CSCI 3151: Assignment 3

NOTE: FINAL EXAM IS ON FRIDAY, APRIL 8, 1700-1930, IN TEACHING LAB 4 IN THE COMPUTER SCIENCE BUILDING.

Q1. k-means clustering of tweeter data in R

Study chapter 10 of the book "R and Data Mining: Examples and Case Studies", by Yanchang Zhao (available online). The code in the chapter is available in `http://web.cs.dal.ca/~eem/3151/lectures/R%20tutorials/6-Twitter-data-mining-k-means-medoids-clustering.R`

Look for the documentation of the *kmeans* function in the *stats* package, and find how to set the number of clusters, the maximum number of iterations, and the number of random initializations (default is 1, if more than one is chosen, then the kmeans will run multiple times and return the best clustering according to the total within-cluster sum of squares).

a) Experiment with different parameter settings, and inspect the resulting clusters (by printing the top few words of each cluster). Discuss and summarize your observations. How meaningful are the clusters obtained?

b) Report on the amount of time k-means clustering requires, on what type of hardware.

Q2. k-medoids clustering of tweeter data in R

K-medoids clustering is an algorithm similar to k-means. It selects data points to represent clusters instead of centroids. The most common realization of k-medoids clustering is the Partitioning Around Medoids (PAM) algorithm, described in `https://en.wikipedia.org/wiki/K-medoids`.

a) R package `cluster` contains an implementation of PAM as function `pam`, documented in `http://cran.r-project.org/web/packages/cluster/cluster.pdf`. Function `pamc` which is available in package `fpc` used in the example of the "R and Data Mining" and documented in `http://cran.r-project.org/web/packages/fpc/fpc.pdf` book is a wrapper that uses `pam` from the cluster package plus additional code to automatically estimate the optimal number of clusters.

Experiment with different parameter settings, and inspect the resulting clusters (by printing the top few words of each cluster). Discuss and summarize your observations. Compare with the results using k-means in Q1.

b) what is the complexity of k-medoids for clustering a document set containing N documents, each represented as a vector of M dimensions, into K clusters using I iterations? Clearly explain your answers.

c) when documents are not represented as vectors, but instead a distance function is available as a black box that takes as input two documents and returns a distance value, k-means cannot be applied. How about k-medoids?

d) Report on the amount of time k-medoids clustering requires, on what type of hardware. Compare with the results using k-means in Q1.

Q3. hierarchical clustering of tweeter data in R

Repeat Q1 using the `hclust` function of package stats (see `https://stat.ethz.ch/R-manual/R-devel/library/stats/html/hclust.html`) and the R and Data Mining book. Experiment with the `ward.D, single, complete` agglomeration methods.

Q4. Part-of-speech tagging

There are 264 distinct words in the Brown Corpus (# 5 in `http://www.nltk.org/nltk_data/`) having exactly three possible tags.

1. Draw a table with the integers 1...10 in one column, and the number of distinct words in the corpus having 1...10 distinct tags in the other column.
2. For the word with the greatest number of distinct tags, print out sentences from the corpus containing the word, one for each possible tag.
3. Find the 10 tags with the greatest number of words. Draw another table with the integer 1...10 in one column, and the number of words of each tag in the second column.

Study sections 5.1 and 5.2 of `http://www.nltk.org/book/ch05.html` .

Q5. Support Vector Machines on text data in R

Review ch. 7 of the book "Machine Learning with R", by Brett Lantz, available on Safari

Run linear and radial basis function SVM on the text data set `http://web.cs.dal.ca/~eem/3151/lectures/data/classic/`. Experiment with different parameter settings to get the best possible classification performance according to 3-fold cross validation. Consult the following URL about the implementation of cross validation.
`http://stackoverflow.com/questions/13347443/how-to-perform-10-fold-cross-validation-with-libsvm-in-r`

Summarize your observations from your experimentation. Which of the classes is most accurately classified?

Optional problems

QO1. Vector algebra review

A linear classifier is defined by a separating hyperplane described by the equation $\sum_{i=1}^{M} w_i d_i + b$, where M is the number of dimensions, d_i is the i-th coordinate of vector d, and w_i is the weight corresponding to the i-th coordinate. The w_i's form vector w. The variables in the equation are the d_i, while w_i and b are considered constant. Keep in mind the one-to-one correspondence between a point P in M-dimensional space and the vector v_P of M coordinates representing the position of the point P in that space (position vector).

a) What happens to the hyperplane if b changes, while the w_i remain fixed?

b) What is the geometric meaning of b in the two- and three-dimensional case ($M = 2, 3$).

c) Find the unit normal vector defined as orthogonal to the separating hyperplane in terms of vector w and b . Remember that the notion of orthogonality is defined by the inner product being zero. A vector orthogonal to a hyperplane is orthogonal to all vectors that lie on the hyperplane.

d) Write a vector equation of the form $(d - P_0).n = 0$ describing the separating hyperplane in terms of the unit normal vector n you found in (c), and a point P_0 on the hyperplane (find a specific point on the plane, for example by setting $d_2 = d_3 = ... = d_M = 0$ and computing d_1). d is the variable vector that represents an arbitrary point on the hyperplane, and the . in the vector equation represents the inner product.

e) What is the geometric meaning of vector $d - P_0$, i.e. what is its relation to the hyperplane? Consider the four points, d, P_0, the origin, and the point that has $d - P_0$ as its position vector. What shape do they form in the case of $M = 2$? Illustrate your answer with a drawing.

For full credit, explain your answers in full.

QO2. Linear classifiers

Study chapter 14 of [MRS] and solve Exercise 14.15.

QO3. Regular expressions

This question is a review of regular expressions.

1. Write a program that makes a modified copy of a text file. In the copy, every string "Fred" (case-insensitive) should be replaced with "Larry". (So, "Manfred Mann" should become "ManLarry Mann".)
2. Modify the previous program to change every "Fred" to "Wilma" and every "Wilma" to "Fred". Now input like "fred&wilma" should look like "Wilma&Fred" in the output.
3. Write a program that prints out any input line that mentions both "wilma" and "fred" (case insensitive).
4. Write a program that counts the number of occurrences of the word "fred" in text. If the given word occurs as a part of a longer word, by having a continuation on left or right with other letters or digits, this occurrence should not be counted. For example the occurrences such as "She loves fred", "wilma-fred", or "fred fred" should be counted, but not "alfred", "fred11", "freds", "9fred", "Fred", "FRED".

In this question, you should read the text form input file (input.txt) and print out the results in output file (output.txt).

Study section 3.4 of `http://www.nltk.org/book/ch03.html`

Create short input.txt files to test your programs above. Submit the input.txt and the corresponding output.txt files.

Reference: Speech and Language Processing by Daniel Jurafsky and James H. Martin, Prentice-Hall, Inc., 2008, ISBN 978-0-13-187321-6.

Additional exercises will be posted up to one week before the due date. It is strongly recommended to start working on the assignment as soon as possible, and to upload solved questions as you complete them (as successive versions of your submission), as per instructions below.

Instructions for submitting the assignment.

Your assignment should have a cover page with the following information:

> CSCI 3151 (winter 2016)
> Assignment X (where X = 1, 2, ...)
> Last name
> First name
> Banner ID
> CS username

The file that you submit for assignment X should be named as:
`CSusername-aX.pdf` If you submit a supplementary file on a specific question Y, name it as:
`CSusername-aX-qY.pdf`

Grading: Each question will be graded with a letter grade, based on content (0.7), and quality of writing / neatness (0.3). If the assignment is so poorly written that content becomes difficult to understand with normal effort, then content mark will be further reduced.

Typesetting assignments is labour-intensive due to the mathematics. Therefore, typeset assignments are not required. If your handwriting is not perfectly legible, you may want to type the text, and fill in the math by hand. If you write your assignment by hand, please scan it for submission (use black marker pen for the handwritten parts, scan as black/white into a pdf file to minimize the file size).

The overall grade for the assignment will be a weighted average of the individual grades. Letter grades are being averaged using their GPA equivalent. No rounding takes place. The meaning of the grades is as per the University Calendar, section 17.1. This style of grading will be used in all evaluation components of this course.