problem set no. 5 — due tuesday 11/18 by 11:59 pm.

problem 1. the task is to partially reproduce the analyses in Cao et al. (JASA, 2000).

goals. the goal of this pset is for you to get familiar with derivations and implementation of the expectation-maximization algorithm, in the context of an interesting applied problem.

problem background. in a communication network, routers and switches connect subnetworks of users. see figures 1 and 8 in the paper. we can measure traffic (packet counts) on ingoing and outgoing links at each router, every five minutes. these measurements are referred to as *link loads*. we want to infer the traffic on all the possible origin-destination (OD) routes, every 5 minutes. these non-observable quantities of interest are referred to as *OD flows*. the main technical issue in this problem is in that the number of link loads at each time t is smaller than the number of corresponding OD flows. in the example network of figure 1b, for instance, there is a router with four subnetworks of users; this situation leads to 8 link loads (the measurements) and 16 OD flows (the latent variables), every five minutes. we will need to make (smoothness) assumptions to get around the main issue.

overview. you will have to implement two EM algorithms for making inference on the OD counts. the two algorithms correspond to the two models described in Sections 3.1 and 4.

question 1.1. (2 point) plot the link loads in file `1router_allcount.dat`, and replicate figure 2. include the axis labels.

question 1.2. (3 point) replicate figure 4 using the link loads in file `1router_allcount.dat`. produce a similar figure using the link loads in file `2router_linkcount.dat`. include the axis labels.

question 1.3. (10 point) derive the EM algorithm for the iid model of section 2.1. (see calculations on pages 1066–1067, for guidance.) most of the details are given in the paper, but should lay out every step explicitly and fill in any gaps. show explicit derivations of the E-step and M-step updates and present a concrete step-by-step algorithm after your derivation (either in pseudocode or similar step-by-step instructions). do not assume any specific value for c or w in your calculations.

question 1.4. (25 point) the EM algorithm derived for the iid model can be used to fit the locally iid model. write down explicitly how to do this. implement the EM algorithm for the locally iid model, use it to fit the link loads in file `1router_allcount.dat`, and replicate figure 5. (here $c = 2$ and $w = 11$.)

question 1.5. (10 point) derive the EM algorithm for the refined model of section 4. (see calculations on pages 1070–1072, for some guidance.) again, show explicit derivations of

the E-step and M-step updates and describe your algorithm step-by-step. do not assume any specific value for c or w in your calculations.

question 1.6. (25 point) fit the refined model using EM to the link loads provide in the file `1router_allcount.dat`, and replicate figure 6 —using partial output obtained in the previous question. include the axis labels.

question 1.8. (10 point) explain the difference in performance between the method of Tebaldi & West (1998) and those of Cao et al. (JASA, 2000); why should one method outperform the other?

question 1.9. (15 point) fit the locally IID and the refined models using EM to the link loads in file `2router_linkcount.dat`, and again, plot the equivalent of figure 6 for this dataset including the axis labels.

EXTRA CREDIT

question 1.10 (25 point) compute the estimates $\hat{x}_t$ for both datasets, under both models using the iterative proportional fitting algorithm described in section 2.4. using the resulting estimates, recreate figure 7 and figure 9, again include the axis labels. additionally, for figure 9: superimpose the OD flows estimates obtained using the locally IID model.

* * *

what to submit. submit 1 zip file. the zip file should contain: the `R` and `slurm` scripts to run the EM algorithm on Odyssey (5 files); the 6 figures described (7/8 files); a latex (or word) document with your answers to the questions (1 file).

please read the following instructions carefully and make sure your submission conforms to the guidelines. this is a computing exercise; if your code does not run it's a problem. any submitted homeworks that do not conform to the specifications will lose points.

please submit written answers (using latex or word) to all relevant questions via email by 11:59 pm on tuesday 11/18. email a ZIP file named as `myfasusername_ps5.ZIP` to stat221.harvard.fall2014@gmail.com (where `myfasusername` is replaced by your actual FAS username). Make sure to include "stat221" in the subject of your email. the ZIP file should contain three `R` scripts:

1. `myfasusername_functions.R`,
2. `myfasusername_1router.R`, and,

3. `myfasusername_2router.R`,

and your slurm scripts:

4. `myfasusername_1router.slurm`, and,
5. `myfasusername_2router.slurm`.

your scripts must run on Odyssey and produce the plots as described above, **as always, assuming all input/output files are in the same folder**.

- the file `myfasusername_functions.R` should contain the functions used to fit the EM algorithm. these functions must be named and have the following required arguments (although extra arguments are permitted as long as they have defaults):
  - `locally_iid_EM <- function(data, c, A){ # Write the function }`
  - `smoothed_EM <- function(data, c, A){ # Write the function }`
- the file `myfasusername_functions.R` should be loaded in by `myfasusername_1router.R` and `myfasusername_1router.R` using:

  ```
  source('myfasusername_functions.R')
  ```

- the figures should be named after the corresponding figure names from the paper as follows:
  1. `myfasusername_fig2.pdf`,
  2. `myfasusername_fig4_1router.pdf`,
  3. `myfasusername_fig4_2router.pdf`,
  4. `myfasusername_fig5.pdf`,
  5. `myfasusername_fig6_1router.pdf`,
  6. `myfasusername_fig6_2router.pdf`,
  7. `myfasusername_fig7.pdf`,
  8. `myfasusername_fig9.pdf` (extra credit only).
- the ZIP file should contain a PDF file named as `myfasusername_ps5.pdf`, as well as the TEX or DOC file with the solution, named as `myfasusername_ps5.tex` or `myfasusername_ps5.doc`.

if you have any questions about code formatting and organization then please ask Panos. happy coding!