Olfactory qualities characterization with data mining techniques : Subgroup discovery and Exceptional Model Mining frameworks

Roland Kotto Kombi
Master 2 TIWe UCBL Lyon 1

Résumé

Parmi la connaissance que nous avons des sens, l'odorat reste certainement celui dont la compréhension reste la plus limitée. En effet, bien que de très nombreuses molécules odorantes aient été répertorié et que l'activation des nerfs olfactifs ait été étudié, il manque un maillon essentiel à la compréhension de l'odorat : la corrélation entre propriétés moléculaires et odeur ressentie. En effet, sans compter la grande variabilité entre l'odorat de sujets aléatoires, il n'existe pas de modèle permettant de déterminer quelle sera l'odeur d'une molécule. En partenariat avec le Centre de Recherche en Neurologie de Lyon (CRNL), le projet Olfamining vise à mettre en évidence la corrélation entre propriétés moléculaires et odeurs et d'en exposer les applications dans le domaine de la parfumerie ou de l'alimentaire. Le CRNL a donc créé et enrichi une base de données sur un échantillon de plus de 1600 molécules odorantes décrites par plus de 1700 propriétés physico-chimiques mais également l'odeur dégagée par la molécule. A partir de cette base de données, nous allons utiliser et améliorer des techniques issues de la fouille de données afin d'extraire les connaissances implicites contenues dans cette base de données.

Abstract

Among human senses, sense of smell is the less understood. Even if many smelling molecules have been precisely described and nervous mechanisms implied in olfaction have been studied, there is a missing element in order to understand completely olfaction : the correlation between molecular properties and smells. Actually, excepting sense of smell variability among a sample of persons, there is no model that predict a molecule smell based on molecular properties. With the partnership of the Neurologic Research Center of Lyon (CRNL), Olfamining project aims to find correlations between some molecular properties and smells with all possible applications implied. CRNL has created and fulfilled a database including more than 1600 smelling molecules, each of them described by more than 1700 molecular properties. From this database, we will use and improve data mining techniques in order to discover knowledge included in this database.

Table of Contents

Chapter 1

Introduction

Olfamining project Nowadays, many industrial products, like food or perfumes, are composed of nice-smelling synthetic molecules in order to please the consumer. It is a relevant challenge for perfume and food industries to be able to predict the odor quality of molecules non-already synthetized. Nevertheless, the few works lead into olfactory qualities characterization, like [10] or [2], consists in the build of partitions of olfactory qualities which match which partitions on molecular properties. The Olfamining project is an ambitious project which aims to bring the missing link between the molecular properties and the human perception.

olfactory qualities An olfactory quality is defined [10] as a significant smell different from the others like vanilin or anisic. There are also families of olfactory qualities like fruity or green. The element that makes those qualities singular is the olfactory receptors configuration they produce when a person smells them. Actually, any human has olfactory perceptors at the base of the nose. olfactory receptors can be seen as nets of that can only catch molecules of a particular kind. When a receptor has captured a molecule, it sends a nervous message to the brain which says that a kind of molecules has been captured. The brain considers all kinds of captured molecules to translate it into a particular smell as decribed in [11].

Scientific issues From a neurobiologic point of view, explained in [2], a gap exists between the molecules and the olfactory receptor configurations they induce. The answer to this unsolved question relies on the molecular properties but is not trivial because, as said before, different molecules considering their shapes or mass, may induce the same olfactory quality. A study on a large range of physical and chemical properties is needed to find the link between molecular properties and human perception as tried in [16].

The ARCTANDER dataset A dataset, labeled ARCTANDER, is composed of 1689 molecules, called dragons, physical and chemical descriptions. Built and fulfilled by our partner of the Neuroscience Research Center of Lyon (CRNL), it respects a standard classification of known olfactory molecules. It can be represented as follow :

Dragon ID	Physical and chemical properties	Associated olfactory qualities
Identifier 1	....	q_i, q_j
Identifier 2	....	q_i
Identifier 3	....	q_i, q_j, q_k
....	....	....

As we can see, the set of associated olfactory qualities to an identifier is a subset of 74 qualities with a variable size. There are 1704 physical and chemical properties to describe each dragons like the volume or the number of carbon atoms.

The use of data mining techniques This dataset appears like the perfect support to discover the relation between physical properties and olfactory qualities. Actually, there are enough properties to highlight patterns into them that induce specific qualities with satisfying probabilities. This task fits with data mining methods which aims to extract implicit knowledge from data, especially patterns defined in [8]. In this study, we focus on descriptive techniques because the aim is not to build a model for comparable datasets.

Formalization of the solution In order to explore only techniques which could answer, even partially, to the problem, we need to set formally the form of our solutions. Let D be a set of data. D can be seen as the cartesian product of sets O and A where O = $\{o_1, ..., o_n\}$ is a set of objects, or individuals, and A = $\{a_1, ..., a_m\}$ is a set of attributes which describe each object. Let an attribute denoted class $\in$ A be the attribute that interest us and V = $\{v_1, ..., v_p\}$ be a set of nominal values. The attribute class is a subset of V and all the a_i are included in $\Re$, with i $\in$ [1, m]. We want to extract the association rules characteristic for the attribute class with respect to D with the following representation :

$$a_{i_1} = v_{j_1}, ..., a_{i_k} = v_{j_k} \longrightarrow \text{class} \subseteq V,$$
$$a_{i_1}, ..., a_{i_k} \in \text{A}\backslash\{\text{class}\}.$$

Chapter 2

Preliminaries

Global notions from data mining must be introduced before the development of related work.

Association rules Association rules are presented in [1] as statements like "93% of football fans look FIFA World Cup final". Let consider the following dataset and the value set O = {Object 1, Object 2, Object 3}:

	Value
Object 1	A,B
Object 2	A
Object 3	A,B

The *support* of a value v is the cardinality of the subset of O where each element is associated to v. For example $support(B) = |\ \{\text{Object 1, Object 3}\}\ | = 2$. An association rule, of the form $value_i \longrightarrow value_j$, determines the probability that $value_i$ implies $value_j$. The $confidence$ of this rule is defined as follow :

$$confidence(value_i \longrightarrow value_j) = \frac{support(value_i \wedge value_j)}{support(value_i)}$$

For example, the association rule confidence of $A \longrightarrow B$ equals $\frac{2}{3}$.

Pattern mining formalization In order to understand how works subgroup discovery, it is important to introduce a fundamental framework in pattern mining presented in [8]. The pattern mining problems can be formalized as follow : Let D be a dataset, L a class of sentences for defining subgroups of D and q a selection predicate. Pattern mining problems are of the form $if\ \theta\ then\ \varphi$ with $\theta, \varphi \in L$. φ defines an 'interesting' subset of D. Moreover, there is a relation order between θ and φ denoted $\preceq : \theta \preceq \varphi$ means that θ is more general than φ and in the opposite way φ is more specialized than θ. The solution of this kind of problem is to find the theory of D included in L with respect to q. It corresponds to the set $Th(L, D, q) = \{\varphi \in L \mid q(D, \varphi)\ is\ true\}$.

Chapter 3

Related work

1 Subgroup Discovery

Subgroups have been introduced in [5] and formalized in [17] like subgroups among a population which are statistically the *most interesting* by their unusualness and their size with respect to a particular attribute of interest. The considered population is composed of individuals, or objects, and a set of attributes which describes each object. Different approaches exist for subgroup discovery : on one hand, an exhaustive approach guarantee the optimal solution with respect to an optimization criterion. On the other hand, many heuristics exploit the unusualness and the size of subgroup in order to evaluate the quality of subgroups generated from a given dataset.

We present in this part the formalization of the subgroup discovery (SD) framework and a unifying framework with compatible works. Then, we explain how subgroups are built while exploring the search space. We expose three important quality measures which define the interestingness of a subgroup. After this, we introduce the heuristic approach of SD. Finally, we present three recent and compatible works which lead to the solution of our issue: SD application to numerical domains, exceptional model mining and diverse subgroup set discovery.

1.1 Formalization of the SD framework

Before exposing subgroup construction and search space exploration, it is necessary to formalize properly what is a subgroup. According to [17], subgroup discovery can be formalized as, considering :

- a dataset D composed of individuals $t_i \in$ T described by attributes a_i and an attribute $class \in$ A
- L_F the language of all attribute-value pairs called $features$
- an evaluation function q which defines the interestingness of any bag of tuples G $\subseteq$ T according to $class$

We want to find :

- a set F $\subseteq L_F$ of features of maximum size $\mid A \mid -1$
- for each f $\in$ F, q(f, D) > 0
- for any $f^{'} \in L_F \setminus \{F\}, q(f^{'}, D) < min_{f \in F} q(f, D)$

However SD is not the only framework which aims to find unusual groups among a population. But we will show that all works are compatible.

Unification in rule learning framework There are many works under pattern mining that are especially related to the discrimination of bag of tuples that are statistically different considering a given model. In [4], the authors suggest an unifying framework more specific than the global framework presented [8] for those works. Let introduce first contrast sets and emerging patterns.

Contrast Set Considering the same dataset defined in subgroup discovery, a contrast set is a conjunction of attributes and values that differ significantly in their distributions across groups. The groups are defined by the attribute of interest, denoted class in subgroup discovery. The representation of contrast set can be the same than subgroup but the mining algorithm is different. For example a rule $X \longrightarrow Y$ is discarded if it does not satisfy the following test of *productivity* :

$$\forall Z \subset X : confidence(Z \longrightarrow Y) < confidence(X \longrightarrow Y)$$

The value $confidence(X \longrightarrow Y)$ is a maximum likehood estimate of conditional probability P(X|Y), which is estimated by the ratio $\frac{count(X,Y)}{count(X)}$. The function $count(A)$ returns the number of examples in the dataset where A is true.

Emerging Pattern Emerging patterns are itemsets whose support increases meaningfully from one data set to another. The aim is to obtain the differenciating characteristic between given classes of data. In order to represent each class of data, we can see an emerging pattern as an association rule with an itemset rule antecedent and a fixed consequent : $Itemset \longrightarrow D_1$. D_1 is a given dataset we want to compare with an other one D_2. To measure the quality of an emerging pattern, we evaluate the ratio of the support across D_1 and D_2 called *growth rate*. The support defines the percentage of examples including a given itemset in a set of examples. In practice this value is defined considering emerging patterns like association rules:

$$GrowthRate(ItemSet, D_1, D_2) = \frac{confidence(ItemSet \longrightarrow D_1)}{1 - confidence(ItemSet \longrightarrow D_1)}$$

The different terminologies used for contrast sets, emerging patterns and subgroups only come from the different communities which develop each of those notions but they can be unified in a rule learning framework introduced in [9] as follow :

Contrast Set Mining	Emergent Pattern Mining	Subgroup Discovery	Rule Learning
contrast set	itemset	subgroup description	rule condition
groups $G_1, ..., G_n$	data sets D_1 and D_2	class/property C	class/concept C_i
attribute-value pair	item	logical (binary) feature	condition
examples in groups $G_1, ..., G_n$	transactions in data sets D_1 and D_2	examples of C and $\overline{C}$	examples of $C...C_n$
examples for which the contrast set is true	transactions containing the itemset	subgroup of instances	covered examples
support of contrast set on G_i, support of contrast set on G_j	support of EP in data set D_1, support of EP in data set D_2	true positive rate, false positive rate	true positive rate, false positive rate

Table 1: Compatibility of definitions between contrast set, emerging pattern and subgroup

With the same idea it is simple to translate each objective defined in the different communities in a unique formalization of the expected result like illustrated in the following figure:

Contrast Set Mining	Emergent Pattern Mining	Subgroup Discovery	Rule Learning
Given examples in G_1 vs. G_j from G_1,...G_i	**Given** transactions in D_1 and D_2 from D_1 and D_2	**Given** in examples C from C and $\overline{C}$	**Given** examples in C_i from C_1 ...C_n
Find $ContrastSet_{i_k} \rightarrow G_i$ $ContrastSet_{j_l} \rightarrow G_j$	**Find** $ItemSet_{1_k} \rightarrow D_1$ $ItemSet_{2_l} \rightarrow D_2$	**Find** $SubgrDescr_k \rightarrow C$	**Find** $\{RuleCond_{i_k} \rightarrow C_i\}$

Table 2: Compatibility of objectives between contrast set, emerging pattern and subgroup

Now we need to see how these formalization can help in subgroup construction and how to explore the search space.

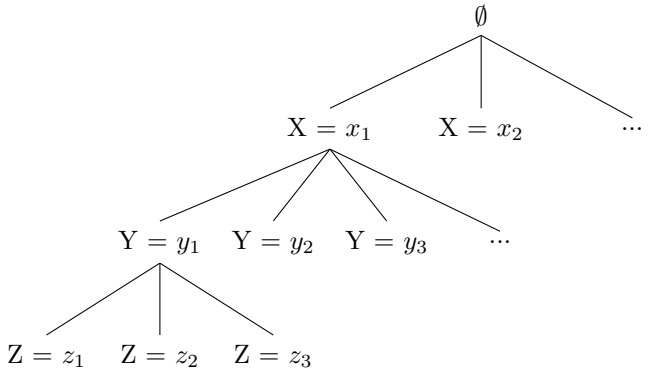
Subgroups construction and exploration

Notations Let consider a dataset S where each individual is described by a set of attributes A = {a_1,...,a_n} and an attribute of interest denoted class. We call A the description space and the attribute class, the model space. The model space can be n-ary or numeric without repercussion on the SD framework execept the choice of the quality measure. We define S^D and S^M respectively the projection of S on the description space and on the model space. If we consider only S^D as dataset, the set of all attribute-value pairs as language L we have the formalism of pattern mining framework less the selection predicate. It is actually the part that makes SD singular. For each l ∈ L, we denote G the bag of tuples that satisfy l and we consider a score, or quality, based on the unusualness of values distribution in G^M according to the values distribution in S^M, with G^M the projection of G on the model space. In many algorithm, we want to consider only the top-k subgroups considering the quality so the selection predicate can be expressed like : is the current subgroup quality better than the $k-th$ quality in the resulting subgroups set. With this matching with pattern mining framework, we can use the same approach.

Search space exploration Let consider the following example :

	X	Y	Z	Class
Object 1	x_1	y_1	z_1	v_1
Object 2	x_2	y_2	z_2	v_2
Object 3	x_3	y_3	z_3	v_3

The description space is composed of the attributes X, Y and Z and the model space corresponds to the attribute Class. The language is composed of pairs L = {X = x_1, X = x_2, ... }. We can then build the search space as a tree :



Here is just a part of the search space in order to show how work SD. It consists in a breadth-first search in the complete approach. It is important to notice that a edge is equivalent to a logical AND. So, while exploring the node labeled Z = z_1 in the partial example, it means considering all tuples which satisfy the description X = x_1 AND Y = y_1 AND Z = z_1. It corresponds to the

tuple Object 1 and his projection on the model space is $\{v_1\}$. The quality is then based on $\{v_1\}$ considering $\{v_1, v_2, v_3\}$. We show later that there is many quality measures from statistics essentially that express the unusualness of a subgroup.

Quality measures Some quality measures can be used on different target types but there are specific ones considering a given target type. The article [14] presents seven of them but only three are interesting considering the experiments presented further.

Meantest The MeanTest (MT) is a quality measure from statistics which fits well on single numeric target. It quantifies the difference between the mean of the target values in the entire set and in a given subgroup. The higher is the MT, the more singular is the subgroup and the larger is his support. Actually, for a given subgroup G, MT(G) close to 0 means that the distribution of target values in the subgroup is very close of the distribution in the entire dataset so the subgroup does not respect the unusualness criterion. Let S be a dataset, G be a subgroup, $\mid G \mid$ be G support size and μ^S and μ^G be respectively the mean of target values in S and in G. The meantest of G can be formalized as follow :

$$MT(G) = \sqrt{\mid G \mid}(\mid \mu^G - \mu^S \mid)$$

Weighted Krimp Gain The Weighted Krimp Gain (WKG) is a quality measure, presented in [15], which uses KRIMP code tables as models. A KRIMP code table is a list of itemsets where each itemset is associated to a code for a given dataset. The list is ordered considering the most frequent itemsets in the dataset and codes are chosen in order to use the minimal number of bits for the entire dataset encoding. The encoding simply consists in the substitution of the itemsets by their associated code. Relying on this concept, WKG evaluates the quality of a subgroup if the KRIMP code table or compressor, for a given subgroup G, encodes better the tuples in G considering the encoding of G using the compressor of the entire dataset. Let S be a dataset, G be a subgroup and L(G $\mid$ CT_x) be the size, in bits, of G encoded with the code table CT which is optimal for a set of tuples x. WKG can be defined as follow :

$$WKG(G \parallel S) = L(G \mid CT_S) - L(G \mid CT_G)$$

Weighted Kullback Leibler divergence The Kullback Leibler divergence (KL), introduced in [6], is a measure from statistics which quantifies the average number of extra bits necessary to encode a sample of a probability distribution P using the optimal encoding for a probability distribution Q instead the optimal one for P. Let S be a matrix with l columns and m lines. Let c_i be and t_j be respectively the $i-th$ column and the $j-th$ line. Finally let X be a random variable taking values on a domain V = $\{v_1, ..., v_n\}$. We assume that each value $S(i, j)$ is a sample of X. Let consider the following example with l = 4, m = 6 and V = $\{0, 1\}$:

	c_1	c_2	c_3	c_4
t_1	1	0	1	0
t_2	1	1	0	0
t_3	0	0	0	0
t_4	1	0	1	1
t_5	0	1	0	1
t_6	1	1	0	1

Then let q be the set of all t_j and p be the subset $\{t_1, t_2, t_3\}$. Let the function Distr(p) associates each value appearing in p with her occurence. In this example we have for p and the following Distr :

$$Distr(p) = \{(0, 8), (1, 4)\},\ Distr(q) = \{(0, 12), (1, 12)\}$$

Let simplify the notations and consider the frequency instead of the occurences. We define P and Q as follow :

$$P = \{(0, \tfrac{2}{3}), (1, \tfrac{1}{3})\},\ Q = \{(0, \tfrac{1}{2}), (1, \tfrac{1}{2})\}$$

The optimal encoding for each value in probability distribution is given by her entropy. So, we can define the number of bits $Code_v^{Prob}$ necessary to encode a value v associated to a probability $pb(v)$ in a probability distribution $Prob$ with the formula :

$$Code_v^{Prob} = -pb(v)\log_2(pb(v))$$

In the example we have the following code lengths :

$$Code_0^P = -\tfrac{2}{3}\log_2(\tfrac{2}{3}) = 0.39$$
$$Code_1^P = -\tfrac{1}{3}\log_2(\tfrac{1}{3}) = 0.52$$
$$Code_0^Q = Code_1^Q = -\tfrac{1}{2}\log_2(\tfrac{1}{2}) = 0.5$$

Finally let t_j^{Prob} be the $j-th$ line of S encoded with the optimal codes of the probability distribution $Prob$. In our example we need the following number of bits to encode t_1 according to P and Q :

$$t_1^P = 2 * 0.39 + 2 * 0.52 = 1.82$$
$$t_1^Q = 4 * 0.5 = 2$$

We can then define the average difference between the code lengths using the KL divergence just knowing P, Q and the domain of the random variable with the formula :

$$KL(P \parallel Q) = \sum_x P(x)\log_2 \tfrac{P(x)}{Q(x)}$$

Assuming that the each attribute-value pair is an independant sample of a random variable X, we want to estimate the divergence between a subgroup G and the entire dataset S. Let consider S as the joint of a description space D and a model or target space M. The description space is composed of description attributes D_i and model attributes M_i. Now let consider G^M and S^M as the projection of G and S on the model space. Instead of computing the probability distribution for G^M and S^M we use the associativity of the KL divergence to estimate the exceptionality of G. Actually, if we consider X as the set of random variables x_i and each column in S^M is a sample of an x_i, we can compute the exceptionality of G from the KL divergences of each column. Let $\hat{P}(M_i = v_i)$ be the probability associate to a value v_i, $\hat{P}(M_i = v_i)$ is computed as follow : $\hat{P}(M_i = v_i) = \frac{|\{t \in S | t^{M_i} = v_i\}|}{|S|}$. The KL divergence of G^M and S^M is then described as the sum of KL divergence of each M_i. In [13], the author suggested a variant of KL which includes the size of the considered subgroup in order to balance between KL divergence value and size of the subgroup, smaller subgroups tend to have greater KL divergence values. WKL is then defined as :

$$WKL(G^M) = \frac{|G|}{|S|} \sum_{i=1}^{l} KL(\hat{P}(G^{M_i}) \parallel \hat{P}(S^{M_i}))$$

Heuristic approach Until now we only considered an exhaustive search space exploration but it is easy to see that the search space grows deeply with the description space and each attribute domain cardinalities. So, heuristic algorithms have been developed in order to increase the scalability of SD.

Beam search One tool from graphs is the use of a beam in order to generate a fixed number of candidates at each step. Let take back the example in 1.1, the first candidates were the 9 attribute-value existing pairs and we generated children for each of them. But now, if we decide to consider only the 3 best candidates, according to their respective qualities, to generate children nodes. Same thing at each level in the search space, we reduce significantly the total number of candidates. The beam size is chosen considering a balance between computation time and subgroups qualities.

Other heuristic approaches exists like the optimistic estimate that is presented in the next work introducted in [3].

1.2 Subgroup discovery in numerical domains

In many cases considered until now in SD, we assumed that the values of the attributes are nominals. It is then simple to formalized subgroups forms with the formula above. But in practice, we often need to deal with continuous attribute values. In this case, the former formalism does not work anymore because the great majority of values are distinct so an another mechanism of exploration is needed to extract subgroups among a dataset.

Discretization approaches The main idea to solve that problem is to discretize, which means turn a database including continuous values into an almost equivalent database, from a data perspective, including only nominal values in order to link with the known formalization of subgroup discovery. The objective of the method is to replace all continuous values by a finite set of intervals. This set is defined in order to cover all continuous value and to respect an optimization criterion ε, considering intervals of the following form :

$$[v_{i_{min}}, v_{i_{max}}],$$

we impose the property on the given criterion ε:

$$\forall \text{i, } v_{i_{max}} \text{ - } v_{i_{min}} = \varepsilon$$

or considering S as the set of object which can be described by the interval $[v_{i_{min}}, v_{i_{max}}]$:

$$\forall \text{i, } |\text{S}| = \varepsilon$$

Even if this discretization method works, the techniques to generate these intervals ignore intervals which overlap some others. Let T $\subset \Re$ be a finite ordered set,the intervals considered are the following ones :

$$]-\infty, t_1],]t_1, t_2],...,]t_{n-1}, \infty]$$

This approach has the drawback to provide only suboptimal results because it implies a loss of information more or less significant depending on the choice of the optimization criterion. So, while using this discretization, the choice of the optimization criterion is balanced by the expected accuracy of the results and the number of intervals generated, because the more intervals there are the more it takes time to find all of them.

Intervals overlap approach : MergeSD A new approach suggested in [3] is to consider some intervals which overlap some others in order to find the most optimal results. The idea is to let the datbase with continuous values and compute a discretization on-the-fly by finding some bounds which include the continuous value with respect to an optimization criterion. Let D be a database with continuous values, sd a subgroup description and sd' a refinement of sd denoted sd' $\succ$ sd, if sd is a subset of sd'. A quality function q defines the interestingness of a subgroup. An optimistic estimate denoted Θ is a function that provides a bound of quality of all refinements of sd. Θ must satisfy the following condition :

$$\forall \text{ subgroups sd, sd', sd'} \succ \text{sd} \Longrightarrow \Theta(\text{sd}) \geq \text{q(sd')}.$$

Then, the aim is to define the quality of a subgroup sd in the database D. Let P be a set of split points, t_l and t_r two split points of P with the property $t_l < t_r$ and A a numerical attribute from D.Let sd and sd' be two subgroups of

lenght $\leq$ k, such that sd $\succ$ sd'$\wedge$ A $\in$ $]t_l$,$t_r]$. For every t' $\in$ $[t_l$,$t_r]$ the quality of sd on D is bounded by maxQ(D, k, sd'$\wedge$ A $\in$ $]t_l$,t'], P) + maxQ(D, k, sd'$\wedge$ A $\in$]t',$t_r]$, P) where maxQ(D, k, sd, P) := $\underset{sd^* \in refinements(sd,D,k,P)}{max}\{q(D, sd^*)\}$. The function refinements(sd, D, k, P) corresponds to the set of all refinements of sd with a length $\leq$ k and with interval endpoints in P.
In order to keep a trace of all bounds generated, a special data structure, denoted *BoundTables* is used. A BoundTables is a 2-D table and there is one for each attribute. The initial value of BoundTables[i, j] = 0 if i = j and ∞ in the other cases. When a value maxQ for the refinements of A $\in$ $]t_i, t_j]$ improves the former value, an update can be done for all super-intervals of A $\in$ $]t_i, t_j]$; it means for all A $\in$ $]t'_i, t'_j]$ such i' $\leq$ i and j' $\geq$ j.

1.3 Exceptional Model Mining

Multi-labeled model space

The existing works Like presented in [12], the main categories of works in learning from multi-label data are oriented around multi-label classification and label ranking. Multi-label classification looks for the creation of a model which is a bipartition between relevant and irrelevant labels in a given dataset. Label ranking algorithms build a hierarchy based on the relevance of labels according to a query instance. So, works focused on multi-label data mining are not adaptable to SD for two reasons :

- We do not consider an order relation between the labels.
- We do not want to ignore a partition of labels. All of them can be relevant.

Statistics on multi-label data Nevertheless, two interesting concepts are introduced in [12] : the label-cardinality and the label-density. Actually those two parameters have an influence on multi-label data mining performances. Let D be a dataset composed of n individuals and let $\mid l_i \mid$ be the occurence of a label l in the i-th individual. The label-cardinality is then defined as :

$$label - cardinality(l) = \frac{1}{m} \sum_{i=1}^{m} \mid l_i \mid$$

Considering all labels l are included in a set of labels L with a cardinality $\mid L \mid$, the label-density is defined as follow :

$$label - cardinality(l) = \frac{1}{m} \sum_{i=1}^{m} \frac{|l_i|}{|L|}$$

We have the elements to deal with the given database in the global framework. Actually the tools already developed around the subgroup discovery are enough efficient to answer our issue. Nevertheless, the multi-labeled model attribute have not been adressed in our lectures. Even if multi-labeled data can be mined for classifiers we do not have the algorithmic base to consider a multi-labeled model space. This is why an adaptation of the database is needed.

EMM framework Introduced recently in [7], Exceptional Model Mining (EMM) is a framework that generalized SD. Actually, until now we see the model space as a single attribute but with EMM we can consider a set of attributes for the model space. In EMM, we want to extract subgroups that have an exceptional model space configuration according to a given model. Nevertheless, the EMM framework is perfectly equivalent to the SD framework in terms of search space and subgroups construction. The element which differs is the quality measure. We can make a distinction between two kinds of EMM quality measures :

- Aggregation of qualities considering quality evaluation on each attribute included in the model space.
- Use of compression matrix methods in order to compare the gain between the compression based on a subgroup and the compression based on a given model considering their projections on the model space.

Considering that framework, we can turn our multi-labeled attribute into a set of single-labeled attribute and apply EMM.

1.4 Diverse Subgroup Set Discovery

Among recent works dealing with EMM, one is particularly interesting because of his flexibility. Actually, in [14], an approach able to handle numerical domains and managing the non-redundancy of the discovered subgroups is presented. The aim is to handle very large data with a generic subgroup selection heuristic. This heuristic is based on a beam search in order to balance time of execution and the result quality. Moreover, this algorithm allows the application of SD or EMM without distinction and provides many quality measures.

Non-redundancy management When data are large and complex, a basic top-k mining gives as result a set of potentially highly redundant subgroups. Actually, a top-k mining consists in a top-down search like explained in the previous part. The search starts with the empty set and a refinement operator, in the first step it builds the descriptions with only one feature (attribute-value pair). Then for a given k, the refinement the k-th best subgroups are selected. At the next level, the top-k list is uptated if there is better subgroup according with a quality measure. But results are potentially highly redundant due to the density of the data and less represented but interesting results are ignored. The evaluation of redundancy, presented in [14], relies on the covers of subgroup sets. Assuming that an ideal subgroup set uniformly cover all tuples of a dataset. The *cover count* of a tuple is the number of times it is covered by a subgroup in a subgroup set. Given a dataset S and a subgroup set G, the expected cover count $\hat{c}$ is defined for a random tuple t $\in$ S as :

$$\hat{c} = \frac{1}{|S|} \sum_{t \in S} c(t, G)$$

The cover count c is defined as follow :

$$c(t, G) = \sum_{g \in G} s_g(t)$$

So the cover redundancy is defined for a given dataset S and a subgroup set G as follow :

$$CR^S(G) = \frac{1}{|S|} \sum_{t \in S} \frac{|c(t,G) - \hat{c}|}{\hat{c}}$$

Non-redundant GSD The non-redundant generalised subgroup discovery (GSD) is the term used to highlight the fact that it covers SD and EMM. In the case of SD, two degrees of redundancy may exist :

- Subgroup descriptions redundancy : All substantially different descriptions are allowed but potential similarities in covers are ignored. Actually, it is possible that a subset of rows are covered by two different descriptions. The elimination of potentially redundant description works on the quality-ordered list of subgroup. A candidate is discarded if at least all the features (attribute-value pair) but one is equivalent to an other description already selected.
- Subgroup cover redundancy : All substantially different covers are allowed. This verification is satisfactory to ensure non-redundant SD. Compute a cover-based subgroup selection is longer than considering the descriptions but provide more diverse results. In order to compute the difference of cover between a subgroup and a set of already selected subgroups Sel, a score is calculated. This score is defined as follow for a subgroup G compared to a set of subgroups Sel :

  $$\Omega(G, Sel) = \frac{1}{|G|} \sum_{t \in G} \alpha^{c(t,Sel)}$$

  The weight parameter α is included in [0, 1]. The larger is the score, the less often tuples in G are already covered by subgroups in Sel. If G covers only tuples uncovered by subgroups included in Sel, $\Omega(G, Sel) = 1$. At each iteration, the subgroup which maximize $\Omega(G, Sel)$ is selected until the wanted number of subgroups is selected.

The DSSD algorithm The pseudo code of the DSSD algorithm presented in [14] is the following one :

Algorithm 3.1: DSSD diverse subgroup set discovery

Input : Dataset S, quality measure φ, parameters j, k, $mincov$, $maxdepth$ and subgroup selection parameters P
Output : R, an approximation of the top-k subgroups G_k

```
function DSSD(S, φ, j, k, mincov, maxdepth, P)
    R ⟵ ∅, Beam ⟵ {∅}, depth = 1
    while depth ≤ maxdepth do
        Cands ⟵ ∅
        for all b in Beam do
            Cands ⟵ Cands ∪ GenerateRefinements(b, mincov)
        end for
        for all c ∈ Cands do
            UpdateTopK(R, j, c,φ(c))
        end for
        Beam ⟵ SubgroupSelection(Cands, φ, P)
        depth ⟵ depth + 1
    end while
    for all r ∈ R do
        ApplyDominancePruning(r, φ)
    end for
    R ⟵ RemoveDuplicates(R)
    R ⟵ SubgroupSelection(R, φ, P)
    return R
end function
```

The exploration space can be seen as a tree where the root is the empty set and each level represents a new refinement step. A refinement consists in the add of one feature, for example $attribute_i < 3$, in order to build a new subgroup description as the conjunction of those features. At each refinement step, the quality measure is computed for all candidates (at the first step it means all possible features) and only the top subset of them is included in the beam according to a given size for the beam. Then each candidate is the beam is used to generate the next refinements with respect to a minimum cover, in number of tuples, $mincov$. The top-k list is updated if some of the new refinements overtake already selected subgroups according to the quality value. Selected subgroups are also analyze considering the potential redundancy considering a given parameter (one of the three kind of redundancy level described earlier). All these steps are repeated until a defined $maxdepth$ is reached.

Chapter 4

Experiments

In this part, we present the experiments made on the dataset given by neuroscientifics and described in the introduction. There are three main experiments :

- The characterization of olfactory qualities considering the entire set of qualities as model space
- The characterization of olfactory qualities considering each of them as model space
- The characterization of the hedonic number, or Hedo number, which is a subjective numerical value associated to a molecule. The greater is the Hedo number, the more pleasant is a molecule smell. This experiment requires an other dataset composed like the ARCTANDER dataset but instead of the associated olfactory qualities, the Hedo number is defined for each molecule.

For each of these experiments, we used the free implementation in C++ of the generic algorithm DSSD presented in 1.4. All post-treatments have been implemented externally in JAVA.

2 Characterization of olfactory qualities with EMM

Interest and objectives

The aim of this experience is to find the most interesting subgroups. So we proceed here to an EMM presented in 1.3. For the expert of the domain, it is relevant to find subgroups in that dataset because it could establish rules from physical and chemical properties to qualities and it could show strong correlations between subsets of qualities.

Dataset and experimental protocole

The dataset The considered dataset can be seen as a the join of the following three parts :

- A first row labeled Dragon_ID which only corresponds to the identifiant of each tuple.
- A collection of 1704 numeric attributes labeled $Attr_i$ which are physical and chemical properties of molecules and correspond to the description space.

– A multilabeled attribute labeled Code_quality which takes values in the powerset of qualities that describe a molecule. Each value is a non-empty subset with a variable size depending on the molecule. Each Code_quality value is in $[1; 74]^+$ and can be associated to more than one molecule. This attribute corresponds to the model space. The model space must be considered as nominal and not as numeric because there is no hierarchy between his values and each code is matched with a label like "vanilin" or "acid".

The main settings of DSSD algorithm we fixed are the following ones :

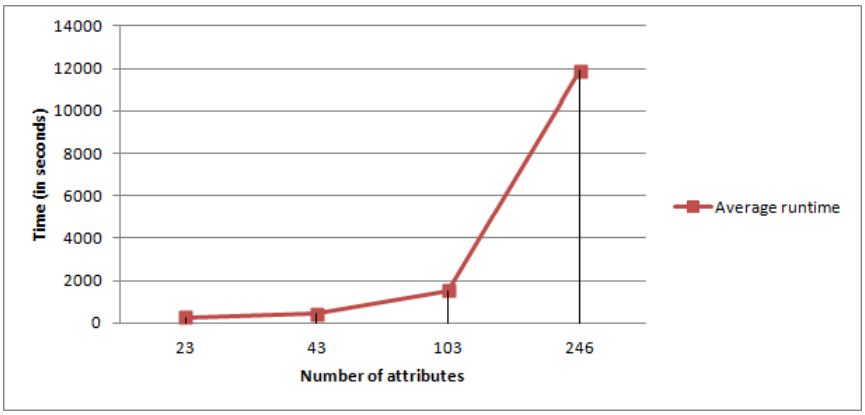
– The top 100 subgroup set is returned.
– The quality measure is WKL presented in 1.1.
– The beam width is 100.
– The maximum depth which corresponds to the maximum number of features in each subgroup description is 10.
– The minimum cover for every considered quality value is 10 tuples.
– Redundancy management is more efficient while using a subgroup cover-based redundancy management even if theoretically it should better while using compression.

Data transformations In order to represent in our relational database the multilabeled quality attribute, a table composed of the two columns Dragon_ID and Code_quality is created. The first column has possibly a same value many times in order to express that a molecule is related to multiple qualities. But each Dragon_ID/Code_Quality pair is unique. The most efficient way to transform those data in order to compute EMM is to binarize the model space. It means that for each quality value (a code i included in [1; 74]), a binary attribute $Quality_i$ is created and is true if the Dragon_ID is associated with the quality code i else false. There is also a selection among description space based on mathematical correlation between numerical attributes. This selection allows us to use only 243 numerical attributes as description space in order to reduce computation time. Actually, each attribute will be discretized on-the-fly and all attribute/discretized value pairs will be rated in the first step of the DSSD algorithm.

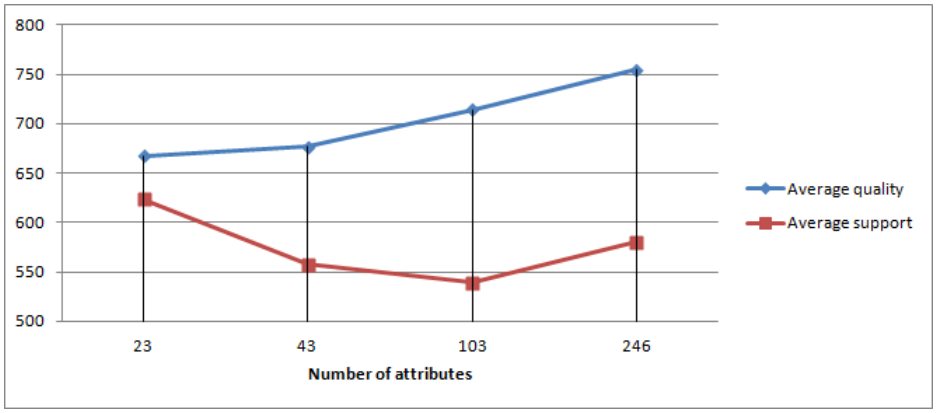
Parameters influences

In this section, we presents the influence of parameters on the free implementation of DSSD. Because most of them have already been experimented in [14], we present only the influence of attributes number in description space. For this, we prepared 4 selections of 23, 43, 103, and 243 attributes. Three attributes are mandatory because they are basic indicators of each molecule. The other attributes have been selected with the help of expert of the domain in order to extract 40 significant attributes (for the two smallest selections) and by the extraction of uncorrelated attributes for the biggest selections. We studied especially the variation of runtime, average subgroup quality, average subgroup descriptions length and redundancy.

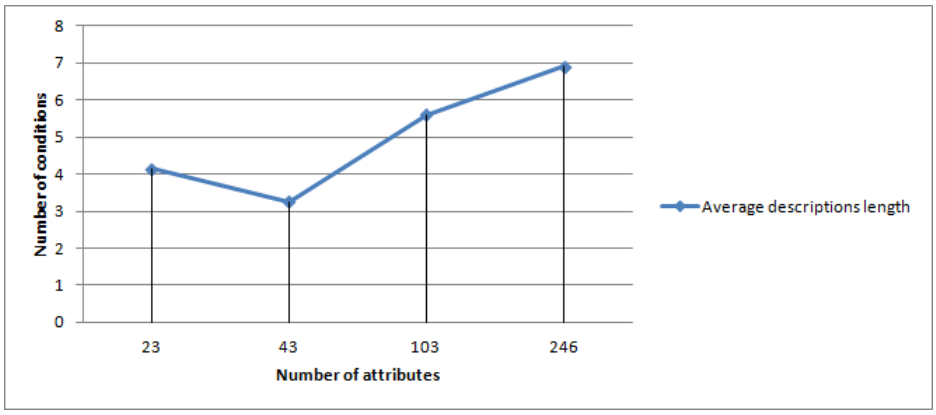
runtime evolution On the graphic below, we can see that the runtime increases with the number of attributes and follows an exponential variation. Considering the exponential evolution of the runtime, we can not test with the full attribute set included in the ARCTANDER dataset because it would take months of execution.



quality evolution The subgroup quality follows a variation close to a linear one. But we can notice that the average support of subgroups globally decreases and it is logic considering that the more attributes there are the more restrictive are the generated conditions. So it means that the positive variation of subgroup quality is due to the exceptionnality of model space configurations like presented in 1.1.



description length evolution Like shown by the figure below, the more attributes there in the description space, the longer are high quality subgroup descriptions.



redundancy We observed that non-redundant GSD, introduced in 3.1, can not provide only non redundant subgroups. But the greater is the description space, the less redundant are subgroups.

- For the first selection (23 attributes), the subgroups present redundancy on descriptions and logically on covers. This redundancy comes from the tight variety of distinct subgroup descriptions tested. Nevertheless, this redundancy concerns less than 10% of the top-100 subgroup set.
- For the three greater selections, there is some cover redundancy induced by maximum description length parameter. Actually, while considering a set of tuples described by a set of 43 to 243 attributes, we can easily guess that we can find a pattern among those attributes with a length greater than 10, which is our maximum description length parameter. This is why two distinct descriptions can cover exactly very close bag of tuples and are undetectable by the non-redundancy management.

Analysis of results

The DSSD algorithm provides as results the following elements at the end of the execution :

- The top 100 subgroup descriptions.
- The support of each subgroup in the top 100. The support is represented by a binary list : 1 if the row satisfies the current subgroup description else 0.
- The subgroup description size, the quality and the support size of each subgroup in the top 100.
- The number of rows, that we call $Occur^G_{q_i}$, covered by each quality q_i in the subgroup G support. Remember that many qualities can cover the same row but of course a quality can have an empty cover in a given subgroup support.
- The number of rows covered by each quality in the entire dataset.

These data defines subgroups according to the given parameters but they involve readability and correctness issues.

The limits

Actually, for an expert of the neuro-biologic domain, the results are not clear and do not appear reliable because a description is not explicitly associate to a subset of olfactory qualities. It is not correct to consider every quality q_i which satisfies $Occur_{q_i}^G > 0$ for a given subgroup G. The descriptions do not aim to isolate strictly a subset of qualities but to maximize the unusualness of qualities distribution even if it means including qualities that are not really specific in this subgroup. Moreover, the qualities do not have equal distributions in the entire dataset, some of them are over-represented (quality "floral" covers a quarter of the entire dataset) and many others are under-represented (only cover 10 to 50 rows in the entire dataset). Moreover, one aim of this experience is also to define correlation between qualities in the model space but the use of WKL implies that model space attributes are independant.

3 Characterization of olfactory qualities with SD

Interest and objectives

After the first experiment, which was the most intuitive considering the global objectives, we defined according to the experts of the domain that it is particularly interesting to find subgroups for each quality considered individually. It could highlight physical and chemical properties depending on the olfactory quality. Moreover, the applications of those kind of very deterministic rules are valuable from a chemical point of view. So, the aim is clearly different because we do not try to find subset of olfactory qualities but find significant descriptions for each of them.

Dataset and experimental protocole

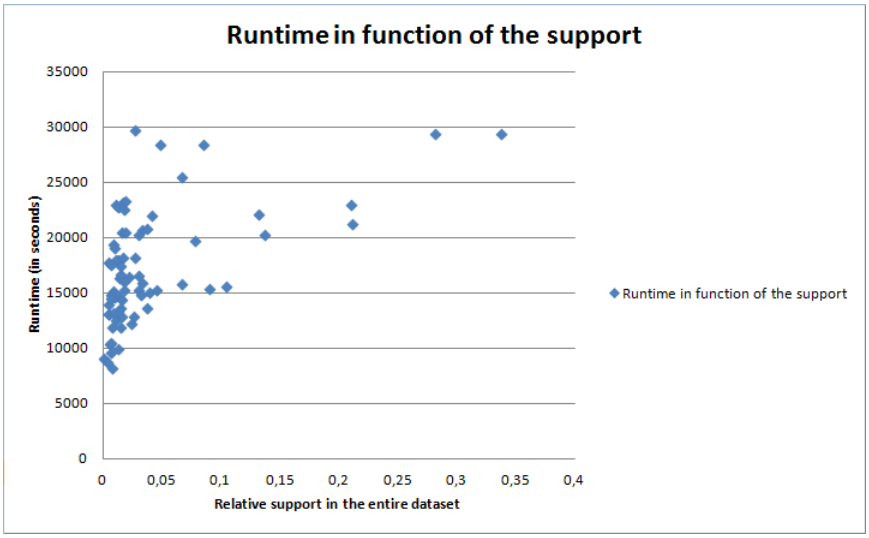
The datasets The input dataset is the same as considered previously but we derivated datasets for each quality. The settings are also the same because the WKL can be used on a single binary target without modifications.

Data transformations On the description space, the exact same selection is done. For the model space, we generated 74 datasets. Each dataset has as model space a single binary attribute which represents an olfactory quality. The model attribute equals true if the quality is associated to the molecule else false.

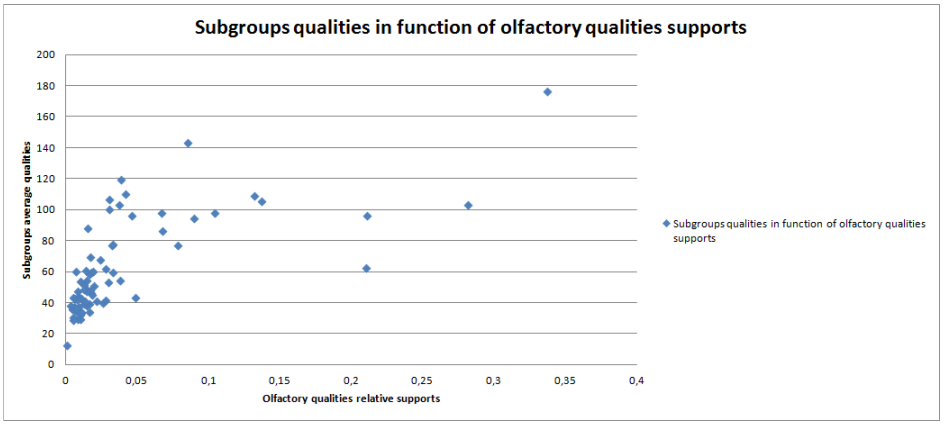
Parameters influence

In this experiment, the parameters are implicit because it is essentially the support of each olfactory quality which plays on the runtime and the average subgroups quality. We expose below this influence for the 74 olfactory qualities included in the ARCTANDER dataset.

runtime evolution Each point represents the runtime for an olfactory quality represented on the X axis by her relative support in the entire dataset. We can observe on the graphic below that the runtime follows globally a logarithmic increase considering the olfactory qualities supports in the entire dataset. It is important to highlight the fact that in this experiment, multiple DSSD executions have been done simultaneously. So the values can not be compared with the former experiment.



subgroup quality evolution Same representation than earlier for the olfactory qualities. Subgroups qualities follow a linear increase considering olfactory qualities supports. It is totally logical considering the WKL formula presented in 1.1.



Analysis of results

Like in the former experiment, the DSSD algorithm provides the same results but instead of each $Occur_{q_i}^G$ and number of rows covered by each quality, it provides the number of true cases included in each subgroup and the number of true cases in the entire dataset, considering the projection of subgroups and the dataset on the binary model space. The results are easier to understand considering the former experiment. Of course, for each olfactory quality, subgroups with high quality are the better.

The limits

Considering the expectations of the olfaction experts, finding subgroups with their qualities and supports is not satisfying. Actually, among a subgroup set for a given olfactory quality, quality values are useful to compare subgroups but are useless while comparing with a subgroup set for an other olfactory quality. It simply comes from the fact than olfactory qualities distribution is not balanced and quality measures relies on this distribution. So, in order to determine which olfactory qualities are the best defined by their respective subgroup set, we will be more interested the average confidence of the association rules $subgroupdescription \longrightarrow olfactory\ quality$ of all subgroups included in a subgroup set.

4 Characterization of Hedo number with SD

Interest and objectives

The Hedonic number, or Hedo number, is a value that rates a molecule on an olfactory point of view. This value is based on the opinion of a sample of population. This experiment has two interests : on one hand, it is very relevant for the experts of the domain to determine descriptions which lead to pleasant or stinking smells. On the other hand, it is interesting for us to perform subgroup discovery on a numeric model space in order to test the limits of MeanTest and determine how we could improve the results.

Dataset and experimental protocole

The dataset The dataset considered in this experiment composed in two parts. The first part is a description space composed of 357 uncorrelated attributes that describe each molecule. The Hedo value is a real in [0.0; 0.8], in our dataset, the greater is the Hedo value, the more pleasant the quality smells. In accordance with expert, we defined stinking molecules like all molecules with a Hedo number under 0.3 and pleasant molecules like ones with a Hedo number greater than 0.6. All molecules between those values are considered neutral.

Parameters The main settings of DSSD algorithm are the following ones for each variant of this experiment:

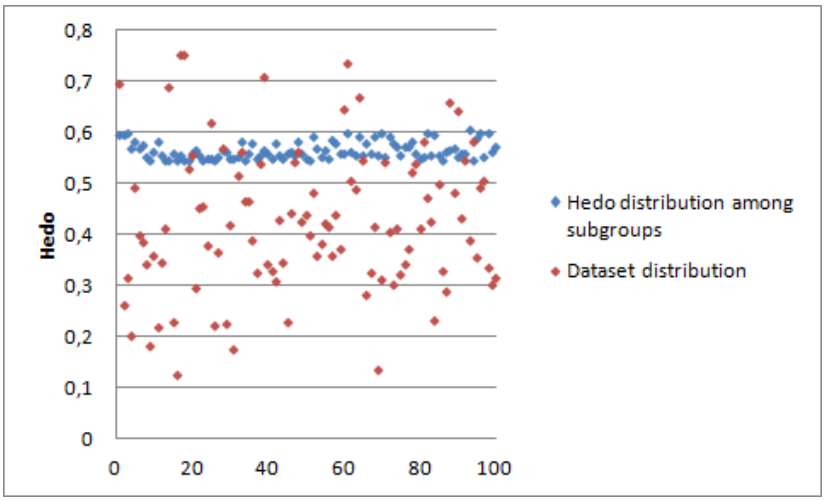
- The top 100 subgroup set is returned.
- The quality measure is the MeanTest presented in 1.1.
- The beam width is 100.
- The maximum depth which corresponds to the maximum number of features in each subgroup description is 10.
- The minimum cover for every considered quality value is 10 tuples.
- Redundancy management is more efficient while using a subgroup cover-based redundancy management.

Analysis of results

The DSSD algorithm provides as results the following elements at the end of the execution :

- The top 100 subgroup descriptions.
- The mean of Hedo numbers included in each subgroup.
- The subgroup description size, the quality and the support size of each subgroup in the top 100.
- The mean of Hedo numbers for the entire dataset.

The first result that appears is that we define only a very tight range of Hedo number so the subgroups does not fulfill the aim to describe pleasant and stinking molecules.



Considering the global Hedo distribution in the dataset, we can guess that there are characterization for stinking and pleasant molecules.

The limits

The limitation is clearly, in this experiment, the ability to extract characterizations for extreme values considering a domain. It comes from the distribution which is not fair enough. Actually, almost 80% of molecules are neutral. So an other approach is needed to reach the objectives.

5 DSSD limits

Through these experiments, we have already extracted subgroups and analyze them in function of some parameters. But we see also that we are not able to answer properly to every questions raised by the Olfamning project for several reasons :

- While considering the multiple attribute model space, we are unable to determine the correct subset of olfactory qualities described significantly by a subgroup description. Moreover, we miss the property which makes the distinction between olfactory qualities significantly described or not by a description.
- With the same model space, we are unable to highlight correlations between attributes included in the model space, which is an aim of this experiment. It is induced directly by the quality measure which is WKL so we need to build a method to discover those hypothetic correlations.
- Until now, the divergence of the model space is based, in EMM, of the mean of each model attribute divergence. But if we want to focus on a subset of model attributes, we must reconsider the WKL.
- In order to describe olfactory qualities, we need to know which ones are described with an important accuracy. It means to determine for each olfactory quality and for each subgroup the confidence of the association rule $subgroupdescription \longrightarrow olfactory\ quality$. With those confidences we can then compare the global quality, or accuracy, of each subgroup set. We need to formalize the computation of confidence from subgroups.
- Finally, concerning Hedo charaterization, we need to reconsider the MeanTest in order to find descriptions for pleasant and stinking molecules. If it does not bring satisfying results we should consider other options.

Chapter 5

Contributions

6 Multi-labeled model attribute management

6.1 Definition of model attribute value

The aim in this first contribution is to define the right subset of model attributes while using EMM with WKL as quality measure. Actually, an attribute can be considered specific for a description if the confidence of the rule *description* $\longrightarrow$ *attribute* is important. This is why we choose to use the *GrowthRate* in order to determine if a model attribute is specific to a subgroup or not. It has the advantage to ignore the difference of true cases distribution between model attributes. Let G be a subgroup, $\bar{G}$ be the entire dataset less G support and $\mid G \mid$ and $\mid \bar{G} \mid$ be the respective support sizes of G and $\bar{G}$, the GrowthRate of a model attribute ma is defined as follow :

$$GrowthRate(ma) = \frac{Occur^{G}_{ma}}{|G|} * \frac{|\bar{G}|}{Occur^{\bar{G}}_{ma}}$$

The *GrowthRate* simply determines the ratio between the probability that the attribute is true in the subgroup and the same probability in the entire dataset.

6.2 Model attributes correlation

While computing EMM, each attribute in the model space is considered independant of the others like explained in 1.1. Even if there is no hierarchy in the model space, some model attributes may be correlated, it means that one model attribute is true only when an other is true too. In order to extract these informations, we need to compute the intersections of model attributes supports in a considered subgroup. The computation of these intersections is in three phases :

- First, the support of each model attribute represented by a subgroup is extracted on the entire dataset.
- Then, the intersections $I^{G}_{ma_i}$ between the subgroup G and each represented model attribute ma_i is computed.
- Finally, the intersections between the $I^{G}_{ma_i}$ and $I^{G}_{ma_j}$ is computed for all represented model attributes ma_i and ma_j. If the intersection between $I^{G}_{ma_i}$ and $I^{G}_{ma_j}$ covers more than a given threshold β considering $I^{G}_{ma_i}$ or $I^{G}_{ma_j}$ then the model attributes ma_i and ma_j are considered correlated. It means that G represents significantly the subset $\{ma_i, ma_j\}$ and not the model attributes separetely.

6.3 Post-treatment application

Considering the definition of α and β, we can apply it to the first experiment as post treatment. After the computation of this post-treatment, which has an execution time really smaller than the DSSD algorithm, we have readable and accurated results. The thresholds α and β are respectively 5 and 80 and we find descriptions for 34 qualities on 74 but all uncorrelated. For example, our post-treatment gives the following results :

Description :
Me5 $>$ 0.00000 and SpAD_A $<$ 1.66100 and PHI $<$ 20.19150 and Se0 $<$ 0.17150
and Sv05 $>$ 6.76050 and Se95 $<$ 3.37800 and BLI $<$ 2.90350 and Se19 $<$ 0.73200
It defines significantly the following qualities :
Quality almond significantly represented with respective growthrate 7.661071428571429
Quality anisic significantly represented with respective growthrate 5.5441964285714285
Quality aromatic significantly represented with respective growthrate 5.040178571428571
Quality hay significantly represented with respective growthrate 9.072321428571428
Quality leathery significantly represented with respective growthrate 20.160714285714285
Quality medicinal significantly represented with respective growthrate 14.515714285714285
Quality peperry significantly represented with respective growthrate 12.096428571428572
Quality phenolic significantly represented with respective growthrate 22.176785714285714
Quality piney significantly represented with respective growthrate 6.336224489795918
Quality smoky significantly represented with respective growthrate Infinity
Quality tarry significantly represented with respective growthrate 42.3375
Quality tobacco significantly represented with respective growthrate 5.376190476190476
Quality vanilin significantly represented with respective growthrate 6.720238095238095

This case is interesting because it represents all the possible results we can have after the post-treatment. The values are at least α and at most Infinity which means that the entire support of a quality is included in a given subgroup support. At least one quality is significantly represented according to α and some of them may be correlated according to β. But considering that a lower value of β means less accurate results concerning the correlation, we consider now the adaptation of WKL for multi-label attributes.

6.4 Generalization of WKL

Until now we have considered the divergence of the model space like the mean of all model attribute divergences as presented in 1.1. But depending on the model space, it is not relevant to consider the global divergence as the sum of each model attribute divergence. This is why we suggest to consider an abstract aggregator function ϑ. This function takes the set of all model attribute divergences and computes a global divergence that respects a maximal number of model attribute to consider k.

$$WKL(G^M) = \frac{|G|}{|S|}\vartheta^k(KL(\hat{P}(G^{M_i}) \parallel \hat{P}(S^{M_i}))), \forall i < k$$

We will consider the following ϑ values : Σ, max and min. The function $\max^k$ means that we consider only k-th most divergent model attributes to compute global divergence, $\min^k$ means we consider the k-th less divergent model attributes and Σ^k means we consider all model attributes. Let m be the cardinality of the model space, the classic WKL can be defined as :

$$WKL(G^M) = \frac{|G|}{|S|} \Sigma^m (KL(\hat{P}(G^{M_i}) \parallel \hat{P}(S^{M_i}))), \forall i < m$$

In the case we consider that each model attribute is a value of a single multi-label model attribute, it is more interesting to consider the global divergence of the model space as the sum of the most divergent model attributes. The question is then how to define the number of most divergent model attributes to consider. This is where we can make the link with multi-label data mining. Considering that label cardinality, presented in 1.3, is the average number of labels associated to an individual, we consider the closest integer greater than the label-cardinality. So, let k be the closest integer greater than the label cardinality of a dataset S, we define WKL as follow :

$$WKL(G^M) = \frac{|G|}{|S|} \max^k (KL(\hat{P}(G^{M_i}) \parallel \hat{P}(S^{M_i}))), \forall i < k$$

6.5 Application of generalized WKL

In the ARCTANDER dataset, the label-cardinality is 2.88 so we will consider the three most divergent model attribute to compute the WKL. With the same parameters and similar rutimes, we can observe that the results are really different. First, we only characterize 11 olfactory qualities on 74 and only 5 of them are characterized with WKL definition presented in [13]. Then, we have many correlated attributes with the same β. Here is an example of result :

- Description :
  Se59 > 30.93650 and Mv4 < 4.50300 and Se19 < 0.76700 and IDET < 11.63700 and Sv4 < 6.00000 and S0K > 0.55900 and Se05 < 0.10400
  It defines significantly the following qualities :
  Quality peperry and Quality woody significantly represented with respective growthrates 13.193277310924369 and 7.380854439677969
  Quality sandalwood and Quality woody significantly represented with respective growthrates 65.96638655462185 and 7.380854439677969
  Quality camphor significantly represented with respective growthrate 8.57563025210084
  Quality violet significantly represented with respective growthrate 6.5966386554621845

Even if there are less olfactory qualities found with this WKL computation, the results are more accurate and expose some correlations among labels. It is also important to notice that subgroups supports are 3 to 5 times smaller than subgroups extracted with the classic WKL definition. In order to validate this approach in this case we also apply the same protocole with the two following variants :

$$WKL(G^M) = \frac{|G|}{|S|} \max^1(KL(\hat{P}(G^{M_i}) \parallel \hat{P}(S^{M_i}))), \forall i < 1$$

$$WKL(G^M) = \frac{|G|}{|S|} \min^1(KL(\hat{P}(G^{M_i}) \parallel \hat{P}(S^{M_i}))), \forall i < 1$$

It consists to consider respectively the most divergent model attribute and the less divergent attribute as global model space divergence. The results are interesting :

- While considering only the most divergent attribute we extract the same subgroups descriptions extracted while considering the 3 most divergent attributes. The difference is in growthrates. Actually, when we consider only the most divergent model attribute, subgroups define often olfactory qualities with infinite qualities but there is less olfactory qualities defined by a description. It validates the fact that considering only the k-th most divergent attributes lead to better results.
- This hypothesis is definitely confirmed while considering the less divergent attribute. Even if 21 olfactory qualities are defined, growthrates are smaller than earlier and there is an important frequency of redundant subgroups.

7 From subgroups to association rules

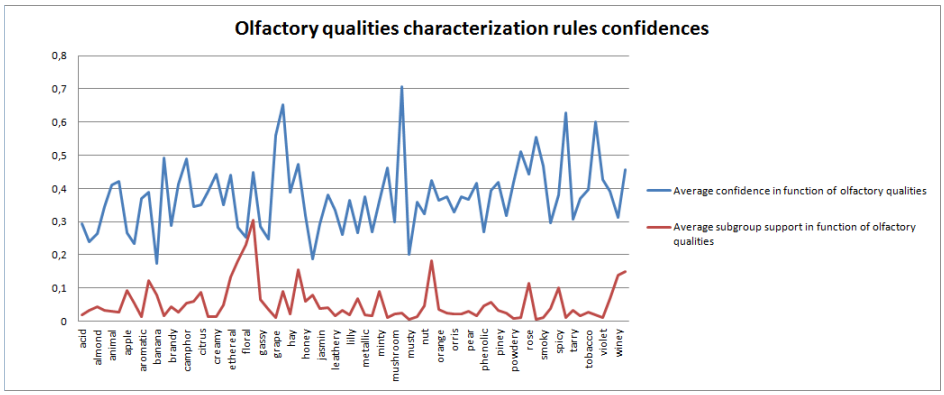
7.1 Confidence of descriptions

The aim is to define with a minimum of additionnal calculs, the confidence of the association rule $subgroupdescription \longrightarrow model\ attribute = value$, where the description is simply a given subgroup description. Let S be a dataset, G a subgroup extracted from S, t a tuple included in S and t^M the projection of t on the model space. In the case of a single n-ary attribute, we can compute that confidence as follow :

$$confidence(G) = \frac{|\forall t \in S, t^M = value \wedge t \in G|}{|S|}$$

7.2 Application

We apply this transformation to the second experiment in order to define the accuracy of subgroups. On the following graph, we highlight the non-correlation between subgroup set quality measure values and subgroup set quality. By quality we mean the average confidence of the association rule $subgroupdescription \longrightarrow model\ attribute = value$ for each subgroup included in a subgroup set for a given olfactory quality.



The greater is the confidence value, the better is the subgroup set in order to define a given olfactory quality. We notice that the greatest values are for weakly represented olfactory qualities in the ARCTANDER dataset.

8 Weighted quality measures adaptation

8.1 Management of subgroup size in Meantest

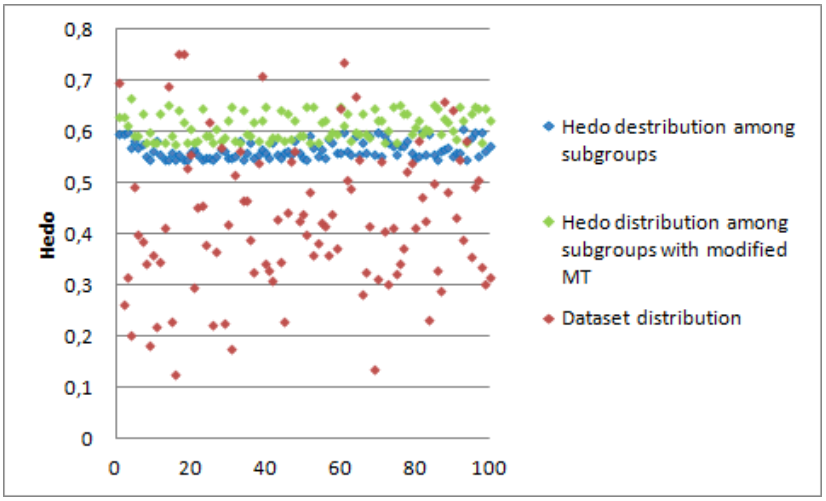
Weighted quality measures have the advantage to highlight the greatest subgroups with high priority but in many cases, interesting unusual subgroups are really small considering their relative supports in a given dataset. This is why we want to reconsider the weighted component of MeanTest. For this we now consider the Meantest quality measure for a subgroup G as :

$$MT(G) =\mid G \mid^{\frac{1}{n}} (\mid \mu^G - \mu^S \mid)$$

The value n is determined in order subgroups supports have the smaller impact on global qualities but they still make a difference between major unusual subgroups and the others. The ideal value provides values of $\mid G \mid^{\frac{1}{n}}$ very close to $\mid \mu^G - \mu^S \mid$ for a given subgroup set.

8.2 Application of modified MeanTest

First, we applied the modification of the MeanTest on the third experiment presented in 4. The best value for n is 3 because a greater value induced too small values. Actually, we had a really uninteresting distribution of Hedo number among subgroups. With the new MeanTest we obtain the following distribution with similar parameters :



As we can see the distribution is better because a bit more balanced on the entire domain of possible values but it remains unsatisfying according to the objectives.

8.3 Characterization of extreme values with WKL

In this part, we consider a single numeric attribute as model space. The problem shown in experiments is that Meantest is very sensitive to values distribution on a given domain. If we want a characterization of the entire domain, the distribution must be very balanced. But in many cases, the distribution is not balanced and subgroups define values closed to the mean of all values in the domain. The use of a discretization turns the numeric attribute and we can use the WKL.

8.4 Application of WKL for extreme values characterization

We can directly apply the discretization and use WKL for the third experiment in order to find characterizations for stinking and pleasant. The following results are extracted after the dataset transformation and the confidence of the rule $description \longrightarrow pleasant/stinking$ has been computed :

- stinking with confidence 0.733 with the description Col3624 < 1.00000 and P_VSA_MR_5 < 28.39950 and MATS4p > -0.24850 and Col3133 < 2.00000 and MATS3m < 0.01950 and GATS3e < 1.14400 and ATSC1s < 8.46600
- pleasant with confidence 1.0 with the description GGI5 > 0.03500 and P_VSA_LogP_2 < 7.60400 and GATS5m < 1.18550 and Eig08_AEA(bo) < 0.50000 and X3Av < 0.17000 and Eig10_AEA(dm) > -1.92250 and GATS3e < 1.44450 and SpMin8_Bh(s) < 0.05750 and Psi_i_0d > -0.01400

Chapter 6

Conclusion

By the use of existing methods, we were able to fulfill, from an algoritmic point of view, the objectives list proposed by the project Olfamining. We have defined the algorithmic context with precision and study incrementally the tools which exactly answer our problems. Some methods and algorithms have not been used because they show limits for our case that are already kwnown. The most interesting approach to answer our problem is subgroup discovery and her generalization to multiple targets, exceptional model mining. Actually, the formalization of a subgroup is the closest from the formalization we made in the presentation of the introduction. Moreover, many tools already studied are used and some works on subgroup discovery directly answer the problem of knowledge discovery in numerical domains. In order to handle the target multi-labeled attribute, we choosed to adapt this attribute to a set of binary attributes and we can mine this set with EMM. Among subgroup discovery and exceptional model mining works, we have chosen one of the most generic algorithm, DSSD, in order to face different type of model spaces. The results with the implementation are understandable by an expert of data mining but present real problems of readability and could lead to wrong interpretations for the experts of the data domain. This is why we proposed post treatments in order to highlight or find meaningfull informations in the data. We also reconsidered quality measures when they do not fit well to our case and tried different data adaptation strategies in order to find the best results. We can then provide readable, correct and in some case adaptable results in function of post treatment parameters.

References

[1] Rakesh Agrawal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, A Inkeri Verkamo, et al. Fast discovery of association rules. *Advances in knowledge discovery and data mining*, 12(1):307–328, 1996.

[2] T. V. Getchell. Functional properties of vertebrate olfactory receptor neurons. *Physiological Reviews*, 66(3):772–818, 1986.

[3] Henrik Grosskreutz and Stefan Rüping. On subgroup discovery in numerical domains. *Data Min. Knowl. Discov.*, 19(2):210–226, 2009.

[4] Francisco Herrera, Cristóbal J. Carmona, Pedro González, and María José del Jesús. An overview on subgroup discovery: foundations and applications. *Knowl. Inf. Syst.*, 29(3):495–525, 2011.

[5] Willi Klösgen. Explora: A multipattern and multistrategy discovery assistant. In *Advances in Knowledge Discovery and Data Mining*, pages 249–271. 1996.

[6] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 03 1951.

[7] Dennis Leman, Ad Feelders, and Arno J. Knobbe. Exceptional model mining. In Walter Daelemans, Bart Goethals, and Katharina Morik, editors, *ECML/PKDD (2)*, volume 5212 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2008.

[8] Heikki Mannila and Hannu Toivonen. Levelwise search and borders of theories in knowledge discovery. *Data Mining and Knowledge Discovery*, 1(3):241–258, 1997.

[9] Petra Kralj Novak, Nada Lavrac, and Geoffrey I. Webb. Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining. *Journal of Machine Learning Research*, 10:377–403, 2009.

[10] Schiffman Susan S. *Characterization of Odor Quality Utilizing Multidimensional Scaling Techniques*, chapter 2, pages 1–21.

[11] Gilles Sicard, Maurice Chastrette, and Nicolas Godinot. Des représentations de l'espace olfactif: des récepteurs à la perception. *Intellectica*, 24:85–107, 1997.

[12] Grigorios Tsoumakas, Ioannis Katakis, and Ioannis P. Vlahavas. Mining multi-label data. In Oded Maimon and Lior Rokach, editors, *Data Mining and Knowledge Discovery Handbook*, pages 667–685. Springer, 2010.

[13] Matthijs van Leeuwen. Maximal exceptions with minimal descriptions. *Data Min. Knowl. Discov.*, 21(2):259–276, 2010.

[14] Matthijs van Leeuwen and Arno J. Knobbe. Diverse subgroup set discovery. *Data Min. Knowl. Discov.*, 25(2):208–242, 2012.

[15] Jilles Vreeken, Matthijs van Leeuwen, and Arno Siebes. Krimp: mining itemsets that compress. *Data Min. Knowl. Discov.*, 23(1):169–214, 2011.

[16] Paul M. Wise, Mats J. Olsson, and William S. Cain. Quantification of odor quality. *Chemical Senses*, 25(4):429–443, 2000.

[17] Stefan Wrobel. An algorithm for multi-relational discovery of subgroups. In Henryk Jan Komorowski and Jan M. Zytkow, editors, *PKDD*, volume 1263 of *Lecture Notes in Computer Science*, pages 78–87. Springer, 1997.