Chapter 7
Evolutionary Algorithms Explaining Support Vector Learning

All truths are easy to understand once they are discovered; the point is to discover them.
Galileo Galilei

7.1 Goals of This Chapter

Even if SVMs are one of the most reliable classifiers for real-world tasks when it comes to accurate prediction, their weak point still lies in the opacity behind their resulting discrimination [Huysmans et al, 2006]. As we have mentioned before, there are many available implementations that offer the possibility to also extract the coefficients of the decision hyperplane (SVM light, LIBSVM). In Chap. 6 we have also presented an easy and flexible alternative means to achieve that. Nevertheless, such output merely provides a weighted formula for the importance of each and every attribute. We have shown that feature selection can extract only those parameters that are actually determinant of the class and solve the issue of redundancy. However, the lack of any particular guidelines of the logic behind the decision making process still remains. This is obviously theoretically desired for a rigorous conceptual behavior, however it is also crucial for domains like medicine, where a good prediction accuracy alone is no longer sufficient for a true decision support for the medical act. While accuracy certainly remains a prerequisite [Belciug and El-Darzi, 2010], [Belciug and Gorunescu, 2013], [Gorunescu and Belciug, 2014] supplementary information on how a verdict had been reached, based on the given medical indicators, is necessary if the computational model is to be fully trusted as a second opinion.

On the other hand, classifiers that are able to derive prototypes of learning are transparent but cannot outperform kernel-based methodologies like the SVMs. The idea to combine two such opposites then sprung in the machine learning community: kernel techniques could bring the prediction force by simulating learning, while transparent classifiers could interpret their results in a comprehensible fashion.

There are many attempts in this sense, and several namely concerning the two subjects of this book: SVMs and EAs. In this context, the last chapter puts forward another novel approach built with the same target. We begin by addressing the existing literature entries (Sect. 7.2), present the new combination (Sect. 7.3) and enhancements (Sect. 7.5, 7.6 and 7.7), all from the experimental perspective (Sect. 7.4).

C. Stoean and R. Stoean, *Support Vector Machines and Evolutionary Algorithms for Classification*, Intelligent Systems Reference Library 69,
DOI: 10.1007/978-3-319-06941-8_7,

7.2 Support Vector Learning and Information Extraction Classifiers

A combination between a SVM and an explanatory classifier can be constructed on two grounds (see Fig. 7.1) [Martens et al, 2007]:

Pedagogical: SVMs establish a new input-output mapping, that is, each sample is labeled with the class predicted by the SVM. The relabeled samples are subsequently used by the information extractor. In other words, the SVM is actually a noise remover, which enables the following decision information extractor to concentrate learning only on correctly labeled data.

Decompositional: SVMs output the support vectors and the second method derives structured explanations from these. The support vectors are in fact the most important examples from the data set, as they shape the decision boundary. The approach also solves the runtime problem for the usual very large data sets connected to real-world problems. Therefore, this triggers both sample selection and noise removal prior to mining underlying rules.

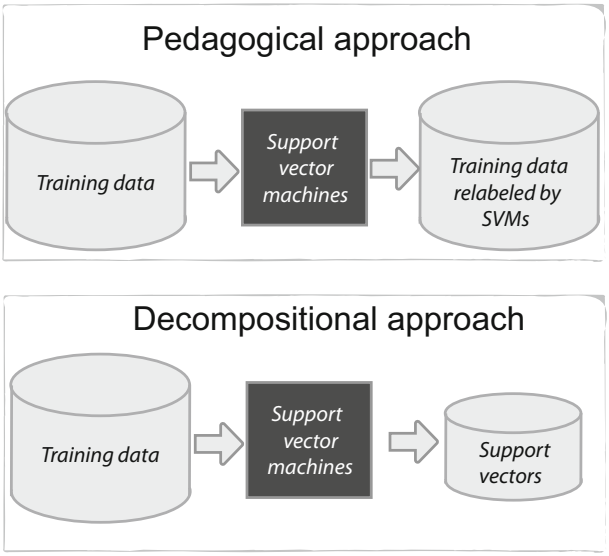

Fig. 7.1 Within the pedagogical approach, SVMs learn from the training set and then the classifier is applied to the same data to relabel it. The decompositional alternative simply extracts a small amount of the training data that represents the support vectors and these are kept with their original labels.

In order to meet its purpose, there are three consequential conditions that the final output of such a combined methodology must obey [Huysmans et al, 2006]:

Accuracy: The predicted targets for previously unseen samples must be more accurate than those derived from the information extractor alone.
Fidelity: Its results must approximate those of the black box SVM.
Comprehensibility: It must offer more comprehensible information than that of the initial learner, whichever form this knowledge may take.

We will next review the current such models that came to our attention. The first list involves different transparent knowledge extraction engines following a trained SVM model [Martens et al, 2007], [Diederich, 2008], [Farquad et al, 2010]:

- The SVM + Prototype approach of [Núñez et al, 2002] defines an ellipsoid through the combination of support vectors and data cluster prototypes to create an if-then decision scheme. The method suffers however from bad scalability.
- In [Fung et al, 2005], the problem is transformed to a simpler, equivalent variant and rules are constructed as hyper cubes by solving linear programs. This is not an advantage, as it can only be applied for linear decision kernels, which are generally not applicable for real-world data sets. Moreover, such an approach loses the strong ability of SVMs to model nonlinearities.
- In [Barakat and Diederich, 2005], the SVM relabeled input-output data are given to decision trees (DT) for the detection of the underlying learning system, while in the study [Barakat and Bradley, 2006] the area under the receiver operation characteristic curve is employed towards the same goal.
- In [Martens et al, 2009], an active learning-based approach is used to extract rules from support vectors.
- Finally, in [Farquad et al, 2010], the support vectors together with the actual output values of their targets are taken and provided to a fuzzy rule based system.

In the papers from the second list below, EAs are used in different formulations to collect the logic (mainly) behind neural networks (NNs) and SVMs. If we regard information extraction from the pedagogical point of view, then it makes no difference if we use SVMs, NNs [Haykin, 1999], [Gorunescu et al, 2011] or any other opaque classifier. That is the reason why we have included extraction from NNs in this list. A second motive is that, of all the combinations between opaque predictors and information extractors, hybridizations between SVM and EAs have been the least often explored.

- The GEX approach [Markowska-Kaczmar and Chumieja, 2004] learns from NNs, uses a special encoding for evolving rules and appoints an island model [Bessaou et al, 2000] to allow the existence of multiple subpopulations, each connected to a label of the problem to be solved. The disadvantage of this technique is that one sample can be covered by multiple rules, while it is not guaranteed that at least one rule will be valid for each class [Huysmans et al, 2006]. A changed EA, with a more elaborate representation for individuals and a Pareto multiobjective optimization behind, is provided later in [Markowska-Kaczmar and Wnuk-Lipinski, 2004].

- The G-REX alternative [Johansson et al, 2010] is a more accurate general technique that uses genetic programming [Langdon and Poli, 2001] to extract rules of various representations from different (opaque or not) models (NNs, random forests). The approach in [Martens et al, 2007] applies the G-REX method to SVMs instead of NNs.
- The methodology in [Ozbakir et al, 2009] achieves a combination between NNs and ant colony optimization [Dorigo and Stützle, 2004], [Pintea, 2014] for the same task.

7.3 Extracting Class Prototypes from Support Vector Machines by Cooperative Coevolution

Within these premises, we can formulate a novel combined method by appointing the CC algorithm (in Chap. 5) to discover the class prototypes after the data set had been processed by the SVM.

7.3.1 Formulation

The construction of the hybridized method under discussion [Stoean and Stoean, 2013a], [Stoean and Stoean, 2013b] is intuitively illustrated in Fig. 7.2. The flow of the technique can be therefore formulated in short as follows:

1. The SVM reshapes the data
   a. either in a pedagogical fashion
   b. or in a decompositional way.
2. The CC is trained on these changed data sets and determines attribute thresholds (as described before in Chap. 5).
3. The resulting attribute thresholding for each class must be
   a. accurate to new samples,
   b. faithful to the opaque model,
   c. as simple and compact as possible, since an intricate and hard to follow architecture may actually offer less comprehensibility.

7.3.2 Scope and Relevance

There are several advantages arising from this new approach for extracting the prediction hidden observations of SVM (as compared to related attempts in the literature):

- EA individuals can directly encode thresholds for problem indicators. Comprehensibility can thus be successfully achieved by generating prototypes for each class of the task, while they are also easily maneuvered by the EA.

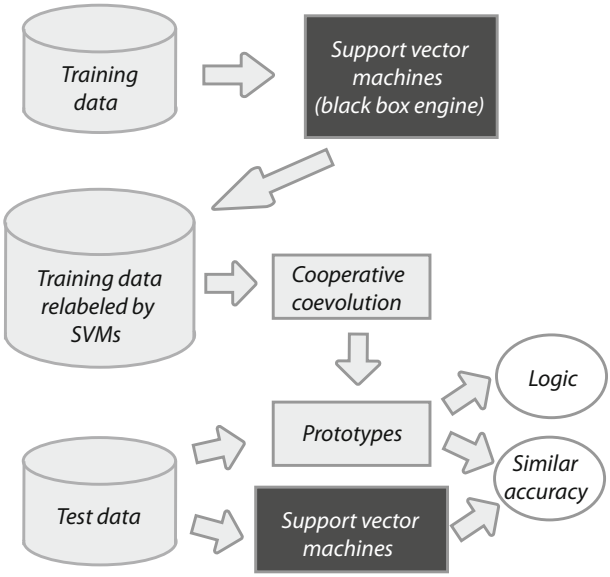

Fig. 7.2 SVMs are used to *clean* the given samples, CC is then trained on the new data set and creates prototypes that are used to better classify the test data. Additionally, CC provides explanations via the prototypes for each class as regards the logic behind the decision making process.

- The IF-THEN format holding conjunctive statements with equal signs for referring thresholds is also simpler to follow. Several inequalities or a complex format (like in [Markowska-Kaczmar and Chumieja, 2004], [Markowska-Kaczmar and Wnuk-Lipinski, 2004] or [Johansson et al, 2010]) cannot but harden the reading of the decision explanations.
- The extraction CC engine, as an EA, is an adaptable framework to implement different possibilities of resolving the task, as previously seen in its application results for classification in Chap. 5.
- As concerns the diversity of resulting class prototypes, CC inherently maintains several distinct concurrent subpopulations, as the number of classes of the problem determines the number of species. This multimodal mechanism is thus more straightforward when evolving distinct prototypes for the different classes of the decision problem (unlike the island model in [Markowska-Kaczmar and Chumieja, 2004]), as each class triggers one population. Thus, prototypes of every class eventually become homogenous, but they remain very different from those of the other species. This is also another reason why CC was preferred as the multimodal engine, instead of the alternative GC (in Chap. 4).
- The encoding is thus even simpler than the usual rule formation, since the class is not part of the prototype, resulting directly from the subpopulation it is connected to.

7.3.3 Particularities of the Cooperative Coevolutionary Classifier for Information Extraction

As before in the CC approach (Chap. 5), each individual (prototype or rule) encodes values for all indicators in the data and its class is given by the population it belongs to. Hence, its formal expression is referred again as (4.3), where the prototype is representative of class y_i of the problem, $i = 1,2,...,k$. The individuals of the starting population are once more randomly initialized, where the value for each of the n attributes is generated following a uniform distribution between the definition bounds of that specific feature. The condition part of an individual then specifies indicator thresholds that designate it as a prototype for the class defined by its population.

The prediction capability of a class prototype is computed after a complete set is formed by selecting one individual from each of the other subpopulations. In the experiments that follow in this chapter, we use a random selection of the individuals from the subpopulations, but different options for the collaborator selection pressure parameter could be considered. The entire prototype collection is then applied to the training data as remodeled by the SVM. For every training data sample, distances to each collected prototype are calculated and the individual that is closest decides its label. The performance of the initial prototype is then given by the prediction accuracy over all training samples.

While comprehensibility is thus primarily resolved through the EA individual representation, the two requirements regarding fidelity to the SVM and high prediction accuracy are met through reference through the CC fitness expression described in the lines above. The actual place of inclusion is when success is measured by comparing the outcome of a sample with the SVM-CC prediction. If the approach is pedagogical, then the actual outcomes for the training data examples are those confirmed by the SVMs. Fidelity is thus addressed as in (7.1) and expresses the percentage of identically labeled samples [Huysmans et al, 2006]. x_i is a sample, y_i^{SVM} is its outcome as predicted by the SVM and y_i^{SVM-CC} that which is provided by SVM-CC, $i = 1,2,...,m$.

$$fidelity^{SVM-CC} = Prob(y_i^{SVM} = y_i^{SVM-CC} | x_i \in [a_1,b_1] \times [a_2,b_2] \times ... \times [a_n,b_n]) \tag{7.1}$$

If the behavior is decompositional, the real outcomes of the support vectors are those given in the initial training data set. Accuracy [Huysmans et al, 2006] is therefore also obeyed as in (7.2).

$$accuracy^{SVM-CC} = Prob(y_i^{real} = y_i^{SVM-CC} | x_i \in [a_1,b_1] \times [a_2,b_2] \times ... \times [a_n,b_n]) \tag{7.2}$$

Finally, as regards the test stage, the corresponding samples are classified by a set of prototypes appointed from the final subpopulations and their predicted outcomes are confronted with those present in the original data set.

The approach is sketched by Algorithm 7.1.

Algorithm 7.1 CC for extracting learning prototypes from SVMs.

Require: A k-class classification problem
Ensure: A rule set with multiple prototypes for each class
begin
if approach is pedagogical **then**
Relabel labels of training data as predicted by the SVM;
else
Collect the support vectors with their actual labels from the data;
end if
$t \leftarrow 0$;
for each species i **do**
Randomly initialize population $P_i(t)$;
end for
for each species i **do**
if approach is pedagogical **then**
Evaluate $P_i(t)$ by selecting collaborators from the other species for every individual and compare classes according to fidelity;
else
Evaluate $P_i(t)$ by selecting collaborators from the other species for every individual and compare classes according to accuracy;
end if
end for
while termination condition is not satisfied **do**
for each species i **do**
Select parents from $P_i(t)$;
Apply genetic operators;
if approach is pedagogical **then**
Evaluate $P_i(t)$ by selecting collaborators from the other species for every individual and compare classes according to fidelity;
else
Evaluate $P_i(t)$ by selecting collaborators from the other species for every individual and compare classes according to accuracy;
end if
Select survivors from $P_i(t)$ to $P_i(t + 1)$;
end for
$t \leftarrow t + 1$;
end while
return a complete set of prototypes for each class
end

7.4 Experimental Results

We want to assess three goals in order to prove the effectiveness of the proposed combined SVM-CC approach for white box extraction:

- fidelity to SVMs;
- accuracy superior to CC;
- comprehensibility superior to SVMs.

The first two aims test the viability of the hybridization, such that the SVM-CC performs better than the CC and comparable to the SVM. Thirdly, SVM-CC must also provide the class prototypes underlying the decision model in a form that must be understandable for the reader. Only after having those validated, we can safely affirm that the methodology accomplishes the theoretical goal of such a combination of classifiers and offers credible practical assistance.

The choice for the test problems comes yet again from the UCI repository and we target breast cancer, diabetes mellitus and iris discrimination.

Once the SVM pedagogical and decompositional steps were over and the training data was relabeled, we plotted the average number of samples in the training set that have the outcomes changed by the SVMs (pedagogical), besides the average number of support vectors (decompositional), both against the total number of training samples for each data set (see Fig. 7.3). Note that the sizes of the bars should be compared only to the ones within the same group. The number of support vectors is higher than the number of samples with changed outcomes, but significantly lower than the cardinal of the complete training data. This observation implies that a major reduction of dimensionality (in the number of samples) is performed prior to the application of CC in the decompositional case. Although for the iris data set the samples whose outcomes are changed (first bar) seem almost absent in the figure, the average value is in fact of 1.87.

The experimental setup is established as follows. As variation operators, we once more choose those common for a continuous encoding, i.e. intermediate recombination and mutation with normal perturbation. The binary tournament type is taken again as the selection operator. The size of each subpopulation is set to 50; it is however always taken k times the value (k being the number of classes), because there exists one population connected to each class. The number of evolutionary loops is considered as 80. The values for the mutation strength, mutation and recombination probabilities are chosen using the SPO [Bartz-Beielstein, 2006]. The values for the probabilities are picked from the [0, 1] interval, while for the mutation strength they are taken from [0, 2]. Each data set is again 30 times randomly split into 2/3 training and 1/3 test samples. The 30 training/test sets are the same in all approaches, for the SVM preprocessing to be identical. The reported prediction accuracy is obtained by averaging, over the 30 different runs, the percent of correctly labeled samples from the test set out of their total. When computing the fidelity to the SVMs, the known labels of the test samples are those given by the SVM output, then a typical accuracy ratio is computed once more.

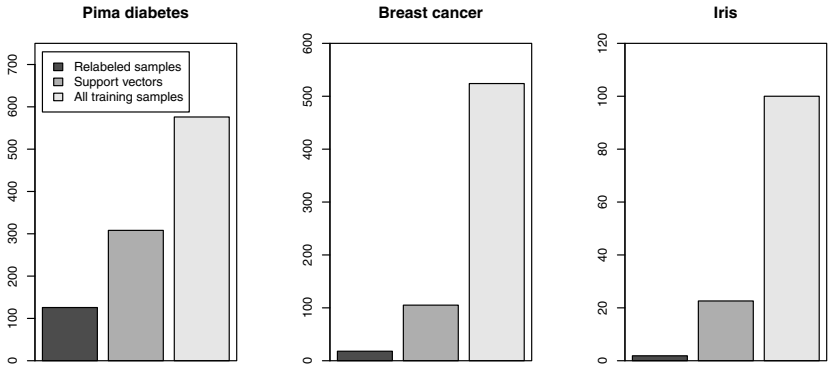

Fig. 7.3 Number of samples (on the vertical axis) changed by the SVMs within the pedagogical methodology, besides the number of support vectors from the decompositional approach and the total number of training samples for each data set

Table 7.1 outlines the prediction accuracies on the corresponding data sets obtained by CC alone, the pedagogical and decompositional SVM-CC, the SVMs and DT [Gorunescu, 2011]. The last methodology is included in the experiments, in order to investigate whether the SVM-CC white box extraction method performs better than a one-step transparent DT model applied directly to the initial data set [Stoean and Stoean, 2013a]. Additionally, we want to test if it may serve as an alternative to the CC as the explanatory engine. The p-values computed via a Wilcoxon rank-sum test show significant differences in results for diabetes (for the SVMs over CC, decompositional SVM-CC and DT) and iris (for both CC and SVM-CC variants over SVMs and DT).

In the last line of Table 7.1, we can also see that the fidelity criterion is obeyed by the hybridized approaches. High fidelity is very much desired, since it implies that the combined approach capably uses the SVM relabeling of the training data to learn the relationship between the values for the attributes and the triggered outcomes. It then uses the information to classify previously unseen data (almost) as efficiently as the SVMs. Fidelity is computed by measuring the similarity of test prediction between the combined approach and the SVM. Good fidelity however also conducts to better prediction accuracy, since SVM-CC usually behaves very similarly to SVMs, as observed in the fidelity outcomes, and SVMs represent a great choice of a classification algorithm to mimic.

If we look at the fidelity results in comparison to the prediction outcomes in Table 7.1, the relabeled samples from the training set clearly represent more accurate data for the SVM-CC approaches, since the fidelity values are with no doubt higher in general than the corresponding accuracies. This proves that SVM labeling eliminates noise from the data and learning becomes more efficient.

Table 7.1 Comparison between prediction accuracy results and standard deviations obtained on the test data sets by the considered approaches averaged over 30 repeated runs. The last rows additionally show fidelity to SVM predictions.

Data set	CC	Pedagogical	Decompositional	SVMs	DT
Average accuracy ± standard deviation (%)					
Breast cancer	96.78 ± 1.21	96.95 ± 1.14	95.92 ± 1.26	96.51± 1.41	94.11 ± 1.61
Pima diabetes	75.12 ± 3.63	76.49 ± 3.31	72.26 ± 3.85	77.31 ± 3.37	74.31 ± 2.92
Iris	97.6 ± 1.58	98 ± 1.46	98.07 ± 1.21	96 ± 2.07	93.73 ± 2.45
Fidelity to SVMs (%)					
Breast cancer	-	98.02	96.46	-	-
Pima diabetes	-	90.71	78.7	-	-
Iris	-	96.93	97.2	-	-

The pedagogical approach appears to be more consistent in accuracy results than the decompositional one. Also, it is more faithful to the SVM outcome. Its predictions from Table 7.1 can also be seen as better than the ones of the CC and closer to those of the SVMs. However, there is a great enhancement in runtime for the SVM-CC decompositional approach, since the training data set is drastically reduced to solely the support vectors.

When comparing the DT results with the ones of the CC alone, we can see that there is a small advantage for the latter: statistical testing confirms that CC is significantly better for the 2 disease diagnosis problems and equal for iris. The direct comparison between CC and DT was performed not only as to check whether the latter could serve as a better alternative for the explanatory algorithm in the two-step approaches, but also to underline the need for such combined techniques. The DT results are far below the ones of the proposed SVM-CC for the 3 data sets, so they cannot be a viable replacement.

Looking simultaneously at Fig. 7.3 and Table 6.1, it can be noticed that there is a strong relation between the average test accuracy results of the SVMs and the number of training samples that they relabeled. This number, plotted as the first bar in each group should be assessed as opposed to the third bar of the group that stands for the total number of training samples. For breast cancer and iris, where test accuracy goes beyond 90%, the number of labels changed by SVMs in the training set is very small. The reason is that such a data set already has the samples of different classes well separated, the noise amount in the data is low, so there are only a few problematic samples from the point of view of the SVMs. A similar correlation can also be observed between a low number of support vectors and the success of the SVMs again for iris and breast cancer.

For an intuitive understanding of the class prototypes, the discovered thresholds for attributes of each the three data sets are plotted in a random run of the pedagogical approach (see Fig. 7.4). One prototype is connected to each class and the latter is designated by a specific symbol. A class prototype is read by following the lines with a certain sign from the first attribute to the last one. The exact discovered

thresholds can be observed in the plots from the left hand side, but not all features have the same domain, so it is not relevant to compare the vertical distances for pairs of attributes. For a proper comparison between the thresholds, the same obtained values are normalized to the [0, 1] interval in the plots on the right hand side. Such a visualization helps to understand where are the thresholds situated on the initial intervals (plots on the left column) and what are the critical differences between attribute values discovered for the different classes (plots on the right column). The breast cancer problem however has each of the nine attributes defined on the same [0, 10] interval, so there is no difference between the positioning of the thresholds on the plots.

We can thus clearly see how some attributes count more than others. For the iris data set, for instance, it is the third attribute that makes a clear difference, while the thresholds for the others are very similar for all outcomes. For the Pima diabetes case, it is normalization that helps in distinguishing the importance of several features. While in the left plot there are many attributes that appear to have threshold values near one another and visually look alike, on the right one we can see that they are actually not that close - see attributes 1, 4, 6-8. The available class prototype set and its picture can prove helpful in supporting practical decision making, since the user can get usually fast aware of both the thresholds of demarcation between the classes and also of the relevance of each problem feature for the task.

7.5 Feature Selection by Hill Climbing – Revisited

When dealing with a large number of indicators, like those that define data in the medical field, where many of the attributes have little discriminative power between the potential outcomes, a means to reduce their number is especially important. The presence of too many attributes can divert classifiers as well as physicians from distinguishing those whose values differentiate between diagnoses. As also previously discussed, feature selection has been shown to help towards a faster and more accurate classification [Akay, 2009]. This commonly takes place before the actual classification, however, it can also be included as part of a cycle inside the classifier, which learns with differently selected features until some condition is met.

We were confronted with this problem while running the experiments for a first study on breast cancer diagnosis and generation of decision explanations by SVM-CC [Stoean and Stoean, 2013b]. We have said before that it had been shown in [Joachims, 1998] that the inner workings of SVMs bypass the dimensionality issue. The CC classifier cannot avoid it, however the adjustability of an EA framework offers the possibility to embed a feature selector within the evolution of classification thresholds [Stoean et al, 2011a], like we have seen before in Chap. 5. Therefore, once again, a dynamic chemistry between chosen indicators and their proper thresholds is performed. This interaction changes thresholds for attributes, as for every new HC individual different dependencies are involved and the SVM-CC is re-initiated with each change in the HC configuration. The reference to fewer indicators additionally offers more comprehensibility to the generated prototype set.

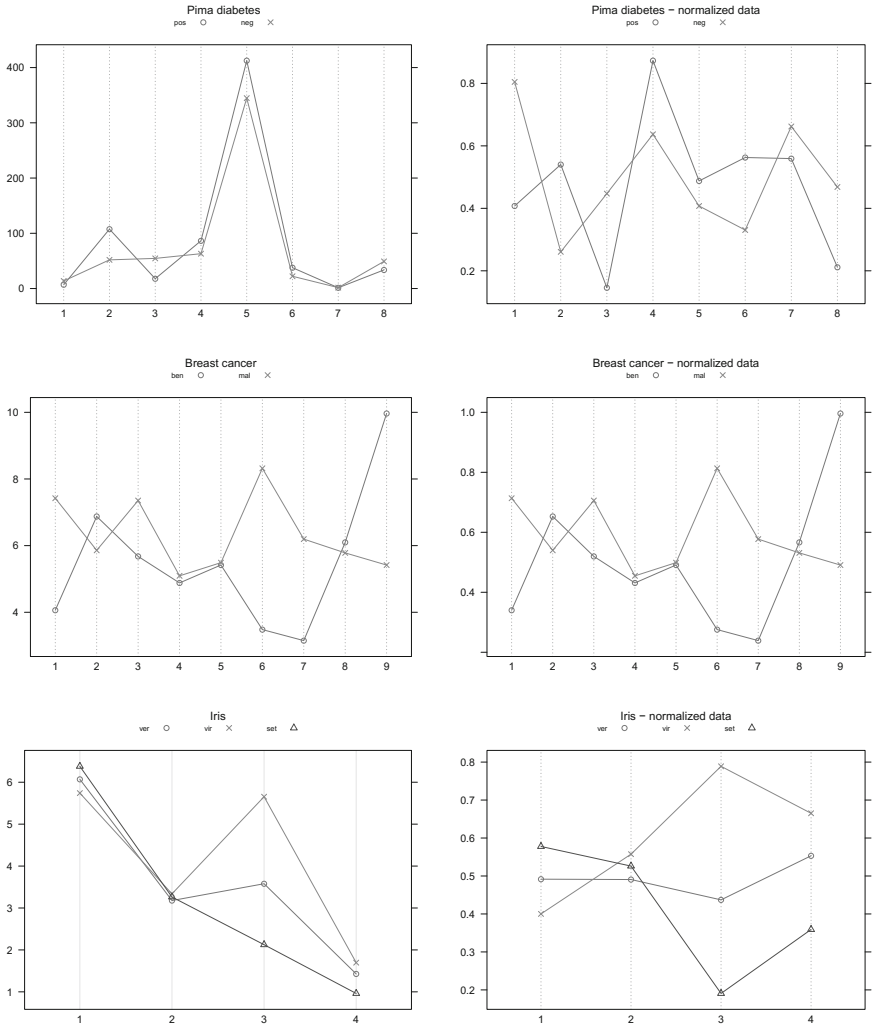

Fig. 7.4 The discovered class prototypes for the three data sets in a random run of the SVM-CC pedagogical approach. First column plots the obtained thresholds for the raw data, while within the second the values are normalized to the interval [0, 1]. The horizontal axis refers the attributes by their number and the vertical one shows their values. The different signs stand for the distinct classes of the problems (excerpt for [Stoean and Stoean, 2013a]).

Since it exhibited better results in the initial experimentation, it was the pedagogical approach that we selected for attaching the feature selector. The experimental setup is changed from the version in Chap. 5, as we can now also refer a goal

accuracy value for each problem (the one attained by the SVM). The HC runs until it reaches this value (best case scenario) or, if no improvement is achieved for 50 iterations, it is re-initialized and re-run. This re-initialization may happen for up to 5 times and if the targeted percent is not reached, it is the current accuracy that is returned. This desired goal is set for trying to break the limits of the SVM-CC algorithm as concerns its obtained average accuracy. The evaluation of the HC individual presumes a typical run of the approach and the obtained accuracy represents the fitness outcome. The HC algorithm conducts an iterated search for picking the most appropriate combination between the attributes of the classification problem and the weights discovered by the SVM-CC method for those features.

The prediction accuracy for the breast cancer problem now reached 97.16%, which is not significantly better than without the HC. The number of attributes is nevertheless reduced in average from 9 to 5. This means not only that noisy information is removed, but the user can also more easily grasp the decision prototypes, and consequently the classification problem. The most important features, i.e., those that are included into several prototypes (subsets of attributes) are thus evidenced. All these nevertheless bring an accompanying longer runtime, as the HC calls the pedagogical method at each iteration.

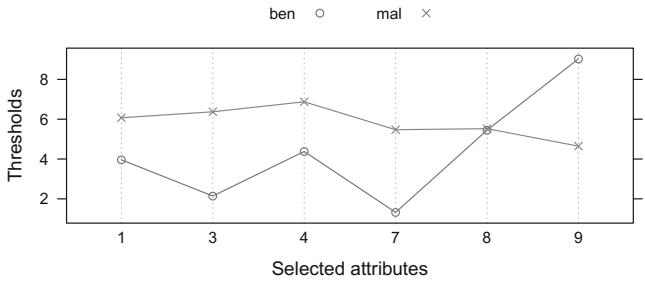

Fig. 7.5 The selected attributes and their generated thresholds after a random run of the HC as feature selector on the breast cancer diagnosis problem (excerpt from [Stoean and Stoean, 2013b])

The HC selected features and their corresponding thresholds are outlined in Fig. 7.5. As previously in Fig. 7.4, the prototypes from Fig. 7.5 provide an intuitive description for the discovered information. Larger distances on the vertical axis mean that there is a greater disparity for the values of that attribute and the indicator makes a clearer difference for the current prototype as opposed to attributes that have the values closer. In fact, it may happen that, for a different training configuration, the thresholds of the same attribute shall be close to each other. The indicator thresholds should hence not be read out of the context, but they should only be analyzed together with the configuration of the entire prototype. Recall that the prototypes are not unique, but they emphasize the connection between the values of different variables.

As before in Chap. 5, we also measure which attributes are more often selected. The results show that all indicators participate to the construction of the prototypes, some more often and some only seldom (Fig. 7.6). This means that good configurations of attribute thresholds can be achieved for different feature subsets, but surely some of them are more important since the best results are obtained when they are always part of the selection.

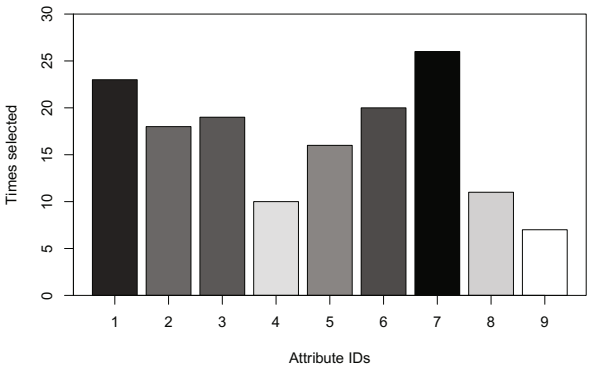

Fig. 7.6 Most often selected attributes (out of 30 HC runs) in breast cancer diagnosis (excerpt from [Stoean and Stoean, 2013b])

The attribute that was selected most by the HC (i.e., attribute 7, as seen in Fig. 7.6) is put to alone discriminate the data between the two classes of breast cancer diagnosis (Fig. 7.7). Based on the values of that specific attribute (on the horizontal axis), the samples are distributed at a high degree (on the vertical axis) between the two classes.

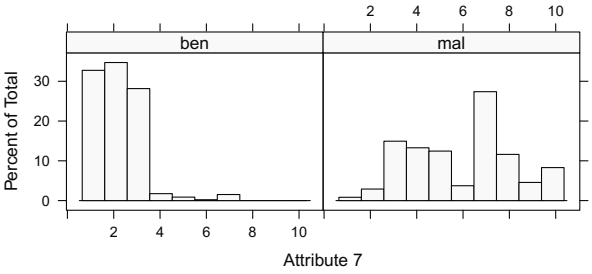

Fig. 7.7 Breast cancer sample distribution between the two outputs (benign and malignant) on the basis of the most important attribute as observed in Fig. 7.6 (excerpt from [Stoean and Stoean, 2013b])

7.6 Explaining Singular Predictions

Reducing the number of attributes by feature selection surely resolves the intricacy of the problem. But a medical expert would still have doubts as concerns the output of the model. A set of prototypes that is computationally easy to apply to a large set of test samples might not be very comprehensible for the user when it should classify a single record. The physician would surely prefer an indication towards

the features that determined the classifier to put a certain diagnostic for a patient [Strumbelj et al, 2010].

To also accomplish this with our approach, after a sample is classified, the absolute differences between the values of that record and the corresponding value of the prototype for that found class can be calculated for each attribute. By ascendingly ordering the attributes according to the values obtained for these absolute differences, a measure of their relevance (first being the most relevant) for the taken decision is achieved. Even if this methodology is extremely simple [Stoean and Stoean, 2013b], it outlines the individual attributes that have the closest values to the weights (as given by the prototypes) of those indicators which determine the diagnosis of a certain class.

The following experiment is further on performed for the breast cancer data: the first two patients with different diagnoses are chosen and the differences mentioned before are computed. Gathering the most important attributes for the first 10 patients taken under equally balanced diagnoses, the order for the most decisive attributes is found 1 and 7 for patients diagnosed as malign and the same attributes but in reverse order for the benign individuals. If we look again at Fig. 7.6, the former experiment revealed the importance of the same two attributes.

Such computations prove very useful in practice because they can be obtained for a current patient individually and they point out which are the most relevant indicators of the diagnosis in that case. By taking only small amounts of data from each problem, some of the features that were previously found as decisive by the more computationally expensive HC (in Fig. 7.5 and 7.6) are confirmed through this simple individual oriented classification.

7.7 Post-Feature Selection for Prototypes

After the evolution of prototypes ends, a set of resulting distinct solutions is selected (i.e., a collaborator from each class) to be tested against samples from the test set. If the decision set is inspected, it is natural that the thresholds for certain attributes may be closer to each other than others in different prototypes. We assumed that these attributes have little or no influence as concerns the classification of a new sample.

Algorithm 7.2 outlines a posterior feature selection to eliminate from each solution the attributes whose thresholds are very close to a mean over all prototypes for the corresponding values [Stoean and Stoean, 2013a]. Besides the decision set, the algorithm receives a positive integer, which is the significance threshold s under which less significant features are discarded. The values for s start from 0, when no attribute is removed, and can be incremented until a prototype remains with no attributes. Actually, the value of parameter s represents a percent of the definition span of the current attribute.

The algorithm begins by creating a vector of mean values, whose size is equal to the number of attributes of the classification problem. The value of locus i represents the average over all thresholds on position i of the considered prototypes. Then, for

each rule and accordingly for each class, we find the attribute that is most distant with respect to the vector of means. This similarity is normalized for each attribute in order to have a relative comparison. Such a value is important in determining to what extent can the significance threshold be increased until a prototype is completely eliminated because all its attributes are marked as unimportant. The fact that each solution has at least one attribute with a significant value is assured in the condition line that verifies if $(b_i - a_i) \cdot s/100 < threshold^l_{dist}$. Subsequently, each attribute of every rule is considered and a difference in absolute value is computed against the rates from the vector of means. If the obtained positive number is lower than the significance threshold, then this attribute is ignored for the current prototype. Note that for classification problems with more than two classes, an attribute may be removed from a prototype, but it is further kept in a complementary one, as it can be important for one class, but insignificant for others.

Algorithm 7.2 Post-feature selection for class prototypes.

Require: The set of k prototypes, k being the number of classes, and a significance threshold s for attribute elimination
Ensure: The k prototypes holding only the relevant attributes
begin
Compute vector *mean* of length n by averaging the values for each attribute threshold over all the prototypes {n is the number of attributes}
for each prototype l **do**
Find $threshold^l_{dist}$ among all $threshold_i$, where $i \in \{1, 2, ..., n\}$, that is the remotest to $mean_i$, i.e., corresponds to $\max_{i=1}^{n} \frac{|\, threshold_i - mean_i \,|}{b_i - a_i}$
end for
for each prototype l **do**
if $(b_i - a_i) \cdot s/100 < threshold^l_{dist}$ **then**
for each $attribute_i$ **do**
if $|threshold_i$ - $mean_i| < (b_i - a_i) \cdot s/100$ **then**
Mark i as a *don't care* attribute for prototype p
end if
end for
end if
end for
end

What remains to be reformulated is the corresponding change in the application of prototypes of different lengths to the test set. The distance from an unknown sample is now applied only to the attributes that matter from the current prototype and it is divided by the number of solely these contributing features. The motivation for this division lies in the fact that some prototypes may have many relevant attributes, others can remain with a very low number and in this way the proportionality of comparison still holds.

We chose again the more stable pedagogical approach for testing this planned enhancement and we tried values for the significance threshold to the maximum possible number. We applied it only to Pima diabetes and breast cancer diagnosis, as the iris data already has only 4 attributes. The obtained accuracy, as well as the number of eliminated attributes for the two data sets, are shown in Fig. 7.8. The horizontal axis contains the values for the significance threshold. On the vertical axis there is one line with accuracies followed by another with the number of removed attributes, in order to have a simultaneous comparison. A change in accuracy is almost absent. The gain is nevertheless represented by the fact that the number of dimensions is substantially reduced and the remaining thresholds for the decisive attributes within the class prototypes can be more easily analyzed and compared.

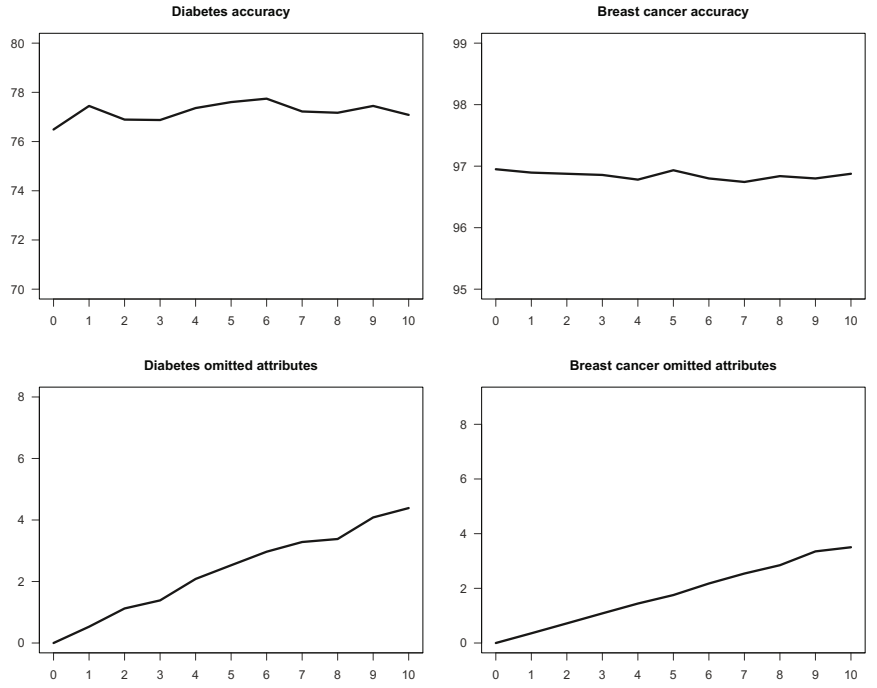

Fig. 7.8 The plots on the first line output the accuracies obtained for Pima diabetes and breast cancer diagnosis, when attributes are eliminated. The graphics from the second line show how many features are discarded for each. The horizontal axis contains values for the significance threshold used in eliminating the attributes, while accuracy (line 1) and number of removed attributes (line 2) are represented on the vertical axis (excerpt from [Stoean and Stoean, 2013a]).

Figure 7.9 plots the new prototypes of every class for the largest significance threshold taken for each case in Fig. 7.8, that is the highest value on the horizontal axis. Since the classification problems targeted here have only two classes, an attribute is eliminated from both prototypes at the same time. If the task is however multi-class, an attribute may be discarded only from certain prototypes. This happens because attribute elimination is applied by making use of an average over the prototypes for all classes. In the binary case, the thresholds for the two prototypes have an equal distance to the mean, so they are both or none eliminated [Stoean and Stoean, 2013a].

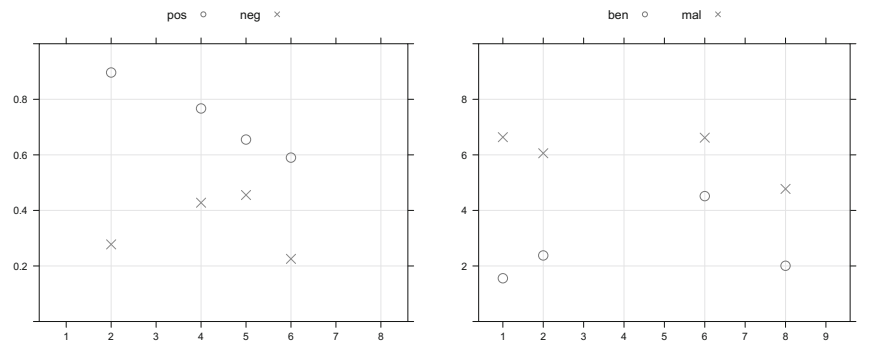

Fig. 7.9 Illustration of the remaining attributes and their threshold values for the highest significance threshold following Fig. 7.8 - Pima diabetes data left and breast cancer diagnosis on the right (excerpt from [Stoean and Stoean, 2013a])

7.8 Concluding Remarks

Following all the earlier experiments and observations, we can draw the following conclusions as to why is SVM-CC – a combination between a kernel-based methodology and a prototype-centered classifier – a good option for white box extraction:

- The EA encoding is simpler than genetic programming rules [Johansson et al, 2010] and the separation into explicit multiple subpopulations, each holding prototypes of one class, is more direct than ant colony optimization [Ozbakir et al, 2009] and easier to control than island models [Markowska-Kaczmar and Chumieja, 2004] or genetic cromodynamics [Stoean et al, 2007], [Stoean and Stoean, 2009a].
- The possibility of easily including a HC into the representation triggers simultaneous feature selection and information extraction.
- The option of allowing only the presence of the informative indicators additionally facilitates a deeper and more pointing understanding of the problem and relevant features, all centered on their discriminative thresholds.

- The derived prototypes discover connections between various values of different attributes.
- Through the individual classification explanation, the expert is able to see a hierarchy of the attributes importance in the automated diagnosis process.
- A second method of reducing the complexity of the decision guidelines by omitting for each class the attributes that had very low influence for the corresponding discrimination prototype leads to a compact formulation of the involved attributes. This helps discover relevant insights on the problem at hand, as well as shows a more understandable picture of the underlying decision making.
- When feature selection is performed, either online or a posteriori, the accuracy does not increase, but comprehensibility nevertheless does grow. This is somehow obvious, since the less informative attributes were probably weighted less and had small or no influence on the resulting predictions.
- If we were however to compare the comprehensibility power of both feature selection approaches, we would say that cleaning each prototype of the insignificant indicators for that outcome leads to better understandability than the online approach that performs global feature selection on the data set. This is due to the fact that the a posteriori method reveals the specific interplay between selected attributes for each particular class in turn.