Interpreting Classifiers by Multiple Views

Stefan Rüping STEFAN.RUEPING@UNI-DORTMUND.DE
Universität Dortmund, LS Informatik 8, 44221 Dortmund, Germany

Abstract

Next to prediction accuracy, interpretability is one of the fundamental performance criteria for machine learning. While high accuracy learners have intensively been explored, interpretability still poses a difficult problem. To combine accuracy and interpretability, this paper introduces an framework which combines an approximative model with a severely restricted number of features with a more complex high-accuracy model, where the latter model is used only locally. Three approaches to this learning problem, based on classification, clustering, and the conditional information bottleneck method are compared.

1. Introduction

More and more data is collected in all kinds of application domains and sizes of data sets available for knowledge discovery increase steadily. On the one hand this is good, because learning with high-dimensional data and complex dependencies needs a large number of examples to obtain accurate results. On the other hand, there are several learning problems which cannot be thoroughly solved by simply applying a standard learning algorithm. While the accuracy of the learner typically increases with example size, other criteria are negatively affected by too much examples, for example interpretability, speed in learning and application of a model, overhead for handling large amounts of data and the ability to interactively let the user work with the learning system (Giraud-Carrier, 1998). This paper deals with the criterion of interpretability of the learned model, which is an important, yet often overlooked aspect for applying machine learning algorithms to real-world tasks. The importance of interpretability stems from the fact that knowledge discovery is not equal to the application of a learning algorithm, but is an iterative and interactive process that requires much manual work from data miners and domain specialists to understand the problem, transform the data prior to learning and interpret and deploy the model afterwards. One cannot hope to successfully solve these problems without substantial insight into the workings and results of the learning algorithm.

The rest of the paper is organized as follows: the next section discusses the concept of interpretability, its relation to multiple views, and introduces the basic ideas of this paper. Section 3 gives an introduction to the information bottleneck method, which will be used later. Section 4 describes the problem of learning local models and its connection to learning with multiple views, while three approaches to the crucial step of detecting local patterns are presented in Section 5. Following that, Section 6 gives some empirical results and Section 7 concludes.

2. Interpretability

The key problem with interpretability is that humans are very limited in the level of complexity they can intuitively understand. Psychological research has established the fact that humans can simultaneously deal with only about seven cognitive entities (Miller, 1956) and are seriously limited in estimating the degree of relatedness of more than two variables (Jennings et al., 1982). An optimal solution of a high-dimensional, large-scale learning task, however, may lead to a very large level of complexity in the optimal solution. Interpretability is very hard to formalize, as it is a subjective concept. In this paper, we use four heuristics to approach the concept of interpretability:

Number of features: the number of features used in a model is a heuristic measure of complexity. While this is not strictly true, as the user may understand even a high number of features if he can relate the feature values to an existing mental concept (e. g., a doctor may explain a large number of symptoms by one disease), this heuristic has

Appearing in *Proceedings of the Workshop on Learning with Multiple Views*, 22^{nd} ICML, Bonn, Germany, 2005.

often been used in practice (Sommer, 1996).

User-defined hypothesis space: A very simple and yet very important finding in practice is that people tend to find those things understandable that they already know. Hence, if a user has much experience with a specific learning algorithm, it may be favorable to keep using this learner, regardless of its accuracy.

Examples and features instead of models: While models are often hard to understand even for experienced machine learning experts, single examples and single features have a clear meaning to domain experts (e. g. instead of a TF/IDF representation of text documents, one can look at the texts themselves).

Split-up into sub-problems: Splitting up a problem into several independent sub-problems reduces the complexity and may still give reasonable results, even if the sub-problems are actually not completely independent.

How can the interpretability problem be solved? Experience shows that one can often find a simple model which provides not an optimal solution, but a reasonably good approximation. The hard work usually lies in improving an already good model. Hence, we can try to find a simple model first and then concentrate on finding more sophisticated models only on those parts of the input space, where the model is not good enough. This will be an easier task because less examples have to be considered and hence one might use a more sophisticated learner. To express the fact that the latter models are only used for small parts of the input space, these models will be called local models. In contrast, the former, more general model will be called the global model (Rüping, 2005).

Implicitly, this setup requires a description of the parts of the input space where the global model is not good enough or a decision rule when to use the global model or the local models. These regions will be called local patterns (Hand, 2002) and we will require the description of the local patterns to be interpretable in the same sense as the global model. The idea, as depicted in Figure 1, is the following: to classify an observation with high accuracy, we see whether it falls into one of the local patterns. In this case, the corresponding local model is used, else the global model is used. When we are more interested in interpretability, it suffices to inspect only the global model, which is an approximation of the complete model, plus the local pattern which characterizes the deviations between the global and the complete model.

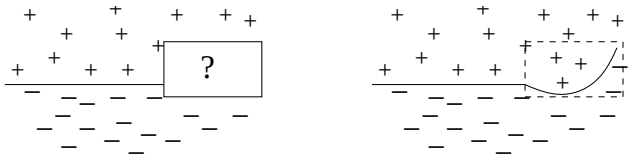

Figure 1. The local model idea. Left: with respect to interpretability, only the global model (horizontal line) is regarded, while the local pattern (rectangle) specifies the region where the global model is not reliable. Right: with respect to accuracy, the pattern specifies when to use the local model (nonlinear function) instead of the global model.

The local patterns distinguish this approach from both ensemble methods and from independently learning an understandable and a high-performance model. In usual ensemble methods, even in the case of easily interpretable base learners the combination of several learners will add a large amount of complexity, whereas using local patterns the model will be kept simple for a large and well-defined part of the input space. In contrast to two independently learned models, the local pattern assures that there is a strict, well-defined correspondence between the two models. The philosophy behind this approach is that accuracy and interpretability are two aspects of the same problem and that their solutions should be as independent as necessary, but as close as possible.

This approach reduces complexity in two ways. First, a less than optimal hypothesis language can be used for the global model, because errors can still be corrected by the local models. This leaves room for choosing a hypothesis language that optimizes criteria other than the prediction error, namely the interpretability of the global model. Second, for the aspect of discovering new knowledge, it may happen that the global model finds only the obvious patterns in the data that domain experts are already aware of. Patterns are more informative, if they contradict what is already known (Guyon et al., 1996). Hence, it may in fact be the case that the local models contain the interesting cases.

Of course, there is also a downside to this approach: The complete model, consisting of three sub-models, may easily be much more complex than a single model designed to optimize accuracy. However, it should be noticed that only the global model and the local patterns are meant to be understandable and that a lower complexity of a competing model yields no improvement if it is still too complex for the user to understand.

In this paper, the interpretability framework consists of restricting the number of features that both the learner and the description of the error regions of the learner may use. This can be seen as constructing two new views on the data: one to define the global model on, and one to discriminate between the global and the local models.

3. Information Bottleneck

The information bottleneck method (Tishby et al., 1999) extracts structure from the data by viewing structure extraction as data compression while conserving relevant information. With the data modeled by a random variable U[1], relevant information is explicitely modeled by a second random variable V, such that there is no need to implicitly model the relevant structure in terms of appropriately choosing distance or similarity measures as in standard clustering algorithm. The idea is to construct a probabilistic clustering, given by a random variable C, such that the mutual information $I(U, C)$ between the data and the clusters is minimized, i. e. C compresses the data as much as possible, while at the same time the mutual information $I(V, C)$ of the relevant variable V and the clusters is maximized, i. e. the relevant structure is conserved. Hence, the random variable C acts as a bottleneck for the information U has about V. Both goals are balanced against each other by a real parameter $\beta > 0$, such that the goal becomes to find a clustering $P(c|u)$ which minimizes

$$F = I(U, C) - \beta I(V, C).$$

It can be shown that this problem can be solved by iterating between the following three equations

$$\begin{aligned} P(c) &= \sum_u P(u)P(c|u) \\ P(v|c) &= \sum_u P(v|u)P(u|c) \\ P(c|u) &\propto P(c)e^{\beta P(v|u)\log P(v|c)}. \end{aligned}$$

The first two equations ensure the consistency of the estimate probabilities, while the third equation gives a functional form of the clustering, depending on the Kullback-Leibler-distance between $P(v|u)$ and $P(v|c)$ (removing factors independent of c). The input consists of the probability distributions $P(u)$ and $P(v|u)$.

[1]We use the letters U, V, W instead of the usual X, Y, Z in order to avoid confusion with the classification features X and labels Y used later

3.1. Condition Information Bottleneck

Gondek and Hoffmann (Gondek & Hofmann, 2004) extend the information bottleneck approach by considering not only information about relevant, but also about irrelevant structure. It is often easier to express what is already known and hence is uninteresting, than to specify what is interesting and relevant. Examples of such irrelevant structures are general categorization schemes and well-known properties of the data, when instead one is interested in how the data differs from what one thinks it looks like.

Conditional information bottleneck (CIB) is formulated by introducing a random variable W to describe the irrelevant information. The learning problem corresponds to that of standard information bottleneck with the new target function

$$F = I(U, C) - \beta I(V, C|W)$$

That is, one wants to maximize the information that C has of V, given that W is already known. In a way, the goal is to extract information orthogonal to what one can already infer via W.

Again, the problem can be solved by iterating between three estimation equations

$$\begin{aligned} P(c) &= \sum_u P(u)P(c|u) \\ P(v|w, c) &= \sum_u P(v|u, w)P(u|w, c) \\ P(c|u) &\propto P(c)e^{\beta \sum_w P(w|u) \sum_y P(v|u,w) \log P(v|w,c)} \end{aligned}$$

The probabilities $P(v|u, w)$, $P(w|u)$ and $P(u)$ have to be given to the learner as input.

4. Local Models and Multiple Views

Local pattern or subgroup detection (Hand, 2002) is defined as the un-supervised detection of high-density regions in the data. That is, one looks for regions in the input space, whose empirical probability with respect to a given a training set is significantly higher than the probability assigned by a default probability measure, which encodes the prior beliefs about the data. The idea is that the user already has some idea of what his data looks like and is interested in cases where his beliefs are wrong.

Local models are an extension of local patterns to the supervised case. Local models are meant to improve the prediction, hence instead of $P(x)$ the interesting quantity is the conditional class probability $P(y|x)$.

We will deal with classification rules only here. Given a global classifier $f(x)$, the goal is to find regions of the input space where $f(x)$ is wrong, plus a new, better classification rule on these regions. To justify the term *local* the error regions are restricted to have a probability below a user-defined threshold τ, such that most of the data will be predicted by the global model.

In order to improve interpretability with local models, we want both the global classifier and the description of the local regions to be interpretable. As discussed in Section 2, this implies that the user may choose any learner that he deems to be appropriate. Treating the learner as a black box, we improve understandability only by restricting the number of features for the global classifier and the local pattern detection can use. In other words, we construct a specific view for the global classifier which is optimized for interpretability of the overall model and a second view for the local patterns, which is optimized for describing the incorrectly predicted examples of the global model. Finally, the local classifier is not restricted in which features to use, as it is not meant to be understandable, but only to increase accuracy.

4.1. Optimizing the Global and Local Models

Selecting a subset of features for the global model is a well investigated problem and can be solved in a general way for example using the wrapper approach (Kohavi & John, 1998). This approach is computer intensive, but can be used for any learner.

The local learner is not restricted by any interpretability considerations at all and we may select the learner which gives the best accuracy. The definition of the local models asks only for the local model to be defined on its corresponding local pattern. Hence, we may use different learners for each pattern, which may be a good idea when there are different reasons that the data deviates from the global model. This means that the detection of local patterns and the construction of models depend on each other and that it may be a good idea to construct them in parallel or let them iteratively improve each other (Rüping, 2005).

However, as in the following we are mainly concerned with the problem of finding adequate local patterns, we simplify things by using only one local model on all local patterns. This model will be learned on all available examples (that is, it is actually a global model), but will be used only on the detected local patterns. This model is expected to perform better than the actual global model because it is not restricted by interpretability constraints.

5. Detecting Local Patterns

Given the global and local models, the goal of local pattern detection in this case is to find a description of the regions in the input space where the local model is better than the global one. Examples lying in these regions will be called local examples here. The goal of local pattern detection here is to optimize the combined learners accuracy while keeping the restriction of interpretability (hypothesis language and number of features) and locality (only a fraction of τ examples in the local patterns).

5.1. Local Patterns as a Classification Task

It is straightforward to define local pattern detection as a classification task: given the examples $(x_i, y_i)_{i=1}^n$ and the global and local learners predictions, define the new label l_i as 1 when (x_i, y_i) is a local example (meaning that the global learner predicts (x_i, y_i) wrong and the local learner is right). Set $l_i = -1$ otherwise. Then learn a classifier using the new labels. This classifier will predict whether the local model should be used on an example instead of the global one.

When the global and local learner agree, it obviously does not matter which prediction one uses. However, this does not mean that these examples can be removed from the local pattern learners training set. As the locality restriction requires that only a fraction of τ examples may lie in the local patterns, it is advisable to include only the local examples in the positive class, where the combined model can be improved by the local model. If the locality restriction is still not met, one will have to reduce the number positive predictions of the local pattern learner even more, e.g. by selecting a proper threshold for learners with a numerical decision function.

For the decision, which classifier to use for the local pattern task, the same interpretability considerations as for the global model apply. In fact, as may be advisable to use the same learner for both tasks. Letting the learner choose different sets of features and a different hypothesis in both tasks may provide enough variation to significantly improve the results.

5.2. Clustering Local Examples

Although in the strict sense detecting local patterns is a classification task in the framework of local models, it can also be solved by a clustering approach. The reason is, that the classification task is actually a quite relaxed one: given that the local model is more accurate than the global one and as long as the locality restriction is fulfilled, it is no problem to include more

examples in the local pattern than necessary. Hence, in this case the performance criterion for the classification is biased very much towards completely covering the positive class (recall) with less emphasis on the negative class.

This task may be solved by clustering the local examples using a density-based clusterer and choosing a threshold on the density to define the regions with highest probability of finding a local example. It is straightforward to optimize these thresholds such that the accuracy of the combined model is maximized.

A clustering approach may be better suited as a classification approach, as clustering not only tries to describe the differences between the local and the non-local examples, but actually tries to find a compact representation of the local examples. This description may give the user a clue why these examples are more complex to classify and may lead to an improved representation of the data.

One can also account for the probabilistic nature of the division of the examples into local and non-local examples. Assume that the training data $(x_i, y_i)_{i=1\ldots n}$ is i. i. d. sampled from $P_{orig}(X \times Y)$[2]. We start by learning a probabilistic classifier f, that is, a classifier whose outputs can be interpreted as the conditional class probability $f(x) = P_{orig}(Y = 1|x)$. Many classifiers either directly give such a probability or give outputs which can be appropriately scaled (Garczarek, 2002; Platt, 1999). We assume that this is the true distribution of positive and negative classes given the observation x and thus arrive at an estimate of $P_{orig}(Y \neq f(x)|x)$. Assuming $P_{orig}(x) = 1/n$ for all x in the training set gives an estimate of

$$P(x) := P_{orig}(x|Y \neq f(x)) = \frac{P_{orig}(Y \neq f(x)|x)}{\sum_x P_{orig}(Y \neq f(x)|x)}$$

the probability of drawing an falsely classified observation. When generating a cluster model, this probability can be used as a weight on how much each example will have to be represented by the cluster.

To enforce the restriction on the number of features used for the model, one can either use a clustering algorithm that incorporates feature reduction by selecting a subset of features that maximizes the density of the projection in this subspace (projected clustering, (Aggarwal et al., 1999)), or by selecting features after the clustering process, for example based on the mutual information $I(x, c)$ of the features x and the cluster membership values c.

[2]In the following, the index *orig* is also used to identify the marginal distributions derived from P_{orig}

5.3. Informed Clustering for Local Patterns

Informed clustering describes the setting where the desired structure to be extracted by a clustering is not only defined implicitly using the distance or similarity function, but also explicit information about relevant and irrelevant structure is given. The conditional information bottleneck algorithm is one such approach, where one can explicitely define irrelevant structure which the clustering algorithm should ignore.

There are two kinds of irrelevant information one could exploit. First, one can use a probabilistic clustering $p(c|x)$ of the complete training observations $(x_i)_{i=1\ldots n}$. This clustering shows, what the data generally looks like and can be used as a background model to discriminate the local examples against. Note that we do not require an information bottleneck clustering at this stage, we could also use any other probabilistic clusterer. It would also be possible to use an existing description of the data set at this stage (e. g. an ontology given by the user).

The other method is to define the prediction of the global model or, more precisely the conditional class probability $P_{orig}(Y = 1|x)$ as irrelevant. The idea here is that the global model is used anyway and that it is better to look for independent sources of information.

In either case, one can arrives at a well-defined probability $P_{cib}(w|u)$. Now one can set up the conditional information bottleneck problem to find the local patterns as follows: identify U with the index i and the relevant features V with the classification features X:

- $P_{cib}(u) = P(x_i) = P_{orig}(x_i|Y \neq f(x_i))$
- $\forall w : P_{cib}(v|u, w) = P_{ib}(v|u)$

In other words, the problem is to compute a probabilistic clustering $P(c|u) = P(c|x_i)$ of the observations x_i which are misclassified by the classifier f (controlled by $P_{cib}(u) = P_{orig}(x_i|Y \neq f(x_i))$), such that the clustering describes how the local examples differ from the complete data set (via defining the cluster information $P_{ib}(v|u)$ of the complete data set as irrelevant) or how the local examples differ from structure from the global model (via $P_{orig}(Y = 1|x)$).

As the information bottleneck method does not return a cluster model, but only the cluster membership probabilities $p(c|x)$, a model that induces these memberships has to be found in order to apply the clustering to new observations. Following the goal of interpretability, it is advantageous to use a k-medoids clustering model, as it is often easier to interpret single examples than models. For each cluster, we choose

that example as medoid, which – in the space of projections on the most relevant features – minimizes the expected distance of the medoid to the examples in the cluster, where expectation is taken with respect to the probability $P_{cib}(x, c)$ for the cluster c.

6. Experiments

In this section, we report results for both an instructive application for classifying music data, which we report in depth, and for a set of standard data sets in order to compare the different local pattern approaches.

6.1. Music Data

In these experiments, a linear Support Vector Machine (Vapnik, 1998) was used as both global and local classifier. Feature selection for the global classifier was performed by repeatedly removing the features with lowest absolute weight in the decision function. The SVM decision functions were probabilistically scaled using the method of (Platt, 1999). The pattern detection based on the CIB method (see Section 5.3) was used, where the initial clustering was obtained by standard information bottleneck and a k-medoids model of the CIB membership values was generated with the cosine similarity measure.

The data set in this experiment consists of 1885 audio files of songs from 8 music genres, combined with user reviews of each of the song as plain text. The classification target was to predict the music taste of a user. From the music files, 50 audio features were generated following the methodology of (Mierswa & Morik, 2005). The text information was represented as frequencies of the 500 most frequent words. The global classifier was learned from the text data only, as it is easy for users to interpret single keywords, while audio features are hard to understand even for experienced experts. The local classifier was learned on the union of the audio and text features. Notice that in this application we have four different views on the data: the keywords from the classification, the keywords from the clustering, the union of the audio and text features for the local classifier and finally the actual songs from the audio files, which the user can actually listen to.

The initial information bottleneck clustering was parameterized to return 8 cluster, in correspondence with the 8 music genres, and its parameter β was set to maximize the correspondence to the genres. However, the most informative words regarding the clustering were HELLO, POWER, BLEND, SOUNDS, BABY, FAT, QUIET, BIT, NIGHT, and GIVE, which do not seem to reveal any obvious genre structure.

Feature selection for classification returned GROOVE, SMOOTH, CHILL, JAZZY, MOOD, FUSION, PIANO, PIECE, PAUL, and JAZZ as the most important features. It is obvious that a certain music taste can be associated with these keywords

The CIB clustering returned TALENT, BABY, SOUNDS, CHECK, NEAT, PASS, TRUE, NICE, SEXY, and CHORUS as the most important features. Interestingly, extracting two medoids from this clustering showed that the first medoid consists only of the word CHORUS with no occurrence of the other keywords and the second medoid consists of the words SOUNDS and NICE with no occurrence of the other keywords. This is a result of the sparse structure of the text data, as a medoid as any other example will have only a few nonzero features. For sparse data it may be instructive to try out a different procedure to describe the CIB clusters. However, the second medoid with the keywords "sounds nice" seems to indicate that there are two aspects to musical taste in this data set, the genre – which the initial clustering was optimized against – (the classifier indicates that the user seems to like jazz) – and the quality of the music independent of the style (whether the song sounds nice or not).

5-fold cross-validation showed an accuracy of the global model of 0.624 ($\sigma = 0.0284$), while the local model achieved an accuracy of 0.670 ($\sigma = 0.0164$) measured over all examples. The combined model achieves an accuracy of 0.649 ($\sigma = 0.0230$). This lies between the accuracies of the global and the local model, which was expected, as the amount of examples that the global and the combined differ on is bounded by the parameter τ (in this experiment, $\tau = 0.05$), which stops the local model from correcting more errors of the global model.

To validate that the increase in performance is indeed a result of the conditional information bottleneck approach, the experiment was repeated with a standard information bottleneck clustering of the global models errors instead of the CIB step (all other parameters left constant). With the same accuracies for the global and local classifiers, the accuracy of the combined classifier dropped to 0.627 ($\sigma = 0.0329$). This proves that the conditional information bottleneck clustering finds novel structure in the errors of the global classifier.

To validate the effect of the parameter τ and the number of features for the CIB clustering, more experiments were conducted. The result can be seen in Table 1. The table shows the accuracies of the global, local and combined models and the disagreement rate (fraction of examples classified differently) between the global and the combined model. We can see that the

Table 1. Influence of the parameters on the performance.

PARAMETERS		ACCURACY			DISAGREE
τ	#FEATURES	GLOBAL	LOCAL	COMBINED	
0.05	10	0.624	0.670	0.648	0.147
0.05	20	0.624	0.670	0.653	0.201
0.025	10	0.646	0.670	0.642	0.070
0.025	20	0.646	0.670	0.646	0.019

combined model performs better when more features for the CIB clustering are present. We also see that the actual disagreement rate is higher than the given threshold τ. This is again a result of the sparse nature of the data, as in the space projected on the most important keywords, several different examples fall together, which prevents a more fine grained control of the number of local examples. An obvious tradeoff between interpretability in terms of number of features and accuracy can be observed here.

6.2. Standard Data Sets

To compare the local pattern approaches, the classification approach using a linear SVM, the clustering approach using an information bottleneck clusterer, and the informed clustering approach using conditional information bottleneck with the global classifiers conditional class probability estimate as irrelevant information were compared on a total of 8 data sets. 6 of the data sets (diabetes, digits, liver, balance, wine and breast-cancer) were taken from the UCI repository of machine learning databases (Murphy & Aha, 1994), and 2 additional real world data sets involving business cycle prediction (business) and intensive care patient monitoring (medicine) were used. The following table sums up the description of the data sets:

Name	Size	Dimension
balance	576	4
breast	683	9
diabetes	768	8
digits	776	64
liver	345	6
wine	178	13
business	157	13
medicine	6610	18

In these experiments, a linear Support Vector Machine (Vapnik, 1998) was used as both global and local classifier. Feature selection for the global classifier was performed by repeatedly removing the features with lowest absolute weight from the decision function until only 10% of the features were left.

Table 2 shows the experiments results, namely the accuracy of the global model, the local model (viewed as a complex global classifier and evaluated on the complete data set) and the combined models using classification, clustering and informed clustering for the local patterns. All results were 5-fold cross-validated. The performance of the combined model lies between the performances of the global and local models, which shows that local models can improve classification performance even under severe restrictions on the number of features (depending on the dimension of the data set, in some case only 1 feature is used). It can, however, not beat a more complex classifier in this case, which is a result of both the locality restriction stopping the local model from classifying more observations and the dimensionality restriction allowing only very coarse local patterns.

Overall, the clustering approach seems to be slightly better than the other, but differences are very small. This may be a sign that the local pattern detection task is essentially limited by the allowed size and complexity of the patterns in terms of number of features and not by the algorithm.

7. Conclusions

Next to accuracy, interpretability is a primary quality criterion for classification rules. Traditionally, these goals have been pursued independent of another. This paper showed that using the framework of local models, it is possible to find a model which combines not only good performance with an easily interpretable approximation, but, even more important, allows to give guarantees about the correspondence of the approximation and the combined classifier in terms of the disagreement rate threshold τ and in terms of an interpretable description of the error regions.

In the proposed algorithm, clustering proves to be an efficient tool for not only discriminating between the global and the local model, but also for describing the difference of both models in terms of a compact representation of the structure in the global models errors.

Table 2. Experimental Results.

Name	Global	Local	combined		
			SVM	IB	CIB
balance	0.644	0.940	0.727	0.909	0.788
breast	0.907	0.969	0.931	0.956	0.951
diabetes	0.760	0.781	0.701	0.748	0.763
digits	0.993	0.996	0.994	0.993	0.994
liver	0.579	0.695	0.635	0.634	0.631
wine	0.927	0.971	0.927	0.943	0.932
business	0.828	0.866	0.828	0.834	0.821
medicine	0.719	0.744	0.749	0.748	0.756

References

Aggarwal, C. C., Procopiuc, C., Wolf, J. L., Yu, P. S., & Park, J. S. (1999). Fast algorithms for projected clustering. *Proceedings of the ACM SIGMOD Conference on Management of Data* (pp. 61–72).

Garczarek, U. (2002). *Classification rules in standardized partition spaces*. Doctoral dissertation, Universität Dortmund.

Giraud-Carrier, C. (1998). Beyond predictive accuracy: What? *ECML'98 Workshop Notes - Upgrading Learning to the Meta-Level: Model Selection and Data Transformation* (pp. 78–85). Technical University of Chemnitz.

Gondek, D., & Hofmann, T. (2003). Conditional information bottleneck clustering. *Proceedings of the 3rd IEEE International Conference on Data Mining, Workshop on Clustering Large Data Sets*.

Gondek, D., & Hofmann, T. (2004). Non-redundant data clustering. *Proceedings ot the 4th IEEE International Conference on Data Mining*.

Guyon, I., Matic, N., & Vapnik, V. (1996). Discovering informative patterns and data cleaning. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy (Eds.), *Advances in knowledge discovery and data mining*, chapter 2, 181–204. Menlo Park, California: AAAI Press/The MIT Press.

Hand, D. (2002). Pattern detection and discovery. In D. Hand, N. Adams and R. Bolton (Eds.), *Pattern detection and discovery*. Springer.

Jennings, D., Amabile, T., & Ros, L. (1982). Informal covariation assessments: Data-based versus theory-based judgements. In D. Kahnemann, P. Slovic and A. Tversky (Eds.), *Judgement under uncertainty: Heuristics and biases*, 211 – 230. Cambridge: Cambridge University Press.

Kohavi, R., & John, G. H. (1998). The wrapper approach. In H. Liu and H. Motoda (Eds.), *Feature extraction, construction, and selection: A data mining perspective*, 33–50. Kluwer.

Mierswa, I., & Morik, K. (2005). Automatic feature extraction for classifying audio data. *Machine Learning Journal*, *58*, 127–149.

Miller, G. (1956). The magical number seven, plus or minus two: Some limits to our capacity for processing information. *Psychol Rev*, *63*, 81 – 97.

Murphy, P. M., & Aha, D. W. (1994). UCI repository of machine learning databases.

Platt, J. (1999). *Advances in large margin classifiers*, chapter Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. MIT Press.

Rüping, S. (2005). Classification with local models. In K. Morik, J.-F. Boulicaut and A. Siebes (Eds.), *Proceedings of the dagstuhl workshop on detecting local patterns*, Lecture Notes in Computer Science. Springer. to appear.

Sommer, E. (1996). *Theory restructering: A perspective on design & maintenance of Knowledge Based Systems*. Doctoral dissertation, University of Dortmund.

Tishby, N., Pereira, F., & Bialek, W. (1999). The information bottleneck method. *Proceedings of the 37-th Annual Allerton Conference on Communication, Control and Computing* (pp. 368–377).

Vapnik, V. (1998). *Statistical learning theory*. Chichester, GB: Wiley.