Πανεπιστήμιο Ιωαννίνων,
Τμήμα Πληροφορικής



# Δεύτερη Σειρά Ασκήσεων

Οι ασκήσεις πρέπει να παραδοθούν μέχρι τη **Δευτέρα, 14 Μαΐου**. Για καθυστερημένες υποβολές ισχύει η πολιτική στην σελίδα του μαθήματος. Κάνετε turn-in, χρησιμοποιώντας την εντολή: turnin assignment2_csXXXX@ple059 <τα αρχεία σας>. Δώσετε επεξηγηματικά ονόματα στα αρχεία σας. Η τελευταία παράδοση μετράει για κάθε άσκηση, και καθορίζει την ώρα παράδοσης. Μπορείτε να παραδώσετε ασκήσεις και μέσω email.

## Άσκηση 1 (15 μονάδες)

Στο βιβλίο του μαθήματος (Εισαγωγή στην Εξόρυξη Δεδομένων, των Tan, Steinbach, και Kumar) εκτός από τους αλγόριθμους clustering που περιγράψαμε στην τάξη, περιγράφονται και οι αλγόριθμοι CLARANS, BIRCH, ROCK, CHAMELEON, DENCLUE. Στο δωρεάν online βιβλίο Mining Massive Datasets των Rajaraman και Ullman, περιγράφεται ο αλγόριθμος CURE (στα αγγλικά). Διαλέξτε ένα από τους παραπάνω αλγορίθμους και περιγράψτε την κεντρική ιδέα του με δικά σας λόγια σε 2-3 παραγράφους. Μπορείτε να διαβάσετε και την σχετική δημοσίευση αν χρειάζεστε περισσότερες λεπτομέρειες.

## Άσκηση 2 (20 μονάδες)

Έχουμε μια συλλογή $D$ από $N$ κείμενα, όπου το κάθε κείμενο είναι μια «σακούλα από λέξεις» διαλεγμένες από κάποιο λεξικό $W$ με $m$ λέξεις. Το κείμενο $d_i$ μπορεί να αναπαρασταθεί ως ένα $m$-διάστατο διάνυσμα, όπου $d_{ij}$ είναι ο αριθμός των φορών όπου η λέξη $w_j$ εμφανίζεται στο κείμενο $d_i$. Θεωρείστε ένα clustering $C = \{c_1 \dots, c_K\}$ των κειμένων στη συλλογή $D$, όπου $1 \leq K \leq N$. Όταν $K = N$ κάθε κείμενο είναι σε ένα cluster μόνο του (singleton clusters), ενώ όταν $K = 1$, όλα τα κείμενα είναι μαζί σε ένα cluster. Ένα cluster $c_i$ είναι η συνένωση όλων των λέξεων από όλα τα κείμενα στο the cluster. Συνεπώς μπορούμε επίσης να το αναπαραστήσουμε ως ένα $m$-διάστατο διάνυσμα: το άθροισμα των διανυσμάτων των κειμένων μέσα στο cluster. Για το cluster $c_i$ χρησιμοποιούμε $n_{ij}$ για τον αριθμό των εμφανίσεων της λέξης $w_j$ στο cluster. Χρησιμοποιούμε $n_i$ για τον αριθμό των λέξεων στο cluster $c_i$, και $n$ για τον αριθμό όλων των λέξεων σε όλα τα clusters. Αν κανονικοποιήσουμε το διάνυσμα για το cluster $c_i$ παίρνουμε μια κατανομή $P(W|c_i)$, όπου $p_{ij} = p(w_j|c_i) = \frac{n_{ij}}{n_i}$ είναι η δεσμευμένη πιθανότητα της λέξης $w_j$ στο cluster $c_i$. Είναι η πιθανότητα αν διαλέξουμε τυχαία μια λέξη από το cluster$c_i$ η λέξη αυτή να είναι $w_j$. Χρησιμοποιούμε $P_i$ για να συμβολίσουμε την κατανομή $P(W|c_i)$. Επίσης, ορίζουμε $p(c_i) = \frac{n_i}{n}$ την πιθανότητα του cluster $c_i$. Είναι η πιθανότητα ότι αν διαλέξουμε τυχαία μια λέξη από όλες τις λέξεις που εμφανίζονται σε όλα τα κείμενα, η λέξη αυτή θα είναι από το cluster $c_i$.

Μπορούμε τώρα να ορίσουμε τη δεσμευμένη εντροπία των λέξεων $W$ δεδομένου του clustering $C$. Έχουμε:

$$H(W|C) = \sum_{c_i \in C} p(c_i) H(W|c_i) = -\sum_{c_i \in C} p(c_i) \sum_{w_j \in W} p(w_j|c_i) \log p(w_j|c_i)$$

Η εντροπία των λέξεων στο cluster $c_i$ είναι $H(W|c_i) = H(P_i)$. Ονομάζουμε την εντροπία $H(P_i)$ την *εντροπία του cluster* $c_i$, και την εντροπία $H(W|C)$, *την εντροπία του clustering* $C$.

Ας θεωρήσουμε τώρα δύο clusters (π.χ., cluster $c_1$ και cluster $c_2$), τα οποία συνενώνουμε για να δημιουργήσουμε ένα νέο cluster $c_*$. Έχουμε ότι $p(c_*) = p(c_1) + p(c_2)$

1. Δείξτε ότι

$$P_* = P(W|c_*) = \frac{p(c_1)}{p(c_*)} P_1 + \frac{p(c_2)}{p(c_*)} P_2$$

Ορίζουμε την γενικευμένη Jensen-Shannon απόκλιση μεταξύ κατανομών $P_1$ και $P_2$ ως

$$D_{JS}(P_1, P_2) = \frac{p(c_1)}{p(c_*)} D_{KL}(P_1||P_*) + \frac{p(c_2)}{p(c_*)} D_{KL}(P_2||P_*)$$

$D_{KL}(P_1||P_*)$ είναι η KL-απόκλιση μεταξύ των κατανομών $P_1$ και $P_*$. Ορίζουμε *C* να είναι το clustering με τα clusters $c_1$ και $c_2$, και *C** το clustering αφού συνενώσουμε τα clusters $c_1$ και $c_2$ στο cluster $c_*$.

2. Δείξτε ότι η αύξηση στην εντροπία είναι

$$H(C^*) - H(C) = p(c_*)\, D_{JS}(P_1, P_2)$$

Υπάρχουν μεθοδολογίες για clustering οι οποίες στοχεύουν στην ελαχιστοποίηση της εντροπίας του clustering, και χρησιμοποιούν την αύξηση της εντροπίας ως μέτρο απόστασης μεταξύ των clusters. Μπορούμε μετά να χρησιμοποιήσουμε ένα agglomerative αλγόριθμο, ή τον *k*-means αλγόριθμο.

Θεωρείστε ένα απλό παράδειγμα όπου το λεξικό αποτελείται από μόνο δύο λέξεις, W = {“sports”, “politics”}, και έχουμε τρία κείμενα $d_1, d_2, d_3$. Το κείμενο $d_1$περιέχει δύο εμφανίσεις της λέξης “sports”, το κείμενο $d_2$ περιέχει τρείς εμφανίσεις της λέξης “sports”, και το κείμενο $d_3$περιέχει πέντε εμφανίσεις της λέξης “politics”.

3. Ποια είναι η εντροπία του κάθε κειμένου και τι σημαίνει αυτό? Ποια είναι η απόσταση των δύο πιο κοντινών κειμένων και τι σημαίνει αυτό? Ποια είναι η εντροπία ενός cluster που περιέχει και τα τρία κείμενα?

## Άσκηση 3 (15 μονάδες)

Ο σκοπός αυτής της άσκησης είναι να πειραματιστείτε με τους αλγόριθμους clustering, και με τα διάφορα κριτήρια επικύρωσης (validation) . Θα χρησιμοποιήσετε το “Iris ” dataset από το UCI repository (http://archive.ics.uci.edu/ml/datasets/Iris), το οποίο περιέχει πληροφορίες για 200 διαφορετικούς τύπους του λουλουδιού Ίρις. Τα λουλούδια έχουν 4 διαφορετικά χαρακτηριστικά, και κατηγοριοποιούνται σε 3 διαφορετικές κλάσεις. Θα τρέξετε τον k-means αλγόριθμο και τις τέσσερεις

παραλλαγές του agglomerative hierarchical clustering αλγορίθμου (min, max, avg, ward). Για τα πειράματα σας μπορείτε να χρησιμοποιήσετε την υλοποίηση του WEKA, ή του MATLAB (πληροφορίες για την χρήση του MATLAB για clustering υπάρχουν στο link από το μάθημα της κα Πιτουρά στη σελίδα του μαθήματος). Μπορείτε να βρείτε το ARFF αρχείο στη σελίδα του μαθήματος, ή να κατεβάσετε τα δεδομένα από το UCI repository.

Για το clustering θα χρησιμοποιήσετε μόνο τα τέσσερα χαρακτηριστικά και όχι την ετικέτα της κλάσης. Οι ετικέτες της κλάσης θα χρησιμοποιηθούν ως εξωτερική αλήθεια (ground truth) για την επικύρωση του παραγόμενου clustering. Υπολογίστε το Precision, Recall,και F-Measure για κάθε συνδυασμό cluster και κλάσης, και το Entropy και Purity για κάθε cluster. Για τον k-means υπολογίστε το average από 10 διαφορετικές επαναλήψεις. Παραδώσετε τους πίνακες με τα αποτελέσματα, και τις παρατηρήσεις σας για την ποιότητα του clustering που παράγουν οι διαφορετικοί αλγόριθμοι.

## Άσκηση 4 (50 μονάδες)

Στο μάθημα περιγράψαμε την Minimum Description Length τεχνική, και την εφαρμογή της στο πρόβλημα του co-clustering. Στην άσκηση αυτή θα πρέπει να εφαρμόσετε την τεχνική στο πρόβλημα της κατάτμησης (segmentation) μιας μονοδιάστατης ακολουθίας. Η είσοδος είναι μια ακολουθία από n αριθμούς με τιμές 0 ή 1. Ο στόχος είναι να σπάσουμε την ακολουθία σε τμήματα, έτσι ώστε το συνολικό κόστος της κωδικοποίησης της ακολουθίας να ελαχιστοποιηθεί.

1. Ορίστε το κόστος της κωδικοποίησης μιας κατάτμησης. Ξεχωρίστε το κόστος της κωδικοποίησης του μοντέλου (model cost), και το κόστος της κωδικοποίησης των δεδομένων βάσει του μοντέλου (data cost). Υπόδειξη: μία κατάτμηση σε K τμήματα ορίζεται από K-1 σημεία διάσπασης (breakpoints).
2. Δώστε ένα ευριστικό (heuristic) αλγόριθμο για το πρόβλημα της εύρεσης μιας κατάτμησης που ελαχιστοποιεί το κόστος. Περιγράψτε τον αλγόριθμο με ψευδοκώδικα.
3. Το πρόβλημα της κατάτμησης μπορεί να λυθεί βέλτιστα σε πολυωνυμικό χρόνο χρησιμοποιώντας δυναμικό προγραμματισμό. Ορίστε τον πίνακα και την σχέση του δυναμικού προγραμματισμού που λύνει το πρόβλημα.
4. Υλοποιείστε ένα αλγόριθμο που να λύνει το πρόβλημα. Τεστάρετε τον αλγόριθμο σας σε μια ακολουθία 1000 τιμών την οποία θα δημιουργήσετε ως εξής:
    a. Διαλέξτε τυχαία 2 σημεία διάσπασης.
    b. Δημιουργείστε τυχαίες τιμές 0/1 για κάθε τμήμα, με διαφορετικές πιθανότητες: στο πρώτο τμήμα η πιθανότητα της τιμής 1 είναι 0.7, στο δεύτερο 0.3, και στο τρίτο 0.9.
    c. Παραδώσετε ένα αρχείο με τα πραγματικά σημεία διάσπασης και αυτά που βρίσκει ο αλγόριθμος σας για 3 διαφορετικά τυχαία inputs.