίσεις τους χρησιμοποιώντας χρώματα, σχήματα και μεγέθη για διαφορετικές μεταβλητές. Όταν ολοκληρωθεί η απεικόνιση, οι χρήστες μπορούν να εργαστούν με το ταμπλό ή το χαρακτηριστικό ιστορίας για να παράγουν

ένα γυαλισμένο τελικό προϊόν. Τα πίνακες ελέγχου Tableau είναι ένας τρόπος για να παρουσιάσετε πολλές απεικονίσεις που είναι βελτιωμένες με σχολιασμούς και φίλτρα και μπορεί να είναι ένας τρόπος για να παρουσιάσετε μια γρήγορη επισκόπηση των δεδομένων κάποιου. Το χαρακτηριστικό γνώρισμα της ιστορίας επιτρέπει στους χρήστες του Tableau να δημιουργήσουν μια ηλεκτρονική παρουσίαση που επιτρέπει στους αναγνώστες να πλοηγούν σε ξεχωριστές προβολές των δεδομένων που παρουσιάζονται σε πιο αφηγηματική μορφή.

Επί του παρόντος, το Tableau είναι διαθέσιμο τόσο για Windows όσο και για Mac, ως δωρεάν έκδοση που βασίζεται σε σύννεφο, που ονομάζεται Tableau Public, καθώς και εκδόσεις desktop, server και online που βασίζονται σε αμοιβές. Οι κύριες διαφορές μεταξύ των δωρεάν και πληρωμένων εκδόσεων του λογισμικού είναι οι τύποι αρχείων που μπορούν να μεταφορτωθούν (η έκδοση που βασίζεται σε αμοιβές προσφέρει περισσότερες επιλογές, συμπεριλαμβανομένης της δυνατότητας απευθείας σύνδεσης με μια ποικιλία βάσεων δεδομένων). τον τρόπο με τον οποίο μπορούν να μοιραστούν τα αρχεία (η πληρωμένη έκδοση επιτρέπει στους χρήστες να εξάγουν αρχεία στην ελεύθερη εφαρμογή Reader Tableau ή να τα ενσωματώσουν σε έναν ιστότοπο, αλλά τα αρχεία Tableau Public μπορούν να προβληθούν μόνο στο Tableau Public ή ενσωματώνοντάς τα σε έναν ιστότοπο). και ο τρόπος αποθήκευσης των αρχείων (η πληρωμένη έκδοση επιτρέπει στους χρήστες να αποθηκεύουν αρχεία τοπικά ή σε απευθείας σύνδεση, ενώ οι χρήστες του Tableau Public πρέπει να αποθηκεύουν τα αρχεία τους στον ιστότοπο του Tableau Public). Ενώ η δωρεάν έκδοση του Tableau Public μπορεί να είναι ένας εξαιρετικός τρόπος για να ξεκινήσετε με το λογισμικό, είναι σημαντικό να θυμάστε ότι όλα τα αρχεία αποθηκεύονται στο προφίλ ενός χρήστη στην ιστοσελίδα του Tableau Public, όπου είναι προσβάσιμα σε οποιονδήποτε με τον ενιαίο εντοπιστή πόρων URL).

5 Περιγραφή Προσέγγισης

5.1 Τεχνικές και Αλγόριθμοι

Θα κάνουμε ανάλυση RFM ως πρώτο βήμα και στη συνέχεια θα συνδυάσουμε το RFM με τους αλγόριθμους πρόβλεψης (k-means).

Η Ανάλυση RFM απαντά σε αυτά τα ερωτήματα:

• Ποιοι είναι οι καλύτεροι πελάτες μας;

• Ποιος έχει τη δυνατότητα να μετατραπεί σε πιο κερδοφόρους πελάτες;

• Ποιοι πελάτες πρέπει να διατηρήσουμε;

• Ποια ομάδα πελατών είναι πιο πιθανό να απαντήσει στην τρέχουσα καμπάνια μας;

Η ανάλυση RFM βασίζεται σε μια απλή τεχνική

Η ανάλυση RFM (πρόσφατη, συχνότητα, νομισματική) αποτελεί αποδεδειγμένο μοντέλο μάρκετινγκ για την κατάτμηση πελατών βάσει συμπεριφοράς. Ομαδοποιεί τους πελάτες με βάση το ιστορικό συναλλαγών τους - πόσο πρόσφατα, πόσο συχνά και πόσο αγόραζαν.

Το RFM βοηθά τους πελάτες να χωριστούν σε διάφορες κατηγορίες ή ομάδες, προκειμένου να προσδιορίσουν τους πελάτες που είναι πιο πιθανό να ανταποκριθούν στις προωθητικές ενέργειες αλλά και για τις μελλοντικές υπηρεσίες εξατομίκευσης.

• RECENCY (R): Ημέρες από την τελευταία αγορά

• ΣΥΧΝΟΤΗΤΑ (F): Συνολικός αριθμός αγορών

• ΝΟΜΙΣΜΑΤΙΚΗ ΑΞΙΑ (M): Συνολικά χρήματα που ο πελάτης αυτός δαπάνησε.

Recency

Για να υπολογίσουμε την πρόσφατη εμφάνιση, πρέπει να επιλέξουμε ένα σημείο ημερομηνίας από το οποίο αξιολογούμε πόσες ημέρες πριν ήταν η τελευταία αγορά του πελάτη.

Συχνότητα

Η συχνότητα μας βοηθά να γνωρίζουμε πόσες φορές ένας πελάτης αγόρασε από εμάς. Για να γίνει αυτό, πρέπει να ελέγξουμε πόσα τιμολόγια έχουν καταχωριστεί από τον ίδιο πελάτη.

Νομισματική Αξία¶

Το νομισματικό χαρακτηριστικό απαντά στην ερώτηση: Πόσα χρήματα δαπάνησε ο πελάτης με την πάροδο του χρόνου;

Για παράδειγμα:

Καλύτεροι πελάτες - Πρωταθλητές: Να τους ανταμείψετε. Μπορούν να υιοθετήσουν πρώιμα νέα προϊόντα. Προτείνετε τους "Ανατρέξτε σε έναν φίλο".

Σε κίνδυνο: Στείλτε τους εξατομικευμένα μηνύματα ηλεκτρονικού ταχυδρομείου για να τα ενθαρρύνετε να ψωνίζουν.

Για να αποκτήσουμε ακόμα μεγαλύτερη εικόνα της συμπεριφοράς των πελατών, μπορούμε να σκάψουμε βαθύτερα στη σχέση μεταξύ των μεταβλητών RFM.

Το μοντέλο RFM μπορεί να χρησιμοποιηθεί σε συνδυασμό με ορισμένα μοντέλα πρόβλεψης, όπως η ομαδοποίηση k-means, η Logistic Regression και η σύσταση για την παραγωγή καλύτερων ενημερωτικών αποτελεσμάτων σχετικά με τη συμπεριφορά των πελατών.

Θα πάμε για k-μέσα αφού έχει χρησιμοποιηθεί ευρέως για την τμηματοποίηση της αγοράς και προσφέρει το πλεονέκτημα ότι είναι απλή στην εφαρμογή.

Η ομαδοποίηση K-μέσων είναι ένας απλός αλγόριθμος μάθησης χωρίς επίβλεψη ο οποίος χρησιμοποιείται για την επίλυση προβλημάτων συσσωμάτωσης. Ακολουθεί μια απλή διαδικασία ταξινόμησης ενός δεδομένου συνόλου δεδομένων σε έναν αριθμό ομάδων, που ορίζονται από το γράμμα "k", το οποίο έχει οριστεί εκ των προτέρων. Τα clusters τοποθετούνται στη συνέχεια ως σημεία και όλες οι παρατηρήσεις ή τα σημεία δεδομένων συνδέονται με το πλησιέστερο σύμπλεγμα, υπολογίζονται, προσαρμόζονται και στη συνέχεια ξεκινάει η διαδικασία χρησιμοποιώντας τις νέες ρυθμίσεις έως ότου επιτευχθεί το επιθυμητό αποτέλεσμα.

Για την ανάλυση του καλαθιού αγοράς θα χρησιμοποιήσουμε τον αλγόριθμο A-priori
Ο αλγόριθμος Apriori είναι ένας κλασικός αλγόριθμος στην εξόρυξη δεδομένων. Χρησιμοποιείται για τα συνήθη σύνολα στοιχείων εξόρυξης και τους σχετικούς κανόνες σύνδεσης. Έχει σχεδιαστεί να λειτουργεί σε μια βάση δεδομένων που περιέχει πολλές συναλλαγές, για παράδειγμα, αντικείμενα που έφεραν οι πελάτες σε ένα κατάστημα.
Είναι πολύ σημαντικό για την αποτελεσματική Ανάλυση Αγοράς Καλάθι και βοηθά τους πελάτες να αγοράζουν τα στοιχεία τους με μεγαλύτερη ευκολία που αυξάνει τις πωλήσεις των αγορών. Έχει επίσης χρησιμοποιηθεί στον τομέα της υγειονομικής περίθαλψης για την ανίχνευση ανεπιθύμητων ενεργειών. Παράγει κανόνες σύνδεσης που υποδεικνύουν ποιοι όλοι οι συνδυασμοί φαρμάκων και τα χαρακτηριστικά των ασθενών οδηγούν σε ανεπιθύμητες ενέργειες.

5.2 Περιγραφή Συνόλου Δεδομένων

Το σύνολο δεδομένων που χρησιμοποιήθηκε στην διπλωματική εργασία πάρθηκε από το https://archive.ics.uci.edu/ml/datasets/Online%20Retail?fbclid=IwAR0wdBleo-Y39LRIfYLv_T4aTlOBBeiVJZmTnkd8z1GOWDPVkNuHG6zsvSs
και περιέχει όλες τις συναλλαγές που πραγματοποιήθηκαν στο χρονικό διάστημα 01/12/2010 – 09/12/2011 για ένα κατάστημα με βάση το Ηνωμένο Βασίλειο.
Ο αριθμός των παρατηρήσεων είναι 541909. Η εταιρεία εξάγει κυρίως προιόντα τα οποία είναι κυρίως δώρα κάθε είδους. Πολλοί από τους πελάτες ασχολούνται με το χονδρικό εμπόριο.

Περιγραφή στηλών του συνόλου δεδομένων:
InvoiceNo: 6ψήφιος μοναδικός κωδικός που περιγράφει την κάθε συναλλαγή. Αν ξεκινάει με το γράμμα ‘C’ σημαίνει πως η παραγγελία ακυρώθηκε
StockCode: 5ψήφιος αριθμός που περιγράφει το κάθε προιόν
Description: Περιγράφει το όνομα του προιόντος

Quantity: Περγράφει τον αριθμό προιόντων ανά συναλλαγή

InvoiceDate: Περιγράφει την ακριβή ημερομηνία που πραγματοποιήθηκε η κάθε συναλλαγή.

UnitPrice: Περιγράφει την τιμή του κάθε προιόντος

CustomerID: 5ψήφιος αριθμός που ανατίθεται σε κάθε καταναλωτή

Country: Το όνομα της χώρας προέλευσης της κάθε παραγγελίας.

6 Αποτελέσματα

6.1 Τα αποτελέσματα των πελατών με μοντέλο RFM

Πριν μετακινήσουμε σε τμήματα πελατών, Ας δούμε την εφαρμογή της αρχής Pareto - κοινώς αναφερόμενη ως κανόνας 80-20 στο σύνολο δεδομένων μας, εφαρμόζοντάς την στις μεταβλητές RFM.

Σύμφωνα με τον κανόνα του Pareto, το 80% των αποτελεσμάτων προέρχεται από το 20% των αιτιών.

Ομοίως, το 20% των πελατών συνεισφέρουν στο 80% των συνολικών σας εσόδων. Ας το επιβεβαιώσουμε γιατί αυτό θα μας βοηθήσει να γνωρίζουμε τους πελάτες στους οποίους θα πρέπει να επικεντρωθεί κατά την εμπορία νέων προϊόντων.

```
#get top 20% of the customers
top_20_cutoff = 3863 *20 /100
top_20_cutoff
```

```
772.6
```

```
#sum the monetary values over the customer with rank <=773
revenueByTop20 = customers_rank[customers_rank['Rank'] <= 772]['Monetary'].sum()
revenueByTop20
```

```
976683.3499999999
```

Στην περίπτωσή μας, το 80% των συνολικών εσόδων δεν επιτυγχάνεται από το 20% των πελατών της TOP, αλλά περίπου αυτό συμβαίνει, επειδή είναι λιγότερο από τους 20% TOP πελάτες που το επιτυγχάνουν. Θα ήταν ενδιαφέρον να μελετήσουμε αυτή την ομάδα πελατών επειδή είναι αυτοί που κάνουν τα περισσότερα μας έσοδα.

How many customers do we have in each segment?

```
print("Best Customers: ",len(rfm_segmentation[rfm_segmentation['RFMScore']=='444']))
print('Loyal Customers: ',len(rfm_segmentation[rfm_segmentation['F_Quartile']==4]))
print("Big Spenders: ",len(rfm_segmentation[rfm_segmentation['M_Quartile']==4]))
print('Almost Lost: ', len(rfm_segmentation[rfm_segmentation['RFMScore']=='244']))
print('Lost Customers: ',len(rfm_segmentation[rfm_segmentation['RFMScore']=='144']))
print('Lost Cheap Customers: ',len(rfm_segmentation[rfm_segmentation['RFMScore']=='111']))
```

```
Best Customers:  356
Loyal Customers:  752
Big Spenders:  966
Almost Lost:  64
Lost Customers:  9
Lost Cheap Customers:  353
```

Τώρα που γνωρίζαμε τα τμήματα των πελατών μας, μπορούμε να επιλέξουμε πώς να στοχεύουμε ή να αντιμετωπίζουμε κάθε τμήμα.

Για παράδειγμα:

Καλύτεροι πελάτες - Πρωταθλητές: Να τους ανταμείψετε. Μπορούν να υιοθετήσουν πρώιμα νέα προϊόντα. Προτείνετε τους "Ανατρέξτε σε έναν φίλο".

Σε κίνδυνο: Στείλτε τους εξατομικευμένα μηνύματα ηλεκτρονικού ταχυδρομείου για να τα ενθαρρύνετε να ψωνίζουν.

Πιστοί πελάτες: Επίσης χρειάζονται ανταμοιβή. Ίσως κάποια έκπτωση στις επόμενες αγορές.

6.2 Αποτελέσματα υλοποίησης του αλγόριθμου Kmeans

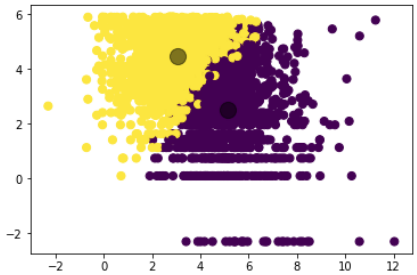
Πραγματοποιήθηκε προσπάθεια διαχωρισμού των πελατών σε 2-9 clusters με τις αντίστοιχες βαθμολογίες

```
For n_clusters = 2 The average silhouette_score is : 0.38940554101987457
For n_clusters = 3 The average silhouette_score is : 0.30380548886126
For n_clusters = 4 The average silhouette_score is : 0.31032684473888267
For n_clusters = 5 The average silhouette_score is : 0.2901550155654932
For n_clusters = 6 The average silhouette_score is : 0.29525762704591585
For n_clusters = 7 The average silhouette_score is : 0.29126695928086827
For n_clusters = 8 The average silhouette_score is : 0.29138117899994703
For n_clusters = 9 The average silhouette_score is : 0.29543652064694675
```

Στην οπτικοποίηση των αποτελεσμάτων φαίνεται για αριθμό clusters ίσον με 2 αφού από εκεί λάβαμε τη μεγαλύτερη βαθμολογία.

Visualize Clusters

```
#create a scatter plot
plt.scatter(matrix[:, 0], matrix[:, 1], c=clusters_customers, s=50, cmap='viridis')
#select cluster centers
centers = kmeans.cluster_centers_
plt.scatter(centers[:, 0], centers[:, 1], c='black', s=200, alpha=0.5);
```




```
# What's the number of customers in each cluster?
pd.DataFrame(pd.Series(clusters_customers).value_counts(), columns = ['NumberCustomers']).T
```

	1	0
NumberCustomers	2337	1526

Σημείωση (Πρόσθετη): Μπορούμε να ελέγξουμε τη διάμεση τιμή κάθε μεταβλητής (Συχνότητα, Νομισματική, Πρόσφατη) σε κάθε σύμπλεγμα, προκειμένου να κατανοήσουμε τι αντιπροσωπεύουν οι πελάτες από κάθε σύμπλεγμα.

6.3 Αποτελέσματα του Αλγορίθμου Apriori

```
basket_germany = create_basket("Germany")
basket2_sets = basket_germany.applymap(encode_units)
basket2_sets.drop('POSTAGE', inplace=True, axis=1)
```

```
frequent_itemsets_germany = apriori(basket2_sets, min_support=0.05, use_colnames=True)
```

```
rules = association_rules(frequent_itemsets_germany, metric="lift", min_threshold=1.2)
rules.head()
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(PLASTERS IN TIN WOODLAND ANIMALS)	(PLASTERS IN TIN CIRCUS PARADE)	0.137856	0.115974	0.067834	0.492063	4.242887	0.051846	1.740427
1	(PLASTERS IN TIN CIRCUS PARADE)	(PLASTERS IN TIN WOODLAND ANIMALS)	0.115974	0.137856	0.067834	0.584906	4.242887	0.051846	2.076984
2	(PLASTERS IN TIN CIRCUS PARADE)	(ROUND SNACK BOXES SET OF 4 FRUITS)	0.115974	0.157549	0.050328	0.433962	2.754455	0.032057	1.488330
3	(ROUND SNACK BOXES SET OF 4 FRUITS)	(PLASTERS IN TIN CIRCUS PARADE)	0.157549	0.115974	0.050328	0.319444	2.754455	0.032057	1.298977
4	(ROUND SNACK BOXES SET OF4 WOODLAND)	(PLASTERS IN TIN CIRCUS PARADE)	0.245077	0.115974	0.056893	0.232143	2.001685	0.028470	1.151290

Σύμφωνα με τα αποτελέσματα, ο πελάτης έχει 4 φορές περισσότερες πιθανότητες να αγοράσει PLASTERS IN TIN WOODLAND ANIMALS από έναν μέσο πελάτη (ανελκυστήρας) αν αγοράζει PLASTERS IN TIN CIRCUS PARADE. Αυτός ο κανόνας είναι "αληθής" σε 42% των περιπτώσεων (εμπιστοσύνη). Αυτό μπορεί να χρησιμοποιηθεί ως διορατικότητα για να συστήσετε PLASTERS IN TIN WOODLAND ANIMALS για τα ζώα που αγόρασαν το πράσινο.

Σε αυτό το σημείο μπορούμε να παρατηρήσουμε την πιθανότητα που υπάρχει να χρησιμοποιήσετε τη δημοτικότητα ενός προϊόντος για να οδηγήσετε τις πωλήσεις ενός άλλου.

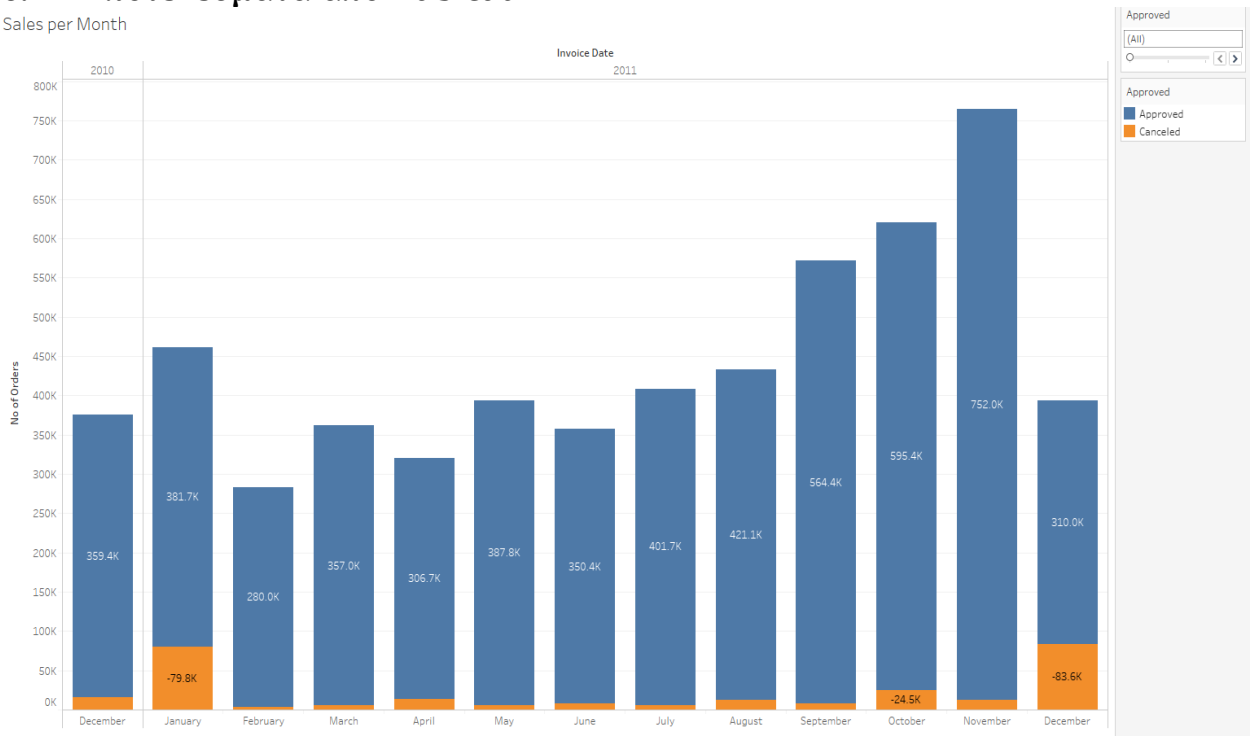
```
2 rules.head()
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift
0	(ALARM CLOCK BAKELIKE PINK)	(ALARM CLOCK BAKELIKE GREEN)	0.102041	0.096939	0.073980	0.725000	7.478947
1	(ALARM CLOCK BAKELIKE GREEN)	(ALARM CLOCK BAKELIKE PINK)	0.096939	0.102041	0.073980	0.763158	7.478947
2	(ALARM CLOCK BAKELIKE GREEN)	(ALARM CLOCK BAKELIKE RED)	0.096939	0.094388	0.079082	0.815789	8.642959
3	(ALARM CLOCK BAKELIKE RED)	(ALARM CLOCK BAKELIKE GREEN)	0.094388	0.096939	0.079082	0.837838	8.642959
4	(ALARM CLOCK BAKELIKE PINK)	(ALARM CLOCK BAKELIKE RED)	0.102041	0.094388	0.073980	0.725000	7.681081

Σύμφωνα με τα αποτελέσματα, ο πελάτης έχει 7 φορές περισσότερες πιθανότητες να αγοράσει ένα ροζ ξυπνητήρι από το μπαλκάλι από έναν μέσο πελάτη (ανελκυστήρας) εάν αγοράσει το ALARM CLOCK BAKELIKE GREEN. Αυτός ο κανόνας είναι "αληθής" στο 77% των περιπτώσεων (εμπιστοσύνη). Αυτό μπορεί να χρησιμοποιηθεί ως διορατικότητα για να συστήσετε ρολόι συναγερμού BAKELIKE PINK για αυτούς που αγόρασαν το πράσινο.

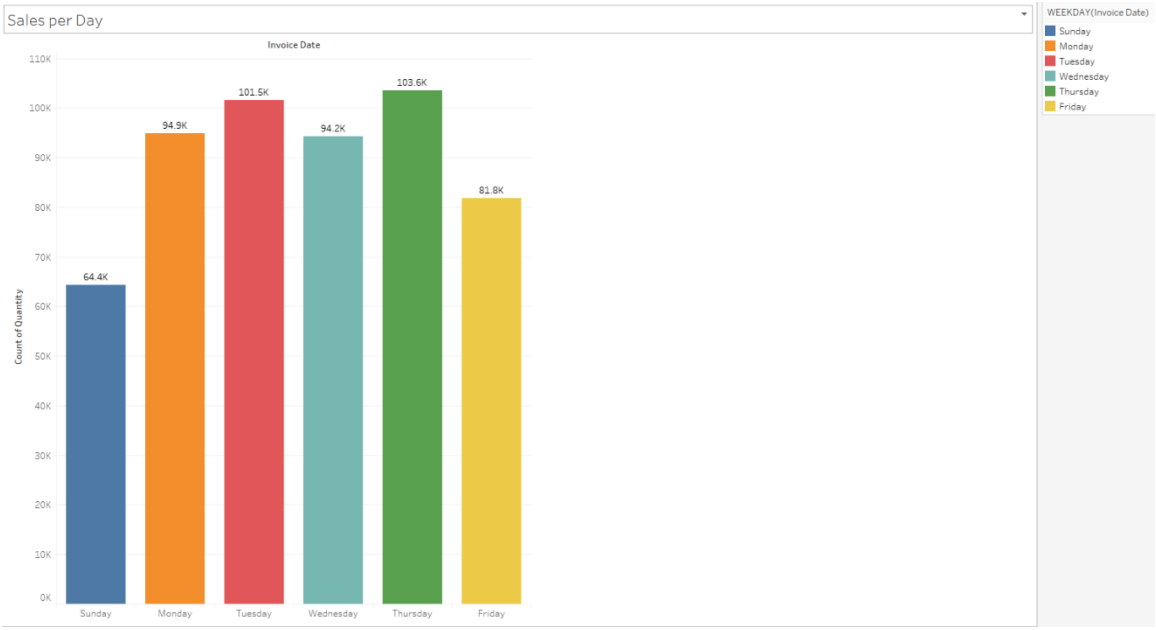
Οι άνθρωποι που αγοράζουν πράσινο ρολόγια συνήθως αγοράζουν ροζ ρολόγια πάρα πολύ. Οι ομάδες μάρκετινγκ στα καταστήματα λιανικής πώλησης θα πρέπει να απευθύνονται σε πελάτες που αγοράζουν πράσινο ρολόγια και ροζ ρολόγια και να τους προσφέρουν μια προσφορά για να αγοράσουν ένα τρίτο στοιχείο, όπως οι λάμπες.

6.4 Αποτελέσματα από Tableau

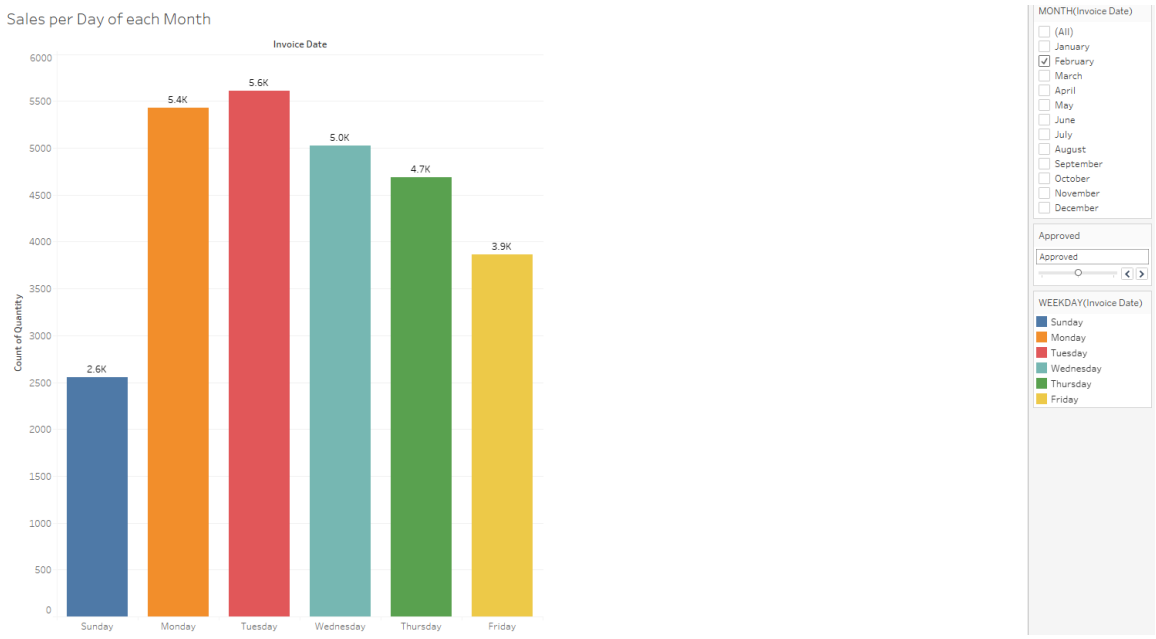


Εικόνα 1

Στο παραπάνω διάγραμμα (1) απεικονίζονται τα συνολικα ποσά που δαπανήθηκαν για όλες τις χώρες κατα τη διαρκεια των 13 μηνών. Με μπλε χρώμα φαίνονται οι παραγγελίες που τελικά έγιναν δεκτές ενώ με το πορτοκαλί οι αντίστοιχες που ακυρώθηκαν. Παρατηρείται ότι οι περισσότερες παραγγελίες σημειώθηκαν το Νοέμβριο, ενώ οι λιγότερες τον Φεβρουάριο.

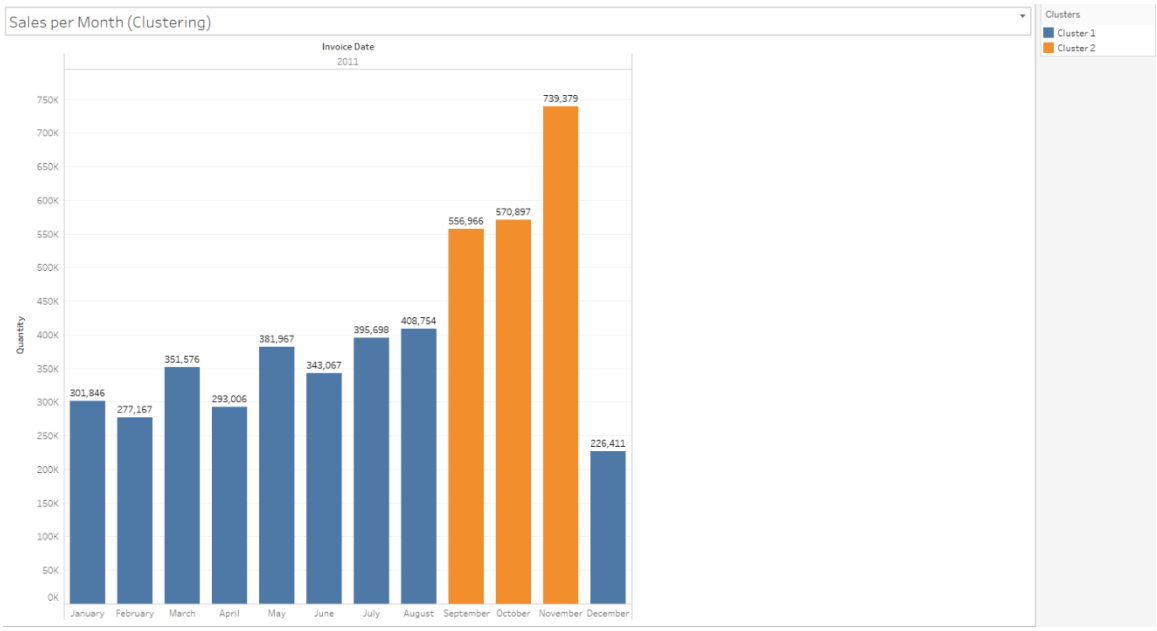


Εικόνα 2



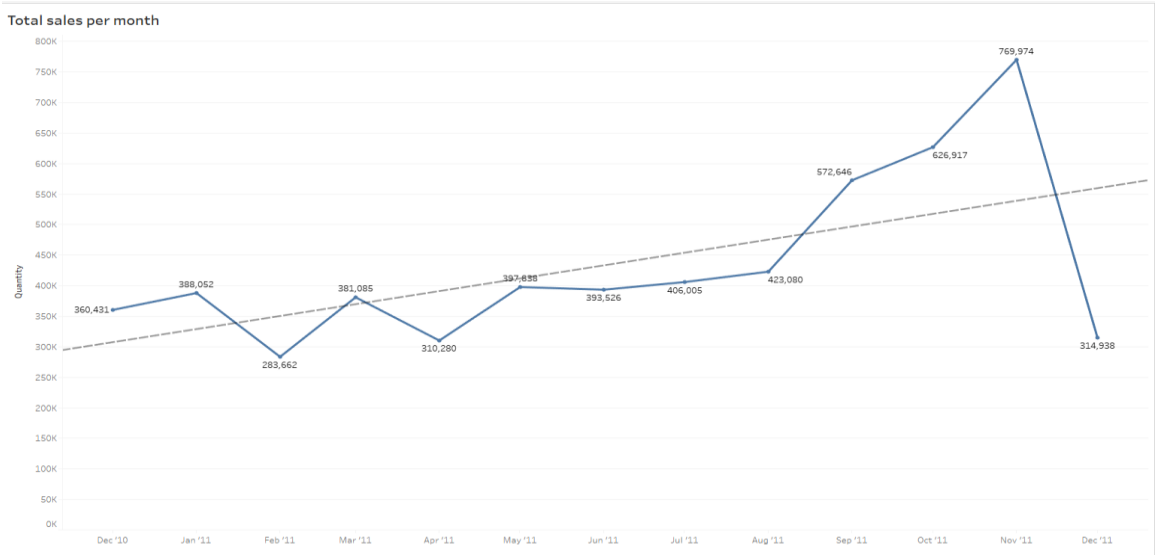
Εικόνα 3

Στα παραπάνω διαγράμματα (2,3) φαίνονται οι παραγγελίες που έγιναν ανα κάθε μέρα της εβδομάδας. Στο πρώτο φαίνονται για όλους τους μήνες οπου παρατηρείται πως οι περισσότερες έγιναν την μέρα Πέμπτη ενώ οι λιγότερες την Κυριακή. Στο δεύτερο απεικονίζεται η ίδια πληροφορία αλλά για τον μήνα Φεβρουάριο. (τυχαία επιλογή). Εδώ είναι αντιληπτό πως παρόλο που συνολικά η μέρα Πέμπτη είναι εκείνη που πραγματοποιήθηκαν οι περισσότερες παραγγελίες στο σύνολο, το μήνα Φεβρουάριο κατατάσσεται τέταρτη σε σειρά, ενώ πρώτη είναι η μέρα Τρίτη.



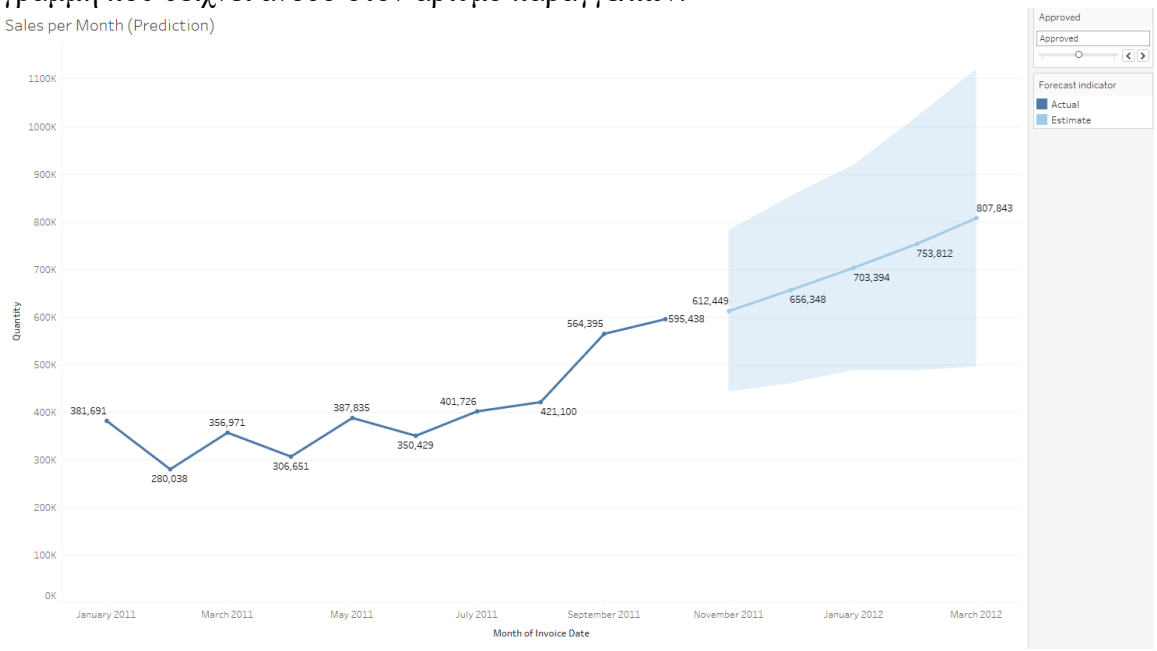
Εικόνα 4

Στο παραπάνω διάγραμμα (4) υλοποιήθηκε ομαδοποίηση με βάση τον αριθμό παραγγελιών που φαίνονται στο πρώτο διάγραμμα και ο διαχωρισμός έγινε σε 2 ομάδες. Με το πορτοκαλί χρώμα είναι οι μήνες Σεπτέμβρης, Οκτώβρης και Νοέμβρης όπου όπως φαίνεται έχουν μεγάλη διαφορά από τους υπόλοιπους μήνες που απεικονίζονται με μπλε χρώμα.



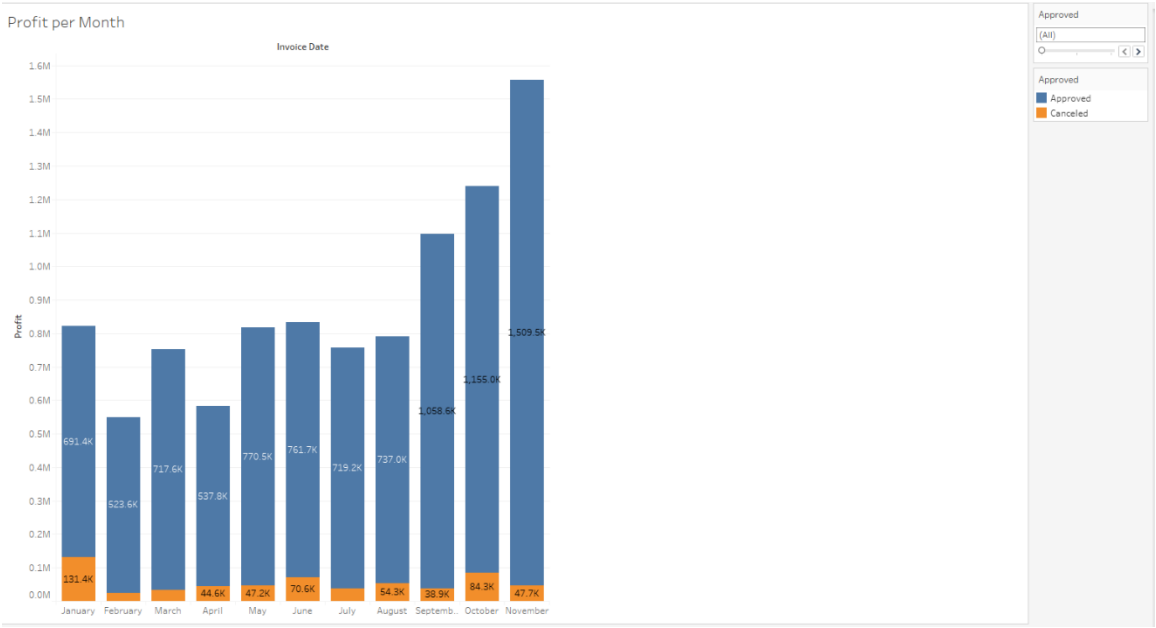
Εικόνα 5

Στο παραπάνω διάγραμμα (5) φαίνεται ο αριθμός παραγγελιών ανα μήνα καθώς και η γραμμή που δείχνει άνοδο στον αριθμό παραγγελιών.

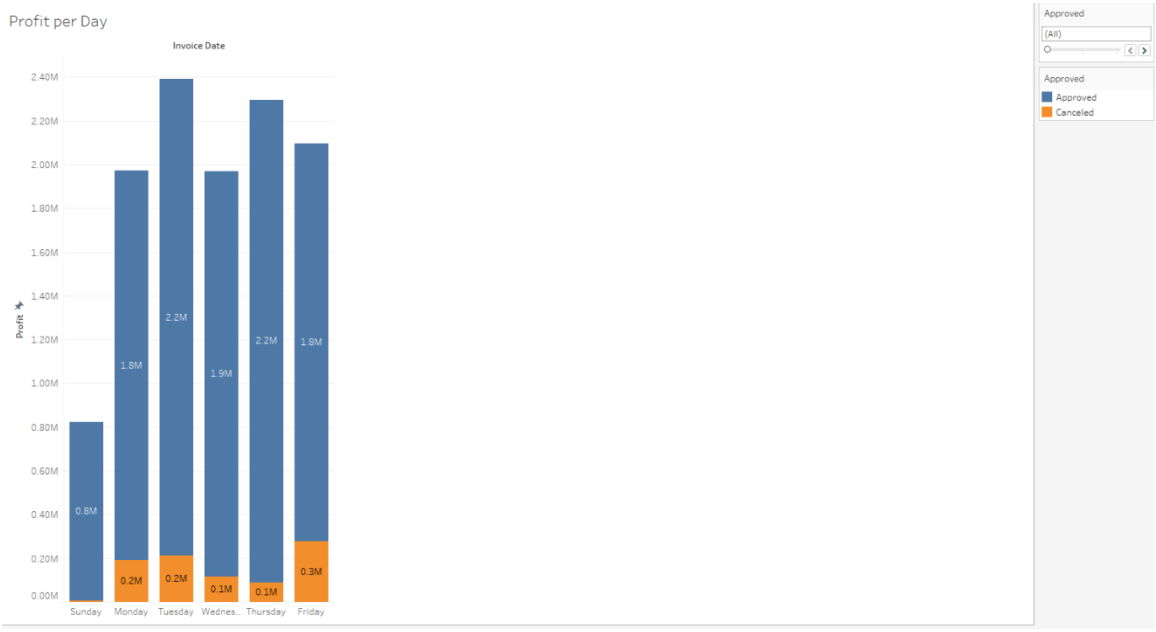


Εικόνα 6

Στο παραπάνω διάγραμμα (6) φαίνεται μια πρόβλεψη για τους τρεις επόμενους μήνες. Να σημειωθεί πως επειδή οι συνολικές παραγγελίες για το μήνα Δεκέμβρη του 2011 ήταν πολυ λιγότερες απο τους προηγούμενους μήνες, δεν ληφθηκε υπόψιν για το συμπέρασμα της πρόβλεψης. Η πρόβλεψη δείχνει πως ο αριθμός παραγγελιών θα αυξηθεί τους επόμενους μήνες, κατι που μπορεί να γίνει αντιληπτο και απο το προηγούμενο διάγραμμα που έδειχνε μια σταθερή άνοδο.



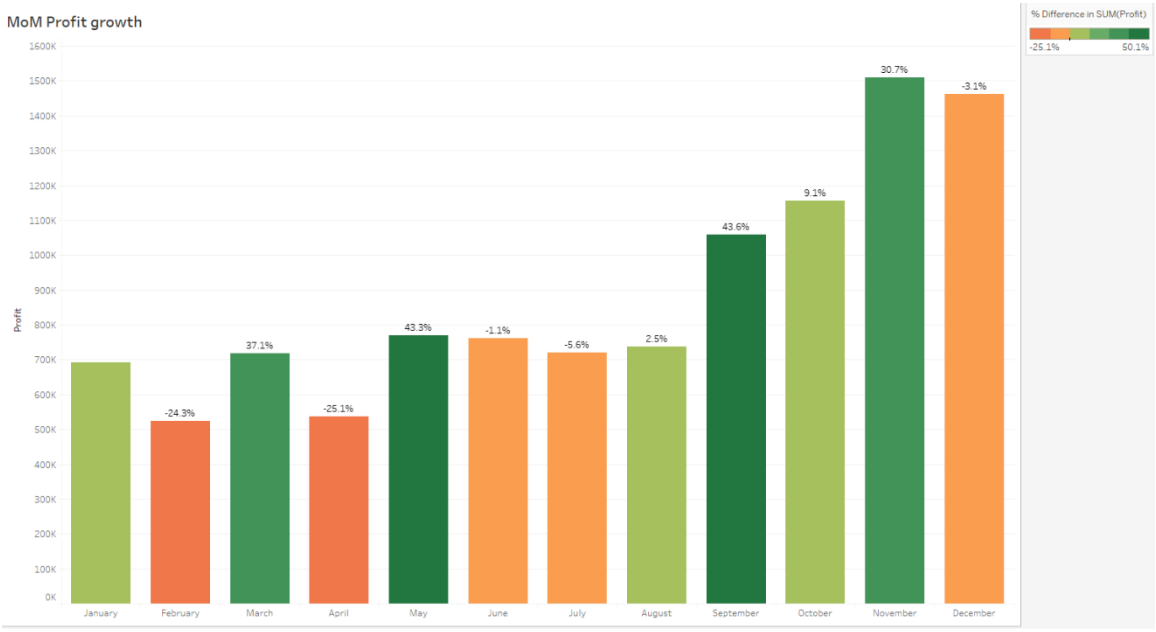
Εικόνα 7



Εικόνα 8

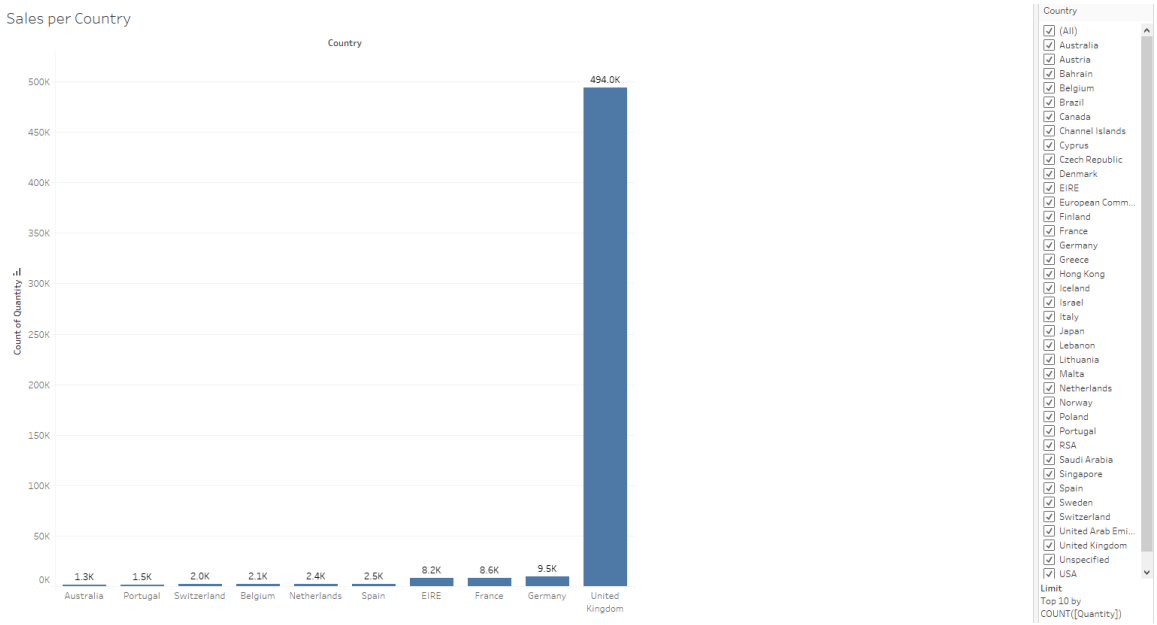
Στα παραπάνω διαγράμματα (7,8) αφού δημιουργήθηκε ένα καινούργιο πεδίο που ονομάστηκε Profit (Κέρδος) απεκονίζεται το κέρδος ανα κάθε μήνα στο πρώτο και ανα κάθε μέρα στο δεύτερο. Να σημειωθεί πως με μπλε χρώμα φαίνεται το κέρδος από τις παραγγελίες και με πορτοκαλί το ποσό (loss) που χάθηκε από τις παραγγελίες που ακθρώθηκαν. Παρατηρείται πως το μήνα Νοέμβρη παρουσιάζεται το μεγαλύτερο κέρδος, ενώ το μήνα Ιανουάριο η μεγαλύτερη απώλεια.

Παράλληλα για το δεύτερο διάγραμμα φαίνεται πως τη μέρα Πέμπτη που υπήρξαν οι περισσότερες παραγγελίες όπως φάνηκε και παραπάνω υπάρχει και το μεγαλύτερο κέρδος ενώ οι περσσότερες απώλειες απο τις ακυρώσεις παραγγελιών σημειώθηκαν τη μέρα Παρασκευή

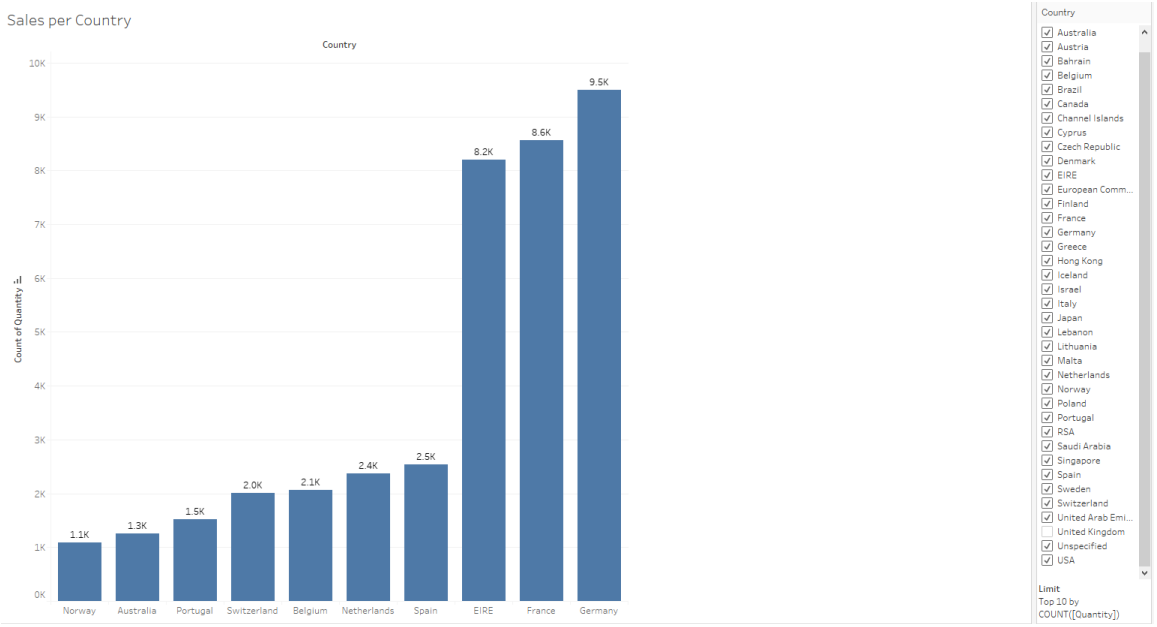


Εικόνα 9

Στο παραπάνω διάγραμμα (9) φαίνεται η διαφορά σε ποσοστό στο καθαρό κέρδος από μήνα σε μήνα. Είναι ένα από τα πιο ενδιαφέροντα γραφήματα καθώς βλέπουμε τις εναλλαγές στον ογκο των παραγγελιών και παρουσιάζεται το ποσοστό αύξησης και πτώσης σε σχέση με τον προηγούμενο μήνα. Παρατηρείται πως το μήνα Μάρτιο έχουμε τη μεγαλύτερη αύξηση με 37.1% σε σχέση με τον προηγούμενο μήνα ενώ η μεγαλύτερη μείωση παρατηρείται ακριβώς τον επόμενο μήνα Απρίλιο με πτώση στο 25.1%

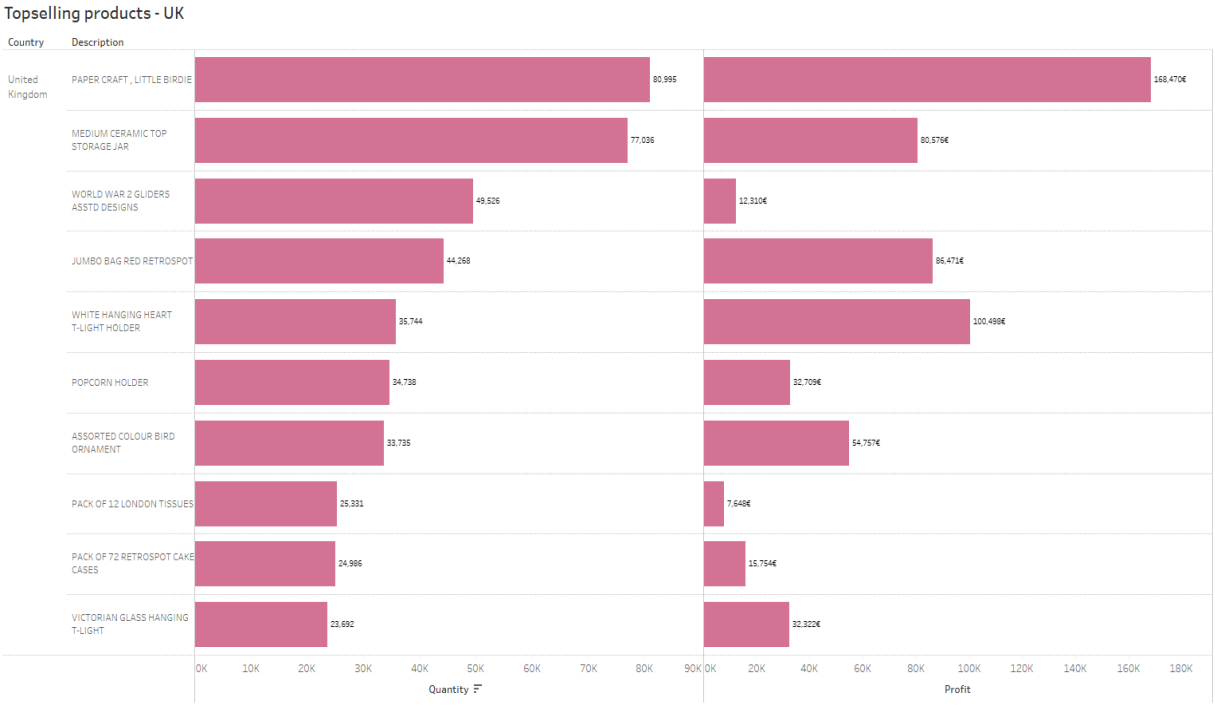


Εικόνα 10

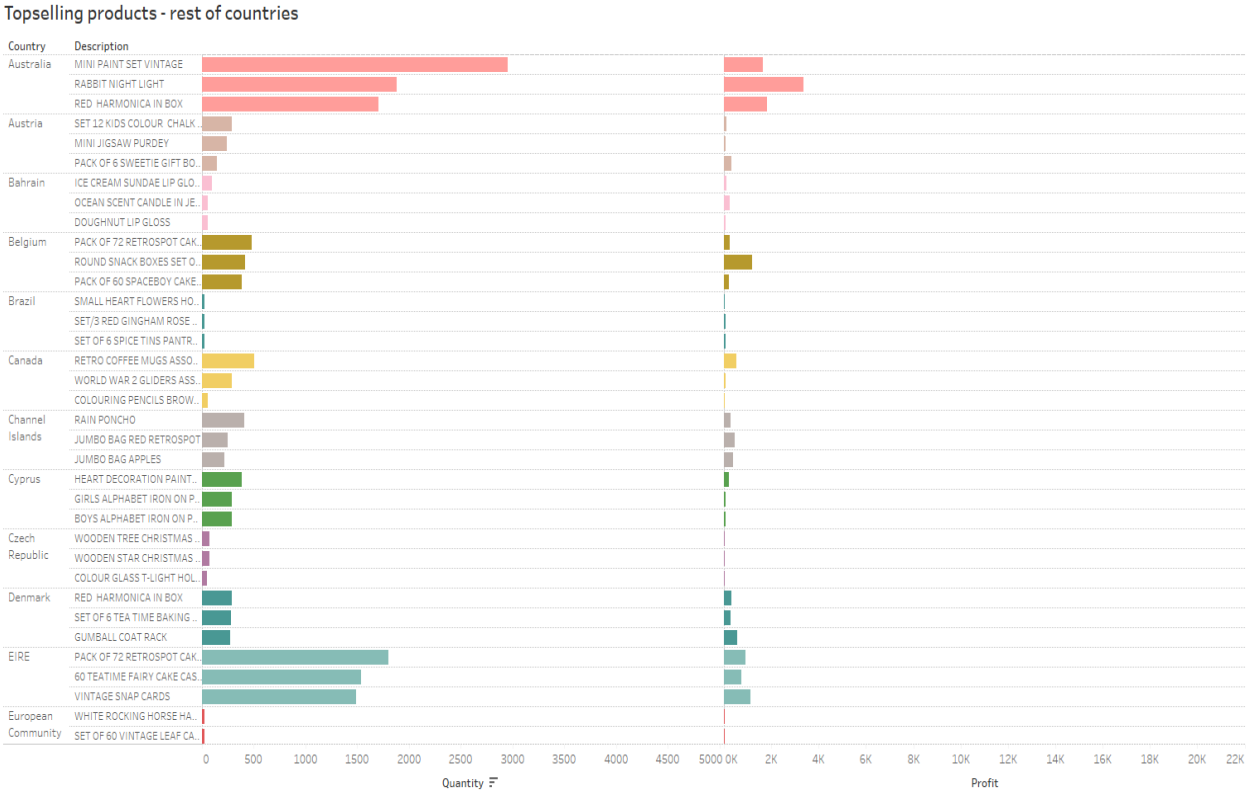


Εικόνα 11

Στα παραπάνω διαγράμματα (10,11) απεικονίζονται ο συνολικός αριθμός παραγγελιών ανα χώρα. Το σύνολο των δεδομένωμ που χρησιμοποιήθηκε αφορούσε κυρίως το Ηνωμένο Βασίλειο, όποτε ο αριθμός είναι πολύ μεγαλύτερος από τις υπόλοιπες χώρες-κράτη, όπως φαίνεται στο πρώτο διάγραμμα. Για το λόγο αυτό στο δεύτερο δεν υπλογίζεται το Ηνωμένο Βασίλειο. Παρατηρείται πώς πρώτη είναι η Γερμανία πολύ κοντα με τη Γαλλία ενώ τη δεκάδα κλείνει η Νορβηγία.



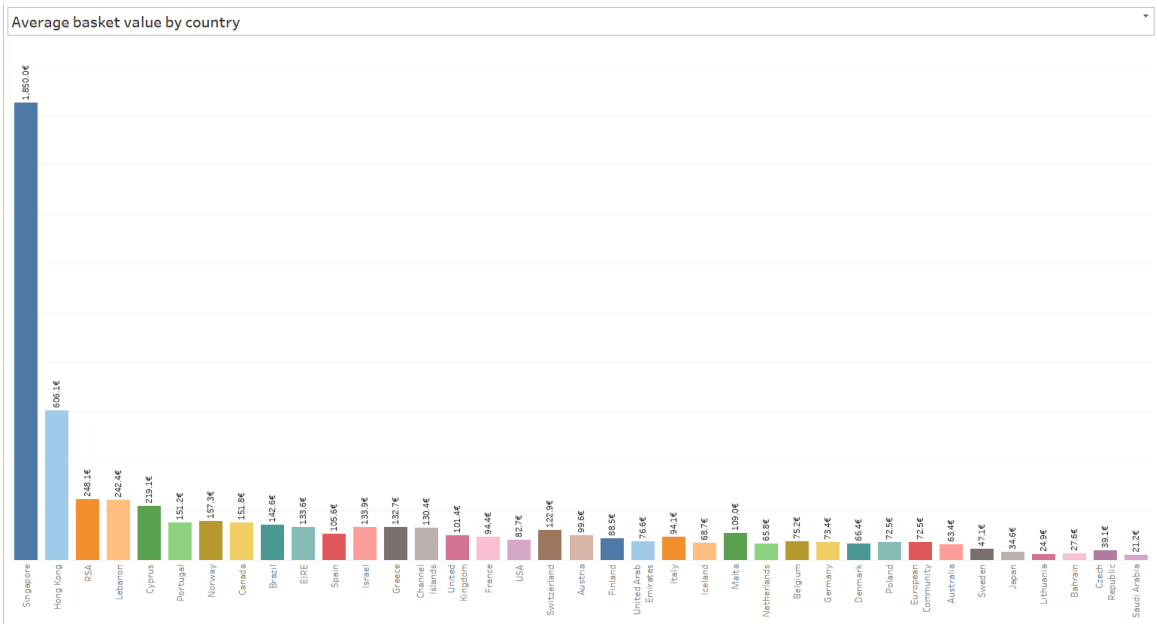
Εικόνα 12



Εικόνα 13

Όπως και προηγουμένως διότι ο αριθμός παραγγελιών στο Ηνωμένο Βασίλειο είναι πολύ μεγαλύτερος από τα υπόλοιπα κράτη, έγινε διαχωρισμός ανάμεσα τους. Στο πρώτο (12) απεικονείζονται τα 10 κορυφαία προιόντα σε αριθμό παραγγελιών και στο συνολικό κέρδος στο Ηνωμένο Βασίλειο όπου φαίνεται πως το PAPER CRAFT, LITTLE BIRD είναι το προιον με τις περισσότερες παραγγελίες αλλά και το μεγαλύτερο κέρδος.

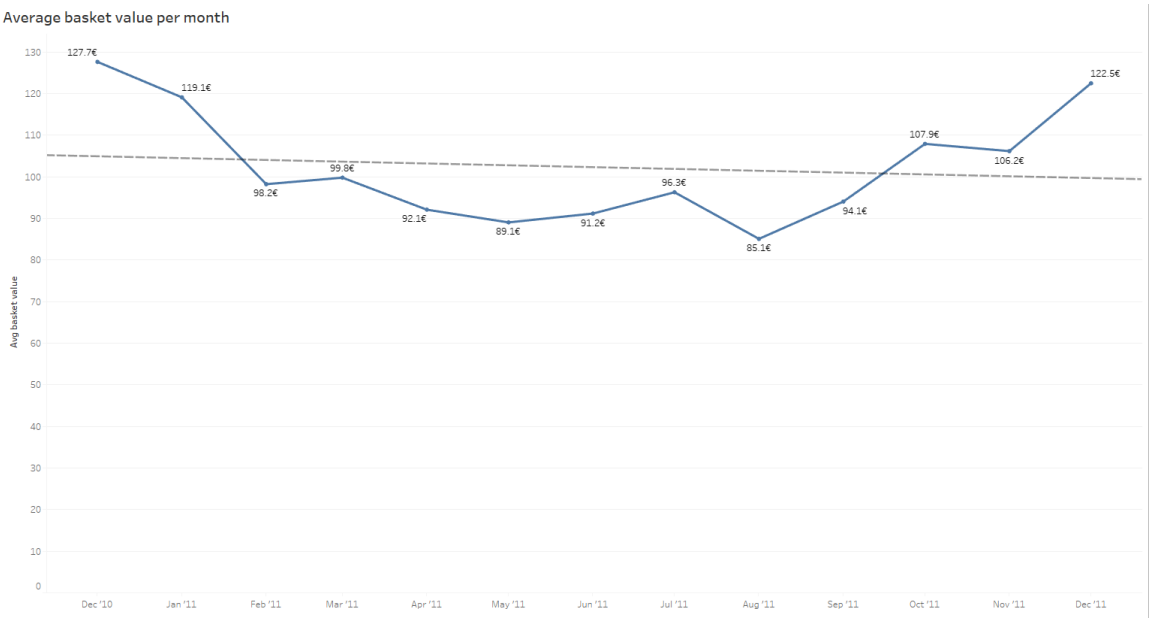
Στο δεύτερο (13) φαίνεται για τις υπόλοιπες χώρες. Να σημειώθει πώς απεικονίζεται ένα μέρος μοόνο του διαγράμματος διότι δεν μπορούσε να απεικονιστεί ολόκληρο



Εικόνα 14

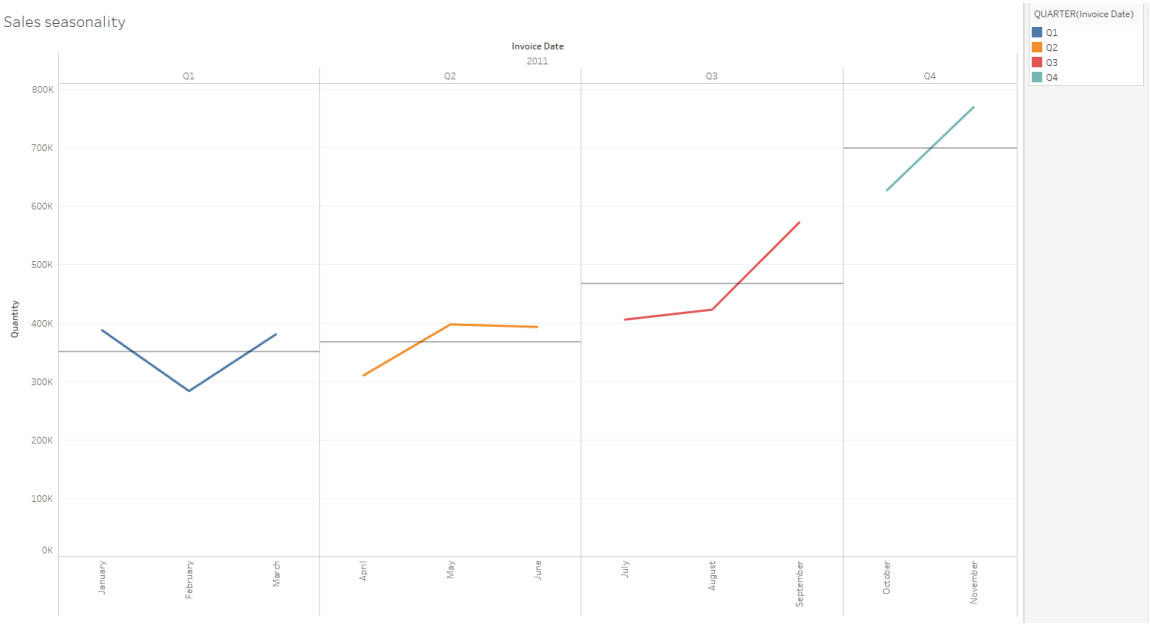
Στο παραπάνω διάγραμμα (14) φαίνεται το μέσο ποσό που δαπανάται απο τους καταναλωτές κάθε χώρας. Ιδιαίτερο ενδιαφέρον παρουσιάζει το γεγονός πως πρώτη χώρα είναιη Σιγκαπούρη όπου το μέσο καλάθι ενός καταναλωτή κοστίζει 1850 ευρώ,

ποσό πολύ υψηλότερο από το δεύτερο που ανήκει στο Χονγκ Κονγκ με 606 ευρώ. Το μέσο ποσό των καταναλωτών του Ηνωμένου Βασιλείου ανέρχεται στα 101 ευρώ και παρατηρείται πως είναι κάτω απο την πρώτη δεκάδα.



Εικόνα 15

Στο παραπάνω διάγραμμα (15) φαίνεται το μέσο κόστος ανά καλάθι καταναλωτή ανα κάθε μήνα. Παρατηρείται πως τους χειμερινούς μήνες έχουμε ποσά που ξεπερνάνε τα 100 ευρω αν καλαθι, ενω τους υπόλοιπους μήνες είναι πολύ κοντά σε αυτό το ποσό.



Εικόνα 16

Στο παραπάνω διάγραμμα (16) απεικονίζεται η εποχικότητα (Seasonality) για τον αριθμό παραγγελιών στα τέσσερα τρίμηνα. Να σημειωθεί πως από τοτελευταίο αφαιρέθηκε ο Δεκέμβρης. Η γκρι γραμμή απεικονίζει τους μέσους όρους και φαίνεται και η αύξηση στον αριθμό που συζητήθηκε σε προηγούμενο διάγραμμα.

7 Συμπεράσματα

7.1 Προοπτική μετά την εφαρμογή της ομαδοποίησης k-means:

Δυστυχώς, δεν επιτύχαμε σαφώς διαχωρισμένες συστάδες. Οι αναθέσεις συμπλεγμάτων είναι μπερδεμένες. (Μπορεί να οφείλεται σε υπερβάσεις που δεν έχουν αφαιρεθεί).

Περιορισμοί της ομαδοποίησης k-mean:

• Δεν υπάρχει καμία εγγύηση ότι θα οδηγήσει στην καλύτερη παγκόσμια λύση.

• Δεν μπορεί να ασχοληθεί με διαφορετικά σχήματα (όχι κυκλικά) και να εξετάσει την πιθανότητα ενός μέλους να ανήκει σε περισσότερα από ένα σύμπλεγμα.

Αυτά τα μειονεκτήματα του k-mean σημαίνουν ότι για πολλά σύνολα δεδομένων (ειδικά χαμηλής διάστασης σύνολα δεδομένων) μπορεί να μην λειτουργεί τόσο καλά όπως ίσως να ελπίζουμε. Εδώ έρχεται το Guassian Mixture Model (GMM) σε βοήθεια παρέχοντας μεγαλύτερη ευελιξία εξαιτίας των συστάδων που έχουν απεριόριστες covariances και επιτρέποντας πιθανοτική ανάθεση συμπλέγματος.

Περαιτέρω Επέκατση: Μια κοινή πρακτική πριν από την ομαδοποίηση: Βασική Ανάλυση Συστατικών (PCA). Το PCA υπολογίζει τις διαστάσεις που μεγιστοποιούν καλύτερα τη διακύμανση. Δίνει οδηγίες σχετικά με τον αριθμό των στοιχείων που πρέπει να ληφθούν υπόψη για το GMM. Βασικά, μειώνει τη διαστασιολογία ενώ διατηρεί τα σημαντικότερα χαρακτηριστικά, τα χαρακτηριστικά (οι συνδυασμοί χαρακτηριστικών περιγράφουν καλύτερα τους πελάτες). Όμως, καθώς δεν ασχολούμαστε με υψηλή διάσταση, δεν θα το κάνουμε για αυτή την περίπτωση.

7.2 Συμπεράσματα οπτικοποίησης μέσω του Tableau

Από την ανάλυση που πραγματοποιήθηκε βγήκαν ενδιαφέροντα αποτελέσματα όσον αναφορά τα κέρδη της εταιρείας, ποιους μήνες και μέρες πραγματοποίησε μεγαλύτερο τζίρο, από ποιες χώρες προέρχονται τα περισσότερα έσοδα.

Όλα τα παραπάνω σε συνδυασμό με το γεγονός ότι η πρόβλεψη έδειξε άνοδο στον αριθμό των παραγγελιών μπορεί να βοηθήσει την εταιρεία να καταστρώσει μια στρατηγική για να αυξήσει ακόμη περισσότερο τους αριθμούς αυτούς.

Βιβλιογραφία

A. Shollo, K. Kautz, Towards an understanding of business intelligence, In the Proceedings of the 21st Australian Conference in Information Systems (ACIS), Brisbane, 2010.

Amazon Web Services, "AWS Lambda," Available: https://aws.amazon. com/lambda/, 2020.

Amazon Web Services, "AWS Lambda@Edge," Available: http://docs. aws.amazon.com/lambda/latest/dg/lambda-edge.html, 2017.

Apache Project. Openwhisk, 2020.

Apex, "Apex: Serverless Architecture," Available: http://apex.run/, 2017.

AWS Greengrass, Available: https://aws.amazon.com/ greengrass/, 2017.

AWS Step Functions, Available: https://aws.amazon.com/ step-functions/, 2017.

Baldini, I et al. (10 June, 2017). Serverless Computing: Current Trends and Open Problems.

Beaver, D., Kumar, S., Li, H. C., Sobel, J., & Vajgel, P. (2010, October). Finding a Needle in Haystack: Facebook's Photo Storage. In *OSDI* (Vol. 10, No. 2010, pp. 1-8).

C. Lowery, "Emerging Technology Analysis: Serverless Computing and Function Platform as a Service," Gartner, Tech. Rep., September 2016.

Cevher, V., Becker, S., & Schmidt, M. (2014). Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics. IEEE Signal Processing Magazine, 31(5), 32-43.

Chen and Zhang, "Data-intensive applications, challenges, techniques and technologies : a survey on big data", vol. 275, 2014, Information Sciences, pp. 314–347

Chen, H., Chiang, R. H., & Storey, V. C. (2012). Business intelligence and analytics: From big data to big impact. *MIS quarterly*, 1165-1188.

Cui, Y. (October 25, 2017). Many-faced threats to Serverless security. https://hackernoon.com/many-faced-threatsto-serverless-security-519e94d19dba hackernoon.com/many-faced-threats-toserverless-security-519e94d19dba

Cukier, K. (2010). *Data, data everywhere: A special report on managing information*. Economist Newspaper.

D. Arnott, G. Pervan, A critical analysis of decision support systems research, Journal of Information Technology, 20 (2005) 67-87.

D. Rotolo and Leydesdorff, “Matching medline / pubmed data with web of science: a routine in r language”, vol. 66, no. 10, 2015, Journal of the Association for Information Science and Technology, pp. 2155–2159.

Deardorff, A. (2016). Tableau (version. 9.1). Journal of the Medical Library Association, 104(2), 182-183.

E. Hammond, “Lambdash: Run sh commands inside AWS Lambda environment,” Available: https://github.com/alestic/lambdash, 2017.

E. Jonas, “Microservices and Teraflops,” Available: http://ericjonas.com/ pywren.html, 2016. (13) E. d. Lara, C. S. Gomes, S. Langridge, S. H. Mortazavi, and M. Roodi, “Poster abstract: Hierarchical serverless computing for the mobile edge,” in 2016 IEEE/ACM Symposium on Edge Computing (SEC), Oct 2016, pp. 109–110.

G. McGrath, J. Short, S. Ennis, B. Judson, and P. Brenner, “Cloud event programming paradigms: Applications and analysis,” in 2016 IEEE 9th International Conference on Cloud Computing (CLOUD), June 2016, pp. 400–406.

Google, “Google Cloud Functions,” Available: https://cloud.google.com/ functions/, 2020.

I. Baldini, P. Castro, P. Cheng, S. Fink, V. Ishakian, N. Mitchell, V. Muthusamy, R. Rabbah, and P. Suter, “Cloud-native, event-based programming for mobile applications,” in Proceedings of the International Conference on Mobile Software Engineering and Systems, ser. MOBILESoft ’16. New York, NY, USA: ACM, 2016, pp. 287–288.

I. Dwyer, “Serverless computing: Developer empowerment reaches new heights,” Available: http://cdn2.hubspot.net/hubfs/553779/PDFs/ Whitepaper Serverless Screen Final V2.pdf, 2016.

Iosup, A., Ostermann, S., Yigitbasi, N., Prodan, R., Fahringer, T., Epema, D.: Performance Analysis of Cloud Computing Services for Many-Tasks Scientific Computing. IEEE Transactions on Parallel and Distributed Systems 22(6), 931–945 (jun 2011)

Iron.io, “Iron.io IronFunctions,” Available: http://open.iron.io/, 2020.

J. S. Hammond, J. R. Rymer, C. Mines, R. Heffner, D. Bartoletti, C. Tajima, and R. Birrell, “How To Capture The Benefits Of Microservice Design,” Forrester Research, Tech. Rep., May 2016.

Jones, R. (December 28, 2016). Gone in 60 milliseconds Offensive security in the serverless age (Rich Jones). https://www.youtube.com/watch?v=byJBR16xUnc

K. uwe Schmidt, D. Anicic, and R. Sthmer, “Event-driven reactivity: A survey and requirements analysis,” in In 3rd International Workshop on Semantic Business Process Management, 2008, pp. 72–86.

Kelleher, J. D., Mac Namee, B., & D'arcy, A. (2015). Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies. MIT press.

Kwon, O., Lee, N., & Shin, B. (2014). Data quality management, data usage experience and acquisition intention of big data analytics. *International journal of information management*, *34*(3), 387-394.

Labrinidis, A., & Jagadish, H. V. (2012). Challenges and opportunities with big data. *Proceedings of the VLDB Endowment*, *5*(12), 2032-2033.

Laney, D. (2001). 3-d data management: Controlling data volume, velocity and variety,” META Group,” Research Note, February 2001.

Leitner, P., Cito, J.: Patterns in the chaos - A study of performance variation and predictability in public iaas clouds. ACM Trans. Internet Techn. 16(3), 15:1–15:23 (2016), http://doi.acm.org/10.1145/2885497

Leitner, P., Scheuner, J.: Bursting with possibilities - an empirical study of creditbased bursting cloud instance types. In: 8th IEEE/ACM International Conference on Utility and Cloud Computing, UCC 2015, Limassol, Cyprus, December 7-10, 2015. pp. 227–236 (2015), http://doi.ieeecomputersociety.org/10.1109/UCC.2015.39

Lightbend. Akka Framework.

M. Bidan, F. Rowe, D. Truex, An empirical study of IS architectures in French SMEs: integration approaches, European Journal of Information Systems, 21 (2012).

M. Gibson, D. Arnott, I. Jagielska, Evaluating the intangible benefits of business intelligence: Review & research agenda, Proceedings of the 2004 IFIP International Conference on Decision Support Systems (DSS2004): Decision Support in an Uncertain and Complex World, 2004, pp. 295- 305.

M. Villamizar, O. Garcs, L. Ochoa, H. Castro, L. Salamanca, M. Verano, R. Casallas, S. Gil, C. Valencia, A. Zambrano, and M. Lang, “Infrastructure cost comparison of running web applications in the cloud using aws lambda and monolithic and microservice architectures,” in 2016 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), May 2016, pp. 179–182.

M. Yan, P. Castro, P. Cheng, and V. Ishakian, “Building a chatbot with serverless computing,” in Proceedings of the 1st International Workshop on Mashups of Things and APIs, ser. MOTA ’16. New York, NY, USA: ACM, 2016, pp. 5:1–5:4.

Malawski, M., Kuzniar, M., Wojcik, P., Bubak, M.: How to Use Google App Engine for Free Computing. IEEE Internet Computing 17(1), 50–59 (Jan 2013)

McGrath, M.G., Short, J., Ennis, S., Judson, B., Brenner, P.R.: Cloud event programming paradigms: Applications and analysis. In: 9th IEEE International Conference on Cloud Computing, CLOUD 2016, San Francisco, CA, USA, June 27- July 2, 2016. pp. 400–406. IEEE Computer Society (2016)

Microsoft, “Azure Functions,” Available: https://azure.microsoft.com/ en-us/services/functions/, 2020.

Mills, S., Lucas, S., Irakliotis, L., Rappa, M., Carlson, T., & Perlowitz, B. (2012). Demystifying big data: a practical guide to transforming the business of government. *TechAmerica Foundation, Washington*.

Muncaster, P. (November 20, 2017). US Army Exposes Terabytes of Surveillance Data. https://www.infosecurity-magazine.com/news/us-army-exposesterabytes/

Oakes, L Yang, K Houck, T Harter, A C Arpaci-Dusseau, and R H ArpaciDusseau. Pipsqueak: Lean Lambdas with Large Libraries. In 2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW), pages 395–400, jun 2017.

OpenLambda, “OpenLambda,” Available: https://open-lambda.org/, 2020.

P.B. Seddon, D. Constantinidis, H. Dod, How Does Business Analytics Contribute to Business Value?, Proceedings of the International Conference on Information Systems Orlando, USA, 2012.

Pedregosa, G. Varoquax, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and M. Brucher, “Scikit-learn: machine learning in python”, vol. 12, 2011, Journal of Machine Learning Research, pp. 2825–2830.

Pirtle, J. (July 25, 2017). Security Best Practices for Serverless Applications.

Podjarny, G. (May 17, 2017). Serverless Security: What’s Left to Protect - Guy Podjarny. https://www.youtube.com/watch?v=CiyUD_rI8D8

Prodan, R., Sperk, M., Ostermann, S.: Evaluating High-Performance Computing on Google App Engine. IEEE Software 29(2), 52–58 (Mar 2012)

R. Agarwal, V. Dhar, Editorial—big data, data science, and analytics: The opportunity and challenge for IS research, Information Systems Research, 25 (2014) 443-448.

R. Sabherwal, I. Becerra-Fernandez, Business intelligence: Practices, Technologies, and Management, John Wiley & Sons, NJ, 2011.

R. Sharma, S. Mithas, A. Kankanhalli, Transforming decision-making processes: a research agenda for understanding the impact of business analytics on organisations, European Journal of Information Systems, 23 (2014) 433-441.

R. Vojta, "AWS journey: API Gateway & Lambda & VPC performance," Available: https://robertvojta.com/ aws-journey-api-gateway-lambda-vpc-performance-452c6932093b, 2016.

S. Hendrickson, S. Sturdevant, T. Harter, V. Venkataramani, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau, "Serverless computation with openlambda," in Proceedings of the 8th USENIX Conference on Hot Topics in Cloud Computing, ser. HotCloud'16. Berkeley, CA, USA: USENIX Association, 2016, pp. 33–39.

Sandvik, Rura. (December 12, 2017). Keynote- Building a culture of security at The New York Times - Appsec 2017. https://www.youtube.com/watch?v=_iCLs4jw_yo

Schroeck, M., Shockley, R., Smart, J., Romero-Morales, D., & Tufano, P. (2012). Analytics: The real-world use of big data. *IBM Global Business Services*, *12*(2012), 1-20.

Seals, T. (November 28, 2017). Pentagon Exposes Top Secret Classified Info to Public Internet. https://www.infosecurity-magazine.com/news/pentagon-exposes-topsecret/

Segal, O. (January 17, 2018). The First "Serverless Architectures Security Top 10" Guide Released. https://www.puresec.io/blog/serverless-top-10-released

Sheridan, K.(January 17, 2018). Where to Find Security Holes in Serverless Architecture. https://www.darkreading.com/cloud/where-to-find-security-holes-inserverless-architecture/d/d-id/1330842

Sparta, "Sparta: A Go framework for AWS Lambda microservices," Available: http://gosparta.io/, 2017.

Spillner, J.: Snafu: Function-as-a-service (faas) runtime design and implementation. CoRR abs/1703.07562 (2017), http://arxiv.org/abs/1703.07562

The Apache Software Foundation, "Apache OpenWhisk," Available: http://openwhisk.org/, 2020.

Villamizar, M., Garces, O., Ochoa, L., Castro, H., Salamanca, L., Verano, M., Casallas, R., Gil, S., Valencia, C., Zambrano, A., Lang, M.: Infrastructure cost comparison of running web applications in the cloud using aws lambda and monolithic and microservice architectures. In: 2016 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid). pp. 179–182 (May 2016)

W. McKinney, "Python for data analysis",1st ed., 2013, O'Reilly Media Inc., pp.453.

Wagner and A. Sood, "Economics of Resilient Cloud Services," ArXiv e-prints, Jul. 2016.

Wagner, B., Sood, A.: Economics of Resilient Cloud Services. In: 1st IEEE International Workshop on Cyber Resilience Economics (Aug 2016), http://arxiv.org/abs/1607.08508

Warzon, "AWS Lambda pricing in context: A comparison to EC2," Available: https://www.trek10.com/blog/lambda-cost/, 2016.

Z. Jourdan, R.K. Rainer, T.E. Marshall, Business Intelligence: An Analysis of the Literature 1, Information Systems Management, 25 (2008) 121-131.