Gibbs sampling for parsimonious Markov models with latent variables

Ralf Eggeling[1], Pierre-Yves Bourguignon[2], André Gohr[1], and Ivo Grosse[1]

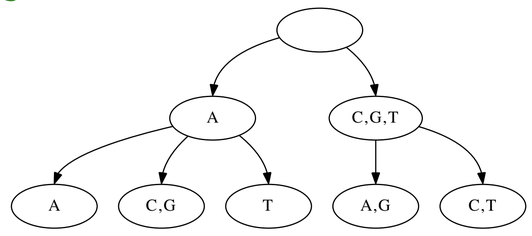
[1]*Institute of Computer Science, Martin-Luther-University Halle-Wittenberg, Germany,*
[2]*Max Planck Institute for Mathematics in the Sciences, Leipzig, Germany*
eggeling@informatik.uni-halle.de

Max Planck Institute for Mathematics in the Sciences

Abstract

We propose a Bayesian model averaging approach for learning mixtures of parsimonious Markov models that is based on Gibbs sampling. The challenging problem is sampling one out of a large number of model structures. We solve it by an efficient dynamic programming algorithm. We apply the resulting Gibbs sampling algorithm to splice site classification [2], an important problem from computational biology, and find the Bayesian approach to be superior to the non-Bayesian classification.

PCTs



Parsimonious Markov models

- model dependencies among adjacent symbols
- based on parsimonious context trees (PCTs) [1]
- allow merging context sequences into sets
- likelihood:

$$P(\mathbf{X}|\vec{\Theta}) = \prod_{\ell=1}^{L} \prod_{\mathbf{w}\in C_{\tau_\ell}} \prod_{a\in\mathcal{A}} (\theta_{\ell\mathbf{w}a}^{\tau_\ell})^{N_{\ell\mathbf{w}a}}$$

- generalize variable order Markov models

Mixture models

- many practical problems involve latent variables
- typical example: mixture of C component models

$$P(\mathbf{X}|\mathbf{\Theta}) = \prod_{i=1}^{N} \sum_{u_i=1}^{C} P(\vec{X}_i|u_i,\mathbf{\Theta}) P(u_i|\mathbf{\Theta})$$

- exact learning infeasible
  →approximative algorithms
  →EM algorithm, Gibbs sampling

Bayesian prediction

- assign probability of X given training data Y
- **traditional** prediction
  → estimate parameters $\mathcal{M}$ and $\theta_\mathcal{M}$ on Y

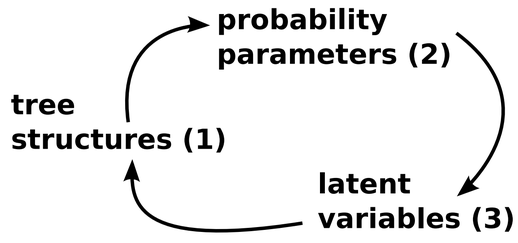
$$P(X|Y) = P(X|\hat{\theta}_{\hat{\mathcal{M}}(Y)}(Y))$$

- **Bayesian** prediction
  → intergrate over parameter space

$$P(X|Y) = \sum_{\mathcal{M}} P(\mathcal{M}) \int P(X|\theta_\mathcal{M}) P(\theta_\mathcal{M}|Y) d\theta_\mathcal{M}$$

Gibbs sampling algorithm

Sampling steps



$$\forall_{c=1}^{C}\forall_{\ell=1}^{L}: \text{ sample } \tau_{c\ell}^{(t)} \text{ from } P(\tau_{c\ell}|\vec{u}^{(t-1)},\mathbf{X}) \quad (1)$$
$$\forall_{i=1}^{C}\forall_{\ell=1}^{L}\forall_{\mathbf{w}\in\mathcal{C}_{\tau_{c\ell}}}: \text{ sample } \theta_{c\ell\mathbf{w}}^{\tau_{c\ell}\ (t)} \text{ from } P(\theta_{c\ell\mathbf{w}}^{\tau_{c\ell}}|\tau_{c\ell}^{(t)},\vec{u}^{(t-1)},\mathbf{X}) \quad (2)$$
$$\forall_{i=1}^{N}: \text{ sample } u_i^{(t)} \text{ from } P(u_i|\mathbf{\Theta}^{(t)},\vec{X}_i) \quad (3)$$

- step (1) → dynamic programming
- step (2) → sampling from Dirichlet distributions
- step (3) → trivial

Structure sampling

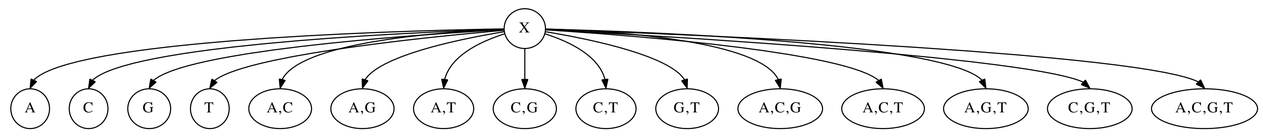
- structure score for each tree:

$$P(\tau_{c\ell}|\vec{u},\mathbf{X}) \propto \prod_{\mathbf{w}\in\mathcal{C}_{\tau_{c\ell}}} \kappa \frac{\mathcal{B}(\vec{N}_{c\ell\mathbf{w}}+\vec{\alpha}_{c\ell\mathbf{w}})}{\mathcal{B}(\vec{\alpha}_{c\ell\mathbf{w}})}$$

- product of leaf scores
- sample tree structure by dynamic programming
- traversal of extended tree (one subtree shown below)
- sample PCT subtree at each inner node according to recursion (right column)

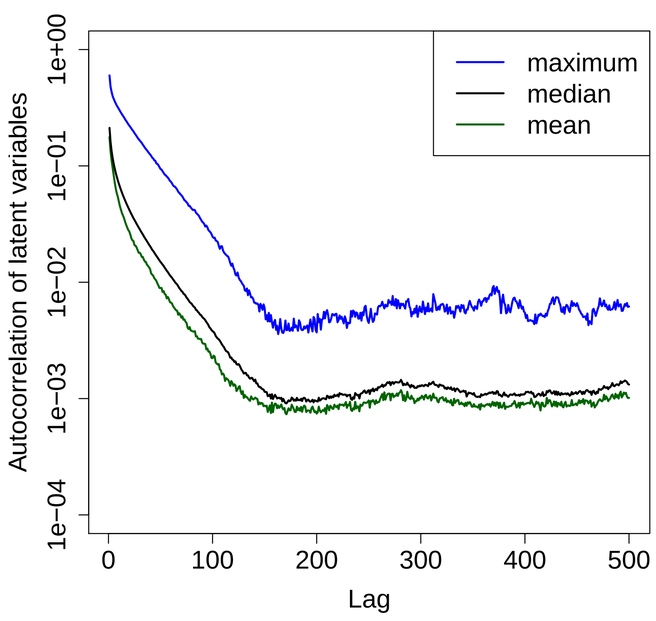
Recursion

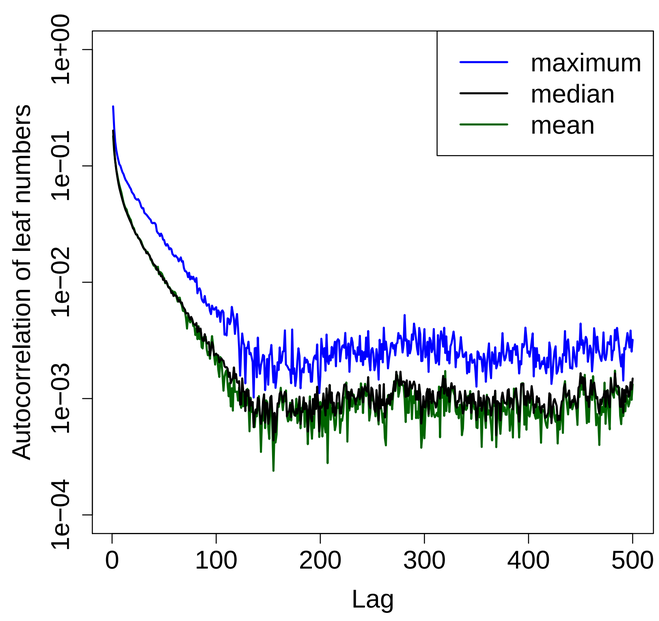
- for each node n in extended tree
- assumption: subtree below n is already valid PCT
- if n is leaf (base case):
  compute $S_n = \kappa \frac{\mathcal{B}(\vec{N}_{c\ell\mathbf{w}}+\vec{\alpha}_{c\ell\mathbf{w}})}{\mathcal{B}(\vec{\alpha}_{c\ell\mathbf{w}})}$
- if n is inner node (recursive step):
  - compute score $S'_\mathbf{p} = \prod_{m\in\mathbf{p}} S_m$ for each valid choice $\mathbf{p}$ of children (labels form a partition of $\mathcal{A}$) of n
  - sample $\mathbf{p}^*$ according to S'
  - discard all children not belonging to $\mathbf{p}^*$
  - set $S_n = S'_{\mathbf{p}^*}$



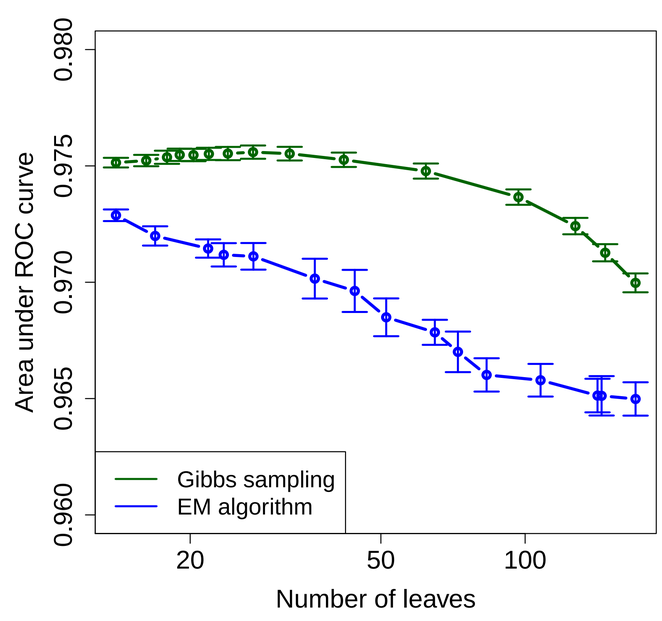
Results

Convergence





Classification



Acknowledgments

This work was funded by *Reisestipendium des allg. Stiftungsfonds der MLU Halle-Wittenberg*, *MPG/CNRS SysBio* research program, and DFG (grant no. GR-3526_1-1). We thank Petri Myllymäki, Teemu Roos, and Antti Honkela for valuable discussions.

References

[1] P.-Y. Bourguignon. *Parcimonie dans les modèles markoviens et applications à l'analyse des séquences biologiques*. PhD thesis, Université Evry Val d'Essonne, 2008.

[2] G. Yeo and C. Burge. Maximum Entropy Modeling of Short Sequence Motifs with Applications to RNA Splicing Signals. *Journal of Computational Biology*, 11(2/3):377–394, 2004.