Kalman Filtering: A PGM implementation

SCJ Robertson

20th April, 2016

Introduction

It is become apparent there huge gaps in my understanding of both the Kalman Filter and PGM basics. This is just a set of legible notes which I hope will expose these misunderstandings.

1. Bayes Nets to Junction trees

Figure 1.1 is the Bayes Net for the Kalman Filter. The control vector, $\mathbf{u}_t = \mathbf{u}$, may be neglected in most cases as it is a constant. Each node is a Gaussian CPD with:

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t|A\mathbf{x}_{t-1} + B\mathbf{u}, R) \tag{1.1}$$
$$p(\mathbf{z}_t|\mathbf{x}_t) = \mathcal{N}(\mathbf{z}_t|C\mathbf{x}_t, Q) \tag{1.2}$$
$$p(\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0|\mu_0, \Sigma_0) \tag{1.3}$$

Figure 1.2 is the network's equivalent Junction Tree. This Junction Tree was created following the procedure given in the PGM: Week 5 notes. Moralization and triangulation shouldn't affect the structure and the induced graph should be an undirected equivalent of the Bayes Net. The maximal cliques will be allocated a single potential, corresponding to their CPDs in Equations 1.1 and 1.2.

1.1 Message passing: Integral-Product algorithm

I assume that the discrete message passing algorithm is replaced with:

$$\delta_{\phi_{\mathbf{x}_t} \to \phi_{\mathbf{x}_{t+1}}}(\mathbf{x}_i) = \int \phi_{\mathbf{x}_t}(\mathbf{x}_t, \mathbf{x}_{t-1}) \delta_{\mathbf{z}_t \to \mathbf{x}_t}(\mathbf{x}_t) \delta_{\mathbf{x}_t \to \mathbf{x}_{t+1}}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1} \tag{1.4}$$

I assme the idea is to use canonical form to ease the marginalization.

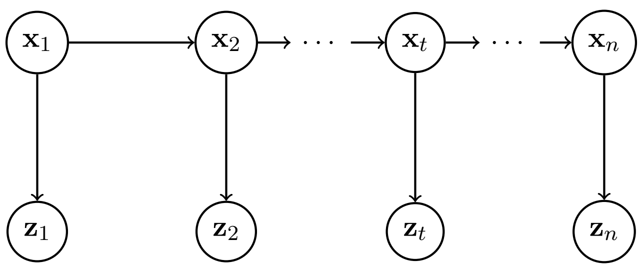

Figure 1.1: A Bayes Net for a linear Kalman filter.

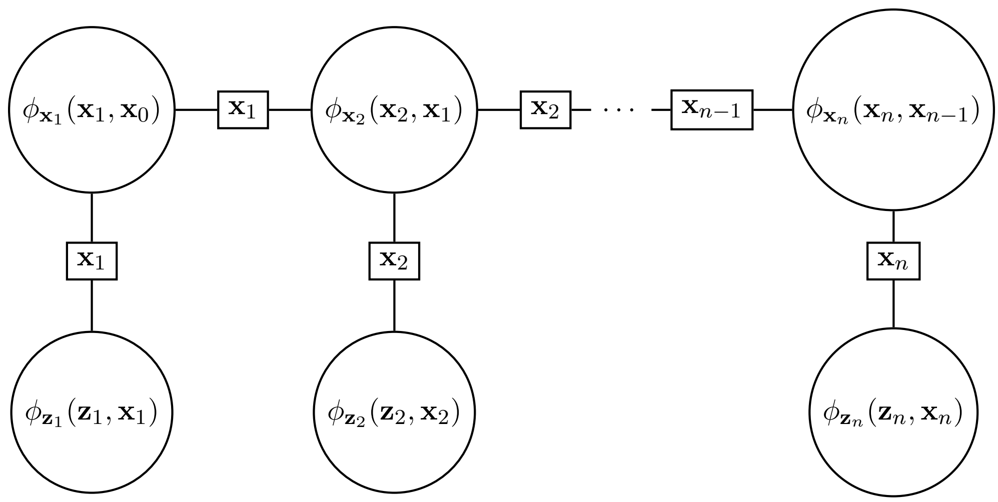

Figure 1.2: The junction tree resulting from the Bayes Net in Figure 1.1.