Data Min Knowl Disc (2010) 21:259–276
DOI 10.1007/s10618-010-0187-5

Maximal exceptions with minimal descriptions

Matthijs van Leeuwen

Received: 30 April 2010 / Accepted: 20 June 2010 / Published online: 23 July 2010

Abstract We introduce a new approach to Exceptional Model Mining. Our algorithm, called EMDM, is an iterative method that alternates between Exception Maximisation and Description Minimisation. As a result, it finds maximally exceptional models with minimal descriptions. Exceptional Model Mining was recently introduced by Leman et al. (Exceptional model mining 1–16, 2008) as a generalisation of Subgroup Discovery. Instead of considering a single target attribute, it allows for multiple 'model' attributes on which models are fitted. If the model for a subgroup is substantially different from the model for the complete database, it is regarded as an exceptional model. To measure exceptionality, we propose two information-theoretic measures. One is based on the Kullback–Leibler divergence, the other on KRIMP. We show how compression can be used for exception maximisation with these measures, and how classification can be used for description minimisation. Experiments show that our approach efficiently identifies subgroups that are both exceptional and interesting.

Keywords Exceptional Model Mining · Subgroup Discovery · Information theory

1 Introduction

Finding regions in a database where the distribution of a target variable is substantially different from its distribution in the whole database, i.e. Subgroup Discovery (Klösgen 2002), has proven to be a very useful paradigm in exploratory data mining. However, allowing only a single target variable limits its possible applications.

Responsible editors: José L Balcázar, Francesco Bonchi, Aristides Gionis, Michèle Sebag.

M. van Leeuwen (✉)
Department of Information and Computing Sciences, Universiteit Utrecht, Utrecht, The Netherlands
e-mail: mleeuwen@cs.uu.nl

Therefore, Leman et al. (2008) recently introduced Exceptional Model Mining (EMM), a generalisation of Subgroup Discovery. Instead of having a single target variable, EMM allows for a set of target variables on which complex models can be fitted. A model fitted on a subgroup is exceptional if it is substantially different from the model fitted on the entire database. All kinds of models can be used, as long as differences between models can be measured. Example quality measures have been given for correlation, regression and classification models.

In the last few years, data with multiple target variables has attracted an increasing amount of attention. Multi-label ranking and classification (Tsoumakas et al. 2010) are two prime examples of this. Applications in very different domains exist, ranging from biology to music and from images to web/text. EMM can also be applied in all these domains; primarily for exploring the data, but exceptional models may also be useful for learning tasks, e.g. to improve classification.

Consider, for example, a dataset with information about the climate and animal presence for areas all over the world. Now, we might be interested in areas for which the animal distribution is very different from the overall distribution. In this setting, the Antarctic might be an interesting subgroup: if it is always very cold, dry and windy (*subgroup description*), then Emperor Penguins and Snow Petrels breed there (*exceptional model*). Thus, the climate information would be the 'description space' and the animal information our 'model space'.

Finding such exceptional models is not an easy task though, since the search space is huge. Essentially, we would have to consider all possible subgroups, i.e. all possible subsets of a database. Using Subgroup Discovery search strategies is possible, but these exploit only the description space for searching. This is far from efficient, as one is trying to find differing distributions in model space by defining subgroups in description space.

1.1 Main contributions

To effectively find exceptional models, we propose to use a completely different search strategy which exploits structure in both description and model space. It starts with a candidate subgroup and iteratively improves it. Each iteration consists of two steps, one for Exception Maximisation (EM) and one for Description Minimisation (DM). Together, the algorithm is called EMDM and finds maximally exceptional models with minimal descriptions.

To determine whether a model is exceptional, an exceptionality measure is needed. The Kullback–Leibler (KL) divergence (Kullback and Leibler 1951) is particularly suited for this, as it is an information-theoretic measure that quantifies how different one probability distribution is from another. We introduce two information-theoretic measures, one based on KL divergence, which treats all variables as independent, and one based on MDL (Rissanen 1978) and Krimp (Siebes et al. 2006), called Krimp Gain (KG), which takes associations between variables into account.

EMDM is a generic algorithm and can be used with different model classes and different types of subgroup descriptions. We give specific instances for both Exception Maximisation and Description Minimisation. The EM-step is closely related to

the exceptionality measure, hence we base this step on information theory and maximise exceptionality using compression. For the DM-step, we use a generic rule-based classifier, RIPPER (Cohen 1995). By inducing classifiers with RIPPER, we obtain minimal subgroup descriptions.

Experiments on a diverse set of datasets show that EMDM efficiently finds exceptional models. Both proposed measures, KL and KG, are tested to see how they perform and which should be used when. We compare structured to random candidate subgroups and compare EMDM to a Subgroup Discovery beam search.

2 Exceptional model mining

We assume that a database $\mathcal{D}$ is a bag of tuples t that all have the same attributes $\{A_1^D, \ldots, A_k^D, A_1^M, \ldots, A_l^M\}$. Each attribute A_i^X has a domain of possible values V_i^X. The total set of attributes A consists of a set of k *description attributes* A^D and a set of l *model attributes* A^M. We will slightly abuse notation by using x^D resp. x^M to denote the projection of x onto its description resp. model attributes, e.g. $t^D = \pi_{A^D}(t)$ and $\mathcal{D}^M = \pi_{A^M}(\mathcal{D})$. Equivalently for individual attributes, e.g. $\mathcal{D}^{A_i^M} = \pi_{A_i^M}(\mathcal{D})$. When discussing attribute-values that are part of the projection onto either the description or model attributes, we write *description data* or *model data*. One of the most important concepts in this paper is the notion of a *subgroup*.

Definition 1 (*Subgroup*) A *subgroup* is a bag of tuples $G \subseteq \mathcal{D}$. $|G|$ denotes the size of this bag.

Definition 2 (*Subgroup description*) A *subgroup description* is an indicator function s for a subgroup, as a function of description attributes A^D. That is, it is a function $s : (V_1 \times \ldots \times V_k) \mapsto \{0, 1\}$, with V_i the domain of A_i^D, and its corresponding subgroup is $G_s = \{t \in \mathcal{D} \mid s(t^D) = 1\}$. Given the set of all subgroup descriptions that define the same subgroup and a function that quantifies description complexity, a *minimal subgroup description* is a subgroup description that minimises description complexity.

Given a subgroup G, we would like to know how 'exceptional' (or interesting) it is, looking only at G^M. For this, we need some sort of model class and a way to induce models. From these models, we measure how exceptional a subgroup G^M is with respect to a database $\mathcal{D}^M$.

Definition 3 (*Exceptionality measure*) Let $\mathcal{D}$ be a database and $\mathcal{G}^M$ the set of all possible subsets of $\mathcal{D}^M$. An *exceptionality measure* is a function $\phi^{\mathcal{D}^M} : \mathcal{G}^M \mapsto \mathbb{R}$ that assigns a numeric value to a subgroup $G^M \subseteq \mathcal{D}^M$.

Note that only model data is used to measure exceptionality. If the model induced on a subgroup is substantially different from the model induced on the entire database (or the subgroup's complement), exceptionality is large and we call this an *exceptional model*. We can now state our problem formally.

Problem 1 (*Exceptional Model Mining Problem*) Suppose we are given a database $\mathcal{D}$, an exceptionality measure ϕ and an exceptionality threshold ϵ. The task is to find all subgroups $\mathcal{G}$ with corresponding minimal subgroup descriptions, such that each $G \in \mathcal{G}$ implies an *exceptional model*, i.e. $\phi^{\mathcal{D}^M}(G^M) \geq \epsilon$.

It is obvious that the search space we are facing is huge: we would have to consider all possible subsets of the database. Exhaustive search is therefore not an option and we have to resort to heuristics. As a consequence, we will find a subset of the subgroups that satisfy the EMM problem.

2.1 Subgroup discovery approach

The EMM search strategy previously presented (Leman et al. 2008) is a straightforward extension of Subgroup Discovery strategies (Klösgen 2002). Such search strategies traverse the subgroup description search space by starting with simple descriptions and refining these along the way, going from generic to specific. In some cases a depth-first or breadth-first search is used, but in most settings a more heuristic strategy like beam search is required. Usually a minimum coverage threshold is used to ensure that a subgroup covers at least a certain number of tuples.

However, there are some disadvantages to this approach. First, it is often required to tune the search parameters to obtain good results. Second, heuristic approaches often suffer from local optima. Third, many subgroup descriptions imply identical subgroups. Additionally, the result usually contains many subgroups that are almost identical. Finally, it is quite likely that complex subgroups are never looked at, because runtime increases exponentially for deeper searches.

3 Information-theoretic exceptional models

In this section we introduce two exceptionality measures. For both, we assume that all model attributes are nominal, i.e. each A_i^M has a nominal domain.

3.1 Kullback–Leibler divergence

The first measure is based on the Kullback–Leibler (KL) divergence (Kullback and Leibler 1951), also called information gain. The KL divergence originates from information theory and has previously been used for data mining tasks, e.g. for clustering (Slonim and Tishby 1999).

It is an asymmetric measure of the difference between two probability distributions P and Q. It assumes that P is the 'true' distribution from which samples are drawn and that Q is a different, 'wrong' distribution. The KL divergence quantifies the number of extra bits which would be required to encode a sample from P using a code based on Q instead of using a code based on P.

For probability distributions P and Q of a discrete random variable, the KL divergence of Q from P is given as

$$\mathrm{KL}(P \| Q) = \sum_x P(x) \log_2 \frac{P(x)}{Q(x)}.$$

The KL divergence has two nice properties. First, it is always larger than or equal to zero, i.e. $\mathrm{KL}(P \| Q) \geq 0$. Second, it can only be zero if the two probability distributions are exactly the same, i.e. $\mathrm{KL}(P \| Q) = 0 \rightarrow P = Q$.

It makes perfect sense to use the KL divergence as exceptionality measure, since exceptionality should quantify how different the data distribution of the subgroup is from the data distribution of the entire database. This is exactly what the KL divergence does. All that we need to do is specify P and Q.

We could model multivariate probability distributions on the complete set of model attributes, but it is not immediately clear how to do this. Instead, we choose to assume that each attribute-value in our database is an independently drawn sample from an underlying, independent discrete random variable. Clearly, this is a Naïve Bayes (Warner et al. 1961) like assumption, which might be too optimistic.

For each attribute $A_i^M \in A^M$, we have a bag of samples that forms database $\mathcal{D}^M$ and we have a bag of samples that forms subgroup G^M. Since we do not know the probability distributions from which these samples were drawn, we estimate them using empirical distributions. That is, the probability of each possible value is the number of times it was sampled divided by the total number of samples. We denote the function which derives such an empirical distribution by $\hat{P}$. Because KL divergences of independent variables can be summed, we can now define KL exceptionality as the sum of KL divergences over all individual attributes, from subgroup to database.

Definition 4 (*KL exceptionality*) Given a database $\mathcal{D}$ and subgroup G, define (independent) *KL exceptionality* as

$$\phi_{\mathrm{KL}}^{\mathcal{D}^M}(G^M) = \sum_{i=1}^{l} \mathrm{KL}(\hat{P}(G^{A_i^M}) \| \hat{P}(\mathcal{D}^{A_i^M})).$$

3.2 Krimp gain

For KL exceptionality, we assumed that all attributes are independent. Because this may not always be realistic, we introduce a second measure that takes associations between attributes into account.

This second measure is based on the *Minimum Description Length* principle (MDL) (Rissanen 1978). The MDL principle states that given a set of models $\mathcal{M}$, the best model $M \in \mathcal{M}$ is the one that minimises the total encoded length, in bits, of both the model and the data encoded with the model.

The models we consider are sets of (frequent) itemsets with associated codes, called *code tables*. The best code table is the code table that compresses the data best. To approximate the optimal code table from a database, we proposed a heuristic algorithm

called KRIMP (Siebes et al. 2006). For this, it needs a database and a set of candidate itemsets. As candidates, frequent itemsets up to a given minimum support *minsup* are used. Note that although KRIMP only operates on itemset data, all nominal data can be easily treated as such.

In subsequent research with KRIMP, we have shown that it captures the underlying distribution of a database very well (Leeuwen et al. 2006). In Leeuwen et al. (2009) we introduced an algorithm that finds large and homogeneous groups in a database. For this purpose, we introduced a measure called *compression gain*, which quantifies how many extra bits are needed to compress a group with the code table of the entire database instead of the code table of the group. As such, it very much resembles the Kullback–Leibler divergence. Both quantify differences between distributions and are based on information theory. The main difference is that KRIMP encodes only present items ('1s'), '0s' are ignored. Furthermore, independent KL exceptionality does not take associations into account, while KRIMP does.

Because we only want to measure differences between distributions and not subgroup sizes, we define *Krimp Gain* (KG) as the average gain per tuple.

Definition 5 (*Krimp Gain*) Let $\mathcal{D}$ be a database, $G \subseteq \mathcal{D}$ a subgroup, and $CT_{\mathcal{D}}$ and CT_G their respective optimal code tables. We define the *Krimp Gain* of group G from $\mathcal{D}$, denoted by $KG(G\|\mathcal{D})$, as

$$KG(G\|\mathcal{D}) = \frac{L(G|CT_{\mathcal{D}}) - L(G|CT_G)}{|G|}$$

with $L(G|CT)$ the size of G, in bits, encoded with code table CT.

Note that, contrary to the KL divergence, Krimp Gain is not strictly positive: negative gains indicate that a group is compressed better as part of the database. This could e.g. be expected for a random subset of the data. For further details, please see Siebes et al. (2006), Leeuwen et al. (2009). Finally, we define the exceptionality measure.

Definition 6 (*KG exceptionality*) Let $\mathcal{D}$ be a database and $G \subseteq \mathcal{D}$ a subgroup. Define *KG exceptionality* as

$$\phi_{KG}^{\mathcal{D}^M}(G^M) = KG(G^M \parallel \mathcal{D}^M).$$

A large advantage of using KRIMP is that the resulting models are code tables, i.e. sets of frequent itemsets, together with information on how often they are used to encode the data. Code tables allow for manual inspection and can be easily interpreted by domain experts.

4 Exception maximisation description minimisation

In Sect. 2, we stated the problem of Exceptional Model Mining. We now introduce a new search strategy that is applicable to a wide range of EMM settings.

The basic idea of the algorithm is to start with some candidate subgroup and improve it iteratively. Problem 1 implies that two objectives have to be optimised for each

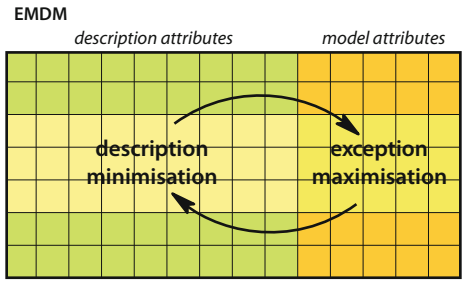

Fig. 1 EMDM in action

Algorithm 1 The EMDM Algorithm

Input: A database $\mathcal{D}$, a set of candidate subgroups $\mathcal{C}$ and exceptionality threshold ϵ.
Output: $\mathcal{S} \subseteq \mathcal{G}$, a subset of all subgroups satisfying the EMM Problem.

```
1: S ← ∅
2: for all G ∈ C do
3:   while G changes do
4:     G ← ExceptionMaximisation(G)
5:     G ← DescriptionMinimisation(G)
6:   end while
7:   if exceptionality(G) ≥ ε then
8:     S ← S ∪ {G}
9:   end if
10: end for
11: return S
```

individual subgroup: (1) maximise exceptionality and (2) minimise description complexity. It is highly unlikely that both can be optimised at the same time, since they are two different goals that may be conflicting. Hence, we propose an iterative algorithm which alternates between maximising exceptionality and minimising the description.

The generic algorithm is illustrated in Fig. 1 and given in more detail in Algorithm 1. The two main steps in the algorithm are Exception Maximisation (EM) and Description Minimisation (DM). Therefore, the algorithm is called Exception Maximisation Description Minimisation (EMDM).

Given a set of candidate subgroups, the algorithm takes each candidate as starting point and refines it to make it into an exceptional model that satisfies the specified exceptionality threshold. Each iteration consists of an EM-step followed by a DM-step. The subgroup may be changed by each of these steps and we continue performing both steps until the subgroup has not changed after a full iteration. Whether a subgroup stabilises very much depends on the data and the choices made for exceptionality and the description. It may not always be possible to find a description that matches an exceptional subgroup. Because of this, it may be required to set a maximum number of iterations in practice.

The two main steps have to be defined according to the specific problem setting. The EM-step depends on the domains of the model attributes, the model class and exceptionality measure and has the goal to change the subgroup such that exceptionality is maximised. The DM-step, on the other hand, depends on the domains of the description attributes, the description class and its complexity function and should

change the subgroup such that a description with low complexity can be assigned. Next, we will propose a specific instance for each step, using compression for the EM-step and classification for the DM-step.

4.1 Exception maximisation through compression

Both information-theoretic measures we introduced in Sect. 3 are based on differences in encoded size, using different codings for different distributions. Although we did not explicitly mention this, both KL and KG assume such a coding scheme and the induced models can therefore be regarded as *compressors*. In case of Krimp Gain, the compressor is simply the code table induced by Krimp. In case of the independent KL divergence, the compressor replaces each (model) attribute-value x by a code of optimal length, based on its marginal probability, i.e. $-\log_2(\hat{P}(A_i^M = x))$.

Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be two databases and let C_1 and C_2 be their respective optimal compressors. Denote the compressed size, in bits, of a tuple $t \in \mathcal{D}$ compressed with C, with $L(t \mid C)$. If we assume that $\mathcal{D}_1$ and $\mathcal{D}_2$ were generated from different underlying distributions, we can now decide to which distribution t most likely belongs, as detailed in Leeuwen et al. (2006):

$$L(t \mid C_1) > L(t \mid C_2) \Rightarrow P(t \mid \mathcal{D}_1) < P(t \mid \mathcal{D}_2).$$

Hence, the Bayes optimal choice is to assign t to the distribution that leads to the shortest code length.

In the current setting, we have a database $\mathcal{D}^M$ and a subgroup G^M. Since $G^M \subseteq \mathcal{D}^M$, the underlying distributions may be very similar. However, assuming that each EM-step starts with a modestly-sized, structured subgroup, G^M is likely to have a bias towards a specific part of the overall distribution of $\mathcal{D}^M$. We will use this bias to maximise exceptionality. For brevity, we omit the M from $\mathcal{D}^M$ and G^M in the remainder of this subsection.

First, we induce compressors $C_\mathcal{D}$ and C_G for $\mathcal{D}$ resp. G. We could then re-build the subgroup by taking those tuples for which $L(t \mid C_G) < L(t \mid C_\mathcal{D})$. This would most likely not increase the difference between the models though, but keep it the same. For this reason, we introduce a *minimal margin* σ : C_G should encode each tuple better than $C_\mathcal{D}$ by at least a certain amount of bits.

We make this margin dependent on the tuple and the database. Since the goal is to maximise exceptionality, we would like to favour tuples that are rare in the overall distribution over those that are common. This translates directly to tuples with relatively long resp. relatively short codes. Thus, the tuple with the shortest code will get the largest margin, such that a code length of 0 would be required to make it to the subgroup (which is impossible). The tuple with the longest code will get a margin of zero, everything in between is linearly scaled. Although the definition of the margin looks quite complex, its visualisation in Fig. 2 shows that its effect is actually quite straightforward.

Definition 7 (*Compression-based exception maximisation*) Assume a database $\mathcal{D}$, subgroup G, and optimal compressors $C_\mathcal{D}$ and C_G induced from $\mathcal{D}$ and G respectively.

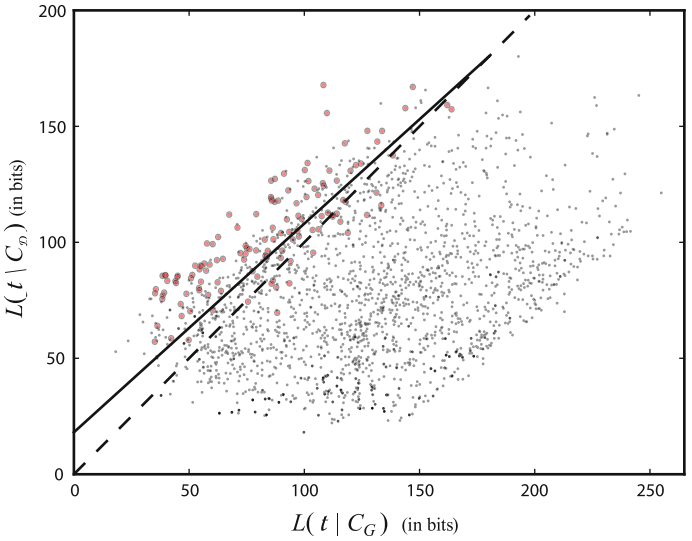

Fig. 2 The effect of the database-dependent margin. Plotted are the compressed sizes of all tuples in *Mammals*, encoded with the KRIMP code tables for the entire database (*vertical axis*) and a candidate subgroup (*horizontal axis*). The tuples that belong to the candidate subgroup are drawn larger. The *dashed line* represents the border imposed by the Bayes optimal choice, the *solid line* represents the border given by Definition 7. The exception maximised subgroup contains all tuples above this border

The exception maximised subgroup is

$$\{t \in \mathcal{D} \mid L(t \mid C_G) < L(t \mid C_{\mathcal{D}}) - \sigma^{\mathcal{D}}(t)\}$$

with

$$\sigma^{\mathcal{D}}(t) = L^{\mathcal{D}}_{min} - (L(t \mid C_{\mathcal{D}}) - L^{\mathcal{D}}_{min}) \times \frac{L^{\mathcal{D}}_{min}}{L^{\mathcal{D}}_{max} - L^{\mathcal{D}}_{min}}$$

$$L^{\mathcal{D}}_{min} = \min_{t \in \mathcal{D}} L(t \mid C_{\mathcal{D}}), L^{\mathcal{D}}_{max} = \max_{t \in \mathcal{D}} L(t \mid C_{\mathcal{D}})$$

4.2 Description minimisation through classification

Finally, we need to specify the subgroup descriptions we will use, and how to find and minimise them. Recall that a subgroup description is simply an indicator function s that returns 0 or 1 given a tuple, and a tuple $t \in \mathcal{D}$ belongs to G_s iff $s(t^D) = 1$. That is, given description data and a subgroup, we have a mapping from description data to {0,1} and the goal is to find a function that mimics this mapping as accurately as possible. This can be easily regarded as a binary classification task. After induction, the classifier 'predicts' a new subgroup.

Definition 8 (*Classification-based description minimisation*) Given are a database $\mathcal{D}$ and subgroup G. For each $t \in \mathcal{D}$, let classlabel l_t be 1 iff $t \in G$ and 0 otherwise.

Let c be the classifier induced on $\mathcal{D}^D$ with classlabels l. The *description minimised subgroup* is

$$\{t \in \mathcal{D} \mid c(t^D) = 1\}$$

Because precious care is taken to prevent classification methods from overfitting, we trust the method to return a description that is as simple as possible. By re-building the subgroup according to the predictions of the classifier, we ensure that we find subgroups for which a simple description can be given.

In this paper we will use RIPPER (Cohen 1995), for three reasons. First of all, because it is rule-based. The resulting rulesets are easy to interpret, which is a necessity for exploratory purposes. Second, it can handle both discrete and continuous data. Third, we did some initial (10-fold cross-validated) experiments with a set of well-known classifiers including RIPPER, in which RIPPER proved to perform well. Accuracy and recall were on par with results obtained with C4.5, Naïve Bayes, and SVM. For all classifiers, we use their respective implementations in Weka (Witten and Eibe Frank 2005), with default settings.

As description complexity, we use the number of conditions in a ruleset.

5 Experiments

As mentioned in Sect. 4, the EMDM algorithm may not always stabilise. Preliminary tests showed that if a subgroup stabilises, this is usually within 10–15 iterations. Therefore, we impose a maximum number of iterations of 25. If a group has not stabilised after this, we consider all subgroups that the algorithm has seen at the end of an iteration and pick that subgroup that has maximal exceptionality as result. We set the minimum exceptionality threshold to 0.

KL and KG compressors are used to encode all tuples in a database. To ensure that a compressor can encode all transactions (and avoid infinite exceptionality), a Laplace correction is applied, meaning that 1 is added to each of the counts. KRIMP with pruning is used and closed frequent itemsets are used as candidates (see Siebes et al. (2006) for details). The relative minimum support thresholds we used for computing Krimp Gain are given in Table 1.

We will use closed frequent itemsets as candidate subgroups, because all model attributes are binary and itemsets capture structure in such data. We only need to

Table 1 Datasets

For each dataset the number of tuples, the number of description and model attributes, and the *minsup* used for KG are given

Dataset	Properties			KG minsup
	$\lvert\mathcal{D}\rvert$	$\lvert A^D\rvert$	$\lvert A^M\rvert$	
Adult	48842	6	99	10%
Emotions	593	72	6	1%
Mammals	2221	67	124	15%
Scene	2407	294	6	1%
Yeast	2417	103	14	1%

consider closed frequent itemsets, as a subgroup is defined by its tuples, i.e. a candidate consists of all tuples in which a given itemset occurs.

We take the *Emotions*, *Scene* and *Yeast* datasets from the 'Mulan' repository (Tsoumakas et al. 2010), and the *Adult* dataset from the UCI repository (Asuncion and Newman 2007). Also, we use the *Mammals* dataset (Heikinheimo et al. 2007), which consists of presence information of European mammals (Mitchell-Jones et al. 1999) and climate information for areas of 50 × 50 kilometres.

Except for *Adult*, each dataset consists of numerical and binary attributes. The nominal attributes of *Adult* are converted to binary attributes, with one binary attribute for each attribute-value. In the remainder of the paper, we consider all numerical attributes as description attributes and all binary attributes as model attributes. Resulting dataset properties are given in Table 1.

5.1 Two example runs

To see how the EM- and DM-steps interact, we first investigate two individual EMDM runs. To this end, we selected two characteristic runs from *Adult* and *Mammals* and plotted subgroup size, exceptionality and description complexity over the runs in Fig. 3. For both runs, KG exceptionality was used.

The upper graphs clearly show that the *Adult* run quickly converges. Exceptionality first increases rapidly while the size of the subgroup decreases; both EM and DM

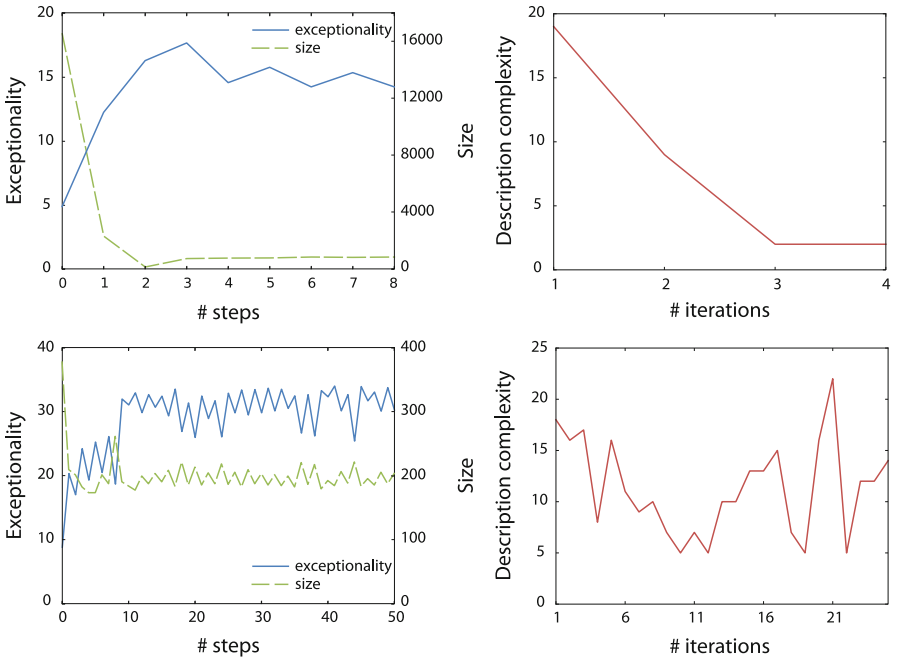

Fig. 3 Two example EMDM runs, for *Adult* (*top*) and *Mammals* (*bottom*)

contribute to this. After this, EM still manages to slightly increase exceptionality at each step, but this is undone by DM and when the description completely converges the EMDM run is finished.

The lower graphs give an example run for *Mammals* that does not satisfy our stopping criterion within 25 iterations. However, looking at exceptionality and size, the subgroup does seem to (almost) stabilise after about 10 steps. The problem is that some fluctuation remains, which is also clear from description complexity.

5.2 Quantitative results

Table 2 shows quantitative results obtained with candidate subgroups defined by closed frequent itemsets, using both KL and KG exceptionality. The first observation is that we end up with far less subgroups than that we start with. For *Yeast*, for example, we start with 487 candidate subgroups and with KL (resp. KG) this results in only 121 (resp. 88) subgroups. This can be explained by the fact that some candidate subgroups simply do not contain an exceptional part or no suitable subgroup description can be found. As a result, the subgroup is empty after a few steps and an empty subgroup is returned. The second observation is that even less unique subgroups are found, 20 resp. 29 for the examples just given, which is not surprising as candidate subgroups can overlap. On the contrary, this confirms that EMDM robustly identifies exceptional models, despite 'noise' in the initial candidates.

Average subgroup size, exceptionality and description are given for the top-10 ranked (with respect to exceptionality) subgroups. Sometimes it is worth looking a bit further down the list, where larger subgroups can be found. For the smaller datasets, KL seems to give more and larger results, which is unsurprising: in these datasets only few correlations exist and a measure that does not need large quantities of data has the advantage.

Table 2 EMDM results

Dataset	Experiment			Subgroups		Top-10 average		
	minsup	$\|\mathcal{C}\|$	Measure	$\|\mathcal{G}\|$	$\|\mathcal{U}\|$	Size	Except.	Descr.
Adult	10%	203	KL	56	19	576	10.9	13.1
			KG	107	107	144	16.2	17.8
Emotions	0.3%	25	KL	17	7	66	2.3	6.7
			KG	6	4	18	3.8	5.3
Mammals	10%	49661	KL	38238	309	5	32.3	2.3
		534	KG	518	516	35	51.0	7.8
Scene	0.1%	12	KL	8	7	238	2.6	25
			KG	9	8	241	2.2	24
Yeast	0.4%	487	KL	121	20	73	4.8	9.8
			KG	88	29	30	7.9	6.5

Minimum support 'minsup' is used to mine closed frequent itemsets, giving $|\mathcal{C}|$ candidates. For each experiment, the number of resulting groups $|\mathcal{G}|$ and the number of unique groups $|\mathcal{U}|$ is given. Average subgroup size, exceptionality and description complexity are given for the top-10 unique subgroups. When less than 10 unique subgroups are found, the average for all subgroups is given. For *Mammals*, $|\mathcal{C}|$ candidates were randomly selected from the complete set of closed frequent itemsets

Average description complexities vary from 5 up to 25 conditions. This indicates that our EMDM approach identifies more complex subgroups than are typically found with Subgroup Discovery approaches, which usually allow for up to 5-10 conditions. One may argue that 25 conditions for a single subgroup is rather specific. This is not a problem though, since a description is a disjunction of conjunctions and could therefore easily be split into a multiple of simpler subgroups that belong together.

Since absolute exceptionality values depend both on the dataset and the measure, these cannot be compared between experiments in Table 2.

5.3 Random vs structured candidates

To investigate whether it is important to use structured candidates, we compare candidates based on frequent closed itemsets to random subgroups and candidates based on KRIMP patterns. For each dataset, we generate 6000 random subgroups and use these as initial candidates. A random subgroup is generated by including each tuple in the database with uniform probability. For each dataset, 3 probabilities between 5% and 40% are chosen and 2000 candidate subgroups are generated with each. These probabilities are chosen such that the random subgroups have roughly the same sizes as the frequent itemset based candidates. We also compare to candidates based on patterns selected by KRIMP.

We define selection ratio as the number of unique subgroups found divided by the number of candidates. The higher this ratio, the less time needed to obtain good results. Table 3 shows selection ratios for the three types of candidates just described. Using random candidates usually leads to ratios close to 0% and never higher than 8%. Performance is much better with closed frequent itemsets as candidates, with an average selection ratio of 33.3%. With KRIMP candidates the highest ratios are obtained, with an average of 38%. This can be attributed to the fact that KRIMP removes redundant patterns and selects only those that are important for the structure in the data.

Although selection ratio does not say anything about subgroup size, exceptionality and description complexity, we observed that these are similar for the results obtained

Table 3 Selection ratio

Dataset	Measure	Random			Frequent			KRIMP		
		$\|\mathcal{C}\|$	$\|\mathcal{U}\|$	%	$\|\mathcal{C}\|$	$\|\mathcal{U}\|$	%	$\|\mathcal{C}\|$	$\|\mathcal{U}\|$	%
Adult	KL	6000	1	0.0	225	19	8.4	68	7	10.3
	KG		232	3.9		107	47.6		45	66.2
Emotions	KL	6000	5	0.1	25	7	28.0	6	4	66.7
	KG		10	0.2		4	16.7		2	33.3
Mammals	KL	6000	3	0.1	49661	309	0.6	252	26	10.3
	KG		466	7.8	534	516	96.6		235	93.3
Scene	KL	6000	0	0.0	12	7	58.3	0	–	–
	KG		1	0.0		8	66.7		–	–
Yeast	KL	6000	0	0.0	487	20	4.1	28	7	25.0
	KG		42	0.7		29	6.0		6	21.4

For each candidate set, *Random*, *Frequent* and KRIMP, the number of candidates $|\mathcal{C}|$, the number of unique subgroups $|\mathcal{U}|$ and selection ratio % is given

with all candidate sets and selection ratio should therefore be the main criterion for candidate selection. For small datasets, using closed frequent itemsets is recommendable because KRIMP may not give enough candidates. For larger datasets, using only KRIMP itemsets seems the best choice.

Runtimes depend very much on the data, the measure, and also varies per candidate. The experiments with *Emotions* and *Scene* finished within minutes, *Adult* and *Yeast* took up to 4 h. *Mammals* with frequent itemsets as candidates ran for 2 weeks, but this can be sped up significantly by using a stringent candidate selection. With KRIMP candidates, *Mammals* with KL needed only 2 h to finish and it took 6 days to complete with KG exceptionality. The implementation could be further optimised, e.g. by integrating the classifier.

5.4 Exceptional is interesting

To show that the exceptional subgroups EMDM discovers are indeed interesting, we give example results for two datasets.

5.4.1 Emotions

Each tuple in *Emotions* represents a music song, from which 8 rhythmic and 64 timbre features were extracted (description attributes). To each song, experts assigned any number of six emotions: *amazed-surprised*, *happy-pleased*, *relaxing-calm*, *quiet-still*, *sad-lonely*, and *angry-fearful* (model attributes).

Since *Emotions* is a relatively small dataset, we take a closer look at the results obtained with KL and closed itemsets as candidates. Figure 4 shows how the relative frequencies (percentages of ones) of the model attributes differ between the complete database and the 7 subgroups found. Frequencies are similar within the database, but the subgroups clearly identify parts of the data in which some attributes are more prevalent than others.

As an example, consider the fourth subgroup, G_4. This subgroup consists of 37 songs that sound mostly happy and relaxing, but certainly not angry, sad or quiet. The subgroup description consists of 4 inequalities, describing the subgroup in terms of rhythmic and timbre features.

5.4.2 Mammals

Figure 5 shows the regions corresponding to 4 typical example subgroups found on *Mammals*. For this, we explored the 20 top-ranked subgroups of 2 experiments, one with KL and one with KG exceptionality. Both experiments used KRIMP patterns as candidates.

With KL as measure, areas in which relatively many rare mammals occur are identified. This can be explained by the fact that all mammals are treated independently by this measure, only the differences between the marginal frequencies of the subgroup and the database matter. The leftmost area can be characterised by relatively many occurrences of the *Arctic fox*, *Skunk bear*, *Norway lemming* and *Reindeer*.

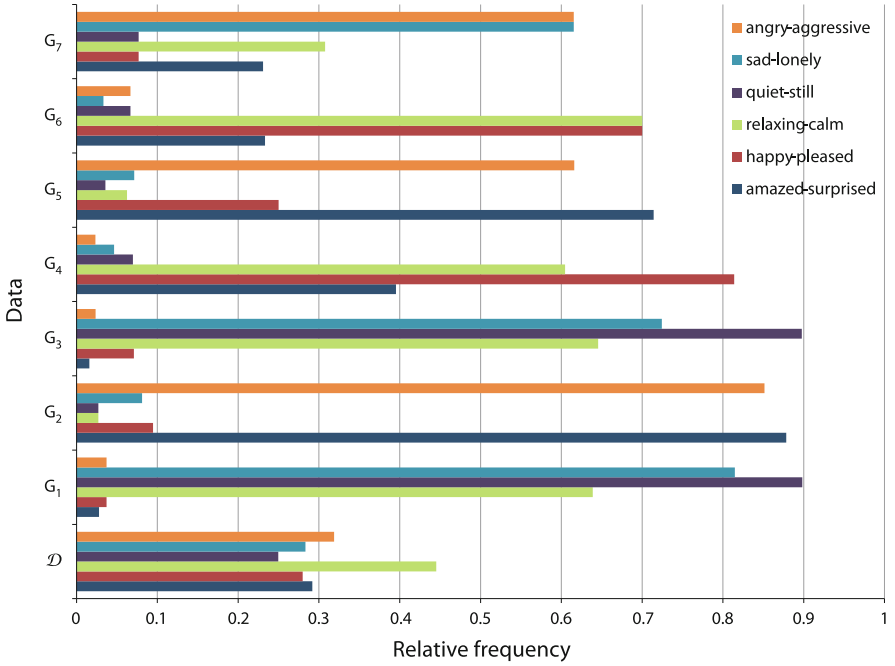

Fig. 4 Relative frequencies for the model attributes. *Emotions* with KL exceptionality

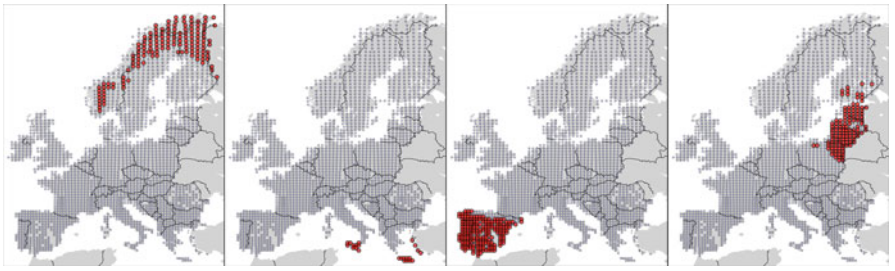

Fig. 5 The regions corresponding to 4 example exceptional models on *Mammals*. The 2 examples on the *left* were found with KL as measure, the 2 on the *right* with KG

The following description belongs to this subgroup: (max temp September $\leq$ 11.1° and max temp October $\leq$ 3.3°) or (max temp September $\leq$ 11.1° and max temp April $\leq$ 3.47° and temperature seasonality $\geq$ 619.22998 and max temp November $\geq -2.56°$).

Mammals that occur relatively often in the second area from the left are the *Cretan spiny mouse*, *Wild goat*, *Sicilian shrew* and *Corsican hare*.

When KG is used as measure, subgroups typically represent areas where large groups of (common and less common) mammals co-occur. In Spain and Portugal, the 3rd example in Fig. 5, the *European otter*, *Western European hedgehog*, *Red fox*, *Common genet* and *Granada hare* very often occur together. The description consists of 16 climate conditions.

Table 4 Beam search results

Dataset	Measure	$\|\mathcal{U}\|$	Top-10 average		
			Size	Except.	Descr.
Adult	KG	344	123	19.4	5
Emotions	KL	4	12	6.5	4.5
Mammals	KG	594	62	53.1	2
Scene	KG	1	20	2.8	5
Yeast	KL	4	12	12.9	4.3

The number of unique subgroups is shown, together with average size, exceptionality and description complexity for the top-10 ranked subgroups

Note that the algorithm knew nothing about the actual locations of the areas, these subgroups were established using only the climate and mammals presence information. Still, geographically sound areas were found.

All in all, KL exceptionality is fast, works well on both *Emotions* and *Mammals*, finds sub-distributions of *Emotions* and rare distributions in *Mammals*. KG exceptionality, on the other hand, is not so suited for small datasets like *Emotions*, but finds large coherent areas in *Mammals*.

5.5 Comparison to beam search

To compare to a Subgroup Discovery approach, we experimented with identical exceptionality measures but a different search strategy. A subgroup discovery beam search strategy was applied, with beam width 200, maximum search depth 5 and a minimum size of 10 (except for *Adult*: 100). Numerical attributes were locally discretised into 5 equi-sized bins, upon refining a candidate subgroup. For each experiment, the 1000 highest ranked subgroups were kept for analysis.

The results of these experiments are shown in Table 4. For each dataset, either KL or KG is shown (results of combinations not shown are either comparable or worse). The maximum runtime was set to 2 weeks. *Mammals* did not finish in time; it only reached a search depth of 2. This shows EMDM finds complex subgroups much quicker than a subgroup discovery approach.

The results show that relatively few unique subgroups were identified for the smaller datasets. Except for *Mammals*, the resulting subgroup sizes and description complexities are almost always identical or very close to the maxima imposed by the parameters. This indicates that (1) smaller subgroups are always deemed more exceptional and this is not counterbalanced by this search approach, and (2) deeper searching is necessary, at the expense of runtime.

6 Related work

Umek et al. (2009) addressed a problem that could be considered an instance of EMM and proposed to use clustering. After clustering the description and the model data independently, a statistical analysis is used to determine which description and model

clusters coincide. A drawback of this approach is that only the computed segments are candidate subgroups.

The specific instance of EMM we consider, i.e. with numerical description data and binary model data, strongly resembles the problem setting of Garriga et al. (2007). Their objective is different, as they find segmentations (or clusterings) of databases consisting of both numeric and binary data.

Similarly, the goal of information-theoretic clustering methods (Andritsos et al. 2004; Slonim and Tishby 1999) is to find a segmentation of a database according to its underlying distributions. This strongly differs from our use of information-theoretic measures, because we look for parts of the data that differ from the overall distribution and explicitly distinguish description and model attributes.

7 Conclusion

We introduce a new algorithm that efficiently finds maximally exceptional subgroups with minimal descriptions. EMDM is a generic algorithm for Exceptional Model Mining that iteratively improves subgroups through Exception Maximisation (EM) and Description Minimisation (DM). Structure in both the description and model spaces is exploited.

Experiments show that both the generic EMDM algorithm and the specific instances given for the EM- and DM-steps perform well. The EM-step is strongly linked to the exceptionality measure and we therefore introduce an instance based on information theory in this paper, the DM-step we propose is generic and uses a rule-based classifier.

We propose two information-theoretic exceptionality measures, one based on the KL divergence and one based on KRIMP. KL exceptionality treats all attributes as independent, is fast, works well on smaller datasets, and finds rare distributions in larger datasets. KG exceptionality, on the other hand, takes associations between attributes into account, works especially well on larger datasets, and finds large coherent regions in the data.

Acknowledgements This research is financially supported by the Netherlands Organisation for Scientific Research (NWO) under project number 612.065.822.

References

Andritsos P, Tsaparas P, Miller RJ, Sevcik KC (2004) LIMBO: scalable clustering of categorical data. In: Proceedings of the EDBT, pp 124–146

Asuncion A, Newman DJ (2007) UCI machine learning repository. http://www.ics.uci.edu/~mlearn/MLRepository.html

Cohen WW (1995) Fast effective rule induction. In: Proceedings of the ICML'95, pp 115–123

Garriga GC, Heikinheimo H, Seppänen JK (2007) Cross-mining binary and numerical attributes. In: Proceedings of the ICDM'07, pp 481–486

Heikinheimo H, Fortelius M, Eronen J, Mannila H (2007) Biogeography of european land mammals shows environmentally distinct and spatially coherent clusters. J Biogeogr 34(6):1053–1064

Klösgen W (2002) Subgroup discovery chapter 16.3. Oxford University Press, Oxford

Kullback S, Leibler RA (1951) On information and sufficiency. Ann Math Stat 22(1):79–86

Leeuwen M, Vreeken J, Siebes A (2006) Compression picks the item sets that matter. In: Proceedings of the ECML PKDD'06 pp 585–592

Leeuwen M, Bonchi F, Sigurbjörnsson B, Siebes A (2009) Compressing tags to find interesting media groups. In: Proceedings of the CIKM'09, pp 1147–1156

Leman D, Feelders A, Knobbe A (2008) Exceptional model mining. In: Proceedings of the ECML/PKDD'08, 2:1–16

Mitchell-Jones AJ, Amori G, Bogdanowicz W, Krystufek B, Reijnders PJH, Spitzenberger F, Stubbe M, Thissen JBM, Vohralik V, Zima J (1999) The atlas of european mammals. Academic Press, London

Rissanen J (1978) Modeling by shortest data description. Automatica 14(1):465–471

Siebes A, Vreeken J, van Leeuwen M (2006) Item sets that compress. In: Proceedings of the SDM'06, pp 393–404

Slonim N, Tishby N (1999) Agglomerative information bottleneck. In: Proceedings of the NIPS'99, pp 617–623

Tsoumakas G, Vilcek J, Spyromitros L (2010) MULAN: a java library for multi-label learning. http://mulan.sourceforge.net/

Umek L, Zupan B, Toplak M, Morin A, Chauchat J-H, Makovec G, Smrke D (2009) Subgroup discovery in data sets with multi-dimensional responses: A method and a case study in traumatology. In: Proceedings of AIME'09, pp 265–274

Warner HR, Toronto AF, Veasey LR, Stephenson R (1961) A mathematical model for medical diagnosis, application to congenital heart disease. J Am Med Assoc 177:177–184

Witten IH, Frank Eibe (2005) Data mining: practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco