$2^{ο}$ Πρότζεκτ στο μάθημα "Ανάκτηση κ' Εξόρυξη Πληροφοριών"

– Εξόρυξη Συχνών Προτύπων με **WEKA** –

παραδοτέο από

Κουνάδη Βασιλική (Α.Μ.: 2022202000102)
Μαζαράκης Ιωάννης (Α.Μ.: 2022202000130)

11 Φεβρουαρίου 2024

Επιβλέπων: Τρυφωνόπουλος Χρήστος

Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Σχολή Οικονομίας και Τεχνολογίας
Πανεπιστήμιο Πελοποννήσου

Περιεχόμενα

1 Οδηγίες εγκατάστασης

Οι παρακάτω οδηγίες στοχεύουν στην εγκατάσταση και τη διαμόρφωση του περιβάλλοντος ανάπτυξης για την Java, χρησιμοποιώντας το Visual Studio Code (VSCode) και το εργαλείο διαχείρισης εξαρτήσεων Maven σε Windows. Ακολουθήστε τα παρακάτω βήματα για να διαμορφώσετε το περιβάλλον ανάπτυξής σας. Οι οδηγίες προέρχονται από τη διάλεξη του κυρίου Νίκου Πλατή πάνω στην JavaFX [1].

Προσοχή για Χρήστες Linux, MacOS:

Για εγκατάσταση σε Linux, ακολουθήστε τις κατάλληλες οδηγίες για τη διανομή λογισμικού σας, καθώς τα προγράμματα μπορούν να εγκατασταθούν από τα αποθετήρια λογισμικού. Σιγουρευτείτε ότι εγκαθιστάτε τις σωστές εκδόσεις που αναφέρονται στις παρακάτω οδηγίες.
Για εγκατάσταση σε MacOS, κατεβάστε τα προγράμματα από τους αντίστοιχους δικτυακούς τόπους. Ενδεχομένως οι οδηγίες για Windows να σας βοηθήσουν σε κάποια σημεία.

Βήμα 1: Λήψη και Εγκατάσταση του Java JDK 19

Κατεβάστε το πακέτο εγκατάστασης του Java JDK 19 από τη διεύθυνση: https://www.oracle.com/java/technologies/javase-downloads.html Προτιμήστε αυτήν την έκδοση για τη χρήση μιας κοινής έκδοσης. Ελέγξτε ότι έχετε εγκαταστήσει την JDK 11 ή νεότερη. Μπορείτε να διατηρήσετε προηγούμενες εκδόσεις της Java, απλά εκτελέστε το πρόγραμμα εγκατάστασης.

Βήμα 2: Λήψη και Εγκατάσταση του Apache Maven

Κατεβάστε το αρχείο zip του Apache Maven από τη διεύθυνση: https://maven.apache.org/download.cgi Αποσυμπιέστε το αρχείο σε έναν προτιμώμενο φάκελο (π.χ., C:\Program Files). Προσθέστε τον φάκελο "bin" αυτού του φακέλου στο Path των Windows (π.χ., C:\Program Files\apache-maven-3.8.4\bin). Για οδηγίες, ακολουθήστε τον σύνδεσμο: https://www.architectryan.com/2018/03/17/add-to-the-path-on-windows-10/ Σε ένα τερματικό, εκτελέστε την εντολή ‘mvn -v‘. Θα πρέπει να εμφανίζει τις εκδόσεις της Java και του Maven. Επανεκκινήστε εάν χρειάζεται.

Βήμα 3: Εγκατάσταση του Visual Studio Code (VSCode)

Κατεβάστε και εγκαταστήστε το Visual Studio Code από τη διεύθυνση: https://code.visualstudio.com/ Στο παράθυρο εγκατάστασης "Select Additional Tasks", επιλέξτε όλα τα checkboxes. Για βασική χρήση του VSCode, ακολουθήστε τις οδηγίες στον σύνδεσμο: https://code.visualstudio.com/docs/getstarted/introvideos Εγκαταστήστε τις παρακάτω επεκτάσεις (Extensions):

- Java Extension Pack
- Visual Studio IntelliCode

Για υποστήριξη της Java στο VSCode, διαβάστε τις σελίδες από τον σύνδεσμο: https://code.visualstudio.com/docs/java/java-tutorial, ειδικά τις σχετικές με τα "Build Tools" και το Maven (κυρίως τις παραγράφους "Exploring Maven project", "Execute Maven commands and goals").

2 Εγχειρίδιο Χρήσης (User Manual) Διεπαφής

2.1 Εισαγωγή

Προκειμένου να εμφανίσουμε τα αποτελέσματα του πρώτου ερωτήματος με πιο αποτελεσματικό και ευέλικτο τρόπο αποφασίσαμε να τα αναπτύξουμε σε μιας διεπαφή σε JavaFX με χρήση Maven. Παρακάτω είναι το εγχειρίδιο για τον χειρισμό της διεπαφής που εξηγεί στα βασικά χαρακτηριστικά και τις λειτουργίες της.

2.2 Πώς να Ξεκινήσετε

Κατά την εκκίνηση της εφαρμογής, βλέπουμε ένα μενού με διάφορα κουμπιά το οποίο μαζί με τις διαφορετικές επιλογές που μπορούμε να επεξεργαστούμε. Σε κάθε περίπτωση για να πιστέψετε στο μενού με τις επιλογές πατήστε το κουμπί "Main Menu". Το πρόγραμμα πιθανώς να καθυστερήσει ελαφρώς μετά την επιλογή κάποιου κουμπιού (περίπου 5 δευτερόλεπτα), καθώς επεξεργάζεται τα δεδομένα και φτιάχνει τα αρχεία (με τα αποτελέσματα αναλυτικά γραμμένα) και τα διαγράμματα.

2.3 Επιλογές για Arff file

- **Arff file format** Με το πάτημα του κουμπιού "Create Arff file format" το πρόγραμμα δημιουργεί το αρχείο csv (με τα περιεχόμενα κάθε καλαθιού), το

οποίο μετέπειτα θα μετατρέψουμε σε .arff και θα επεξεργαστούμε εντός του WEKA.

- **Category Arff file format** Με το πάτημα του κουμπιού "Create Category Arff file format" το πρόγραμμα δημιουργεί το αρχείο csv (με τα περιεχόμενα κάθε καλαθιού, οργανωμένα ανά κατηγορίες προϊόντων), το οποίο μετέπειτα θα μετατρέψουμε σε .arff και θα επεξεργαστούμε εντός του WEKA.

- **Ages Arff file format** Με το πάτημα του κουμπιού "Create Ages Arff file format" το πρόγραμμα δημιουργεί τα 3 αρχεία csv (ένα αρχείο για κάθε ηλιακή ομάδα, με τα περιεχόμενα κάθε καλαθιού ενός πελάτη της αντίστοιχης ηλικίας, οργανωμένα ανά κατηγορίες προϊόντων), το οποίο μετέπειτα θα μετατρέψουμε σε .arff και θα επεξεργαστούμε εντός του WEKA.

2.4 Δημοφιλή προϊόντα

Η παραπάνω επιλογή "Show Top Products" εμφανίζει το διάγραμμα δημοφιλίας των προϊόντων με φθίνουσα κατάταξη. Για την καλύτερη προσαρμογή του διαγράμματος σε κάθε είδους οθόνη εμφανίζονται από default 10 μόνο προϊόντα (στην αρχή εμφανίζονται τα πρώτα 10 δημοφιλέστερα προϊόντα).

- Με το πάτημα του κουμπιού "See Next Chart" μπορούμε να προχωρήσουμε στο διάγραμμα με τα επόμενα 10 προϊόντα με μικρότερη δημοφιλία.

- Με το πάτημα του κουμπιού "See Previous Chart" μπορούμε να πάμε πίσω στο διάγραμμα με τα προηγούμενα 10 προϊόντα με μεγαλύτερη δημοφιλία.

- Εάν θέλουμε να αλλάξουμε το πλήθος των προϊόντων του εμφανίζονται στο διάγραμμα, καθορίζουμε την επιθυμητή τιμή στο textfield και η πατάμε enter ή το κουμπί "Change num of products".

Σημείωση: Όταν αλλάζουμε το πλήθος των προϊόντων του εμφανίζονται στο διάγραμμα, συνήθως χρειάζεται να αλλάξουμε ελαφρώς το μέγεθος του παραθύρου, για τη σωστή προσαρμογή και στοίχιση του διαγράμματος.

2.5 Συχνοί πελάτες

Η παραπάνω επιλογή "Show Top Customers" εμφανίζει το διάγραμμα με το πλήθος των προϊόντων που αγόρασαν οι πελάτες στο κατάστημα με φθίνουσα κατάταξη εμφανίσεων. Η λογική λειτουργίας της επιλογής είναι η ίδια με της επιλογής "Show Top Products" (δείτε 2.4).

2.6 Δημοφιλή προϊόντα Χριστούγεννα/Πάσχα/Καλοκαίρι

Οι επιλογές "Show Top Products on Christmas", "Show Top Products on Easter", "Show Top Products during Summer" εμφανίζουν τα διάγραμμα δημοφιλίας των προϊόντων με φθίνουσα κατάταξη , για τις προκαθορισμένες χρονικές περιόδους των Χριστουγέννων, Πάσχα και καλοκαιριού αντίστοιχα. Η λογική λειτουργίας των επιλογών είναι η ίδια με της επιλογής "Show Top Products" (δείτε 2.4), με κύρια διαφορά ότι τα αρχικά διαγράμματα αναφέρονται στα Χριστούγεννα/Πάσχα/Καλοκαίρι του έτους 2022. Για να πλοηγηθείτε στα στατιστικά των εορτών ανά τα έτη επιλέγετε τα κουμπιά "See next Christmas/Easter/summer" και "See previous Christmas/Easter/summer" (το αντίστοιχο για κάθε περίοδο).

2.7 Συναλλαγές ανά εβδομάδα/μήνα/έτος

Οι επιλογές "Show Transactions per Week", "Show Transactions per Month", "Show Transactions per Year" εμφανίζουν τα διάγραμμα των αριθμών συναλλαγών στο κατάστημα ανάεβδομάδα, μήνα και έτος αντίστοιχα.

- Η επιλογή "Show Transactions per Week" δείχνει τον αριθμό συναλλαγών κάθε μέρα μιας εβδομάδας, και μπορούμε να πλοηθούμε στις εβδομάδες με τα κουμπιά "See next Week chart" και "See previous Week chart".

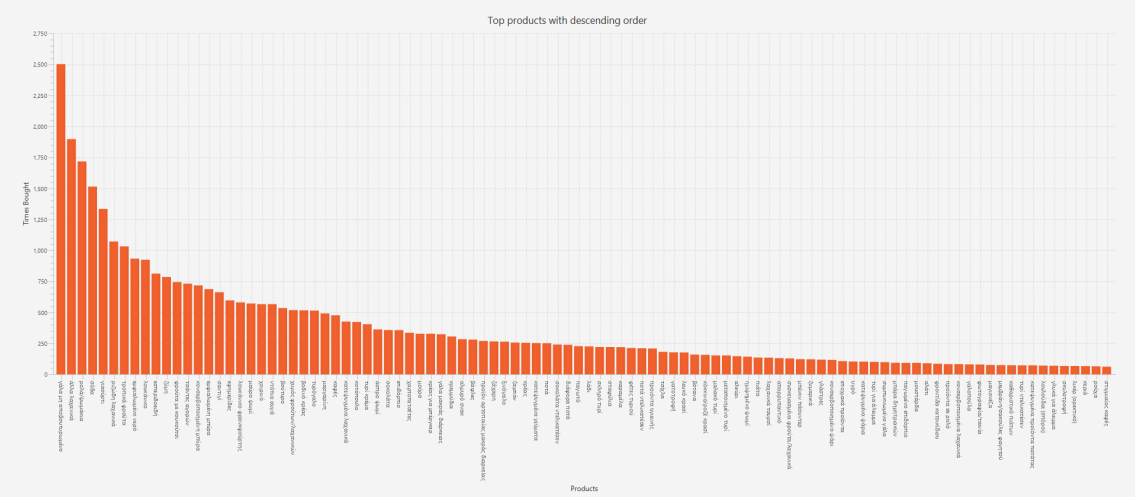
- Η επιλογή "Show Transactions per Month" δείχνει τον αριθμό συναλλαγών κάθε μέρα ενός μήνα, και μπορούμε να πλοηθούμε στους μήνες με τα κουμπιά "See next Month chart" και "See previous Month chart".

- Η επιλογή "Show Transactions per Year" δείχνει τον αριθμό συναλλαγών κάθε μήνα ενός ετός, και μπορούμε να πλοηθούμε στις εβδομάδες με τα κουμπιά "See next Year chart" και "See previous Year chart".

3 Ανάλυση δεδομένων

3.1 Δημοφιλή προϊόντα

Από προγραμματιστικής πλευράς (Controller2), για να καταλήξουμε στα παρακάτω αποτελέσματα [1] διαβάζουμε το αρχείο CSV με τις συναλλαγές, βρίσκουμε τα διαφορετικά προϊόντα και μετράμε τη συχνότητα εμφάνισης κάθε προϊόντος. Χρησιμοποιούμε ένα HashMap για να αποθηκεύει προϊόντα ως κλειδιά και την αντίστοιχη εμφάνιση τους ως value. Έπειτα βάζουμε τα δεδομένα σε ArrayList (ένα ArrayList για το αρχείο, και ένα άλλο όπου τα προϊόντα γίνονται decode σε UTF-8 για χρήση στην FXML) για να τα ταξινομήσουμε με βάση τις εμφανίσεις τους, και συμπληρώνουμε το RandomAccessFile με τα προϊόντα και τις εμφανίσεις τους. Τέλος, δημιουργούμε ένα barchart από την JavaFX, εμφανίζοντας τα προϊόντα στον άξονα x και τις αντίστοιχες εμφανίσεις τους στον άξονα y. Τα αποτελέσματα βρίσκονται στο αρχειο "productsFiles.csv", που παράγεται αυτόματα με το πάτημα του αντίστοιχου κουμπιού.



Εικόνα 1: Δημοφιλία προϊόντων

Από τα παραπάνω δεδομένα μπορούμε να κάνουμε τις εξής παρατηρήσεις: Πρώτον, τα βασικά είδη όπως το γιαούρτι, η σόδα, τα ρολά/ψωμάκια, άλλα λαχανικά και το μη αποβουτυρωμένο γάλα είναι σταθερά δημοφιλή, γεγονός που υποδεικνύει ότι αυτά τα είδη πιθανότατα αποτελούν τις πιο απαραίτητες καθημερινές ανάγκες ενός νοικοκυριού.

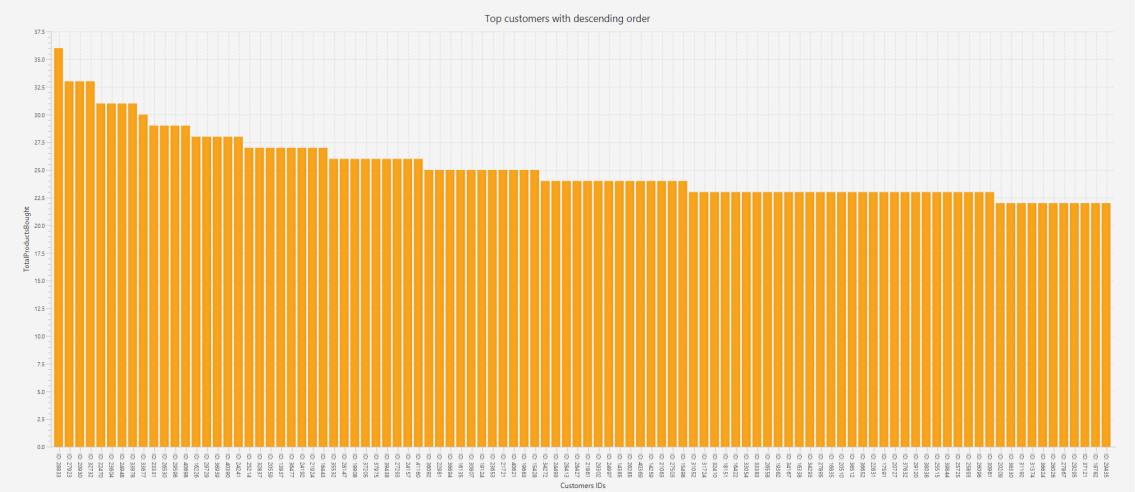
[1] Παραθέτουμε στις φωτογραφίες διαγραμμάτων, μόνο τα πρώτα 100 προϊόντα ενδεικτικά, για να εξασφαλίσουμε την καλύτερη σαφήνεια των αποτελεσμάτων για τις οθόνες που είχαμε διαθέσιμες. Όσο μεγαλύτερη είναι η οθόνη που χρησιμοποιείται, τόσο καλύτερα εμφανίζει όλα τα 166 προϊόντα η διεπαφή.

Δεύτερον, η έμφαση στα φρέσκα προϊόντα, συμπεριλαμβανομένων των κρεμμυδιών, των λαχανικών, των εσπεριδοειδών, των τροπικών και άλλων φρούτων, υποδηλώνει μια πιθανώς υψηλή ποιότητας φρούτων και λαχανικών τοπικής παραγωγής στο mini market (καθώς είναι σύνηθες για ένα μικρό mini market να προμηθεύεται φρούτα και λαχανικά από τοπικούς παραγωγούς σε καλές τιμές, με υψηλή ποιότητα προϊόντος). Επιπλέον, η ζήτηση σε προϊόντα όπως κατεψυγμένα γεύματα, κονσέρβες, κατεψυγμένα λαχανικά, κατεψυγμένο κοτόπουλο και κατεψυγμένα φρούτα τονίζει τη σημαντική ανάγκη για γρήγορες και εύκολες λύσεις γευμάτων μεταξύ των πελατών, καθώς ο πολυάσχολος τρόπος ζωής και οι χρονικοί περιορισμοί οδηγούν τους πελάτες να αναζητούν που απαιτούν ελάχιστη προετοιμασία. Τέλος, οι συνεχείς πωλήσεις αλκοολούχων ποτών, συμπεριλαμβανομένης της εμφιαλωμένης μπύρας, του ουίσκι, του κρασιού prosecco, του ρούμι και των διαφόρων κρασιών, υποδηλώνουν την ανάγκη αγοράς αλκοολούχων ποτών, πιθανόν για εορταστικές ή κοινωνικές εκδηλώσεις (με μεγάλη πιθανότητα να είναι αγορές για συγκεντρώσεις της τελευταίας στιγμής, όπου οι μεγάλες αλυσίδες σουπερμάρκετ δεν μπορούν να εξυπηρετήσουν λόγω ωραρίων).

3.2 Συχνοί πελάτες

Από προγραμματιστικής πλευράς (Controller3), για να καταλήξουμε στα παρακάτω αποτελέσματα διαβάζουμε το αρχείο CSV με τις συναλλαγές, βρίσκουμε τους διαφορετικούς πελάτες και τις διαφορετικές ημερομηνίες αγορών τους, μαζί με τα προϊόντα που αγόρασαν σε αυτή την επίσκεψη[2]. Χρησιμοποιούμε ένα TreeMap της μορφής Map¡key custID, value Map¡key date, value basket¿¿, για να αποθηκεύει τα δεδομένα αγορών μας. Έπειτα βάζουμε τα δεδομένα σε ArrayList για να τα ταξινομήσουμε με βάση τα πόσα προϊόντα έχει αγοράσει συνολικά ο κάθε πελάτης (αθροίζοντας τα μήκη των value του εμφωλευμένου Map), και συμπληρώνουμε το RandomAccessFile με τους πελάτες και το πλήθος των προϊόντων που έχουν αγοράσει. Τέλος, δημιουργούμε ένα barchart από την JavaFX, εμφανίζοντας τους πελάτες στον άξονα x και το πλήθος των προϊόντων που έχουν αγοράσει στον άξονα y. Τα αποτελέσματα βρίσκονται στο αρχειο "customersBuyMost.csv", που παράγεται αυτόματα με το πάτημα του αντίστοιχου κουμπιού.

[2]Το Controller3 παράγει και το "basketFile.csv" το οποίο αποτελέι βοηθητικό αρχείο ελέγχων για τα arff format files στα άλλα κουμπιά.



Εικόνα 2: Επισκέψεις πελατών

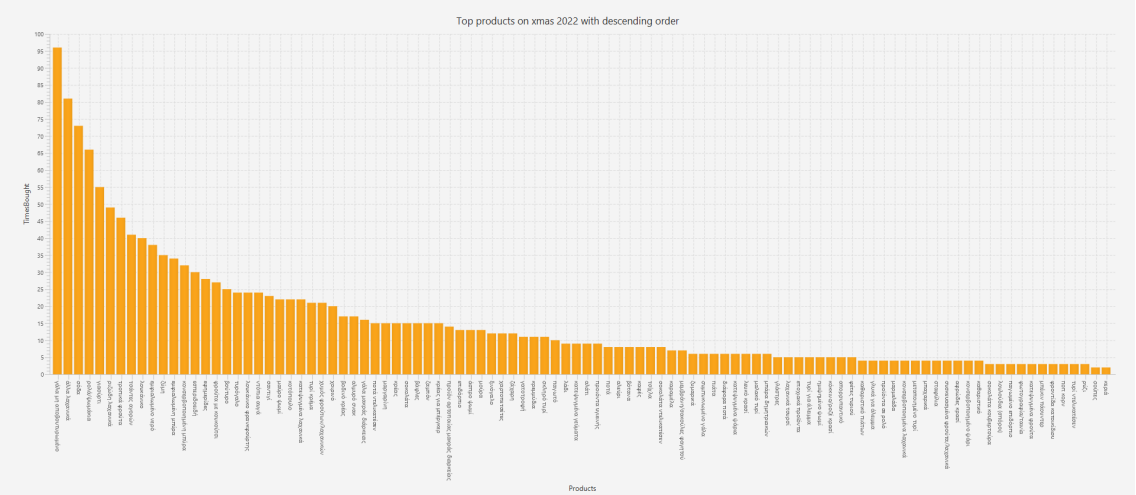
Από τα παραπάνω δεδομένα δεν είναι εύκολο να βγάλουμε συμπεράσματα εάν δεν έχουμε περαιτέρω πληροφορίες για το ποιοί είναι αυτοί οι πελάτες ή το που μένουν. Οι καλύτεροι πελάτες του mini market έχουν ψωνήσει από εκεί περίπου 25 με 30 προϊόντα συνολικά εντός 2 ετών, το οποίο δεν είναι πολλές αγορές γενικότερα. Συνήθως κάποιος πελάτης φαίνεται να εχει ψωνησει απο το mini market 2 προϊόντα συνολικά. Αυτά τα στατιστικά, καθώς και ο μεγάλος αριθμός (3898) διαφορετικών πελατών, μας δείχνει πως γενικότερα το mini market είναι μάλλον μία λύση ανάγκης για τους περισσότερους, χωρίς να εξαρτάται κάποιος πελάτης σταθερά από αυτό.

3.3 Δημοφιλή προϊόντα ανά χρονική περίοδο

Από προγραμματιστικής πλευράς (Controller7, Controller8, Controller9), για να καταλήξουμε στα παρακάτω αποτελέσματα διαβάζουμε το αρχείο CSV με τις συναλλαγές, βρίσκουμε τις διαφορετικές ημερομηνίες αγορών και τα διαφορετικά προϊόντα σε αυτές, μαζί με το πόσες φορές αγοράστηκαν. Χρησιμοποιούμε ένα TreeMap της μορφής Map¡key date, value Map¡key product, value timesBought¿¿, για να αποθηκεύει τα δεδομένα αγορών μας. Έπειτα δημιουργούμε ένα ArrayList στο οποίο αποθηκεύουμε ένα HashMap για κάθε διαφορετικό έτος δεδομένων μας (2022, 2023), της μορφής Map¡key product, value timesBought¿. Στη συνέχεια διατρέχουμε το TreeMap με όλες τις ημερομηνίες αγορών και ελέγχουμε αν η κάθε συναλλαγή εντάσσεται στις εορτινές ημερομηνίες, αν ναι τότε γεμίζουμε το σωστό HashMap με τα δεδομένα του εμφωλευμένου Map. Τέλος, βάζουμε τα δεδομένα σε ArrayList (όπου τα προϊόντα γίνονται decode σε UTF-8 για χρήση στην FXML) για να τα ταξινομήσουμε με βάση

τις εμφανίσεις των διαφορετικών προϊόντων. Τέλος, δημιουργούμε ένα barchart από την JavaFX, εμφανίζοντας τα προϊόντα στον άξονα x και τις αντίστοιχες εμφανίσεις τους στον άξονα y.

3.3.1 Δημοφιλή προϊόντα τα Χριστούγεννα

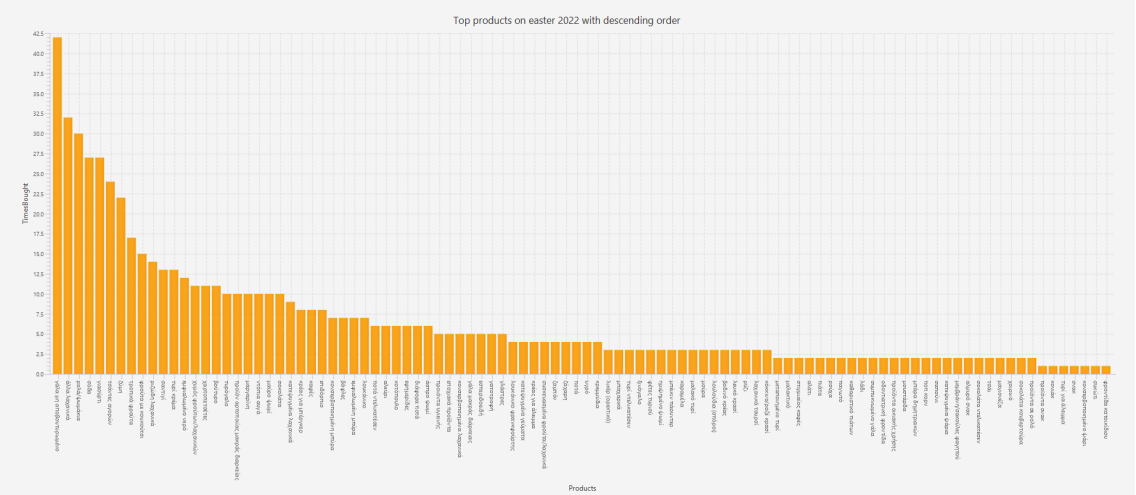


Εικόνα 3: Δημοφιλία προϊόντων τα Χριστούγεννα

Από τα παραπάνω δεδομένα βλέπουμε ότι κατά τη περίοδο των Χριστουγέννων και της Πρωτοχρονιάς, η αγοραστική συμπεριφορά των καταναλωτών επηρεάζεται ελαφρώς: Αρχικά, τα ποτά όπως η σόδα, η κονσερβοποιημένη μπύρα, η εμφιαλωμένη μπύρα, το κόκκινο/ροζέ κρασί, το λευκό κρασί και ο αφρώδες κρασί, αλλά και είδη όπως λουκάνικο, λουκάνικο Φρανκφούρτης, κατεψυγμένα φρούτα, κατεψυγμένα επιδόρπια και κατεψυγμένα προϊόντα πατάτας, αλλά και μαγειρικά σκεύη, παρουσιάζουν αυξημένη ζήτηση. Αυτή η αύξηση υποδηλώνει ότι οι άνθρωποι προμηθεύονται ποτά και φαγητό για πάρτι, συγκεντρώσεις και μαγείρεμα κατά τη διάρκεια της εορταστικής περιόδου. Τα σνακ και τα γενικότερα γλυκίσματα σημειώνουν επίσης άνοδο στις πωλήσεις, συμπεριλαμβανομένων προϊόντων όπως σοκολάτα, παγωτό, καραμέλα, επιδόρπιο. Αυτή η άνοδος αντανακλά το πνεύμα των εορτών, με τους ανθρώπους να τείνουν να επιτρέπουν στον εαυτό τους και στην οικογένεια τους απολαύσεις, όπως γλυκά και επιδόρπια ως μέρος των εορταστικών εορτασμών. Επιπλέον, υπάρχει αύξηση στις πωλήσεις ειδών πρώτης ανάγκης για το σπίτι, όπως χαρτοπετσέτες, τσάντες για ψώνια, καθαριστικά και απορρυπαντικά. Αυτό υποδηλώνει ότι οι πελάτες προετοιμάζουν ενεργά τα σπίτια τους για τους επισκέπτες, που πιθανόν σημαίνει περισσότερη έμφαση στη καθαριότητα. Τέλος,

η περίοδος των γιορτών παρουσιάζει επίσης αύξηση στις αγορές ειδών δώρου (όπως διάφορα είδη κρασιού, λουλούδια ή ρούχα) και διακοσμητικών ειδών (όπως κεριά και λουλούδια). Αυτά τα προϊόντα υποδηλώνουν μια στροφή προς τη γιορτινή διακόσμηση και τα δώρα κατά τη διάρκεια των Χριστουγεννιάτικων επισκέψεων.

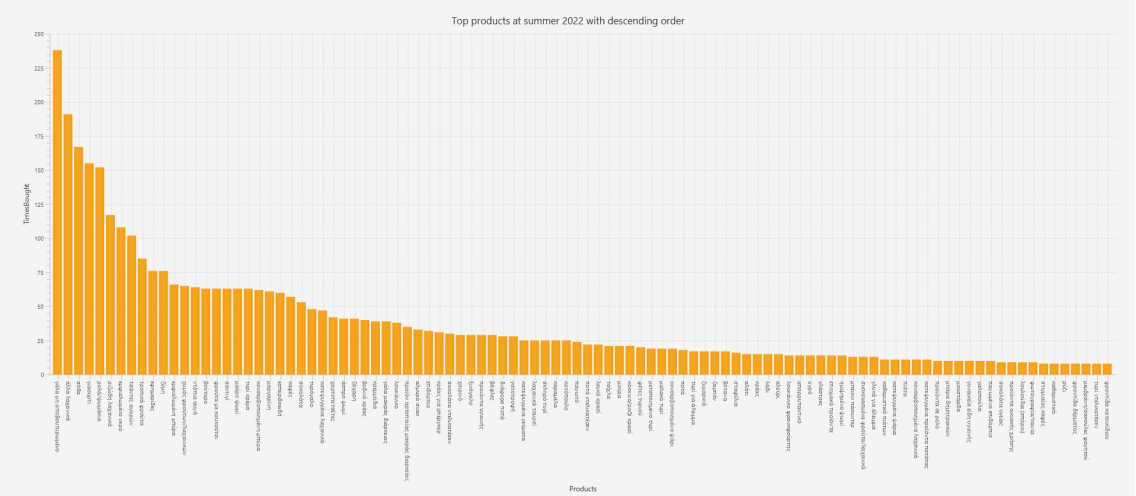
3.3.2 Δημοφιλή προϊόντα το Πάσχα



Εικόνα 4: Δημοφιλία προϊόντων το Πάσχα

Από τα παραπάνω δεδομένα βλέπουμε ότι κατά τη περίοδο του Πάσχα, η αγοραστική συμπεριφορά των καταναλωτών επηρεάζεται με παρόμοιο τρόπο με αυτή κατά την περίοδο των Χριστουγέννων (δείτε 3.3.1), και πιθανόν και άλλων εορταστικών περιόδων.

3.3.3 Δημοφιλή προϊόντα το καλοκαίρι



Εικόνα 5: Δημοφιλία προϊόντων το καλοκαίρι

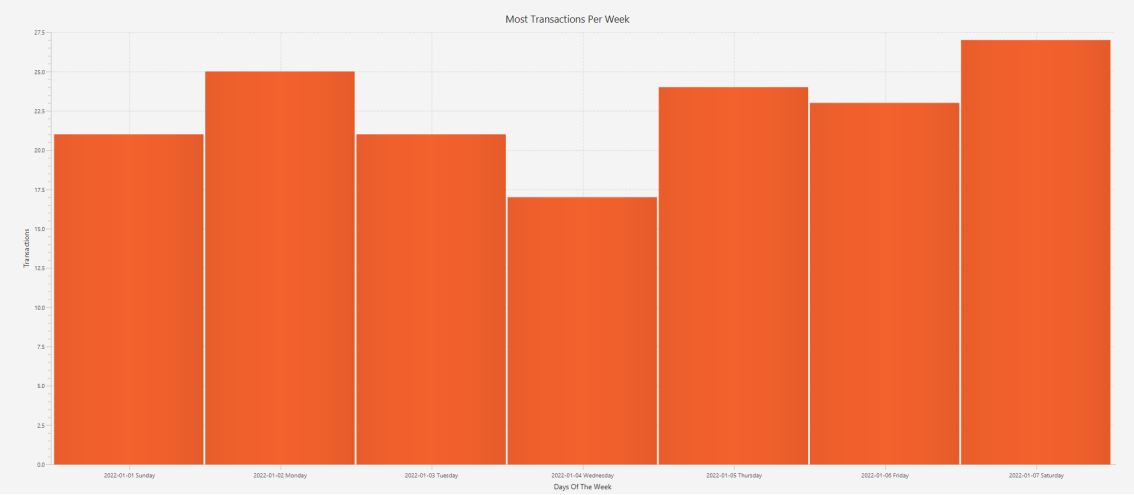
Από τα παραπάνω δεδομένα βλέπουμε ότι κατά τους καλοκαιρινούς μήνες, η αγοραστική συμπεριφορά των καταναλωτών επηρεάζεται ελαφρώς: Η αύξηση σε πωλήσεις των λαχανικών και κυρίως τω φρούτα (και κυρίως των τροπικών φρούτων) τονίζει τη σχέση τους με τα καλοκαιρινά δροσερά, φρέσκα γεύματα/επιδόρπια και την απόλαυση εποχικών (κυρίως) φρούτων σε πχ. φρουτοσαλάτες. Η κατανάλωση ποτών σημειώνει αξιοσημείωτη αύξηση κατά τη διάρκεια του καλοκαιριού, με προϊόντα όπως το εμφιαλωμένο νερό, η σόδα, η εμφιαλωμένη μπύρα και ο χυμός φρούτων να εμφανίζονται σε σχετικά υψηλή θέση. Αυτή η άνοδος στις πωλήσεις ποτών πιθανότατα αντιστοιχεί σε αυξημένες ανάγκες ενυδάτωσης και δροσιάς εξαιτίας της ζέστης. Το ψήσιμο στη σχάρα και το μαγείρεμα σε εξωτερικούς χώρους αναδεικνύονται ως δημοφιλείς καλοκαιρινές δραστηριότητες, όπως αποδεικνύεται από τις πωλήσεις ειδών όπως λουκάνικο, χοιρινό, μοσχάρι, κρέας για μπέργκερ, κοτόπουλο και κατεψυγμένα ψάρια. Αυτά τα προϊόντα συνδέονται συνήθως με μπάρμπεκιου και υπαίθριες συγκεντρώσεις, οι οποίες μπορεί και να συνδέονται με τις πωλήσεις εμφιαλωμένης μπύρας, κονσερβοποιημένης μπύρας, κρασιού και σνακ, συνήθη ποτά σε αντίστοιχες εκδηλώσεις και πάρτι.

3.4 Συναλλαγές ανά χρονική περίοδο

Από προγραμματιστικής πλευράς (Controller4, Controller5,Controller6), για να καταλήξουμε στα παρακάτω αποτελέσματα, διαβάζουμε το αρχείο CSV με τις συναλλαγές, βρίσκουμε τις διαφορετικές ημερομηνίες αγορών και τα διαφορετικά προϊόντα σε αυτές,

μαζί με το πόσες φορές αγοράστηκαν. Χρησιμοποιούμε ένα TreeMap της μορφής Map¡key date, value Map¡key product, value timesBought¿¿, για να αποθηκεύει τα δεδομένα των συναλλαγών μας. ArrayList Τέλος, βάζουμε τα δεδομένα σε ArrayList (όπου τα προϊόντα γίνονται decode σε UTF-8 για χρήση στην FXML) για να τα ταξινομήσουμε με βάση τις εμφανίσεις των διαφορετικών προϊόντων. Τέλος, δημιουργούμε ένα barchart από την JavaFX, εμφανίζοντας τα προϊόντα στον άξονα x και τις αντίστοιχες εμφανίσεις τους στον άξονα y.
Τα αποτελέσματα βρίσκονται στα αρχεια "topDayPerYear.csv", "weeksTransactions.csv" (Controller 4), "monthBuys.csv" (Controller 6) αντίστοιχα που παράγονται αυτόματα με το πάτημα του αντίστοιχου κουμπιού.

3.4.1 Συναλλαγές ανά εβδομάδα

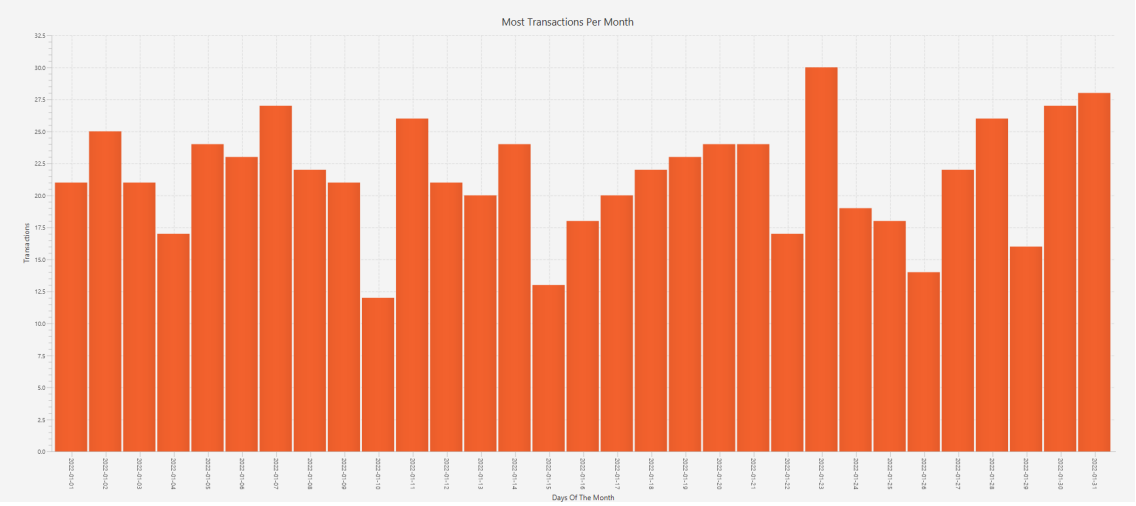


Εικόνα 6: Συναλλαγές εβδομάδας 01-07/01/2022

Ομαδοποιόντας τα δεδομένα ανά ημέρες της εβδομάδας βλέπουμε τις τάσεις στις ημέρες πωλήσεων το 2022 και το 2023. Από τα παραπάνω δεδομένα παρατηρούμε ότι η Δευτέρα αναδεικνύεται σταθερά ως η ημέρα με τις υψηλότερες πωλήσεις και τα δύο χρόνια, υποδεικνύοντας την έντονη προτίμηση των πελατών για αγορές στην αρχή της εβδομάδας. Τα Σαββατοκύριακα παρουσιάζουν επίσης σημαντική δραστηριότητα πωλήσεων, με το Σάββατο να προηγείται το 2022 και την Κυριακή το 2023, υποδηλώνοντας ότι οι πελάτες χρησιμοποιούν τον ελεύθερο χρόνο τους για αγορές ή το γεγονός ότι οι μεγάλες αλυσίδες σουπερμάρκετ είναι κλειστές τα Σαββατοκύριακα. Παρά τις διακυμάνσεις στην ημέρα των κορυφαίων πωλήσεων μεταξύ ετών, οι

συνολικές πωλήσεις παραμένουν σχετικά σταθερές, υποδηλώνοντας μια σταθερή βάση πελατών και μοτίβα αγορών.

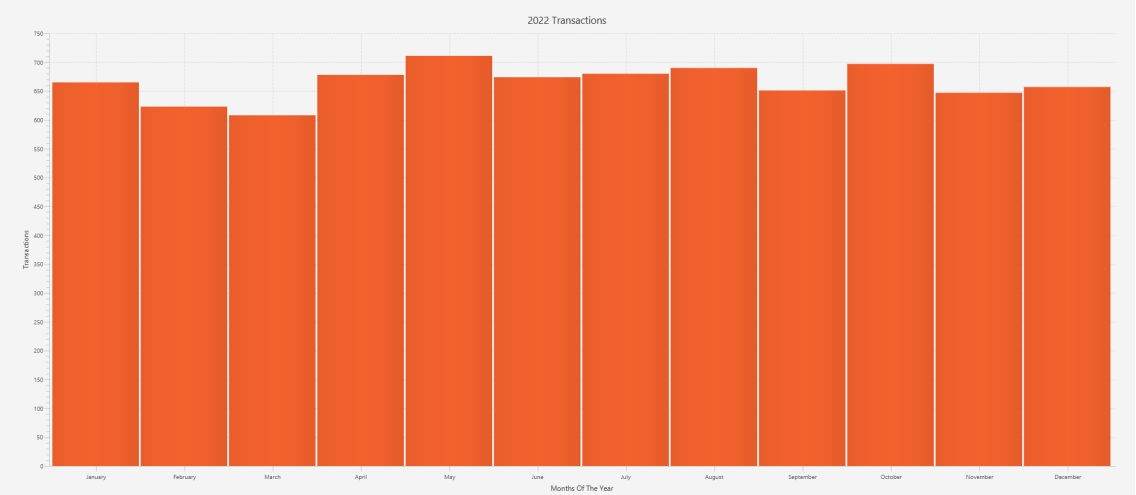
3.4.2 Συναλλαγές ανά μήνα



Εικόνα 7: Συναλλαγές μηνός Ιανουαρίου 2022

Ομαδοποιόντας τα δεδομένα ανά μήνες του έτους βλέπουμε τις τάσεις στους μήνες πωλήσεων το 2022 και το 2023. Από τα παραπάνω δεδομένα παρατηρούμε ότι υπάρχει μια ευδιάκριτη εποχιακή τάση με τις κορυφαίες πωλήσεις να σημειώνονται στους μεσαίους μήνες του έτους (Μάιος με Αύγουστος), πιθανώς επηρεαζόμενες από παράγοντες όπως ο καιρός, ο ελεύθερος χρόνος, τα τοπικά γεγονότα και κυρίως ο τουρσιμός. Ωστόσο, μια σύγκριση από έτος σε έτος δείχνει διακυμάνσεις, με ορισμένους μήνες να παρουσιάζουν υψηλότερες πωλήσεις το 2022 σε σύγκριση με το 2023 και αντίστροφα. Συνολικά, υπάρχει μια ανησυχητική μείωση στις πωλήσεις από το 2022 έως το 2023 για πολλούς μήνες, υποδηλώνοντας την ανάγκη για περαιτέρω ανάλυση σε πιθανές αιτίες, όπως αλλαγές στη συμπεριφορά των καταναλωτών ή τη δυναμική της αγοράς.

3.4.3 Συναλλαγές ανά έτος



Εικόνα 8: Συναλλαγές έτους 2022

Συναθροίζοντας τα παραπάνω δεδομένα, παρατηρούμε ότι το mini market παρουσίασε αξιοσημείωτη πτώση στις πωλήσεις από το 2022 έως το 2023, με τα νούμερα να πέφτουν από 8897 σε 7966. Αυτή η μείωση υποδηλώνει πιθανά προβλήματα, όπως αλλαγές στη συμπεριφορά των καταναλωτών, οικονομικές συνθήκες ή αυξημένο ανταγωνισμό. Για να αντιμετωπιστεί αυτή η πτώση, το mini market θα πρέπει να διεξάγει μια ανάλυση των τάσεων πωλήσεων (όπως η ανάλυση συχνών προτύπων που θα ακολουθήσει) για να αποκτήσει καλύτερες στρατηγικές μάρκετινγκ.

4 Μετατροπή σε ARFF file

4.1 Προεπεξεργασία δεδομένων

Το WEKA προκειμένου να κάνει εξαγωγή συχνών προτύπων σε πρόβλημα καλαθιού supermarket, απαιτεί έναν συγκεκριμένο τρόπο καταγραφής των δεδομένων των καλαθιών στο αρχείο arff. Έτσι, πρωτού προχωρήσουμε στη μετατροπή του αρχείου σε arff, έπρεπε να ανακατασκευάσουμε το csv με κατάλληλη μορφή [2], για να μπορεί να το επεξεργαστεί το WEKA στην arff μορφή του στη συνέχεια.
Από προγραμματιστικής άποψης, διαβάζουμε το αρχείο CSV με τις συναλλαγές, βρίσκουμε τα διαφορετικά προϊόντα και μετράμε τη συχνότητα εμφάνισης κάθε προϊόντος. Χρησιμοποιούμε ένα HashMap για να αποθηκεύει προϊόντα ως κλειδιά και την αντίστοιχη εμφάνιση τους ως value. Έπειτα βάζουμε τα δεδομένα σε ArrayList για να

τα ταξινομήσουμε με βάση τις εμφανίσεις τους. Έτσι, συμπληρώνουμε στο αρχείο την πρώτη γραμμή με τα 166 διαφορετικά προϊόντα ως header. Στη συνέχεια, βρίσκουμε τους διαφορετικούς πελάτες και τις διαφορετικές ημερομηνίες αγορών τους, μαζί με τα προϊόντα που αγόρασαν σε αυτή την επίσκεψη. Έτσι δημιουργούμε καλάθια αγορών. Χρησιμοποιούμε ένα TreeMap της μορφής Map¡key custID, value Map¡key date, value basket¿¿, για να αποθηκεύει τα δεδομένα των καλαθιών μας. Για κάθε επόμενη γραμμή που δείχνει συναλλαγή ακολουθεί το εξής: Δημιουργούμε ένα LinkedHashMap (δηλαδή ένα HashMap στο οποίο οι εγγραφές γεμίζουν με τη σειρά που τις εισάγουμε, άρα τη σειρά των προϊόντων του header). Αυτό έχει ως keys τα 166 διαφορετικά προϊόντα και ως value του καθενός είναι ",". Στη συνέχεια διατρέχουμε το TreeMap με την αντίστοιχη συναλλαγή και εάν το προϊόν περιέχεται στη συναλλαγή αντικαθιστούμε το "," με "t,". Αφού διατρέψουμε και τα 166 προϊόντα γράφουμε αυτή τη γραμμή/συναλλαγή στο αρχείο, και το ίδιο επαναλαμβάνεται για την επόμενη συναλλαγή.

Το αρχείο csv που κατασκευάστηκε ("arffFile.csv") έχει ως πρώτη γραμμή/header τα 166 διαφορετικά προϊόντα που μπορεί να αγοράσει ο κάθε πελάτης, χωρισμένα με ",". Αυτά στη συνέχεια αποτελούν τα attributes το arff αρχείου.

Έπειτα, κάθε γραμμή του αρχείου csv αποτελεί ένα καλάθι με boolean λογική, με την ίδια κατάξη/σειρά προϊόντων με το header, χωρισμένα με ",". Σε κάθε γραμμή καλαθιού, με τη σειρά των προϊόντων στο header, σημειώνεται η τιμή "t", εάν το προϊόν έχει αγοραστεί, και καμία τιμή (null), εάν το προϊόν δεν έχει αγοραστεί.

Παρόμοια διαδικασία χρησιμοποιήσαμε και για το αρχείο csv που οργανώνει τα καλάθια σε κατηγορίες προϊόντων ("categoryArffFile.csv"), και για τα αρχεία csv για την κάθε ηλικίακή ομάδα ("arffFile18-35.csv", "arffFile36-59.csv", "arffFile60-80.csv").

Για τις κατηγορίες προϊόντων πρώτα, αναθέσαμε τα 166 διαφορετικά προϊόντα σε 31 κατηγορίες μέσω ενός Map¡key category, value productsinCategory¿, και έπειτα έγινε η ίδια επεξεργασία με το LinkedHashMap. Για τις ηλικιακές ομάδες πελατών, πρώτα χωρίσαμε τα καλάθια ανάλογα με την ηλικία του πελάτη (συγκρίνοντας το customer id/key του TreeMap με το αρχείο ηλικιών), και έπειτα έγινε η ίδια επεξεργασία με το LinkedHashMap, στο σωστό αρχείο ανά πελάτη.

4.2 Μετατροπή στο WEKA

Τα βήματα που ακολουθήσαμε [3]για να μετατρέψουμε το csv αρχείο μας σε arff ήταν τα εξής:

- Αφού εκκινήσαμε το WEKA στον Explorer, στην κατηγορία Preprocess, ανοίξαμε το αρχείο csv μας με "Open file..."(όπου και μπορούσαμε να δούμε τα γενικά στατιστικά του).

- Έπειτα επιλέξαμε να σώσουμε εκ νέου το αρχείο "Save..." και αφού επιλέξαμε το directory αποθήκευσης, επιλέξαμε το extension Arff data files (*.arff).

5 Εξαγωγή συχνών προτύπων και προσφορών στα προϊόντα

5.1 Εξαγωγή συχνών προτύπων και κανόνων

Στον Explorer του WEKA και έπειτα στην κατηγορία Assosiate, μπορούμε να επιλέξουμε τον αλγόριθμο Apriori για την εξαγωγή συχνών προτύπων/itemsets και κανόνων συσχετίσεων, αφού πρώτα ορίσουμε τις παραμέτρους του [4].
Αρχικά πρέπει να ορίσουμε το threshold του support (σε πόσες συναλλαγές τουλάχιστον πρέπει να περιέχεται ένα προϊόν/itemset ώστε να το θεωρούμε σημαντικό), το οποίο στο WEKA παίρνει την upper minimum τιμή του (από αυτήν ξεκινά ο αλγόριθμος) και μια lower minimum τιμή του (εδώ πιθανώς να φτάσει). Μία μεταβλητή delta μειώνει αναδρομικά το threshold του support, αν ο minimum αριθμός κανόνων που θέλουμε δεν έχουν συμπληρωθεί.
Προκειμένου να αποφασίσουμε την τιμή του minSupport πρέπει να εξετάσουμε τον αριθμό των συναλλαγών μας αλλά και τη συχνότητα αγοράς των προϊόντων μέσα σε αυτές. Συνήθως η τιμή του minSupport είναι 0.5, δηλαδή το προϊόν εμφανίζεται σε τουλάχιστον το 50% των συναλλαγών. Αυτό σημαίνει ότι για τα δικά μας δεδομένα, όπου έχουμε 14963 συναλλαγές θα έπρεπε ένα προϊόν για να θεωρείτε σημαντικό, να εμφανίζεται τουλάχιστον σε 7482 συναλλαγές. Αυτό ωστόσο για τα δικά μας δεδομένα αποτελεί πρόβλημα καθώς κανένα προϊόν δεν εμφανίζεται σε τόσες συναλλαγές. Το προϊόν με τη μεγαλύτερη συχνότητα εμφανίσης είναι το "γάλα μη αποβουτυρωμένο" με 2502 εμφανίσεις το οποίο είναι μόλις το 10% των 14963 συναλλαγών. Άρα εφόσον το συχνότερο μάλιστα εμφανιζόμενο προϊόν βρίσκεται μόνο στο 10% των συναλλαγών μας πρέπει να μειώσουμε ακόμα περισσότερο το minSupport (έτσι ώστε να μπορέσουμε να αποκτήσουμε και κανόνες με άλλα προϊόντα) και να αναθεωρήσουμε τι εννοούμε ως "σημαντικό προϊόν" στα δεδομένα. Θεωρήσαμε λοιπόν ως μια λογική τιμή minSupport το 0.01, δηλαδή το 1% = "150 τουλάχιστον εμφανίσεις" των συναλλαγών, το οποίο μας επιτρέπει να χρησιμοποιήσουμε περίπου τα μισά από τα 166 προϊόντα μας για

να φτιάξουμε κανόνες συσχετίσεων (καθώς ακόμα και το 5% = "749 τουλάχιστον εμφανίσεις" μας επέτρεπε να φτιάξουμε κανόνες με μόνο 10 προϊόντα). Παρόλα αυτά από τώρα μπορούμε να φανταστούμε πώς ό,τι κανόνες και αν παράξουμε στην συνέχεια δεν θα είναι αρκετά ισχυροί ή έμπιστοι, εφόσον η επανάληψη εμφανίσεων προϊόντων στα δεδομένα μας είναι πολύ σπάνια και διασκορπισμένη, το οποίο μας ανάγκασε να πάρουμε ενα τόσο μικρό ποσοστό minSupport (προκειμένου να παράξουμε έστω και μερικούς κανόνες).

Στη συνέχεια πρέπει να ορίσουμε το threshold του confidence (ποια είναι η ελάχιστη πιθανότητα 2 προϊόντα/itemsets να αγοραστούν μαζί). Συνήθως το minConfidence κυμαίνεται στο 0.6, δηλαδή 60% πιθανότητα 2 προϊόντα/itemsets να αγοραστούν μαζί. Παρόλα αυτά με με τόσο μικρές συχνότητες εμφανίσεων στα δεδομένα μας, είναι πολύ δύσκολο να φτιάξουμε ζευγάρια τα οποία αγοράζονται μαζί τόση βεβαιότητα. Έτσι, μετά από αλλεπάλληλες δοκιμές, το καλύτερο confidence κανόνα που μπορούσαμε να καταφέρουμε ήταν το 0.16, δηλαδή 16% πιθανότητα τα 2 προϊόντα/itemsets να αγοραστούν μαζί, με το lift του κανόνα (την πιθανότητα τα δύο προϊόντα/itemsets να σχετίζονται) να είναι εξίσου κακής ποιότητας/μικρότερη του 1. Παρόλα αυτά αποφασίσαμε να ορίσουμε το minConfidence στο 0.05 δηλαδή στο 5% βεβαιότητας έτσι ώστε να έχουμε κάποιους κανόνες να παρουσιάσουμε, ακόμα και αν δεν είναι πλήρως αξιόπιστοι.

Μέρος των αποτελεσμάτων μαζί με τους κανόνες στους οποίους καταλήξαμε παρουσιάζονται παρακάτω (το πλήρες output του Apriori βρίσκεται στο αρχείο "apriori_prod.file". Θεωρήσαμε λογικό να καταλήξουμε στους 10 καλύτερους κανόνες, που είχαν τουλάχιστον confidence 10%. Επίσης παρατηρούμε ότι η εξαγωγή itemsets με περισσότερα από 2 προϊόντα δεν ήταν δυνατή με τους περιορισμούς μας (καθώς το support είναι αντιμονότονο, δηλαδή όσο μεγαλώνουμε το ήδη υπάρχον itemset -υποσύνολο-, τόσο μειώνεται το support του νέου μεγαλύτερου itemset -συνόλου-), αλλά και δίχως τους περιορισμούς οι τιμές των support και confidence θα ήταν υπερβολικά χαμηλές και θα συμπεριλαμβάνονταν άσκοπα όλα τα προϊόντα. Οπότε όλοι οι κανόνες προέκυψαν από τα 2-itemsets.

```
Apriori
=======
Minimum support: 0.01 (90 instances)
Minimum metric <confidence>: 0.05
Number of cycles performed: 199
```

```
Generated sets of large itemsets:
Size of set of large itemsets L(1): 89
Size of set of large itemsets L(2): 37

Best rules found:
1. εμφιαλωμένη μπύρα=t 678 => γάλα μη αποβουτυρωμένο=t 107 (c:0.16)
2. λουκάνικο=t 903 => γάλα μη αποβουτυρωμένο=t 134 (c:0.15)
3. εφημερίδες=t 582 => γάλα μη αποβουτυρωμένο=t 84 (c:0.14)
4. ντόπια αυγά=t 555 => γάλα μη αποβουτυρωμένο=t 79 (c:0.14)
5. λουκάνικο φρανκφούρτης=t 565 => γάλα μη αποβουτυρωμένο=t 79 (c:0.14)
6. λουκάνικο φρανκφούρτης=t 565 => άλλα λαχανικά=t 77 (c:0.14)
7. χοιρινό=t 555 => γάλα μη αποβουτυρωμένο=t 75 (c:0.16)
8. φρούτα με κουκούτσι=t 734 => γάλα μη αποβουτυρωμένο=t 99 (c:0.13)
9. εσπεριδοειδή=t 795 => γάλα μη αποβουτυρωμένο=t 107 (c:0.13)
10. τσάντες αγορών=t 712 => γάλα μη αποβουτυρωμένο=t 95 (c:0.13)
```

5.2 Προτάσεις προσφορών και χωροταξίας

Παρά τους περιορισμούς των δεδομένων μας και τα χαμηλά επίπεδα εμπιστοσύνης, έχουμε κάποιες πληροφορίες για πιθανά πρότυπα συμπεριφοράς και συχνών αγορών των πελατών. Από τους κανόνες βλέπουμε πως το μη αποβουτυρωμένο γάλα αναδεικνύεται ως βασικό προϊόν (καθώς είναι το πιο συχνά εμφανιζόμενο στα καλάθια), και συσχετιζόμενο με διάφορα άλλα είδη γενικότερα όμως άσχετα μεταξύ τους.

Για να αξιοποιήσει αυτές τις συσχετίσεις, το mini market μπορεί να εισαγάγει στοχευμένες εκπτώσεις και προσφορές τύπου "1+1". Για παράδειγμα, οι πελάτες που αγοράζουν εμφιαλωμένη μπύρα, λουκάνικα Φρανκφούρτης, χοιρινό, εφημερίδες, τοπικά αυγά, φρούτα, εσπεριδοειδή ή τσάντες αγορών μπορούν να επωφεληθούν από εκπτώσεις ή δωρεάν μη αποβουτυρωμένο γάλα ως κίνητρο. Ομοίως, θα μπορούσαν να εισαχθούν συμπληρωματικές προσφορές "1+1", όπως η μια δωρεάν εφημερίδα με την αγορά εμφιαλωμένη μπύρα ή ενός/δύο δωρεάν αυγών με αγορές λουκάνικων ή μια τσάντα αγορών να δίνεται δώρο με αγορές αυτών των προϊόντων. Αυτές οι προσφορές όχι μόνο ενθαρρύνουν τους πελάτες να αγοράσουν περισσότερα, αλλά ενισχύουν και την αίσθηση της αξίας.

Η στρατηγική τοποθέτηση ειδών στο mini mark είναι σημαντική για τη μεγιστοποίηση των πωλήσεων. Με βάση τους κανόνες συσχέτισης, τα σχετικά προϊόντα θα πρέπει να ομαδοποιούνται. Για παράδειγμα, η τοποθέτηση εμφιαλωμένης μπύρας, λουκάνικων

Φρανκφούρτης, χοιρινού, εφημερίδων, αυγών, φρούτων, εσπεριδοειδών ή τσαντών αγορών σε κοντινή απόσταση μπορεί να διευκολύνει ευκαιρίες διασταυρωμένων πωλήσεων με μη αποβουτυρωμένο γάλα. Επιπλέον, η τοποθέτηση του μη αποβουτυρωμένου γάλακτος σε εμφανές σημείο, ίσως κοντά στην είσοδο ή σε περιοχές με μεγάλη κίνηση, διασφαλίζει την ορατότητα και την προσβασιμότητά του στους πελάτες, ενθαρρύνοντας περαιτέρω τις παρορμητικές αγορές.

5.3 Συγκριση Apriori και FPGrowth

ο αλγόριθμος FPGrowth με τις ίδιες παραμέτρους με τον Apriori (minSupport= 0.01, minConfidence= 0.05, NumOfRules=10), είχε ως αποτέλεσμα ακριβώς τους ίδιους κανόνες, όπως φαίνεται παρακάτω (το πλήρες output του FPGrowth βρίσκεται στο αρχείο "fpgrowth_prod.file"):

```
FPGrowth
=========
Relation:     arrffFile
Instances:    14963
Attributes:   166
              [list of attributes omitted]
=== Associator model (full training set) ===

FPGrowth found 47 rules (displaying top 10)

Best rules found:
1. εμφιαλωμένη μπύρα=t 678 => γάλα μη αποβουτυρωμένο=t 107 (c:0.16)
2. λουκάνικο=t 903 => γάλα μη αποβουτυρωμένο=t 134 (c:0.15)
3. εφημερίδες=t 582 => γάλα μη αποβουτυρωμένο=t 84 (c:0.14)
4. ντόπια αυγά=t 555 => γάλα μη αποβουτυρωμένο=t 79 (c:0.14)
5. λουκάνικο φρανκφούρτης=t 565 => γάλα μη αποβουτυρωμένο=t 79 (c:0.14)
6. λουκάνικο φρανκφούρτης=t 565 => άλλα λαχανικά=t 77 (c:0.14)
7. χοιρινό=t 555 => γάλα μη αποβουτυρωμένο=t 75 (c:0.16)
8. φρούτα με κουκούτσι=t 734 => γάλα μη αποβουτυρωμένο=t 99 (c:0.13)
9. εσπεριδοειδή=t 795 => γάλα μη αποβουτυρωμένο=t 107 (c:0.13)
10. τσάντες αγορών=t 712 => γάλα μη αποβουτυρωμένο=t 95 (c:0.13)
```

Η κύρια διαφορά των δύο αλγορίθμων παρατηρείται κυρίως στο θέμα του χρόνου εκτέλεσης και της χρήσης πόρων.
Ο αλγόριθμος Apriori χρειάστηκε περίπου 14.36 δευτερόλεπτα και χρησιμοποίησε το 12% της χρησιμοποιούμενης CPU για να εξάγει τους παραπάνω 10 κανόνες.
Ο αλγόριθμος FPGrowth χρειάστηκε περίπου 1.14 δευτερόλεπτα και χρησιμοποίησε μόλις το 1% της χρησιμοποιούμενης CPU για να εξάγει τους ίδιους 10 κανόνες.
Οπότε είναι σαφές πως, ενώ ο FPGrowth στο WEKA δεν μας επιτρέπει να δούμε τα itemset/συχνά πρότυπα, ότι ως αλγόριθμος είναι εξίσου αποτελεσματικός, με μεγάλο κέρδος στην ταχύτητα και στη διαχείριση πόρων σε σχέση με τον Apriori.
Στα παρακάτω ερωτήματα, τα οποία εκτελούνται σε ομάδες προϊόντων, όπου έχουμε πολύ λιγότερα attributes (από 1066 σε 31), ο Apriori και ο FPGrowth έχουν πάρόμοια σύντομους χρόνους εκτέλεσης και μικρή κατανάλωση πόρων. Οπότε συμπεραίνουμε ότι τουλάχιστον η ταχύτητα των αλγορίθμων, εξαρτάται εν μέρη από τα πόσα διαφορετικά items (attributes) έχουμε στην βάση με τις συναλλαγές μας.

6 Εξαγωγή συχνών προτύπων και προσφορών σε κατηγορίες προϊόντων

6.1 Κατηγοριοποίηση προϊόντων

Αρχικά πρέπει να ομαδοποιήσουμε χειροκίνητα τα προϊόντα μας σε παρεμφερείς κατηγορίες ή "διαδρόμους σουπερ-μάρκετ", με τρόπο με τον οποίο τα προϊόντα της κατηγορίας θα είναι συναφή/σχετικά μεταξύ τους. Οι κατηγορίες που έχουμε δημιουργήσει και τα προϊόντα που συμπεριλαμβάνονται σε αυτές, περιέχονται στο αρχείο "MyProdCategories.txt"), ταξινομημένες βάσει τον αριθμό των συναλλαγών που εμφανίζεται τουλαχιστον ένα προϊον της κατηγορίας. Ένα παράδειγμα κατηγοριοποίησης είναι στην κατηγορία ονόματι "ΨΩΜΙΑ" εισάγαμε τα προϊόντα "παξιμάδι", "άσπρο ψωμί", "ρολά/ψωμάκια", "μαύρο ψωμί" και "ημιψημένο ψωμί". Αν ένα ή περισσότερα από αυτά υπάρχουν σε μία συναλλαγή, τότε έχουμε μια εμφάνιση για τη κατηγορία "ΨΩΜΙΑ".

6.2 Εξαγωγή συχνών προτύπων και κανόνων

Στον Explorer του WEKA και έπειτα στην κατηγορία Assosiate, μπορούμε να επιλέξουμε τον αλγόριθμο Apriori για την εξαγωγή συχνών προτύπων/itemsets και κανόνων συσχετίσεων, αφού πρώτα ορίσουμε τις παραμέτρους του.

Αρχικά πρέπει να ορίσουμε το threshold του support (σε πόσες συναλλαγές τουλάχιστον πρέπει να περιέχεται ένα προϊόν/itemset ώστε να το θεωρούμε σημαντικό). Προκειμένου να αποφασίσουμε την τιμή του minSupport πρέπει να εξετάσουμε τον αριθμό των συναλλαγών μας αλλά και τη συχνότητα αγοράς των κατηγοριών μέσα σε αυτές. Συνήθως η τιμή του minSupport είναι 0.5, δηλαδή το προϊόν εμφανίζεται σε τουλάχιστον το 50% των συναλλαγών. Αυτό σημαίνει ότι για τα δικά μας δεδομένα, όπου έχουμε 14963 συναλλαγές θα έπρεπε μια κατηγορία προϊόντων για να θεωρείτε σημαντική, να εμφανίζεται τουλάχιστον σε 7482 συναλλαγές. Αυτό ωστόσο για τα δικά μας δεδομένα αποτελεί πρόβλημα καθώς καμία κατηγορία δεν εμφανίζεται σε τόσες συναλλαγές. Η κατηγορία με τη μεγαλύτερη συχνότητα εμφανίσης είναι τα "ΦΡΟΥΤΑ & ΛΑΧΑΝΙΚΑ" με 5520 εμφανίσεις το οποίο είναι το 30% των 14963 συναλλαγών. Άρα εφόσον η συχνότερο εμφανιζόμενη κατηγορία βρίσκεται μόνο στο 30% των συναλλαγών μας πρέπει να μειώσουμε ακόμα περισσότερο το minSupport (έτσι ώστε να μπορέσουμε να αποκτήσουμε και κανόνες με άλλες κατηγορίες) και να αναθεωρήσουμε τι εννοούμε ως "σημαντική κατηγορία" στα δεδομένα. Θεωρήσαμε λοιπόν ως μια λογική τιμή minSupport το 0.05, δηλαδή το 5% = "750 τουλάχιστον εμφανίσεις" των συναλλαγών, το οποίο μας επιτρέπει να χρησιμοποιήσουμε κάτι λιγότερο τις μισές από τις 31 κατηγορίες μας για να φτιάξουμε κανόνες συσχετίσεων. Παρόλα αυτά από τώρα μπορούμε να φανταστούμε πώς ό,τι κανόνες και αν παράξουμε στην συνέχεια δεν θα είναι αρκετά ισχυροί ή έμπιστοι, εφόσον η επανάληψη εμφανίσεων προϊόντων στα δεδομένα μας είναι πολύ σπάνια και διασκορπισμένη, το οποίο μας ανάγκασε να πάρουμε ενα τόσο μικρό ποσοστό minSupport (προκειμένου να παράξουμε έστω και μερικούς κανόνες).

Ωστόσο συγκριτικά με την εξαγωγή συχνών προτύπων στα προϊόντα, τα αποτελέσματα που θα λάβουμε με τις κατηγορίες, λογικά θα είναι καλύτερα και πιο έμπιστα, καθώς οι κατηγορίες έχουν μεγαλύτερη συχνότητα εμφανίσεων (αφού περιλαμβάνουν μέσα διάφορα προϊόντα).

Στη συνέχεια πρέπει να ορίσουμε το threshold του confidence (ποια είναι η ελάχιστη πιθανότητα 2 προϊόντα/itemsets να αγοραστούν μαζί). Συνήθως το minConfidence κυμαίνεται στο 0.6, δηλαδή 60% πιθανότητα 2 προϊόντα/itemsets να αγοραστούν μαζί. Παρόλα αυτά με με τόσο μικρές συχνότητες εμφανίσεων στα δεδομένα μας, είναι πολύ δύσκολο να φτιάξουμε ζευγάρια τα οποία αγοράζονται μαζί τόση βεβαιότητα. Έτσι, μετά από αλλεπάλληλες δοκιμές, το καλύτερο confidence κανόνα που μπορούσαμε να καταφέρουμε ήταν περίπου 0.3, δηλαδή 30% πιθανότητα οι 2 κατηγορίες/itemsets να αγοραστούν μαζί, με το lift του κανόνα (την πιθανότητα τα δύο προϊόντα/itemsets να σχετίζονται) να είναι εξίσου κακής ποιότητας/μικρότερη του 1. Παρόλα αυτά απο-

φασίσαμε να ορίσουμε το minConfidence στο 0.1 δηλαδή στο 10% βεβαιότητας έτσι ώστε να έχουμε κάποιους κανόνες να παρουσιάσουμε, ακόμα και αν δεν είναι πλήρως αξιόπιστοι.

Μέρος των αποτελεσμάτων μαζί με τους κανόνες στους οποίους καταλήξαμε παρουσιάζονται παρακάτω (το πλήρες output του Apriori βρίσκεται στο αρχείο "apriori_category.file").

Θεωρήσαμε λογικό να καταλήξουμε στους 10 καλύτερους κανόνες, που είχαν τουλάχιστον confidence 10%. Επίσης παρατηρούμε ότι η εξαγωγή itemsets με περισσότερα από 2 προϊόντα δεν ήταν δυνατή με τους περιορισμούς μας (καθώς το support είναι αντιμονότονο, δηλαδή όσο μεγαλώνουμε το ήδη υπάρχον itemset -υποσύνολο-, τόσο μειώνεται το support του νέου μεγαλύτερου itemset -συνόλου-), αλλά και δίχως τους περιορισμούς οι τιμές των support και confidence θα ήταν υπερβολικά χαμηλές και θα συμπεριλαμβάνονταν άσκοπα όλες οι κατηγορίες. Οπότε όλοι οι κανόνες προέκυψαν από τα 2-itemsets.

Ωστόσο συγκριτικά με την εξαγωγή συχνών προτύπων στα προϊόντα, τα αποτελέσματα που λάβαμε με τις κατηγορίες, ήταν γενικότερα αρκετά καλύτερης ποιότητας καθώς το confidence των περισσότερων κανόνων κατηγοριών ήταν πάνω από το 20% (όχι αρκετά καλό, αλλά σίγουρα καλύτερο από το maximum 15% που πρόκυπτε από τα προϊόντα μόνο). Επίσης παρατηρούμε ότι οι κανόνες των κατηγοριών έχουνε μεγάλη σχέση με τους αυτούς των προϊόντων: π.χ. όπου βλέπουμε "ΚΡΕΑΤΙΚΑ" συνεισφέρει το "χοιρινό" και το "λουκάνικο Φρανκφούρτης", όπου βλέπουμε "ΦΡΟΥΤΑ & ΛΑΧΑΝΙΚΑ" συνεισφέρουν τα "φρούτα με κουκούτσι", τα "εσπεριδοειδή" και τα "άλλα λαχανικά", όπου βλέπουμε "ΓΑΛΑΚΤΟΚΟΜΙΚΑ" συνεισφέρει το "μη αποβουτυρωμένο γάλα", κτλ., γεγονός που μας δίνει μια παρόμοια συμπεριφορά τους κανόνες. Συμπερασματικά, η εξόρυξη συχνών προτύπων και κανόνων είναι πολύ πιο αποτελεσματική σε ομαδοποιημένα προϊόντα, όπου με ακόμα πιο περιοριστικές παραμέτρους, λάβαμε πολύ καλύτερους και έμπιστους κανόνες.

```
Apriori
=======
Minimum support: 0.04 (599 instances)
Minimum metric <confidence>: 0.1
Number of cycles performed: 48

Generated sets of large itemsets:
Size of set of large itemsets L(1): 12
Size of set of large itemsets L(2): 11
```

```
Best rules found:
 1. ΑΛΚΟΟΛΟΥΧΑ=t 2468 => ΦΡΟΥΤΑ & ΛΑΧΑΝΙΚΑ=t 704 (c:0.29)
 2. ΚΡΕΑΤΙΚΑ=t 2702 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 742 (c:0.27)
 3. ΚΡΕΑΤΙΚΑ=t 2702 => ΦΡΟΥΤΑ & ΛΑΧΑΝΙΚΑ=t 737 (c:0.27)
 4. ΑΡΤΟΠΟΙΕΙΑ=t 2808 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 743 (c:0.26)
 5. ΑΛΚΟΟΛΟΥΧΑ=t 2468 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 640 (c:0.26)
 6. ΨΩΜΙΑ=t 2650 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 667 (c:0.25)
 7. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 4851 => ΑΡΤΟΠΟΙΕΙΑ=t 743 (c:0.15)
 8. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 4851 => ΚΡΕΑΤΙΚΑ=t 742 (c:0.15)
 9. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 4851 => ΨΩΜΙΑ=t 667 (c:0.14)
10. ΦΡΟΥΤΑ & ΛΑΧΑΝΙΚΑ=t 5520 => ΚΡΕΑΤΙΚΑ=t 737 (c:0.13)
```

6.3 Προτάσεις προσφορών και χωροταξίας

Παρά τους περιορισμούς των δεδομένων μας και τα μέτρια επίπεδα εμπιστοσύνης, έχουμε κάποιες πληροφορίες για πιθανά πρότυπα συμπεριφοράς και συχνών αγορών των πελατών. Από τους κανόνες βλέπουμε πως οι πιο δημοφιλής κατηγορίες συνδεόνται αρκετά μεταξύ τους πληροφορία που πρέπει να αξιοποιήσουμε.

Για να αξιοποιήσει αυτές τις συσχετίσεις, το mini market μπορεί να εισαγάγει στοχευμένες εκπτώσεις και προσφορές τύπου "1+1". Για παράδειγμα, αναγνωρίζοντας τη σχέση μεταξύ αλκοόλ και φρούτων/λαχανικών, το mini market θα μπορούσε να εισαγάγει μια εκπτωτική προσφορά όπου πελάτες που αγοράζουν αλκοόλ έχουν έκπτωση σε επιλεγμένα φρούτα ή λαχανικά. Επιπλέον, αξιοποιώντας τη συσχέτιση μεταξύ κρέατος και γαλακτοκομικών προϊόντων, το mini market μπορεί να εισάγει συνδυαστικές προσφορές. Για παράδειγμα, οι πελάτες που αγοράζουν μια συγκεκριμένη ποσότητα κρέατος θα μπορούσαν να λάβουν έκπτωση σε γαλακτοκομικά προϊόντα όπως το γάλα ή το γιαούρτι. Επιπλέον, αναγνωρίζοντας τη συσχέτιση μεταξύ ψωμιού και ξανά των γαλακτοκομικών προϊόντων, το mini market μπορεί να εισαγάγει προσφορές "1+1", όπως οι πελάτες που αγοράζουν ένα καρβέλι ψωμί θα μπορούσαν να λάβουν κάποιο επιλεγμένο γαλακτοκομικό προϊόν, όπως γάλα ή βούτυρο. Αυτό ενθαρρύνει τους πελάτες να ολοκληρώσουν τις ανάγκες αγορών τους εντός του καταστήματος και αυξάνει το συνολικό μέγεθος του καλαθιού.

Η στρατηγική τοποθέτηση ειδών στο mini mark είναι σημαντική για τη μεγιστοποίηση των πωλήσεων. Με βάση τους κανόνες συσχέτισης, οι σχετικές κατηγορίες πρέπει να βρίσκονται κοντά μεταξύ τους ή η μία αμέσως μετά την άλλη, όπως μια λίστα σούπερ-

μάρκετ. Για παράδειγμα, η τοποθέτηση γαλακτοκομικών προϊόντων πρέπει να είναι σε σχετικά κεντρική θέση, δίπλα στο τμήμα του κρέατος, ή σε επόμενο διάδρομο, αξιοποιώντας τη σχέση μεταξύ κρέατος και γαλακτοκομικών, διευκολύνοντας τις βολικές αγορές για τους πελάτες. Τοποθετώντας το τμήμα αλκοόλ δίπλα στο τμήμα φρούτων και λαχανικών, το mini market ενθαρρύνει τις αγορές μεταξύ των κατηγοριών, παρά το γεγονός ότι η συσχέτιση είναι σχετικά αδύναμη. Ομοίως, η τοποθέτηση ειδών αρτοποιίας κοντά στο τμήμα ψωμιού αξιοποιεί την ήδη υπάρχουσα εννοιολογική συσχέτιση μεταξύ των προϊόντων αρτοποιίας και του ψωμιού, παρακινώντας τους πελάτες να εξερευνήσουν σχετικές προσφορές.
Γενικότερα μια καλή τακτική χωροταξίας θα ήταν διάδρομοι που οδηγούν από την μία στην άλλη κατηγορία, πχ. ο πελάτης ξεκινά από τα αλκοολούχα (που είναι δημοφιλές προϊόν, ακόμα και μόνο του στο καλάθι, εφόσον όμως είναι στην αρχή ο πελάτης αναγκάζεται να διασχίσει όλο το κατάστημα και να δει αν χρειάζεται κάτι άλλο μέχρι το ταμείο), ακολουθεί η μαναβική, έπειτα τα ψωμιά και η αρτοποιία, μετά τα γαλακτοκομικά και τα τυροκομικά (στη μέση/κέντρο του καταστήματος ώστε να συνδέεται πριν και μετά με τα πάντα), και τέλος τα κρεατικά. Τέλος, η διασφάλιση ότι μικρά απλά αντικείμενα όπως καραμέλες, τσίχλες και εφημερίδες, πρέπει να τοποθετούνται στρατηγικά κοντά στην περιοχή ταμείου, καθώς ενθαρρύνει τις παρορμητικές αγορές, ενισχύοντας περαιτέρω τα συνολικά έσοδα από τις πωλήσεις.

7 Εξαγωγή συχνών προτύπων και προσφορών βάσει ηλικιακών ομάδων

7.1 Ηλικιακές ομάδες

Αρχικά να σημειωθεί ότι τα καλάθια είανι οργανωμένα σε κατηγορίες προϊόντων όπως περιγράφεται στο 6.1, καθώς τα αποτελέσματα που λαμβάναμε μας δίναν στατιστικά και κανόνες, καλύτερους και πιο έμπιστους. Όσον αφορά τις ηλικιακές ομάδες, αποφασίσαμε να χωρίσουμε 3 διακριτές κατηγορίες ανάλογα με τις ηλικίες των πελατών. Παρατηρήσαμε ότι το εύρος των ηλικιών κυμαίνεται από 18 έως 80 ετών. Αρχικά δημιουργήσαμε την κατηγορία 18-35 ετών που αφορά τους νέους, έπειτα την κατηγορία 36-59 ετών που αφορά την μέση ηλικία, και τέλος στην κατηγορία 60-80 ετών που αφορά της ηλικιωμένους. Έτσι το εύρος ηλικιών της κάθε κατηγορίας ήταν αρκετά μεγάλο και η κάθε κατηγορία ήταν αρκετά χαρακτηριστική/ξεκάθαρη, ώστε να μπορούμε να βγάλουμε εμπιστά αποτελέσματα με τις λιγότερες δυνατές εκτελέσεις του Apriori. Κάθε ηλικιακή ομάδα έχει το δικό της csv αρχείο με τις συναλλαγές που

έχουνε γίνει από πελάτες που ανήκουν σε αυτήν, και αποκτά το δικό της arff αρχείο για εισοδο στην εκτέλεση του Apriori.
Αναλύοντας τα δεδομένα από όλες τις ηλικιακές ομάδες, τα οποία περιέχονται στο αρχείο "MyProdCategories.txt". Παρατηρήσαμε ότι οι περισσότερες αγορές γίνονται από τη μέση ηλικία, έπειτα από την τρίτη ηλικία και τέλος από τους νεότερους. Παρόλα αυτά τα μοτίβα στις αγορές των πελατών δεν αλλάζουν σημαντικά ειδικά για τις κατηγορίες προϊόντων με τις περισσότερες αγορές όπως είναι οι κατηγορίες "ΦΡΟΥΤΑ & ΛΑΧΑΝΙΚΑ", "ΓΑΛΑΚΤΟΚΟΜΙΚΑ", "ΜΗ-ΑΛΚΟΟΛΟΥΧΑ ΠΟΤΑ", "ΑΡΤΟΠΟΙΕΙΑ", "ΚΡΕΑΤΙΚΑ", "ΨΩΜΙΑ", τα οποία παρέμειναν σταθερά στις πρώτες θέσεις. Στις κατώτερες θέσεις υπήρξαν μικρές αλλαγές στις κατατάξεις, αλλά όχι σημαντικές. Επίσης οι αναλογίες των αγορών ανά ηλικιακή ομάδα σε σχέση με τη συνολικές συναλλαγές που αναλύσαμε πρωτύτερα παραμένουν οι ίδιες. Δηλαδή η μείωση των αριθμών συναλλαγών με τη μείωση αγοράς σγκεκριμένων κατηγοριών προϊόντων είναι ανάλογη. Άρα οδηγούμαστε στο συμπέρασμα ότι τα νούμερα των παραμέτρων που θα επιλέξουμε στη συνέχεια δεν θα χρειαστεί να διαφοροποιηθούν ιδιαίτερα, σε σχέση με τον Apriori στο 6.2.

7.2 Εξαγωγή συχνών προτύπων και κανόνων

Στον Explorer του WEKA και έπειτα στην κατηγορία Assosiate, μπορούμε να επιλέξουμε τον αλγόριθμο Apriori για την εξαγωγή συχνών προτύπων/itemsets και κανόνων συσχετίσεων, αφού πρώτα ορίσουμε τις παραμέτρους του.
Εφόσον οι αναλογίες αγορών προϊόντων με τις συναλλαγές (παρά την διαφορά στον αριθμό συναλλαγών) παραμένουν ίδιες σε όλα τα αρχεία, οι παράμετροι support και confidence που θα ορίσουμε θα είναι ίδιες για όλα τα αρχεία ηλικιακών ομάδων.
Αρχικά πρέπει να ορίσουμε το threshold του support (σε πόσες συναλλαγές τουλάχιστον πρέπει να περιέχεται ένα προϊόν/itemset ώστε να το θεωρούμε σημαντικό). Προκειμένου να αποφασίσουμε την τιμή του minSupport πρέπει να εξετάσουμε τον αριθμό των συναλλαγών μας αλλά και τη συχνότητα αγοράς των κατηγοριών μέσα σε αυτές. Όπως ακριβώς και στο 6.2, η αναλογία του top προϊόντος δίνει support 30%, οπότε πρέπει να μειώσουμε ακόμα περισσότερο το minSupport. Θεωρήσαμε λοιπόν μια ως μια λογική τιμή minSupport το 0.05, δηλαδή το 5% των συναλλαγών, όπως και στο προηγούμενο ερώτημα, καθώς οι αναλογίες διατηρούνται ίδιες. Παρόλα αυτά από τώρα μπορούμε να φανταστούμε πώς ό,τι κανόνες και αν παράξουμε στην συνέχεια δεν θα είναι αρκετά ισχυροί ή έμπιστοι, εφόσον η επανάληψη εμφανίσεων προϊόντων στα δεδομένα μας είναι πολύ σπάνια και διασκορπισμένη, το οποίο μας ανάγκασε να πάρουμε ενα τόσο μικρό ποσοστό minSupport (προκειμένου να παράξουμε έστω και μερικούς κανόνες).

Στη συνέχεια πρέπει να ορίσουμε το threshold του confidence (ποια είναι η ελάχιστη πιθανότητα 2 προϊόντα/itemsets να αγοραστούν μαζί). Όπως ακριβώς και στο 6.2, με με τόσο μικρές συχνότητες εμφανίσεων στα δεδομένα μας, είναι πολύ δύσκολο να φτιάξουμε ζευγάρια τα οποία αγοράζονται μαζί τόση βεβαιότητα. Έτσι, μετά από αλλεπάλληλες δοκιμές, το καλύτερο confidence κανόνα που μπορούσαμε να καταφέρουμε ήταν περίπου 0.3, δηλαδή 30% πιθανότητα οι 2 κατηγορίες/itemsets να αγοραστούν μαζί, με το lift του κανόνα (την πιθανότητα τα δύο προϊόντα/itemsets να σχετίζονται) να είναι εξίσου κακής ποιότητας/μικρότερη του 1. Παρόλα αυτά αποφασίσαμε να ορίσουμε το minConfidence στο 0.1 δηλαδή στο 10% βεβαιότητας έτσι ώστε να έχουμε κάποιους κανόνες να παρουσιάσουμε, ακόμα και αν δεν είναι πλήρως αξιόπιστοι.
Μέρος των αποτελεσμάτων μαζί με τους κανόνες στους οποίους καταλήξαμε παρουσιάζονται παρακάτω (το πλήρες output των Apriori βρίσκεται στα αρχεία "apriori_categ18-35.file", "apriori_categ36-59.file", "apriori_categ60-80.file"). Θεωρήσαμε λογικό να καταλήξουμε στους 10 καλύτερους κανόνες, που είχαν τουλάχιστον confidence 10%. Επίσης παρατηρούμε ότι η εξαγωγή itemsets με περισσότερα από 2 προϊόντα δεν ήταν δυνατή με τους περιορισμούς μας (καθώς το support είναι αντιμονότονο, δηλαδή όσο μεγαλώνουμε το ήδη υπάρχον itemset -υποσύνολο-, τόσο μειώνεται το support του νέου μεγαλύτερου itemset -συνόλου-), αλλά και δίχως τους περιορισμούς οι τιμές των support και confidence θα ήταν υπερβολικά χαμηλές και θα συμπεριλαμβάνονταν άσκοπα όλες οι κατηγορίες. Οπότε όλοι οι κανόνες προέκυψαν από τα 2-itemsets.

```
Apriori 18-35
==============
Minimum support: 0.03 (121 instances)
Minimum metric <confidence>: 0.1
Number of cycles performed: 49

Generated sets of large itemsets:
Size of set of large itemsets L(1): 20
Size of set of large itemsets L(2): 29

Best rules found:
1. ΓΑΫΚΑ=t 422 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 143 (c:0.34)
2. ΒΙΟΛΟΓΙΚΑ=t 289 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 88 (c:0.3)
3. ΤΫΡΟΚΟΜΙΚΑ=t 353 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 102 (c:0.29)
4. ΒΙΟΛΟΓΙΚΑ=t 289 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 83 (c:0.29)
```

```
5. ΑΛΚΟΟΛΟΫΧΑ=t 651 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 186 (c:0.29)
6. ΤΑΜΕΙΟΫ=t 405 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 114 (c:0.28)
7. ΤΑΜΕΙΟΫ=t 405 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 112 (c:0.28)
8. ΚΡΕΑΤΙΚΑ=t 731 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 199 (c:0.27)
9. ΑΛΚΟΟΛΟΫΧΑ=t 651 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 172 (c:0.26)
10. ΤΥΡΟΚΟΜΙΚΑ=t 353 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 90 (c:0.25)
```

```
Apriori 36-59
==============
Minimum support: 0.04 (233 instances)
Minimum metric <confidence>: 0.1
Number of cycles performed: 48

Generated sets of large itemsets:
Size of set of large itemsets L(1): 13
Size of set of large itemsets L(2): 11

Best rules found:
1. ΑΛΚΟΟΛΟΫΧΑ=t 963 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 278 (c:0.29)
2. ΑΛΚΟΟΛΟΫΧΑ=t 963 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 265 (c:0.28)
3. ΨΩΜΙΑ=t 1033 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 284 (c:0.27)
4. ΚΡΕΑΤΙΚΑ=t 1055 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 290 (c:0.27)
5. ΚΡΕΑΤΙΚΑ=t 1055 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 289 (c:0.27)
6. ΑΡΤΟΠΟΙΕΙΑ=t 1106 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 287 (c:0.26)
7. ΨΩΜΙΑ=t 1033 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 263 (c:0.25)
8. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 1880 => ΚΡΕΑΤΙΚΑ=t 289 (c:0.15)
9. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 1880 => ΑΡΤΟΠΟΙΕΙΑ=t 287 (c:0.15)
10. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 1880 => ΑΛΚΟΟΛΟΫΧΑ=t 265 (c:0.14)
```

```
Apriori 60-80
==============
Minimum support: 0.04 (204 instances)
Minimum metric <confidence>: 0.1
```

```
Number of cycles performed: 48

Generated sets of large itemsets:
Size of set of large itemsets L(1): 12
Size of set of large itemsets L(2): 10

Best rules found:
1. ΑΛΚΟΟΛΟΫΧΑ=t 854 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 240 (c:0.28)
2. ΚΡΕΑΤΙΚΑ=t 916 => ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 248 (c:0.27)
3. ΨΩΜΙΑ=t 900 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 239 (c:0.27)
4. ΑΡΤΟΠΟΙΕΙΑ=t 947 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 248 (c:0.26)
5. ΚΡΕΑΤΙΚΑ=t 916 => ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 239 (c:0.26)
6. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 1656 => ΑΡΤΟΠΟΙΕΙΑ=t 248 (c:0.15)
7. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 1656 => ΚΡΕΑΤΙΚΑ=t 239 (c:0.14)
8. ΓΑΛΑΚΤΟΚΟΜΙΚΑ=t 1656 => ΨΩΜΙΑ=t 239 (c:0.14)
9. ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 1922 => ΚΡΕΑΤΙΚΑ=t 248 (c:0.13)
10. ΦΡΟΫΤΑ & ΛΑΧΑΝΙΚΑ=t 1922 => ΑΛΚΟΟΛΟΫΧΑ=t 240 (c:0.12)
```

Η σύγκριση των κανόνων συσχέτισης μεταξύ των ηλικιακών ομάδων αποκαλύπτει τόσο κοινές τάσεις όσο και μοναδικές προτιμήσεις. Γενικά, οι συνολικές συναλλαγές πελατών αποδίδουν κανόνες με μεγαλύτερο confidence και εμπιστοσύνη, υποδεικνύοντας όμως πολλές ομοιότητες. Ωστόσο, η ηλικιακή ομάδα 18-35 επιδεικνύει ελαφρώς ισχυρότερο confidence, και ενώ οι συσχετισμοί όπως το αλκοόλ με τα φρούτα/λαχανικά παραμένουν σε όλες τις ηλικιακές ομάδες σταθερές, οι υπόλοιπες διαφοροποιούνται ελαφρώς. Οι νεότεροι πελάτες (18-35) προτιμούν γλυκά με φρούτα/λαχανικά ή βιολογικά προϊόντα με γαλακτοκομικά, ανταγωνιζόμενοι μοναδικές συνήθειες. Οι πελάτες μέσης ηλικίας (36-59) ευθυγραμμίζονται στενά με τη γενική κατανάλωση, αλλά εμφανίζουν ελαφρώς μεγαλύτερη εμπιστοσύνη. Οι μεγαλύτεροι πελάτες (60-80) εμφανίζουν παρόμοιες συμπεριφορές με μικρές διαφορές, υποδηλώνοντας ποικίλες αγοραστικές συνήθειες. Συνοπτικά, υπάρχουν γενικές τάσεις, και οι παραλλαγές που σχετίζονται με την ηλικία δεν διαμορφώνουν σημαντικά τη συμπεριφορά.

7.3 Προτάσεις προσφορών και χωροταξίας

Παρά τους περιορισμούς των δεδομένων μας και τα μέτρια επίπεδα εμπιστοσύνης, έχουμε κάποιες πληροφορίες για πιθανά πρότυπα συμπεριφοράς και συχνών αγορών των πελατών ανά ηλικιακή ομάδα. Από τους κανόνες βλέπουμε πως οι πιο δημοφιλής

κατηγορίες συνδεόνται αρκετά μεταξύ τους, πληροφορία που πρέπει να αξιοποιήσουμε, ώστε να αλλάξουμε ελαφρώς τις προηγούμενες προτάσεις μας. Παρολαυτά οι προτάσεις μας δεν θα αλλάξουν κατά πολύ σε σχέση με αυτές στο 6.3.
Για την ηλικιακή ομάδα 18-35, η εστίαση στις εκπτώσεις για γλυκά και βιολογικά προϊόντα σε συνδυασμό με φρούτα/λαχανικά καλύπτει τις διαφορετικές προτιμήσεις τους. Επιπλέον, οι συνδυαστικές προσφορές για αλκοόλ με φρούτα/λαχανικά ή γαλακτοκομικά προϊόντα, ή για είδη αρτοποιίας σε συνδυασμό με γαλακτοκομικά προϊόντα ή φρούτα/λαχανικά μπορεί να έχει μεγάλη απήχηση στις αγοραστικές τους συνήθειες.
Στην περίπτωση της ηλικιακής ομάδας 36-59 ετών, είναι απαραίτητη η συνέχιση των εκπτώσεων σε αλκοόλ με φρούτα/λαχανικά και η παροχή συνδυασμένων προσφορών προϊόντων για κρέας και γαλακτοκομικά προϊόντα. Η διατήρηση των προσφορών για ψωμί και γαλακτοκομικά προϊόντα ευθυγραμμίζεται επίσης με τα καταναλωτικά τους πρότυπα. Η στρατηγική τοποθέτηση των προϊόντων είναι σημαντική, διασφαλίζοντας ότι το κρέας και τα γαλακτοκομικά προϊόντα τοποθετούνται μαζί για να καλύψουν την ισχυρή σχέση αυτής της ηλικιακής ομάδας μεταξύ τους.
Για την ηλικιακή ομάδα 60-80, η έμφαση στις εκπτώσεις στο αλκοόλ με φρούτα/λαχανικά και οι συνδυαστικές προσφορές για κρέας και γαλακτοκομικά προϊόντα παραμένει σημαντική. Η εφαρμογή προσφορών για ψωμί με γαλακτοκομικά προϊόντα ή φρούτα/λαχανικά μπορεί να ενθαρρύνει αποτελεσματικά την αγοραστική τους συμπεριφορά. Η διασφάλιση ότι το αλκοόλ τοποθετείται κοντά στο τμήμα φρούτων/λαχανικών και η στρατηγική τοποθέτηση του κρέατος και των γαλακτοκομικών προϊόντων μαζί ανταποκρίνεται στις προτιμήσεις και τις συσχετίσεις τους.

8 Παραλλαγή Apriori με παράγοντα τιμής προϊόντων

Ο αλγόριθμος Apriori εξάγει συχνά πρότυπα/itemsets, με είσοδο μια βάση δεδομένων με transactions/συναλλαγές και ένα νούμερο minimum Support. Εάν αλλάξουμε ελαφρώς τον αλγόριθμο και εισάγουμε ένα threshold για την τιμή των προϊόντων, με παρόμοια λογική όπως το minimum Support, μπορούμε να φτίαξουμε τον Apriori έτσι ώστε να συμπεριλαμβάνει συχνά itemsets (δηλαδή εμφανιζόμενα περισσότερες φορές από το minimum Support), και κάτω από το threshold κάποια τιμής max Price. Παρακάτω[3] παραθέτουμε ψευδοκώδικα για τον original Apriori [5], με τις απαραίτητες αλλαγές ώστε να συμπεριλαμβάνει το threshold τιμής να είναι γραμμένες με κόκκινο.

[3]Η γραμματοσειρά είναι μικρή λόγω στοίχισης και χώρου.

Algorithm 1 Modified Apriori Algorithm

```
1: function Apriori(transactions, minSupport, maxPrice)
2:     L1 ← find_freq_1_itemsets(transactions, minSupport, maxPrice)
3:     freq_itemsets ← L1
4:     k ← 2
5:     while freq_itemsets is not empty do
6:         C_k ← generate_candidates(freq_itemsets, k)
7:         L_k ← prune_candidates(C_k, minSupport, maxPrice)
8:         freq_itemsets ← L_k
9:         k ← k + 1
10:    end while
11:    return freq_itemsets
12: end function

13: function find_freq_1_itemsets(transactions, minSupport, maxPrice)
14:    σ[] ← {}
15:    for trans in transactions do
16:        for item in trans do
17:            σ[item] + +
18:        end for
19:    end for
20:    for item in σ[] do
21:        if σ[item] ≥ minSupport AND item's price ≤ maxPrice then
22:            L1 ← L1 ∪ {item}
23:        end if
24:    end for
25:    return L1
26: end function

27: function generate_candidates(freq_itemsets, k)
28:    candidates ← {}
29:    for itemset1 in freq_itemsets do
30:        for itemset2 in freq_itemsets do
31:            if itemset1[: k − 1] = itemset2[: k − 1] AND itemset1[k − 1] < itemset2[k − 1] then
32:                cand ← itemset1 ∪ itemset2
33:                candidates ← candidates ∪ {cand}
34:            end if
35:        end for
36:    end for
37:    return candidates
38: end function

39: function prune_candidates(candidates, minSupport, maxPrice)
40:    pruned_candidates ← {}
41:    for cand in candidates do
42:        if find_support(cand) < minSupport OR sum_prices(items in cand) > maxPrice then
43:            pruned_candidates ← pruned_candidates ∪ {cand}
44:        end if
45:    end for
46:    return pruned_candidates
47: end function
```

Αρχικά ο Apriori πρέπει να βρει τα συχνά itemsets με 1 αντικείμενο/item, όπου αυτό το item πρέπει να έχει support μεγαλύτερο από το minimum Support (δηλαδή να εμφανίζεται σε περισσότερες συναλλαγές από το minimum Support). Άρα στη συνάρτηση find_freq_1_itemsets, για κάθε item σε κάθε συναλλαγή/transaction από τη βάση των συναλλαγών μας, μετράμε τις φορές που εμφανίζεται. Έπειτα συγκρίνουμε αυτόν τον αριθμό για κάθε item, με το minimum Support, και αν πράγματι είναι μεγαλύτερο τότε το προσθέτουμε στην λίστα L1 με τα συχνά 1-itemsets. Εφόσον έχουμε κάνει και την παραλλαγή με τις τιμές, πρέπει εκτός από το με το minimum Support, να συγκρίνουμε και την τιμή του αντικειμένου εάν ξεπερνάει ή όχι το max Price που έχουμε ορίσει.
Στη συνέχεια ορίζουμε μεταβλητή k, η οποία μας δείχνει πόσα items περιέχονται στα itemset κάθε επανάληψης. Η πρώτη τιμή φυσικά είναι k = 2 και οι επαναλήψεις γίνονται μέχρι να αδειάσει η λίστα των συχνό itemsets (freq_itemsets), όπου το k όλο και αυξάνεται (δηλαδή τα itemsets μεγαλώνουν σε μέγεθος). Σε κάθε επανάληψη βρίσκουμε τα υποψήφια itemsets μεγέθους k (μεγαλώνοντας τα ήδη υπάρχοντα συχνά itemsets) με χρήση της συνάρτησης generate_candidates, και έπειτα "κλαδεύουμε"/αφαιρούμε από τα υποψήφια, τα itemsets που δεν πληρούν τους περιορισμούς μας, με τη συνάρτηση prune_candidates. Έτσι τελικά καταλήγουμε μόνο με τα itemsets τα οποία είναι συχνά (εμφανίζεται σε περισσότερες συναλλαγές από το minimum Support) και πληρούν τους περιορισμούς μας (πχ. τιμής).
Στη συνάρτηση generate_candidates προσπαθούμε να ενώσουμε 2 ήδη υπάρχοντα itemsets. Εάν τα 2 itemsets έχουν το ίδιο prefix (δηλαδή τα αρχικά κομμάτια τους είναι ίδια) και το τελευταίο item είναι διαφορετικό και λεξικογραφικά πρώτο, τότε δημιουργούμε ένα νέο υποψήφιο itemset, και το προσθέτουμε στη λίστα με τα υποψηφια.
Στη συνάρτηση prune_candidates ελέγχουμε εάν κάθε υποψήφιο itemset απο τη λίστα candidates πληρεί τους περιορισμούς μας. Άρα ελέγχουμε εάν κάθε υποψήφιο itemset έχει λιγότερες εμφανίσεις από το minimum Support και εάν το άθροισμα των τιμών κάθε item μέσα στο itemset ξεπερνά την max Price. Αν συμβαίνει οτιδήποτε από τα δύο, τότε το itemset δεν θεωρείται συχνό, "κλαδεύεται"/δεν επιστρέφεται και δεν συμπεριλαμβάνεται στην τελική λίστα συχνών προτύπων που εξάγουμε.

References

[1] Πλατής, "01_tools." Το πρώτο σετ διαφανειών στο μάθημα "Προηγμένα θέματα προγραμματισμού" για την εγκατάσταση εργαλείων του μαθήματος, 2016. Πρόσβαση στις 28-12-2023 από το Eclass του μαθήματος.

[2] G. M. Weiss, "Arff file, the supermaarket example." https://storm.cis.fordham.edu/~gweiss/data-mining/weka-data/supermarket.arff. Πρόσβαση στις 24-01-2024.

[3] S. Deegalla, "Data analytics using weka: Convert csv to arff using weka." https://youtu.be/cs8zaLECzRs?feature=shared, 2020. Πρόσβαση στις 24-01-2024.

[4] I. H. W. Eibe Frank, Mark A. Hall, "The weka workbench." https://www.cs.waikato.ac.nz/ml/weka/Witten_et_al_2016_appendix.pdf. Πρόσβαση στις 24-01-2024.

[5] Wikipedia, "Apriori algorithm." https://en.wikipedia.org/wiki/Apriori_algorithm, 2023. Πρόσβαση στις 24-01-2024.