Author(s): Helske, Jouni; Eerola, Mervi; Tabus, Ioan

Title: Minimum Description Length Based Hidden Markov Model Clustering for Life Sequence Analysis

Year: 2010

Version: Accepted version (Final draft)

Please cite the original version:

Helske, J., Eerola, M., & Tabus, I. (2010). Minimum Description Length Based Hidden Markov Model Clustering for Life Sequence Analysis. In Workshop on Information Theoretic Methods in Science and Engineering. http://sp.cs.tut.fi/WITMSE10/Proceedings/index.html

MINIMUM DESCRIPTION LENGTH BASED HIDDEN MARKOV MODEL CLUSTERING FOR LIFE SEQUENCE ANALYSIS

Jouni Helske[1]*, Mervi Eerola*[2]*, Ioan Tabus*[1]

[1] Department of Signal Processing, Tampere University of Technology,
P.O Box 553, FIN-33101 Tampere, FINLAND, Jouni.Helske@jyu.fi, Ioan.Tabus@tut.fi
[2]Methodology Centre for Human Sciences, University of Jyväskylä
P.O.Box 35, FIN-40014 University of Jyväskylä, FINLAND, Mervi.Eerola@jyu.fi

ABSTRACT

In this article, a model-based method for clustering life sequences is suggested. In the social sciences, model-free clustering methods are often used in order to find typical life sequences. The suggested method, which is based on hidden Markov models, provides principled probabilistic ranking of candidate clusterings for choosing the best solution. After presenting the principle of the method and algorithm, the method is tested with real life data, where it finds eight descriptive clusters with clear probabilistic structures.

1. INTRODUCTION

In social science applications the goal of sequence analysis is usually to find a typology of life sequences in the data. Model-free methods, such as optimal matching for pairwise alignment of sequences combined with clustering methods like Ward's agglomerative algorithm, are typically used [1]. The aim of this article was to develop a model-based method providing a simple probabilistic description of the sequence data and helping to find typical life paths. Hidden Markov models (HMMs) were chosen as a tool for these tasks. Hidden Markov models have been widely used earlier in biological sequence analysis [2] and speech recognition [3]. In both biological sequence analysis and speech recognition, there are usually large datasets available, whereas the datasets of life events are typically much smaller (both in number of sequences and length of single sequence).

Methods for clustering with hidden Markov models have been developed before [4] [5], but most of these methods implicitly assume very long observation sequences, since a distinct HMM for each sequence is used as a starting point of clustering. Other methods, such as Matryoshka algorithm [6], have been developed for the case of continuous valued observations.

In this article, a model-based method for clustering and analysing life sequence data is suggested.

2. METHODS

2.1. Hidden Markov Models

A Hidden Markov model consisting of m hidden states and l distinct observation symbols is described by the parameters $\boldsymbol{\lambda} = (\mathbf{A}, \mathbf{B}, \boldsymbol{\delta})$, where **A** is the state transition probability matrix, **B** is the observation symbol emission probability matrix and $\boldsymbol{\delta}$ is the initial distribution of states.

The element $a_{i,j}$ in matrix **A** gives the probability of transiting from state s_i to state s_j. The element $b_{i,j}$ in matrix **B** gives the probability of state s_i emitting observation symbol o_j. The element δ_i of the initial distribution $\boldsymbol{\delta}$ gives the probability of starting from state s_i.

Figure 1 shows the graphical representation of a hidden Markov model. The upper part of the figure shows the transition matrix **A** as a directed graph where the three states are the nodes, and non-zero transitions probabilities $a_{i,j}$ are shown as arcs between nodes. The probability of starting from state i is shown inside the node i as δ_i. The stacked bars below each state represent the symbol emission distribution at the state, where each nonzero emission probability $b_{i,j}$ is represented with a distinctly colored bar, having the height proportional to $b_{i,j}$. For example, the second state can only emit two different symbols, nps (with probability of ≈ 0.75) and nds (with probability of ≈ 0.25). Arcs with zero probability are not drawn.

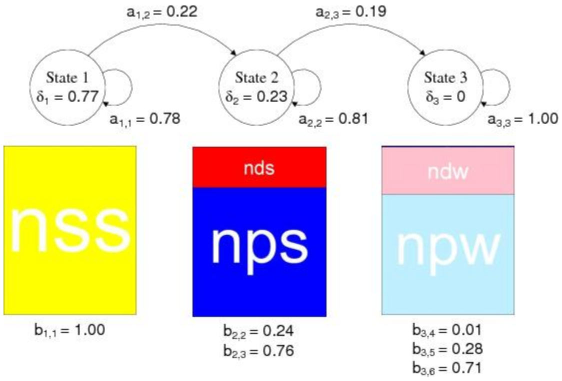


Figure 1. Example of HMM: $s_i \in \{1, 2, 3\}$, $o_i \in \{"nss", "nds", "nps", "npo", "ndw", "npw"\}$.

The parameters of the model $\boldsymbol{\lambda}$ are usually estimated by Baum-Welch algorithm minimizing the negative log-likelihood

$$L = -\log \prod_{i=1}^{N} P(\mathbf{o}^i|\boldsymbol{\lambda}), \qquad (1)$$

where $\mathbf{o}^i = (o_1^i, \ldots, o_n^i)$ is the ith observation sequence, and N is the total number of observation sequences. All the sequences have equal lengths n. The probability of the observation sequence given the model is

$$\begin{aligned} P(\mathbf{o}^i|\boldsymbol{\lambda}) &= \sum_{\mathbf{s}\in S^n} \Big[P(o_1^i|s_1)P(s_1) \\ &\quad \times \prod_{j=2}^{n} P(o_j^i|s_j)P(s_j|s_{j-1}) \Big] \\ &= \sum_{\mathbf{s}\in S^n} b_{s_1,o_1^i}\delta_{s_1} \prod_{j=2}^{n} b_{s_j,o_j^i} a_{s_{j-1},s_j}, \qquad (2) \end{aligned}$$

where the state sequences $\mathbf{s} = (s_1, \ldots, s_n)$ take all possible values in the state space $S^n = \{1, \ldots, m\}^n$. Despite the apparent complexity of (2), $P(\mathbf{o}^i|\boldsymbol{\lambda})$ can be efficiently calculated by an iterative algorithm [3].

The most probable path of hidden states given the observation sequence $\mathbf{o}^i$ can be efficiently calculated by the Viterbi algorithm and is denoted $\hat{\mathbf{s}}^i = (\hat{s}_1^i, \ldots, \hat{s}_n^i)$. For more detailed description of hidden Markov models and their properties, see for example [3].

2.2. Discussion on the interpretation of HMM for life sequences

One rationale behind using the HMM for life sequence analysis will be the attempt to identify similar sequences based on similar "hidden"or simplified state trajectories. The existence of similar hidden sequence of states can be attributed to both external factors, common to groups of populations, or to internal behavioral similarities for individuals with similar features. Finding hidden dynamics is thus important for analyzing and grouping the sequences and also for understanding the relationships between the factors that are measured. The significance of the hidden states in life sequences is dependent on the chosen structure of the hidden Markov models. The goals of our analysis are two-folded:

(G1) to group the similar sequences in a small number of clusters and

(G2) to group those symbols that act similarly within a cluster or under certain temporal contexts.

Our analysis is exploratory, we test a number of HMM structures and choose the optimal one(s) according to an information theoretic criterion. We then check the resulting optimal structure and find interpretations for the two types of groupings. We note that usual training of HMM acts along a full mixture model: the likelihoods are computed by summing over all possible sequences of hidden states. However, we are interested in explaining the data using clusters and therefore we defined the goal of our optimization problem by a different criterion, which makes use of the cluster structure. Differently than in simple clustering, where sequences are grouped based on a similarity measure applied to the sequence of observations, we now define the clusters in terms of the distinct parameters of the underlying HMMs.

2.3. Likelihood functions accounting for clustering with HMMs

Suppose that we constructed $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_K$, the K HMMs by which we want to define K clusters. We have a number of ways to decide for each sequence to which cluster it belongs. A hard decision will be based on the likelihood of the observations, conditional on the most likely state sequence. Let

$$\begin{aligned} \hat{P}(\mathbf{o}^i;\boldsymbol{\lambda}_k) &= \max_{\mathbf{s}\in S^n} P(\mathbf{o}^i|\mathbf{s};\boldsymbol{\lambda}_k)P(\mathbf{s}|\boldsymbol{\lambda}_k)P(\boldsymbol{\lambda}_k) \\ &\overset{def}{=} P(\mathbf{o}^i|\hat{\mathbf{s}}^i(\mathbf{o}^i);\boldsymbol{\lambda}_k) \\ &\quad \times P(\hat{\mathbf{s}}^i(\mathbf{o}^i)|\boldsymbol{\lambda}_k)P(\boldsymbol{\lambda}_k), \qquad (3) \end{aligned}$$

which is the maximum likelihood obtained for the maximizing state sequence $\hat{\mathbf{s}}^i(\mathbf{o}^i)$. The quantity $-\log \hat{P}(\mathbf{o}^i;\boldsymbol{\lambda}_k)$ is the codelength necessary to transmit the sequence $\mathbf{o}^i$ using the model $\boldsymbol{\lambda}_k$. A natural decision will be to use for the sequence $\mathbf{o}^i$ that model which requires the smallest codelength, so the clustering of the sequences is done according to

$$\hat{\boldsymbol{\lambda}}(\mathbf{o}^i) = \arg\max_{\boldsymbol{\lambda}_k} \hat{P}(\mathbf{o}^i;\boldsymbol{\lambda}_k). \qquad (4)$$

Under this decision strategy, the training of the K HMMs will have the goal of minimizing the overall codelength

$$L_1 = -\sum_{i=1}^{N} \log \hat{P}(\mathbf{o}^i;\hat{\boldsymbol{\lambda}}(\mathbf{o}^i)). \qquad (5)$$

A second clustering strategy is still based on the codelength for encoding the sequence $\mathbf{o}^i$ based on the best cluster, but now the coding strategy is defined by a mixture of all the hidden states in a given cluster. Given a cluster $\boldsymbol{\lambda}_k$, we construct the distribution

$$\overline{P}(\mathbf{o}^i;\boldsymbol{\lambda}_k) = \sum_{\mathbf{s}\in S^n} P(\mathbf{o}^i|\mathbf{s};\lambda_k)P(\mathbf{s}|\boldsymbol{\lambda}_k), \qquad (6)$$

which being a distribution can be used to encode the sequence $\mathbf{o}^i$ in $-\log_2 \overline{P}(\mathbf{o}^i;\boldsymbol{\lambda}_k)$ bits. Thus a natural strategy of choosing the best cluster for the sequence $\mathbf{o}^i$ is

$$\overline{\boldsymbol{\lambda}}(\mathbf{o}^i) = \arg\max_{\boldsymbol{\lambda}_k} \overline{P}(\mathbf{o}^i;\boldsymbol{\lambda}_k). \qquad (7)$$

Under this decision strategy, the training of the K HMM's will have the goal of minimizing the overall codelength

$$L_2 = -\sum_{i=1}^{N} \log \overline{P}(\mathbf{o}^i;\overline{\boldsymbol{\lambda}}(\mathbf{o}^i)). \qquad (8)$$

2.4. Algorithm for clustering with HMMs based on MDL

The algorithm is presented first for a given number of clusters K and number of states m in each cluster. In order to find an initial clustering, one HMM $\boldsymbol{\lambda}$ is fitted to the whole data, with the transition probability matrix $\mathbf{A}$ constrained such that it contains K block diagonal matrices. This produces a single HMM with K "submodels". The estimation of the parameters of this block diagonal HMM is done with the usual Baum-Welch algorithm by minimizing (1). In order to reduce the risk of being trapped in a poor local minima, a large number of initial values for the model parameters are used.

The block diagonal model $\boldsymbol{\lambda}$ obtained in the initial stage is split into K separate HMMs, $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_K$, using the parameters corresponding the kth block of the block diagonal model as the parameters of the kth HMM, $\boldsymbol{\lambda}_k, k = 1, \ldots, K$. The most probable cluster for each observation sequence $\mathbf{o}^i$ is the $\boldsymbol{\lambda}_k$ that maximizes the $P(\mathbf{o}^i|\boldsymbol{\lambda}_k)$ in (4) or (7). After finding the most probable cluster for each sequence, we get the empirical probabilities $P(\boldsymbol{\lambda}_k)$. Now instead of minimizing the negative log-likelihood of equation (1) by Baum-Welch, we minimize the negative log-likelihood (5) or (8). The expressions (5) or (8) are minimized numerically, for example using the nlm routine in the R environment [7].

Finally a modified minimum description length criterion is used to evaluate the goodness of fit:

$$MDL_c = 2L_c + p\log(Nn), \qquad (9)$$

where L_c is calculated by (5) or (8), p is the number of model parameters with non-zero values and Nn is the total number of data points (number of sequences times the length of sequence). The MDL_c accounts for encoding the parameters of the models and thus penalizes for too complex models and helps avoiding overfitting. After finding the minima of L_c, the model parameters can be tuned by setting some small probabilities in $\mathbf{A}$ and $\mathbf{B}$ to zero (and scaling the corresponding row probabilities so they sum to unity), which can provide smaller MDL_c than the parameters that minimize L_c.

Models with number of clusters K and states m in each cluster varying in sensible region are then fitted to the data, and the model with the lowest MDL_c is chosen as the best model. As allowing different number of hidden states in each cluster would make the search space explode, an assumption that there are equal number of hidden states in each cluster makes the search of the optimal model much simpler. In future work, ways to relax this assumption are possibly studied.

3. APPLICATION

The example data (the HELS study, Salmela-Aro & Nurmi) consist of 207 first year students in 1991 at the University of Helsinki. Their life events from three life domains were recorded retrospectively in 2008.

1. Parenthood: having/not having children (c/n).
2. Partnership: living single (s), living in a partnership (p), living separated/divorced/widowed (d).
3. Study and career: studying full-time (s), working (w), or doing something else (being in army, maternal leave, unemployed etc) (o).

From the 18-year follow-up, life sequences were constructed as character strings (such as nsw) resulting in 18 possible extended symbol combinations for each year, see the table on top of Figure 2.

In order to find the best model for clustering and describing the data, models with K and m ranging from 2 to 10 were fitted to the data. In this application, we chose to minimize (5).

The parameter values that minimized (1), the log-likelihood of the initial block-diagonal model, did not change significantly when minimizing (5), and thus the sequence assignments were the same. However, this may not hold in all cases and needs to be investigated.

Table 1 shows the number of clusters and number of hidden states in each cluster for the ten models with the smallest MDL_c value. In all, MDL_c favored models with large number of clusters and small number of hidden states in a cluster. The model with eight clusters and three hidden states in each cluster had the lowest MDL_c value. In comparison, if the minima of the (1) would have been used in calculation of MDL_c, the best model would have been the model with only two clusters and nine states in both clusters. The MDL_c value of that model was 12530.0 when using (5) instead of (1). The hidden states of the best model were interpreted in terms of the most probable symbols emitted, resulting in 11 distinct hidden states composed by the original 18 observation symbols. Their coding is given at the top of Figure 3 (on the right). Based on the hidden states, the model parameters and the hidden structure of the model (Figure 2), the clusters were given the following descriptions. They represent the typical path of a member of the corresponding cluster.

Table 1. Ten fitted models with lowest MDL_c value.

K	m	MDL_c
8	3	10867.7
9	3	10937.2
7	3	11002.9
8	4	11082.9
10	3	11099.5
4	3	11120.7
6	3	11155.5
5	3	11178.7
5	4	11226.4
7	4	11254.8

1. Graduation-anchored life paths.
2. Unstable partnerships, early enter to working life.
3. Fast starters, early and stable partnership with children when studying.

4. Mainly singles with traditional working history.
5. Work-oriented unstable partnerships without children.
6. Slow starters, late and stable partnership with children.
7. Family with fragmented working history.
8. Untypical life paths, singles having children, single-supporters.

4. CONCLUSION

The suggested method was able to find clusters of life paths in the sequence data. Compared to the usual model-free alignment methods, which are based on pairwise distances of the sequences and their clustering, our method has a clear probabilistic model structure. It was assumed that the sequence data is composed by a mixture of life pathways. This mixture of pathways was modelled as a hidden layer which can be interpreted as an underlying developmental structure from which the observed sequences are noisy realizations. Unlike biological sequences, life sequences tend to be much shorter and less variable, but usually some external information and theoretical knowledge about the developmental processes exist. The probabilistic structure of the method allows to incorporate that information by modelling the transition probabilities. The efficient estimation of such models, as well as finding ways to measure the interaction of multiple life domains in time is left for further work.

5. ACKNOWLEDGMENTS

We thank the Helsinki Longitudinal Student Study (HELS) by K. Salmela-Aro & J-E Nurmi for letting us use the data for our purposes.

6. REFERENCES

[1] A. Abbott and A. Tsay, “Sequence analysis and optimal matching methods in sociology: Review and prospect,” *Sociological Methods & Research*, vol. 29, pp. 3–33, Aug. 2000.

[2] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*, Cambridge University Press, Cambridge, 1998.

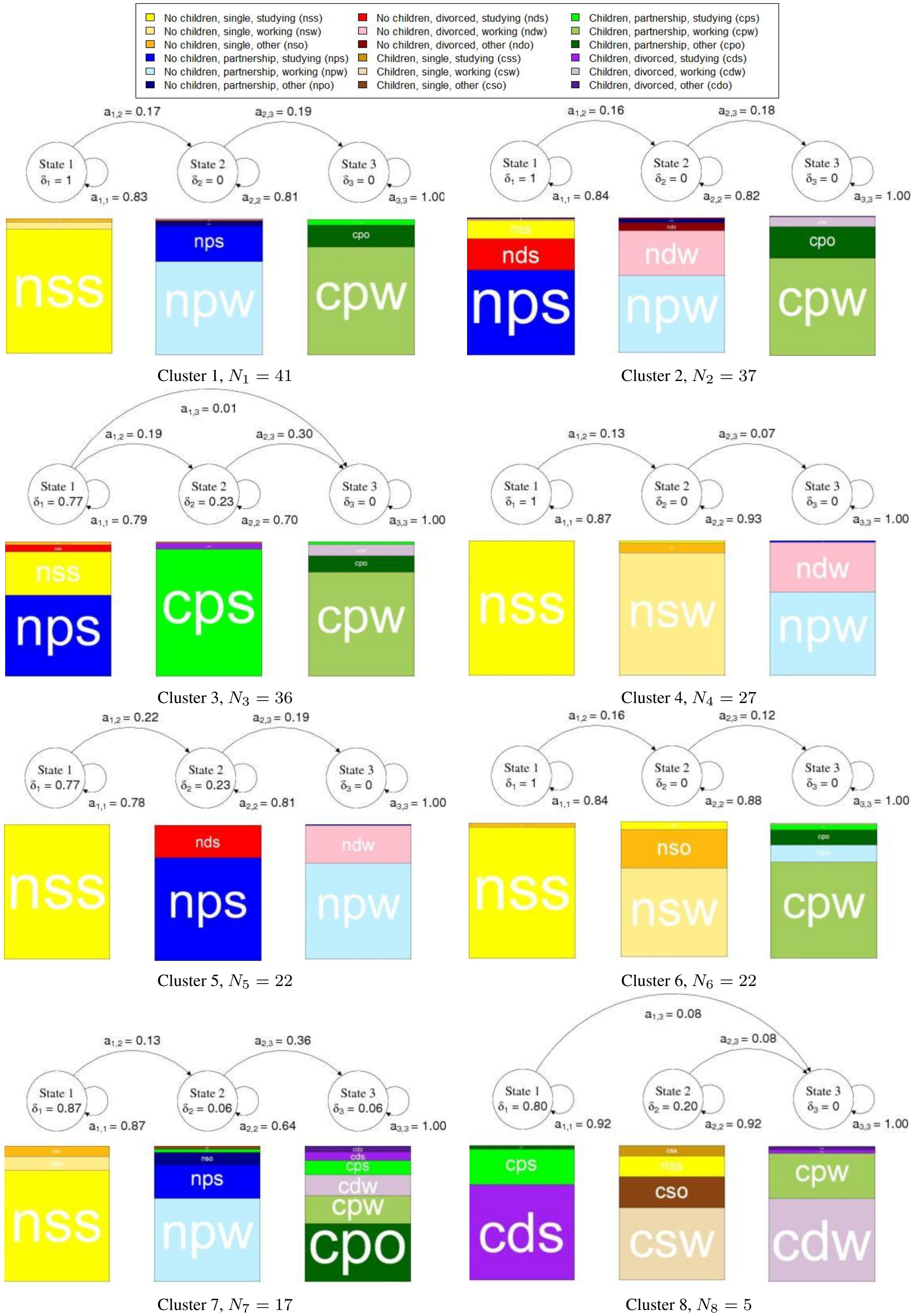
[3] L. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” in *Proceedings of the IEEE*, Feb. 1989, pp. 257–286.

[4] P. Smyth, “Clustering sequences with hidden Markov models,” in *Advances in Neural Information Processing Systems*. 1997, pp. 648–654, MIT Press.

[5] M. Bicego, V. Murino, and M.A.T. Figueiredo, “Similarity-based clustering of sequences using hidden Markov models,” in *Machine Learning and Data Mining in Pattern Recognition*. 2003, pp. 86–95, Springer.

[6] C. Li and G. Biswas, “Applying the hidden Markov methodology for unsupervised learning of temporal data,” *International Journal of Knowledge Based Intelligent Engineering Systems*, vol. 6, pp. 152–160, Jul. 2002.

[7] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2009, ISBN 3-900051-07-0.



Cluster 1, $N_1 = 41$

Cluster 2, $N_2 = 37$

Cluster 3, $N_3 = 36$

Cluster 4, $N_4 = 27$

Cluster 5, $N_5 = 22$

Cluster 6, $N_6 = 22$

Cluster 7, $N_7 = 17$

Cluster 8, $N_8 = 5$

Figure 2. The 8 clusters found by our algorithm. Shown in each subfigure is the structure of the HMM for cluster k, with the hidden state transitions A_k and the symbol emission distribution B_k. N_k is the number of sequences assigned to cluster k.

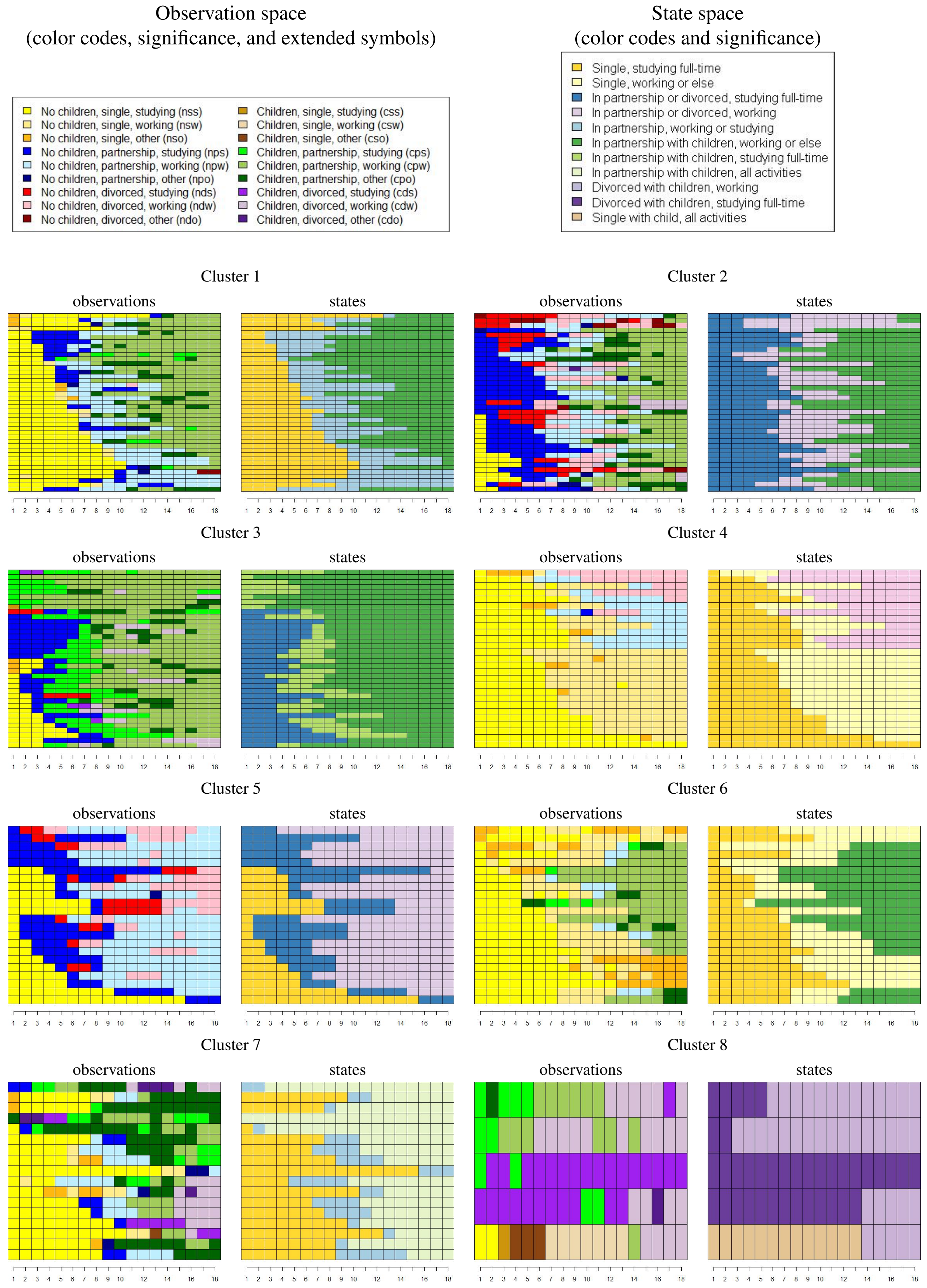


Figure 3. Observation sequences and their color coding (left) and the most probable paths of hidden states and their color coding (right) for each cluster. The most probable paths are sequences of 11 hidden states based on a similar composition of observed states in Fig 2. Note that the colors on the right and left figures may not in all cases have the same interpretation.