18.413: Error-Correcting Codes Lab February 24, 2004

Lecture 6

Lecturer: Daniel A. Spielman

6.1 Introduction

Begin by describing LDPC codes, and how they are described by many local constraints. Point out that random graphs locally look like trees (from the birthday paradox), and so we will learn to do belief propagation on trees. But first, we must learn to do BP on the simplest of trees: with just 2 and three nodes.

6.2 Two variables

We begin with a further examination of our fundamental formula in the case of just two variables. Let X_1 be a variable taking values in the alphabet A_1 and let X_2 be a variable taking values in the alphabet A_2. Then let $\mathcal{C} \in A_1 \times A_2$ be a code.

Assume that we choose $(X_1, X_2) \in \mathcal{C}$ uniformly at random, transmit over a channel, and receive (Y_1, Y_2).

We will show

Lemma 6.2.1.

$$P^{post}\left[X_1 = a_1 | Y_1 Y_2 = b_1 b_2\right] = c_{b_1,b_2} P^{prior}\left[X_1 = a_1\right] P^{ext}\left[X_1 = a_1 | Y_1 = b_1\right] P^{ext}\left[X_1 = a_1 | Y_2 = b_2\right],$$

where

$$P^{prior}\left[X_1 = a_1\right] = \frac{|\{a_2 : (a_1, a_2) \in \mathcal{C}\}|}{|\mathcal{C}|}$$

As we already know that

$$\mathrm{P}^{post}\left[X_1 = a_1 | Y_1 Y_2 = b_1 b_2\right] = c_{b_1,b_2} \mathrm{P}^{prior}\left[X_1 = a_1\right] \mathrm{P}^{ext}\left[X_1 = a_1 | Y_1 Y_2 = b_1 b_2\right],$$

so it suffices to prove

Lemma 6.2.2.

$$P^{ext}\left[X_1 = a_1 | Y_1 Y_2 = b_1 b_2\right] = c_{b_1,b_2} P^{ext}\left[X_1 = a_1 | Y_1 = b_1\right] P^{ext}\left[X_1 = a_1 | Y_2 = b_2\right].$$

Proof. We begin by examining the right-hand-sides. We have

$$\mathrm{P}^{ext}\left[X_1 = a_1 | Y_1 = b_1\right] = c_{b_1} \mathrm{P}\left[Y_1 = b_1 | X_1 = a_1\right] \tag{6.1}$$

and

$$\begin{aligned}
\mathrm{P}^{ext}\left[X_1 = a_1 | Y_2 = b_2\right] &= c_{b_2} \mathrm{P}\left[Y_2 = b_2 | X_1 = a_1\right] \\
&= c_{b_2} \sum_{a_2 : (a_1, a_2) \in \mathcal{C}} \mathrm{P}\left[Y_2 = b_2 | X_1 X_2 = a_1 a_2\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \\
&= c_{b_2} \sum_{a_2 : (a_1, a_2) \in \mathcal{C}} \mathrm{P}\left[Y_2 = b_2 | X_2 = a_2\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right]. \qquad (6.2)
\end{aligned}$$

Now, we examine the left-hand-side:

$$\begin{aligned}
\mathrm{P}^{ext}\left[X_1 = a_1 | Y_1 Y_2 = b_1 b_2\right] &= c_{b_1, b_2} \mathrm{P}\left[Y_1 Y_2 = b_1 b_2 | X_1 = a_1\right] \\
&= c_{b_1, b_2} \sum_{a_2 : (a_1, a_2) \in \mathcal{C}} \mathrm{P}\left[Y_1 Y_2 = b_1 b_2 | X_1 X_2 = a_1 a_2\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \\
&= c_{b_1, b_2} \sum_{a_2 : (a_1, a_2) \in \mathcal{C}} \mathrm{P}\left[Y_1 = b_1 | X_1 = a_1\right] \mathrm{P}\left[Y_2 = b_2 | X_2 = a_2\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \\
&= c_{b_1, b_2} \mathrm{P}\left[Y_1 = b_1 | X_1 = a_1\right] \sum_{a_2 : (a_1, a_2) \in \mathcal{C}} \mathrm{P}\left[Y_2 = b_2 | X_2 = a_2\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right].
\end{aligned}$$

To conclude, we observe that this last term is the product of (6.1) and (6.2).

□

6.3 Simplifying computations

6.4 Three Variables

We now consider the situation in which X_1, X_2 and X_3 lie in A_1, A_2 and A_3, and $(X_1, X_2) \in \mathcal{C}_{12} \subseteq A_1 \times A_2$ and and $(X_2, X_3) \in \mathcal{C}_{23} \subseteq A_2 \times A_3$. In particular, we will assume that (X_1, X_2, X_3) are chosen uniformly subject to this condition.

The variables (X_1, X_2, X_3) then satisfy what the book calls the "Markov" property. That is, for all a_1, a_2, a_3,

$$\mathrm{P}\left[X_1 X_3 = a_1 a_3 | X_2 = a_2\right] = \mathrm{P}\left[X_1 = a_1 | X_2 = a_2\right] \mathrm{P}\left[X_3 = a_3 | X_2 = a_2\right].$$

In this case, we can say that all the information that X_3 contains about X_1 is transmitted through X_2. This fact can be used to simplify the belief computation.

Lemma 6.4.1.

$$P^{ext}\left[X_1 = a_1 | Y_2 Y_3 = b_2 b_3\right] = \sum_{a_2 : (a_1, a_2) \in \mathcal{C}} P\left[X_2 = a_2 | X_1 = a_1\right] P^{ext}\left[X_2 = a_2 | Y_2 = b_2\right] P^{ext}\left[X_2 = a_2 | Y_3 = b_3\right].$$

That is, the computation of $\mathrm{P}^{ext}\left[X_1 = a_1 | Y_2 Y_3 = b_2 b_3\right]$ can be done in two stages: in the first we compute $\mathrm{P}^{ext}\left[X_2 = a_2 | Y_3 = b_3\right]$ for each a_2, and in the second we sum over the a_2s.

Proof of Lemma 6.4.1. We have

$$
\begin{aligned}
&\mathrm{P}^{ext}\left[X_1 = a_1 | Y_2 Y_3 = b_2 b_3\right] \\
&\sim \mathrm{P}\left[Y_2 Y_3 = b_2 b_3 | X_1 = a_1\right] \\
&= \sum_{a_2 : (a_1, a_2) \in \mathcal{C}_{12}} \mathrm{P}\left[Y_2 Y_3 = b_2 b_3 | X_1 X_2 = a_1 a_2\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \\
&= \sum_{a_2 : (a_1, a_2) \in \mathcal{C}_{12}} \sum_{a_3 : (a_2, a_3) \in \mathcal{C}_{23}} \mathrm{P}\left[Y_2 Y_3 = b_2 b_3 | X_1 X_2 X_3 = a_1 a_2 a_3\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \mathrm{P}\left[X_3 = a_3 | X_1 X_2 = a_1 a_2\right] \\
&= \sum_{a_2 : (a_1, a_2) \in \mathcal{C}_{12}} \sum_{a_3 : (a_2, a_3) \in \mathcal{C}_{23}} \mathrm{P}\left[Y_2 Y_3 = b_2 b_3 | X_2 X_3 = a_2 a_3\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \mathrm{P}\left[X_3 = a_3 | X_2 = a_2\right] \\
&= \sum_{a_2 : (a_1, a_2) \in \mathcal{C}_{12}} \sum_{a_3 : (a_2, a_3) \in \mathcal{C}_{23}} \mathrm{P}\left[Y_2 = b_2 | X_2 = a_2\right] \mathrm{P}\left[Y_3 = b_3 | X_3 = a_3\right] \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \mathrm{P}\left[X_3 = a_3 | X_2 = a_2\right] \\
&= \sum_{a_2 : (a_1, a_2) \in \mathcal{C}_{12}} \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \mathrm{P}\left[Y_2 = b_2 | X_2 = a_2\right] \sum_{a_3 : (a_2, a_3) \in \mathcal{C}_{23}} \mathrm{P}\left[Y_3 = b_3 | X_3 = a_3\right] \mathrm{P}\left[X_3 = a_3 | X_2 = a_2\right] \\
&\sim \sum_{a_2 : (a_1, a_2) \in \mathcal{C}_{12}} \mathrm{P}\left[X_2 = a_2 | X_1 = a_1\right] \mathrm{P}\left[Y_2 = b_2 | X_2 = a_2\right] \mathrm{P}^{ext}\left[X_2 = a_2 | Y_3 = b_3\right]
\end{aligned}
$$

□

6.5 Trees

A *hypergraph* is given by a collection of vertices $x_1, \ldots, x_n$ and a collection of edges $e_1, \ldots, e_m$, where each $e_i \subseteq \{x_1, \ldots, x_n\}$. A *path* in a hypergraph is a sequence of vertices $x_{i_1}, \ldots, x_{i_k}$ such that each consecutive pair in the sequence lie in some edge. That is, for each $2 \leq j \leq k$, there exists l such that $\left\{x_{i_{j-1}}, x_{i_j}\right\} \subseteq e_l$. A hypergraph is `connected` if for each pair of vertices there is at least one path connecting them.

A hypergraph is a *tree* if for each pair of vertices there is exactly one path connecting them. An equivalent definition is that for each i, if one replaces each edge e_l containing x_i by $e_l' = e_l - \{x_i\}$, then the hypergraph is disconnected.

If we have variables $X_1, \ldots, X_n$ chosen uniformly subject to constraints, each of which involves only the variables in an edge, and the corresponding hypergraph is a tree, then Lemma 6.4.1 can be extended to an algorithm for belief computation in the tree.