6.047/6.878/HSPH IMI.231/HST.507 Fall 2018
Problem Set 2: Clustering and Classification

Due Monday, October 8 at 11:59pm (submit on the course website)

Submit a zip file on Stellar of a directory named `kerberos_ps2` containing:

- A PDF file named `kerberos_ps2.pdf` with your written answers, which should include all plots you are referencing.
- A directory named `code` with all the code you are submitting

Do not submit archive formats other than zip as Stellar does not handle them properly. In your answers to the questions please refer to the appropriate file name where your code for that problem is located. Unless skeleton code has been provided, feel free to use any programming language you are comfortable with, as long as you structure and comment your code to make it concise and legible.

1 Naive Bayes Classification

In this problem, we will use a Naive Bayes classifier to label fragments of the genome based on sequence properties.

(a) Suppose we want to classify sequence fragments into categories (represented by random variable Y): genes, regulatory motifs, or repetitive elements. We want to use the following features: length X_1, GC content (proportion of bases which are G or C) X_2, and *complexity* X_3 (intuitively, what fraction of possible k-mers are observed).

Does the naive Bayes assumption hold in this setting? Explain why or why not.

(b) Regardless of whether the naive Bayes assumption holds, we can still build a classifier. (Surprisingly, naive Bayes classifiers perform well in many applications where this assumption does not hold.) To simplify, we will discretize each of the features.

Given the training set below, write down the maximum likelihood estimates (recall these are relative frequencies) of each of the conditional probability distributions $P(X_i \mid Y)$ and the prior probability distribution $P(Y)$.

GC Content	Length	Complexity	Class
Low	Long	High	Gene
Low	Long	Low	Gene
High	Long	High	Repeat
Medium	Short	High	Motif
Medium	Short	Low	Motif
High	Long	Low	Repeat
High	Short	High	Motif
Medium	Long	High	Gene
High	Long	Low	Repeat
High	Short	High	Motif

(c) Given the model, compute the maximum a posteriori estimate of the class of the new observation below. (Hint: Is it necessary to compute the denominator in Bayes Theorem?)

GC Content	Length	Complexity
Medium	Long	Low

2 Classification of conserved regions

In this problem, we will use simulation to study the problem of classifying conserved sequence fragments given multiple alignments of four species. Submit all code you write.

(a) To simplify our classification problem, we will consider *alignment scores* at each position. We define the alignment score of a column of a multiple alignment to be the number of unique pairs that share the same symbol. An example multiple alignment and the score for each column is given below:

```
GACTA
TACTA
AGTTA
CTTAA
-----
01236
```

Consider two models C for conserved regions and N for unconserved regions. Assuming the alignment score at every position is independent, the conditional probability of observing a particular score in a column given each model is tabulated below:

Score	N	C
0	0.1	0.05
1	0.35	0.15
2	0.25	0.2
3	0.2	0.3
6	0.1	0.3

Compute the conditional probabilities of observing each of the following alignments given each of the models:

```
ACGACGACTA
CAGACGCTGA
TTCCTCTGAT
AGATGTGACT
```

```
ACAACGAGTA
AAAACGAATA
TCATCGAGTT
ACATCTAACT
```

(b) Simulate 10,000 sequences S of alignment scores of length 10 from N. How often is $\mathrm{P}(\mathrm{S} \mid \mathrm{C}) > \mathrm{P}(\mathrm{S} \mid \mathrm{N})$?

(c) Simulate 10,000 sequences S of alignment scores of length 10 from C. How often is $\mathrm{P}(\mathrm{S} \mid \mathrm{N}) > \mathrm{P}(\mathrm{S} \mid \mathrm{C})$?

(d) One way to reduce the rate of classification errors on short fragments is to favor using scores that are better at discriminating between the two models. Please provide:

i a pair of score values that is good at discriminating between the two models and

ii a pair of score values that is not good at discriminating between the two models.

Would the rate of classification errors decrease if we dismissed any alignment (column) with a score of 0?

(e) How could we reduce the rate of classification errors for much longer sequences?

3 K-means clustering

In this problem, you will implement k-means clustering on the expression profiles of two genes across a set of breast cancer patients. We have collected expression data from a pair of tissue types from the same set of 700 patients. We now wish to find clusters in this data that correspond to different breast cancer subtypes.

To run the code in this problem, you will have to either install R on your computer (visit `https://cran.r-project.org/`) or **run the command** `add R` **if you are running the code on Athena**. The kmeans zipped folder available through the Materials tab on the Stellar course website contains the following files:

- kmeans.py, which contains skeleton functions you will have to implement
- kmeans_plot.R, which plots the k-means clusters at each iteration of the algorithm (don't mess with this code unless you know your way around R and want to make your plots look prettier)
- a set of tissue$*$_data.txt files, which contain gene expression data from 700 patients on a series of tissues

(a) Your first task is to add code to kmeans.py to implement the k-means algorithm. To do this, you will have to complete the assignPoints and recalculateCtrs functions, and then add calls to these functions to the main() function in kmeans.py where indicated. Submit your version of kmeans.py.

(b) Run your code on tissue1 using the command `python kmeans.py tissue1`. If your implementation is correct, the algorithm will converge in four steps. Submit the plots generated by the code.

(c) Now run your code on tissue2 (your algorithm should converge in six steps this time). What went wrong? What strategy would you employ to find the settings of the algorithm so that it identifies the most obvious clusters, assuming you couldn't see the clusters ahead of time?

Experiment with the code in main() to try different approaches. Use insights from your strategy to make corresponding changes to the main() function in kmeans.py (you should only have to tinker with one line). Run the algorithm again, and describe how your solution addressed the problem using some of the output plots for reference.

Hand in your write-up, with the figures in the same document if possible.

(d) Describe how you would implement fuzzy k-means clustering using the above set of functions. You can either submit another version of kmeans.py with your changes, or submit pseudocode that has the same structure. Don't worry about running the fuzzy k-means or generating any more plots. (Bonus: Describe how you would visualize the steps of fuzzy k-means, showing the cluster(s) each point has been assigned to and each of the centroids as above.)

4 Final project preparation

(a) Before reading further, summarize each of your top three project ideas in a few sentences. You do not need to submit these.

(b) **Evaluate previous proposals.** We have made proposals from previous years available on the course website. Find the two proposals most closely related to your research interests. What do you find most exciting about the proposal? What would you do differently for that proposal? What aspects of the area did the proposal leave unaddressed, and how would you address them?

We expect your projects to be independent of these past projects; however, thinking critically about related work will help guide your own projects.

(c) **Evaluate scientific papers.** Find two papers published in the last several years closely related to your research interests. What do you find most exciting about them? What questions have they left open? What details did they not address, and how would you address them?

We suggest looking at journals such as *Nature Genetics*, *Nature Biotechnology*, *PLoS Computational Biology*, *PLoS Genetics*, *Bioinformatics*, and *BMC Bioinformatics*. We don't expect your projects to be submission-ready immediately upon completion, but critically reading the literature will give you a better sense of open problems in your area of interest and prior work you can build upon.

(d) **Write up a project idea.** Return to the ideas you wrote down in (a) in light of what you have read in previous proposals and the literature. Write up at least one of your ideas in detail (several paragraphs). How you would execute the idea? What challenges do you anticipate? What resources do you need to gather to succeed?

(e) **Find a partner.** Read through the student profiles (available on the course website) to find potential partners with similar interests and/or complementary skills. List the three people you are most likely to contact in order to form a team.

You don't have to contact them in advance, or get their approval for listing them here. We encourage you to contact potential collaborators early.

5 Hidden Markov Model classification of CpG islands (6.878 only)

In this problem, we will implement the eight-state Hidden Markov model described in lecture to annotate regions as CpG islands, or regions with high CpG dinucleotide frequency. Recall the model has states A^+, C^+, G^+, T^+ which emit nucleotides within CpG islands and states A^-, C^-, G^-, T^- which emit nucleotides outside CpG islands. Submit all code you write.

(a) Train the model by computing the maximum likelihood estimates of the model parameters (recall these are relative frequencies). Describe and justify how you handle zeroes in the estimated parameters. Estimate the initial state distribution and justify the method you used to do so.

Submit plain text files containing the transition probability matrix (space-separated entries, one row per line), the emission probability matrix (one row per state, one column per nucleotide; space-separated entries, one row per line) and the initial state distribution (one entry per line). (Hint: you may find that you need to do a *very* small amount of smoothing on the values in your emission matrix in order to avoid calculating the logarithm of 0.)

The training data is the sequence of human chromosome 21 and an existing CpG island annotation which we will use as ground truth (available at the URLs below):

`ftp://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr21.fa.gz`

`ftp://hgdownload.cse.ucsc.edu/goldenPath/hg19/database/cpgIslandExt.txt.gz`

The schema for the CpG island annotation database table is available at the URL given below. You will only need columns 2–4 ("chrom", "chromStart", "chromEnd").

`ftp://hgdownload.cse.ucsc.edu/goldenPath/hg19/database/cpgIslandExt.sql`

(b) Use the Viterbi algorithm to annotate CpG islands in the region surrounding the SRY (sex determining region Y)-box 10 gene (*SOX10*). We are interested in the region between positions 38,000,000–39,000,000 of human chromosome 22. The reference sequence of chromosome 22 is available at the URL below:

`ftp://hgdownload.cse.ucsc.edu/goldenPath/hg19/chromosomes/chr22.fa.gz`

Submit a BED file with your annotated regions. You only need to include chromosome and position information. The BED file format specification is available at the URL below:

`https://genome.ucsc.edu/FAQ/FAQformat.html#format1`

(c) Evaluate the performance of the model by computing its false positive and false negative rates on the test data against the ground truth annotation. Define a true positive to be a predicted CpG island of which at least 50% overlaps a true CpG island.

Are these two rates equal? If not, what causes this bias?

(d) Could we improve the performance of the model by tuning parameters? If so, describe how you would do so (you do not have to implement your suggestions). If not, describe and justify some modifications to the model which could reduce its error rate.

(e) One change to your model that you might consider is to add more states to it. Nevertheless, the number of states will still be finite. Explain why this limits the model's ability to capture phenomena that span arbitrarily large sequences.

(f) One alternative approach to improve the quality of annotations is combining multiple lines of evidence. Describe and justify some biological criteria for filtering the output of a sequence-based CpG island classifier to improve its performance.