Deriving Quantitative Models for Correlation Clusters*

Elke Achtert, Christian Böhm, Hans-Peter Kriegel, Peer Kröger, Arthur Zimek

Institute for Informatics
Ludwig-Maximilians-Universität München
Oettingenstr. 67, 80538 Munich, Germany
{achtert,boehm,kriegel,kroegerp,zimek}@dbs.ifi.lmu.de

ABSTRACT

Correlation clustering aims at grouping the data set into correlation clusters such that the objects in the same cluster exhibit a certain density and are all associated to a common arbitrarily oriented hyperplane of arbitrary dimensionality. Several algorithms for this task have been proposed recently. However, all algorithms only compute the partitioning of the data into clusters. This is only a first step in the pipeline of advanced data analysis and system modelling. The second (post-clustering) step of deriving a quantitative model for each correlation cluster has not been addressed so far. In this paper, we describe an original approach to handle this second step. We introduce a general method that can extract quantitative information on the linear dependencies within a correlation clustering. Our concepts are independent of the clustering model and can thus be applied as a post-processing step to any correlation clustering algorithm. Furthermore, we show how these quantitative models can be used to predict the probability distribution that an object is created by these models. Our broad experimental evaluation demonstrates the beneficial impact of our method on several applications of significant practical importance.

Categories and Subject Descriptors

I.5.3 [**Pattern Recognition**]: Clustering—*Algorithms*

General Terms

Algorithms

Keywords

Data mining, clustering, correlation clustering, cluster description, cluster model

*Partly supported by the German Ministry for Education, Science, Research and Technology (BMBF) under grant no. 031U212F within the BFAM project.

KDD'06, August 20–23, 2006, Philadelphia, Pennsylvania, USA.

1. INTRODUCTION

The detection of correlations between different features in a given data set is a very important data mining task. High correlation of features may result in a high degree of collinearity or even a perfect one, corresponding to approximate linear dependencies between two or more attributes. These dependencies can be arbitrarily compex, one or more features might depend on a combination of several other features. In the data space, dependencies of features are manifested as lines, planes, or, generally speaking, hyperplanes exhibiting a relatively high density of data points compared to the surrounding space. Knowing of correlations is traditionally used to reduce the dimensionality of the data set by eliminating redundant features. However, detection of correlated features may also help to reveal hidden causalities that are of great interest to the domain expert.

Recently, correlation clustering [6] has been introduced as a novel concept of knowledge discovery in databases to detect dependencies among features and to cluster those points that share a common pattern of dependencies. It corresponds to the marriage of two widespread ideas: First, correlation analysis performed e.g. by principle component analysis (PCA) and, second, clustering which aims at identifying local subgroups of data objects sharing high similarity. Correlation clustering groups the data set into subsets called correlation clusters such that the objects in the same correlation cluster are all associated to a common hyperplane of arbitrary dimensionality. In addition, many algorithms for correlation cluster analysis also require the objects of a cluster to exhibit a certain density, i.e. feature similarity.

Correlation clustering has been successfully applied to several application domains (see e.g. [3, 24, 6]). For example, costumer recommendation systems are important tools for target marketing. For the purpose of data analysis for recommendation systems, it is important to find homogeneous groups of users with similar ratings in subsets of the attributes. In addition, it is interesting to find groups of users with correlated affinities. This knowledge can help companies to predict customer behavior and thus develop future marketing plans. In molecular biology, correlation clustering is an important method for the analysis of several types of data. In metabolic screening, e.g., the collected data usually contain the concentrations of certain metabolites in the blood of thousands of patients. In such data sets, it is important to find homogeneous groups of patients with correlated metabolite concentrations indicating a common metabolic disease. Thus, several metabolites can be linearly

dependent on several other metabolites. Uncovering these patterns and extracting the dependencies of these clusters is a key step towards understanding metabolic or genetic disorders and designing individual drugs. A second example where correlation clustering is a sound methodology for data analysis in molecular biology is DNA microarray data analysis. Microarray data comprise the expression levels of thousands of genes in different samples such as experimental conditions, cells or organisms. Roughly speaking, the expression level of a gene indicates how active this gene is. The recovering of dependencies among different genes in certain conditions is an important step towards a more comprehensive understanding of the functionality of organisms which is a prominent aspect of systems biology. When the samples represent some patients, it is important to detect homogeneous groups of persons exhibiting a common linear dependency among a subset of genes in order to determine potential pathological subtypes of diseases and to develop individual treatments.

In all these cases, however, knowing merely of the existence of correlations among some features is just a first step. It is far more important to reveal quantitatively and as exactly as possible which features contribute to which dependencies as a second step. Having performed this second step, modelling a system becomes possible, that describes the respective underlying data quantitatively as well as qualitatively. Thus, in order to gain the full practical potentials from correlation cluster analysis, this second step is urgently needed. All existing approaches to correlation clustering usually focus only on the first step of detecting the clusters. To the best of our knowledge, there is no method for the second step of extracting quantitative correlation cluster information.

In this paper, we describe an approach to handle this second step of data analysis. We introduce general concepts for extracting quantitative information on the linear dependencies within a correlation cluster such that domain experts are able to understand the correlations and dependencies in their data. In fact, our method can be applied to any correlation clusters, regardless of what correlation clustering algorithm produced the results. As output, we obtain a set of linear equations that are displayed to the user. These equations can be used to understand the dependencies hidden in the analyzed data set and to create complex real-life models. As an example, how this information can be used for further analysis, we additionally introduce a framework to predict the probability that a new object is generated by a specific model of the derived ones.

The remainder of this paper is organized as follows. In Section 2 we review related work on correlation clustering and existing approaches for deriving descriptions of quantitative dependencies among several attributes. Our concepts to derive quantitative models of correlation clusters are proposed in Section 3. Section 4 presents a broad experimental evaluation where we demonstrate the practical importance of our approach. The paper concludes in Section 5.

2. RELATED WORK

2.1 Correlation Clustering

The expectation maximization (EM) [8] is one of the first clustering algorithms that can detect correlation clusters. The EM algorithm tries to model the data distribution of a data set using a mixture of non-axis parallel Gaussian distributions. Let us note that the EM algorithm cannot distinguish between correlation clusters and full-dimensional clusters without any correlation.

ORCLUS [3] is a k-means style correlation clustering algorithm and, thus, can be seen as a specialization of the EM algorithm that detects only correlation clusters. The correlation clusters are allowed to exist in arbitrarily oriented subspaces represented by a set of Eigenvectors. The number of clusters k and the average dimensionality l of the correlation clusters are the input parameters of ORCLUS.

In [6] the algorithm 4C, a combination of DBSCAN [9] and PCA, is presented to find correlation clusters. The user must specify several parameters, including: ε and μ, defining minimum density of a cluster, a threshold δ to decide which principal axes of a cluster are relevant for the correlation, and the dimensionality λ of the computed correlation clusters.

Let us note that none of the proposed approaches to correlation clustering provides a cluster model including an explicit description of the correlations within the cluster.

CURLER [20]aims at detecting arbitrary, non-linear correlations. It uses the concept of micro-clusters that are generated using an EM variant and then are merged to uncover correlation clusters. The correlations underlying the found clusters are not necessarily linear. Furthermore, CURLER assumes each data object to belong to all clusters simultaneously, but with different probabilities for each cluster assigned. By merging several clusters according to their cosharing level, the algorithm on the one hand becomes less sensitive to the predefined number of clusters k. On the other hand, the user becomes disabled to directly derive a model describing the correlations, since the original k models are no longer persistent in the resulting clustering. However, we focus on linear correlations between features. Thus, the non-linear correlations uncovered by CURLER are orthogonal to our approach.

Recently, several subspace clustering algorithms [4, 2, 14, 17, 5] were proposed to find clusters in axis-parallel projections of the data space. These algorithms are not able to capture local data correlations and find clusters of correlated objects since the principal axes of correlated data are arbitrarily oriented.

Pattern-based clustering methods [24, 21, 16, 15] aim at finding groups of objects that exhibit a similar trend in a subset of attributes. This problem is also known as coclustering or biclustering [12, 7]. In contrast to correlation clustering, pattern-based clustering limits itself to a very special form of correlation where all attributes are positively correlated. It does not include negative correlations or correlations where one attribute is determined by two or more other attributes. Thus, bi-clustering or pattern-based clustering could be regarded as a special case of correlation clustering, as more extensively discussed in [6].

2.2 Quantitative Association Rules

An interesting approach to derive descriptive models of quantitative relationships among subsets of attributes is known as quantitative association rule mining. Some earlier approaches to this task loose information requiring discretization of attributes (e.g. [19]) or representation of numerical values in a rule's right-hand side by some statistical characterizations, e.g. the mean or sum of the values (cf. [22]). Dis-

cretization of attributes, moreover, does not overcome the restriction to axis parallel dependencies. Recently, Rückert et al. [18] proposed to base quantitative association rules on half-spaces, thus allowing the discovery of non-axis-parallel rules and possibly accounting for cumulative effects of several variables. The rules derived by this approach are of the form "if the weighted sum of some variables is greater than a threshold, then a different weighted sum of variables is with high probability greater than a second threshold". This approach has been shown to be useful in detecting some rules of gene-expression data sets [10]. However, these association rules do not yet uncover continuous linear dependencies, but stick to certain thresholds, reflecting the boundaries of half-spaces.

2.3 Regression Analysis

A task very similar to the one tackled in this paper is linear and multiple regression analysis (e.g. cf. [11] for an overview). The general purpose of linear regression is to learn a linear relationship between a "predictor" variable and a "response" variable. Multiple regression extends this task by allowing multiple "predictor" variables. Other non-linear regression models can be used to learn non-linear relationships among the predictor and the response variables. However, the main difference between regression analysis and our approach is that in regression analysis, the predictor variables are assumed to be independent. Since correlation clusters are defined to consist of points that exhibit a linear dependency among a set of attributes, we want to identify these dependencies when deriving a quantitative model for each cluster. Obviously, we cannot define any independent variable(s), i.e. we cannot derive a set of predictor variables. Thus, regression analysis cannot be applied to derive quantitative models for correlation clusters as envisioned in this paper.

3. DERIVING QUANTITATIVE MODELS FOR CORRELATION CLUSTERS

In the following we assume $\mathcal{D}$ to be a database of n feature vectors in a d-dimensional real-valued feature space, i.e. $\mathcal{D} \subseteq \mathbb{R}^d$. A cluster is a subset of those feature vectors exhibiting certain properties, e.g. the members of a cluster may be close to each other in the feature space compared to non-members, or – in case of correlation clustering – they may be close to a common regression line, while other points are not. Generally, clustering algorithms as those reviewed above can provide (implicitely or explicitely) a description of the found clusters by means of a *covariance matrix* per cluster.

Formally, let $\mathcal{C}$ be a cluster, i.e. $\mathcal{C} \subseteq \mathcal{D}$, and $\bar{x}_\mathcal{C}$ denote the centroid (mean) of all points $x \in \mathcal{C}$. The *covariance matrix* $\mathbf{\Sigma}_\mathcal{C}$ of $\mathcal{C}$ is defined as:

$$\mathbf{\Sigma}_\mathcal{C} = \frac{1}{|\mathcal{C}|} \cdot \sum_{x \in \mathcal{C}} (x - \bar{x}_\mathcal{C}) \cdot (x - \bar{x}_\mathcal{C})^\mathbf{T}$$

In general, the covariance matrix describes a distribution of attributes. EM-like algorithms utilize such a description of a distribution of attributes to derive a Gaussian model that may have created the observed data. In case of correlation clusters, however, a far more adequate description may be possible. Indeed, the fact, that correlations between features have been found, even disqualifies the covariance matrix as an adequate model of a correlation cluster, since it is sort of a probabilistic model of scatter around a certain mean value. Strong correlations as in correlation clusters, on the other hand, do suggest not only probabilistic scatter, but linear dependencies, and (by a higher perspective of interpretation) perhaps even functional or causal relations. Thus, we will now consider the intrinsic properties of correlation clusters, and how to make use of them in order to derive a more appropriate model covering dependencies quantitatively.

3.1 Correlation Clusters

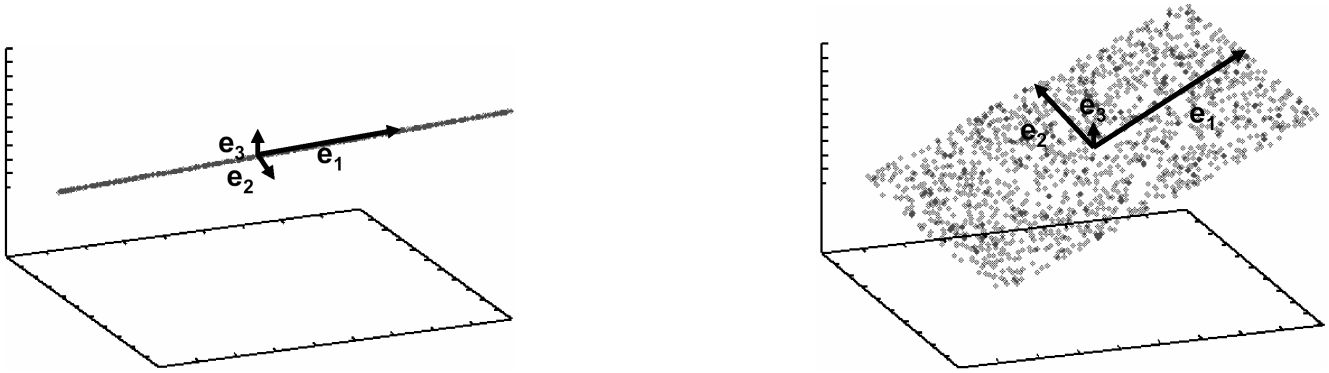
Consider a correlation cluster $\mathcal{C}$ that is derived using any algorithm capable of finding correlation clusters. Since the covariance matrix $\mathbf{\Sigma}_\mathcal{C}$ of $\mathcal{C}$ is a square matrix, it can be decomposed into the *Eigenvalue matrix* $\mathbf{E}_\mathcal{C}$ of $\mathbf{\Sigma}_\mathcal{C}$ and the *Eigenvector matrix* $\mathbf{V}_\mathcal{C}$ of $\mathbf{\Sigma}_\mathcal{C}$ such that

$$\mathbf{\Sigma}_\mathcal{C} = \mathbf{V}_\mathcal{C} \cdot \mathbf{E}_\mathcal{C} \cdot \mathbf{V}_\mathcal{C}^\mathbf{T}$$

The Eigenvalue matrix $\mathbf{E}_\mathcal{C}$ is a diagonal matrix holding the Eigenvalues of $\mathbf{\Sigma}_\mathcal{C}$ in decreasing order in its diagonal elements. The Eigenvector matrix $\mathbf{V}_\mathcal{C}$ is an orthonormal matrix with the corresponding Eigenvectors of $\mathbf{\Sigma}_\mathcal{C}$.

Now we define the correlation dimensionality of $\mathcal{C}$ as the number of dimensions of the (arbitrarily oriented) subspace which is spanned by the major axes in $\mathbf{V}_\mathcal{C}$. Let us note, that the correlation dimensionality is closely related to the intrinsic dimensionality of the data distribution. If, for instance, the points in $\mathcal{C}$ are located near a common line, the correlation dimensionality of these points will be 1. This means that we have to determine the principal components (Eigenvectors) of the points in $\mathcal{C}$. The Eigenvector associated with the largest Eigenvalue has the same direction as the first principal component, the Eigenvector associated with the second largest Eigenvalue determines the direction of the second principal component and so on. The sum of the Eigenvalues equals the trace of the square matrix $\mathbf{\Sigma}_\mathcal{C}$ which is the total variance of the points in $\mathcal{C}$. Thus, the obtained Eigenvalues are equal to the variance explained by each of the principal components, in decreasing order of importance. The correlation dimensionality of a set of points $\mathcal{C}$ is now defined as the smallest number of Eigenvectors explaining a portion of at least α of the total variance of $\mathcal{C}$. These ideas are illustrated in Figure 1. Figure 1(a) shows a correlation cluster of correlation dimensionality 1 corresponding to a (perfect) line. Only one Eigenvector (e_1) explains the total variance of $\mathcal{C}$. Figure 1(b) shows a correlation cluster of correlation dimensionality 2 that corresponds to a (perfect) plane. Here, two Eigenvectors explain the total variance of $\mathcal{C}$. Let us note that in the displayed examples, the correlations are perfect, i.e. there is no deviation from the hyperplane but all points within the set perfectly fit to the hyperplane. However, in real-world data sets, this is a quite unrealistic scenario. A threshold α may account for that fuzziness to define an adequate dimensionality of the correlation hyperplane. We call the dimensionality of a hyperplane neglecting a certain amount of deviation in orthogonal direction *correlation dimensionality*. The correlation dimensionality is defined more formally in the following.

DEFINITION 1 (CORRELATION DIMENSIONALITY). *Let $\alpha \in]0, 1[$. Then the* correlation dimensionality $\lambda_\mathcal{C}$ *of a set of points $\mathcal{C}$ is the smallest number r of Eigenvalues e_i in*



(a) 1-dimensional correlation cluster

(b) 2-dimensional correlation cluster

Figure 1: Correlation dimensionality of correlation clusters.

the $d \times d$ Eigenvalue matrix $\mathbf{E}_\mathcal{C}$ explaining a portion of at least α of the total variance:

$$\lambda_\mathcal{C} = \min_{r \in \{1,\ldots,d\}} \left\{ r \;\middle|\; \frac{\sum_{i=1}^{r} e_i}{\sum_{i=1}^{d} e_i} \geq \alpha \right\}$$

Typically, values for α are choosen between 0.8 and 0.9. For example, $\alpha = 0.85$ denotes that the obtained principal components explain 85% of the total variance. In the following, we denote the $\lambda_\mathcal{C}$-dimensional affine space which is spanned by the major axes of $\mathcal{C}$, i.e. by the $\lambda_\mathcal{C}$ first Eigenvectors of $\mathcal{C}$ and translated by, e.g. the mean vector $\bar{x}_\mathcal{C}$, the *correlation hyperplane* of $\mathcal{C}$.

Thus, the correlation dimensionality $\lambda_\mathcal{C}$ is the dimensionality of the affine space containing all points of the set $\mathcal{C}$ allowing a small deviation corresponding to the remaining portion of variance of $1 - \alpha$. The remaining, neglected variance scatters along the Eigenvectors $e_{\lambda_\mathcal{C}+1}, \ldots, e_d$.

We therefore distinguish between two disjoint sets of Eigenvectors:

Definition 2 (strong and weak Eigenvectors). *We call the first $\lambda_\mathcal{C}$ Eigenvectors of $\mathbf{V}_\mathcal{C}$* strong Eigenvectors. *The strong Eigenvectors of $\mathbf{V}_\mathcal{C}$ are denoted by $\check{\mathbf{V}}_\mathcal{C}$. The remaining Eigenvectors are called* weak Eigenvectors. *We denote the weak Eigenvectors by $\hat{\mathbf{V}}_\mathcal{C}$.*

For an illustration see again Figure 1: In the correlation cluster of correlation dimensionality 1 (Figure 1(a)) e_1 is a *strong Eigenvector* whereas e_2 and e_3 are *weak Eigenvectors*. In the correlation cluster of correlation dimensionality 2 (Figure 1(b)) e_1 and e_2 are *strong Eigenvectors* whereas e_3 is a *weak Eigenvector*. The Eigenvectors are overexemplified in this example. Suppose they were scaled by their corresponding Eigenvalues. If no variance remains along an Eigenvector, as it may e.g. appear for e_2 and e_3 in Figure 1(a), this Eigenvector will disappear since the corresponding Eigenvalue becomes zero.

While the correlation hyperplane is spanned by the *strong* Eigenvectors, it is equally well defined by the *weak* Eigenvectors that are orthogonal to this hyperplane in $\mathbb{R}^d$. Furthermore, describing the correlation cluster by means of the weak Eigenvectors (instead of the strong Eigenvectors) directly yields an equality system that defines not only the corresponding hyperplane, but also allows to directly inspect the underlying dependencies among attributes numerically, as we will show in more detail subsequently.

3.2 Deriving Quantitative Models for Correlation Clusters

Let $\mathcal{C}$ be a λ-dimensional correlation cluster in $\mathcal{D}$ ($\mathcal{C} \subseteq \mathcal{D}$). Thus, there are λ strong Eigenvectors and $d - \lambda$ weak Eigenvectors in the describing matrix of Eigenvectors derived by PCA on the points of cluster $\mathcal{C}$. A λ-dimensional hyperplane defining the correlation cluster $\mathcal{C}$ is therefore completely defined by the mean point (centroid) $\bar{x}_\mathcal{C} = (\bar{x}_1 \cdots \bar{x}_d)^T$ of all points belonging to cluster $\mathcal{C}$ and the set of weak Eigenvectors, $\hat{\mathbf{V}}_\mathcal{C}$, that are normal vectors to the hyperplane. Then we can derive the following equation system to describe the hyperplane, consisting of $d - \lambda$ equations:

$$\begin{aligned}
v_{(\lambda+1),1}(x_1 - \bar{x}_1) + v_{(\lambda+1),2}(x_2 - \bar{x}_2) + \cdots + v_{(\lambda+1),d}(x_d - \bar{x}_d) &= 0 \\
v_{(\lambda+2),1}(x_1 - \bar{x}_1) + v_{(\lambda+2),2}(x_2 - \bar{x}_2) + \cdots + v_{(\lambda+2),d}(x_d - \bar{x}_d) &= 0 \\
&\vdots \\
v_{d,1}(x_1 - \bar{x}_1) + v_{d,2}(x_2 - \bar{x}_2) + \cdots + v_{d,d}(x_d - \bar{x}_d) &= 0
\end{aligned}$$

where $v_{i,j}$ is the value at column i, row j in the Eigenvector matrix $\mathbf{V}_\mathcal{C}$ of $\mathcal{C}$. As we have pointed out, only the weak Eigenvectors are relevant. Thus we can equivalently denote this equation system by

$$\hat{\mathbf{V}}_\mathcal{C}^\mathbf{T} \cdot x = \hat{\mathbf{V}}_\mathcal{C}^\mathbf{T} \cdot \bar{x}_\mathcal{C}.$$

The defect of $\hat{\mathbf{V}}_\mathcal{C}^\mathbf{T}$ gives the number of free attributes, the other attributes may actually be involved in linear dependencies. Basically, these dependencies are revealed by transforming the equation system using Gauss-Jordan elimination. The thus derived reduced row echelon form of the matrix is known to be unique [25]. The unique form does, of course, not provide new information, but it is easily comparable to alternative solutions and conveniently interpretable by inspecting experts. To enhance numerical stability, we suppose to use total pivoting for the Gauss-Jordan elimination.

By construction, the equation system is – at least approximately – fulfilled for all points $x \in \mathcal{C}$. But, furthermore, it suggests a quantitative model for the cluster. This model could be evaluated using retained data points. Besides, as

we will see below, it may also serve as a predictive model to classify new data points.

In summary, we propose the following general method to derive quantitative models of clusters in a dataset of feature vectors $\mathcal{D} \subset \mathbb{R}^d$:

1. Run a clustering algorithm on $\mathcal{D}$ that is able to find correlation clusters, i.e. use e.g. 4C or ORCLUS. However, also k-means or DBSCAN is possible, provided that a proper distance function taking into account the correlation dimension is used. If the result may be restricted to clusters of *positively* correlated features, even the usage of any general biclustering or pattern-based clustering algorithm will be possible. The decision for a specific clustering algorithm will also determine whether or not a data object may belong to several clusters simultaneously. In our experiments we use COPAC [1], a new correlation clustering algorithm that is shown to improve over 4C as well as ORCLUS w.r.t. efficiency, effectivity, and robustness.
2. For each correlation cluster $\mathcal{C}_i \subset \mathcal{D}$ found in the previous step:
   (a) Derive the covariance matrix $\mathbf{\Sigma}_{\mathcal{C}_i}$.
   (b) Select the weak Eigenvectors $\hat{\mathbf{V}}_{\mathcal{C}_i}$ of $\mathbf{\Sigma}_{\mathcal{C}_i}$ with respect to a certain α.
   (c) Derive the equation system describing the correlation hyperplane:
   $$\hat{\mathbf{V}}_{\mathcal{C}_i}^{\mathbf{T}} \cdot x = \hat{\mathbf{V}}_{\mathcal{C}_i}^{\mathbf{T}} \cdot \bar{x}_{\mathcal{C}_i}$$
   (d) Apply Gauss-Jordan elimination to the derived equation system to obtain a unique description of quantitative dependencies by means of the reduced row echelon form of the equation system.

3.3 Interpretation of Correlation Cluster Models

Suppose by applying this method we obtain the following solution describing a cluster in a 5-dimensional feature space $\mathbb{R}^5$:

$$\begin{aligned} 1x_1 + 0x_2 + c_1x_3 + 0x_4 + e_1x_5 &= f_1 \\ 0x_1 + 1x_2 + c_2x_3 + 0x_4 + e_2x_5 &= f_2 \\ 0x_1 + 0x_2 + 0x_3 + 1x_4 + e_3x_5 &= f_3 \end{aligned}$$

This would provide a quantitative model describing a correlation cluster of correlation dimensionality 2 (corresponding to the number of free attributes, or, equivalently, the number of *strong* Eigenvectors) where we have linear dependencies among

- x_1, x_3, and x_5
- x_2, x_3, and x_5
- x_4 and x_5

by given factors c_1, e_1, c_2, e_2, and e_3.

Note that we must not draw any conclusions concerning causalities between attributes. But relations between certain attributes are quantitatively and uniquely defined. To resolve these relations to any formula that suggests a causality we have to rely on the domain knowledge of experts. However, we believe that uncovered quantitative relationships will lead to refined experiments and help to finally explore supposable causalities. Thus, we could choose experimental settings involving either

- x_4 and x_5, or
- x_2, x_3, and x_5, or
- x_1, x_3, and x_5,

and changing the quantities in relation to each other. The dependencies revealed in the original experiment could have been interpreted such as fall or rise of an arbitrary subset of $S \subset \{x_1, x_3, x_5\}$ caused fall or rise of the remaining subset $\{x_1, x_3, x_5\} \setminus S$. Further experiments could refine the model by excluding certain combinations of causal models. Of course, the three variables, x_1, x_3, and x_5, may also simply be connected by a fourth variable, that has not been monitored so far. Thus, trivially, a quantitative connection will never guarantee a direct causal relationship. Furthermore, in many domains, one-way causal relationships provide only one part of the whole picture, since systems often are regulated by negative-feedback-loops, that make causalities circular. Nevertheless, modelling parts of a complex system remains useful even under restrictive constraints (as shown e.g. for genetic regulatory interaction networks, cf. [13]).

3.4 Sample Application: Predictive Models

Having derived a descriptive model, it can be refined by determining an average distance of the cluster members from the correlation hyperplane. Such deviations are typically to be expected in natural systems. At least, one has to account for errors in measurement. The distance of a point to a hyperplane is thereby naturally defined as the Euclidean distance to its perpendicular projection onto the hyperplane, i.e.:

$$d(x, \mathcal{C}) = ||x - \bar{x}_{\mathcal{C}} - \text{proj}_{\mathcal{C} - \bar{x}_{\mathcal{C}}}(x - \bar{x}_{\mathcal{C}})||,$$

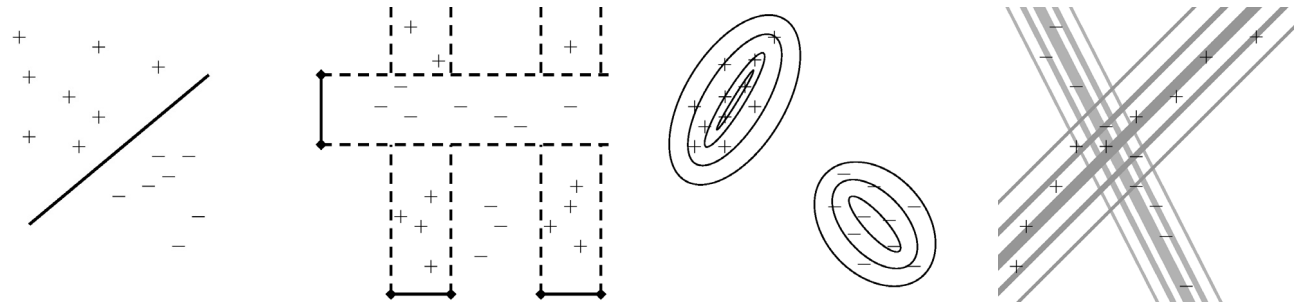
where $\mathcal{C}$ denotes the idealized hyperplane of a correlation cluster. By definition, the hyperplane $\mathcal{C}$ is an affine space, that is a subspace translated by $\bar{x}_{\mathcal{C}}$, the mean vector of all points of the cluster corresponding to $\mathcal{C}$. $\text{proj}_{\mathcal{S}} : \mathbb{R}^n \to \mathbb{R}^n$ denotes the perpendicular projection of a vector to an arbitrary subspace $\mathcal{S}$ of $\mathbb{R}^n$. If $\mathcal{S}$ is given by an orthonormal basis, e.g. the set of strong Eigenvectors derived for the corresponding correlation cluster, $\{s_1, \cdots, s_{\lambda_{\mathcal{S}}}\}$, then

$$\text{proj}_{\mathcal{S}}(x) = \langle x, s_1 \rangle s_1 + \langle x, s_2 \rangle s_2 + \cdots + \langle x, s_{\lambda_{\mathcal{S}}} \rangle s_{\lambda_{\mathcal{S}}}.$$

Assuming the deviations fit to a Gaussian distribution with $\mu = 0$, the standard deviation σ of the distances of all cluster members suffices to define a Gaussian model of deviations from the common correlation hyperplane. For each of the derived models, the probability is given for a new data object to be generated by this specific Gaussian distribution. A set of models for a set of correlation clusters can therefore provide a convenient instrument for classification in the perspective of different linear dependencies among the data. The probability that an object x was generated by the jth of n Gaussian distributions, C_j, is given by

$$P(\mathcal{C}_j|x) = \frac{\frac{1}{\sigma_j\sqrt{2\pi}} e^{-\frac{1}{2\sigma_j^2}\left(d\left(x,\mathcal{C}_j\right)\right)^2}}{\sum_{i=1}^{n} \frac{1}{\sigma_i\sqrt{2\pi}} e^{-\frac{1}{2\sigma_i^2}(d(x,\mathcal{C}_i))^2}}.$$

Compared to many traditional classification algorithms, like SVM or kNN, our predictive models do not only provide a separating boundary between classes (cf. Figure 2(a)), but



(a) Linear decision boundaries

(b) Axis parallel decision rules

(c) Density functions

(d) Deviations from hyperplanes

Figure 2: Decision models of different types of classifiers

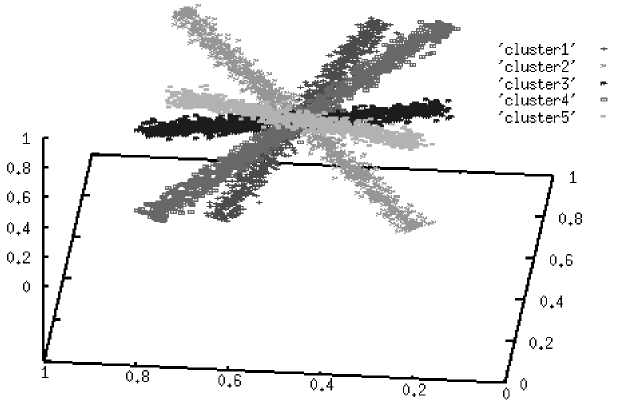


Figure 3: Synthetic data set DS1.

also give a meaningful definition of the class. So do other classifiers, like decision trees or rule based learners, but their descriptions usually are limited to (at least in sections) axis parallel decision boundaries (cf. Figure 2(b)). The models provided by the EM algorithm or other Bayesian learners differ from our models in that they simply define a scattering around a mean point, using a quadratic form distance function or a density function for a certain probability distribution (cf. Figure 2(c)). For underlying linear dependencies, a quadratic distance function will resemble our models only if the dependencies are perfectly expressed in the data without any aberrations. Accounting for some variance perpendicular to a hyperplane, while the hyperplane represents a linear dependency among several attributes, is a novel approach among the family of classification algorithms (cf. Figure 2(d)).

4. EVALUATION

In our experiments we use the correlation clustering algorithm COPAC [1] to generate the correlation clusters in a preprocessing step to our method. We choose this algorithm due to its efficiency, effectivity, and robustness. In each case, parameters for clustering were chosen according to [1]. Let us again note that any other (correlation) clustering algorithm is applicable for preprocessing.

4.1 Synthetic data sets

For our experiments we used several synthetic data sets containing correlation clusters in the unit cube of $\mathbb{R}^d$ that have been generated by a generic data generator. The generated correlation clusters form a λ-dimensional hyperplane which is specified by an equation system of $d - \lambda$ equations. The distances of the points to the hyperplane are normally distributed with a specified standard deviation and a mean of zero.

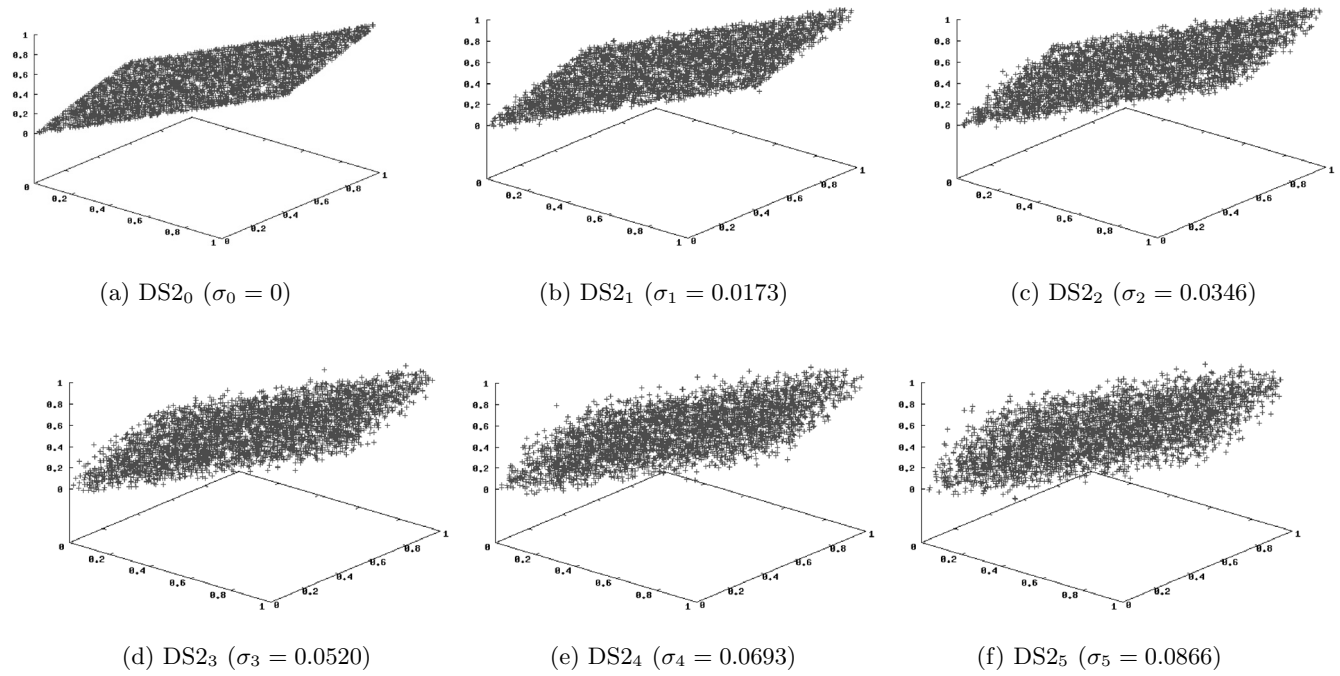
The first data set DS1 consists of five correlation clusters, each forming a line of 1,000 points in $\mathbb{R}^3$ (cf. Figure 4). In each cluster, the distances of the points to the correlation lines are normally distributed with a standard deviation of about 1.5% of the maximum distance in the unit cube. The purpose of this data set is to demonstrate the capability of our proposed method to obtain a quantitative model for the correlation clusters. As it can be seen in Table 1 we derived a good approximation of the equation systems that define the models for the correlation clusters despite the obviously strong jitter in the data set.

In the second experiment we evaluated our method on data sets with varying standard deviation. We generated six data sets ($DS2_0$, ..., $DS2_5$) forming a 2-dimensional hyperplane in $\mathbb{R}^3$ with different values for the standard deviation of the distances. The values for the standard deviation were set to $\sigma_0 = 0\%$ up to $\sigma_5 = 5\%$ of the maximum distance in the unit cube (cf. Figure 4). The results are shown in Table 2. As expected, with increasing standard deviation of the distances, the detected correlation models suffer from a slight blurring, i.e. the coefficients of the models slightly deviate from the exact coefficients. However, the general correlations are still detected and also the hidden quantitative relationships are still uncovered rather clear even if the points stronger deviate from the optimal hyperplane. In general, our proposed method has proven to be rather robust w.r.t. small jitter.

In addition to the reported experiments on 3-dimensional data, we performed several experiments on higher dimensional data. In all experiments, we achieved results of similar high quality, i.e. all linear dependencies hidden in the data were correctly uncovered. Due to space limitations and clearness reasons, we omit a detailed presentation of these results.

Table 1: Dependencies on DS1 data.

	Generated dependencies	Generated standard deviation	Found dependencies
cluster 1	$x1 - x3 = 0$ $x2 + 0.5x3 = 0.75$	$\sigma = 0.0246$	$x1 - 1.0069x3 = -0.0035$ $x2 + 0.5065x3 = 0.7537$
cluster 2	$x1 - x3 = 0$ $x2 - x3 = 0$	$\sigma = 0.0243$	$x1 - 1.0027x3 = -0.0028$ $x2 - 0.9901x3 = 0.0022$
cluster 3	$x1 + x3 = 1$ $x2 - x3 = 0$	$\sigma = 0.0238$	$x1 + 1.0008x3 = 1.0005$ $x2 - 1.0011x3 = 0.0000$
cluster 4	$x1 - x3 = 0$ $x2 + x3 = 1$	$\sigma = 0.0246$	$x1 - 1.0009x3 = 0.0000$ $x2 + 0.9999x3 = 0.9995$
cluster 5	$x1 + x3 = 1$ $x2 + x3 = 1$	$\sigma = 0.0249$	$x1 + 0.9975x3 = 0.9988$ $x2 + 0.9968x3 = 0.9992$



(a) $DS2_0$ ($\sigma_0 = 0$) (b) $DS2_1$ ($\sigma_1 = 0.0173$) (c) $DS2_2$ ($\sigma_2 = 0.0346$)

(d) $DS2_3$ ($\sigma_3 = 0.0520$) (e) $DS2_4$ ($\sigma_4 = 0.0693$) (f) $DS2_5$ ($\sigma_5 = 0.0866$)

Figure 4: Synthetic data sets with different values for standard deviation.

4.2 Real world data sets

Wages data. The Wages data set[1] consists of 534 11-dimensional observations from the 1985 Current Population Survey. Since most of the attributes are not numeric, we used only 4 dimensions (A=age, YE=years of education, YW=years of work experience, and W=wage) for correlation analysis.

COPAC detected three correlation clusters in this data set. The resulting dependencies of these clusters are summarized in Table 3. The first cluster consists only of people having 12 years of education, whereas the second cluster consists only of people having 16 years of education. Furthermore, in both of these clusters the difference between age and work experience is a specific constant, namely years of education plus 6, which makes perfectly sense. Additionally, for the first cluster, we found a dependency between wage and age: the wage equals a constant plus a small factor times the age of an employee, i.e., the older an employee, the more he earns. This relationship is independent from the attribute work experience. Note that years of education is a constant where this condition holds. In the third cluster only those employees are grouped which started school in the age of 6 years and after graduation immediately began working. Thus, the sum of years of education and work experience equals the age minus 6.

[1] `http://lib.stat.cmu.edu/datasets/CPS_85_Wages`

Table 2: Dependencies on DS2 data.

	Generated dependencies	standard deviation	Found dependencies
$DS2_0$	$x1 - 0.5x2 - 0.5x3 = 0$	$\sigma = 0$	$x1 - 0.5000x2 - 0.5000x3 = 0.0000$
$DS2_1$	$x1 - 0.5x2 - 0.5x3 = 0$	$\sigma = 0.0173$	$x1 - 0.4989x2 - 0.5002x3 = 0.0000$
$DS2_2$	$x1 - 0.5x2 - 0.5x3 = 0$	$\sigma = 0.0346$	$x1 - 0.5017x2 - 0.4951x3 = 0.0016$
$DS2_3$	$x1 - 0.5x2 - 0.5x3 = 0$	$\sigma = 0.0520$	$x1 - 0.5030x2 - 0.5047x3 = -0.0059$
$DS2_4$	$x1 - 0.5x2 - 0.5x3 = 0$	$\sigma = 0.0693$	$x1 - 0,4962x2 - 0.5106x3 = -0.0040$
$DS2_5$	$x1 - 0.5x2 - 0.5x3 = 0$	$\sigma = 0.0866$	$x1 - 0.4980x2 - 0.4956x3 = 0.0064$

Table 3: Dependencies on Wages data.

cID	derived dependencies
1	$YE = 12$ $YW - 1 \cdot A = -18$ $W - 0.07 \cdot A = 5.14$
2	$YE = 16$ $YW - 1 \cdot A = -22$
3	$YE + 1 \cdot YW - 1 \cdot A = -6$

Gene expression data. This data set was derived from an experimental study of apoptosis in human tumor cells[2]. Apoptosis is a genetically controlled pathway of cell death. The data set contains the expression level of 4610 genes at five different time slots (5, 10, 15, 30, 60 minutes) after initiating the apoptosis pathway.

We analyzed two correlation clusters detected by COPAC. The derived dependencies of these clusters are depicted in Table 4. The attributes are abbreviated by Mi, where i denotes the time slot of this attribute, e.g. $M5$ denotes time slot "5 minutes". The first cluster contains several genes that are located at the mitochondrial membrane. The first four time slots exhibit a negative linear relationship with $M60$. Similar observations can be made for the second cluster that contains several genes that are related to the tumor necrosis factor (RNF). The uncovered dependencies suggest that the activity of the corresponding genes decrease with proceeding cell death. The strong negative correlations among genes related to mitochondria (cluster 1) indicates that the volume of the energy metabolism (which is located in mitochondria) is decreasing over time. In addition, the correlation among the genes related to RNF makes sense since the dying cells are tumor cells.

Breast cancer data. We also applied our method to four correlation clusters found in the Wisconsin Breast Cancer data derived from UCI ML Archive[3]. This data set measures nine biomedical parameters characterizing breast cancer type in 683 humans (humans with missing values were removed from the data set). The parameters include Clump Thickness (attribute "A1"), Uniformity of Cell Size ("A2"), Uniformity of Cell Shape ("A3"), Marginal Adhesion ("A4"), Single Epithelial Cell Size ("A5"), Bare Nuclei ("A6"), Bland Chromatin ("A7"), Normal Nucleoli ("A8"), and Mitoses ("A9").

The derived dependencies of the four clusters are depicted in Table 5. Let us note that each cluster only contains humans suffering from a benign tumor type. The patients suffering from a malignant tumor type were classified as noise. The dependencies in the first cluster are quite clean and indicate a constant behaviour of seven attributes. In addition, $A5$ is related to $A7$. The models of the remaining clusters are quite complex. Mostly, the first attributes which measure an aggregated information about the shape and the size of the tumor cells exhibit a relationship to more specific measurements on single parts of the tumor. In general, since the clusters only contain benign tumors, our results indicate that this mostly harmless tumor type can still be explained and modelled by linear relationships among the measurements, whereas the more dangerous tumor type cannot be explained or modelled through any linear relations among the measurements.

4.3 Applying Quantitative Models to Class Prediction

Last but not least, we briefly discuss a further potential application of our method that utilizes the derived models for subsequent data analysis. As sketched above, the quantitative models generated by our method can e.g. be used to predict the class of a new object. To evaluate this potential, we used three 2-dimensional synthetic data sets each with 5 classes. The first data set ("$DS3_0$") contains 50 points per class, the second and the third data sets ("$DS3_1$" and "$DS3_2$") each contain 100 points per class. Each class is generated according to a certain linear dependency. The class distributions in $DS3_0$ and $DS3_1$ exhibit a jitter of 0.5% of the maximum distance in the unit cube, whereas the jitter of the classes in $DS3_2$ is 0.75%. The third data set is depicted in Figure 5. Note that these data sets are rather artificial and are only applied for a proof of principle.

We compared the classification accuracy of our sketched classifier to several other standard learning approaches. For this comparison we used the WEKA framework [23] with standard parameter settings, in particular, kNN (IBk) with $k = 1$ (best results reported), SVM (SMO), rule-based learner (PART), Naive Bayes, decision tree (J48), and multinomial

[2]The data are donated by our project partners.
[3]`http://www.ics.uci.edu/~mlearn/MLSummary.html`

Table 4: Dependencies on Gene expression data.

cID	derived dependencies	sample gene names
1	$M5 - 1.05 \cdot M60 = -0.12$ $M10 - M60 = -0.17$ $M15 - M60 = 0$ $M30 - 1.1 \cdot M60 = 0.11$	NDUFB10, MTRF1, TIMM17A, TOM34, CPS1, NM44, COX10, FIBP, TRAP1, MTERF, ME2, HK1, HADHA, ASAH2, CPS1, CA5A, BNI3PL
2	$M5 - 0.98 \cdot M60 = 0$ $M10 - 0.98 \cdot M60 = 0$ $M15 - 0.97 \cdot M60 = 0$ $M30 - 0.97 \cdot M60 = 0$	TNFRSF6, TNFRSF11A, TNFRSF7, TNFRSF1B, TNFRSF10B,TNFRSF5, TNFRSF1A, TRAF5, TRAF2, TNFSF12

Table 5: Dependencies on Wisconsin breast cancer data.

cID	derived dependencies
1	$A1 = 2$ and $A2 = 1$ and $A3 = 1$ and $A4 = 1$ and $A6 = 1$ and $A5 - 0.1 \cdot A7 = 1.9$ $A8 = 1$ and $A9 = 1$
2	$A1 - 0.4 \cdot A4 + 0.7 \cdot A5 - 0.2 \cdot A6 + 0.9 \cdot A7 - 24 \cdot A8 = -20.9$ $A2 + 0.03 \cdot A4 - 0.05 \cdot A5 + 0.02 \cdot A6 + 0.02 \cdot A7 - 0.3 \cdot A8 = 0.8$ $A3 + 0.2 \cdot A4 + 0.1 \cdot A5 + 0.1 \cdot A6 + 0.2 \cdot A7 - 1.8 \cdot A8 = 0.3$
3	$A1 + 82.2 \cdot A6 + 7.8 \cdot A7 - 42 \cdot A8 - 18.5 \cdot A9 = 38.5$ $A2 - 1.9 \cdot A6 - 0.2 \cdot A7 + 0.9 \cdot A8 + 1.8 \cdot A9 = 1.5$ $A3 - 60.1 \cdot A6 - 6.5 \cdot A7 + 25.1 \cdot A8 + 141 \cdot A9 = 97.5$ $A4 - 7.2 \cdot A6 - 0.4 \cdot A7 - 1.1 \cdot A8 + 15.6 \cdot A9 = 7.6$ $A5 - 18.8 \cdot A6 - 1.4 \cdot A7 - 0.5 \cdot A8 + 45.9 \cdot A9 = 26.1$
4	$A1 - 5.4 \cdot A5 + 1.6 \cdot A6 - 0.1 \cdot A7 + 1 \cdot A8 - 16.3 \cdot A9 = -21.1$ $A2 + 1.7 \cdot A5 - 0.6 \cdot A6 + 0.2 \cdot A7 - 0.7 \cdot A8 - 9.9 \cdot A9 = -6.5$ $A3 - 1.8 \cdot A5 - 0.8 \cdot A6 - 0.3 \cdot A7 - 0.7 \cdot A8 - 11.9 \cdot A9 = -8.5$ $A4 - 2.3 \cdot A5 - 0.2 \cdot A6 + 0.2 \cdot A7 + 0.4 \cdot A8 + 8.6 \cdot A9 = 6.5$

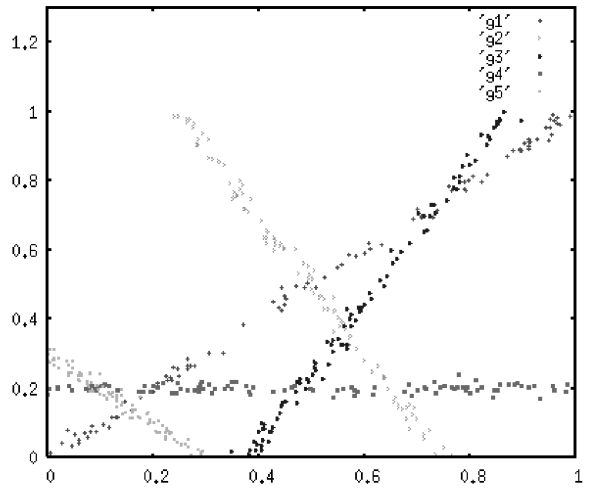


Figure 5: Data set $DS3_2$.

logistic regression (Logistic). The results are depicted in Table 6. As it can be seen, our approach significantly outperforms most of the other approaches, except kNN, in terms of accuracy.

Let us note that standard classifiers will most likely produce comparative or even better results if the classes are generated through models that cannot be captured by our concepts of linear dependencies. However, our small example may show that if the classes are generated by a model of linear dependencies as captured by our proposed concepts, our method obviously yields a better prediction accuracy than standard supervised learners.

Table 6: Comparison of different classifiers in terms of accuracy (in %).

	Our method	IBk	SMO	PART	NB	J48	Log.
$DS3_0$	95	91	62	82	65	82	67
$DS3_1$	94	94	54	85	64	83	60
$DS3_2$	91	91	58	81	60	83	57

5. CONCLUSIONS

Several correlation clustering algorithms have been proposed recently. However, none of these algorithms derive a quantitative model for each correlation cluster which is urgently needed in order to gain the full practical potentials from correlation cluster analysis. In this paper, we describe an original approach to derive quantitative information on the linear dependencies within correlation clusters. Our concepts are independent of the clustering model and can thus

be applied as a post-processing step to any correlation clustering algorithm. Furthermore, as a sample application of our approach, we sketched how these quantitative models can be used to predict the probability distribution that an object is created by these models. Our broad experimental evaluation demonstrates the beneficial impact of our method on several applications of significant practical importance. We exemplified how our method can be used in conjunction with a suitable clustering algorithm to gain valuable and important knowledge about complex relationships in real-world data.

6. REFERENCES

[1] E. Achtert, C. Böhm, H.-P. Kriegel, P. Kröger, and A. Zimek. Robust, complete, and efficient correlation clustering. submitted.

[2] C. C. Aggarwal, C. M. Procopiuc, J. L. Wolf, P. S. Yu, and J. S. Park. Fast algorithms for projected clustering. In *Proceedings of the SIGMOD Conference, Philadelphia, PA*, 1999.

[3] C. C. Aggarwal and P. S. Yu. Finding generalized projected clusters in high dimensional space. In *Proceedings of the SIGMOD Conference, Dallas, TX*, 2000.

[4] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proceedings of the SIGMOD Conference, Seattle, WA*, 1998.

[5] C. Böhm, K. Kailing, H.-P. Kriegel, and P. Kröger. Density connected clustering with local subspace preferences. In *Proceedings of the 4th International Conference on Data Mining (ICDM), Brighton, U.K.*, 2004.

[6] C. Böhm, K. Kailing, P. Kröger, and A. Zimek. Computing clusters of correlation connected objects. In *Proceedings of the SIGMOD Conference, Paris, France*, 2004.

[7] Y. Cheng and G. M. Church. Biclustering of expression data. In *Proceedings of the 8th International Conference Intelligent Systems for Molecular Biology (ISMB), San Diego, CA*, 2000.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–31, 1977.

[9] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd ACM International Conference on Knowledge Discovery and Data Mining (KDD), Portland, OR*, 1996.

[10] E. Georgii, L. Richter, U. Rückert, and S. Kramer. Analyzing microarray data using quantitative association rules. *Bioinformatics*, 21(Suppl. 2):ii1–ii8, 2005.

[11] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Academic Press, 2001.

[12] J. A. Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, 67(337):123–129, 1972.

[13] D. Husmeier. Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics*, 19(17):2271–2282, 2003.

[14] K. Kailing, H.-P. Kriegel, and P. Kröger. Density-connected subspace clustering for high-dimensional data. In *Proceedings of the 4th SIAM International Conference on Data Mining (SDM), Orlando, FL*, 2004.

[15] J. Liu and W. Wang. OP-Cluster: Clustering by tendency in high dimensional spaces. In *Proceedings of the 3th International Conference on Data Mining (ICDM), Melbourne, FL*, 2003.

[16] J. Pei, X. Zhang, M. Cho, H. Wang, and P. S. Yu. MaPle: A fast algorithm for maximal pattern-based clustering. In *Proceedings of the 3th International Conference on Data Mining (ICDM), Melbourne, FL*, 2003.

[17] C. M. Procopiuc, M. Jones, P. K. Agarwal, and T. M. Murali. A Monte Carlo algorithm for fast projective clustering. In *Proceedings of the SIGMOD Conference, Madison, WI*, 2002.

[18] U. Rückert, L. Richter, and S. Kramer. Quantitative association rules based on half-spaces: an optimization approach. In *Proceedings of the 4th International Conference on Data Mining (ICDM), Brighton, U.K.*, pages 507–510, 2004.

[19] R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. In *Proceedings of the SIGMOD Conference, Montreal, Canada*, pages 1–12, 1996.

[20] A. K. H. Tung, X. Xu, and C. B. Ooi. CURLER: Finding and visualizing nonlinear correlated clusters. In *Proceedings of the SIGMOD Conference, Baltimore, ML*, 2005.

[21] H. Wang, W. Wang, J. Yang, and P. S. Yu. Clustering by pattern similarity in large data sets. In *Proceedings of the SIGMOD Conference, Madison, WI*, 2002.

[22] G. I. Webb. Discovering associations with numeric variables. In *Proceedings of the 7th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), San Francisco, CA*, pages 383–388, 2001.

[23] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2nd edition, 2005.

[24] J. Yang, W. Wang, H. Wang, and P. S. Yu. Delta-Clusters: Capturing subspace correlation in a large data set. In *Proceedings of the 18th International Conference on Data Engineering (ICDE), San Jose, CA*, 2002.

[25] T. Yuster. The reduced row echelon form of a matrix is unique: A simple proof. *Mathematics Magazine*, 57(2):93–94, 1984.