Multi-Assignment Clustering for Boolean Data

Andreas P. Streich[*] ANDREAS.STREICH@INF.ETHZ.CH
Mario Frank[*†] MARIO.FRANK@INF.ETHZ.CH
David Basin BASIN@INF.ETHZ.CH
Joachim M. Buhmann JBUHMANN@INF.ETHZ.CH

Department of Computer Science, ETH Zurich, Universitätsstrasse 6, CH-8006 Zurich, Switzerland
[*] equally contributing [†] corresponding author

Abstract

Conventional clustering methods typically assume that each data item belongs to a single cluster. This assumption does not hold in general. In order to overcome this limitation, we propose a generative method for clustering vectorial data, where each object can be assigned to multiple clusters. Using a deterministic annealing scheme, our method decomposes the observed data into the contributions of individual clusters and infers their parameters.

Experiments on synthetic Boolean data show that our method achieves higher accuracy in the source parameter estimation and superior cluster stability compared to state-of-the-art approaches. We also apply our method to an important problem in computer security known as role mining. Experiments on real-world access control data show performance gains in generalization to new employees against other multi-assignment methods. In challenging situations with high noise levels, our approach maintains its good performance, while alternative state-of-the-art techniques lack robustness.

1. Introduction

Data clustering is the unsupervised learning task of dividing data into groups. If the goal is not only to group data, but also to infer the hidden structure responsible for generating the data, a source is associated with each cluster. Conventional clustering methods assume that each data item belongs to a single cluster. This assumption is appropriate if membership in one group excludes membership in other groups and thus the dataset can be partitioned into homogeneous and disjoint subsets.

However, the assumption of mutually exclusive cluster memberships fails for many domains. The properties of many data sets can be better explained in the more general setting, where data items can belong to multiple clusters. Speaking in generative terms, a data item is interpreted as a combination of emissions of all the sources it belongs to — an interpretation similar to the one presented in (Streich & Buhmann, 2008). Consider, for instance, clustering the preferences of children: Being a chocolate addict does not exclude being fond of dinosaurs and thus being part of the reptile-lovers cluster.

This point of view goes far beyond fuzzy clustering, where the exclusivity constraint is just weakened: an object belongs to a given number of clusters each with some percentage (this necessitates less preference for chocolate if the preference for reptiles rises — explain this to your children!). If cluster memberships are represented as vectors of binary indicator variables, fuzzy clustering allows indicator values between 0 and 1. In ordinary clustering and fuzzy clustering, the membership indicators of each object sum to 1, whereas in Multi-Assignment Clustering, this sum can be any integer greater than or equal to 1.

In this paper, we present a novel approach, called Multi-Assignment Clustering (MAC), for clustering Boolean vectorial data that can simultaneously belong to multiple clusters. In our generative model, each component of each data vector is either drawn from a signal distribution (given by the clusters the data item belongs to) or from an independent global noise distribution. We present an expectation-maximization (EM-)algorithm where the source prototypes of the clusters and the cluster memberships of each data item are simultaneously estimated as well as the mixture

Appearing in *Proceedings of the 26^{th} International Conference on Machine Learning*, Montreal, Canada, 2009. Copyright 2009 by the author(s)/owner(s).

weight of the global noise source.

In experiments with synthetic data, our method recovers the cluster prototypes with significantly higher accuracy than alternative methods (see Section 2), especially for datasets with high noise level. Furthermore, the assignment of data items to clusters has superior stability under resampling.

We apply our model to an important problem arising in computer security: the automated engineering of roles for role-based access control (RBAC) or "role mining". An RBAC system is defined by assignments from users to roles and from roles to permissions. Roles can thus be interpreted as sets of permissions. A user is granted a permission if he is assigned to a role that contains this permission. The goal of role mining is to identify roles in an access-control system consisting of direct assignments of users to permissions.

The assignment of users to multiple roles is a property of RBAC, which is explicitly permitted by the NIST standard for role-based access control (Ferraiolo et al., 2001). This design decision simplifies the structure of the roles. Users can be equipped with a set of basic permissions (check e-mail, change desktop background, etc.) by a common role. Specialized permissions can then be granted by assigning users to additional roles that are not shared by all users. As users can be assigned to multiple roles, one can equip the users with needed permissions using only a small total number of roles. In contrast, if users could be only assigned to single roles, then each user would require a role specially tailored to his job. As a result, the total number of roles needed in the system would be much higher.

For enterprize systems with tens of thousands of users and permissions, manual role engineering is time-consuming, costly, and error-prone. Therefore, automated approaches that find meaningful roles are highly desirable. We report on experimental results on access-control data showing that our approach finds good solutions even under difficult conditions.

The remainder of the paper is organized as follows. We review related work in Section 2 before we introduce our model in Section 3 and derive the inference steps of the model parameters for a deterministic annealing algorithm. In Section 4, we report on experimental results with synthetic and real-world data. We conclude with a theoretical discussion of our method in Section 5 and an outlook in Section 6.

2. Related Work

We present here three state-of-the-art methods for clustering binary data and explain how they differ from our method and from each other.

Binary independent component analysis (BICA) is a linear factor model for binary data proposed by Kabán and Bingham in (2008). As in standard ICA, the cluster prototypes are assumed to be orthogonal. BICA employs non-negative mixing coefficients for the assignment to multiple clusters, which sum up to 1.

A Bayesian framework for inference in the Noisy-OR model (Pearl, 1988) is presented by Wood (2006). The Noisy-OR model combines emissions of multiple hidden sources by Boolean disjunction and adds a global drift towards 1. The Indian Buffet Process (IBP) (Griffiths & Ghahramani, 2005) is used as a prior for the cluster assignments. IBP assumes an infinite number of clusters of which only a finite set is responsible to explain the observed data. Wood (2006) proposes a Gibbs sampling scheme to infer the Noisy-OR parameters. We will refer to this model as the Infinite Noisy-OR (INO) throughout this article.

The Discrete Basis Problem solver (DBPs) (Miettinen et al., 2006) is a greedy algorithm that picks the cluster centroids from a candidate set. Candidates are computed using association rule mining (Agrawal et al., 1993). A predefined number of centroids is then chosen and assigned to the objects such that the data set is optimally approximated. DBPs has no underlying probabilistic model.

The role-mining problem (RMP) is presented in (Vaidya et al., 2007). The problem of automated engineering of roles, or role mining, goes back to (Kuhlmann et al., 2003). Since then, a number of combinatorial methods have been proposed that approximate a direct user-permission assignment matrix with roles as best possible, e.g. (Colantonio et al., 2008; Ene et al., 2008). Even though not originally designed for this application, we consider the DBPsolver as the best representative for these combinatorial methods. In (Frank et al., 2008) a probabilistic model corresponding to RMP is derived and its exclusive clustering variant is tested using the Gibbs sampling algorithm of Kemp et al. (2006). We did not find any probabilistic role-mining method that supports multi-cluster (multi-role) solutions.

3. Generative Data Model

In this section, we propose a model for clustering binary vectorial data. Let N be the number of objects, D the number of dimensions, and K the number of clusters. We are given a binary data matrix $\mathbf{x} \in \{0,1\}^{N \times D}$. Row i is denoted by $x_{i\cdot}$ and column j by $x_{\cdot j}$. This notation will be used for all matrices.

We aim to explain each $x_{i\cdot}$ as the disjunction of the prototypes of all clusters the data item i belongs to. The matrix of Boolean prototypes is denoted by $\mathbf{u} \in \{0,1\}^{K \times D}$, where $u_{k\cdot}$ denotes the prototype of the cluster k. The cluster memberships are coded by the binary assignment matrix $\mathbf{z} \in \{0,1\}^{N \times K}$, where z_{ik} states whether object i belongs to cluster k. With these entities, the decomposition of $\mathbf{x}$ can be formalized by the Boolean matrix product $\mathbf{x} = \mathbf{z} \otimes \mathbf{u}$, where $\otimes$ is defined such that $x_{ij} = \bigvee_k [z_{ik} \wedge u_{kj}]$.

3.1. Probabilistic Model

Finding optimal matrices $\mathbf{z}$ and $\mathbf{u}$ for the decomposition of $\mathbf{x}$ is NP-hard (Vaidya et al., 2007). A probabilistic representation allows us to drastically simplify the optimization problem. Therefore, we introduce independent random variables[1] $\beta_{kj} := p(u_{kj} = 0)$ for the deterministic centroids $\mathbf{u}$. The matrix of all Bernoulli parameters is denoted by $\beta \in [0,1]^{K \times D}$.

We propose a mixture model where x_{ij} is either drawn from a signal or a noise component. The probability of x_{ij} under the signal model is

$$p_S(x_{ij} \mid \mathbf{z}, \beta) = \left[1 - \prod_{k=1}^{K} \beta_{kj}^{z_{ik}}\right]^{x_{ij}} \left[\prod_{k=1}^{K} \beta_{kj}^{z_{ik}}\right]^{1-x_{ij}} . \quad (1)$$

Note that some x_{ij} might be 1 because any of the sources to which the data item i belongs emits a 1 in dimension j. Conversely, x_{ij} is 0 only if all contributing sources emit a 0 in dimension j. This is reflected in (1) by the product over k. To simplify notation, we modify this expression: We replace the indicator vector $z_{i\cdot}$ by the assignment set $\mathcal{L}_i$, containing the indices of all clusters that $x_{i\cdot}$ belongs to, i.e. $\mathcal{L}_i := \{k \in \{1,..,K\} \,|\, z_{ik} = 1\}$. $\mathbb{L}$ denotes the set of all possible assignment sets which, if no constraints are imposed, is the power set of the set of clusters. Accordingly, we define $\beta_{\mathcal{L}_ij} := \prod_{k \in \mathcal{L}_i} \beta_{kj}$. As the conjunction of two Bernoulli distributed binary random variables is again Bernoulli distributed, $\beta_{\mathcal{L}_ij}$ can be interpreted as the parameter of the Bernoulli-distribution describing the data with label set $\mathcal{L}_i$. However, we emphasize that $\beta_{\mathcal{L}_ij}$ is only used for notational convenience. In all computations, it will be computed based on the parameters of single clusters, which are the only source parameters of the model.

With this notation, the probability distribution of x_{ij} given the parameters of the signal model is

$$p_S(x_{ij} \mid \mathcal{L}_i, \beta) = [1 - \beta_{\mathcal{L}_ij}]^{x_{ij}} [\beta_{\mathcal{L}_ij}]^{1-x_{ij}} . \quad (2)$$

Alternatively, x_{ij} can be sampled from a global noise source with independent Bernoulli distribution parameterized by r (indicating the probability of a 1), i.e.

$$p_N(x_{ij} \mid r) = r^{x_{ij}} (1-r)^{1-x_{ij}} . \quad (3)$$

Unlike the Noisy-OR model, this noise process is symmetric and can produce both 0s and 1s.

The indicator variable ξ_{ij} defines whether x_{ij} is sampled from the signal distribution (ξ_{ij}=0) or the noise distribution (ξ_{ij}=1). The combined distribution of x_{ij} is thus

$$p_M(x_{ij} \mid \mathcal{L}_i, \beta, r, \xi_{ij}) = p_N(x_{ij} \mid r)^{\xi_{ij}} \, p_S(x_{ij} \mid \mathcal{L}_i, \beta)^{1-\xi_{ij}} .$$

We assume that the elements of the matrix $\mathbf{x}$ are conditionally independent given the model parameters. Furthermore, we consider ξ_{ij} to be Bernoulli distributed with the parameter $\epsilon := p(\xi_{ij} = 1)$ called *bit randomization probability.* Thus we have

$$p_M(\mathbf{x} \mid \mathbf{z}, \beta, r, \xi) = \prod_{i,j} p_M(x_{ij} \mid \mathcal{L}_i, \beta, r, \xi)$$

$$p_M(\mathbf{x}, \xi \mid \mathbf{z}, \beta, r, \epsilon) = p_M(\mathbf{x} \mid \mathbf{z}, \beta, r, \xi) \prod_{i,j} \epsilon^{\xi_{ij}} (1-\epsilon)^{1-\xi_{ij}} .$$

Marginalizing out ξ, we get the data likelihood as

$$\begin{aligned} p_M(\mathbf{x} \mid \mathbf{z}, \beta, r, \epsilon) &= \sum_{\{\xi\}} p_M(\mathbf{x}, \xi \mid \mathbf{z}, \beta, r, \epsilon) \\ &= \prod_{i,j} (\epsilon \cdot p_N(x_{ij}) + (1-\epsilon) \cdot p_S(x_{ij})) . \end{aligned}$$

The costs of assigning a data item i to a label set $\mathcal{L}$ is defined as the negative logarithm of the likelihood of the data item, given its assignment to the label set $\mathcal{L}$:

$$R_{i,\mathcal{L}} = -\sum_j \log \left[\epsilon \cdot p_N(x_{ij}|r) + (1-\epsilon) \cdot p_S(x_{ij}|\mathcal{L}, \beta)\right] .$$

The normalized responsibilities $\gamma_{i,\mathcal{L}}$ and the Lagrange functional F are defined as:

$$\gamma_{i,\mathcal{L}} = \frac{c_{i,\mathcal{L}}}{\sum_{\mathcal{L}'} c_{i,\mathcal{L}'}} \qquad F = -T \sum_i \log \left[\sum_{\mathcal{L}} c_{i,\mathcal{L}}\right] , \quad (4)$$

with the auxiliary variable $c_{i,\mathcal{L}} := \exp(-R_{i,\mathcal{L}}/T)$. The expected risk over all data items and label sets is then

$$R := \mathbb{E}[R_{i,\mathcal{L}}] = \sum_i \sum_{\mathcal{L}} \gamma_{i,\mathcal{L}} R_{i,\mathcal{L}} . \quad (5)$$

3.2. Inference

The model parameters β_{pq} (for $p \in \{1, \ldots, K\}$ and $q \in \{1, \ldots, D\}$) as well as the expected bit randomization probability ϵ and the noise parameter r

[1] Defining β_{kj} as the probability of u_{kj} being zero simplifies computations as compared to standard Bernoulli parameters, which indicate the probability for a 1.

are inferred by deterministic annealing (DA) (Rose et al., 1992; Buhmann & Kühnel, 1993). DA is a gradient descent optimization method that provides a smooth transition from the uniform distribution (having maximum entropy H) to a solution with minimal expected risk R. The Lagrange functional is $F = R - TH$, where the Lagrange parameter T (the computational temperature) controls the trade-off. Minimizing F at a given temperature T is thus equivalent to maximizing the entropy H under a constraint on the expected risk R. By gradually decreasing the temperature T, we obtain a homotopy method, which keeps the gradient-based expectation maximization scheme from getting trapped in local minima.

In the estimation step, for each i and $\mathcal{L}$, the risk $R_{i,\mathcal{L}}$ of assigning data item i to the label set $\mathcal{L}$ and the resulting responsibility $\gamma_{i,\mathcal{L}}$ is computed given the current centroids β. In the maximization step, we first determine the optimal values for ϵ and r and then use these values in the equations to determine the optimal source parameters β. The individual steps are described below.

We choose an initial temperature and a constant rate cooling scheme ($T \leftarrow \alpha \cdot T$, with $\alpha < 1$) as described in (Rose et al., 1992). The optimization algorithm terminates if the responsibilities $\gamma_{i,\mathcal{L}}$ have converged to one of the cluster sets $\mathcal{L}$, for all objects i.

Bit-Randomization Probability ϵ: The extremality condition of the free energy with respect to ϵ, the expected ratio of noisy elements in the matrix $\mathbf{x}$, is given by

$$\frac{\partial F}{\partial \epsilon} = -\sum_i \sum_{\mathcal{L}} \gamma_{i,\mathcal{L}} \sum_j \left(g_{\mathcal{L},j,1}^{x_{ij}} \cdot g_{\mathcal{L},j,0}^{1-x_{ij}} \right) = 0 \ , \tag{6}$$

$$\text{with} \quad g_{\mathcal{L},j,1} := \frac{r - (1-\beta_{\mathcal{L},j})}{\epsilon r + (1-\epsilon)\,(1-\beta_{\mathcal{L},j})}$$

$$g_{\mathcal{L},j,0} := \frac{(1-r) - \beta_{\mathcal{L},j}}{\epsilon\,(1-r) + (1-\epsilon)\beta_{\mathcal{L},j}} \ .$$

The values of the source parameters β and the responsibilities γ are kept fixed in this part of the algorithm. The value of ϵ^* with $F'(\epsilon^*) = 0$ is determined using bisection search.

Noise Parameter r: The extremality condition for the probability r to emit a 1 by the noise process is

$$\frac{\partial F}{\partial r} = \sum_{i,j} \sum_{\mathcal{L}} \gamma_{i\mathcal{L}} \cdot h_{\mathcal{L},j,1}^{x_{ij}} \cdot h_{\mathcal{L},j,0}^{1-x_{ij}} = 0 \ , \tag{7}$$

$$\text{with } h_{\mathcal{L},j,1} := -1/\left(\epsilon r + (1-\epsilon)\,(1-\beta_{\mathcal{L}j})\right)$$

$$h_{\mathcal{L},j,0} := 1/\left(\epsilon\,(1-r) + (1-\epsilon)\beta_{\mathcal{L}j}\right) \ .$$

Source Parameters β: We derive the optimality condition for F with respect to β_{pq} as

$$\frac{\partial F}{\partial \beta_{pq}} = (1-\epsilon) \sum_i \sum_{\mathcal{L} \in \mathcal{L}_p} \left(f_{\mathcal{L},j,1}^{x_{iq}} \cdot f_{\mathcal{L},j,0}^{1-x_{iq}} \cdot \gamma_{i,\mathcal{L}} \right) = 0 \ , \tag{8}$$

$$\text{with } f_{\mathcal{L},j,1} := -\beta_{\mathcal{L}\backslash p,q} / \left(\epsilon r + (1-\epsilon)\,(1-\beta_{\mathcal{L},q})\right)$$

$$f_{\mathcal{L},q,0} := \beta_{\mathcal{L}\backslash p,q} / \left(\epsilon\,(1-r) + (1-\epsilon)\beta_{\mathcal{L},j}\right)$$

$$\beta_{\mathcal{L}\backslash p,j} := \prod_{k \in \mathcal{L}, k \neq p} \beta_{k,j} \ ,$$

with $\mathcal{L}_p := \{\mathcal{L} \in \mathbb{L} | p \in \mathcal{L}\}$. Equation 8 determines the value β_{pq} that minimizes F. The corresponding bisection search typically needs less than a dozen iterations to determine these values. ϵ and r are fixed to the value computed in the first part of the maximization step. $\gamma_{i,\mathcal{L}}$ is also kept fix while both $\beta_{\mathcal{L},q}$ and $\beta_{\mathcal{L}\backslash p,q}$ are computed based on the source parameters β. We use the values of the previous iterations for all $\beta_{p'q'}$, with $p' \neq p$ or $q' \neq q$. Only at the end of the maximization step are the previous values of β overwritten. These updates for the β-parameters are done independently for all p and q. Namely, the independence assumption for different cluster parameters in the same dimension is a simplification. However, it drastically reduces the computational costs, keeps the results independent of the order in which the computations are done, and distorts the results only negligibly.

The Matlab implementation of Multi-Assignment Clustering is available from the authors.

4. Experiments and Results

In this section, we first introduce performance measures and then present experimental results on synthetic data. Afterwards, we report on our findings for role mining on a user-permission assignment dataset from a large enterprize.

4.1. Evaluation criteria

The lack of a general formal objective function in clustering implies the absence of a unique general evaluation criterion for clustering solutions. We evaluate our experimental results using the measures that we introduce in this section. We denote by $\hat{\mathbf{u}}$ and $\hat{\mathbf{z}}$ the estimated decomposition of the data matrix $\mathbf{x}$. $\hat{\mathbf{x}}$ denotes the reconstruction of $\mathbf{x}$, i.e. $\hat{\mathbf{x}} := \hat{\mathbf{u}} \otimes \hat{\mathbf{z}}$. Furthermore, in tests on synthetic data, we make comparisons between the noise-free data matrix $\mathbf{x}^S := \mathbf{z} \otimes \mathbf{u}$ and the noisy data matrix $\mathbf{x}$.

Coverage Rate: This is the ratio of the number of matrix elements that are 1 in both $\mathbf{x}$ and $\hat{\mathbf{x}}$ to the number of elements equal to 1 in $\mathbf{x}$, i.e.

$$\text{cov} := |\{(i,j)|x_{ij} = \hat{x}_{ij} = 1\}| \,/\, |\{(i,j)|x_{ij} = 1\}| \ .$$

Coverage is a standard evaluation criterion in the role-mining literature.

Instability: The notion of stability captures the requirement of stable clustering. Namely, a clustering solution based on a dataset should be reproducible for a different data set drawn from the same distribution (Lange et al., 2004). Training a classifier using the cluster solution of one data set and applying it to a second data set gives the desired measure. The instability of the clustering solution is the minimum disagreement under all permutations between the classification solution and the clustering result on the second dataset. Formally, let $\mathbf{x}^{(1)}$ ($\mathbf{x}^{(2)}$) be the first (second) data set, each containing N samples and let $\hat{\mathbf{z}}^{(1)}$ ($\hat{\mathbf{z}}^{(2)}$) be the cluster assignments inferred on the first (second) data set. Let $\phi_{(1)}$ be the classifier trained on $\mathbf{x}^{(1)}$, with $\hat{\mathbf{z}}^{(1)}$ used as label sets. Note that we need a classifier capable of handing multi-labels. Finally, let $\pi_{|\mathbb{L}|}$ be a permutation on $|\mathbb{L}|$ objects. Then, the instability is computed as

$$\frac{|\mathbb{L}|}{|\mathbb{L}|-1}\frac{1}{N}\min_{\pi_{|\mathbb{L}|}}\left\{\sum_{i=1}^{n} 1_{\pi_{|\mathbb{L}|}\left(\phi_{(1)}(x_i^{(2)})\right)\neq \hat{z}_{i\cdot}^{(2)}}\right\} \ ,$$

where equality means equality of the complete assignment vector $\hat{z}_{i\cdot}$, i.e. the data item is assigned to the same set of clusters. We use a nearest neighbor classifier with Hamming distance for $\phi_{(1)}$. As a random assignment to one of $|\mathbb{L}|$ cluster sets achieves a stability measure of $1 - \frac{1}{|\mathbb{L}|}$, the factor $\frac{|\mathbb{L}|}{|\mathbb{L}|-1}$ allows us to compare clustering solutions with different numbers of clusters.

Average Hamming Distance: This measure states how accurately the centroids of the underlying clusters are estimated.

$$hamm := \frac{1}{K}\min_{\pi_K}\sum_k d^H(u^S_{k\cdot}, \hat{u}_{\pi_K(k)\cdot}) \ ,$$

where π_K is a permutation of K elements and $d^H(\cdot,\cdot)$ is the Hamming distance between two binary vectors.

4.2. Experiments on Synthetic Data

We ran experiments with artificially generated data in order to compare the estimators of the source prototypes obtained by different clustering techniques. For the results presented in this section, we first chose a set of three source prototypes as illustrated in Fig. 1. An object can be assigned to any combination of these sources. A data item that is assigned to none of the sources is explained solely by the noise process.

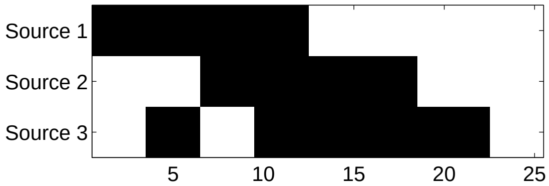

Figure 1. Prototypes of the three sources used to generate the synthetic data. Black indicates a 1, white a 0.

Afterwards, we sampled equally many data items from single sources and from the disjunction of multiple sources. Furthermore, some data items were generated by the noise distribution only. We ran experiments with different noise levels from 0% to 90%, for a total number of 500 data items. The results for MAC as well as for Infinite Noisy-OR (INO), Discrete Basis Problem solver (DBPs), and Binary Independent Component Analysis (BICA) are illustrated in Fig. 2.

The average Hamming distance between the true and estimated centroids is shown in the upper graph (this magnitude equals the percentage of differing bits). The MAC algorithm outperforms its three competitors on all noise levels below 80%. Since the orthogonality assumption of BICA is not fulfilled by the data, its estimated centroids have a high deviation from the true ones. INO usually chooses a higher number of sources than the number K used to generate the data. In order to compare with the true centroids, we took the K sources that agreed the best, thereby granting this method a slight advantage.

In terms of stability (lower graph), all methods are affected by increasing noise. BICA is relatively instable also on noise-free data but seems to be a bit more robust to noise than DBPs and INO. Except for very high noise levels ($> 70\%$), MAC achieves higher stability than the other three methods.

4.3. Experiments on Real-World Data

In this section, we evaluate our algorithm on the role-mining problem. The dataset we use for our experiments comes from the IT system of a bank from which we were given a user-permission assignment matrix with 4900 users and 1300 permissions.

Before explaining the results, we first describe the properties of a good set of roles. A role-based access control system should generalize well to new users. The existing set of roles should suffice to endow a new employee with all permissions he needs to accomplish his tasks. In contrast, a role system that

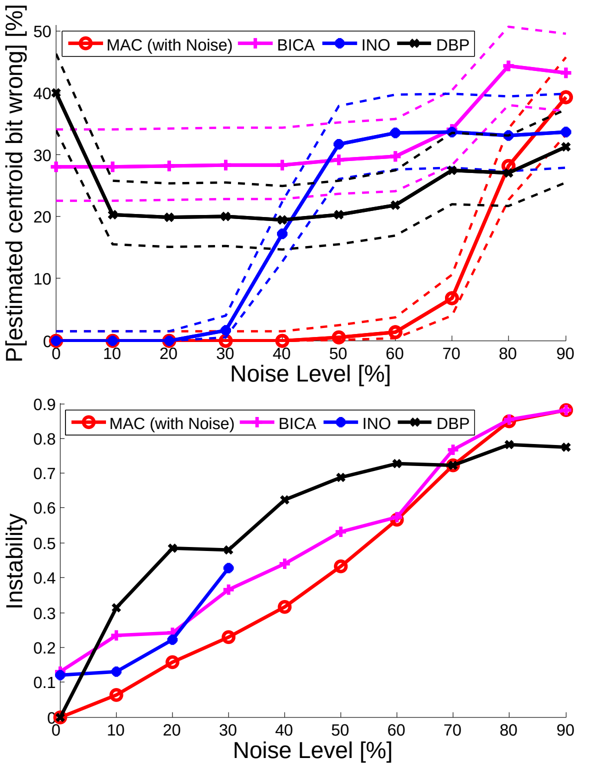

Figure 2. Average Hamming distance between true and estimated source prototypes (upper graph) and cluster instability (lower graph) on synthetic data. Solid lines indicate the average over 10 data sets with random noise, the dashed lines standard deviation. For INO, we only plotted instability results of experiments where the same number of clusters was inferred in both data sets. To improve readability, sample standard deviation (around 0.1) is not plotted in the lower graph.

generalizes poorly may well replace the existing direct user-permission assignments but will require the construction of new roles when new users are added.

Given the above, our emphasis in these experiments is on the ability of the model to generalize to new data. In order to quantify this ability, we split the given data in two disjoint sets. The first 2000 users are used to infer the roles. We randomly chose a ratio κ of permissions from the remaining users in order to decide which roles to give them. The generalization ability of the role system is determined by how accurately the permissions of the remaining users are predicted by these roles. In the left column of Fig. 3, we plot the coverage (upper graph) and the prediction error (lower graph) for this generalization experiment.

Our algorithm took roughly 2 hours (Matlab) to compute these results. Both DBPs (C++) and BICA (Matlab) finished within 15 minutes. The implementation of INO (Matlab) provided by the authors of (Wood, 2006) ran for roughly two weeks.

The original dataset has a relatively simple structure and does not suffice to assess the capabilities of the algorithms to deal with noisy, complicated data. In order to simulate a more complex scenario, we produced artificial users by Boolean addition of the first and second 500 permissions of each user. The resulting user-permission matrix exhibited substantially more structure. Additionally, 33% of the matrix entries were replaced by random bits to increase the noise level.

To evaluate the generalization performance, we again took the unused set of users of the modified user-permission matrix, but without the artificial noise. The results are shown in the right column of Fig. 3. Compared with the results on the original data set, MAC is clearly more robust with respect to the additional noise and still recovers roles that allow one to accurately describe the permissions of new users. Both DBPs and BICA are significantly worse in both coverage and prediction error. INO seems to be more robust against this additional noise, but is also clearly outperformed by MAC.

5. Theoretical Considerations

In this section, we describe some of our model's theoretical properties. Recall that $\mathbb{L}$ is the set of admissible cluster assignments. Let $L := |\mathbb{L}|$ be the cardinality of $\mathbb{L}$ and M the maximal number of clusters an object can belong to. $\mathbb{L}$ can be encoded as a binary membership matrix $\mathbf{z}^{\mathbb{L}} \in \{0,1\}^{L \times K}$. We denote a row in $\mathbf{z}^{\mathbb{L}}$ as a cluster set. The l^{th} row indicates the clusters contained in the cluster set l (we assume an arbitrary, but fixed numbering of the cluster sets). We can now decompose the assignment matrix $\mathbf{z}$ as $\mathbf{z} = \mathbf{z}^{\mathcal{L}} \otimes \mathbf{z}^{\mathbb{L}}$, with $\mathbf{z}^{\mathcal{L}} \in \{0,1\}^{N \times L}$ encoding the exclusive assignment of data items to cluster sets.

With this notation, the decomposition $\mathbf{x}^S = \mathbf{z} \otimes \mathbf{u}$ can be extended to $\mathbf{x}^S = \left(\mathbf{z}^{\mathcal{L}} \otimes \mathbf{z}^{\mathbb{L}}\right) \otimes \mathbf{u}$, which in turn is equivalent to $\mathbf{x}^S = \mathbf{z}^{\mathcal{L}} \otimes \left(\mathbf{z}^{\mathbb{L}} \otimes \mathbf{u}\right)$. The inflated matrix $\mathbf{u}^{SAC} := \mathbf{z}^{\mathbb{L}} \otimes \mathbf{u}$ can be interpreted as the cluster centroids in the corresponding clustering problem with mutually exclusive memberships (Single-Assignment Clustering). While such a scenario is asymptotically equivalent to the model presented in this paper, it ignores the high dependency between centroids associated with different cluster sets and must estimate a much larger number of parameters. This explains why, for finite sample size, the proposed model yields more accurate parameter estimates, at the price of a more involved optimization problem. Details will be provided in a forthcoming publication.

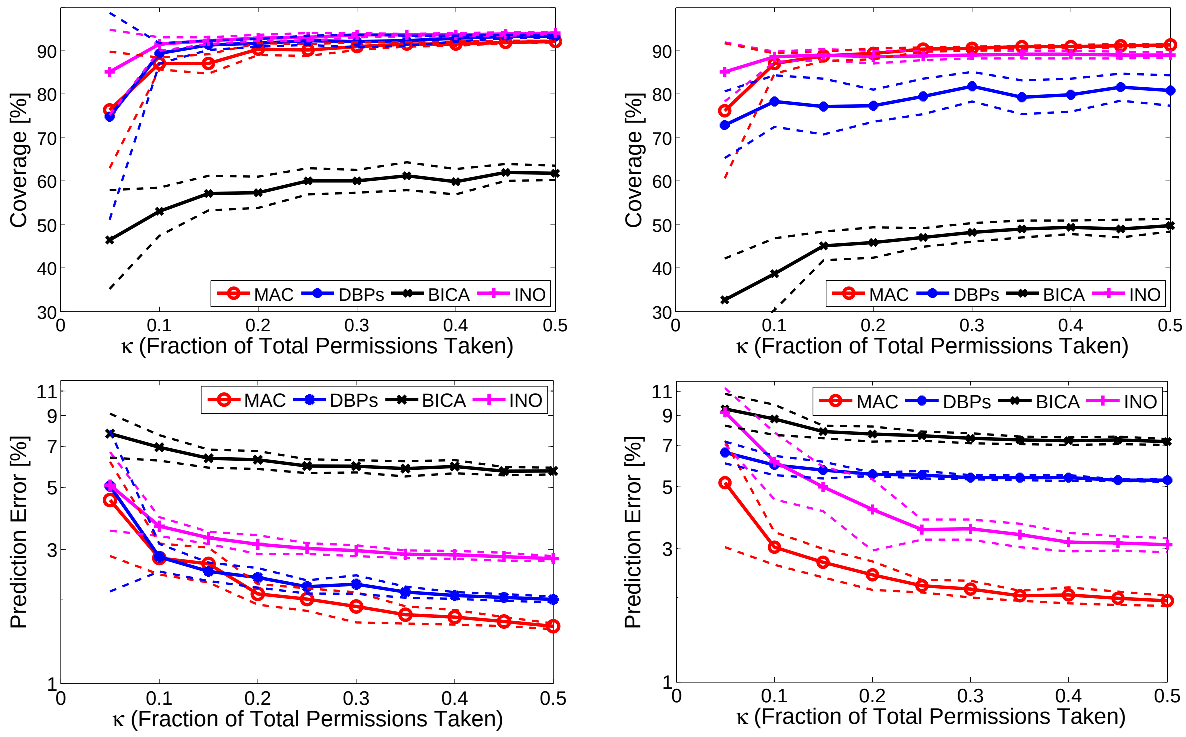

Figure 3. Generalization error of roles inferred from the original data set (left column) and from the modified data set with higher complexity and 33% noise (right column). 30 sources are used for the methods MAC, DBPs, and BICA. Both data sets consist of 2000 users and 500 permissions. The estimated bit randomization probability ϵ is 3% on the original data, and 38% on the modified data. r is estimated to be around 50%. We use a logarithmic scale in the plots for the prediction error in order to improve the resolution in the low-error part. The plots show that the proposed MAC method is competitive on the original data and clearly outperforms DBPs, BICA, and INO under conditions with high noise.

The formulation introduced above highlights a possible challenge of the proposed model, namely the large number of possible cluster sets, which can make the computations for parameter estimation, (6), (7) and (8), very time-consuming. Note that in general, we have $L = 2^K$ or $L = \sum_{m=0}^{M} \binom{K}{M}$ if the number of clusters an object can be assigned to is bounded by M. However, we emphasize that the large complexity is due to the data itself and not by the model that tries to explain it. Using a smaller number of clusters or cluster sets than in the generative process necessarily leads to a mismatch between the real underlying distribution and the distribution inferred by the clustering algorithm. The proposed formalism allows one to control the model complexity in a transparent way. Using prior knowledge on the complexity from the application, we can thus drastically reduce the running time compared to nonparametric methods, which do not offer such a mechanism.

Finally, we point out that the described model has an inherent preference for a sparse centroid matrix $\mathbf{u}$: Since the disjunction $\vee$ has no inverse, the number of 1s in the matrix row $\mathbf{x}_{i\cdot}$ is non-decreasing in the number of 1s in row $\mathbf{z}_{i\cdot}$ of the assignment matrix. Probabilistically speaking, the probability $\beta_{\mathcal{L}_i,j}$ of $x_{ij} = 0$ is the product of the factors $\beta_{k,j} \in [0,1]$ and thus decreases with the number of factors $|\mathcal{L}|$.
In order to explain matrix rows with few entries 1, the prototypes are to be chosen sparsely, while data rows containing a larger number of 1s are explained as the superposition of several sparse prototypes. Conversely, in a setting where the sources are likely to emit 1s in many dimensions, sparse rows of the data matrix are explained by the overall noise process. Such a configuration is thus less likely than a setting with sparse sources. Hence, it will be dropped in favor of a configuration with sparse source prototypes.
We believe that this preference for sparse prototypes pushes the learning algorithm away from a large number of local optima. We assume that this mechanism is the reason for the superior stability and source parameter retrieval observed in the experiments, as presented in Section 4.
In the application of role engineering, roles with a min-

imal number of permission, which suffice to grant the users needed permissions, correspond to an important security design principle, termed *least privilege*: Users should have as few permissions as possible, since assigning superfluous permissions to a user constitutes an unnecessary security risk. Our results obtained for the coverage and the prediction error show that MAC outperforms competing methods in prediction accuracy (see Fig. 2, upper graph) and thus minimizes security risks.

6. Conclusion

We have presented a new approach to clustering complex Boolean data where data items can be assigned to multiple clusters.

We carried out comparative studies on synthetic Boolean data sampled from a generative model. These experiments demonstrate that our approach produces cluster assignments of superior stability and more accurate cluster parameter estimates compared to state-of-the-art approaches. Moreover, the generalization performance of our approach is substantially more robust with respect to the addition of noise than competing models. Our experimental results on access-control data show an analogous behavior. Our method outperforms the state-of-the-art approaches and the most pronounced improvements are in scenarios with complex and noisy user-permission assignments.

Acknowledgements: The work of AS was in part funded by CTI grant Nr. 8539.2;2 EPSS-ES. This work was partially supported by the Zurich Information Security Center. It represents the views of the authors.

References

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *Int Conf on Mngm of Data*, *22*, 207–216.

Buhmann, J., & Kühnel, H. (1993). Vector quantization with complexity costs. *IEEE Trans on Information Theory* (pp. 1133–1145).

Colantonio, A., Di Pietro, R., & Ocello, A. (2008). A cost-driven approach to role engineering. *Symp on Appl Comp* (pp. 2129–2136).

Ene, A., Horne, W., Milosavljevic, N., Rao, P., Schreiber, R., & Tarjan, R. E. (2008). Fast exact and heuristic methods for role minimization problems. *Symp on Access Control Models and Technologies* (pp. 1–10).

Ferraiolo, D. F., Sandhu, R., Gavrila, S., Kuhn, D. R., & Chandramouli, R. (2001). Proposed NIST standard for role-based access control. *ACM Trans Inf Syst Secur*, *4*, 224–274.

Frank, M., Basin, D., & Buhmann, J. M. (2008). A class of probabilistic models for role engineering. *Conf on Computer and Communications Security* (pp. 299–310).

Griffiths, T. L., & Ghahramani, Z. (2005). Infinite latent feature models and the indian buffet process. *Conf on Neural Information Processing Systems* (pp. 475–482).

Kabán, A., & Bingham, E. (2008). Factorisation and denoising of 0-1 data: A variational approach. *Neurocomputing*, *71*, 2291 – 2308.

Kemp, C., Tenenbaum, J. B., Griffths, T. L., Yamada, T., & Ueda, N. (2006). Learning systems of concepts with an infinite relational model. *Nat Conf on Artificial Intelligence* (pp. 763–770).

Kuhlmann, M., Shohat, D., & Schimpf, G. (2003). Role mining - revealing business roles for security administration using data mining technology. *Symp on Access Control Models and Technologies* (pp. 179–186).

Lange, T., Roth, V., Braun, M. L., & Buhmann, J. M. (2004). Stability-based validation of clustering solutions. *Neural Computation*, *16*, 1299–1323.

Miettinen, P., Mielikäinen, T., Gionis, A., Das, G., & Mannila, H. (2006). The Discrete Basis Problem. *Proc of Principles and Practice of Knowledge Discovery in Databases* (pp. 335–346).

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems : Networks of plausible inference.* Morgan Kaufmann.

Rose, K., Gurewitz, E., & Fox, G. (1992). Vector quantization by deterministic annealing. *IEEE Trans on Information Theory* (pp. 2210–2239).

Streich, A. P., & Buhmann, J. M. (2008). Classification of multi-labeled data: A generative approach. *Europ Conf on Machine Learning* (pp. 390–405).

Vaidya, J., Atluri, V., & Guo, Q. (2007). The Role Mining Problem: Finding a minimal descriptive set of roles. *Symp on Access Control Models and Technologies* (pp. 175–184).

Wood, F. (2006). A non-parametric bayesian method for inferring hidden causes. *Conf on Uncertainty in Artificial Intelligence* (pp. 536–543).