KTH Computer Science
and Communication

Implementation of “The Train” problem

Statistical Methods in Applied Computer Science, DD2447
Final Project

JIM HOLMSTRÖM JIMHO@KTH.SE
JOHAN GARDELL JGAR@KTH.SE

Teacher: Jens Lagergren

Contents

Chapter 1

The Problem

1.1 Problem formulation [1]

G is undirected and all vertices has 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 $\sigma : V(G) \to \{L, R\}$, 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:

$$p(s|G,O) = \sum_{\sigma} p(s,\sigma|G,O) = \sum_{\sigma} p(s|\sigma,G,O)p(\sigma|G,O) = \sum_{\sigma} \frac{p(s,G,O|\sigma)p(\sigma|G,O)}{p(G,O|\sigma)} \tag{1.1}$$

The probability p is 0.05. Given $G, \sigma, s \in V(G)$ (s is a stop position), $O \in \{L, R, 0\}^T$ (observed switch signals), we can compute $p(s, G, O|\sigma)$ using DP and in each step compute the probability of going from some position s' to s in t steps and observing $O_{1:t} = \{o_i\}_{i=1}^{t}$ 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 $\mathcal{O}(N^2 T)$, 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 $O_{1:t}$). Let $f = (u, v)$ and $g = (w, v)$ be the two edges that are incident with v but different from e.

[1] This formulation has some minor differences but it should still be equivalent with the original one.

$$c(s,t) = \begin{cases} 1/N, & t = 0 \\ \left[c((u,f),t-1) + c((w,g),t-1)\right](1-p) & e = 0 \wedge o_t = 0 \\ \left[c((u,f),t-1) + c((w,g),t-1)\right]p & e = 0 \wedge o_t \neq 0 \\ \left[c((u,f),t-1)\right](1-p) & e = L \wedge o_t = L \wedge f = 0 \\ \left[c((u,f),t-1)\right]p & e = L \wedge o_t \neq L \wedge f = 0 \\ \left[c((u,f),t-1)\right](1-p) & e = R \wedge o_t = R \wedge f = 0 \\ \left[c((u,f),t-1)\right]p & e = R \wedge o_t \neq R \wedge f = 0 \end{cases} \tag{1.2}$$

Chapter 2

Implementation

*The ignoring of G. email Jens bout it

*MCMC *basic basic MCMC abstract theory *MCMC used for estimating E[f(σ)]$without going through all \sigma which is intractable. * why is E[f(\sigma)] = sum f(\sigma) p(\sigma) =$
$sum_sampled f(\sigma) * how it relates to our problem * sigma graph with the weights Q(sigma, sigma') which 2^N stat$
$devil) * different algorithms for implementing MCMC * compare the different ones shortly (what they ass$
$explain why we chose it * how we chose Q what are the requirements for it * hamming distance measure and pick$
$why hamming? a more natural distance for switch settings than say euclidean or such. *$
$abstractly relate to the graph (what is this type of graph called?) * describe the marginalization over states in p($

Chapter 3

Results & Discussion