Conditional Chow-Liu Tree Structures for Modeling Discrete-Valued Vector Time Series

Technical Report UCI-ICS 04-04
Information and Computer Science
University of California, Irvine

Sergey Kirshner and Padhraic Smyth
School of Information and Computer Science
University of California, Irvine
Irvine, CA 92697-3425
{skirshne,smyth}@ics.uci.edu

Andrew W. Robertson
International Research Institute for Climate Prediction (IRI)
The Earth Institute at Columbia University
Monell 230, 61 Route 9W,
Palisades, NY 10964
awr@iri.columbia.edu

March 17, 2004
Revised: March 29, 2004

Abstract

We consider the problem of modeling discrete-valued vector time series data using extensions of Chow-Liu tree models to capture both dependencies across time and dependencies across variables. We introduce conditional Chow-Liu tree models, an extension to standard Chow-Liu trees, for modeling conditional rather than joint densities. We describe learning algorithms for such models and show how they can be used to learn parsimonious representations for the output distributions in hidden Markov models. We illustrate how these models can be applied to the important problem of simulating and forecasting daily precipitation occurrence for networks of rain stations. To illustrate the effectiveness of the models, we compare their performance versus a number of alternatives using historical precipitation data from Southwestern Australia and the Western United States. We illustrate how the structure and parameters of the models can be used to provide an improved meteorological interpretation of such data.

Keywords: Hidden Markov models, Chow-Liu trees, Precipitation

1 Introduction

In this paper we consider the problem of modeling discrete-time, discrete-valued, multivariate time-series. For example, consider M time-series where each time-series can take B values. The motivating application in this paper is modeling of daily binary rainfall data ($B = 2$) for a spatial network of M stations (where typically M can vary from 10 to 100). Modeling and prediction of rainfall is an important problem in the atmospheric sciences. A common application, for example, is simulating realistic daily rainfall patterns for a 90-day season, to be used as input for detailed crop-modeling simulations (e.g., Semenov and Porter 1995). A number of statistical methods have been developed for modeling daily rainfall time-series at single stations — first-order Markov models and various extensions (also known as "weather generators") have proven quite effective for single-station rainfall modeling in many geographic regions. However, there has been less success in developing models for multiple stations that can generate simulations with realistic spatial and temporal correlations in rainfall patterns (Wilks and Wilby 1999).

Direct modeling of the dependence of the M daily observations at time t on the M observations at time $t-1$ requires an exponential in M number of parameters. which is clearly impractical for most values of M of interest. In this paper we look at the use of hidden Markov models (HMMs) to avoid this problem — an HMM uses a K-valued hidden first-order Markov chain to model time-dependence, with the M outputs at time t being conditionally independent of everything else given current state value at time t. The hidden state variable in an HMM serves to capture temporal dependence in a low-dimensional manner, i.e., with $O(K^2)$ parameters instead of being exponential in M. From the physical point of view, an attractive feature of the HMM is that the hidden states can be interpreted as underlying "weather states" (Hughes et al. 1999, Robertson et al. 2003).

Modeling the instantaneous multivariate dependence of the M observations on the state at time t would require B^M parameters per state if the full joint distribution were modeled (which would defeat the purpose of using the HMM to reduce the number of parameters). Thus, approximations such as assuming conditional independence (CI) of the M observations are often used in practice (e.g., see Hughes and Guttorp 1994), requiring $O(KMB)$ parameters.

While the HMM-CI approach is a useful starting point it suffers from two well-known disadvantages for an application such as rainfall modeling: (1) the assumed conditional independence of the M outputs on each other at time t can lead to inadequate characterization of the dependence between the M time-series (e.g., unrealistic spatial rainfall patterns on a given day), (2) the assumed conditional independence of the M outputs at time t from from the M outputs at time $t-1$ can lead to inadequate temporal dependence in the M time-series (e.g., unrealistic occurrences of wet days during dry spells).

In this paper we investigate Chow-Liu tree structures in the context of providing improved, yet tractable, models to address these problems in capturing output dependencies for HMMs. We show how Chow-Liu trees can be used to directly capture dependency among the M outputs in multivariate HMMs. We also introduce an extension called conditional Chow-Liu trees to provide a class of dependency models that are well-suited for modeling multivariate time-series data. We illustrate the application of the proposed methods to two large-scale precipitation data sets.

The paper is structured as follows. Section 2 formally describes existing models and our extensions. Section 3 describes how to perform inference and to learn both the structure and parameters for the models. Section 4 describes an application and analyzes the performance of the models. Finally, Section 5 summarizes our contributions and outlines possible future directions.

2 Model Description

We begin this section by briefly reviewing Chow-Liu trees for multivariate data before introducing the conditional Chow-Liu tree model. We then focus on vector time-series data and show how the conditional Chow-Liu tree model and hidden Markov models can be combined.

2.1 Chow-Liu Trees

Chow and Liu (1968) proposed a method for approximating the joint distribution of a set of discrete variables using products of distributions involving no more than pairs of variables. If $P(\mathbf{x})$ is an M-variate distribution on discrete variables $V = \left(x^1, \ldots, x^M\right)$, the Chow-Liu method constructs a distribution $T(\mathbf{x})$ for which the corresponding Bayesian and Markov network is tree-structured. If $G_T = (V, E_T)$ is the Markov network associated with T, then

$$T(\mathbf{x}) = \frac{\prod_{(u,v)\in E_T} T(x^u, x^v)}{\prod_{v\in V} T(x^v)^{degree(v)}} = \prod_{(u,v)\in E_T} \frac{T(x^u, x^v)}{T(x^v)\,T(x^u)} \prod_{v\in V} T(x^v). \tag{1}$$

Widely used expression to measure how different one distribution is from another, the Kullback-Liebler divergence $KL(P,T)$ between distributions P and T is defined as

$$KL(P,T) = \sum_{\mathbf{x}} P(\mathbf{x}) \log \frac{P(\mathbf{x})}{T(\mathbf{x})}.$$

Chow and Liu showed that in order to minimize $KL(P,T)$ the edges for the tree E_T have to be selected to maximize the total mutual information of the edges $\sum_{(u,v)\in E_T} I(x^u, x^v)$ where mutual information between variables x^u and x^v is defined as

$$I(x^u, x^v) = \sum_{x^u}\sum_{x^v} P(x^u, x^v) \log \frac{P(x^u, x^v)}{P(x^u)\,P(x^v)}. \tag{2}$$

This can be accomplished by calculating mutual information $I(x^u, x^v)$ for all possible pairs of variables in V, and then solving the maximum spanning tree problem, with pairwise mutual information from Equation 2 as edge weights (e.g., Cormen et al. 1990). Once the edges are selected, the probability distribution T on the pairs of vertices connected by edges is defined to be the same as P:

$$\forall (x^u, x^v) \in E_T \quad T(x^u, x^v) = P(x^u, x^v),$$

and the resulting distribution T minimizes $KL(P,T)$. Figure 1 outlines the algorithm for finding T.

If each of the variables in V takes on B values, finding the optimal tree T has time complexity $O\left(M^2B^2\right)$ for the mutual information calculations and $O\left(M^2\right)$ for finding the minimum spanning tree, totalling $O\left(M^2B^2\right)$. (For the case of sparse high-dimensional data, Meilă (1999) showed that the Chow-Liu algorithm can be sped up.) In practice, P is often an empirical distribution on the data, and thus calculation of pairwise counts of variables (used in mutual information calculations) has complexity $O\left(NM^2B^2\right)$ where N is the number of vectors in the data.

Algorithm ChowLiu(P)
Inputs: Distribution P over domain V; procedure MWST(weights) that outputs a maximum weight spanning tree over V

1. Compute marginal distributions $P(x^u, x^v)$ and $P(x^u)$ $\quad \forall u, v \in V$
2. Compute mutual information values $I(x^u, x^v)$ $\quad \forall u, v \in V$
3. $E_T = \text{MWST}(\{I(x^u, x^v)\})$
4. Set $T(x^u, x^v) = P(x^u, x^v) \quad \forall (u, v) \in E_T$

Output: T

Figure 1: Chow-Liu algorithm (very similar to Meilă and Jordan 2000)

The advantages of Chow-Liu trees include (a) the existence of a simple algorithm for finding the optimal tree [1], (b) the parsimonious nature of the model (the number of parameters is linear in dimensionality of the space), and (c) the resulting tree structure T often has a simple intuitive interpretation. While there are other algorithms that retain the idea of a tree-structured distribution, while allowing for more complex dependencies (e.g., thin junction trees, Bach and Jordan 2002), these algorithms have higher time complexity than the original Chow-Liu algorithm and do not guarantee optimality of the structure that is learned. Thus, in the results in this paper we focus on Chow-Liu trees under the assumption that they are a generally useful modeling technique in the context of multivariate time dependent data.

2.1.1 Mixtures of Trees

Meilă and Jordan (2000) proposed a richer class of structures by describing a mixture model with Chow-Liu tree distributions as the mixture components. A probability distribution on an M-dimensional set X is defined as

$$P(\mathbf{x}) = \sum_{i=1}^{K} P(z = i)\, T_i(\mathbf{x}) \;\; \forall \mathbf{x} \in \mathbf{X}$$

where the latent variable z indicates the component of a mixture, and $T_1, \ldots, T_K$ are component probability distributions with a Chow-Liu tree structure for each mixture component. The tree structures $T_1, \ldots, T_K$ can be constrained to be the same or allowed to differ. Meilă and Jordan (2000) also describe how to perform inference with this model, and how to learn both the structure and the parameters using the EM algorithm.

2.2 Conditional Chow-Liu Forests

It is common in practice (e.g., in time-series and in regression modeling) that the data to be modelled can be viewed as consisting of two sets of variables, where we wish to model the conditional distribution $P(\mathbf{x}|\mathbf{y})$ of one set $\mathbf{x}$ on the other set $\mathbf{y}$. We propose an extension of the Chow-Liu

[1]In fact, if we change the structure to allow cliques of size more than 2 in the graph G_T, the problem of finding optimal approximation distribution becomes NP-hard (Chickering 1996, Srebro 2003).

Algorithm CondChowLiu(P)
Inputs: Distribution P over domain $V_x \cup V_y$; procedure MWST(V, weights) that outputs a maximum weight spanning tree over V

1. (a) Compute marginal distributions $P(x^u, x^v)$ and $P(x^u)$ $\forall u, v \in V_x$
   (b) Compute marginal distributions $P(y^u)$ and $P(y^u, x^v)$ $\forall u \in V_y, v \in V_x$
2. (a) Compute mutual information values $I(x^u, x^v)$ $\forall u, v \in V_x$
   (b) Compute mutual information values $I(y^u, x^v)$ $\forall u \in V_y, v \in V_x$
   (c) Find $u(v) = \arg\max_{u \in V_y} I(y^u, x^v)$ $\forall v \in V_x$
   (d) Let $V' = V_x \cup \{v'\}$, and set $I\left(x^{v'}, x^v\right) = I\left(y^{u(v)}, x^v\right)$ $\forall v \in V_x$
3. (a) $E_{T'} = \text{MWST}(V', \mathbf{I})$
   (b) $E_x = \{(u,v) \mid u, v \in V_x, (u,v) \in E_{T'}\}$
   (c) $E_y = \{(u(v), v) \mid v \in V_x, (v, v') \in E_{T'}\}$.
4. (a) Set $T(x^u, x^v) = P(x^u, x^v)$ $\forall (u,v) \in E_x$
   (b) Set $T(y^u, x^v) = P(y^u, x^v)$ $\forall (u,v) \in E_y$

Output: T

Figure 2: Conditional Chow-Liu algorithm

method to model such conditional distributions. As with Chow-Liu trees, we want the corresponding probability distribution to be factored into a product of distributions involving no more than two variables. Pairs of variables are represented as an edge in a corresponding graph with nodes corresponding to variables in $V = V_x \cup V_y$. However, since all of the variables in V_y are observed, we are not interested in modeling $P(\mathbf{y})$, and do not wish to restrict $P(\mathbf{y})$ by making independence assumptions about the variables in V_y. The structure for an approximation distribution T will be constructed by adding edges such as not to introduce paths involving multiple variables from V_y.

Let $G_F = (V, E_F)$ be a forest, a collection of disjoint trees, containing edges E_x between pairs of variables in V_x and edges E_y connecting variables from V_x and V_y, $E_F = E_x \cup E_y$. If the probability distribution $T(\mathbf{x}|\mathbf{y})$ has G_F for a Markov network, then similar to Equation 1:

$$\begin{aligned} T(\mathbf{x}|\mathbf{y}) &= \prod_{(u,v)\in E_x} \frac{T(x^u, x^v)}{T(x^u)\,T(x^v)} \prod_{v \in V_x} T(x^v) \prod_{(u,v)\in E_y} \frac{T(y^u, x^v)}{T(y^u)\,T(x^v)} \\ &= \prod_{(u,v)\in E_x} \frac{T(x^u, x^v)}{T(x^u)\,T(x^v)} \prod_{v \in V_x} T(x^v) \prod_{(u,v)\in E_y} \frac{T(x^v|y^u)}{T(x^v)}. \end{aligned} \tag{3}$$

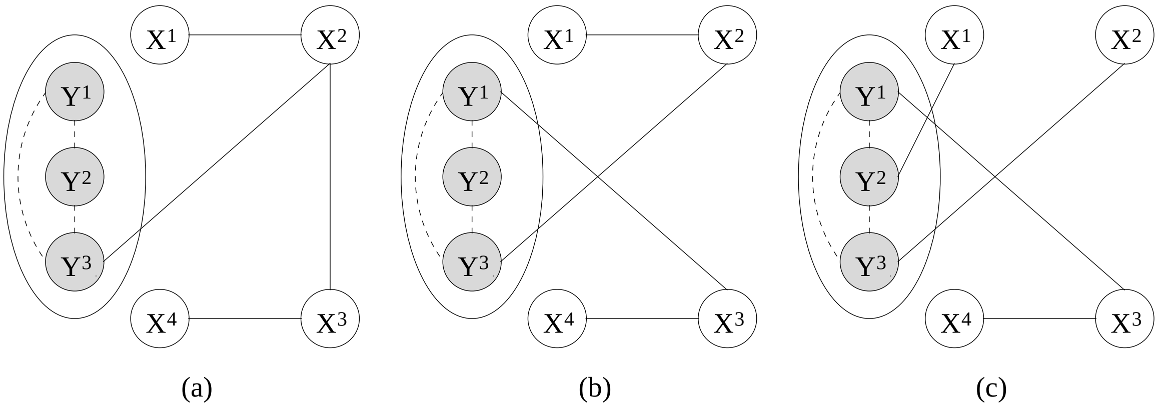

Figure 3: Conditional CL forest for a hypothetical distribution with (a) 1 component (b) 2 components (c) 3 components.

We will again use KL-divergence, this time between conditional distributions P and T, as an objective function:

$$KL(P,T) = \sum_{\mathbf{y}} P(\mathbf{y}) \sum_{\mathbf{x}} P(\mathbf{x}|\mathbf{y}) \log \frac{P(\mathbf{x}|\mathbf{y})}{T(\mathbf{x}|\mathbf{y})}.$$

It can be shown that the optimal probability distribution T with corresponding Markov network G_F

$$\forall (u,v) \in E_x,\ T(x^u, x^v) = P(x^u, x^v)$$

and

$$\forall (u,v) \in E_y,\ T(y^u, x^v) = P(y^u, x^v).$$

As with the unconditional distribution, we wish to find pairs of variables to minimize

$$KL(P,T) = \sum_{v \in V_x} H[x^v] - H[\mathbf{x}|\mathbf{y}] - \sum_{(u,v) \in E_x} I(x^u, x^v) - \sum_{(u,v) \in E_y} I(y^u, x^v)$$

where $H[x^v]$ denotes the entropy of $P(x^v)$, and $H[\mathbf{x}|\mathbf{y}]$ denotes the conditional entropy of $P(\mathbf{x}|\mathbf{y})$. Both $H[\mathbf{x}]$ and $H[\mathbf{x}|\mathbf{y}]$ are independent of E_F, so as in the unconditional case, we need to solve a maximum spanning tree problem on the graph with nodes $V_y \cup V_x$ while not allowing paths between vertices in V_y (alternatively, assuming all nodes in V_y are connected).

The algorithm for learning the conditional Chow-Liu (CCL) distribution is shown in Figure 2. More details about the algorithm are provided in Appendix A. Due to the restrictions on the edges, the CCL networks can contain disconnected tree components (referred to as forests). These CCL forests can consist of 1 to $\min\{|V_y|, |V_x|\}$ components. (See Figure 3 for an illustration.)

2.2.1 Chain CL Forests

We now return to our original goal of modeling time-dependent data. Let $\mathbf{R}_t = (R_t^1, \ldots, R_t^M)$ be a multivariate (M-variate) random vector of data with each component taking on values $\{0, \ldots, B-1\}$. By $\mathbf{R}_{1:T}$ we will denote observation sequence $\mathbf{R}_1, \ldots, \mathbf{R}_T$[2].

[2]The notation is overloaded as T denotes both the length of the sequence and an approximating tree distribution.

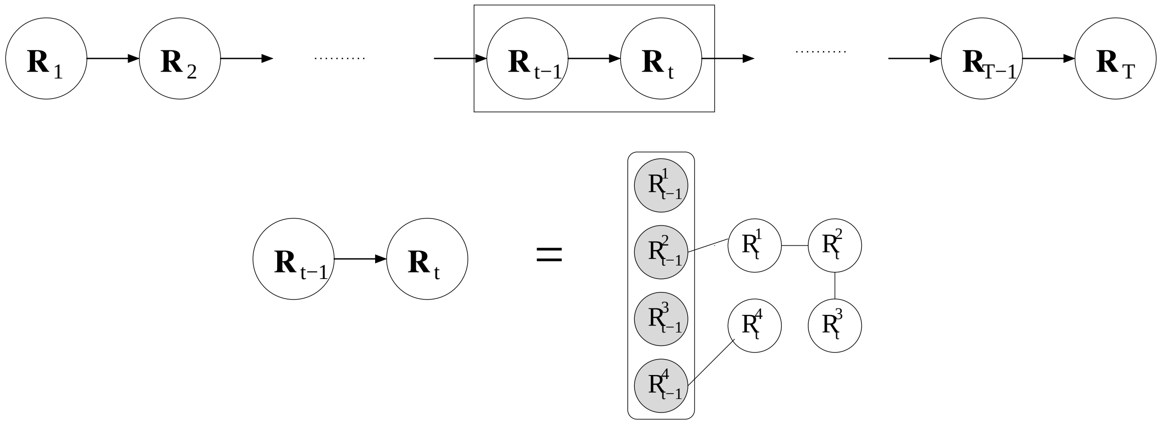

Figure 4: Graphical model for a hypothetical CCLF

A simple model for such data can be constructed using conditional Chow-Liu forests. For this chain Chow-Liu forest model (CCLF), the data for a time point t is modeled as a conditional Chow-Liu forest given data at point $t-1$ (Figure 4):

$$P\left(\mathbf{R}_{1:T}\right) = \prod_{t=1}^{T} T\left(\mathbf{R}_t|\mathbf{R}_{t-1}\right)$$

where

$$T\left(\mathbf{R}_t = \mathbf{r}|\mathbf{R}_{t-1} = \mathbf{r}'\right) =$$
$$= \prod_{(u,v)\in E_V} \frac{T\left(R^u_t = r^u, R^v_t = r^v\right)}{T\left(R^v_t = r^v\right) T\left(R^u_t = r^u\right)} \prod_{v\in R_t} T\left(R^v_t = r^v\right) \prod_{(u,v)\in E_{Ii}} \frac{T\left(R^v_t = r^v|R^u_{t-1} = r'^u\right)}{T\left(R^v_t = r^v\right)}.$$

Note that learning the structure and parameters of CCLF requires one pass through the data to collect appropriate counts and to calculate joint probabilities of pairs of variables, and one subsequent run of the CondChowLiu tree algorithm.

2.3 Hidden Markov Models

An alternative approach to modeling $\mathbf{R}_{1:T}$ is to use a hidden-state model to capture temporal dependence. Let S_t be the hidden state for observation t, taking on one of K values from 1 to K, where $S_{1:T}$ denotes sequences of length T of hidden states.

A first-order HMM makes two conditional independence assumptions. The first assumption is that the hidden state process, $S_{1:\mathsf{T}}$, is first-order Markov:

$$P\left(S_t|S_{1:t-1}\right) = P\left(S_t|S_{t-1}\right) \tag{4}$$

and that this first-order Markov process is homogeneous in time, i.e., the $K \times K$ transition probability matrix for Equation 4 does not change with time. The second assumption is that each vector

$\mathbf{R}_t$ at time t is independent of all other observed and unobserved states up to time t, conditional on the hidden state S_t at time t , i.e.,

$$P\left(\mathbf{R}_t|S_{1:t},\mathbf{R}_{1:t-1}\right) = P\left(\mathbf{R}_t|S_t\right). \tag{5}$$

Specifying a full joint distribution $P\left(\mathbf{R}_t|S_t\right)$ would require $O(B^M)$ joint probabilities per state, which is clearly impractical even for moderate values of M. In practice, to avoid this problem, simpler models are often used, such as assuming that each vector component R_t^j is conditionally independent (CI) of the other components, given the state S_t, i.e.,

$$P\left(\mathbf{R}_t|S_t\right) = P(R_t^1,\ldots,R_t^M|S_t) = \prod_{j=1}^{M} P(R_t^j|S_t).$$

We will use this HMM-CI as our baseline model in the experimental results section later in the paper—in what follows below we explore models that can capture more dependence structure by using CL-trees.

2.4 Chow-Liu Structures and HMMs

We can use HMMs with Chow-Liu trees or conditional Chow-Liu forests to model the output variable given the hidden state. HMMs can model temporal structure of the data while the Chow-Liu models can capture "instantaneous" dependencies between multivariate outputs as well as additional dependence between vector components at consecutive observations over time that the state variable does not capture.

By combining HMMs with the Chow-Liu tree model and with the conditional Chow-Liu forest model we obtain HMM-CL and HMM-CCL models, respectively. The set of parameters $\mathbf{\Theta}$ for these models with K hidden states and B-valued M-variate vector sets consists of a $K \times K$ transition matrix $\mathbf{\Gamma}$, a $K \times 1$ vector $\mathbf{\Pi}$ of probabilities for the first hidden state in a sequence, and Chow-Liu trees or conditional forests for each hidden state $\mathbf{T} = \{T_1,\ldots,T_K\}$. Examples of graphical model structures for both the HMM-CL and HMM-CCL are shown in Figures 5 and 6 respectively. The likelihood of $\mathbf{\Theta}$ can then be computed as

$$\begin{aligned}
L\left(\mathbf{\Theta}\right) &= P\left(\mathbf{R}_{1:T}|\mathbf{\Theta}\right) = \sum_{S_{1:T}} P\left(S_{1:T},\mathbf{R}_{1:T}|\mathbf{\Theta}\right) \\
&= \sum_{S_{1:T}} P\left(S_1|\mathbf{\Theta}\right) \prod_{t=2}^{T} P\left(S_t|S_{t-1},\mathbf{\Theta}\right) \prod_{t=1}^{T} P\left(\mathbf{R}_t|S_t,\mathbf{R}_{t-1},\mathbf{\Theta}\right) \\
&= \sum_{i_1=1}^{K} \pi_{i_1} T_{i_1}\left(\mathbf{R}_1\right) \sum_{t=2}^{T} \sum_{i_t=1}^{K} \gamma_{i_{t-1}i_t} T_{i_t}\left(\mathbf{R}_t|\mathbf{R}_{t-1}\right)
\end{aligned}$$

with $P\left(\mathbf{R}_t|S_t,\mathbf{R}_{t-1},\mathbf{\Theta}\right) = P\left(\mathbf{R}_t|S_t,\mathbf{\Theta}\right)$ and $T_i\left(\mathbf{R}_t|\mathbf{R}_{t-1}\right) = T_i\left(\mathbf{R}_t\right)$ for the HMM-CL.

For hidden state S_{t-1} taking value i, the probability distribution $P\left(\mathbf{R}_t|\mathbf{\Theta}\right)$ is just a mixture of Chow-Liu trees (Meilă and Jordan 2000) with mixture coefficients $(\gamma_{i1},\ldots,\gamma_{iK})$ equal to the i-th row of the transition matrix Γ.

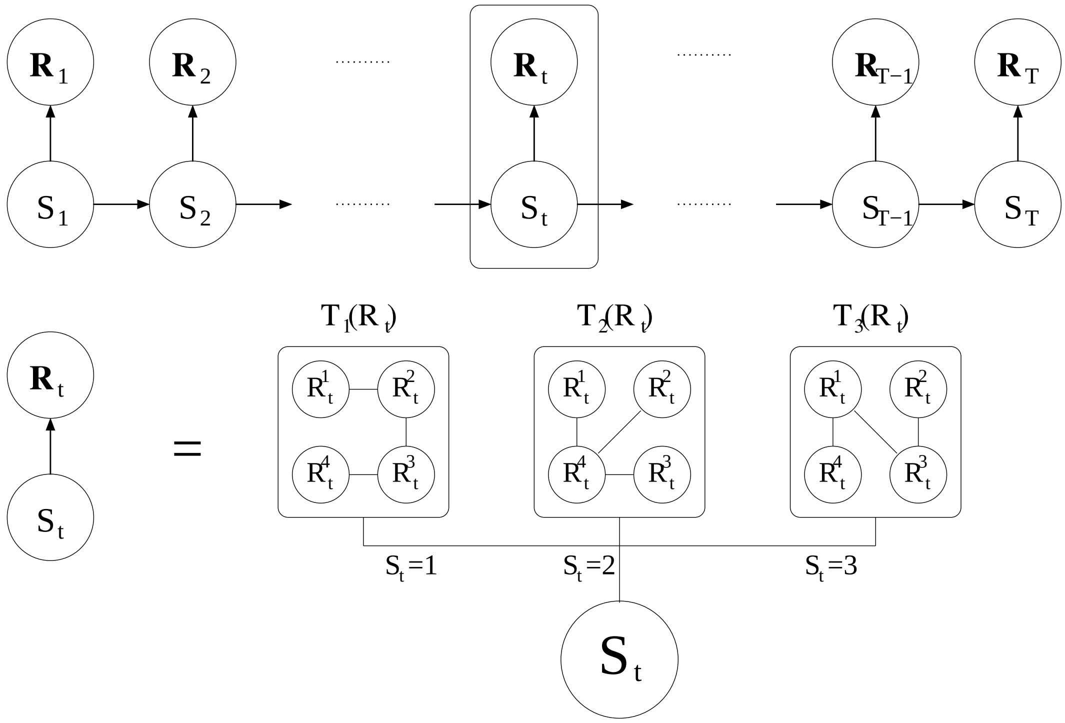

Figure 5: Graphical model interpretation of a hypothetical HMM-CL

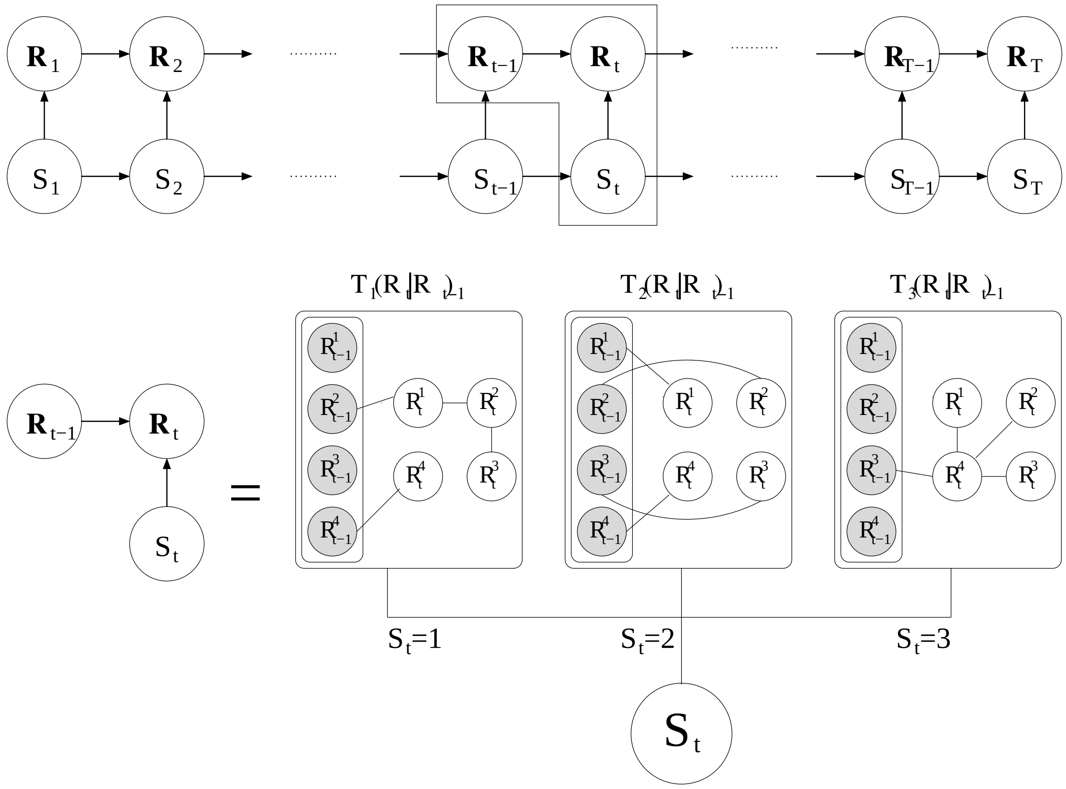

Figure 6: Graphical model interpretation of a hypothetical HMM-CCL

As a side note, since the outputs can depend directly on outputs at the previous time step in an HMM-CCL, the model can be viewed as a constrained form of autoregressive HMM (AR-HMM, Rabiner 1989) with the log-likelihood defined as

$$\begin{aligned} L(\Theta) &= P(\mathbf{R}_{1:T}|\Theta) = \sum_{S_{1:T}} P(S_{1:T}, \mathbf{R}_{1:T}|\Theta) \\ &= \sum_{S_{1:T}} P(S_1|\Theta) \prod_{t=2}^{T} P(S_t|S_{t-1}, \Theta) \prod_{t=1}^{T} P(\mathbf{R}_t|S_t, \mathbf{R}_{t-1}, \Theta). \end{aligned}$$

Note that a fully connected (unconstrained) AR-HMM would require $O\left(KB^{2M} + K^2\right)$ parameters.

3 Inference and Learning of HMM-based Models

In this section we discuss both (a) learning the structure and the parameters of the HMM-CL and HMM-CCL models discussed above, and (b) inferring probability distributions of the hidden states for given a set of observations and a model structure and its parameters. We outline how these operations can be performed for both the HMM-CL and HMM-CCL.

3.1 Inference of the Hidden State Distribution

The probability of the hidden variables $S_{1:T}$ given complete observations $\mathbf{R}_{1:T}$ can be computed as

$$P(S_{1:T}|\mathbf{R}_{1:T}) = \frac{P(S_{1:T}, \mathbf{R}_{1:T})}{\sum_{S_{1:T}} P(S_{1:T}, \mathbf{R}_{1:T})}.$$

The likelihood (denominator) cannot be calculated directly since the sum is exponential in T. However, the well-known recursive Forward-Backward procedure can be used to collect the necessary information in $O\left(TK^2M\right)$ without exponential complexity (e.g., Rabiner 1989). The details are provided in Appendix B.

3.2 Learning

Learning in HMMs is typically performed using the Baum-Welch algorithm (Baum et al. 1970), a variant of the Expectation-Maximization (EM) algorithm (Dempster et al. 1977). Each iteration of EM consists of two steps. First (E-step), the estimation of the posterior distribution of latent variables is accomplished by the Forward-Backward routine. Second (M-step), the parameters of the models are updated to maximize the expected log-likelihood of the model given the distribution from the M-step. The structures of the trees are also updated in the M-step.

The parameters Π and Γ are calculated in the same manner as for regular HMMs. Updates for $T_1, \ldots, T_K$ are computed similar to the algorithm for mixtures of trees (Meilă and Jordan 2000). Suppose $\mathbf{R}_{1:T} = \mathbf{r}_{1:T}$. Let T_i' denote the Chow-Liu tree for $S_t = i$ under the updated model. It can be shown (see Appendix B) that to improve the log-likelihood one needs to maximize

$$\sum_{i=1}^{K} \left(\sum_{\tau=1}^{T} P(S_\tau = i|\mathbf{R}_{1:T} = \mathbf{r}_{1:T}) \right) \sum_{t=1}^{T} P_i(\mathbf{r}_t) \log T_i'(\mathbf{r}_t)$$

where $P_i(\mathbf{r}_t) = \frac{P(S_t=i|\mathbf{R}_{1:T}=\mathbf{r}_{1:T})}{\sum_{\tau=1}^{T} P(S_\tau=i|\mathbf{R}_{1:T}=\mathbf{r}_{1:T})}$. This can be accomplished by separately learning Chow-Liu structures for the distributions P_i, the normalized posterior distributions of the hidden states calculated in the E-step. The time complexity for each iteration is then $O\left(TK^2M\right)$ for the E-step and $O\left(TK^2 + KTM^2B^2\right)$ for the M-step.

4 Experimental Results

To demonstrate the application of the HMM-CL and HMM-CCL models, we consider the problem of modelling precipitation occurrences for a network of rain stations. The data we examine here consists of binary measurements (indicating precipitation or not) recorded each day over a number of years for each of a set of rain stations in a local region. Figures 7 and 8 show networks of such stations in Southwestern Australia and Western U.S., respectively.

The goal is to build models that broadly speaking capture both the temporal and spatial properties of the precipitation data. These models can then be used to simulate realistic rainfall patterns over seasons (e.g., 90-day sequences), as a basis for making seasonal forecasts (Robertson et al. 2003), and to fill in missing rain station reports in the historical record.

Markov chains provide a well-known benchmark for modelling precipitation time-series at individual stations (e.g., Wilks and Wilby 1999). However, it is non-trivial to couple multiple chains together so that they exhibit realistic spatial correlation in simulated rainfall patterns. We also compare against the simplest HMM with a conditional independence (CI) assumption for the rain stations given the state. This model captures the marginal dependence of the stations to a certain degree since (for example) in a "wet state" the probability for all stations to be wet is higher, and so forth. However, the CI assumption clearly does not fully capture the spatial dependence, motivating the use of models such as HMM-CL and HMM-CCL.

In the experiments below we use data from both Southwestern Australia (30 stations, 15 184-day winter seasons beginning May 1) and the Western United States (8 stations, 39 90-day seasons beginning December 1). In fitting HMMs to this type of precipitation data the resulting "weather states" are often of direct scientific interest from a meteorological viewpoint. Thus, in evaluating these models, models that can explain the data with fewer states are generally preferable.

We use leave-one-out cross-validation to evaluate the fit of the models to the data. For evaluation we use two different criteria. We compute the log-likelihood for seasons not in the training data, normalized by the number of binary events in the left-out sets (referred to here as scaled log-likelihood). We also compute the average classification error in predicting observed randomly-selected station readings that are deliberately removed from the training data and then predicted by the model. This simulates the common situation of missing station readings in real precipitation records. The models considered are the independent Markov chains model (or "weather generator" model), the chain Chow-Liu forest model, the HMM with conditional independence (HMM-CI), the HMM with Chow-Liu tree emissions (HMM-CL), and the HMM with conditional Chow-Liu tree emissions (HMM-CCL). For HMMs, K is chosen corresponding to the largest scaled log-likelihood for each model—the smallest such K is then used across different HMM types for comparison.

The scatter plots in Figures 9 and 10 show the scaled log-likelihoods and classification errors for the models on the left-out sets. The y-axis is the performance of the HMM-CCL model, and the x-axis represents the performance of the other models (shown with different symbols). Higher implies better performance for log-likelihood (on the left) and worse for error (on the right). The

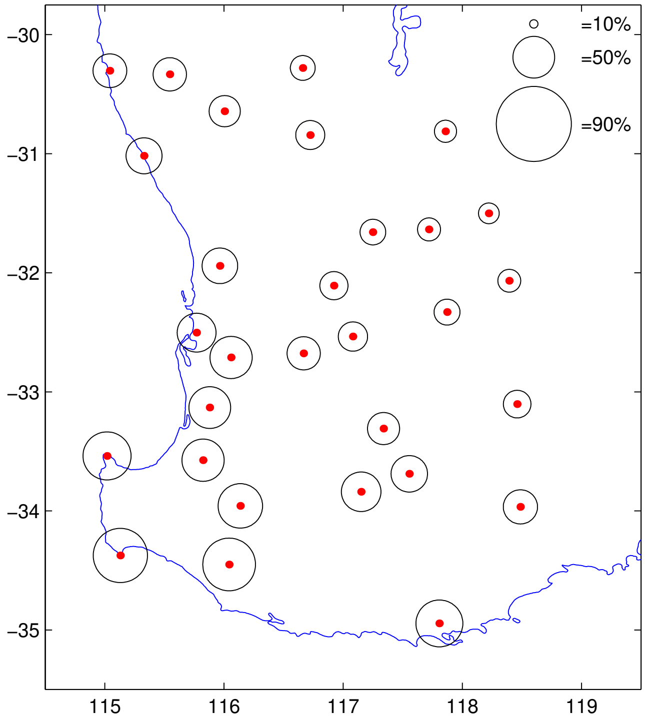

Figure 7: Stations in the Southwestern Australia region. Circle radii indicate marginal probabilities of rainfall ($> 0.3mm$) at each location.

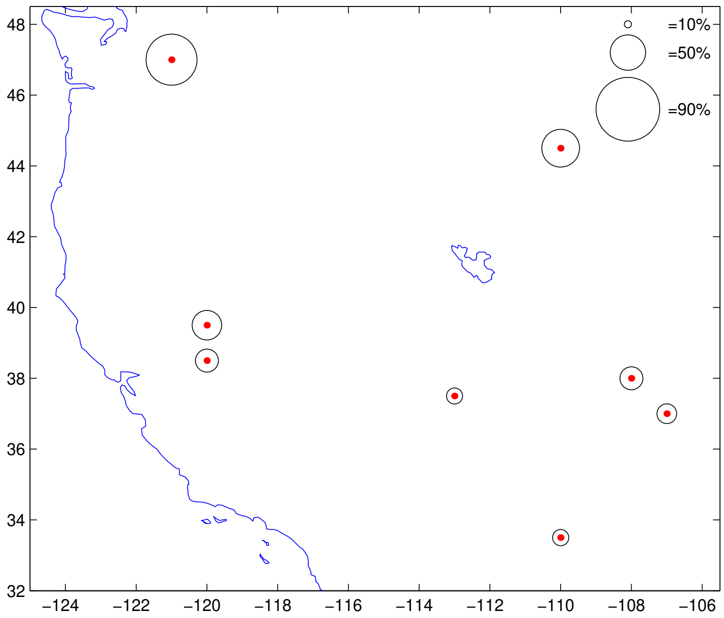

Figure 8: Stations in the Western U.S. region. Circle radii indicate marginal probabilities of rainfall ($> 0mm$) at each location.

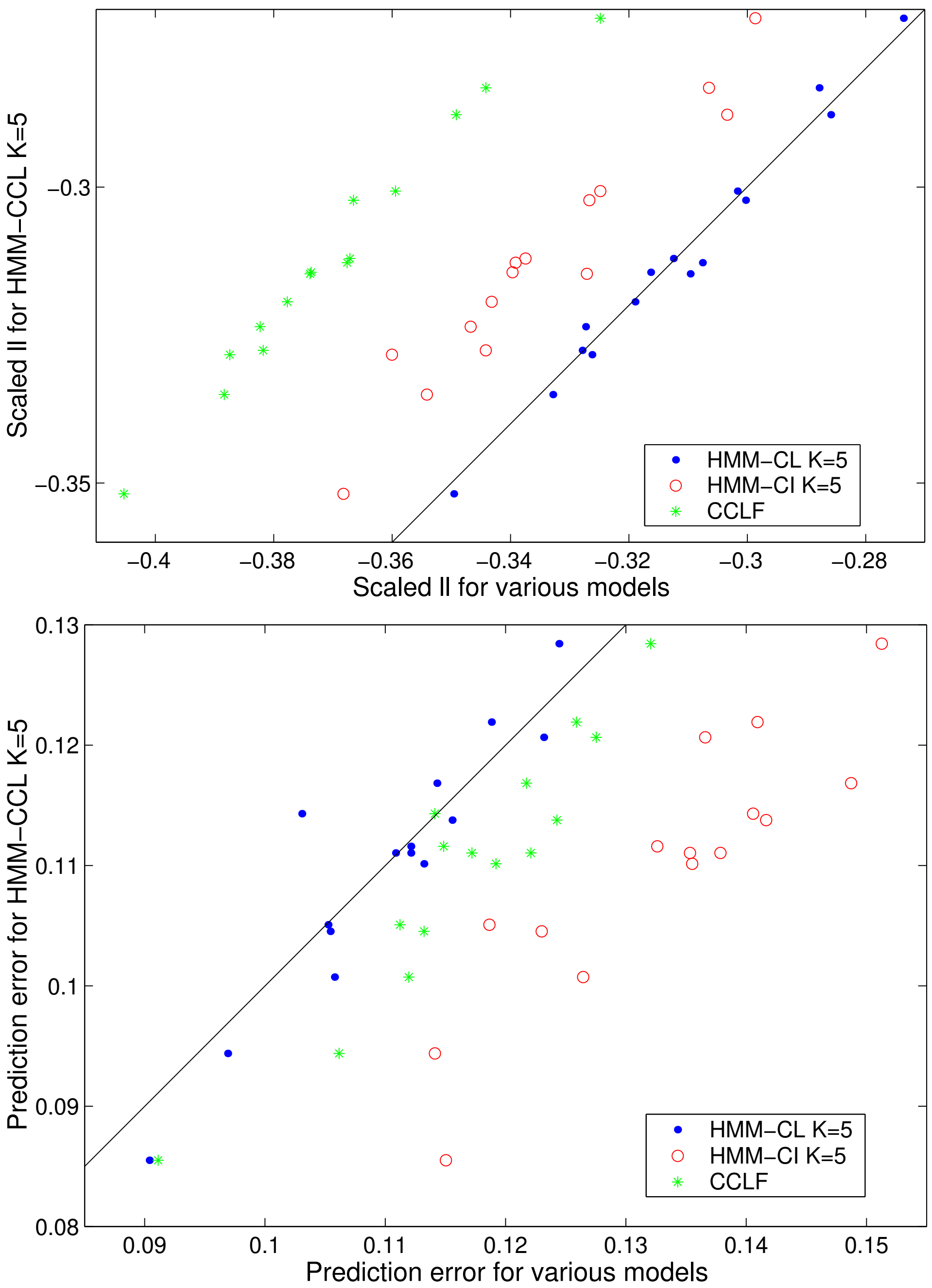

Figure 9: Southwestern Australia data: scatterplots of scaled log-likelihoods (top) and average prediction error (bottom) obtained by leave-one-winter-out cross-validation. The line corresponds to $y = x$. The independent chains model is not shown since it is beyond the range of the plot (average $ll = -0.6034$, average error $= 0.291$).

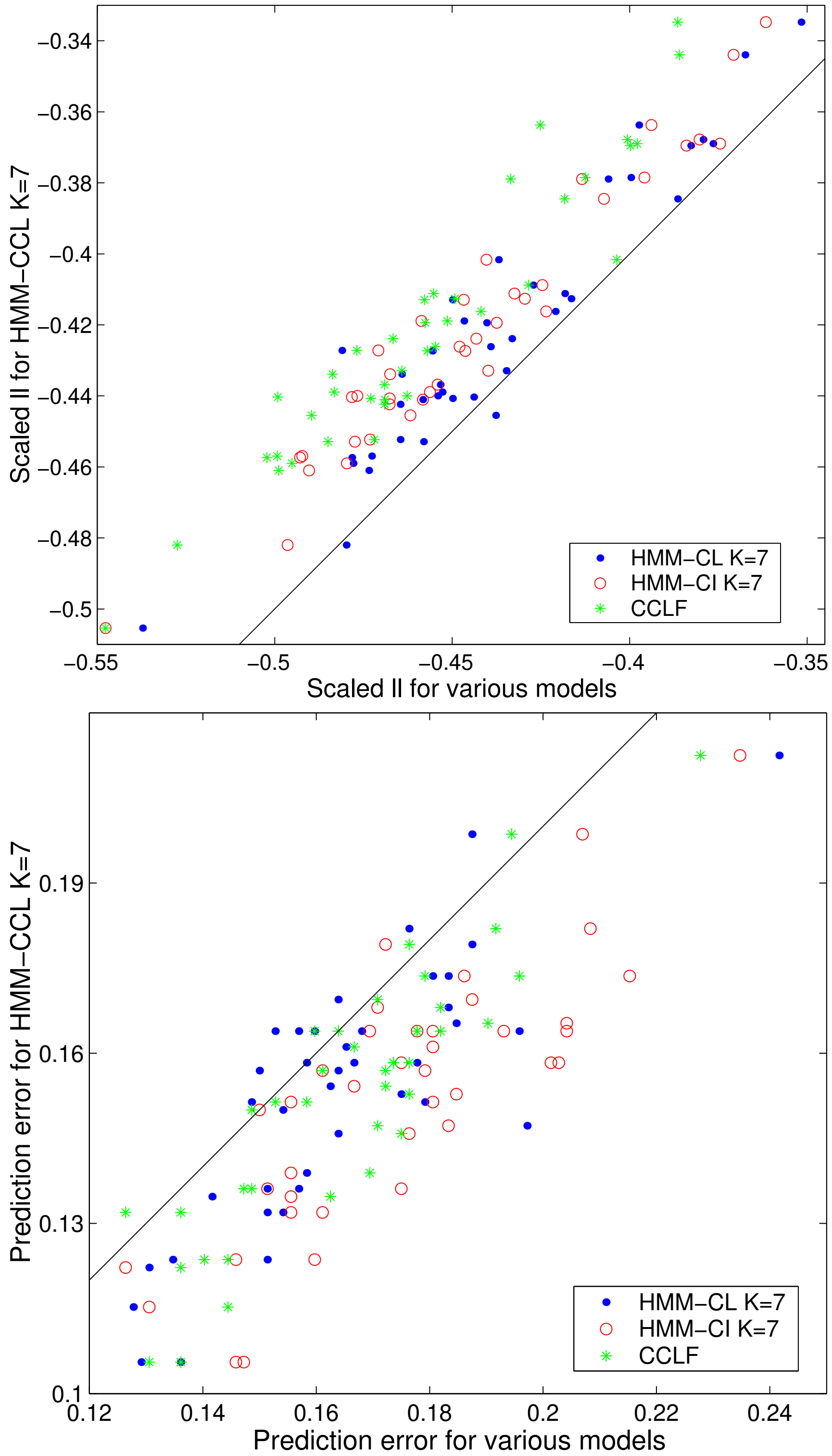

Figure 10: Western U.S. data: Scatterplots of scaled log-likelihoods (top) and average prediction error (bottom) obtained by leave-one-winter-out cross-validation. The line corresponds to $y = x$. The independent chains model is not shown since it is beyond the range of the plot (average $ll = -0.5204$, average error $= 0.221$).

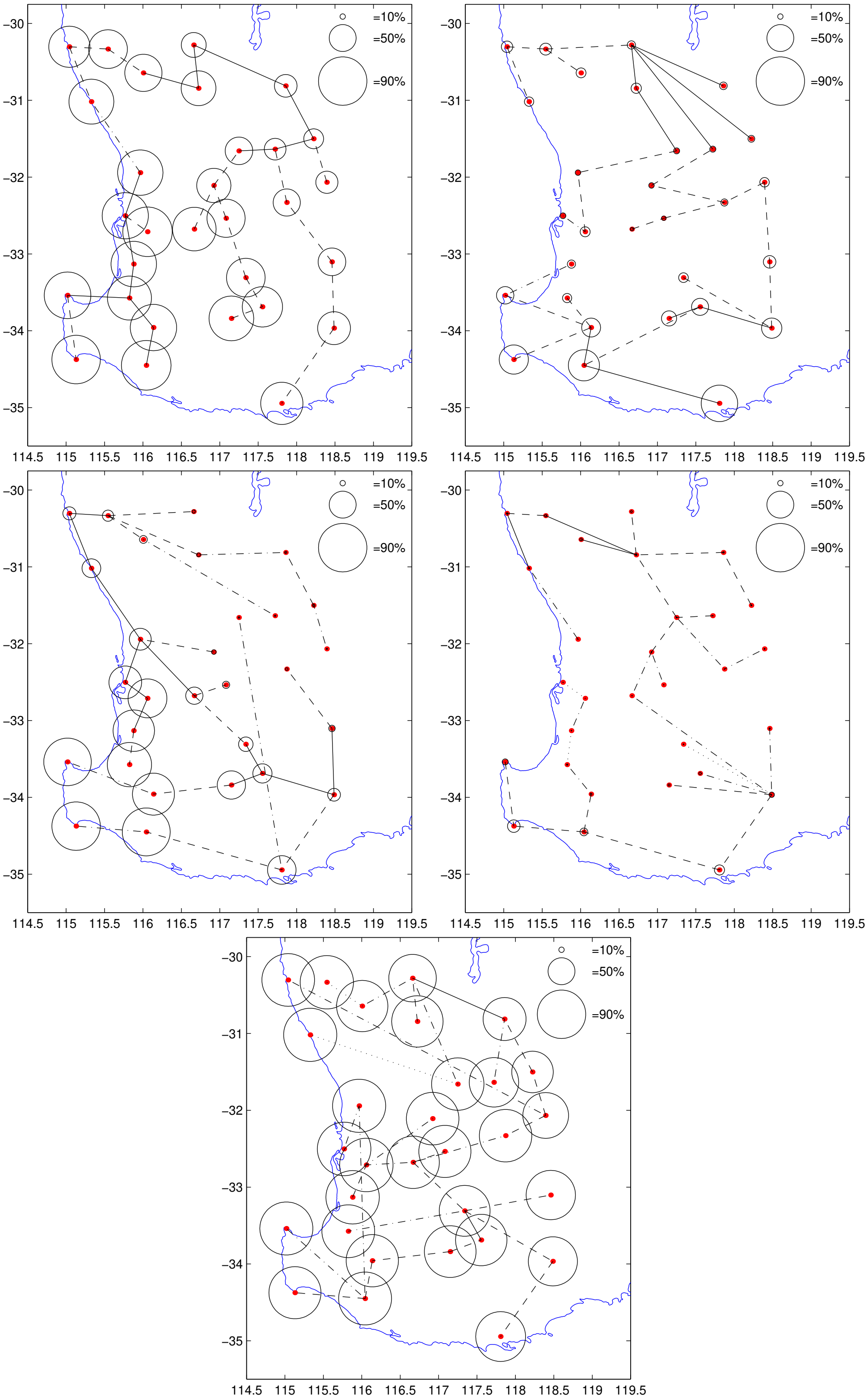

Figure 11: Graphical interpretation of the hidden states for a 5-state HMM-CL trained on South-western Australia data. Circle radii indicate the precipitation probability for each station given the state. Lines between the stations indicate the edges in the graph while different types of lines indicate the strength of mutual information of the edges.

HMM-CL and HMM-CCL models are systematically better than the CCLF and HMM-CI models, for both score functions, and for both data sets. The HMM-CCL model does relatively better than the HMM-CL model on the U. S. data. This is explained by the fact that the Australian stations are much closer spatially than the U.S. stations, so that for the U.S. the temporal connections that the HMM-CCL adds are more useful than the spatial connections that the HMM-CL model is limited to.

Examples of the Chow-Liu tree structures learned by the model are shown in Figure 11 for the 5-state HMM-CL model trained on all 15 years of Southwestern Australia data. The states learned by the model correspond to a variety of wet and dry spatial patterns. The tree structures are consistent with the meteorology and topography of the region (Hughes et al. 1999). Winter rainfall over SW Australia is large-scale and frontal, impacting the southwest corner of the domain first and foremost. Hence, the tendency for correlations between stations along the coast during moderately wet weather states. Interesting correlation structures are also identified in the north of the domain even during dry conditions.

Figures 12 and 14 demonstrate the spatial nature of the dependencies in the Southwestern Australia data. The structure of the conditional Chow-Liu forests contains very few edges corresponding to temporal dependence as the stations are spatially close, and spatial dependencies contain more information than the temporal ones. In contrast, Figures 13 and 15 suggest that the spatial dependencies of the Western U.S. data is weak which is consistent with the geographical sparsity of the stations.

5 Conclusions

We have investigated a number of approaches for modelling multivariate discrete-valued time series. In particular we illustrated how Chow-Liu trees could be embedded within hidden Markov models to provide improved temporal and multivariate dependence modeling in a tractable and parsimonious manner. We also introduced the conditional Chow-Liu forest model, a natural extension of Chow-Liu trees for modeling conditional distributions such as multivariate data with temporal dependencies. Experimental results on real-world precipitation data indicate that these models provide systematic improvements over simpler alternatives such as assuming conditional independence of the multivariate outputs. There are a number of extensions that were not discussed in this paper but that can clearly be pursued, including (a) using informative priors over tree-structures (e.g., priors on edges based on distance and topography for precipitation station models), (b) models for real-valued or mixed data (e.g., modelling precipitation amounts as well as occurrences), (c) adding input variables to the HMMs (e.g., to model "forcing" effects from atmospheric measurements—for initial results see Robertson et al. (2003)), and (d) performing more systematic experiments comparing these models to more general classes of dynamic Bayesian networks where temporal and multivariate structure is learned directly.

Acknowledgements

We would like to thank Stephen Charles of CSIRO, Australia, for providing us with the Western Australia data. This work was supported by the Department of Energy under grant DE-FG02-02ER63413.

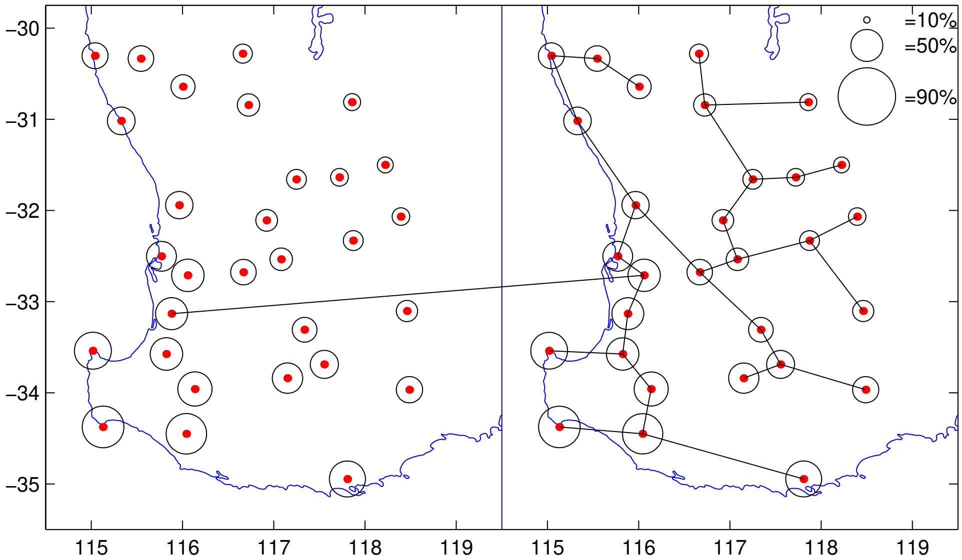

Figure 12: Graphical interpretation of the CCLF trained on Southwestern Australia data. Circle radii indicate the precipitation probability for each station. Lines between the stations indicate the edges in the graph while different types of lines indicate the strength of mutual information of the edges. The left side of the plot corresponds to observations $\mathbf{R}_{t-1}$ while the right side to $\mathbf{R}_t$.

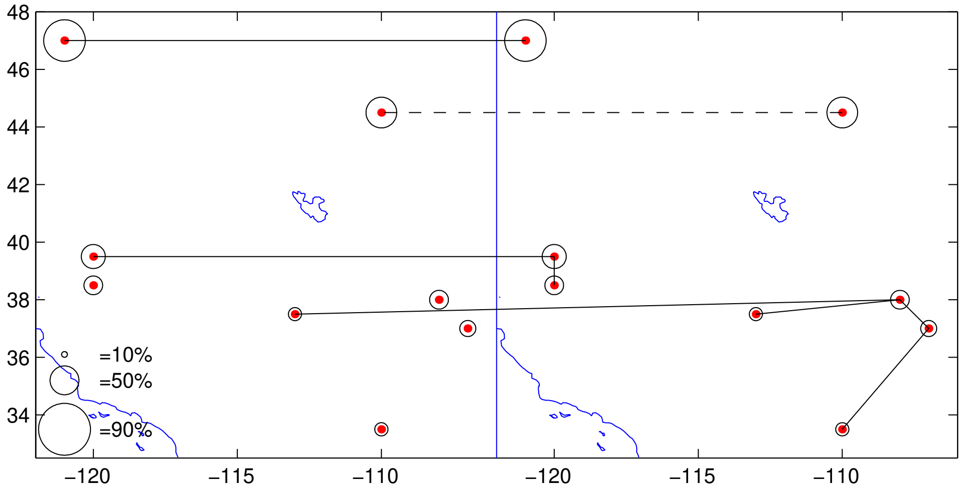

Figure 13: Graphical interpretation of the CCLF trained on Western U.S. data. Circle radii indicate the precipitation probability for each station. Lines between the stations indicate the edges in the graph while different types of lines indicate the strength of mutual information of the edges. The left side of the plot corresponds to observations $\mathbf{R}_{t-1}$ while the right side to $\mathbf{R}_t$.

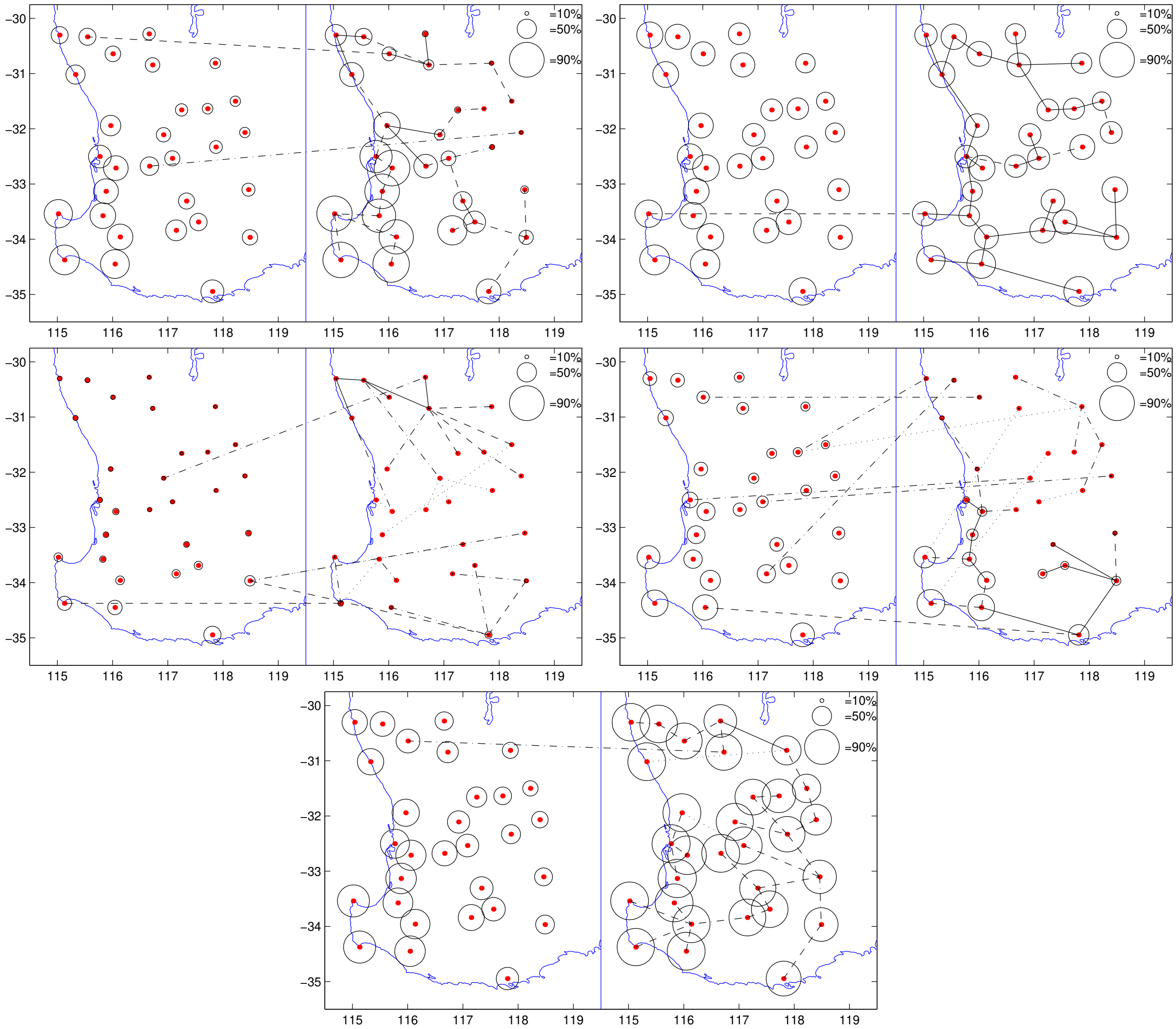

Figure 14: Graphical interpretation of the hidden states for a 5-state HMM-CCL trained on Southwestern Australia data. Circle radii indicate the precipitation probability for each station given the state. Lines between the stations indicate the edges in the graph while different types of lines indicate the strength of mutual information of the edges. The left side of the plot corresponds to observations $\mathbf{R}_{t-1}$ while the right side to $\mathbf{R}_t$.

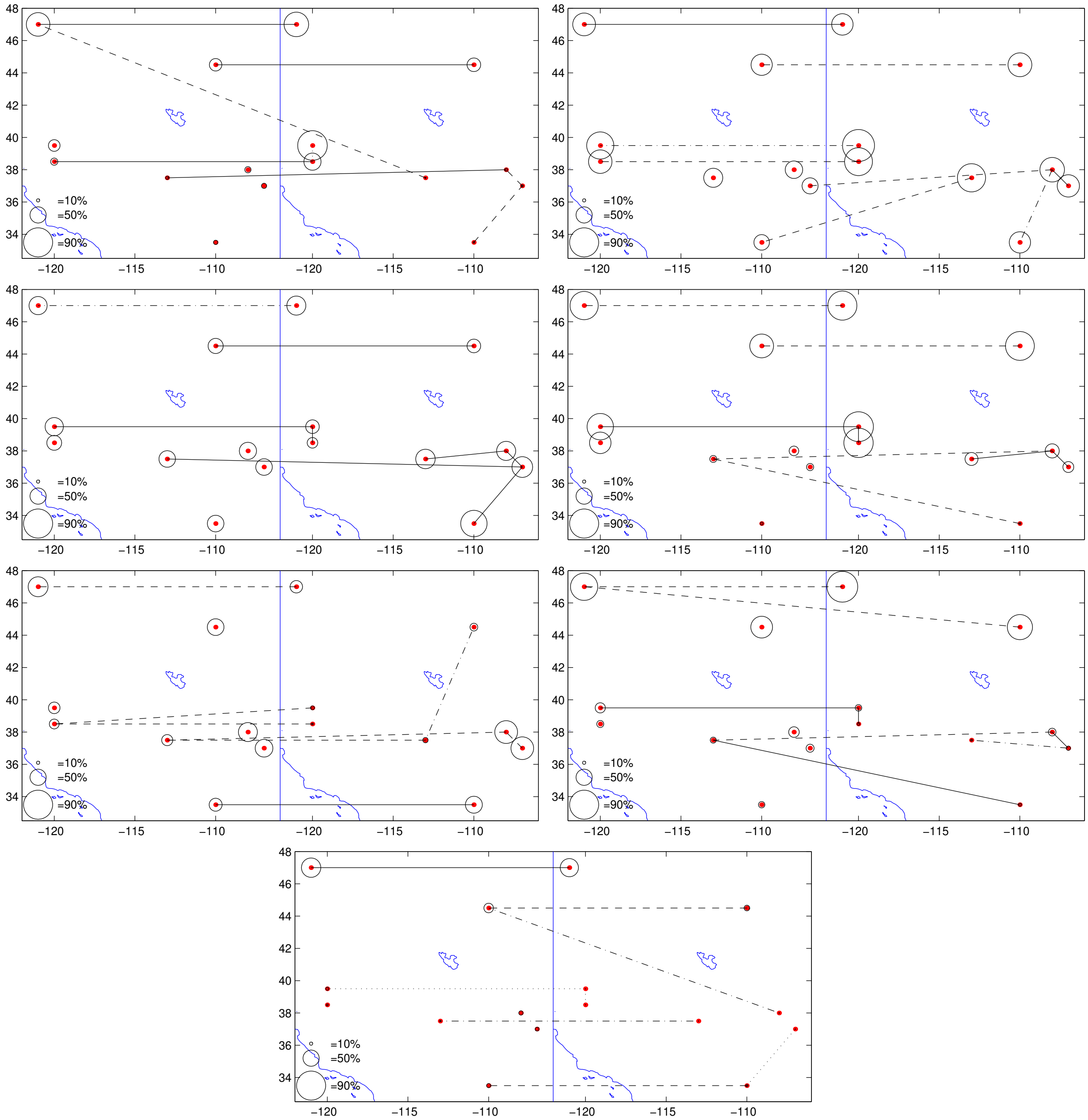

Figure 15: Graphical interpretation of the hidden states for a 7-state HMM-CCL trained on Western U.S. data. Circle radii indicate the precipitation probability for each station given the state. Lines between the stations indicate the edges in the graph while different types of lines indicate the strength of mutual information of the edges. The left side of the plot corresponds to observations $\mathbf{R}_{t-1}$ while the right side to $\mathbf{R}_t$.

A Tree Structure Optimality

In this section we will prove the results related to finding optimal tree structures. The proof consists of two parts. In the first part, we consider a Bayesian network interpretation of the approximating distribution T, and show that to minimize KL-distance between T and the true distribution P, P and T must agree on all conditional distributions for individual variables. This implies that probability distributions on the edges on the tree must agree with P. The second part describing how to select the edges of the tree was originally described by Chow and Liu (1968).

Theorem 1. *Let P be a distribution on the multivariate $\mathbf{X}$ on discrete variables $V = \{x^1, \ldots, x^M\}$. Let T be another distribution on $\mathbf{X}$. Assume a Bayesian network B_T describing the distribution T with parents (x^i) denoting a set of nodes needed to describe a decomposition of the joint probability distribution T on $\mathbf{x} \in \mathbf{X}$*

$$T(\mathbf{x}) = \prod_{i=1}^{M} T\left(x^i | parents\left(x^i\right)\right). \tag{6}$$

Then the distribution T minimizing $KL(P,T)$ must agree with P for all distribution components, i.e.,

$$T\left(x^i | parents\left(x^i\right)\right) = P\left(x^i | parents\left(x^i\right)\right) \quad i = 1, \ldots, M.$$

Proof. We are interested in finding T such that

$$T = \arg\min_{T^\star} KL\left(P, T^\star\right) = \arg\max_{T^\star} \sum_{\mathbf{x}} P(\mathbf{x}) \log T^\star(\mathbf{x})$$

with $T^\star$ having decomposition described by B_T. Let $\mathbf{x}^{V_i} = \{x^i\} \cup parents\left(x^i\right)$. Then by Equation 6

$$\sum_{\mathbf{x}} P(\mathbf{x}) \log T^\star(\mathbf{x}) = \sum_{i=1}^{M} \sum_{\mathbf{x}^{V_i}} P\left(\mathbf{x}^{V_i}\right) \log T^\star\left(x^i | parents\left(x^i\right)\right) \tag{7}$$

$$= \sum_{i=1}^{M} \sum_{\mathbf{x}^{V_i}} P\left(\mathbf{x}^{V_i}\right) \log P\left(x^i | parents\left(x^i\right)\right) - \sum_{i=1}^{M} KL\left(P\left(x^i | parents\left(x^i\right)\right), T^\star\left(x^i | parents\left(x^i\right)\right)\right) \tag{8}$$

The maximum of Expression 7 is achieved when the Kullback-Liebler divergences are minimized in Expression 8. The KL-divergence $KL(P,T)$ has a unique minimum point with value 0 for $P \equiv T$. Thus Expression 7 is minimized when and only when

$$T^\star\left(x^i | parents\left(x^i\right)\right) = P\left(x^i | parents\left(x^i\right)\right) \quad i = 1, \ldots, M.$$

It also follows that

$$\min_{T^\star} KL\left(P, T^\star\right) = \sum_{\mathbf{x}} P(\mathbf{x}) \log P(\mathbf{x}) - \sum_{i=1}^{M} \sum_{\mathbf{x}^{V_i}} P\left(\mathbf{x}^{V_i}\right) \log P\left(x^i | parents\left(x^i\right)\right).$$

□

Chow and Liu (1968) showed that if T has a tree structure, i.e. $parents\left(x^i\right)$ consists of not more than one x^j, then there is an efficient greedy method for finding the optimal such tree. If

$G_T = (V, E_T)$ a Markov network associated with the tree, then by Equation 1

$$\begin{aligned} KL(P,T) &= \sum_{\mathbf{x}} P(\mathbf{x}) \log P(\mathbf{x}) - \sum_{v \in V} \sum_{x^v} P(x^v) \log P(x^v) \\ &\quad - \sum_{(u,v) \in E_T} \sum_{x^u, x^v} P(x^u, x^v) \log \frac{P(x^u, x^v)}{P(x^u) P(x^v)} \\ &= -H[\mathbf{x}] + \sum_{v \in V}^{K} H[x^v] - \sum_{(u,v) \in E_T} I(x^u, x^v). \end{aligned} \tag{9}$$

Since the entropies $H[\mathbf{x}] = -\sum_{\mathbf{x}} P(\mathbf{x}) \log P(\mathbf{x})$ and $H[x^v] = \sum_{x^v} P(x^v) \log P(x^v)$ in Expression 9 are independent of the structure, $KL(P,T)$ is minimized by selecting edges E_T to maximize the sum of mutual informations $\sum_{(u,v) \in E_T} I(x^u, x^v)$.

Similarly, for conditional Chow-Liu forests with structure $G_F = (V, E_F)$ defined in Section 2.2, $KL(P,T)$ can be computed as

$$\begin{aligned} KL(P,T) &= \sum_{\mathbf{x},\mathbf{y}} P(\mathbf{x},\mathbf{y}) \log \frac{P(\mathbf{x}|\mathbf{y})}{T(\mathbf{x}|\mathbf{y})} \\ &= \sum_{\mathbf{x},\mathbf{y}} P(\mathbf{x},\mathbf{y}) \log P(\mathbf{x}|\mathbf{y}) - \sum_{v \in V_x} \sum_{x^v} P(x^v) \log P(x^v) \\ &\quad - \sum_{(u,v) \in E_x} \sum_{x^u, x^v} P(x^u, x^v) \frac{P(x^u, x^v)}{P(x^u) P(x^v)} - \sum_{(u,v) \in E_y} \sum_{y^u, x^v} P(x^v, y^u) \frac{P(x^v|y^u)}{P(x^v) P(y^u)} \\ &= -H[\mathbf{y}|\mathbf{x}] + \sum_{v \in V_x} H[x^v] - \sum_{(u,v) \in E_x} I(x^u, x^v) - \sum_{(u,v) \in E_y} I(y^u, x^v). \end{aligned} \tag{10}$$

Since neither H in the Expression 10 depends on the network structure, KL can be minimized by maximizing the sum of mutual informations I. This problem is equivalent to finding a maximum weight spanning tree in a graph with vertices $V = V_x \cup V_y$ where all nodes in V_y are already connected, and the weights of edges connecting pairs of nodes in V_x or pairs with one node in V_x and one in V_y are determined by appropriate mutual informations. We can view V_y as a supernode with weights from this supernode v' to a node $v \in V_x$ determined as the maximum mutual information from any node in V_y to v, i.e.,

$$weight\left(v', v\right) = \max_u I(y^u, x^v).$$

B Expectation-Maximization Algorithm for HMM-CL and HMM-CCL

We describe the details of how to learn the parameters $\mathbf{\Theta}$ of HMM-CL and HMM-CCL models on the data consisiting of multiple sequences of multi-variate discrete time series.

Assume that the data set $\mathcal{D}$ consists of N sequences each of length T (the learning algorithm can be easily generalized for sequences of unequal lengths). Let $\mathbf{r}_{nt} = \left(r_{nt}^1, \ldots, r_{nt}^M\right)$ denote an observation vector for time point t of sequence n, and let S_{nt} be the hidden state for the same time point. Assume that each of r_{nt}^m can take one of B values from the set $\mathcal{B} = \{0, \ldots, B-1\}$. (Again, the algorithm easily generalizes to sets of variables with different numbers of possible values.) By $\mathbf{r}_{n1:nT}$ or $\mathbf{r}_n$ we will denote the n-th observation sequence, and by $S_{n1:nT}$ or $\mathbf{S}_n$ — the sequence of hidden states corresponding to the observation sequence $\mathbf{r}_n$. We assume that each of the observed sequences is conditionally independent of the other sequences given the model.

Consider the set of parameters specified in Section 2.4. Under the HMM-CL or HMM-CCL, the log-likelihood $l(\mathbf{\Theta})$ of the observed data is defined as:

$$\begin{aligned} l(\mathbf{\Theta}) &= \ln P(\mathbf{r}_1, \ldots, \mathbf{r}_N|\mathbf{\Theta}) = \sum_{n=1}^{N} \sum_{\mathbf{S}_n} P(\mathbf{S}_n, \mathbf{r}_n|\mathbf{\Theta}) \\ &= \sum_{n=1}^{N} \ln \sum_{\mathbf{S}_n} P(S_{n1}|\mathbf{\Theta}) \prod_{t=2}^{T} P(S_{nt}|S_{n,t-1}, \mathbf{\Theta}) \prod_{t=1}^{T} P(\mathbf{r}_{nt}|S_{nt}, \mathbf{r}_{n,t-1}, \mathbf{\Theta}) \\ &= \sum_{n=1}^{N} \sum_{i_{n1}=1}^{K} \pi_{i_{n1}} T_{i_{n1}}(\mathbf{r}_{n1}) \sum_{t=2}^{T} \sum_{i_{nt}=1}^{K} \gamma_{i_{n,t-1} i_{nt}} T_{i_{nt}}(\mathbf{r}_{nt}|\mathbf{r}_{n,t-1}) \end{aligned}$$

with $T_{i_{nt}}(\mathbf{r}_{nt}|\mathbf{r}_{n,t-1}) \equiv T_{i_{nt}}(\mathbf{r}_{nt})$ for HMM-CL. (For HMM-CCL, $T_{i_{n1}}(\mathbf{r}_{n1})$ can be computed by summing $T_{i_{n1}}(\mathbf{r}_{n1}|\mathbf{r}')$ over all values $\mathbf{r}'$. It can be done efficiently using Equation 3 if we store $P(y^u, x^v)$ instead of $P(x^v|y^u)$.)

We seek the value of the parameters $\mathbf{\Theta}$ that maximizes the log-likelihood expression. This maximizing value cannot be obtained analytically—however, the EM algorithm provides an iterative method of climbing the $l(\mathbf{\Theta})$ surface in parameter space $\mathbf{\Theta}$. Starting with an initial set of parameters $\mathbf{\Theta}^0$, we iteratively calculate new sets of parameters improving the log-likelihood of the data at each iteration. Once a convergence criterion is reached, the last set of parameters $\hat{\mathbf{\Theta}}$ is chosen as the solution. This process of initialization followed by iterative "uphill" movement until convergence is repeated for several random initializations of $\mathbf{\Theta}^0$ and the $\hat{\mathbf{\Theta}}$ that corresponds to the largest value of $l(\hat{\mathbf{\Theta}})$ is chosen as the maximum likelihood estimate. The resulting solution $\hat{\mathbf{\Theta}}$ is not guaranteed to be at the global maximum. Since different initializations $\mathbf{\Theta}^0$ result in different trajectories through the parameter space and, eventually, in different local maxima, it is advisable to try several different initial values $\mathbf{\Theta}^0$ thus finding potentially different local maxima of the log-likelihood surface and choosing the one with the largest value.

At iteration r, parameters $\mathbf{\Theta}^{r+1}$ are selected to maximize

$$\begin{aligned} Q(\mathbf{\Theta}^r, \mathbf{\Theta}^{r+1}) &= E_{P(\mathbf{S}_1,\ldots,\mathbf{S}_N|\mathbf{r}_1,\ldots,\mathbf{r}_N,\mathbf{\Theta}^r)} \ln P(\mathbf{S}_1, \ldots, \mathbf{S}_N, \mathbf{r}_1, \ldots, \mathbf{r}_N|\mathbf{\Theta}^{r+1}) \\ &= \sum_{n=1}^{N} \sum_{\mathbf{S}_n} P(\mathbf{S}_n|\mathbf{r}_n, \mathbf{\Theta}^r) \ln P(\mathbf{S}_n, \mathbf{r}_n|\mathbf{\Theta}^{r+1}). \end{aligned}$$

It can be shown that $l(\mathbf{\Theta}^{r+1}) - l(\mathbf{\Theta}^r) \geq Q(\mathbf{\Theta}^r, \mathbf{\Theta}^{r+1}) - Q(\mathbf{\Theta}^r, \mathbf{\Theta}^r)$, so by maximizing $Q(\mathbf{\Theta}^r, \mathbf{\Theta}^{r+1})$, we guarantee an improvement in log-likelihood.

$Q(\mathbf{\Theta}^r, \mathbf{\Theta}^{r+1})$ is maximized in two steps. In the first, the E-step, we calculate $P(S_n|\mathbf{r}_n, \mathbf{\Theta}^r)$. In the second, the M-step, we maximize $Q(\mathbf{\Theta}^r, \mathbf{\Theta}^{r+1})$ with respect to the parameters in $\mathbf{\Theta}^{r+1}$. While it is infeasible to calculate and store probabilities of $N * K^T$ possible sequences of hidden states $P(\mathbf{S}_n|\mathbf{r}_n, \mathbf{\Theta}^r)$ as suggested in the E-step, it turns out we need only a manageable set of

$N * T * K$ probabilities $A_{nt}(i) = P(S_{nt} = i|\mathbf{r}_n, \mathbf{\Theta}^r)$ and $N * (T-1) * K^2$ probabilities $B_{nt}(i,j) = P(S_{nt} = i, S_{n,t-1} = j|\mathbf{r}_n, \mathbf{\Theta}^r)$ to perform optimization in the M-step. If $\mathbf{\Theta}^{r+1} = \{\mathbf{\Pi}', \mathbf{\Gamma}', \mathbf{T}'\}$, then

$$\begin{aligned}
Q\left(\mathbf{\Theta}^r, \mathbf{\Theta}^{r+1}\right) &= \sum_{n=1}^{N} \sum_{\mathbf{S}_n} P(\mathbf{S}_n|\mathbf{r}_n, \mathbf{\Theta}^r) \ln P\left(\mathbf{S}_n, \mathbf{r}_n|\mathbf{\Theta}^{r+1}\right) \\
&= \sum_{n=1}^{N} \sum_{\mathbf{S}_n} P(\mathbf{S}_n|\mathbf{r}_n, \mathbf{\Theta}^r) \left(\sum_{t=1}^{T} \ln P\left(\mathbf{r}_{nt}|S_{nt}, \mathbf{r}_{n,t-1}, \mathbf{\Theta}^{r+1}\right) \right. \\
&\quad \left. + \ln P\left(S_{n1}|\mathbf{\Theta}^{r+1}\right) + \sum_{t=2}^{T} \ln P\left(S_{nt}|S_{n,t-1}, \mathbf{\Theta}^{r+1}\right) \right) \\
&= \sum_{n=1}^{N} \sum_{t=1}^{T} \sum_{i=1}^{K} A_{nt}(i) \ln T_i'(\mathbf{r}_{nt}|\mathbf{r}_{n,t-1}) \qquad (11) \\
&\quad + \sum_{n=1}^{N} \sum_{i=1}^{K} A_{n1}(i) \ln \pi_i' + \sum_{n=1}^{N} \sum_{t=2}^{T} \sum_{i=1}^{K} \sum_{j=1}^{K} B_{nt}(i,j) \ln \gamma_{ji}'. \qquad (12) \\
&= Q_R + Q_S
\end{aligned}$$

where Q_R is equal to the Expression 11 and Q_S equal to the Expression 12.

The quantities A_{nt} and B_{nt} can be calculated using the recursive Forward-Backward procedure (Rabiner 1989). For each value of each hidden state, we recursively calculate a summary of information preceding the state (α_{nt}) and following the state (β_{nt}) as follows:

$$\alpha_{nt}(i) = P(S_{nt} = i, \mathbf{r}_n|\mathbf{\Theta}^r) \text{ and } \beta_{nt}(i) = P(\mathbf{r}_{n,t+1:nT}|S_{nt} = i, \mathbf{r}_{nt}, \mathbf{\Theta}^r).$$

Then

$$\begin{aligned}
\alpha_{n1}(i) &= \pi_i T_i(\mathbf{r}_{n1}) \text{ and } \alpha_{n,t+1}(j) = T_j(\mathbf{r}_{n,t+1}|\mathbf{r}_{nt}) \sum_{i=1}^{K} \gamma_{ij} \alpha_{nt}(i), \; t = 2, \ldots, T; \\
\beta_{nT}(i) &= 1 \text{ and } \beta_{nt}(i) = \sum_{j=1}^{K} \gamma_{ij} T_j(\mathbf{r}_{n,t+1}|\mathbf{r}_{nt}) \beta_{n,t+1}(j), \; t = T-1, \ldots, 1.
\end{aligned}$$

Once the values of α and β are obtained, the values of A and B can be computed:

$$A_{nt}(i) = \frac{\alpha_{nt}(i) \beta_{nt}(i)}{\sum_{k=1}^{K} \alpha_{nT}(k)} \text{ and } B_{nt}(i,j) = \frac{T(\mathbf{r}_{nt}|\mathbf{r}_{n,t-1}) \gamma_{ji} \alpha_{n,t-1}(j) \beta_{nt}(i)}{\sum_{k=1}^{K} \alpha_{nT}(k)}$$

The log-likelihood $l(\mathbf{\Theta})$ can be computed as

$$l(\mathbf{\Theta}) = \sum_{n=1}^{N} \ln P(\mathbf{r}_n|\mathbf{\Theta}) = \sum_{n=1}^{N} \ln \sum_{i=1}^{K} P(S_{nT} = i, \mathbf{r}_n|\mathbf{\Theta}) = \sum_{n=1}^{N} \ln \sum_{i=1}^{K} \alpha_{nt}(i).$$

For the M-step, Q_R and Q_S can be maximized separately. The most direct way to maximize Q_S is to take partial derivatives of Q_S (with added Lagrangians to adjust for constraints) with respect to $\mathbf{\Pi}$ and $\mathbf{\Gamma}$ and to make all of the partial derivatives zero. When applied we obtain

$$\pi_i' = \frac{\sum_{n=1}^{N} A_{n1}(i)}{N} \text{ and } \gamma_{ji}' = \frac{\sum_{n=1}^{N} \sum_{t=2}^{T} B_{nt}(i,j)}{\sum_{n=1}^{N} \sum_{t=1}^{T-1} A_{nt}(j)}.$$

Q_R can be maximized similar to the case of mixture of trees (Meilă and Jordan 2000):

$$\begin{aligned} Q_R &= \sum_{n=1}^{N}\sum_{t=1}^{T}\sum_{i=1}^{K} A_{nt}(i) \ln T_i'(\mathbf{r}_{nt}|\mathbf{r}_{n,t-1}) \\ &= \sum_{i=1}^{K}\left(\sum_{\nu=1}^{N}\sum_{\tau=1}^{T} A_{\nu\tau}(i)\right)\sum_{n=1}^{N}\sum_{t=1}^{T} P_i(\mathbf{r}_{nt}) \log T_k'(\mathbf{r}_{nt}|\mathbf{r}_{n,t-1}) \end{aligned}$$

where $P_i(\mathbf{r}_{nt}) = \frac{A_{nt}(i)}{\sum_{\nu=1}^{N}\sum_{\tau=1}^{T} A_{\nu\tau}(i)}$. This can be accomplished by separately learning Chow-Liu structures for the distributions P_i, $i = 1, \ldots, K$, the normalized posterior distributions of the hidden states calculated in the E-step.

References

F. R. Bach and M. I. Jordan. Thin junction trees. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 569–576, Cambridge, MA, 2002. MIT Press.

L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, 41(1): 164–171, February 1970.

D. M. Chickering. Learning Bayesian networks is NP-complete. In D. Fisher and H.-J. Lenz, editors, *Learning from data: AI and statistics V*, pages 121–130. Springer-Verlag, New York, 1996.

C. K. Chow and C. N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, IT-14(3):462–467, May 1968.

T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms.* The MIT Electrical Engineering and Computer Science Series. MIT Press/McGraw Hill, 1990.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via EM algorithm. *Journal of the Royal Statistical Society Series B-Methodological*, 39(1):1–38, 1977.

J. P. Hughes and P. Guttorp. Incorporating spatial dependence and atmospheric data in a model of precipitation. *Journal of Applied Meteorology*, 33(12):1503–1515, December 1994.

J. P. Hughes, P. Guttorp, and S. P. Charles. A non-homogeneous hidden Markov model for precipitation occurrence. *Journal of the Royal Statistical Society Series C Applied Statistics*, 48(1): 15–30, 1999.

M. Meilă. An accelerated Chow and Liu algorithm: Fitting tree distributions to high-dimensional sparse data. In I. Bratko and S. Dzeroski, editors, *Proceedings of the Sixteenth International Conference on Machine Learning (ICML'99)*, pages 249–57. Morgan Kaufmann, 1999.

M. Meilă and M. I. Jordan. Learning with mixtures of trees. *Journal of Machine Learning Research*, 1(1):1–48, October 2000.

L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of IEEE*, 77(2):257–286, February 1989.

A. W. Robertson, S. Kirshner, and P. Smyth. Hidden Markov models for modeling daily rainfall occurrence over Brazil. Technical Report 03-27, School of Information and Computer Science, University of California, Irvine, 2003.

M. A. Semenov and J. R. Porter. Climatic variability and the modeling of crop yields. *Agricultural and Forest Meteorology*, 73(3-4):265–283, March 1995.

N. Srebro. Maximum likelihood bounded tree-width Markov networks. *Artificial Intelligence*, 143 (1):123–138, January 2003.

D. S. Wilks and R. L. Wilby. The weather generation game: a review of stochastic weather models. *Progress in Physical Geography*, 23(3):329–357, 1999.