Mixture Models from Multiresolution 0-1 Data

Prem Raj Adhikari and Jaakko Hollmén

Helsinki Institute for Information Technology (HIIT), and
Department of Information and Computer Science (ICS)
Aalto University School of Science,
PO Box 15400, FI-00076 Aalto, Espoo, Finland
{prem.adhikari,jaakko.hollmen}@aalto.fi

Abstract. Multiresolution data has received considerable research interest due to the practical usefulness in combining datasets in different resolutions into a single analysis. Most models and methods can only model a single data resolution, that is, vectors of the same dimensionality, at a time. This is also true for mixture models, the model of interest. In this paper, we propose a multiresolution mixture model capable of modeling data in multiple resolutions. Firstly, we define the multiresolution component distributions of mixture models from the domain ontology. We then learn the parameters of the component distributions in the Bayesian network framework. Secondly, we map the multiresolution data in a Bayesian network setting to a vector representation to learn the mixture coefficients and the parameters of the component distributions. We investigate our proposed algorithms on two data sets. A simulated data allows us to have full data observations in all resolutions. However, this is unrealistic in all practical applications. The second data consists of DNA aberrations data in two resolutions. The results with multiresolution models show improvement in modeling performance with regards to the likelihood over single resolution mixture models.

Keywords: Multiresolution data, Mixture Models, Bayesian Networks.

1 Introduction

A phenomenon or a data generating process measured in different levels of accuracy results in multiresolution data. This difference in accuracy arises because of improvement in measurement technology [1]. Newer generation technology measures finer units of data producing data in fine resolution. In contrast, older generation technology measures only the coarse units of data producing data in coarse resolution. Thus, accumulation of data over long duration of time results in multiresolution data. The availability of multiresolution data ranges across diverse application domains such as computer vision, signal processing, telecommunications, and biology [2].

The domain of scale-space theory [4], and wavelets [5] have close affinity with the domain of multiresolution modeling thus widening the scope of multiresolution modeling research. Furthermore, multiresolution data falls under one of the

J. Fürnkranz, E. Hüllermeier, and T. Higuchi (Eds.): DS 2013, LNAI 8140, pp. 1–16, 2013.

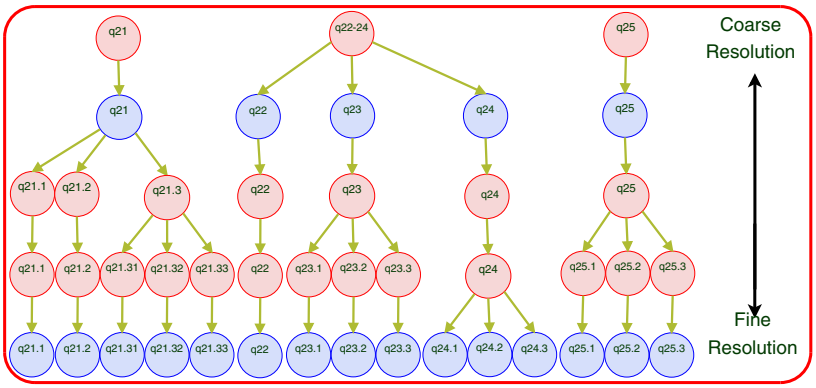

Fig. 1. A typical dichotomy of universals and particulars in giving rise to multiresolution data. Figure shows a part of chromosome–17 in five different resolutions of nomenclature as defined by ISCN [3].

essential ontological dichotomies of universals and particulars [6]. For example in cytogenetics, an application area of interest, International System for Human Cytogenetic Nomenclature (ISCN) has a standardized nomenclature for the parts of the genome. It has defined five different resolutions of the chromosome band: 300, 400, 550, 700, and 850 [3]. In other words, there are 862, and 311 regions in a genome in resolution 850 (fine resolution), and 300 (coarse resolution), respectively. Figure 1 shows an example of multiresolution data resulting from the ISCN nomenclature which is also our application area of interest. Figure 1 shows a part of chromosome–17 in five different resolutions forming a tree structure among different chromosome bands. Here, the same part of genome measured in different levels of detail generating multiresolution data.

Finite Mixture Models are semi-parametric probability density functions represented as the weighted sum of component densities of chosen probability distributions such as Gaussian, Bernoulli, or Poisson [7,8]. Mixture models have found wide spectrum of uses such as clustering [9], density estimation [10], modeling heterogeneity [11], handling missing data [12], and model averaging. They are versatile because of their suitability for any choice of data distribution, either discrete or continuous, and flexibility in the choice of component distributions [8]. However, mixture models in their basic form only operate on single data resolution, and is unable to model multiresolution data. The only mixture modeling solution to multiresolution data are to model the different resolutions separately and at best compare the findings. Cancer is not a single disease but a heterogeneous collection of several diseases [13]. Therefore, we use mixture models to model cancer patients discussed in Section 4.2 because mixture models are well known for their ability to model heterogeneity.

In our previous work, we transform the multiresolution data to a single resolution using different deterministic transformation methods, and model the resulting single resolution data [14]. Results in [14] shows improvement in the performance of mixture models through multiresolution analysis compared to

single resolution analysis. We also proposed a multiresolution mixture model based on merging of mixture components across different resolutions in [15]. The improvement in [15] is that the models assimilate the information contained in other data resolutions. Furthermore, transformations here are in the model domain unlike in the data domain as in [14]. In all scenarios above, the mixture models are generated in a single resolution and directly unusable in multiresolution scenarios without modification.

In the past, research has considered formulating the multiresolution mixture model in different application areas. For instance, the multiresolution Gaussian mixture model in [16] approximates the probability density, and adapts to smooth motions. The design of the model is for a specific choice of data distribution, and generates trees of decreasing variance, and consequently a tree of Gaussians. Unlike an actual multiresolution model, this essentially models single resolution data using a tree from the same data for multiple Gaussians on different scales with different variance. Furthermore, difference in the pyramid structure present in other domains limit its general applicability.

Authors have increased the efficiency and robustness of learning mixture models using multiresolution kd-trees [9,17]. Furthermore, authors in [10] have proposed the a mixture of tree distributions in a maximum likelihood framework. Similarly, authors in [18,19] use multiresolution binary trees to learn discrete probability distribution. However, it is impossible to represent all multiresolution data as kd-trees which are binary in nature. In addition, the focus in [18,19] is in modeling single resolution data. Additionally, multiresolution trees are also used in object recognition [20], and as binary space portioning trees [21]. However, the authors in [20,21] use them in the context of recursive neural networks, and geometric representations for information visualization, respectively.

In this paper, we propose a multiresolution mixture model whose components are Bayesian networks denoting the hierarchical structure present in multiresolution data. We learn the parameters of each component distribution in a Bayesian network framework. Component distributions in the form of Bayesian network is useful also to impute the missing data resolutions considering them as missing values. Finally, we transform the multiresolution data in the Bayesian network representation to a single data in vector form to learn the mixing coefficients and the parameters of the mixture model in a maximum likelihood framework using the EM algorithm.

2 Bayesian Networks of Multiresolution Data

Bayesian networks bring the disciplines of graph theory and probability together to elegantly represent complex real-world phenomena dealing with uncertainty [22,23]. A Bayesian network consists of nodes or vertices that encode information about the system in the form of probability distributions, and links or arcs or edges that denote the interconnections or interactions between nodes in the form of conditional independence [22]. It analytically represents a joint distribution over a large number of variables. Furthermore, it treats learning

and inference simultaneously, seamlessly merges unsupervised and supervised learning, and also provides efficient methods for handling missing data [23].

2.1 Component Distributions of Multiresolution Hierarchy as Bayesian Networks

A Bayesian network is a directed acyclic graph (DAG) that describes a joint distribution over the set of random variables $X_1, \ldots, X_d$ such that

$$P(X_1, \ldots, X_d) = \prod_{i=1}^{\mathrm{d}} P(X_i \mid \text{parents}(X_i)), \tag{1}$$

where parents(X_i) are the set of vertices from which there is an edge to X_i. Figure 2 shows an example of a Bayesian network of six random variables A, B, C, D, E, and F. The six vertices represent the six random variables, and the five directed edges represent the conditional dependencies (independence). We can define conditional independence in a Bayesian network as: A variable is conditionally independent of all the variables in the network given its Markov blanket. The Markov blanket of a variable is the set of its parents, its children, and the other parents of its children.

Data in multiple resolutions share a commonality because they measure the same phenomenon. A single feature in the coarse resolution corresponds to one or more features in the fine resolution. We can exploit this information from the application area to determine the relationships between data resolutions

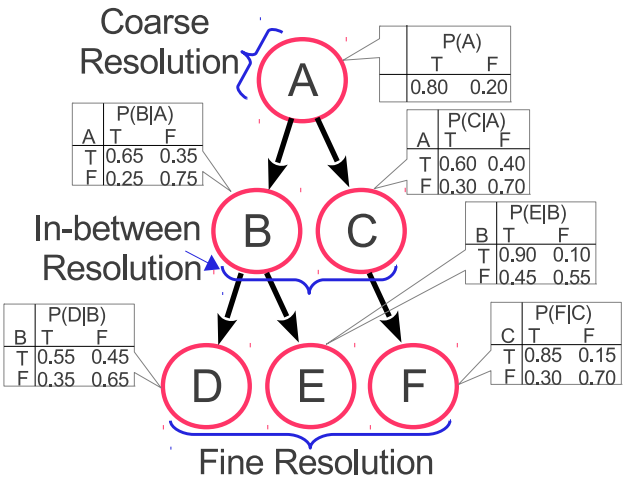

Fig. 2. Network representation of the multiresolution data where ancestors denote data in coarse resolution and leaves denote data in fine resolution. The network a simple Bayesian network of six random variables with five edges along with the associated probability tables.

and consequently, the structure of the Bayesian network. The data features in the coarse resolution form the root and branches near the root of the network. Similarly, the data features in the fine resolution form the branches towards the leaves and the leaves of the tree. Additionally, we can assume that the directed arrows originate from the features in the coarse resolution for computational efficiency.

Each vertex of a Bayesian network bears a corresponding conditional probability distribution (CPD). The CPD specifies that a child takes a certain value with a probability depending the value of its parents [22]. In the figure, for example the variables D, and E are conditionally independent given B. We can simplify the joint probability distribution of A, B, C, D, E, and F using the conditional independences in Figure 2 as:

$$\begin{aligned} P(A,B,C,D,E,F) &= P(A|B,C,D,E,F)P(B|A,C,D,E,F)\ldots \\ &\quad P(C|A,B,D,E,F)P(D|A,B,C,E,F)\ldots \\ &\quad P(E|A,B,C,D,F)P(F|A,B,C,D,E) \\ &= P(A)P(B|A)P(C|A)P(D|B)P(E|B)P(F|C) \end{aligned} \tag{2}$$

The CPD of a discrete variable is represented in a table as shown in Figure 2. It enumerates each possible set of values for the variable and its parents. Algorithms based on maximum likelihood (MLE) and maximum a posteriori estimates (MAP) can learn the parameters of the Bayesian network with a known structure [24]. In our application, the structure of Bayesian networks comes from the domain knowledge. The depth of the Bayesian network depends on the number of resolutions in multiresolution data. Learning a Bayesian network of known structure involves determining the CPD of the variables in the network. We learn the CPD of the variables using the Maximum Likelihood Estimate (MLE) [22].

2.2 Missing Resolutions in Multiresolution Data

Missing data has received considerable research interest because of their abundant occurrence in many domains [12,25]. The problem of missing data escalates when some resolutions (entire data) in a multiresolution setting are missing. Therefore, when the values are missing in multiresolution analysis, one or more resolutions (entire data) will be missing. This is unlike the typical missing data problems where small number of variables in some samples will be missing. For example, data in a coarse resolution can be missing while data in other resolutions are available. Bayesian networks have a seamless ability to handle missing data [12]. Therefore, learning the Bayesian networks also helps to generate the data in missing resolutions because Bayesian networks are generative models.

We can impute the missing values using marginal inference in Bayesian Networks. Marginal inference is the process of computing the distribution of a subset of variables conditioned on another subset [22]. We can calculate the marginal inference for a joint distribution P(A,B,C) given the evidence B=true as:

$P(A \mid B = true) \propto \sum_{C} P(A, B = true, C).$

Authors have proposed algorithms such as variable elimination, and sum product algorithm to compute marginal inference [22]. We draw samples under the given evidence from consistent junction trees using the BRMLToolbox [22].

3 Multiresolution Mixture Model of Multivariate Bernoulli Distributions

Mixture Models are semi–parametric latent variable models that models a statistical distribution by a weighted sum of parametric distributions [7,8,22]. They are flexible for accurately fitting any complex data distribution for a suitable

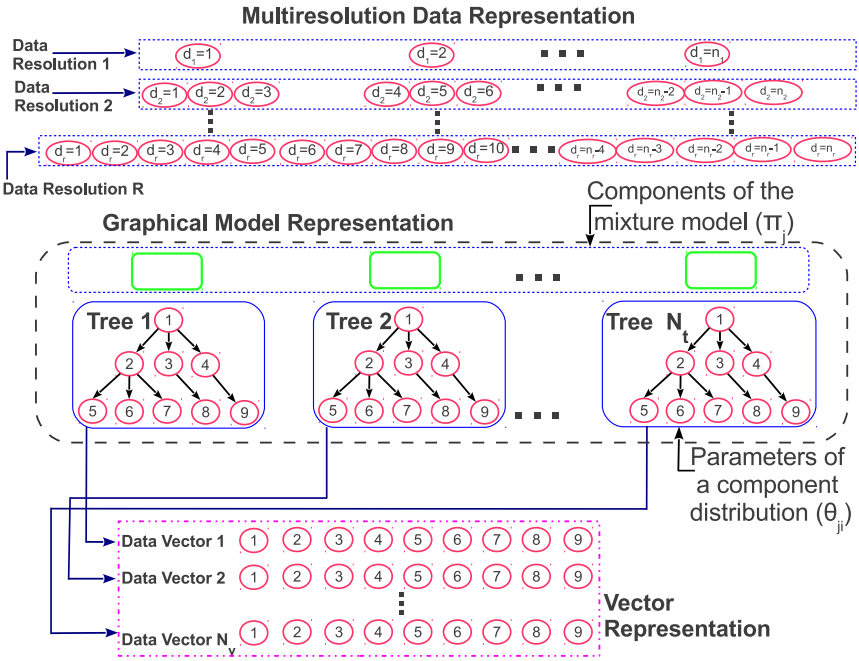

Fig. 3. Top panel shows the general representation of multiresolution data. There are 'r' data resolutions of different dimensionality. The multiresolution data representation is then transformed to a Bayesian network representation which in turn is mapped to a vector of single resolution data. The Bayesian network representation in the middle panel (within the dashed rectangle) also depicts a mixture model having the Bayesian networks as the components for data in multiple resolutions. The three solid rectangles on the top represent different mixture coefficients. Similarly, the three network of nodes denote the three component distributions where each vertex defines a parameter of the component distribution. The numbers inside the nodes denote the position of the variable in the vector representation with regards to dimensionality. The dash–dotted rectangle in the bottom of the figure shows the vector representation of the data derived from the Bayesian network representation. In the figure, N_t, and N_v denote the number of networks abreast adjacently in the multiresolution data, and the number of samples in multiresolution data mapped to a vector representation, respectively.

choice of data distribution, and a good enough number of mixture components. Expectation Maximization (EM) algorithm helps to learn the maximum likelihood parameters of the mixture model [26]. The EM algorithm requires prior knowledge of the number of components in the mixture model. Model selection is the process of determining the number of components in the mixture model [8]. In our previous work, we have tried to solve the problem of model selection in mixture models [11,27,28]. We learn mixture model of different complexities in a cross-validation setting and select a model with the number of components that gives the best generalization performance.

A multiresolution data is a collection of different component distributions as shown in the middle panel (within the dashed rectangle) of Figure 3. The multiresolution components in the middle panel (within the dashed rectangle) of Figure 3 encode the relationships between different resolutions of multiresolution data. The structure of the component distribution comes from the domain knowledge. Thus, the problem with regard to model selection in multiresolution data culminates to determining the optimal number of such component distributions present in the data. Similarly, learning the parameters of the component distributions involves learning the parameters of those networks.

In the general framework for the EM algorithm, we can assign only a single probability value to a node in the mixture model [26]. However, each variable in Bayesian network consists of minimum of two probability values denoting the CPD of the nodes. Therefore, in this contribution, we map the Bayesian network to vector representation to learn a multiresolution mixture model of Bayesian networks. This simple and intuitive solution proposed in this contribution transforms Bayesian networks to vectors with increasing dimensionality representing increasing depth of the Bayesian network. The first element of the vector will be the root node in the first generation i.e. coarsest resolution. Similarly, the last element of the vector will be leaf node of the last generation i.e. finest resolution arranged from left to right as shown in the bottom panel (within the dashed dotted rectangle) of Figure 3. In the middle and the bottom panel of Figure 3, the number inside the vertices denote the relative position of the variable in the vector representation with regards to dimensionality. Multiresolution mixture components transformed to a vector representation will have the same dimensionality because the structure of component distributions are identical with one another. However, component distributions have different parameters.

Vector representation of Bayesian networks eases modeling multiresolution data in one resolution. Furthermore, it increases the number of data samples. The number of samples in the vector form, N_v will be $N_v = N \times N_t$. Here, N is the number of samples of data in each resolution. Similarly, N_t is the number of Bayesian networks present in the data along the dimension corresponding to one sample. Furthermore, the data dimensionality will be considerably reduced as the depth of the Bayesian network is generally small. Additionally, Bayesian networks provide the sparsity [24], making data dimensionality in the vector representation d_v is smaller than that of the finest resolution of the multiresolution data. Furthermore, the vector representation has larger number of data samples

than the original Bayesian networks representation, $d_v < \max(d_r) \ll N_t \ll N_v$. Here, d_r is the dimensionality of data in different resolutions. An increase in the number of data samples and a reduction in dimensionality facilitates the learning of mixture models because they require a large number of samples to accommodate the increasing dimensionality of the data.

We can describe the mixture model of multivariate Bernoulli distributions for a 0-1 data [7] as:

$$p(\mathbf{x} \mid \Theta) = \sum_{j=1}^{\mathrm{J}} \pi_j \prod_{i=1}^{\mathrm{d}} \theta_{ji}^{x_i} (1 - \theta_{ji})^{1-x_i}. \tag{3}$$

Here, $\boldsymbol{\Theta} = \{J,\ \{\pi_j, \theta_j\}_{j=1}^{J}\}$ denotes the parameters of the mixture model of multivariate Bernoulli distributions. Here, π_j denotes the mixing proportions which sum to 1. Similarly, θ_{ji} is the probability that a random variable of the j^{th} component in the i^{th} dimension will take the value of 1. In multiresolution scenario, i differs for each resolution. Therefore, we have to model different resolutions with different models. We can formulate Equation 3 with respect to log likelihood to learn the mixture model using the EM algorithm in a maximum likelihood framework [8] as:

$$\mathcal{L}(\Theta) = \sum_{n=1}^{\mathrm{N}} \log\ P(x_n \mid \Theta) = \sum_{n=1}^{\mathrm{N}} \log \left[\sum_{j=1}^{\mathrm{J}} \pi_j \prod_{i=1}^{\mathrm{d}} \theta_{ji}^{x_{ni}} (1 - \theta_{ji})^{1-x_{ni}} \right]. \tag{4}$$

Given the number of mixture components, J, parameterized by $\Theta = \{\pi_j, \theta_j\}$, the EM algorithm can learn the mixture model that maximizes the likelihood in Equation 4. Model selection using cross-validated log likelihood can determine the number of mixture components, J [11,27,28].

4 Experimental Data

We experiment with the proposed methodology on two multiresolution data: an artificial data, and a chromosomal aberrations data, both in three resolutions.

4.1 Artificial Multiresolution 0-1 Dataset

In some application areas the relationships between different resolutions in a multiresolution setting are well known [6]. We can exploit such knowledge to artificially generate realistic multiresolution data. We initially fixed the structure of a Bayesian network to the the components of the mixture model shown in the middle panel (within the dashed rectangle) of Figure 3. Five such Bayesian networks were abreast along the dimensionality of the data. The dimensionality of data in three different resolutions are 5, 15, and 25 respectively. Firstly, we generate the data in the finest resolution i.e. having dimensionality of 25.

We fix two parameters to sample the data of given dimensionality: X is uniformly distributed in the range $[0, 1]$, and l is normally distributed with mean 0.1 and standard deviation 0.04.

$$X \sim U\,[0,1] \times d \text{ and } l \sim N\left(\mu = 0.1, \sigma^2 = 0.04\right) \times d$$

where d denotes the data dimension.

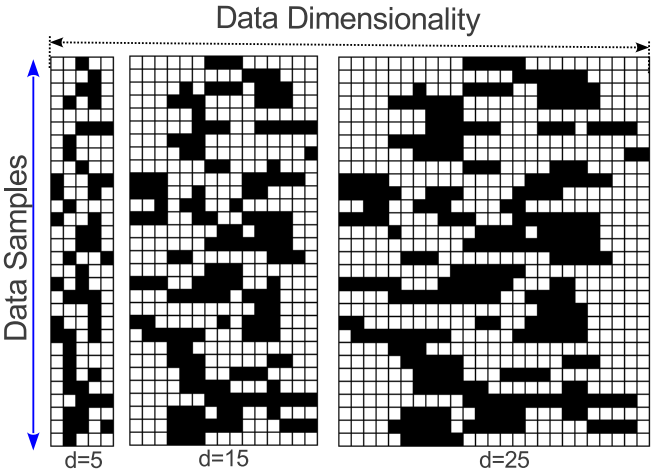

Fig. 4. First thirty samples of three different resolutions of the artificial 0-1 data. Black denotes 1s and white denotes 0s.

First, we create a matrix of the required size with all zeros. We divide the unit interval in 25 equal parts to generate 25–dimensional data. The parameter, X, defines the beginning of an aberration, and l defines the length of the aberration. We randomly choose a data sample for aberration and flip zeros to ones from dimensionality X of length l i.e. from dimension X to dimension $X+l$. We ignore the the lengths that are greater than the dimension $25 < (X+l)$ to maintain the data dimensionality. We continue this process iteratively until the number of 1s is approximately 55-60% of the data to mimic chromosomal aberration datasets.

Domain knowledge of chromosomal aberrations informs us about the typical length of aberrations. Furthermore, aberrations never span across the centromere. Therefore, we break an aberration that is longer than a predefined length, 15 in the experiments, randomly either on the left or the right of the centromere. We fix the centromere after the 10^{th} dimension, i.e. variable on the 10^{th} dimension is on the left side of the centromere, and the variable on the 11^{th} dimension is on the right side of the centromere.

Figure 4 shows artificial data in three different resolutions with dimensionalities 5, 15, and 25, respectively. Figure 4 also shows that similar to chromosomal aberration data, the artificial data are sparse, and spatially dependent. We gain the knowledge of relationships between data in different resolutions from the

Bayesian network as shown in the middle panel (within the dotted rectangle) of Figure 3. We apply that knowledge to downsample the data in dimensionality 25 to a dimensionality of 15, and then 5 using the majority voting downsampling method proposed in [14]. In experiments we fix the number of samples of the dataset to 1000 which is similar to the chromosomal aberration dataset.

4.2 Multiresolution Chromosomal Aberration Dataset

The causes and consequences of chromosomal aberration such as amplification, deletion, and duplication have significant roles in cancer research [13]. DNA copy number amplifications have been defined as the hallmarks of cancer [11,28]. Chromosomal aberrations are early markers of cancer risk and their detection and analysis has considerable clinical merits [11,29]. The two chromosomal aberration datasets in two resolutions have a dimensionality of 393, and 862 in coarse, and fine resolution, respectively. Both datasets are used in [11,27,28]. Datasets are available from the authors on request. The sources of the two datasets were different and correspond to different cancer patients in the two different resolutions.

We experiment chromosome–wise to constrain the complexity of learning the mixture model because the data are high dimensional and samples are small. Complexity of mixture models increases quadratically with the dimensionality. For example, the number of samples in chromosome 17 is 342 and 2716 in coarse and fine resolution, respectively. Therefore, we correspond the first 342 samples in fine resolution to the samples in coarse resolution. We then downsample the next 342 samples (samples 343 to 684) to a resolution between coarse and fine resolution such that the depth and structure of the resulting Bayesian network are similar to those of the networks used for artificial dataset. We ignore the remaining 2032 samples in the fine resolution.Similarly, the network present in the real world dataset differ from the artificial dataset. We select the most representative network covering more than 50% of the data and ignore the other networks. The structures of the networks are similar (often number of types of trees in the dataset is about ≈ 3).

5 Experiments and Results

The experimental studies in this paper are a two-step procedure because the algorithm models multiresolution data in two steps. Firstly, we learn the component distributions in a Bayesian network framework from different resolutions of the data. Secondly, we model multiresolution data after transforming the Bayesian networks to vectors using mixture models.

5.1 Experiments with Bayesian Networks of Multiresolution Data

From the knowledge of multiresolution data relationships in Section 4, we generate the five different Bayesian networks abreast adjacently along the dimensionality of the data. We use BRMLToolbox to encode and generate Bayesian

network [22]. We use a maximum likelihood framework to learn the conditional probabilities of the network.

As discussed in Section 2.2, some of the resolutions (entire datasets) can be missing in a multiresolution scenario. We use the prowess of Bayesian networks in handling missing data [12] to impute missing resolutions. In experimental setup: firstly, we learn the parameters of the component distributions in Bayesian network framework via maximum likelihood. We then ascertain the performance of component distributions as Bayesian network models especially with respect to their ability to impute missing values. We artificially generate two scenarios: where one, and two resolutions of data are missing. We draw the samples from a consistent junction tree in the Bayesian network under the given evidence using BRMLToolbox [22]. The number of samples equal that of the original dataset. We then compare the re-sampled data with the original data.

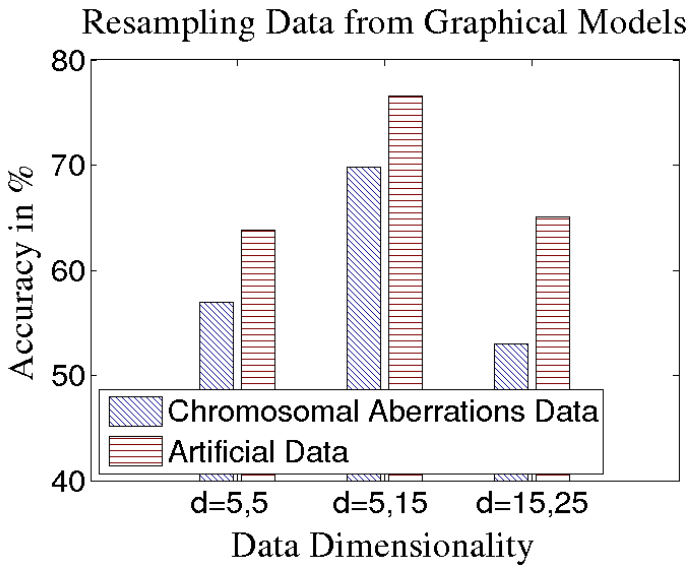

Fig. 5. The accuracy in re-sampling the data in missing resolutions conditioned on the data in other available resolutions. Comma in the X-axis separates the dimensionalities of the artificial data and the chromosomal aberration data, i.e. d=x,y denotes dimensionality (d) = (chromosomal aberrations data dimension (x), artificial data dimension (y)).

We calculate the matrix difference between the original data and the data sampled from the Bayesian networks. The difference in the binary data is sum of the number of places where 0s are 1s, and 1s are 0s. The difference is comparatively small in smaller dimensions than in the larger dimensions because the cumulative difference depends on the size of the dataset. Therefore, we calculated the accuracy of element-wise matching of two datasets as shown in Figure 5. Accuracy is the percentage of places where each element of both the matrices are equal.

In artificial data, when the data resolution with dimensionality 15 is missing, other data resolutions with dimensionality 5, and 25 are available, and we need to impute only data in resolution 15. Similarly, Data resolutions with dimensionalities 5, and 25 are missing when the data resolution with dimensionality 15 is available. Therefore, we should impute both the resolutions with dimensionalities 5, and 15. In this case, we can simply run single resolution analysis. However, we try to artificially create this scenario to demonstrate that our algorithm performs reliably under harsh conditions of large amount of missing values. In chromosomal aberrations data, coarse and the in–between data resolutions have the same dimensionality of 5. Therefore, when the in–between data resolution with dimensionality 5 is missing, coarse, and fine data resolution with dimensionality 5, and 15 are available, respectively. Similarly, the coarse, and fine data resolutions with dimensionalities 5, and 15 are missing when the data in the in–between resolution with dimensionality 5 are available.

The results in Figure 5 show that accuracy of matching is higher when two resolutions of data are available and only the data in a single resolution are missing. When only one data resolution is available, and we need to impute two resolutions in the coarse and the fine resolution, the accuracy is poorer. This result is intuitive because the number of known variables is smaller than the number of missing variables when two data resolutions are missing. Similarly, accuracy is poorer in high dimensional data (fine resolution) compared to data with lower dimensionality (coarse resolution). This discrepancy is the result of the curse of dimensionality phenomenon. Overall, the results show that the model of component distributions as Bayesian networks produces plausible results.

5.2 Experiments on Mixture Modeling of Multiresolution Data

In experimental setup, firstly, we transform the multiresolution data to the Bayesian network representation as shown in the Figure 3. Secondly, we transform the Bayesian network representation to the vector representation after imputing missing values (if any) as explained in Section 3. In the vector representation, the transformed multiresolution data have same dimensionality. The EM algorithm learns the mixture model with a priori knowledge of the number of components for data. As in [11,27,28], we use model selection in a 10-fold cross-validation setting to select the appropriate number of mixture components.

We train models of different complexities in a ten-fold cross–validation setting and select the model with the best generalization performance. Figure 6 shows that both training and validation likelihood steadily increases until the number of components is 5, then smoothen and flatten after the number of components is 5. This suggests that 5 is the appropriate number of mixture components.

After selecting the number of components, we train 200 different models of the same complexity and choose the model that produces the best likelihood on the data to ameliorate the problem of local optima in the EM algorithm. We also perform similar experiments with data in each resolution to select the number of mixture components and train the mixture model as a comparison with the results of the multiresolution model. Table 1 shows the variation in the number

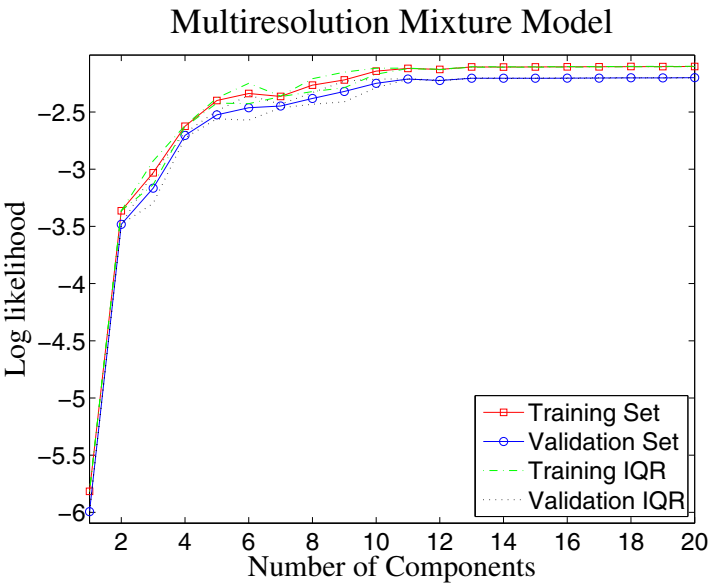

Fig. 6. Model selection in a 10-fold cross-validation setting in multiresolution artificial data. Final results for the same model selection are denoted by a boldfaced row in Table 1. Averaged training and validation likelihood along with their corresponding Inter Quartile Range (IQR) for each training and validation run has also been plotted. The selected number of components is 5.

Table 1. The results of mixture modeling on single resolution and multiresolution models. Here, J and $\mathcal{L}$, denote the selected number of components and the log likelihood obtained by the best model, respectively.

Artificial Data			**Chromosome-17**		
Datasets	**Results**		**Datasets**	**Results**	
(Dimension)	**J**	$\mathcal{L}$	**Dimension)**	**J**	$\mathcal{L}$
Single Resolution (5)	3	-3.24	Single Resolution (5)	4	-2.23
Single Resolution (15)	6	-8.32	Single Resolution (5)	3	-2.17
Single Resolution (25)	7	-12.84	Single Resolution (15)	5	-3.73
Multiresolution (9)	**5**	**-2.40**	Multiresolution (5)	4	-2.14

of components required to fit the data in different resolutions. Furthermore, the likelihood is considerably smaller in single resolution showing improvement in mixture modeling because of the use of multiple resolutions.

Figure 7 shows the log likelihood of three single resolution models and a multiresolution model. We trained the mixture model initialized at random in a ten-fold cross-validation setting with the selected number of components to convergence, i.e. until the increase in log-likelihood is small, 0.0001 in the experiments. The shorter the bar the better the result as Y-axis depicts negative log likelihood.

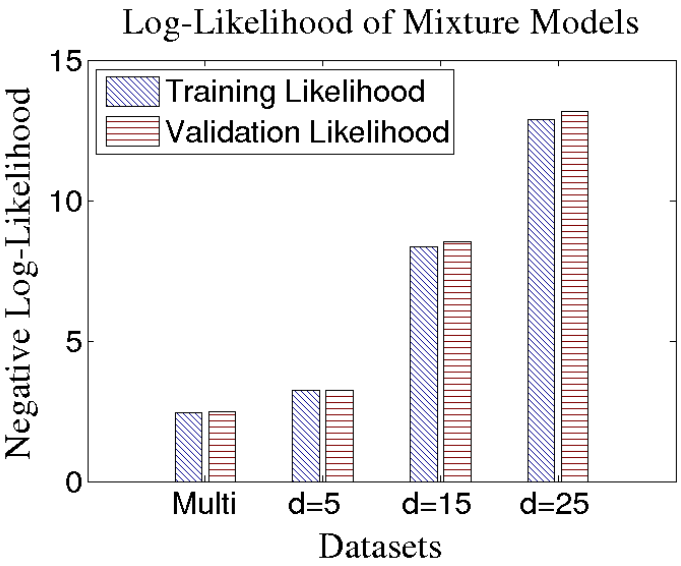

Fig. 7. The log likelihood of three mixture models in single resolution and a multiresolution mixture model trained in a 10-fold cross-validation setting after selecting the number of components. The Y-axis shows the negative log likelihood, therefore, the shorter the bar, the better the result.

We select a different number of components for each dataset as shown in Table 1. The results also show that multiresolution mixture model outperforms the single resolution models. Log likelihood is comparatively smaller in dimensionalities of 15, and 25 because of the increased dimensionality of the data. The likelihood of the multiresolution model is better than the data with the smallest dimensionality of 5 in single resolution although the dimensionality of the multiresolution data is 9. The Table 1 also shows similar results on chromosomal data.

6 Summary and Conclusions

In this paper, we proposed a mixture model of multiresolution components to model multiresolution 0-1 data. Firstly, we design the multiresolution components of the mixture model as Bayesian networks with the knowledge of the hierarchy of resolutions from the domain ontology. We then learn the CPD of the networks from the multiresolution data. Secondly, we transform the multiresolution component distributions to vector representation and learn the mixture model in a ten-fold cross validation setting. We experimented with the algorithm on a multiresolution artificial dataset and also on a multiresolution chromosomal aberration dataset. The experimental results show that the proposed approach of multiresolution modeling outperforms single resolution models.

Acknowledgments. The work is funded by Helsinki Doctoral Programme in Computer Science – Advanced Computing and Intelligent Systems (**Hecse**), and Finnish Centre of Excellence for Algorithmic Data Analysis Research (**ALGODAN**). I would also like to thank colleague Mikko Korpela for reading through the manuscript and suggesting the improvements in presentation.

References

1. Garland, M.: Multiresolution Modeling: Survey & Future Opportunities. In: Eurographics 1999 – State of the Art Reports, pp. 111–131 (1999)
2. Willsky, A.S.: Multiresolution Markov Models for Signal and Image Processing. Proceedings of the IEEE 90(8), 1396–1458 (2002)
3. Shaffer, L.G., Tommerup, N.: ISCN 2005: An International System for Human Cytogenetic Nomenclature(2005) Recommendations of the International Standing Committee on Human Cytogenetic Nomenclature. Karger (2005)
4. Lindeberg, T.: Scale-space theory: A basic tool for analysing structures at different scales. Journal of Applied Statistics 21(2), 224–270 (1994)
5. Vetterli, M., Kovačevic, J.: Wavelets and Subband Coding. Prentice-Hall, Inc., Upper Saddle River (1995)
6. Russell, B.: On the Relations of Universals and Particulars. Proceedings of the Aristotelian Society 12, 1–24 (1911)
7. Everitt, B.S., Hand, D.J.: Finite Mixture Distributions. Chapman and Hall, London (1981)
8. McLachlan, G.J., Peel, D.: Finite Mixture Models. Probability and Statistics – Applied Probability and Statistics Section, vol. 299. Wiley, New York (2000)
9. Moore, A.: Very Fast EM-based Mixture Model Clustering Using Multiresolution KD–trees. In: Kearns, M., Cohn, D. (eds.) Advances in Neural Information Processing Systems, pp. 543–549. Morgan Kaufmann (April 1999)
10. Meilâ, M., Jordan, M.I.: Learning with mixtures of trees. Journal of Machine Learning Research 1, 1–48 (2000)
11. Myllykangas, S., Tikka, J., Böhling, T., Knuutila, S., Hollmén, J.: Classification of human cancers based on DNA copy number amplification modeling. BMC Medical Genomics 1(15) (May 2008)
12. Marlin, B.M.: Missing data problems in machine learning. PhD thesis, University of Toronto (2008)
13. Kirsch, I.R.: The Causes and Consequences of Chromosomal Aberrations, 1st edn. CRC Press (December 1992)
14. Adhikari, P.R., Hollmén, J.: Patterns from multiresolution 0-1 data. In: Proceedings of the ACM SIGKDD Workshop on Useful Patterns, UP 2010, pp. 8–16. ACM, New York (2010)
15. Adhikari, P.R., Hollmén, J.: Multiresolution Mixture Modeling using Merging of Mixture Components. In: Hoi, S.C.H., Buntine, W. (eds.) Proceedings of the Fourth Asian Conference on Machine Learning, ACML 2012, JMLR Workshop and Conference Proceedings, Singapore, vol. 25, pp. 17–32 (2012)
16. Wilson, R.: MGMM: multiresolution Gaussian mixture models for computer vision. In: Proceedings of 15th International Conference on Pattern Recognition, vol. 1, pp. 212–215 (2000)

17. Ng, S.-K., McLachlan, G.J.: Robust Estimation in Gaussian Mixtures Using Multiresolution Kd-trees. In: Sun, C., Talbot, H., Ourselin, S., Adriaansen, T. (eds.) Proceedings of the 7th International Conference on Digital Image Computing: Techniques and Applications, pp. 145–154. CSIRO Publishing (2003)
18. Bellot, D.: Approximate discrete probability distribution representation using a multi–resolution binary tree. In: Proceedings of 15th IEEE International Conference on Tools with Artificial Intelligence, pp. 498–503 (2003)
19. Sanchís, F.A., Aznar, F., Sempere, M., Pujol, M., Rizo, R.: Learning Discrete Probability Distributions with a Multi-resolution Binary Tree. In: Corchado, E., Yin, H., Botti, V., Fyfe, C. (eds.) IDEAL 2006. LNCS, vol. 4224, pp. 472–479. Springer, Heidelberg (2006)
20. Bianchini, M., Maggini, M., Sarti, L.: Object Recognition Using Multiresolution Trees. In: Yeung, D.-Y., Kwok, J.T., Fred, A., Roli, F., de Ridder, D. (eds.) SSPR&SPR 2006. LNCS, vol. 4109, pp. 331–339. Springer, Heidelberg (2006)
21. Huerta, J., Chover, M., Quiros, R., Vivo, R., Ribelles, J.: Binary space partitioning trees: a multiresolution approach. In: Proceedings of 1997 IEEE Conference on Information Visualization, pp. 148–154 (1997)
22. Barber, D.: Bayesian Reasoning and Machine Learning. Cambridge University Press (2012)
23. Jordan, M.I.: Graphical Models. Statistical Science (2004)
24. Heckerman, D.: A Tutorial on Learning With Bayesian Networks. In: Jordan, M.I. (ed.) Learning in Graphical Models, pp. 301–354. MIT Press, USA (1999)
25. Enders, C.K.: Applied Missing Data Analysis, 1st edn. The Guilford Press (2010)
26. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, Series B 39(1), 1–38 (1977)
27. Adhikari, P.R., Hollmén, J.: Fast Progressive Training of Mixture Models for Model Selection. In: Ganascia, J.-G., Lenca, P., Petit, J.-M. (eds.) DS 2012. LNCS (LNAI), vol. 7569, pp. 194–208. Springer, Heidelberg (2012)
28. Tikka, J., Hollmén, J., Myllykangas, S.: Mixture Modeling of DNA copy number amplification patterns in cancer. In: Sandoval, F., Prieto, A.G., Cabestany, J., Graña, M. (eds.) IWANN 2007. LNCS, vol. 4507, pp. 972–979. Springer, Heidelberg (2007)
29. Lu, X., Shaw, C.A., Patel, A., Li, J., Cooper, M.L., Wells, W.R., Sullivan, C.M., Sahoo, T., Yatsenko, S.A., Bacino, C.A., Stankiewicz, P., Ou, Z., Chinault, A.C., Beaudet, A.L., Lupski, J.R., Cheung, S.W., Ward, P.A.: Clinical Implementation of Chromosomal Microarray Analysis: Summary of 2513 Postnatal Cases. PLoS ONE 2(3), e327 (2007)