NONREVERSIBLE POPULATION MCMC WITH APPLICATIONS

Jukka Corander

Department of Mathematics and Statistics

Geilo, Jan 2014

Primer on Bayesian model selection - I

- Let $\mathbf{x}$ denote a generic data set.

Primer on Bayesian model selection - I

- Let **x** denote a generic data set.
- We consider a finite set $\{p(\cdot|\theta_\delta, \delta \in \Delta\}$ of probability models.

Primer on Bayesian model selection - I

- Let **x** denote a generic data set.
- We consider a finite set $\{p(\cdot|\theta_\delta, \delta \in \Delta\}$ of probability models.
- Here δ is taken as the structural layer of a probability model, such the set of covariates in regression, set of edges in a graphical model, etc.

Primer on Bayesian model selection - I

- Let **x** denote a generic data set.
- We consider a finite set $\{p(\cdot|\theta_\delta, \delta \in \Delta\}$ of probability models.
- Here δ is taken as the structural layer of a probability model, such the set of covariates in regression, set of edges in a graphical model, etc.
- The parameter $\theta_\delta \in \Theta_\delta$ represents the quantitative layer, taking values on a space dependent on the specific structure δ.

Primer on Bayesian model selection - II

- The Bayesian approach specifies formally the predictive, or marginal data distribution as the mixture

$$p(\mathbf{x}) = \sum_{\delta \in \Delta} p(\delta) p(\mathbf{x}|\delta) = \sum_{\delta \in \Delta} p(\delta) \int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta),$$

Primer on Bayesian model selection - II

- The Bayesian approach specifies formally the predictive, or marginal data distribution as the mixture

$$p(\mathbf{x}) = \sum_{\delta \in \Delta} p(\delta) p(\mathbf{x}|\delta) = \sum_{\delta \in \Delta} p(\delta) \int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta),$$

- where $\mu(\theta_\delta)$ is a prior probability measure,

Primer on Bayesian model selection - II

- The Bayesian approach specifies formally the predictive, or marginal data distribution as the mixture

$$p(\mathbf{x}) = \sum_{\delta \in \Delta} p(\delta) p(\mathbf{x}|\delta) = \sum_{\delta \in \Delta} p(\delta) \int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta),$$

- where $\mu(\theta_\delta)$ is a prior probability measure,
- and $p(\mathbf{x}|\theta_\delta)$ is the conditional data distribution, or the likelihood, given θ_δ.

Primer on Bayesian model selection - II

- The Bayesian approach specifies formally the predictive, or marginal data distribution as the mixture

$$p(\mathbf{x}) = \sum_{\delta \in \Delta} p(\delta) p(\mathbf{x}|\delta) = \sum_{\delta \in \Delta} p(\delta) \int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta),$$

- where $\mu(\theta_\delta)$ is a prior probability measure,
- and $p(\mathbf{x}|\theta_\delta)$ is the conditional data distribution, or the likelihood, given θ_δ.
- The value of $p(\delta)$ can be interpreted as the prior predictive weight, or the prior probability of the structural layer δ, such that $\sum_{\delta \in \Delta} p(\delta) = 1$.

Primer on Bayesian model selection - II

- The Bayesian approach specifies formally the predictive, or marginal data distribution as the mixture

$$p(\mathbf{x}) = \sum_{\delta \in \Delta} p(\delta) p(\mathbf{x}|\delta) = \sum_{\delta \in \Delta} p(\delta) \int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta),$$

- where $\mu(\theta_\delta)$ is a prior probability measure,
- and $p(\mathbf{x}|\theta_\delta)$ is the conditional data distribution, or the likelihood, given θ_δ.
- The value of $p(\delta)$ can be interpreted as the prior predictive weight, or the prior probability of the structural layer δ, such that $\sum_{\delta \in \Delta} p(\delta) = 1$.
- We assume that the integral $\int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta)$ can be calculated analytically, either exactly or approximately.

Primer on Bayesian model selection - III

- In structural model learning one is typically interested in the posterior probabilities of $\delta \in \Delta$, as measures of the model plausibility to explain the information carried by the data.

Primer on Bayesian model selection - III

- In structural model learning one is typically interested in the posterior probabilities of $\delta \in \Delta$, as measures of the model plausibility to explain the information carried by the data.
- Note that the prior typically acts like Occham's razor, i.e. *regularizes* the model dimension through $\int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta)$ (more about that on next slide).

Primer on Bayesian model selection - III

- In structural model learning one is typically interested in the posterior probabilities of $\delta \in \Delta$, as measures of the model plausibility to explain the information carried by the data.
- Note that the prior typically acts like Occham's razor, i.e. *regularizes* the model dimension through $\int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta)$ (more about that on next slide).
- The posterior over models is defined as

$$p(\delta|\mathbf{x}) = \frac{p(\delta)p(\mathbf{x}|\delta)}{\sum_{\delta \in \Delta} p(\delta)p(\mathbf{x}|\delta)}$$

Primer on Bayesian model selection - III

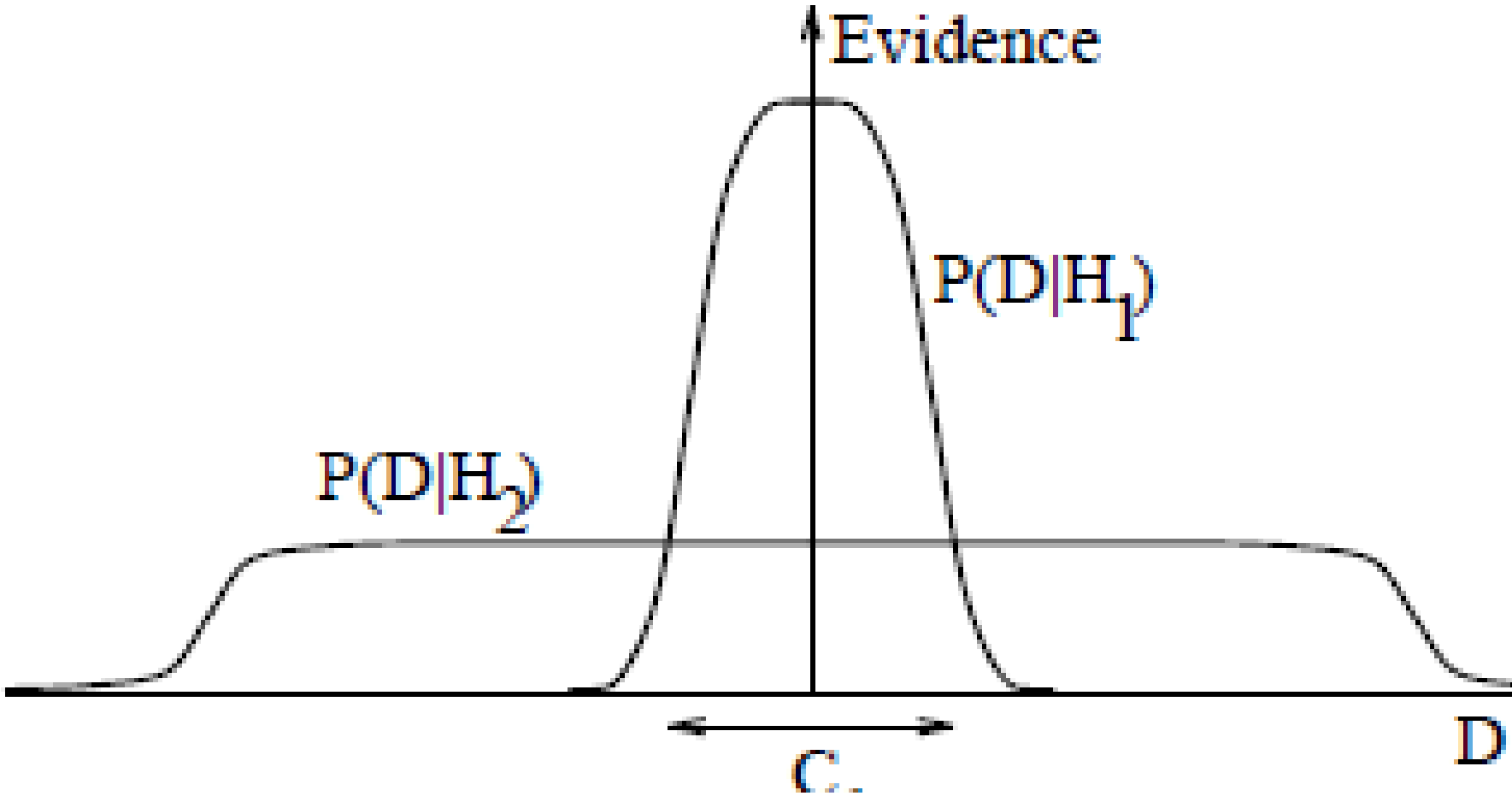
- In structural model learning one is typically interested in the posterior probabilities of $\delta \in \Delta$, as measures of the model plausibility to explain the information carried by the data.
- Note that the prior typically acts like Occham's razor, i.e. *regularizes* the model dimension through $\int_{\Theta_\delta} p(\mathbf{x}|\theta_\delta) d\mu(\theta_\delta)$ (more about that on next slide).
- The posterior over models is defined as

$$p(\delta|\mathbf{x}) = \frac{p(\delta)p(\mathbf{x}|\delta)}{\sum_{\delta \in \Delta} p(\delta)p(\mathbf{x}|\delta)}$$

- Calculation of the posterior probabilities is in general intractable due to the size of Δ, but they can be approximated with MCMC, or one can e.g. try to estimate the mode

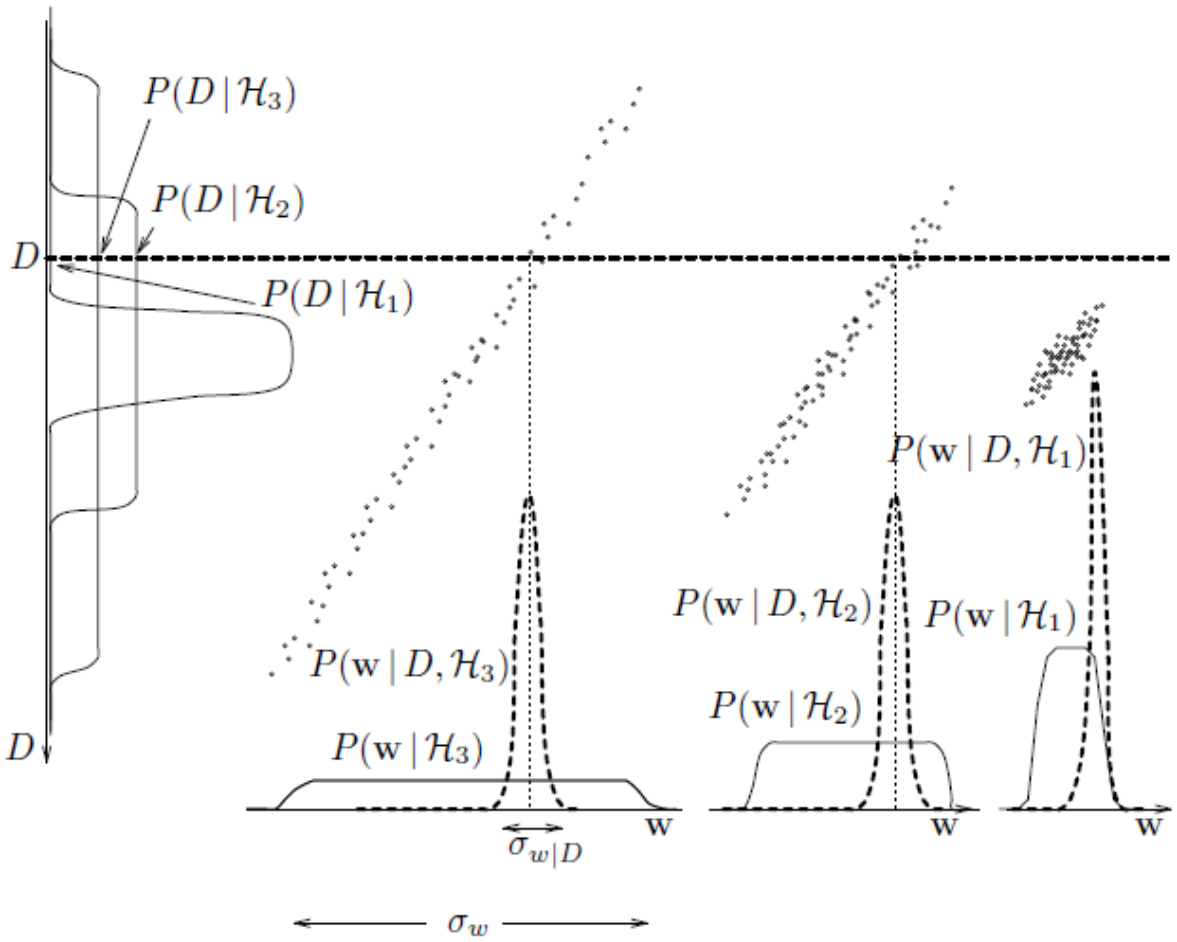
$$\arg\max_{\delta \in \Delta} p(\delta|\mathbf{x})$$

Model regularization - a primer by David MacKay, part I



Bayesian models specify a predictive distribution in the data space!

Model regularization - a primer by David MacKay, part II



Approximating model posterior with MCMC

- One of the most commonly used MCMC algorithms is the Metropolis-Hastings (MH) algorithm.

Approximating model posterior with MCMC

- One of the most commonly used MCMC algorithms is the Metropolis-Hastings (MH) algorithm.
- It is defined through the following transition kernel, governing the probability of transition from the current state δ_t to a proposal state δ^* as:

$$\min\left(1, \frac{p(\delta^*)p(\mathbf{x}|\delta^*)}{p(\delta_t)p(\mathbf{x}|\delta_t)}\frac{q(\delta_t|\delta^*)}{q(\delta^*|\delta_t)}\right),$$

Approximating model posterior with MCMC

- One of the most commonly used MCMC algorithms is the Metropolis-Hastings (MH) algorithm.
- It is defined through the following transition kernel, governing the probability of transition from the current state δ_t to a proposal state δ^* as:

$$\min\left(1, \frac{p(\delta^*)p(\mathbf{x}|\delta^*)}{p(\delta_t)p(\mathbf{x}|\delta_t)}\frac{q(\delta_t|\delta^*)}{q(\delta^*|\delta_t)}\right),$$

- where $q(\delta^*|\delta_t)$ is the probability of choosing state δ^* as the candidate for the next state at δ_t,

Approximating model posterior with MCMC

- One of the most commonly used MCMC algorithms is the Metropolis-Hastings (MH) algorithm.
- It is defined through the following transition kernel, governing the probability of transition from the current state δ_t to a proposal state δ^* as:

$$\min\left(1, \frac{p(\delta^*)p(\mathbf{x}|\delta^*)}{p(\delta_t)p(\mathbf{x}|\delta_t)}\frac{q(\delta_t|\delta^*)}{q(\delta^*|\delta_t)}\right),$$

- where $q(\delta^*|\delta_t)$ is the probability of choosing state δ^* as the candidate for the next state at δ_t,
- and $q(\delta_t|\delta^*)$ is the probability of restoration of the current state.

Approximating model posterior with MCMC

- When the MH proposal mechanism is deliberately chosen, the algorithm can be used to generate an aperiodic, irreducible, and reversible Markov chain, whose time homogeneous distribution equals the posterior distribution.

Approximating model posterior with MCMC

- When the MH proposal mechanism is deliberately chosen, the algorithm can be used to generate an aperiodic, irreducible, and reversible Markov chain, whose time homogeneous distribution equals the posterior distribution.
- From a realization $\{\delta_t, t = 0, 1, ...\}$ of such a chain, the posterior probabilities can be consistently estimated as

$$p_T(\delta|\mathbf{x}) = T^{-1} \sum_{t=1}^{n} I(\delta_t = \delta),$$

such that $p_T(\delta|\mathbf{x}) \longrightarrow p(\delta|\mathbf{x})$, as $T \longrightarrow \infty$.

Potential issues in MCMC

- Construction of efficient proposals $q(\delta^* | \delta_t)$ to avoid low acceptance rate.

Potential issues in MCMC

- Construction of efficient proposals $q(\delta^* | \delta_t)$ to avoid low acceptance rate.
- Slow convergence to the relevant areas of the posterior.

Potential issues in MCMC

- Construction of efficient proposals $q(\delta^*|\delta_t)$ to avoid low acceptance rate.
- Slow convergence to the relevant areas of the posterior.
- High variance of estimates of functions of $p(\delta|\mathbf{x})$.

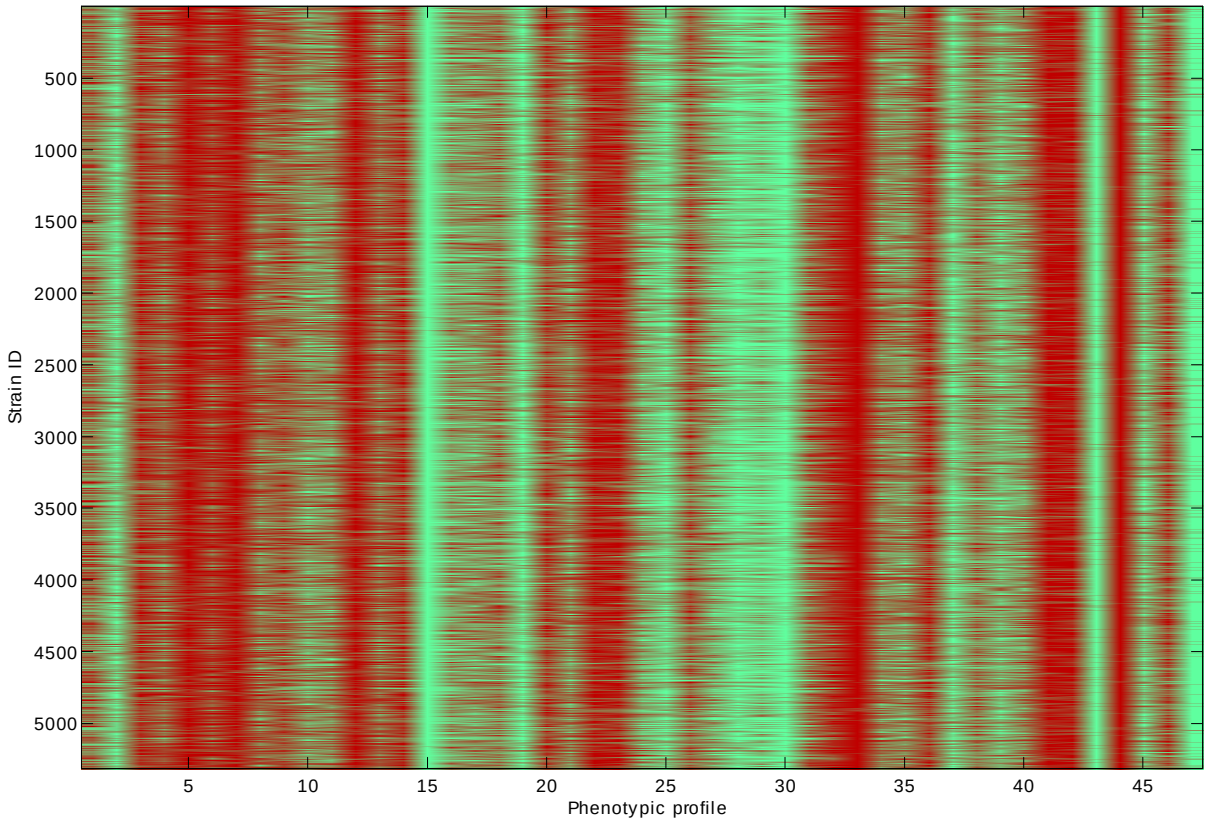
Potential issues in MCMC

- Construction of efficient proposals $q(\delta^*|\delta_t)$ to avoid low acceptance rate.
- Slow convergence to the relevant areas of the posterior.
- High variance of estimates of functions of $p(\delta|\mathbf{x})$.
- Missing posterior mode completely in practice.

Potential issues in MCMC

- Construction of efficient proposals $q(\delta^*|\delta_t)$ to avoid low acceptance rate.
- Slow convergence to the relevant areas of the posterior.
- High variance of estimates of functions of $p(\delta|\mathbf{x})$.
- Missing posterior mode completely in practice.
- Solutions?

What doesn't kill you will only make you stronger (Stronger - Kanye West)!



Motivating example - clustering with stochastic partitions

- Consider a set N of n data items that should be clustered.

Motivating example - clustering with stochastic partitions

- Consider a set N of n data items that should be clustered.
- For $i \in N$ we have available finite feature vectors $\mathbf{x}_i$, such that each element x_{ij} in $\mathbf{x}_i$ belongs to a discrete alphabet, $x_{ij} \in \mathcal{X}_j = \{1, ..., r_j\}$, $r_j \geq 2$, $j = 1, ..., d$.

Motivating example - clustering with stochastic partitions

- Consider a set N of n data items that should be clustered.
- For $i \in N$ we have available finite feature vectors $\mathbf{x}_i$, such that each element x_{ij} in $\mathbf{x}_i$ belongs to a discrete alphabet, $x_{ij} \in \mathcal{X}_j = \{1, ..., r_j\}$, $r_j \geq 2$, $j = 1, ..., d$.
- For any non-empty subset $s \subseteq N$ of the items, let $\mathbf{x}^{(s)}$ denote jointly their feature vectors.

Motivating example - clustering with stochastic partitions

- Consider a set N of n data items that should be clustered.
- For $i \in N$ we have available finite feature vectors $\mathbf{x}_i$, such that each element x_{ij} in $\mathbf{x}_i$ belongs to a discrete alphabet, $x_{ij} \in \mathcal{X}_j = \{1, ..., r_j\}$, $r_j \geq 2$, $j = 1, ..., d$.
- For any non-empty subset $s \subseteq N$ of the items, let $\mathbf{x}^{(s)}$ denote jointly their feature vectors.
- Let $S = (s_1, ..., s_k)$, $1 \leq k \leq n$, be a partition (=clustering) of N.

Motivating example - clustering with stochastic partitions

- Consider a set N of n data items that should be clustered.
- For $i \in N$ we have available finite feature vectors $\mathbf{x}_i$, such that each element x_{ij} in $\mathbf{x}_i$ belongs to a discrete alphabet, $x_{ij} \in \mathcal{X}_j = \{1, ..., r_j\}$, $r_j \geq 2$, $j = 1, ..., d$.
- For any non-empty subset $s \subseteq N$ of the items, let $\mathbf{x}^{(s)}$ denote jointly their feature vectors.
- Let $S = (s_1, ..., s_k)$, $1 \leq k \leq n$, be a partition (=clustering) of N.
- In general we don't know in advance what k should be representative for our data.

Clustering with stochastic partitions

- Assume we have a stochastic candidate partition S of N.

Clustering with stochastic partitions

- Assume we have a stochastic candidate partition S of N.
- Allocation of a particular subset of the items into the same cluster s_c implies the use of a common predictive model $p(\mathbf{x}^{(s_c)})$ for their observed features $\mathbf{x}^{(s_c)}$, i.e.

$$p(\mathbf{x}^{(s_c)}) = \int_{\Theta_{s_c}} p(\mathbf{x}^{(s_c)}|\theta_{s_c}) d\mu(\theta_{s_c})$$

Clustering with stochastic partitions

- Assume we have a stochastic candidate partition S of N.
- Allocation of a particular subset of the items into the same cluster s_c implies the use of a common predictive model $p(\mathbf{x}^{(s_c)})$ for their observed features $\mathbf{x}^{(s_c)}$, i.e.

$$p(\mathbf{x}^{(s_c)}) = \int_{\Theta_{s_c}} p(\mathbf{x}^{(s_c)}|\theta_{s_c}) d\mu(\theta_{s_c})$$

- Under certain form of exchangeability (more later) this leads to the predictive model $p(\mathbf{x}^{(N)}|S) = \prod_{c=1}^{k} p(\mathbf{x}^{(s_c)})$ for all data, conditional on the partition S.

Bayesian predictive learning in clustering

- The basis of predictive learning is quantified by the probability measure

$$p(\mathbf{x}^{(N)}) = \sum_{S \in \mathcal{S}} p(\mathbf{x}^{(N)}|S)p(S), \tag{1}$$

where $p(S)$ describes the *a priori* uncertainty about S, and $p(\mathbf{x}^{(N)}|S)$ is the (prior) predictive distribution of the feature data given the clustering S.

Bayesian predictive learning in clustering

- The basis of predictive learning is quantified by the probability measure

$$p(\mathbf{x}^{(N)}) = \sum_{S \in \mathcal{S}} p(\mathbf{x}^{(N)}|S)p(S), \tag{1}$$

  where $p(S)$ describes the *a priori* uncertainty about S, and $p(\mathbf{x}^{(N)}|S)$ is the (prior) predictive distribution of the feature data given the clustering S.

- The conditional (posterior) distribution of S given the data is determined by Bayes' rule:

$$p(S|\mathbf{x}^{(N)}) = \frac{p(\mathbf{x}^{(N)}|S)p(S)}{\sum_{S \in \mathcal{S}} p(\mathbf{x}^{(N)}|S)p(S)}.$$

Predictive model under generalized exchangeability

- To obtain a more concrete form for the predictive distribution, various assumptions may be utilized.

Predictive model under generalized exchangeability

- To obtain a more concrete form for the predictive distribution, various assumptions may be utilized.
- In the most basic model, we assume that the observed sequences of features $\mathbf{x}^{(s)}$ are *unrestrictedly infinitely exchangeable* (Bernardo and Smith, 1994).

Predictive model under generalized exchangeability

- To obtain a more concrete form for the predictive distribution, various assumptions may be utilized.
- In the most basic model, we assume that the observed sequences of features $\mathbf{x}^{(s)}$ are *unrestrictedly infinitely exchangeable* (Bernardo and Smith, 1994).
- Then, we obtain a unique probabilistic characterization of the cluster data as

$$p(\mathbf{x}^{(s)}) = \int_{\Theta_{sc}} \prod_{j=1}^{d} \prod_{l=1}^{r_j} p_{jl}^{n_{jl}} p(\theta_{s_c}),$$

where p_{jl} is understood as the limit of the relative frequency obtained through the sufficient statistics n_{jl}

$$p_{jl} = \lim_{|s| \to \infty} \frac{n_{jl}}{|s|}.$$

Predictive model under generalized exchangeability

- To obtain a more concrete form for the predictive distribution, various assumptions may be utilized.
- In the most basic model, we assume that the observed sequences of features $\mathbf{x}^{(s)}$ are *unrestrictedly infinitely exchangeable* (Bernardo and Smith, 1994).
- Then, we obtain a unique probabilistic characterization of the cluster data as

$$p(\mathbf{x}^{(s)}) = \int_{\Theta_{s_c}} \prod_{j=1}^{d} \prod_{l=1}^{r_j} p_{jl}^{n_{jl}} p(\theta_{s_c}),$$

  where p_{jl} is understood as the limit of the relative frequency obtained through the sufficient statistics n_{jl}

$$p_{jl} = \lim_{|s| \to \infty} \frac{n_{jl}}{|s|}.$$

- The symbol θ_{s_c} refers jointly to all parameters p_{jl} over different values and features.

Predictive model under generalized exchangeability

- By extending the unrestricted exchangeability assumption to hold over the clusters $s_1, \ldots, s_k$, we obtain the joint probabilistic characterization for $\mathbf{x}^{(N)}$ as

$$p(\mathbf{x}^{(N)}|S) = \int_{\Theta} \prod_{c=1}^{k} \prod_{j=1}^{d} \prod_{l=1}^{r_j} p_{cjl}^{n_{cjl}} p(\theta|S), \tag{2}$$

where n_{cjl} represents now the number of copies of value l for feature j observed among the items in s_c, and p_{cjl}, θ, and $p(\theta|S)$, are defined analogously to the previous slide.

Predictive model under generalized exchangeability

- By extending the unrestricted exchangeability assumption to hold over the clusters $s_1, ..., s_k$, we obtain the joint probabilistic characterization for $\mathbf{x}^{(N)}$ as

$$p(\mathbf{x}^{(N)}|S) = \int_{\Theta} \prod_{c=1}^{k} \prod_{j=1}^{d} \prod_{l=1}^{r_j} p_{cjl}^{n_{cjl}} p(\theta|S), \quad (2)$$

where n_{cjl} represents now the number of copies of value l for feature j observed among the items in s_c, and p_{cjl}, θ, and $p(\theta|S)$, are defined analogously to the previous slide.

- This model was formally derived in a molecular biological context in Corander et al. (2007, Bull Math Biol).

Predictive model under generalized exchangeability

- By extending the unrestricted exchangeability assumption to hold over the clusters $s_1, ..., s_k$, we obtain the joint probabilistic characterization for $\mathbf{x}^{(N)}$ as

$$p(\mathbf{x}^{(N)}|S) = \int_{\Theta} \prod_{c=1}^{k} \prod_{j=1}^{d} \prod_{l=1}^{r_j} p_{cjl}^{n_{cjl}} p(\theta|S), \quad (2)$$

where n_{cjl} represents now the number of copies of value l for feature j observed among the items in s_c, and p_{cjl}, θ, and $p(\theta|S)$, are defined analogously to the previous slide.

- This model was formally derived in a molecular biological context in Corander et al. (2007, Bull Math Biol).
- It is a representation of the statistical uncertainty arising about genetic population structure under the assumptions of unlinked markers and random mating populations (k is unknown).

Predictive model under generalized exchangeability

- To finally obtain an explicit form of the predictive density, we specify the prior beliefs $p(\theta|S)$, according to the product Dirichlet distribution

$$Q(\boldsymbol{\theta}|S) \propto \prod_{c=1}^{k} \prod_{j=1}^{d} \prod_{l=1}^{r_j} p_{cjl}^{\lambda_{cjl}-1},$$

with a suitably determined hyperparameter $\lambda_{cjl} > 0$, for all index values.

Predictive model under generalized exchangeability

- To finally obtain an explicit form of the predictive density, we specify the prior beliefs $p(\theta|S)$, according to the product Dirichlet distribution

$$Q(\boldsymbol{\theta}|S) \propto \prod_{c=1}^{k} \prod_{j=1}^{d} \prod_{l=1}^{r_j} p_{cjl}^{\lambda_{cjl}-1},$$

  with a suitably determined hyperparameter $\lambda_{cjl} > 0$, for all index values.

- Then, the explicit predictive probability for the feature data equals

$$p(\mathbf{x}^{(N)}|S) = \prod_{c=1}^{k} \prod_{j=1}^{d} \frac{\Gamma(\sum_{l=1}^{r_j} \lambda_{cjl})}{\Gamma(\sum_{l=1}^{r_j} \lambda_{cjl} + n_{cjl})} \prod_{l=1}^{r_j} \frac{\Gamma(\lambda_{cjk} + n_{cjl})}{\Gamma(\lambda_{cjl})}.$$

Predictive learning in practice

- The practical applicability of the Bayesian learning approach is dependent on our ability to identify classifications associated with high posterior probabilities.

Predictive learning in practice

- The practical applicability of the Bayesian learning approach is dependent on our ability to identify classifications associated with high posterior probabilities.
- In general, complete enumeration of $\mathcal{S}$ is not feasible, and therefore, an algorithm-based method is necessary to obtain estimates such as the posterior mode

$$\hat{S} = \arg\max_{S \in \mathcal{S}} p(S|\mathbf{x}^{(N)}).$$

Predictive learning in practice

- The practical applicability of the Bayesian learning approach is dependent on our ability to identify classifications associated with high posterior probabilities.
- In general, complete enumeration of $\mathcal{S}$ is not feasible, and therefore, an algorithm-based method is necessary to obtain estimates such as the posterior mode

$$\hat{S} = \arg\max_{S \in \mathcal{S}} p(S|\mathbf{x}^{(N)}).$$

- A solution to such an estimation problem was suggested by Corander *et al.* (2004), where a MH algorithm for searching the space of partitions was introduced.

Predictive learning in practice

- The practical applicability of the Bayesian learning approach is dependent on our ability to identify classifications associated with high posterior probabilities.
- In general, complete enumeration of $\mathcal{S}$ is not feasible, and therefore, an algorithm-based method is necessary to obtain estimates such as the posterior mode

$$\hat{S} = \arg\max_{S \in \mathcal{S}} p(S|\mathbf{x}^{(N)}).$$

- A solution to such an estimation problem was suggested by Corander *et al.* (2004), where a MH algorithm for searching the space of partitions was introduced.
- One can use also more advanced estimators that average the evidence of merging items to the same cluster, but these are often computationally very demanding.

Ordinary Metropolis-Hastings learning

- A Metropolis-Hastings algorithm for learning classifications can be defined by the transition kernel of a Markov chain, which determines the probability of a transition from a current classification S to a new proposal classification S^*, as

$$\min\left(1, \frac{p(\mathbf{x}^{(N)}|S^*)}{p(\mathbf{x}^{(N)}|S)} \frac{q(S|S^*)}{q(S^*|S)}\right), \tag{3}$$

where $q(S^*|S)$ is the probability of choosing classification S^* as the new candidate when in S, and $q(S|S^*)$ is the probability of restoration of the current classification S.

Ordinary Metropolis-Hastings learning

- The proposal mechanism to derive S^* from S considered by Corander *et al.* (2004) was constructed from the following four different possibilities, that are similar to those commonly used in reversible jump -samplers:

Ordinary Metropolis-Hastings learning

- The proposal mechanism to derive S^* from S considered by Corander *et al.* (2004) was constructed from the following four different possibilities, that are similar to those commonly used in reversible jump -samplers:
- With probability $1/2$, merge two randomly chosen classes s_c, s_{c*}.

Ordinary Metropolis-Hastings learning

- The proposal mechanism to derive S^* from S considered by Corander *et al.* (2004) was constructed from the following four different possibilities, that are similar to those commonly used in reversible jump -samplers:
- With probability $1/2$, merge two randomly chosen classes s_c, s_{c*}.
- With probability $1/2$ split a randomly chosen class s_c into two new classes, whose cardinalities are uniformly distributed between 1 and $|s_c| - 1$, and whose elements are randomly chosen from s_c.

Ordinary Metropolis-Hastings learning

- The proposal mechanism to derive S^* from S considered by Corander *et al.* (2004) was constructed from the following four different possibilities, that are similar to those commonly used in reversible jump -samplers:
- With probability $1/2$, merge two randomly chosen classes s_c, s_{c*}.
- With probability $1/2$ split a randomly chosen class s_c into two new classes, whose cardinalities are uniformly distributed between 1 and $|s_c| - 1$, and whose elements are randomly chosen from s_c.
- Move an arbitrary item from a randomly chosen class $s_c, |s_c| > 1$, into another randomly chosen class s_{c^*}.

Ordinary Metropolis-Hastings learning

- The proposal mechanism to derive S^* from S considered by Corander *et al.* (2004) was constructed from the following four different possibilities, that are similar to those commonly used in reversible jump -samplers:
- With probability $1/2$, merge two randomly chosen classes s_c, s_{c*}.
- With probability $1/2$ split a randomly chosen class s_c into two new classes, whose cardinalities are uniformly distributed between 1 and $|s_c| - 1$, and whose elements are randomly chosen from s_c.
- Move an arbitrary item from a randomly chosen class $s_c, |s_c| > 1$, into another randomly chosen class s_{c^*}.
- Choose one item randomly from each of two randomly chosen classes s_c and s_{c^*}, and exchange them between the classes.

Ordinary Metropolis-Hastings learning

- The proposal mechanism to derive S^* from S considered by Corander *et al.* (2004) was constructed from the following four different possibilities, that are similar to those commonly used in reversible jump -samplers:
- With probability $1/2$, merge two randomly chosen classes s_c, s_{c*}.
- With probability $1/2$ split a randomly chosen class s_c into two new classes, whose cardinalities are uniformly distributed between 1 and $|s_c| - 1$, and whose elements are randomly chosen from s_c.
- Move an arbitrary item from a randomly chosen class $s_c, |s_c| > 1$, into another randomly chosen class s_{c^*}.
- Choose one item randomly from each of two randomly chosen classes s_c and s_{c^*}, and exchange them between the classes.
- These kinds of proposals are very commonly used in Bayesian MCMC for variable-dimensional models.

Estimating the posterior

- For a realization of the Markov chain $\{S_t, t = 0, 1, ...\}$, the standard convergence result holds as

$$\lim_{T \to \infty} p_T(S|\mathbf{x}^{(N)}) = p(S|\mathbf{x}^{(N)}), \tag{4}$$

where

$$p_T(S|\mathbf{x}^{(N)}) = T^{-1} \sum_{t=1}^{T} I(S_t = S) \tag{5}$$

is the relative frequency of occurrence of state S.

Estimating the posterior

- For a realization of the Markov chain $\{S_t, t = 0, 1, ...\}$, the standard convergence result holds as

$$\lim_{T \to \infty} p_T(S|\mathbf{x}^{(N)}) = p(S|\mathbf{x}^{(N)}), \tag{4}$$

  where

$$p_T(S|\mathbf{x}^{(N)}) = T^{-1} \sum_{t=1}^{T} I(S_t = S) \tag{5}$$

  is the relative frequency of occurrence of state S.
- Also, we know that the true posterior optimum $\hat{S}$ will be identified when $T \to \infty$.

Estimating the posterior

- For a realization of the Markov chain $\{S_t, t = 0, 1, ...\}$, the standard convergence result holds as

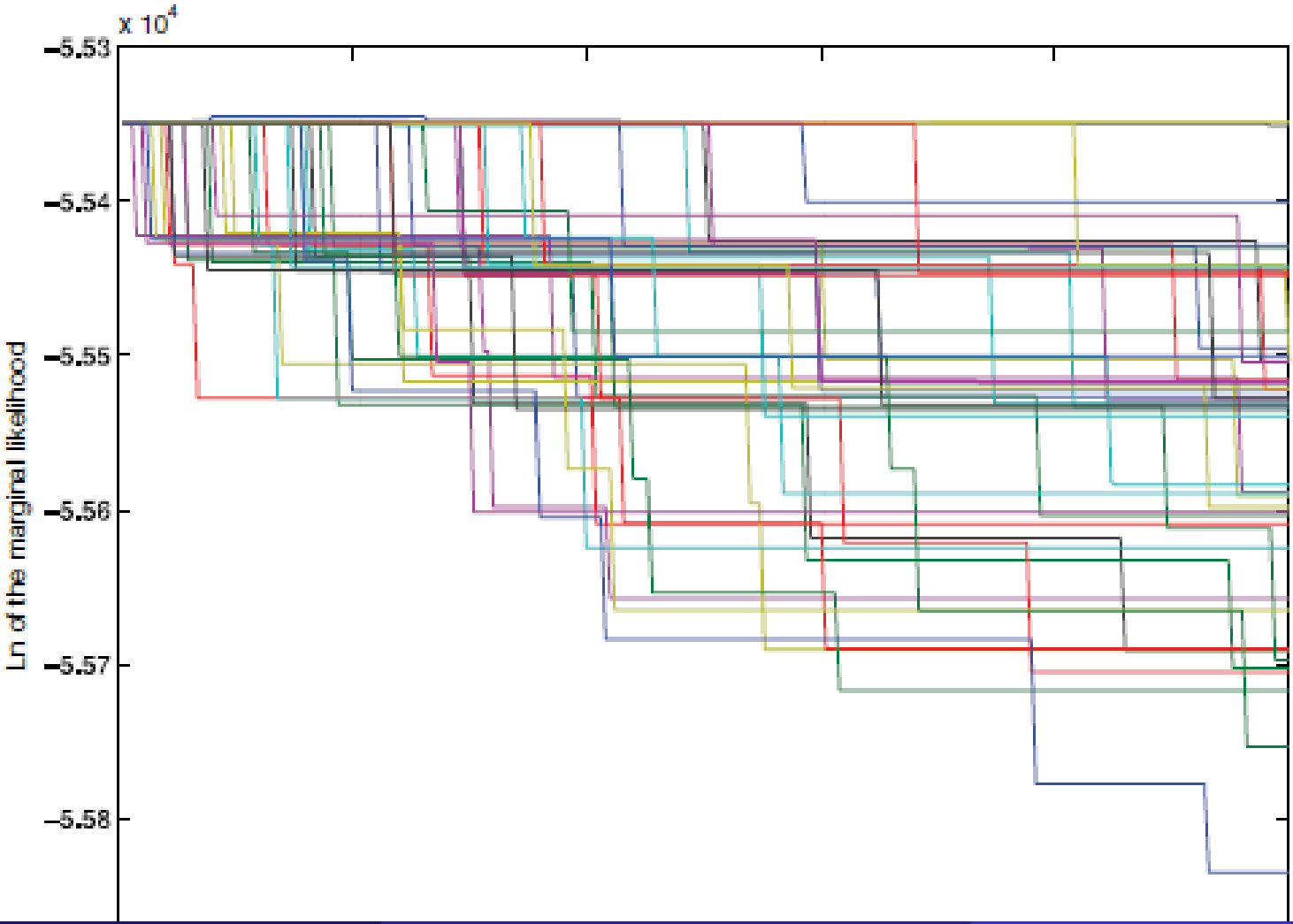
$$\lim_{T \to \infty} p_T(S|\mathbf{x}^{(N)}) = p(S|\mathbf{x}^{(N)}), \tag{4}$$

  where

$$p_T(S|\mathbf{x}^{(N)}) = T^{-1} \sum_{t=1}^{T} I(S_t = S) \tag{5}$$

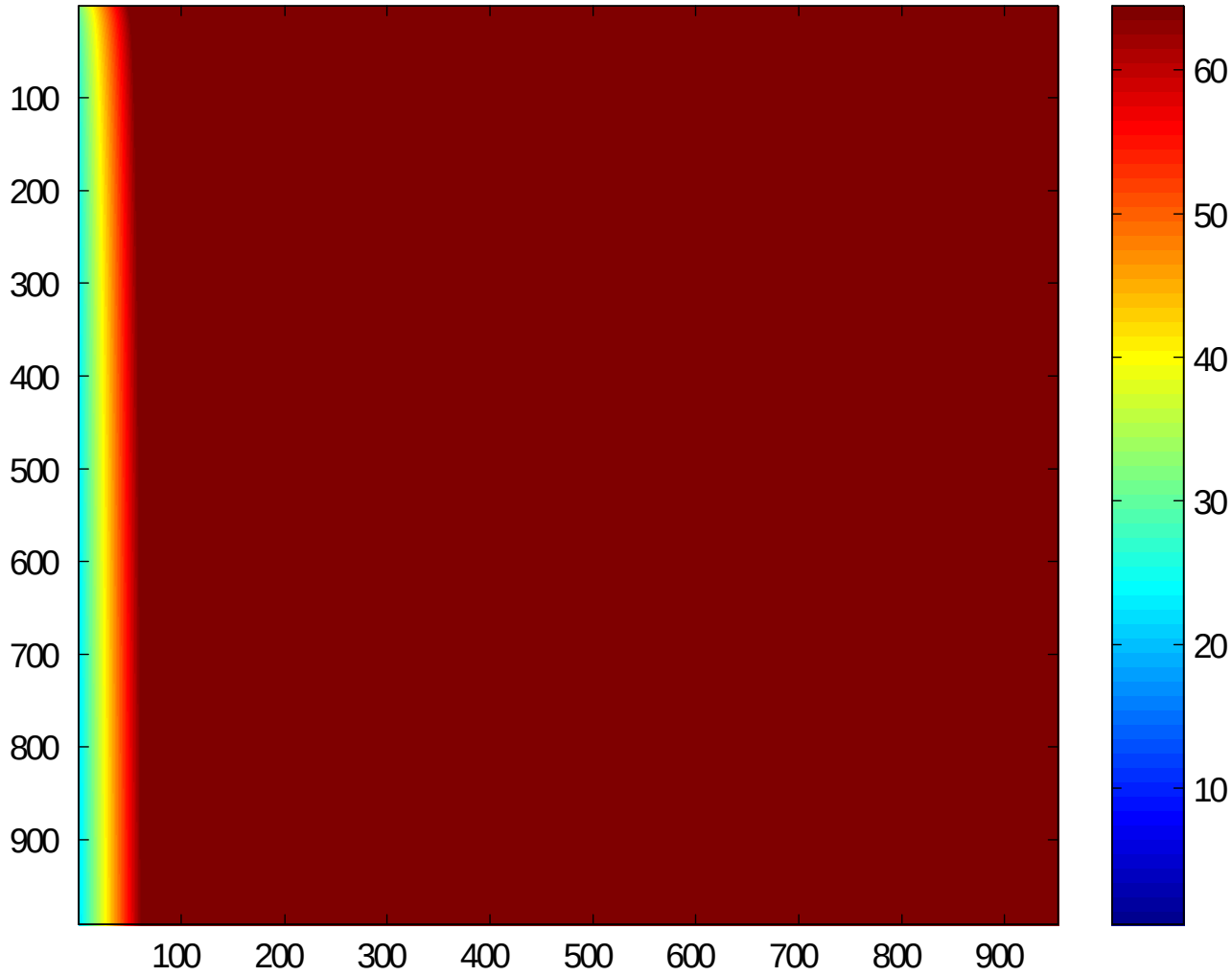
  is the relative frequency of occurrence of state S.
- Also, we know that the true posterior optimum $\hat{S}$ will be identified when $T \to \infty$.
- BUT, the above standard estimates can have enormous variance and be strongly biased in practice, when the state space is large (cf. phylogenetics and estimation of topologies).

What happened when the algorithm was applied to the data shown earlier?



Explicit illustration of the skweness of

$$\log \frac{q(S|S^*)}{q(S^*|S)}$$



How to solve the problem?

- Introduce a non-reversible MCMC algorithm for parallel learning.

How to solve the problem?

- Introduce a non-reversible MCMC algorithm for parallel learning.
- Consider a positive recurrent non-reversible Markov chain $\{S_t, t = 0, 1, ...\}$ with transition kernel

$$\min\left(1, \frac{p(\mathbf{x}^{(N)}|S^*)}{p(\mathbf{x}^{(N)}|S)}\right),$$

combined with fixed proposal distributions that satisfy certain constraints.

How to solve the problem?

- Introduce a non-reversible MCMC algorithm for parallel learning.
- Consider a positive recurrent non-reversible Markov chain $\{S_t, t = 0, 1, ...\}$ with transition kernel

$$\min\left(1, \frac{p(\mathbf{x}^{(N)}|S^*)}{p(\mathbf{x}^{(N)}|S)}\right),$$

  combined with fixed proposal distributions that satisfy certain constraints.
- In the simplest case, such a sampler can use the same proposal distributions as the previous one.

How to solve the problem?

- Introduce a non-reversible MCMC algorithm for parallel learning.
- Consider a positive recurrent non-reversible Markov chain $\{S_t, t = 0, 1, ...\}$ with transition kernel

$$\min\left(1, \frac{p(\mathbf{x}^{(N)}|S^*)}{p(\mathbf{x}^{(N)}|S)}\right),$$

  combined with fixed proposal distributions that satisfy certain constraints.
- In the simplest case, such a sampler can use the same proposal distributions as the previous one.
- But we can do even better by allowing multiple processes to learn from each other!

Nonreversible parallel sampler

- Let $\{S_{tj}, t = 0, 1, ...; j = 1, ..., m\}$ and $\{Z_t, t = 0, 1, ...\}$ be $m + 1$ stochastic processes defined as follows:

Nonreversible parallel sampler

- Let $\{S_{tj}, t = 0, 1, ...; j = 1, ..., m\}$ and $\{Z_t, t = 0, 1, ...\}$ be $m + 1$ stochastic processes defined as follows:
- *Define a sequence of strictly decreasing probabilities* $\{\alpha_t, t = 1, 2, ...\}$, *such that* $\alpha_t > \alpha_{t+1}$, *and* $\alpha_t \to 0$ *as* $t \to \infty$.

Nonreversible parallel sampler

- Let $\{S_{tj}, t = 0, 1, ...; j = 1, ..., m\}$ and $\{Z_t, t = 0, 1, ...\}$ be $m + 1$ stochastic processes defined as follows:
- *Define a sequence of strictly decreasing probabilities* $\{\alpha_t, t = 1, 2, ...\}$, *such that* $\alpha_t > \alpha_{t+1}$, *and* $\alpha_t \to 0$ *as* $t \to \infty$.
- *Define the stochastic process* $\{Z_t, t = 0, 1, ...\}$ *as* $Z_0 = 0$, *and* $P(Z_t = 1) = \alpha_t, P(Z_t = 0) = 1 - \alpha_t$, *independently for* $t = 1, 2, ...$.

Nonreversible parallel sampler

- Let $\{S_{tj}, t = 0, 1, ...; j = 1, ..., m\}$ and $\{Z_t, t = 0, 1, ...\}$ be $m + 1$ stochastic processes defined as follows:
- *Define a sequence of strictly decreasing probabilities* $\{\alpha_t, t = 1, 2, ...\}$, *such that* $\alpha_t > \alpha_{t+1}$, *and* $\alpha_t \to 0$ *as* $t \to \infty$.
- *Define the stochastic process* $\{Z_t, t = 0, 1, ...\}$ *as* $Z_0 = 0$, *and* $P(Z_t = 1) = \alpha_t, P(Z_t = 0) = 1 - \alpha_t$, *independently for* $t = 1, 2, ...$.
- *Let* $S_{0j}, j = 1, ..., m$, *be arbitrary initial states of* $\{S_{tj}, t = 0, 1, ...; j = 1, ..., m\}$.

Nonreversible parallel sampler

- *Given a realization* $\{Z_t, t = 0, 1, ...\}$*, the transition mechanism of the processes* $\{S_{tj}, t > 0; j = 1, ..., m\}$ *depends on values of* Z_t *according to the following.*

Nonreversible parallel sampler

- *Given a realization* $\{Z_t, t = 0, 1, ...\}$, *the transition mechanism of the processes* $\{S_{tj}, t > 0; j = 1, ..., m\}$ *depends on values of* Z_t *according to the following.*
- *For each* t, *such that* $Z_t = 0$, *transition from* S_{tj} *to the next state* $S_{(t+1)j}$ *is determined according to the nonreversible MH kernel, for* $j = 1, ..., m$.

Nonreversible parallel sampler

- *Given a realization* $\{Z_t, t = 0, 1, ...\}$, *the transition mechanism of the processes* $\{S_{tj}, t > 0; j = 1, ..., m\}$ *depends on values of* Z_t *according to the following.*
- *For each* t, *such that* $Z_t = 0$, *transition from* S_{tj} *to the next state* $S_{(t+1)j}$ *is determined according to the nonreversible MH kernel, for* $j = 1, ..., m$.
- *For each* t, *such that* $Z_t = 1$, *transition from* S_{tj} *to the next state* $S_{(t+1)j}$ *is determined according to the following distribution over the space* $\mathcal{S}_t = \{S_{tj}, j = 1, ..., m\}$ *of candidate states:*

$$P_t(S_{(t+1)j} = S_{tj}) = \frac{p(S_{tj})p(\mathbf{x}|S_{tj})}{\sum_{j=1}^{m} p(S_{tj})p(\mathbf{x}|S_{tj})},$$

independently for $j = 1, ..., m$.

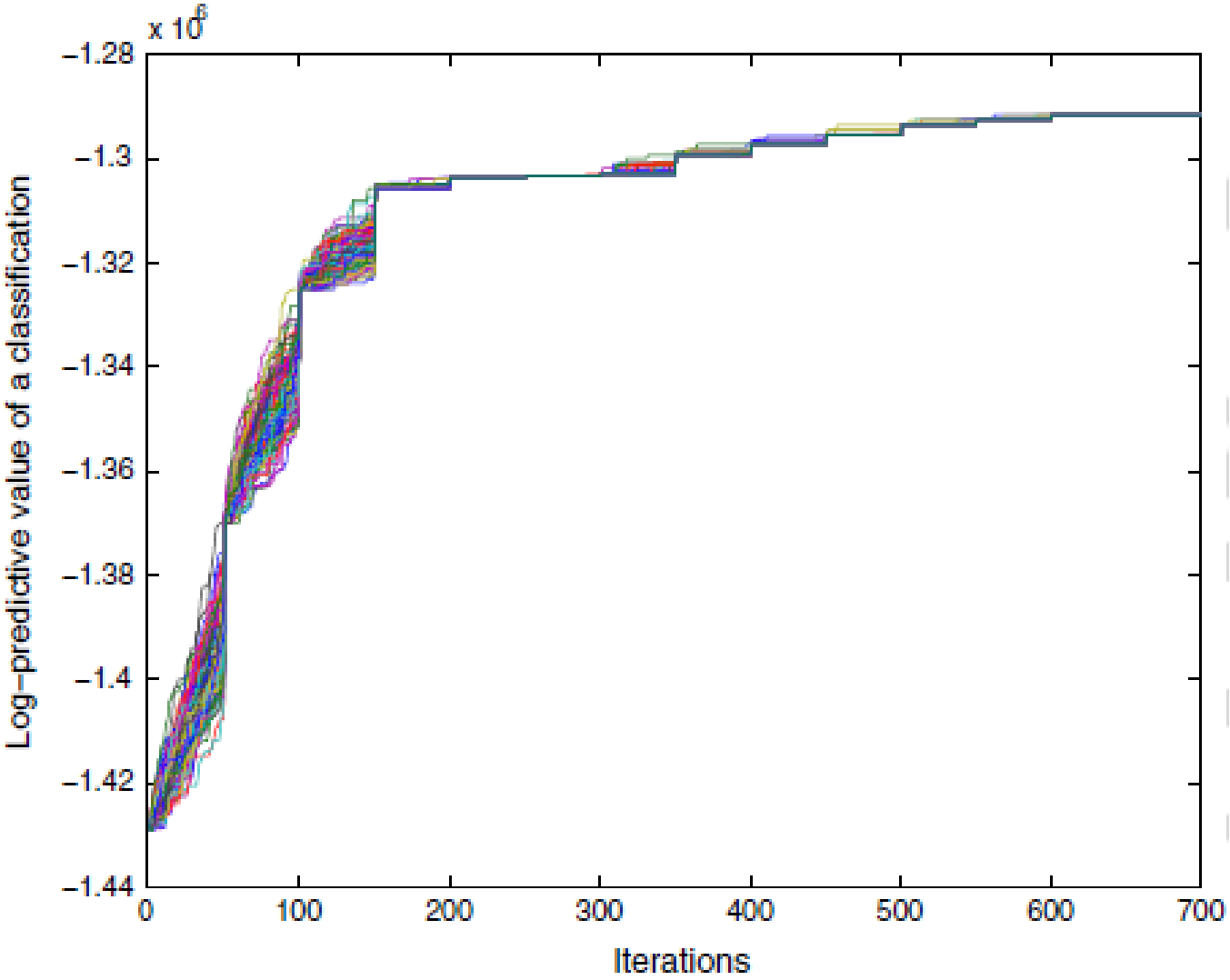
- Consistent estimates of $p(S|\mathbf{x}^{(N)})$ can be obtained using this sampler and it allows for smart search operators!

- Consistent estimates of $p(S|\mathbf{x}^{(N)})$ can be obtained using this sampler and it allows for smart search operators!
- The following estimate is consistent and it can have much smaller variance than the relative frequency based estimate:

$$\hat{p}(S|\mathbf{x}^{(N)}) = \frac{p(\mathbf{x}^{(N)}|S)p(S)}{\sum_{S \in \mathcal{S}_t} p(\mathbf{x}^{(N)}|S)p(S)},$$

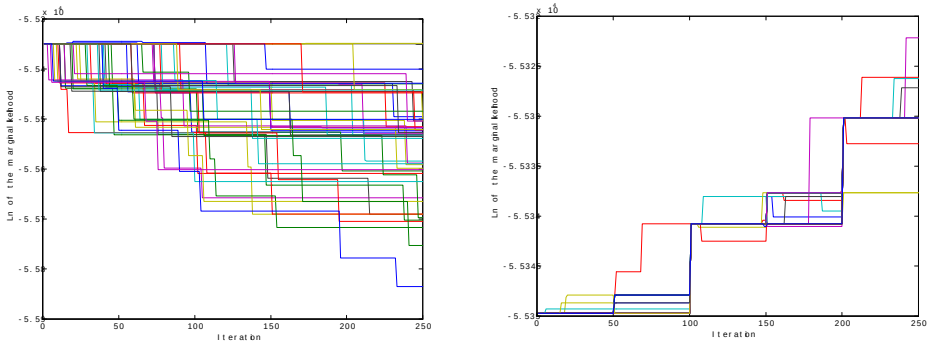
where $\mathcal{S}_n$ is the space of distinct states visited by time n by the non-reversible process.

An illustration of the behavior of parallel processes



$\log_e p(\mathbf{x}|S)$ (vertical axis) of 100 dependent processes for a molecular

Comparison between the reversible and nonreversible processes



Predictive abilities ($\log_e p(\mathbf{x}|S)$, vertical axis) of the Bayesian classification model for the *Enterobacteriaceae* data, over 250 iterations of 50 processes all started at the same initial configuration, produced with the reversible (left) and non-reversible (right) MH algorithms.

Optimization of dynamical graphical models stochastic nonreversible search vs hill-climbing

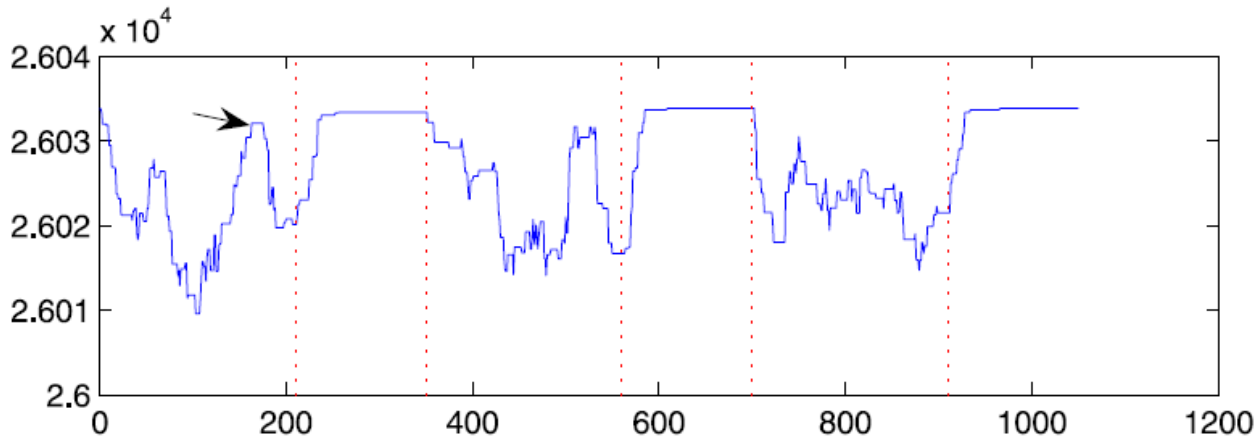
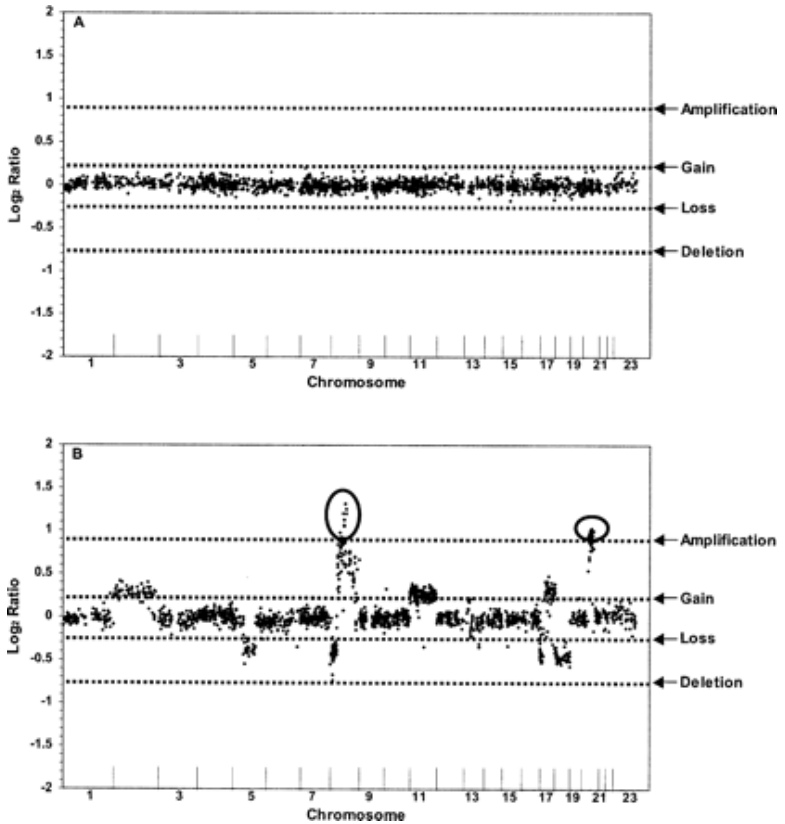


Fig. 4 Trace plot of BEC values of models visited during a search which was started from the optimal model for the air pollution data and carried out by repeating three times a sequence of six stochastic and four greedy iterations. The periods of stochastic and greedy iterations are separated by *dotted vertical lines*. The highest score found during the stochastic iterations is marked with an *arrow* and it is 5.7 times less probable than the overall optimal model

Marttinen and Corander (Machine Learning, 2009)

Bayesian profiling of cancer tissues using comparative genomic hybridization (CGH) data.



Control tissue

Cancer tissue

From Nakao et al. Carcinogenesis 2004.

Analyzing multiple CGH profiles

- Assume we have a database of binarized DNA alteration profiles for various cancer samples.

Analyzing multiple CGH profiles

- Assume we have a database of binarized DNA alteration profiles for various cancer samples.
- How do we:

Analyzing multiple CGH profiles

- Assume we have a database of binarized DNA alteration profiles for various cancer samples.
- How do we:
- Identify samples with similar profiles?

Analyzing multiple CGH profiles

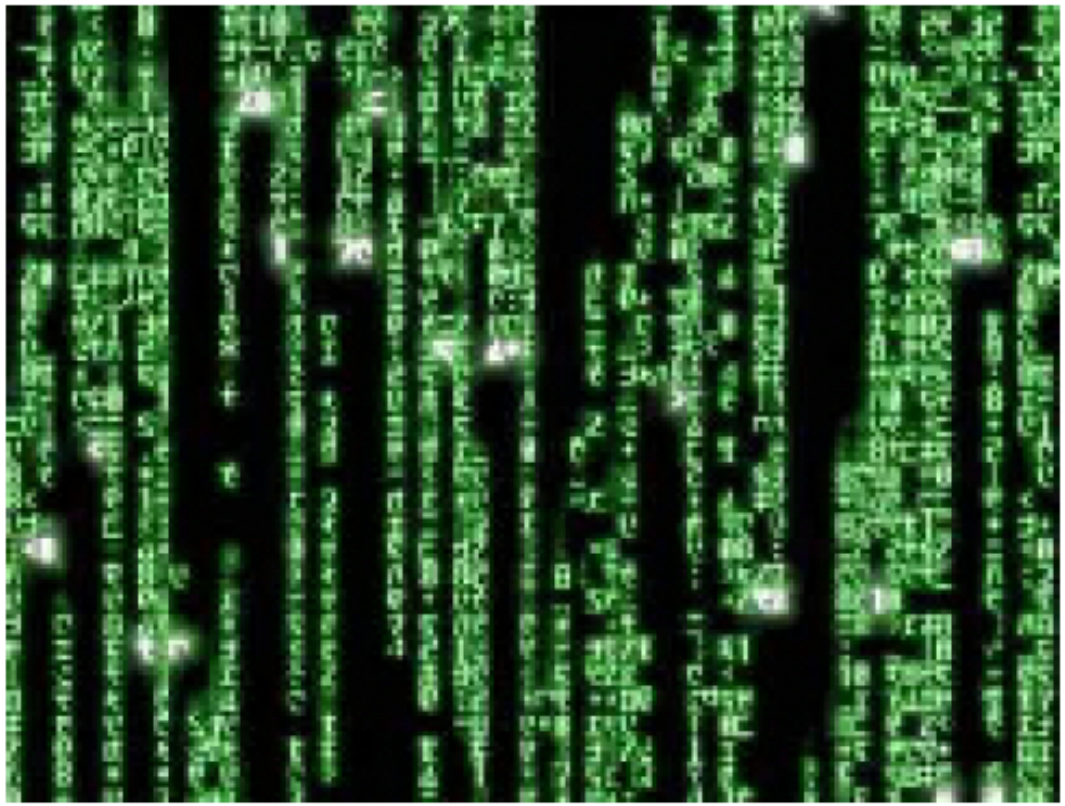
- Assume we have a database of binarized DNA alteration profiles for various cancer samples.
- How do we:
- Identify samples with similar profiles?
- Identify chromosomic areas that are signatures for certain cancer subtypes?

Analyzing multiple CGH profiles

- Assume we have a database of binarized DNA alteration profiles for various cancer samples.
- How do we:
- Identify samples with similar profiles?
- Identify chromosomic areas that are signatures for certain cancer subtypes?
- Is it really that difficult?

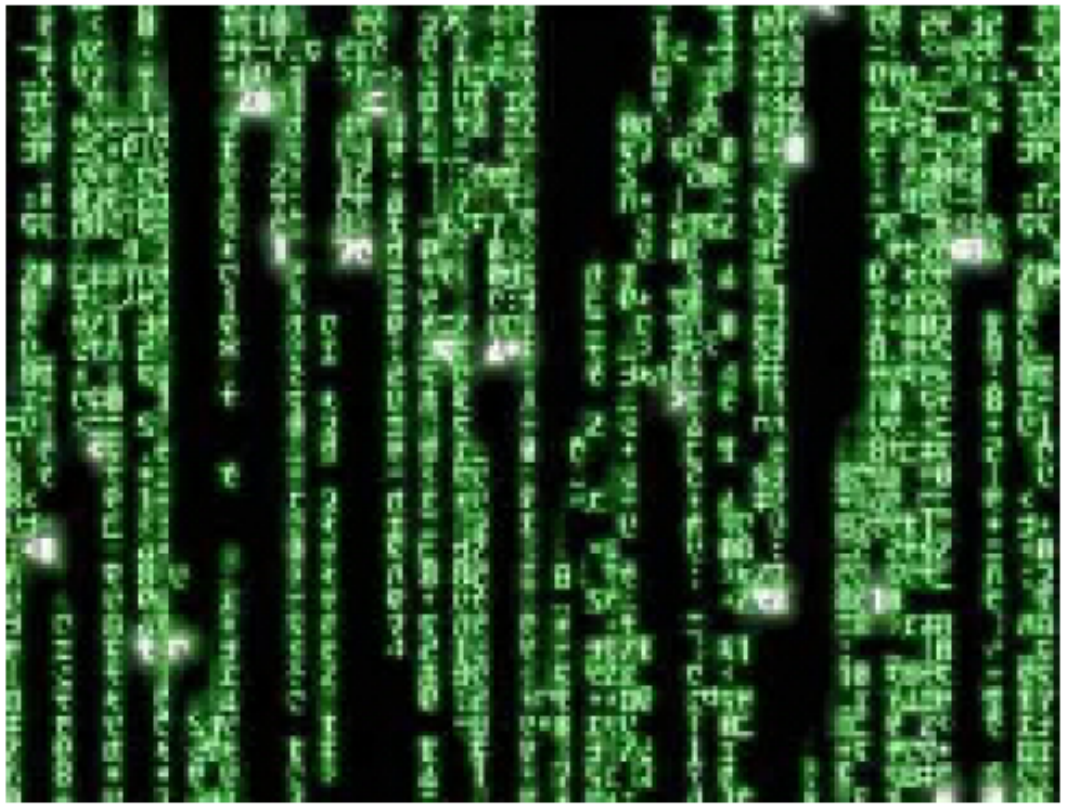
Let's try it out

- Can you see clearly what's in the picture below?



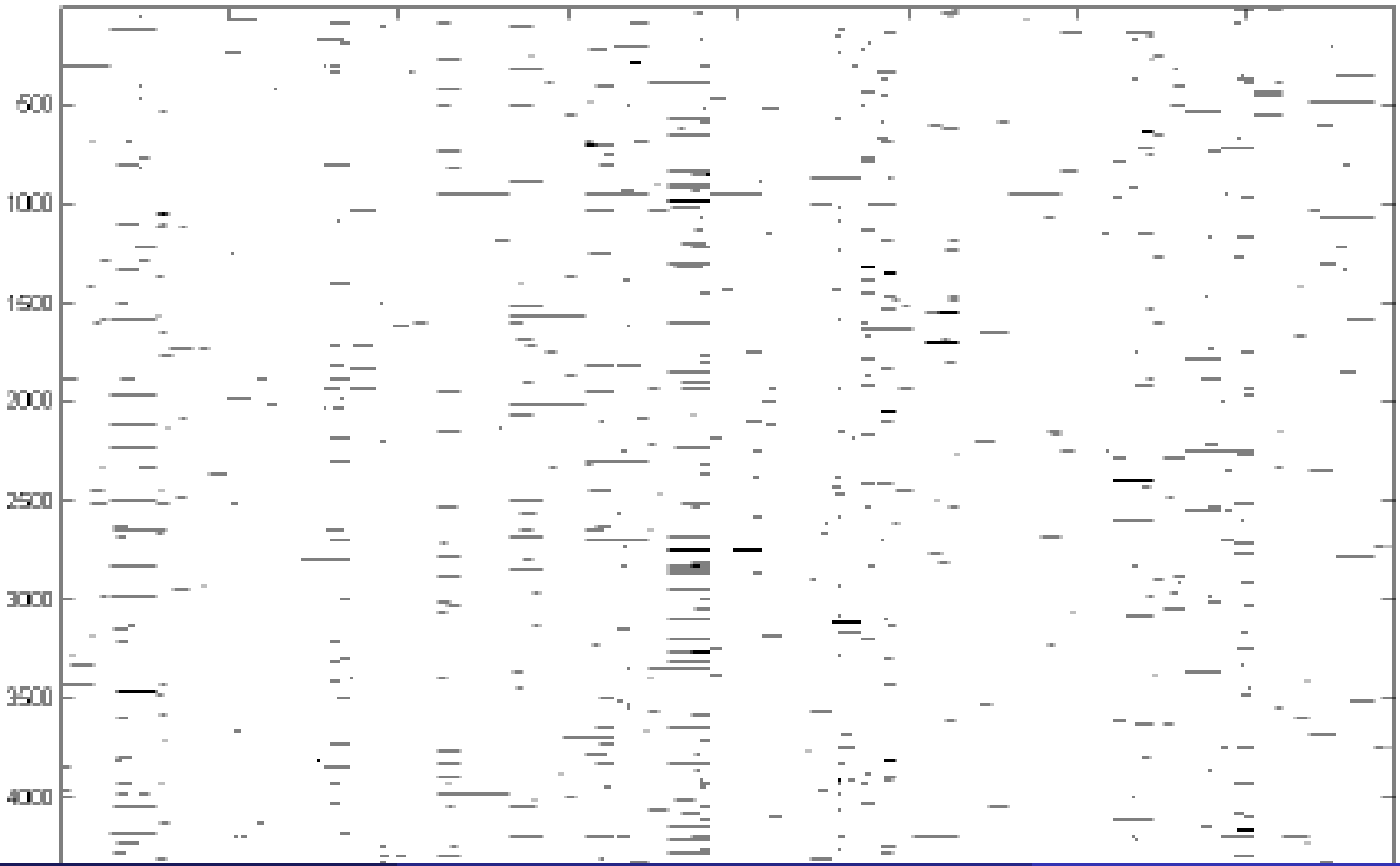
Let's try it out

- Can you see clearly what's in the picture below?



- Oops, wrong picture.

5000 cancer samples (rows) and amplifications (black) on sub-band resolution (columns). Easy to see the patterns, right?



But hey, use data mining tools!

- Various approaches to exploring such data exist.

But hey, use data mining tools!

- Various approaches to exploring such data exist.
- k-means algorithm, dendrograms, SOM...

But hey, use data mining tools!

- Various approaches to exploring such data exist.
- k-means algorithm, dendrograms, SOM...
- Easy to use and fast, but difficult to draw firm conclusions from the results.

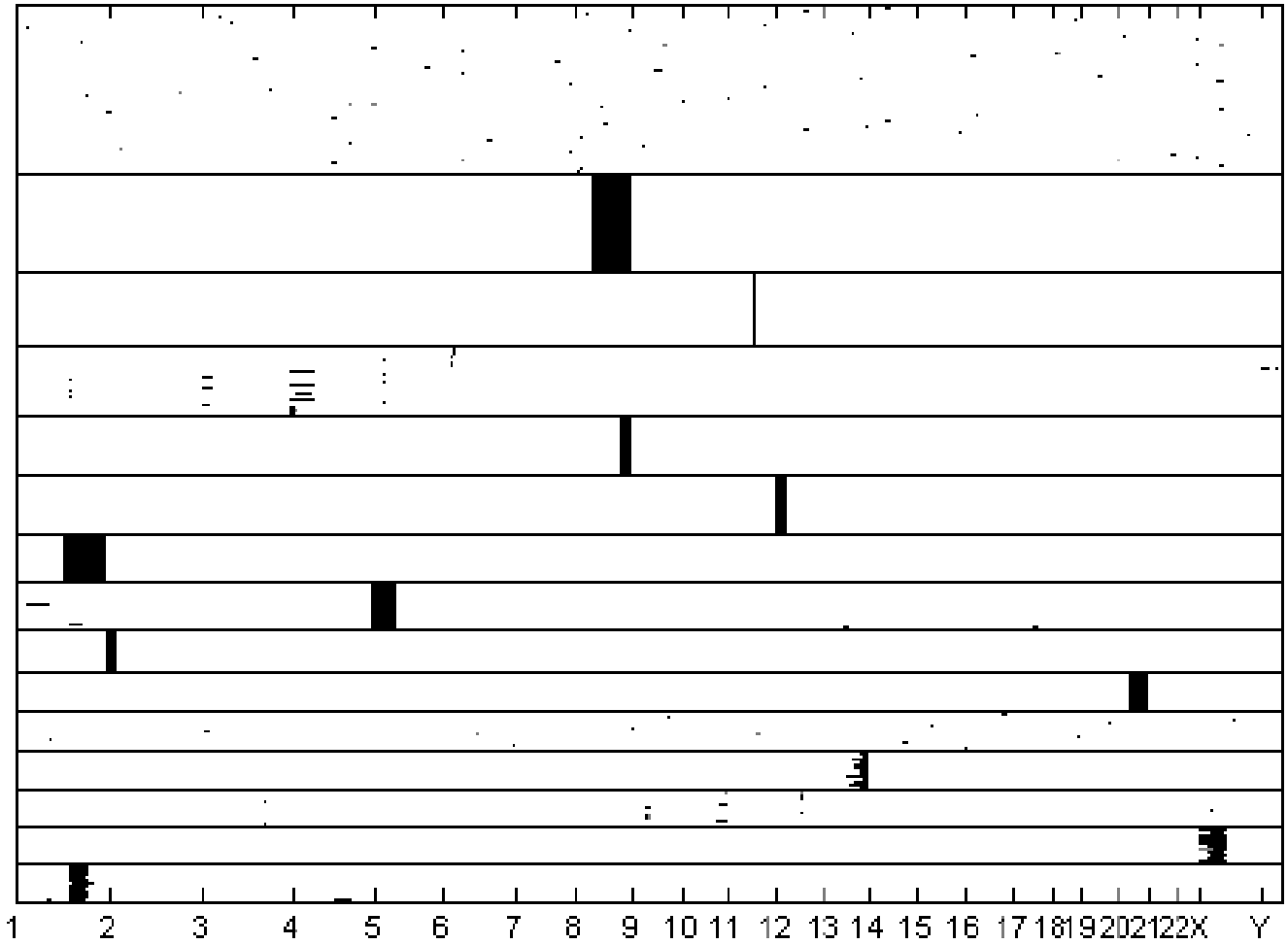
But hey, use data mining tools!

- Various approaches to exploring such data exist.
- k-means algorithm, dendrograms, SOM...
- Easy to use and fast, but difficult to draw firm conclusions from the results.
- Model-based methods, e.g. mixture models fitted with EM algorithm or Gibbs sampler.

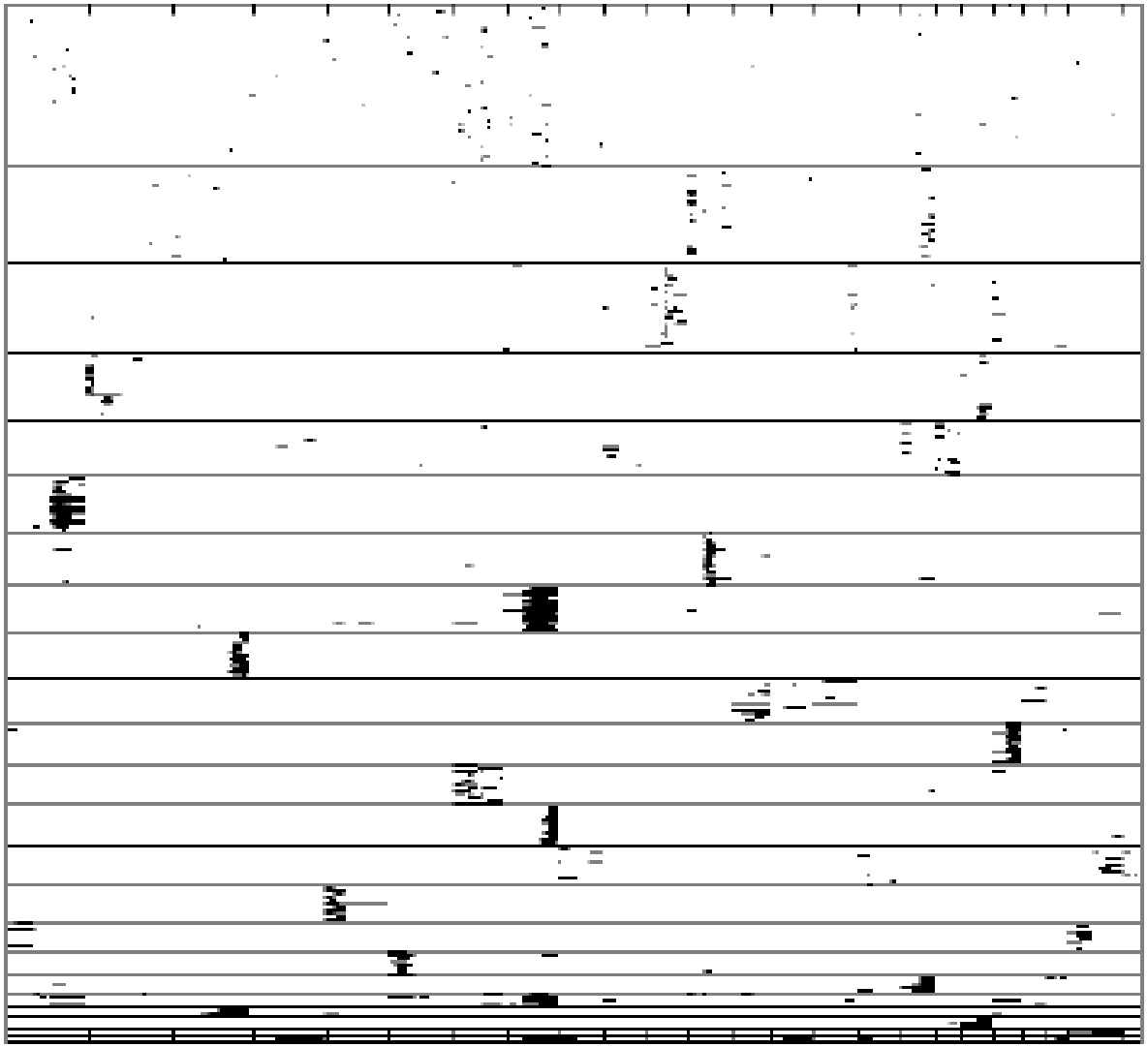
But hey, use data mining tools!

- Various approaches to exploring such data exist.
- k-means algorithm, dendrograms, SOM...
- Easy to use and fast, but difficult to draw firm conclusions from the results.
- Model-based methods, e.g. mixture models fitted with EM algorithm or Gibbs sampler.
- More challenging to use and slower, but may provide DIRECT focus on the biologically relevant questions.

Let's try k-means with, say, 291 clusters.



Oops, I forgot k-means cannot estimate k. Let's instead use a Bayesian mixture model which can learn k.



What's wrong?

- A standard mixture model assumes that all observed features contribute to the grouping of the samples.

What's wrong?

- A standard mixture model assumes that all observed features contribute to the grouping of the samples.
- If not true, results may be strongly biased.

What's wrong?

- A standard mixture model assumes that all observed features contribute to the grouping of the samples.
- If not true, results may be strongly biased.
- Need to focus on the biological questions.

What's wrong?

- A standard mixture model assumes that all observed features contribute to the grouping of the samples.
- If not true, results may be strongly biased.
- Need to focus on the biological questions.
- How many subgroups of cancer samples do exist in my database, such that they are similar w.r.t. DNA alterations?

What's wrong?

- A standard mixture model assumes that all observed features contribute to the grouping of the samples.
- If not true, results may be strongly biased.
- Need to focus on the biological questions.
- How many subgroups of cancer samples do exist in my database, such that they are similar w.r.t. DNA alterations?
- Which DNA alterations are relevant for which subgroups?

What's wrong?

- A standard mixture model assumes that all observed features contribute to the grouping of the samples.
- If not true, results may be strongly biased.
- Need to focus on the biological questions.
- How many subgroups of cancer samples do exist in my database, such that they are similar w.r.t. DNA alterations?
- Which DNA alterations are relevant for which subgroups?
- Which genomic regions are simply uninformative (noise) for the clustering purposes?

I have an idea, let's build a Bayesian model that incorporates our biological knowledge!

1. Data from noise regions should follow roughly the same distribution for all samples.

I have an idea, let's build a Bayesian model that incorporates our biological knowledge!

1. Data from noise regions should follow roughly the same distribution for all samples.
2. The DNA alterations that characterize the biological behavior of a certain cancer subtype should be present with high probability if a sample belongs to this subtype.

I have an idea, let's build a Bayesian model that incorporates our biological knowledge!

1. Data from noise regions should follow roughly the same distribution for all samples.
2. The DNA alterations that characterize the biological behavior of a certain cancer subtype should be present with high probability if a sample belongs to this subtype.
3. Other, less characteristic DNA alterations may also be present among samples with varying probabilities.

How does such a model work?

- Define a partition (clustering) of the cancer samples as a model parameter.

How does such a model work?

- Define a partition (clustering) of the cancer samples as a model parameter.
- The number of groups and their contents are *a priori* unknown.

How does such a model work?

- Define a partition (clustering) of the cancer samples as a model parameter.
- The number of groups and their contents are *a priori* unknown.
- Define a division of the amplification variables into noise and informative ones.

How does such a model work?

- Define a partition (clustering) of the cancer samples as a model parameter.
- The number of groups and their contents are *a priori* unknown.
- Define a division of the amplification variables into noise and informative ones.
- Separate from the informative amplifications those believed to be group–specific.

How does such a model work?

- Define a partition (clustering) of the cancer samples as a model parameter.
- The number of groups and their contents are *a priori* unknown.
- Define a division of the amplification variables into noise and informative ones.
- Separate from the informative amplifications those believed to be group–specific.
- The status of each amplification variable is an unknown parameter.

How does such a model work?

- Define a partition (clustering) of the cancer samples as a model parameter.
- The number of groups and their contents are *a priori* unknown.
- Define a division of the amplification variables into noise and informative ones.
- Separate from the informative amplifications those believed to be group–specific.
- The status of each amplification variable is an unknown parameter.
- The frequencies of amplifications conditional on the status are unknown parameters.

How does such a model work?

- Define a partition (clustering) of the cancer samples as a model parameter.
- The number of groups and their contents are *a priori* unknown.
- Define a division of the amplification variables into noise and informative ones.
- Separate from the informative amplifications those believed to be group–specific.
- The status of each amplification variable is an unknown parameter.
- The frequencies of amplifications conditional on the status are unknown parameters.
- Specify prior distributions for all model parameters.

How to do learning with the model in practice?

- The model poses an enormous computational challenge due to the complex parameter space.

How to do learning with the model in practice?

- The model poses an enormous computational challenge due to the complex parameter space.
- Use non-reversible stochastic optimization to fit the model.

How to do learning with the model in practice?

- The model poses an enormous computational challenge due to the complex parameter space.
- Use non-reversible stochastic optimization to fit the model.
- The stochastic search algorithm alternates between grouping and samples and updating statuses of the amplification variables.

How to do learning with the model in practice?

- The model poses an enormous computational challenge due to the complex parameter space.
- Use non-reversible stochastic optimization to fit the model.
- The stochastic search algorithm alternates between grouping and samples and updating statuses of the amplification variables.
- The algorithm has been implemented in software BASTA, which is freely available from www.helsinki.fi/bsg

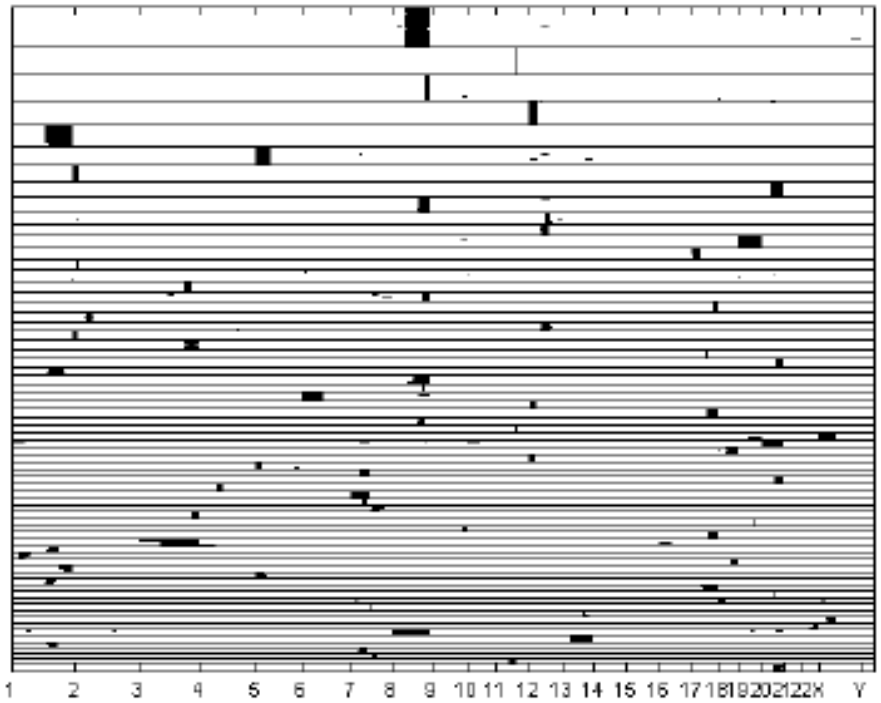
How to do learning with the model in practice?

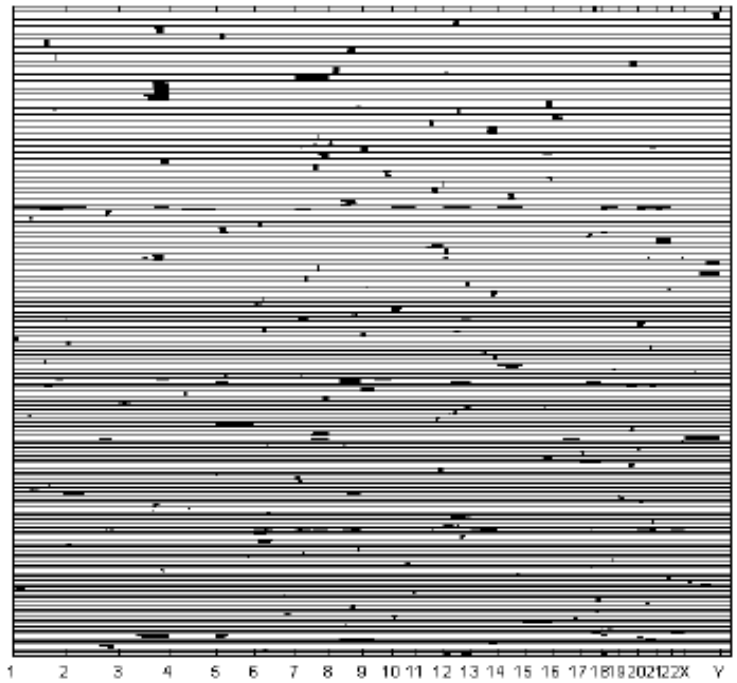
- The model poses an enormous computational challenge due to the complex parameter space.
- Use non-reversible stochastic optimization to fit the model.
- The stochastic search algorithm alternates between grouping and samples and updating statuses of the amplification variables.
- The algorithm has been implemented in software BASTA, which is freely available from www.helsinki.fi/bsg
- Details of the method are presented in Marttinen et al. (2009, BMC Bioinformatics).

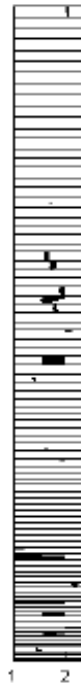
How to do learning with the model in practice?

- The model poses an enormous computational challenge due to the complex parameter space.
- Use non-reversible stochastic optimization to fit the model.
- The stochastic search algorithm alternates between grouping and samples and updating statuses of the amplification variables.
- The algorithm has been implemented in software BASTA, which is freely available from www.helsinki.fi/bsg
- Details of the method are presented in Marttinen et al. (2009, BMC Bioinformatics).
- Our experiments showed a very solid performance for the method.

BASTA results for the data (parts I-III).



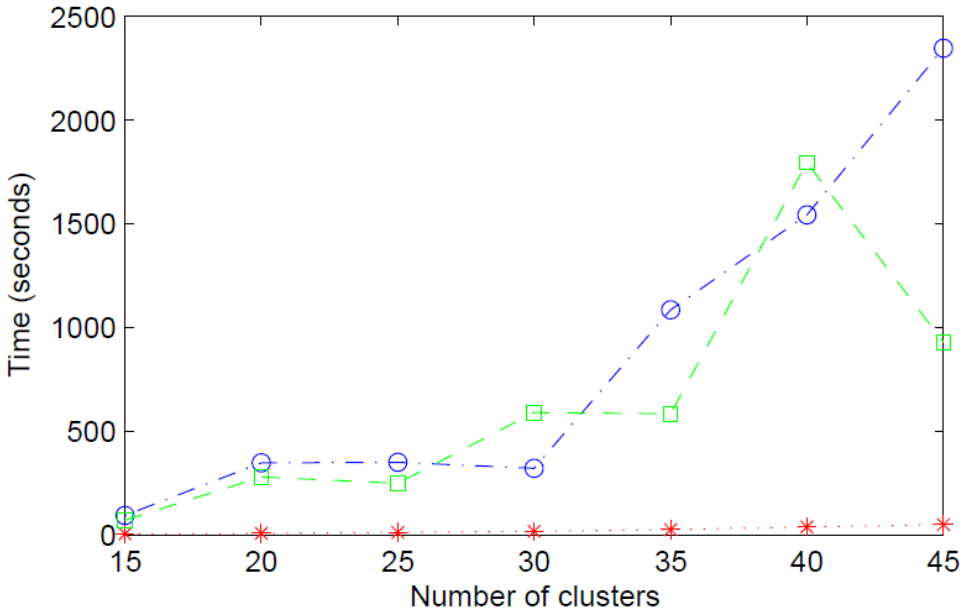


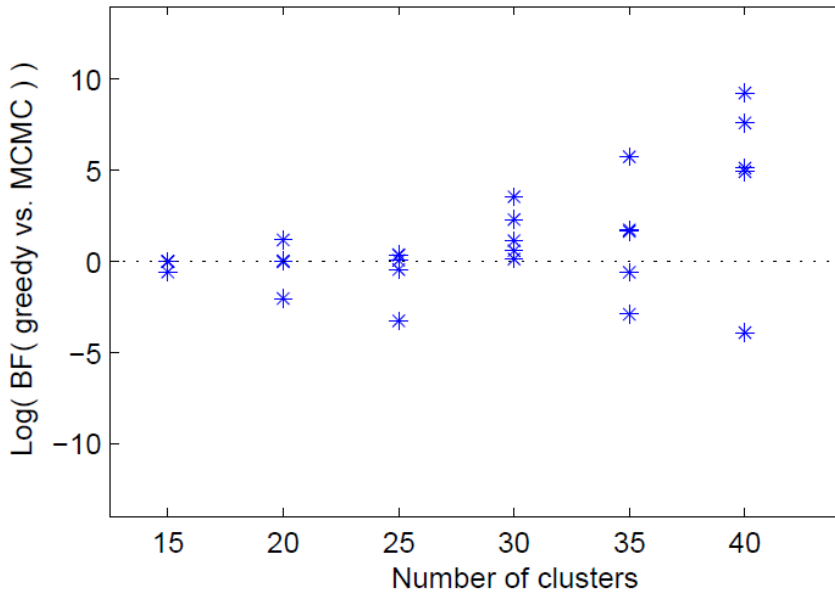


Moral of the story?

It may be quite dangerous to use data mining methods when not aware if the assumptions hidden in them meet the characteristics of the biological observations...

Comparison between a hill-climbing version of the nonreversible process and Gibbs sampler





Left panel: the time complexity of Gibbs sampler as a function of time required to identify as good solution as that found by a smart stochastic search. Right panel: Bayes factor values for the best clustering solutions found by the smart search against those found by Gibbs sampler. In both cases x-axis corresponds to the number of clusters in the generating model.

Extensions...

- We have developed a number of variants of the stochastic nonreversible optimization for different clustering, classification and semi-supervised classification models, papers and sw available from www.helsinki.fi/bsg.

Extensions...

- We have developed a number of variants of the stochastic nonreversible optimization for different clustering, classification and semi-supervised classification models, papers and sw available from www.helsinki.fi/bsg.
- Ongoing work to develop hybrid algorithms combining IS with nonreversibility.

Extensions...

- We have developed a number of variants of the stochastic nonreversible optimization for different clustering, classification and semi-supervised classification models, papers and sw available from www.helsinki.fi/bsg.
- Ongoing work to develop hybrid algorithms combining IS with nonreversibility.
- This story ends now, but the work still goes on...