JMLR: Workshop and Conference Proceedings 10: 45-54 The Fourth Workshop on Feature Selection in Data Mining

Effective Wrapper-Filter hybridization through GRASP Schemata

M.A. Esseghir messeghir@sfr.fr
Artois University, Faculty of Applied Sciences of Bethune,
TechnoPark Futura, 62400, France.

Editor: Huan Liu, Hiroshi Motoda, Rudy Setiono, and Zheng Zhao

Abstract

Of all of the challenges which face the selection of relevant features for predictive data mining or pattern recognition modeling, the adaptation of computational intelligence techniques to feature selection problem requirements is one of the primary impediments. A new improved metaheuristic based on *Greedy Randomized Adaptive Search Procedure* (GRASP) is proposed for the problem of Feature Selection. Our devised optimization approach provides an effective scheme for wrapper-filter hybridization through the adaptation of GRASP components. The paper investigates, the GRASP component design as well as its adaptation to the feature selection problem. Carried out experiments showed Empirical effectiveness of the devised approach.

Keywords: Feature selection, Combinatorial optimization, Hybrid modeling, GRASP, Local Search.

1. Introduction

Researchers in machine learning, data mining, pattern recognition and statistics have developed a number of methods for dimensionality reduction based on usefulness and classification accuracy estimates of both individual features and subsets. In fact, Feature Selection (FS) tries to select the most relevant attributes from raw data, and hence guide the construction of the final classification model or decision support system. From one hand, the majority, of learning scheme, are being relying on feature selection either as independent pre-processing technique or as an embedded stage within the leaning process (Guyon and Elisseff, 2003). On the other hand, both feature selection and data mining techniques struggle to gain the attended reliability, especially, when they face high dimensional data (Liu and **Motoda**, 2008).

In this paper, we propose, a new hybrid search technique through the adaptation of the GRASP heuristic to the FS problem. The devised approach investigates the effective wrapper-filter combination by exploiting the intrinsic properties of the GRASP heuristic. The main motivations for this proposal are three folds: (i) filter-wrapper collaboration might enhance the relevance of the selected feature subsets (ii) local search approaches have shown their effectiveness in FS with both sequential deterministic procedures (*i.e.* SFFS (Somol et al., 1999), IFFS (Nakariyakul and Casasent, 2009), *etc*) and stochastic approaches (*i.e.* Hill Climbing (Kohavi and John, 1997), Simulated Annealing (Guyon et al., 2006) and Tabu search (Yus, 2009)). The GRASP is a multi-start heuristic based on local search (iii) endowing basic sequential search procedures with both filter guidance and the stochastic ability to alleviate FS challenging problems like local minimas and nesting effect (Guyon et al., 2006; Liu and **Motoda**, 2008).

The remainder of this paper is organized in five sections. Section 2 formalizes the feature selection problem and gives an overview of representative approaches. Section 3 briefly introduces the GRASP heuristic as well as its recent application to feature selection. Section 4 details the devised GRASP for FS. Section 5 compares and assesses GRASP alternatives behaviors empirically. Finally, Section 6 concludes the paper and provide some directions of future research.

2. Feature selection: basics and background

Let D be a data set with F as a set of features such that $|F| = n$, and let X ($X \subseteq F$) be a subset of F. Let $J(X)$ the function that assesses the relevance of the features subset X. The problem of feature selection states the selection of a subset Z such that:

$$J(Z) = max_{X \subseteq F} J(X) \tag{1}$$

In other words, the retained feature subset should be compact and representative of the dataset objects or the underlying context. This might be done by both removing redundant noisy or/and irrelevant attributes by keeping the minimal information loss.

For a given dataset of n features, the exploration would require the examination of 2^n possible subsets. Consequently, the search through the feasible solutions search space is a *np*-hard combinatorial problem (Liu and **Motoda**, 2008). An exhaustive exploration of the feature space seems to be impractical, especially, when n became large or even moderate. Numerous reference approaches have been proposed for the identification of salient features with the highest predictive power (Guyon et al., 2006; Liu and **Motoda**, 2008). The representative approaches could be, broadly, categorized into two classes: *filters* and *wrappers* (Guyon and Elisseff, 2003; Guyon et al., 2006).

Filters

Considered as the earliest approaches to feature selection, filter methods discard irrelevant features, without any reference to a data mining technique, by applying independent search which is mainly based on the assessment of intrinsic attribute properties and their relationship with the data set class (*i.e.* Relief, Symmetrical uncertainty, Pearson correlation, *etc*) (Liu and **Motoda**, 2008). The main advantage of the filter methods is their reduced computational complexity which is due to the *simple* independent criterion used for feature evaluation. In most of the cases, filters provide a ranking based on scores reflecting attribute usefulness to the class.

Wrappers

When feature selection is based on a wrapper, attributes are simultaneously evaluated using a classification algorithm. The subset exploration requires a heuristic search strategy. Kohavi *et al.* (Kohavi and John, 1997) were the first to advocate the wrapper as a general framework for feature selection in machine learning. Numerous studies have used the above framework with different combinations of the evaluation and search components. Featured search technique are ranging from greedy sequential attribute selection methods (*i.e.* SFS, SBE, Floating search (Somol et al., 1999)) to randomized and stochastic methods (*i.e.* GRASP (Yus, 2009), TABU, BEAM, Genetic Algorithm (Guyon et al., 2006)) . The wrapper methods often provide better results than filter ones because they consider a classifier within the evaluation process. We should note that feature selection methods based on wrappers are computationally expensive compared to filters, due to the cost of iterative running of the classification algorithm (Guyon and Elisseff, 2003).

The motivation to hybrid approaches design is the exhibited multidisciplinary problem nature and the need to overcome the pitfalls of one approach by the advantage of the other one. The simplest form of recombination is to use both filters and wrappers. The common scheme of combination entails a couple of steps. The first one applies filter to reduce the search space, while the second step explores with a wrapper the subsets built from the yielded features returned by the first step (Liu and **Motoda**, 2008).

3. GRASP

This section covers paper materials by introducing GRASP heuristic principle, components and its recent application to feature selection modeling. The Greedy Randomized search procedure (GRASP) is meta-heuristic for combinatorial optimization problems (Resende, 1999; Resende et al., 2002). Usually known as multi-start procedure, GRASP is based on an iterative process which constructs a solution then fine tune it, through a local search. The multi-start property enlarges the search coverage by exploring different regions of the search space without being influenced by the previous solutions found.

GRASP heuristic was successfully applied to numerous combinatorial problem ranging from scheduling (Aiex et al., 2003) and quadratic assignment (Ahuja et al., 2000) to data mining (Ahmadi and Osman, 2005).

3.1 GRASP Components

The recent optimization scheme proposed by GRASP (Resende, 1999) applyies an iterative local search scheme based on incremental solution construction and neighborhood exploration. The iterative process consists of two stages: *the construction* of a feasible solution and *local search*.

The construction stage builds a solution, incrementally, using a *Restricted List of Candidates RCL* The RCL is, generally, formed by the best solution elements (*i.e.* elements which can improve the current solution). Solutions are built using a random selection from the RCL.

Once the solution is generated, it passes through the second stage. Within the second stage the solution is iteratively, refined by local search until it reaches a local optima. This procedure is mainly based on neighborhood generation and the exchange of the current solution by the best solution among neighbors. The procedure restarts until no improvement could be gained. The pseudo-codes of both GRASP local search (LS) procedures adapted to the problem of FS will be detailed, below, by Algorithm 1 and 2 in the following section.

The multi-start property of GRASP allows the search process to be not trapped in local minima and to explore different regions of the search space, without being constrained or influenced by the best solution found.

Algorithm 1: G.R.A.S.P.

Input:
F: Initial Feature set
C: Target class Attribute
β: Threshold
d: number of attributes to select
n_{max}: attempts number
Output:
S_{best} : Selected Features

Begin
- $S \leftarrow \emptyset$
- $S_{best} \leftarrow S$
- **While** *Stop Criterion not Satisfied* **do**
  - //Construction stage
  - **Foreach** $f_i \in F$ **do**
    - $g_i \leftarrow IGV(f_i, C)$
  - $Sol_{list} \leftarrow \emptyset$
  - **repeat**
    - $S \leftarrow \emptyset$
    - **repeat**
      - $min \leftarrow argmin_i(g_i)$, $max \leftarrow argmax_i(g_i)$
      - $RCL_{list} = \{v_j, g_j \leq \alpha.g_{max} + (1-\alpha)g_{min}\}$
      - Randomly select $v_j \in \{v_j \in RCL_{list}, v_j \notin S\}$
      - $S \leftarrow S \cup \{v_j\}$
      - $RCL \leftarrow RCL \setminus \{v_j\}$
    - **until** $|S| == d$;
    - $S.fitness$=Evaluate(S,Cla)
    - $Sol_{list} \leftarrow \{S\} \cup Sol_{list}$
  - **until** $|Sol_{list}| = n_{max}$;
  - $S \leftarrow$ getBest(Sol_{list});
  - // iterative local search
  - $S \leftarrow LocalSearch(S)$
  - **If** $S.fitness > S_{best}.fitness$ **then**
    - $S_{best} \leftarrow S$
- Return (S_{best})

End

Algorithm 2: Iterative Local Search procedure

Input:
F: Initial Feature set
C: Target class Attribute
Cla: a classifier for solution evaluation
S: input Solution
Output:
S' : result of local search

Begin
- $S1 \leftarrow S$, $S_{best} \leftarrow S1$
- Stop $\leftarrow false$
- **repeat**
  - $Sol_{list} \leftarrow NH(S1, F)$
  - $\forall X \in Sol_{list}, Evaluate(X, Cla)$
  - $S1 \leftarrow getBest(Sol_{list})$
  - **If** $S1.fitness > S_{best}.fitness$ **then**
    - $S_{best} \leftarrow S1$
  - **Else**
    - Stop $\leftarrow true$
- **until** *(Stop = true)*;
- *S'* $\leftarrow S_{best}$
- Return (*S'*)

End

3.2 GRASP for Feature Selection

The application of GRASP to the FS problem was, recently provided by Yusta in (Yus, 2009). The proposed GRASP was compared to effective FS search techniques like to genetic algorithms, tabu search and SFFS.

The GRASP proposed in (Yus, 2009), is illustrated by Algorithm 1. The algorithm is based on two main stages, namely solution construction (Lines 6-21) and local search procedure (*see* Algorithm 2). The first stage constructs a fixed number of solutions (n_{max}), and the best one will be selected as a candidate for the second stage.Solution are constructed according to the attributes retained by the RCL list. The RCL is based on the In-Group Variability (IGV function) criterion (*see eq.* 2).

$$IGV(f_j, C) = \sum_i (f_j^i - \mu_{C(i)})^2 \tag{2}$$

Where f_i^j and $\mu_{C(i)}$ denote respectively the *i-th* value of the attribute f_j and the mean $\mu_{C(i)}$ of f_j values for the instances belonging to the same class as the instance i. Besides the attribute selection, is controlled by the parameter α (Lines 11-17). In fact, it controls the degree of randomness of the procedure.

The second stage applies a hill climbing procedure to the solution provided by the first stage. The pseudo-code of the iterative LS is illustrated by algorithm 2. Each iteration generates neighborhood solutions and exchange current solution with best neighbors if it can improve classification accuracy. The neighborhood structure proposed, by Yusta in (Yus, 2009), is based on attribute replacement and is given by the equation 3:

$$NH(S) = \{X, X = S \cup \{f_i\} \setminus \{f_j\}, \forall f_i \in F \setminus S, \forall f_j \in S\} \tag{3}$$

Such a neighborhood structure $NH(S)$ consider all combinations of attribute exchange. Consequently, LS is sensitive to the number of selected features. The neighborhood exploration becomes prohibitive even for moderate value of n. The computational complexity is in the order of $O(p * m)$. [1]

4. Proposed GRASP-FS

In this section, we investigate, the proposed a new GRASP schemata for FS. We focus on a set commonly used local search procedures and filters. Next, we try to adapt and deploy them as GRASP components.

Since the GRASP scheme is based on a restricted list of candidates, this list could be represented by features that seems to be relevant or those that might provide incremental usefulness to the selected feature subset. For the GRASP construction stage we opt for selection scheme capable of generating attribute ranking. Hence, the score associated to features will serve as selection criterion for the RCL generation. The second stage of GRASP tries to enhance solutions by an iterative neighborhood exploration. The subsets are assessed according to a classification criterion (*i.e.* generalization error rate). The quality of solution fine-tuning, mainly, depends on the nature of the involved neighborhood structure of LS.

1. p and m respectively denote the number of selected and non-selected features ($p + m = n$).

We devise a number of LS procedures based on different neighborhood structures inspired from well known sequential search procedures. The following two sections, detail different design alternatives for both RCL and local search GRASP component.

4.1 RCL generation

Comparatively to the GRASP scheme proposed by Yusta in (Yus, 2009), the same construction phase steps (*see* Algo. 1 Lines 6-21) are adopted, except the procedure which generates the RCL the (*see* Algo. 1 Line 6-7).

Any filter criterion could be, instead, used to build RCL. In this paper, we opt for three well known and different selection schema: *ReliefF* (Robnik-Sikonja and Kononenko, 1997), *Symmetrical Uncertainty* (SU) (Guyon et al., 2006), and *FCBF* (Yu and Liu, 2003).

Typically, filters return solutions based on the selection of features with the highest scores. Once the initial RCL is generated[2] , the variables are randomly selected to build GRASP first stage solutions. Such a selection schema have, at least, tow benefits: (i) reducing the risk of selection of, only, highly correlated relevant features (ii) the combination of features with moderate usefulness, which are not highly relevant to the target, might promote interaction among selected attributes.

4.2 Local Search Procedures

The local search (LS) is applied at the second stage of the GRASP. It aims at the improvement of the solution provided by the GRASP first stage process. An interesting aspect that could motivate the wrapper choice as component of the GRASP second stage, is the successful application of local search methods in FS modeling (*i.e.* Tabu search, Simulated annealing, Hill climbing) (Guyon et al., 2006).

In this paper, we devise effective LS procedures inspired from successfully search techniques adapted to the FS. The following paragraphs detail the neighborhood structures that will be deployed within the local search procedures. They will be, also, discussed in the context of FS search space exploration.

Bit-Flip Local search (BF) explores neighboring solutions by adding or removing one feature at a time. For solutions encoded with binary strings this operator inverts one bit value for each neighbor. In comparison to the sequential search procedures, the generated neighborhood covers both solutions explored in SFS (*see eq. 5*) and SBE (*see eq. 6*) The bit-Flip operator (BF) neighborhood is illustrated by the equation 4.

$$NH_{BF}(S) = \{X | X = NH^{+}(S) \cup NH^{-}(S)\} \tag{4}$$

$$NH^{+}(S) = \{X | X = S \cup \{f_i\}, \forall f_i \in F, f_i \notin S\} \tag{5}$$

$$NH^{-}(S) = \{X | X = S \setminus \{f_i\}, \forall f_i \in S\} \tag{6}$$

The problem of nesting effect encountered with both sequential forward and backward procedures is alleviated by the merge of the neighborhoods explored by both procedures.

2. using filter criterion

Attribute-Flip (AF) local search procedure constructs neighborhood using a permutation between a selected and a non-selected features (*see eq.* 3). This neighborhood structure was used, by Yusta in (Yus, 2009), as a local search procedure. The two operators explore different region of the current solution neighborhood. There is no overlapping regions ($NH_{BF}(S) \cap NH_{AF}(S) = \emptyset$) and the second neighborhood structure is much larger than the first which would require more computational time.

Local search based floating search (SFFS1). Since, SFS and SBE approaches could be seen as local search procedures, floating searches (SFFS and SFBS) could be also considered as an improved version of both sequential procedures and their associated local search. In fact, solutions explored by an iteration of SFFS are those generated by the union of $NH_1 = NH^+(S)$ and the conditional application of the backward search to the best improvement provided by NH_1. Note that the LS based on SFFS1 neighborhood is not comparable to that using AF local search. AF applies $NH^+(.)$ and $NH^-(.)$ to the same initial solution while, with SFF1, $NH^-(.)$ is applied to the improved solution after the application of $NH^+(.)$. Besides, there is no risk of cycling, because the Neighborhood procedures are only applied to improved solutions.

Local search based floating search 2 (SFF2) In the case of SFF2, the same floating search scheme as in SFF1 is adopted, however the backward procedure $NH^-(.)$ is not applied once but the backtrack is applied iteratively repeated until no improvement can be reached. Comparatively to SFF1, SFF2 requires more computational time than the first floating alternative but might lead to more compact subset size.

5. Empirical study

In this section, we empirically assess the behavior of proposed GRASP schema as well as a selection of the devised components. They will be, also, compared to the baseline GRASP $\langle IGV, AF \rangle$ proposed by Yusta in (Yus, 2009), where reported results have confirmed the superiority of GRASP over Tabu search, Genetic and Memetic algorithms, and SFFS approach.

Five benchmark datasets were used to validate GRASP components: Sonar, Ionosphere, SpamBase, Audiology and Arrhythmia with respectively 60, 34, 57, 69 and 279 attributes. These datasets are provided by the UCI repository (Blake and Merz, 1998). Reported results, correspond to the average values of at least 50 trial runs. Means, Standard deviation and statistical test validation (t-Test with confidence level of 97.5%) are also provided.

Two types of results are proposed: (i) those corresponding to the best solution fitness (generalization error rate) yielded from the GRASP search. K-Nearest Neighbors (KNN) is used as wrapper classifier ($K = 3$) (ii) the validation on independent data set instances of the resulting features subsets using Artificial neural network (ANN) and Naive Bayes(Guyon et al., 2006). The selection of different classification paradigms for both search and validation make the validation less biased and independent of wrapper classifier. Besides, the validation stage is based on 10-folds cross-validation technique.

Data	Model		Fitness (%)	Validation Error (%)		CPU Time(s)	Gain %
	RCL	LS		ANN	NB		(Yus, 2009)
Sonar	IGV	AF	15,89(1,71)	32,83(3,40)	40,30(3,01)	14183,88(7196)	
	Relief	AF	$14,29(1,88)^{+}$	$31{,}31(3{,}66)^{+}$	$39{,}85(2{,}71)^{+}$	14915,13(7493)	10,07%
	SU	AF	$12{,}79(1{,}13)^{+}$	$30{,}54(3{,}06)^{+}$	$39{,}53(2{,}51)^{+}$	15867(8524)	19,51%
	FCBF	AF	$13{,}46(0{,}00)^{+}$	$31{,}32(1{,}30)^{+}$	$37{,}27(1{,}70)^{+}$	14920(7450)	15,29%
Audiology	IGV	AF	49,12(1,96)	52,4(3,14)	54,05(0,15)	343915(280683)	
	Relief	AF	$46{,}74(3{,}92)^{+}$	$51{,}54(4{,}47)^{+}$	54,09(0,22)	$337756(248912)^{-}$	4,85%
	SU	AF	$33{,}36(3{,}2)^{+}$	$40{,}52(4{,}55)^{+}$	54,16(0,25)	350761(267582)	32,08%
	FCBF	AF	$36{,}08(4{,}72)^{+}$	$40{,}53(6{,}91)^{+}$	54,06(0,14)	338687(258913)	26,55%
Arrhythmia	IGV	AF	39,72(1,57)	41,98(2,04)	43,61(1,65)	183959(132900)	
	Relief	AF	40,17(1,76)	42,7(2,05)	44,22(1,74)	$170925(114978)^{+}$	-1,13%
	SU	AF	$36{,}15(1{,}89)^{+}$	$39{,}73(2{,}42)^{+}$	44,56(1,79)	$173505(115749)^{+}$	8,99%
	FCBF	AF	$33{,}82(1{,}26)^{+}$	$39{,}33(2{,}11)^{+}$	$43{,}54(1{,}76)^{+}$	176065(117012)	14,85%
Ionosphere	IGV	AF	5,63(0,91)	16,34(1,98)	17,9(2,13)	22316(12626)	
	Relief	AF	5,95(1,00)	15,51(2,26)	$16{,}97(2{,}04)^{+}$	21855(12081)	-5,68%
	SU	AF	5,76(0,98)	$15{,}21(2{,}48)^{+}$	$17{,}31(2)^{+}$	24031(14531)	-2,31%
	FCBF	AF	$3{,}51(0{,}32)^{+}$	$16{,}33(0{,}92)^{-}$	15,73(0,89)	$21973(11873)^{+}$	37,66%
SpamBase	IGV	AF	16,47(1,04)	19,91(1,50)	20,23(1,57)	347062(190196)	
	Relief	AF	$16{,}43(1{,}05)^{-}$	$19{,}59(2{,}19)^{+}$	$19{,}58(1{,}72)^{+}$	$338671(185750)^{+}$	0,24%
	SU	AF	$14{,}18(1{,}12)^{+}$	$15{,}89(1{,}66)^{+}$	$17{,}13(2{,}27)^{+}$	311037(156931)	20,77%
	FCBF	AF	$13{,}05(0{,}84)^{+}$	$15{,}96(2{,}18)^{+}$	$15{,}31(1{,}88)^{+}$	331498(181414)	20,97%

[3]rests format: $m(sd)^{+/-}$:m: Mean; sd:Standard deviation;(+/−):T-test validity

Table 1: GRASP with RCL based filters

Data	Model		Fitness (%)	Validation Error (%)		CPU Time(s)	Gain %
	RCL	LS		ANN	NB		(Yus, 2009)
Sonar	IGV	AF	15,89(1,71)	32,83(3,40)	40,30 (3,01)	14183(7196)	
	IGV	BF	28,68(1,58)	33,59(4,90)	41,22(4,19)	15215(7922)	-80,49%
	IGV	SFF1	$5{,}92(2{,}05)^{+}$	$31{,}14(3{,}47)^{+}$	$40{,}05(3{,}08)^{+}$	14481(6652)	62,74%
	IGV	SFF2	$6{,}6(1{,}9)^{+}$	$31{,}26(3{,}38)^{+}$	$38{,}75(3{,}62)^{+}$	$12208(5244)^{+}$	58,46%
Audiology	IGV	AF	49,12(1,96)	52,4(3,14)	54,05(0,15)	343915(280683)	
	IGV	BF	68,78(1,34)	69,59(2,87)	72,64(2,4)	$322789(248574)^{+}$	-40,02%
	IGV	SFF1	$29{,}41(1{,}47)^{+}$	$41{,}78(3{,}96)^{+}$	54,09(0,22)	$234579(98248)^{+}$	40,13%
	IGV	SFF2	$30{,}99(1{,}21)^{+}$	$41{,}12(2{,}81)^{+}$	54,08(0,17)	$209670(111471)^{+}$	36,91%
Arrhythmia	IGV	AF	39,72(1,57)	41,98(2,04)	43,61(1,65)	183959(132900)	
	IGV	BF	49,47(1,02)	44,8(1,98)	46,34(1,59)	$160659(96674)^{+}$	-24,55%
	IGV	SFF1	$25{,}38(2{,}57)^{+}$	$38{,}27(2{,}9)^{+}$	$43{,}41(1{,}81)^{+}$	$156343(73463)^{+}$	36,10%
	IGV	SFF2	$24{,}42(2{,}67)^{+}$	$36{,}64(2{,}99)^{+}$	$42{,}64(1{,}71)^{+}$	$152719(89546)^{+}$	38,52%
Ionosphere	IGV	AF	5,63(0,91)	16,34(1,98)	17,9(2,13)	22316(12626)	
	IGV	BF	12,63(0,67)	$15{,}78(3{,}01)^{+}$	$17{,}38(2{,}03)^{+}$	$21374(11603)^{-}$	-124,33%
	IGV	SFF1	$2{,}27(0{,}52)^{+}$	$14{,}77(1{,}64)^{+}$	$17{,}35(1{,}2)^{+}$	$18561(8464)^{+}$	59,68%
	IGV	SFF2	$2{,}48(0{,}56)^{+}$	$15{,}42(1{,}44)^{+}$	$17{,}66(1{,}03)^{+}$	$16035(6597)^{+}$	55,95%
SpamBase	IGV	AF	16,47(1,04)	19,91(1,50)	20,23(1,57)	347062(190196)	
	IGV	BF	23,72(1,1)	22,38(3,46)	21,59(2,43)	328341(17611)	-44,02%
	IGV	SFF1	$6{,}85(0{,}73)^{+}$	$12{,}28(1{,}27)^{+}$	$14{,}90(2{,}66)^{+}$	532610(281813)	58,41%
	IGV	SFF2	$6{,}87(0{,}84)^{+}$	$12{,}05(1{,}26)^{+}$	$15{,}11(2{,}75)^{+}$	496728(206866)	58,29%

[4]rests format: $m(sd)^{+/-}$:m: Mean; sd:Standard deviation;(+/−):T-test validity

Table 2: GRASP with different local search procedures

5.1 Construction Phase

In the first stage of the empirical study, we assess the behaviors of the baseline GRASP with the devised GRASP scheme which is based on Filters to both built RCL and construct solutions.

For each experiment we present, for each dataset, on columns, best solution fitness (lowest error rate %), test accuracy on independent dataset, average runtime CPU, cardinality of best solution ($\#features$) and the gain Toward Baseline GRASP fitness. In Addition to the, average , standard deviation values of the different trials, *t-test* was used for the assessment of the statistical validity of the obtained results toward the baseline method. Table 1 provides results for each data set. Globally, according to the gain (last column) obtained with a GRASP scheme generating the RCL with filters, the baseline method is outperformed in most of the cases.

Fortunately, the improvement obtained with fitness values is confirmed with validation stage (independent data, and different classification techniques for validation). In most of the cases the mean values and t-tests showed decrease of the generalization errors. The overall improvement, points out the reliability of the approach, particularly the filters enlisted in the selection of suitable features. All filters enhance at least once, both fitness and validation accuracies. Surprisingly, Relief scores used in the RCL build, seems to be the less relevant filter used in the first stage of GRASP whereas GRASP alternatives based on FCBF confirm a slight superiority over those ones using SU.

5.2 Local search enhancement

The local search of the baseline method uses Attribute Flip neighborhood whereas the proposed GRASP uses local search procedures inspired from deterministic sequential searches.

The devised local search procedures are deployed within new GRASP instances using the IGV criterion on the First stage. Table 2 compare and evaluate the fours GRASP instances. Even though, the solutions provided by the first GRASP stage are based on IGV criterion, some of the devised local search procedures have succeed to outperform the baseline algorithm. Indeed, local search alternatives adopting floating selection, have empirically confirmed their superiority over Yusta GRASP. On the other hand, the neighborhood structure based on the selection or removal of one attribute (NH_{BF}), the less effective fine tuning scheme.

Besides, the overall improvement of the new devised GRASP local search procedures are most significant that the improvements afforded by the use of filters. In any case, the adapted new GRASP scheme is based have empirically shown that enhancements could be afforded by filters in first stage as well as wrappers in second stage.

6. Conclusion

We devise a new GRASP approach for feature selection capable of hybridizing filters and wrappers. The effectiveness of the different GRASP components combinations were assessed empirically. Carried out results, confirms the robustness of the hybridization schemata and motivates us to investigate in depth both algorithmic and behavioral aspects of further combinations issues, scalability study, and adaptation to high dimensional problems.

References

Different metaheuristic strategies to solve the feature selection problem. *Pattern Recognition Letters*, 30(5):525 – 534, 2009. ISSN 0167-8655.

Samad Ahmadi and Ibrahim H. Osman. Greedy random adaptive memory programming search for the capacitated clustering problem. *European Journal of Operational Research*, 162(1):30–44, 2005.

R.K. Ahuja, J.B. Orlin, and A. Tiwari. A greedy genetic algorithm for the quadratic assignment problem. *Computers and Operations Research*, 27:917–934, 2000.

R.M. Aiex, S. Binato, and M.G.C. Resende. Parallel GRASP with path-relinking for job shop scheduling. *Parallel Computing*, 29:393–430, 2003.

C. Blake and C. Merz. *UCI repository of machine learning databases*, 1998). (http :/www.ics.uci.edu/ mlearn/MLRepository.html.

I. Guyon and A. Elisseff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, March 2003.

I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh. *Feature Extraction, Foundations and Applications*. Series Studies in Fuzziness and Soft Computing. Springer, 2006.

R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97:273–324, 1997.

H. Liu and H. **Motoda**. *Computational methods of feature selection*. Chapman and Hall/CRC Editions, 2008.

Songyot Nakariyakul and David P. Casasent. An improvement on floating search algorithms for feature subset selection. *Pattern Recogn.*, 42(9):1932–1940, 2009. ISSN 0031-3203.

Mauricio G. C. Resende. Greedy randomized adaptive search procedures (grasp). *Journal of Global Optimization*, 6:109–133, 1999.

Mauricio G. C. Resende, Mauricio G. C, Resende, and Celso C. Ribeiro. Greedy randomized adaptive search procedures, 2002.

Marko Robnik-Sikonja and Igor Kononenko. An adaptation of relief for attribute estimation in regression. In *ICML*, pages 296–304, 1997.

P. Somol, P. Pudil, J. Novovičová, and P. Paclík. Adaptive floating search methods in feature selection. *Pattern Recogn. Lett.*, 20(11-13):1157–1163, 1999. ISSN 0167-8655.

L. Yu and H. Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. pages 856–863. Twentieth International Conference on Machine Leaning (ICML), 2003.