Three projects for DD2447 Statistical Methods in Applied Computer Sci
ence fall 2014.

Below three projects can be found: (i) The train, (ii) The magic word, and (iii) your own project (which you only can do if we already have given your project an OK).
The deadline is 150118 at 19.00. Please email your solutions to me and Kristoffer. If you want to discuss a project that you have formulated, please contact us. Also, if we want to discuss your solutions, we will contact you. So make sure to read your email.

You should implement a generator and an inference algorithm. (The inference algorithm should use the methodologies described during the course Particle Filters, MCMC, etcetera.) Testing by generating and inferring as well reporting the results is mandatory.

Best,

Jens

Project 1: The train.
Imagine a model railway with a single train. You know the map of the tracks including the position of all the switches, but you don't know current states of the switches, or where the train is currently located. Each switch has three connections: 0,L,R. If the train comes from the direction of L or R, it always leaves in the direction 0. If the train comes from the direction 0, it will leave in either direction L or R, depending on the state of the switch. The switch has prior probability 1/2 for each direction, but will remain the same throughout the train run.
You are receiving a stream of signals from the train, each signal specifying the direction in which train has passed a switch: 0L, 0R, L0, or R0; you do not know, however, which switch the train has passed. Also, the sensors are noisy, and with a certain probability p, the train reports a random signal instead of a real direction in which it passed the switch.

We are given G, which is undirected and all vertices have degree 3. At each vertex the edges are labeled 0,L, and R (an edge can have different labels at different vertices). So a vertex is a switch. Start positions and switch settings have uniform priors.

A switch setting is a function σ: $V(G) \rightarrow \{L,R\}$, which has the natural interpretation. By a position we mean a pair (v,e), where $v \in V(G)$ and $e \in E(G)$, with the interpretation that the train has passed v and exited through e.

Below we give a DP algorithm for $p(s, O| G, \sigma)$. We will estimate $p(\sigma | G, O)$ using MCMC and then $p(s | G, O)$ using

$$p(s | G, O) = \sum_\sigma p(s, \sigma | G, O) = \sum_\sigma p(s | \sigma, G, O)\, p(\sigma | G, O)$$
$$= \sum_\sigma p(s, O | G, \sigma) p(\sigma | G, O) / p(O | G, \sigma)$$

The probability p is 0.05. Given G, σ, $s \in V(G)$ (s is a stop position), $O \in \{L,R,0\}^T$ (observed switch signals), we can compute $p(s, O | G, \sigma)$ using DP. The formulation actually induces a HMM. The natural way to compute this probability is to do DP and in each step compute the probability of going from some position s´ to s in t steps and observing $o_1,\ldots,o_t$ when the switch settings are σ. By doing this for all stop positions s and then summing out the stop position, we obtain $p(O| G, \sigma)$ in time $O(N^2T)$, where $N=|V(G)|$.

The states in our HMM are positions. The transition probabilities are always 1, i.e., given how we enter a vertex it is uniquely determined how we exit (since switches are fixed). Also, when passing a switch the correct direction of the label of the position is emitted with probability 1-p and any different direction is emitted with probability p/2.

Let c(s,t) be computed as below (we want c(s,t) to be the probability of going from some position s´ to s=(v,e) in t steps and observing $o_1,\ldots,o_t$). Let f=(u,v) and g=(w,v) be the two edges that are incident with v but different from e.

1. c(s,0) = 1/N
2. If e is labeled 0 and ot=0, then c(s,t) =[c((u,f),t-1)+c((w,g),t-1)] (1-p).
3. If e is labeled 0 and ot≠0, then c(s,t) =[c((u,f),t-1)+c((w,g),t-1)] p.
4. If e is labeled L, the switch at v is set to L, ot=L, and f is labeled 0 at v (i.e., w.l.o.g. assume the latter), then c(s,t) =c((u,f),t-1)(1-p)
5. If e is labeled L, the switch at v is set to L, ot≠L, and f is labeled 0 at v, then c(s,t) =c((u,f),t-1)p
6. If e is labeled R, the switch at v is set to R, ot=R, and f is labeled 0 at v, then c(s,t) =c((u,f),t-1)(1-p)
7. If e is labeled R, the switch at v is set to R, ot≠R, and f is labeled 0 at v, then c(s,t) =c((u,f),t-1)p
8. If e is labeled L, the switch at v is set to R, then c(s,t) = 0
9. If e is labeled R, the switch at v is set to L, then c(s,t) = 0

Project 2: The magic word
In this project you are supposed to implement a generator and a Gibbs sampler for the magic word model described below. You are also supposed to generate data and estimate the posterior using your Gibbs sampler. Finally, you are supposed to provide evidence that you have succeeded (plots showing convergence etcetera). The core observations necessary to design the Gibbs sampler can be found below.

The following generative model magic word model generates N sequences of length M: s_1, …,s_N where $s_i = s_{i,1},\ldots,s_{i,M}$. All sequences are over the alphabet [K]. Each of these sequences has a “magic” word of length w hidden in it and the rest of the sequence is called background.

First, for each i, a start position r_i for the magic word is sampled uniformly from [M-w+1]. Then the j:th positions in the magic words are sampled from $q_j(x)$, which is $\mathrm{Cat}(x|\theta_j)$ where θ_j has a $\mathrm{Dir}(\theta_j|\alpha)$ prior. All other positions in the sequences are sampled from the background distribution q(x), which is $\mathrm{Cat}(x|\theta)$ where θ has a $\mathrm{Dir}(\theta|\alpha')$ prior.

We no describe a Gibbs sampler that can be used for estimating the posterior over start positions after having observed $s_1,\ldots,s_N$. The sampler is collapsed and we do know α and α´.

We are interested in the posterior $p(r_1,\ldots,r_N \mid D)$ where D is a set of sequences $\{s_1,\ldots s_N\}$ generated by the model and r_i is the start position of the magic word in the i:th sequence si.

We will estimate this posterior sing a Gibbs sampler where the states are vectors of start positions $(r_1,\ldots,r_K)$.

Notice that, for the Dirchlet-categorical (Dirchlet-multinoulli) distributions for the j:th position of the magic words, the marginal likelihood is

$$p(D) =[\Gamma(\sum_k \alpha_k)/\Gamma(K + \sum_k \alpha_k)] \prod_k \Gamma(N_k + \alpha_k)/\Gamma(\alpha_k) \qquad (1)$$

where N_k is the count of symbol k in the j:th column of the magic words. For the background, the marginal likelihood is

$$p(D) =[\Gamma(\sum_k \alpha'_k)/\Gamma(B + \sum_k \alpha'_k)] \prod_k \Gamma(B_k + \alpha'_k)/\Gamma(\alpha'_k) \qquad (2)$$

where B is the number of background positions (i.e., K(N-w)) and B_k the count of symbol k in the background.

Let R= $(r_1,\dots,r_K)$. The full conditional $p(r_i \mid R_{-i}, D)$ can be expressed as follows

$$p(r_i \mid R_{-i}, D) = p(R,D)/p(R_{-i},D) \propto p(R,D) = p(D \mid R)p(R) \propto p(D \mid R)$$
$$= p(\text{background} \mid R) \prod_j p(\text{ column } j \mid R).$$

Clearly, the last expression can be computed using (1) and (2).

Project 3: Your own project.

Unless you already received an ok for this it is now to late to do this project. Formulate an interesting generative model. Implement a generator and an algorithm for estimating ML or posterior, based on methodologies from the course. You are also supposed to generate data and apply your algorithm. Finally, you are supposed to provide evidence that you have succeeded (plots showing convergence etcetera).