

### Lab 3

#### CO902 – Probabilistic and statistical inference – 2012-13 Term 2

Lecturer: Tom Nichols

Finish last week's problem...

1. A *Markov chain* is a sequence of RVs  $X_i$ , having the property that the distribution of each RV given all the “previous” ones is equal to the distribution of the RV given only its immediate predecessor (*first order Markov assumption*). That is,

$$P(X_i | X_1 \dots X_{i-1}) = P(X_i | X_{i-1}).$$

Consider the case where all RVs are binary, such that  $X_i = \{0, 1\}$ . Then, the conditional probabilities  $P(X_i | X_{i-1})$  can be written as a *transition matrix*  $\mathbf{T}$ , with  $T_{jk} = P(X_i = k | X_{i-1} = j)$ . The marginal distribution  $P(X_1)$  is called the *initial distribution* and is denoted by  $P_0$ . In this problem you will simulate, estimate and predict with this Markov chain model.

- a. Write a program *simulate\_markov\_chain* which given initial distribution  $P_0$ , transition matrix  $\mathbf{T}$ , and sample size  $n$ , generates  $n$  samples from the chain.
- b. Write a program *estimate\_transition\_matrix*, which given data  $X = [X_1 \dots X_n]$ ,  $X_i = \{0, 1\}$  returns an estimate of the transition matrix  $\mathbf{T}$ . (It is OK to just return a subset of  $\mathbf{T}$ , as long as  $\mathbf{T}$  can be uniquely determined from what you return).
- c. Contrast the ability of the Bernoulli MLE and *estimate\_transition\_matrix* to predict an unseen observation generated from a Markov chain. That is, at each of a range of sample sizes:
  - i. Generate data using *simulate\_markov\_chain* (with uniform initial distribution  $P_0$  and a symmetric transition matrix with entries  $T_{jk} = 0.9$  for  $j=k$ , and 0.1 otherwise).
  - ii. Apply the two estimators to *all but the last observation*.
  - iii. Use your parameter/transition matrix estimates to *predict* this unseen observation.
  - iv. Compare the prediction with the truth to see how often it's correct.

Doing all this a large number of times will reveal the average prediction error of the two approaches. What happens to the error rate of the iid Bernoulli approach as sample size  $n$  grows? What happens to the Markov Chain method? Can you suggest why?

2. Download the file *spam\_data.mat* from the website. This contains a data vector  $X$  where each component corresponds to an email in a database.  $X_i = 1$  signifies that the email in question contained the word “free”. Each of these 1000 emails was hand-labelled by a human as spam or not-spam; a second vector “label” contains these labels, with  $label_i = 1/0$  signifying that email  $i$  was determined to be spam/not-spam. Assuming  $P(\text{spam}) = P(\text{non spam}) = 0.5$ , what do you need to estimate from the dataset  $X$  in order to be able to make a prediction regarding the spam status of a new email?
  - a. Write a program *learn\_spam\_params* that learns the required parameters.
  - b. Write a second program *predict\_spam* that, given a data vector  $X_{\text{new}}$ , returns a yes/no prediction for each email message in  $X_{\text{new}}$ . In addition to a prediction, give a probability expressing the confidence of your classification.
  - c. Split the 1000 observations into “train” and “test” samples. Apply *learn\_spam\_params* to the training samples to learn the parameters, and then apply *predict\_spam* to the test data and measure the prediction accuracy. For comparison, also run *predict\_spam* with the training samples. Which has higher accuracy?
3. [If time]. Revisit problem 1 of Lab 2, using the Bayesian MAP estimator we just learned about. The “add one success, one failure” estimator considered in Lab 2's 1.b can be seen as the MAP estimator with  $\alpha, \beta=2$ . Try different beta priors, and see how MAP compares with the MLE at various sample sizes, say  $n=[1:9 \ 10:10:90 \ 100:100:1000]$ . In particular, see what happens when the prior is flat (1,1) or plausible (2,2) vs. when it is

really badly chosen (say (10,10) with  $\theta=0.8$ , or even worse (3,7)). Plot sample size on a log scale so you can see what's going on. Do you see a general pattern?