Optimizing Feature Engineering through Induced Hybrid Wrappers, Automatic Thresholds and Filter Ensembling with Rank Aggregation

MSc Research Project
Data Analytics

Ketan Karande
x17100062

School of Computing
National College Of Ireland

Supervisor: Dr. Pramod Pathak
Dr. Paul Stynes
Dr. Dympna O Sullivan

National College of Ireland

National College of Ireland
Project Submission Sheet – 2017/2018
School of Computing

Student Name:	Ketan Navanath Karande
Student ID:	x17100062
Programme:	Data Analytics
Year:	2017-18
Module:	MSc Research Project
Lecturer:	Dr.Pramod Pathak, Dr. Paul Stynes, Dr. Dympna O'Sullivan
Submission Date:	17/09/2018
Project Title:	Optimizing Feature Engineering through Induced Hybrid Wrappers, Automatic Thresholds and Filter Ensembling with Rank Aggregation
Word Count:	6700

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	
Date:	14th September 2018

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
3. Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Optimizing Feature Engineering through Induced Hybrid Wrappers, Automatic Thresholds and Filter Ensembling with Rank Aggregation

Ketan Navanath Karande
x17100062
MSc Research Project in Data Analytics

14th September 2018

Abstract

With evolving big data, the emergence of data dimensionality has surged exponentially. Because of which, researchers are working round the clock to revamp the process of feature selection. The core methods include filters that assign ranks to the features, wrappers that create feature subsets and hybrids that combine the core concepts of relevancy and redundancies from filters and wrappers respectively. Feature ranking is not the only problem since the need of thresholds to limit the number of top-ranked features to be used in the training models is also imperative. These thresholds are dependent on the datasets, termed as fixed or can be determined automatically. This study is speared towards-1. finding best filters for given thresholds, 2. finding conditions in which ensembles of filters are required, 3. finding if the novel approach of creating automatic thresholds is superior to fixed thresholds 4. finding best wrappers 5. testing a novel approach of induced hybrids to achieve relevancy in wrappers and 6. finding the thresholds that causes overfitting. The achieved results prove that the novel approaches introduced in the study have revised the process of feature selection through automatic thresholds that can handle overfitting, through ensembles that boost performance and through induced hybrid that boost relevancy in wrappers.

1 Introduction

Data analytics and predictive modelling is not only about just foreseeing the future but it also shares a great relationship with understanding the data and features in it. In different fields like genetic functioning as mentioned in Liu et al. (2018), image processing as mentioned in Canedo et al. (2015), bio-informatics as mentioned in Yun et al. (2016) etc the type of data is mostly wide rather than being long. This means that the number of features often exceed the number of observations that makes the work of training models even difficult. The traditional approach guides to put all the features in the training model and proceed with the statistical processes. However, the approach of starting the process with pre-selected features and fitting them in the training model is seen to pick up its pace since last few years. But the selection of features and their relevance is often

a very precarious process that varies according to the dataset and the feature selection process used.
The process of feature selection is divided in two main parts namely filter and wrapper. The filter processes are good in assigning ranks to the features according to their importance, however, they often fail to take redundancy i.e high correlation amongst features into account. The wrapper method, on the other hand, creates subset of features that handle redundancy very well but fall short in finding the degree of relevance i.e predictive power of the features. The only way to achieve equilibrium between relevancy and redundancy is to combine these two methods so that one can balance the other.
In this study, an attempt has been made to combine these approaches through a novel idea of finding the induced hybrid which is an optimum wrapper subset deduced from combining ranks of features achieved through different filter methods to achieve subset complexity weight i.e feature weight and best performing threshold. These induced hybrids are designed to maintain parity with relevancy achieved from filters and redundancy achieved through wrappers. The ranks are combined using the means and the principle of stability selection i.e taking the most stable rank of the feature also known as mode in layman terms and they are used to find complexity weights of the subset. The performance of the induced hybrid is tested against individual wrappers to find the best solutions. The process of combining ranks achieved through different filters is termed as filter ensemble. In ensemble, the methods used are all rank aggregation and separated rank aggregation where all rank aggregation is taking means and modes of ranks from all the 12 filters at the same time, while separated rank aggregation is taking means and modes of the ranks of the best performing filters at particular thresholds.
Thresholds guide the total number of features to be used and is important to determine the usefulness of the features with regards to fitting the model and reducing the model complexity i.e number of features used to fit a training model. These thresholds can have different predefined values. But it has been observed that the performance of fixed thresholds is often subjective to the type of the dataset. The fixed thresholds used are log2 and 10-25-50 percent of the total number of features. This study has attempted to solve this threshold subjectivity through a novel approach to create an automatic threshold that depends on performance of different wrapper methods and has seen to work marvels against the fixed thresholds. The problem of overfitting that may arise due to wrong judgment of thresholds is solved by this novel approach of automatic thresholding. The overfitting of the thresholds is explained by the learning curve plots which are explained in the upcoming sections. This learning curve plots help in understanding if the fitted training model performs equally for train and test sets or do they show sign of overfitting.
The aim of the study was to tackle the problems of relevancy, redundancy, usefulness, and overfitting of features through introducing novel approaches of automatic thresholding and induced hybrids along with filter ensembles through rank aggregation. It was aimed at improving the process of feature engineering through robust ensembles, hybridization, and optimized thresholding.
The impending sections of this paper are organized as follows: Section 2 describes the relevant work carried out in the field of feature selection, Section 3 describes the methodology that was undertaken throughout the course of this study, Section 4 focuses on the design and implementation of the study which used the concepts introduced in methodology, Section 5 evaluates the findings achieved through the study,and Section 6 points out the concluding points of the study that may open routes to future work in the field.

2 Related Work

Research on feature selection is an upcoming area of interest and lot of recent studies were targeted on designing new methods to optimize it. The survey of related work in the field is divided in 4 sections specifically focused on particular topics that form the heart of this study. The sections are:

2.1 Automatic thresholds and overfitting

The methodology implemented in this study is strongly motivated by the work done by Seijo-Pardo et al. (2017) and Seijo-Pardo et al. (2019). This researcher strongly believes that with emergence of wide data, the need for developing advanced methods for feature selection is at the highest. According to Hoque et al. (2014) filters provide feature rankings in order of their relevance in predicting the response variable but fail to determine the number of features that are useful in final predictions. Because of which finding right thresholds is crucial. In the research by Seijo-Pardo et al. (2019), the author talks about introducing automatic thresholds that can help in feature selection by using feature complexities to find the feature importance. It stresses on the way in which automatic thresholding can help in reducing redundancies amongst the features and yet choosing the most relevant ones. These automatic thresholds tackle overfitting for every dataset as opposed to the fixed thresholds that are very particular to specific datasets and datatypes. Based on this ideology a progressive thinking has been put to work in this study by creating automatic threshold through use of several wrapper methods and using them with filters to reduce model complexity and achieve better accuracies without causing overfitting. In this study, learning curve plots were used for identifying overfitting thresholds. As mentioned by Zhang et al. (2018), learning curve plots help in detecting the overfits through simultaneous plotting of accuracies achieved through the model for training and the testing sets. If the trajectory of train sets show increment in accuracies at different thresholds as opposed to the test sets showing decrements then the event is termed as an overfit. Such graphs have been plotted for all the filters used for all the datasets to track the overfitting phenomenon.

2.2 Ensembles of feature selection methods

Researches done by Seijo-Pardo et al. (2017),Seijo-Pardo et al. (2019),Rodríguez et al. (2018),Wang et al. (2010) establish the fact that ensemble in feature selection methods can also improve the quality of relevant feature identification. Rodríguez et al. (2018) explores this ensemble ideology by considering feature selection methods like classification training models which classifies the feature into 2 classes namely "Important" and "Not Important". This perspective of looking at feature selection methods clearly widens the ways in which features and subset of features are developed. In this study, an ensemble of 12 different filter-based methods and 8 different wrapper-based methods has been deployed to develop automatic thresholds and hybrid wrappers that can merge relevancy with redundancy and enhance the model performance. Thus, this study introduces 2 novel approaches for calculating wrapper complexities through induced hybrids and filter ensembles and designing automatic thresholds through wrappers.

2.3 Ensembles with rank aggregation

Over the years there have been many unique ways designed to create ensembles for feature selection, especially through rank aggregation. Researches by Yun et al. (2016), Prati (2012),Dittman et al. (2013),Brancotte et al. (2015),Smetannikov et al. (2017),Yoon et al. (2005),Nassif et al. (2017),Chatterjee et al. (2018),Seijo-Pardo et al. (2019) represent studies from various fields like bio-informatics and genetic coding on multiple dataset with varied features and classes and applications of model ensembles. Amongst this studies Brancotte et al. (2015) talks about the use of Borda count method to handle rank aggregation for features who are placed at same ranks by multiple filter methods. This concept of rank aggregation is tested and proved by their application on multiple datasets by Prati (2012),Dittman et al. (2013),Seijo-Pardo et al. (2017),Rodríguez et al. (2018) and it has been seen for majority that the concept of aggregating ranks of features acquired through different methods and then using new ranked features limited with thresholds to train the model actually works. Identical to this approach, this study combines all the feature rankings by using stability selection and means to find the combined order of features.

2.4 Hybridization in feature selection

The methods of feature selection are filter and wrapper with first being suitable for feature ranking and the latter for creating feature subsets. The filters in feature selection are known to handle relevancy but fail in considering redundancy and are independent of the learning models, leading to faster computing abilities as backed by Morán-Fernández et al. (2017). However, the wrappers, as mentioned in Chandrashekar and Sahin (2014), and Rodriguez-Galiano et al. (2018), help in creating feature subsets according to redundancies but fail to consider relevancy and are dependent on the learning models leading to slower computing abilities. In order to tackle these operational flaws of filters and wrappers, researches by Li-Yeh Chuang and Yang (2008),Smetannikov et al. (2017),Liu et al. (2018),Wang and Feng (2018) introduce the concepts of hybrid models that combine these methods of feature selection and deliver features with higher relevant ranks and lower redundant subsets. As per the researchers Vora and Yang (2018)and Zainudin et al. (2017), hybrids reduce the curse of dimensionality by benefitting from the pros and cons of filters and wrappers. Study done by Smetannikov et al. (2017) shows how hybrid models can be used on metadata to extract important and limited features and reduce time of model comprehensibility without causing overfitting issues. As opposed to this Nakariyakul (2018) works on creating a hybrid model based on the interaction effect of the variables but lacks in dealing with datasets with less number of features. In this study, a novel approach of induced methods has been used to replicate the advantages of hybrid models through combining feature ranks from filter methods to calculate total complexity weights of wrapper subsets. This method sneaks a slight hint of hybrid models to achieve the best subset which is more lighter than other subsets, more comprehensible and more accurate. In this subtle way of hybridization, it was aimed at identifying the best wrappers that can handle relevancy and redundancy for a particular dataset. Research by Hancer et al. (2018) refers to using top ranking features acquired through filters to achieve max relevancy and min redundancy. But this approach of using top ranking features limits itself to filters and has been handled in this study by introducing top ranking features for calculating complexity weights in wrappers.

2.5 Summary of the survey

After scrutinizing the survey and using Seijo-Pardo et al. (2017) and Seijo-Pardo et al. (2019) as the benchmark, it can be said that a lot of work has been done on developing ensembles for feature selection but the combination of filters used in them are limited to 5-7 and hence are not able to take advantage of other filter methods. In contrast, in this study, a combination of 12 different filters has been used to create the ensembles. Many researches mentioned in previous section have put good efforts to design hybrid models that can balance the relevancy and redundancy, but none has used the novel approach of calculating wrapper complexity through ensembles of filter methods, that is introduced in this study. The development of automatic thresholds is currently the prime topic of research in the field of statistics. Researchers have developed ways to create automatic thresholds, but have failed to use the benefits of wrappers to create feature subsets. This study introduces a novel approach of using wrapper subsets to find automatic threshold through implied weighted aggregation. Thus, the undertaken study through its novel approaches aims at achieving success in designing automatic thresholds and robust hybrid models derived through different rank aggregation methods.

3 Methodology

In this study, a 2 stage approach is undertaken that solves the 6 research questions. This approach includes the use of 12 different filters and 8 different wrappers along with 4 different approaches of feature ensembling.
The filter methods excel in computing time and finding relevant features but lack in tackling redundancies and determining feature thresholds. While the wrapper methods often excel in tackling redundancies and creating thresholds but lack in finding relevancy of the features and computing times. The 12 filters namely boruta, fscaret, chi-square, information gain, gain ratio, symmetric uncertainty, oneR, random forest (accuracy), random forest (impurity), relief, caret and maximum relevancy minimum redundancy (mRMR) that are extensively used in this study are combined through various ways like all rank aggregation and separated rank aggregation and limited through novel approaches like automatic thresholds. The 8 wrappers used in this study are best first search, exhaustive search, forward search, backward search, hill climb search, correlation feature selection, consistency, and SVM-recursive feature elimination.
Owing to these advantages and limitations of filters and wrappers, the hybrid models are designed to choose the features by balancing their respective relevancies, redundancies and computation times. They develop better feature selection environment by creating subsets of relevant features with min redundancies amongst them. This study aims at using these concepts of filter, wrapper, hybrids and introduces 2 more concepts of ensembles and automatic thresholds to improve feature selection process
The methodology of the study is focused on answering the following research questions not necessarily in the stated order
1.Finding the filter methods that works best at a particular threshold
2.Finding if the novel approach of automatic threshold outperforms the traditional approach of fixed thresholds and tackles the overfitting issues

3.Finding the conditions in which the performance of the ensemble of filter methods is superior to individual methods
4.Finding the best wrapper that sustains throughout different datasets and successfully outperforms other wrapper models
5.Finding if the novel approach of induced hybrid wrapper is superior to other wrappers and if it can achieve max relevancy and min redundancy
6.Finding automatic thresholds through weighted aggregation on wrapper subsets
The methodology adopted in this study to solve these questions is explained through the following sections

3.1 Filters, Ensembles and thresholds

Figure 2 illustrates the procedure to answer the 3 questions related to performances of filters and ensemble of filters at different thresholds including the issues related to automatic/fixed thresholds and overfitting. The process starts with step 1 and 2 were all the 12 filters as mentioned in the figure 2 are implemented on the dataset and their individual rankings are noted. The achieved feature rankings are combined using all rank aggregation and separated rank aggregation techniques as mentioned in steps 3 and 4. These methods represent filter ensemble in this study. These ensembles are done in 4 different ways.

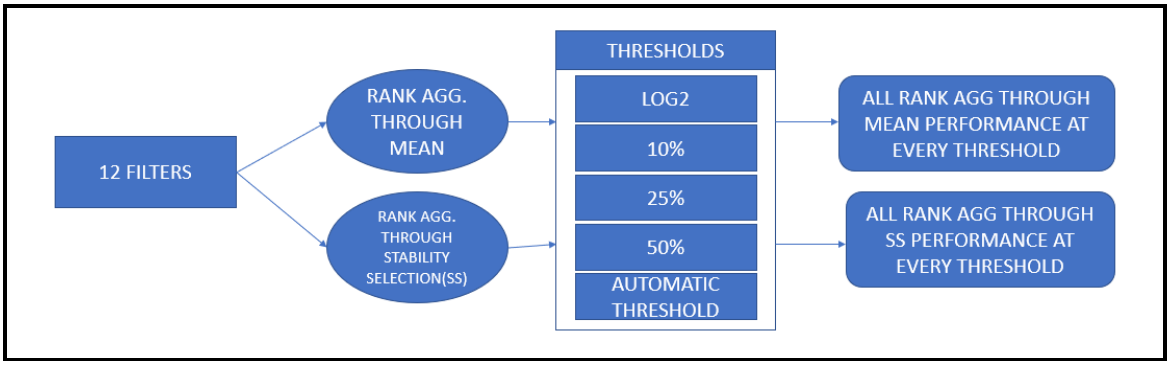


Figure 1: ALL RANK AGGREGATION ENSEMBLE

The first 2 ways as mentioned in figure 1 are performed through aggregating the ranks of features and selecting the stable ranks of features made available through all the 12 filters. The other 2 ways of ensembling as mentioned in figure 3 termed as separated rank aggregation only combines the filters that have shown good performance for a particular threshold. Through this separated rank aggregation method, a total of 10 different ensembles is achieved with each working optimally at a particular threshold. Out of these 4 different rank aggregation methods shown in figure 1 and figure 3, a single method is adopted that guarantees optimal performance. Further, at step 5 in figure 2 the individual as well as the optimal ensemble derived filters are downsized by using the calculated thresholds. These downsized rankings are later fitted using support vector machines as the training model in step 6 to inscribe their performances. The accuracies achieved through individual and ensembled filters are later compared in steps 7, 8, and 9 to produce the final results.

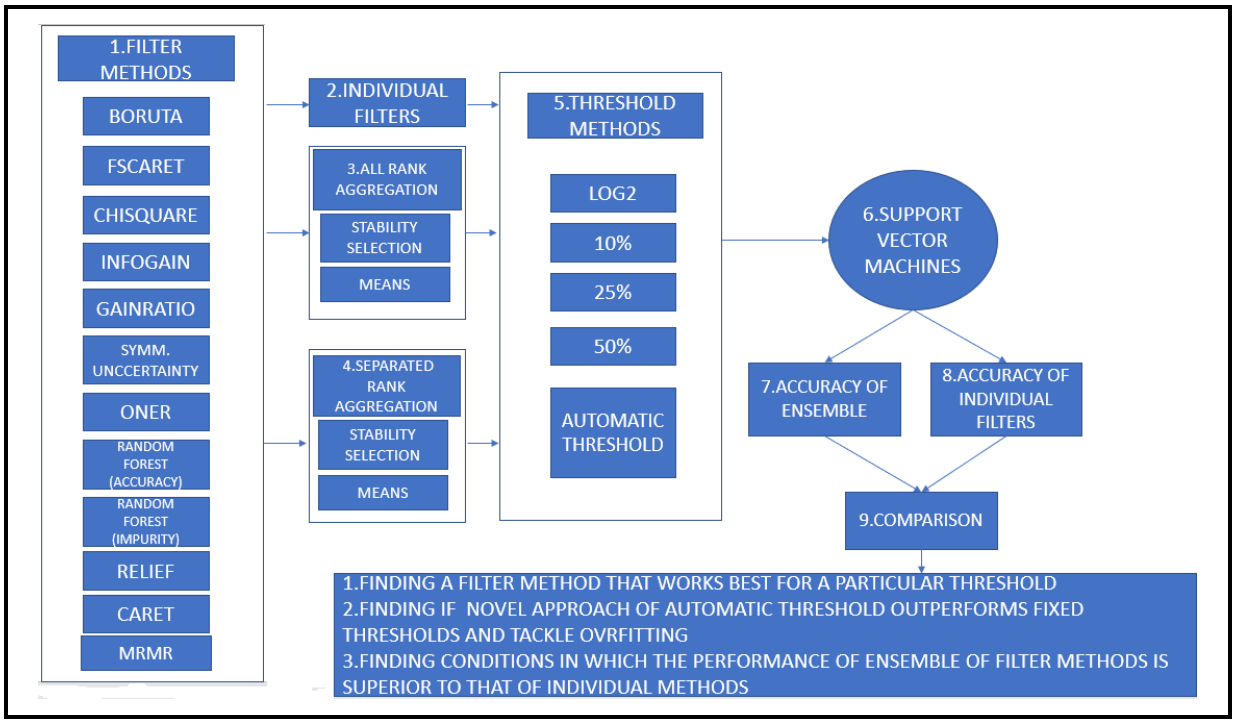


Figure 2: FILTERS

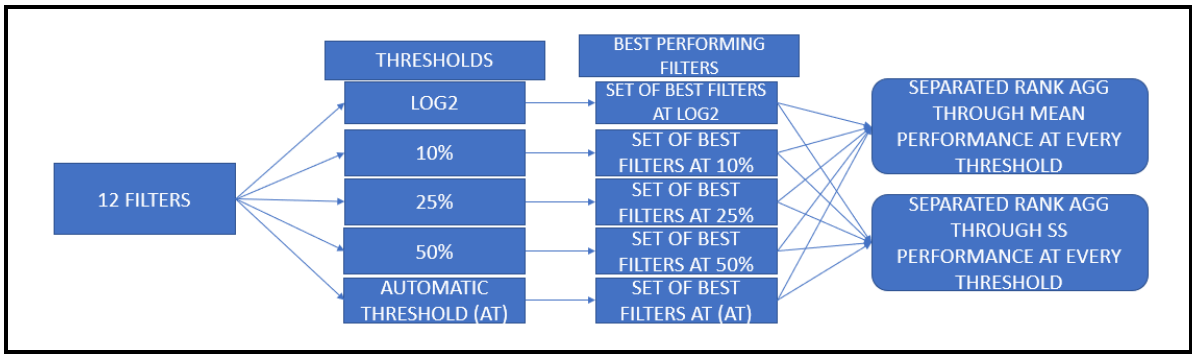


Figure 3: SEPARATED RANK AGGREGATION ENSEMBLE

3.2 Wrappers and Hybrids

Figure 5 illustrates the procedure to answer the 3 questions related to performances of wrappers, induced hybrids, and designing automatic thresholds. In this approach, different wrappers are used to create feature subsets as mentioned in step 1. Then the subsets are assigned weights with respect to their performances when implemented with SVM. Better the accuracy, higher is the weight. Using this weight and the subset length, the weighted average is calculated in steps 2 and 3, thus finding the automatic thresholds. This process is clearly highlighted in figure 4

Following this, based on the ranks achieved in optimal ensemble in figure 1, figure 2 and figure 3, the subset complexity weight is calculated in step 4 that gives information about the features added by different wrappers in their respective subsets and their total

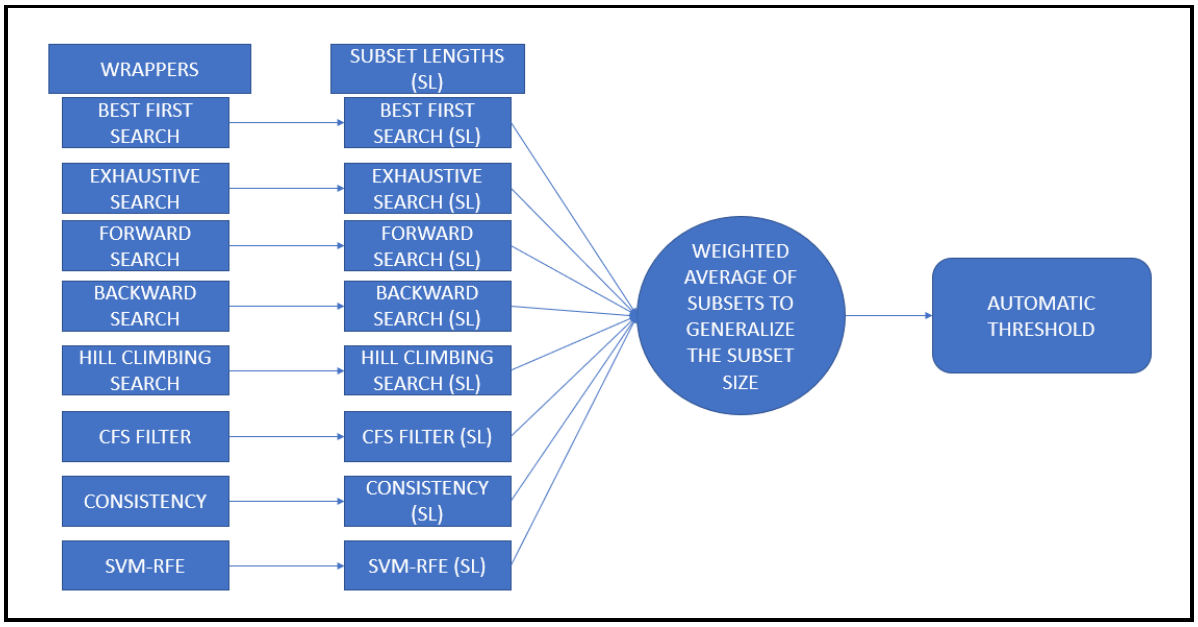


Figure 4: AUTOMATIC THRESHOLDS

rank sum. Further, the weight and length of the subsets is compared to determine the induced hybrid in step 5 and as represented in figure 6. This induced hybrid helps in finding the wrappers that are successful in managing relevancy along with their nature of tackling redundancy for a particular type of dataset and is represented by the lowest subset complexity weight. After determining the hybrid wrapper, it is compared with the remaining wrappers in step 6 to validate if the novel approach of hybrid wrapper is able to balance relevancy and redundancy by outperforming the other wrappers. Thus determining the best wrapper in step 7.

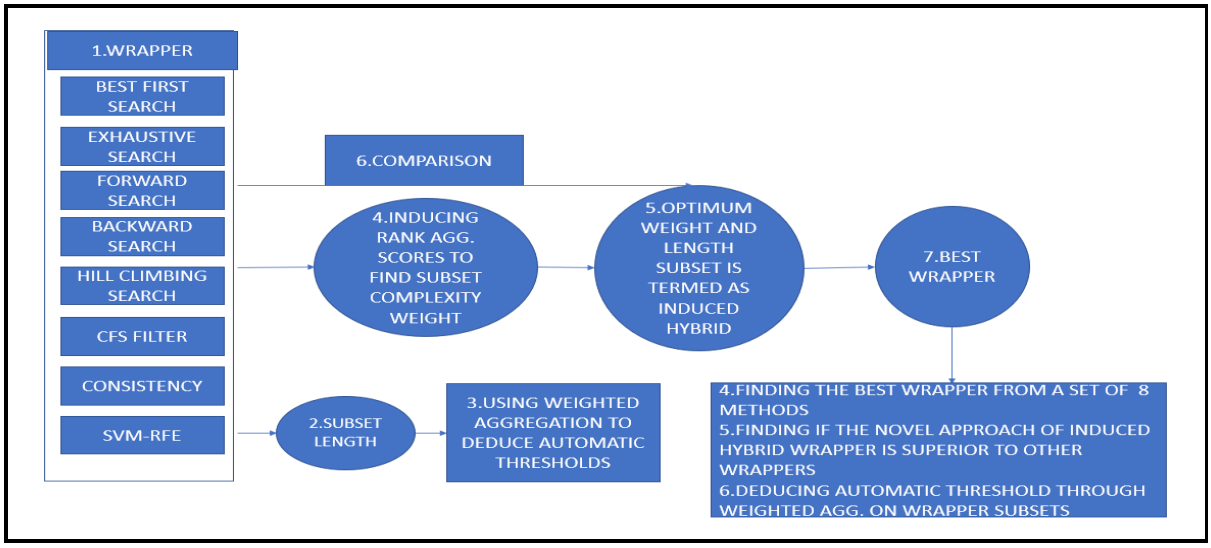


Figure 5: WRAPPERS

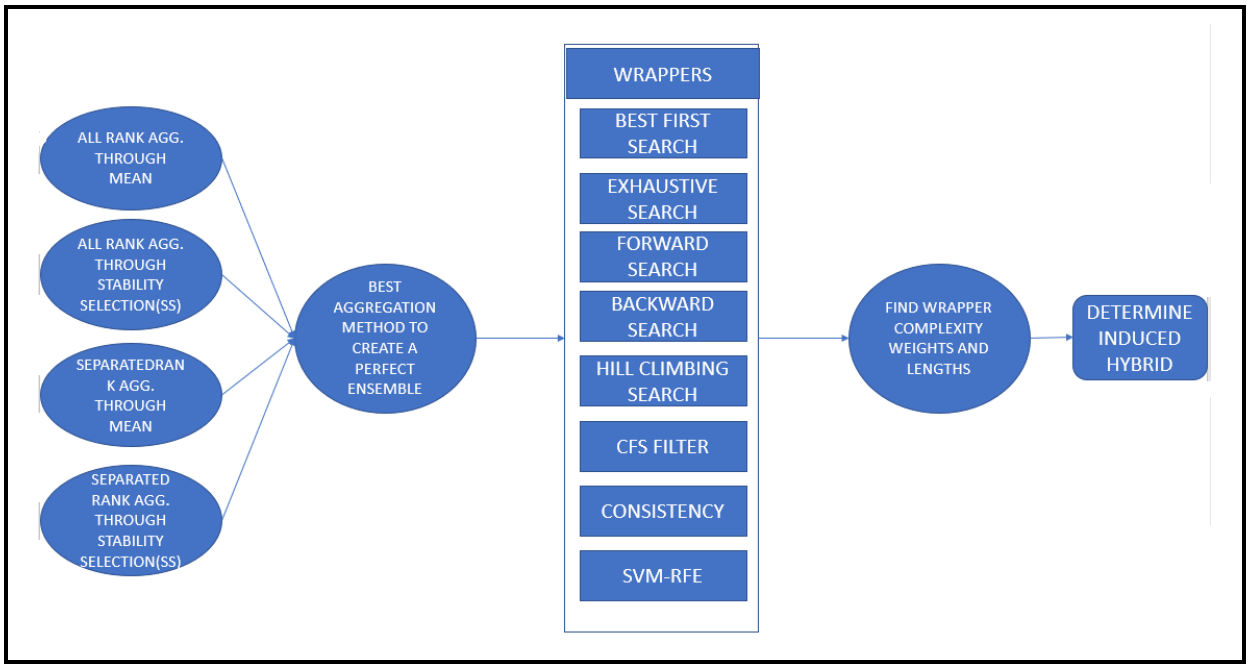


Figure 6: INDUCED HYBRIDS

3.3 Brief of the methodology

Following the mentioned sections, the study is carried forward for implementation. The methodology explained through aforementioned design figures are inspired from the work done in the benchmark studies. The novelty of the study lies within the development of automatic thresholds and induced hybrids through extensive use of filter and wrapper methods as mentioned earlier and their ensemble through varied rank aggregation methods. This methodology, although being very extensive has been specifically used to overcome overfitting issues in feature selection, pinpoint the needs for ensembles, and lack of relevancy tracking in wrappers. Using this methodology designs, the implementation was executed as mentioned in the next section.

4 Implementation

In this study, a total of 10 different datasets with feature size ranging from 9 to 857 have been subjected to rigorous pre-processing required for feature selection. These datasets are taken from the UCI ML repository [1] [2] [3] [4] [5] [6] [7] [8] [9] [10] out of which the first 5

[1] https://archive.ics.uci.edu/ml/machine-learning-databases/yeast/
[2] http://archive.ics.uci.edu/ml/machine-learning-databases/spambase/
[3] http://archive.ics.uci.edu/ml/machine-learning-databases/madelon/
[4] http://archive.ics.uci.edu/ml/machine-learning-databases/connect-4/
[5] http://archive.ics.uci.edu/ml/machine-learning-databases/isolet/
[6] http://archive.ics.uci.edu/ml/machine-learning-databases/ionosphere/
[7] http://archive.ics.uci.edu/ml/machine-learning-databases/musk, clean/
[8] http://archive.ics.uci.edu/ml/machine-learning-databases/00282/
[9] http://archive.ics.uci.edu/ml/machine-learning-databases/spectrometer/
[10] http://archive.ics.uci.edu/ml/machine-learning-databases/00233/

datasets are taken directly from the study by Seijo-Pardo et al. (2017) for research purposes.
Table 1 notates the important details like observation and feature size of the datasets used in this study.

DATASET NO.	DATASET NAME	NO. OF OBSERVATIONS	NO. OF FEATURES
1	Yeast	1484	9
2	Spambase	4601	58
3	Madelon	4400	501
4	Connect4	28543	43
5	Isolet	7797	618
6	Ionosphere	351	34
7	musk, clean	476	168
8	Voice	126	309
9	Spectrometer	531	102
10	CNAE	1080	857

Table 1: DETAILS of USED DATASETS

Since the motive of the study is to design new ways to optimize the process of feature selection, the datasets chosen have varied number of features, that have given the study a broad platform to experiment over different filter and wrapper methods and to identify their dependencies. Each dataset is subjected to the basic outlier and missing value treatments along with label and dummy encodings for a few. This steps beside being the core essentials have also helped in pointing the algorithms towards the desired outcomes.
The machine used for this study had a Intel(R) i7-7500U CPU 2.7 GHz x64 based processor with 16 GB RAM. This experiment was performed using the statistical tool R with certain packages to be installed as the prerequisites. The packages include fselector, fscaret, caret, boruta, random forest, dplyr, mlbench, mRMRe, and e1071. This packages have inbuilt filter and wrapper methods along with provisions for deploying training models like support vector machines that has been used in this study as well.
The programming codes are created according to the methodology mentioned in the above section and all the 10 datasets are subjected to it. Since the process of this study includes heavy processing and extensive datasets, it was necessary to store the results of the coding as you proceed with them. These results are later summarized to answer the 6 research questions that the study has focused on solving.
In general, the implementation of the study follows the following design. The designs mentioned in the methodology are used for calculating its core components like rank aggregation., automatic threshold, and induced hybrids.

As per figure 7, the dataset begins with outlier and missing value treatments followed by stratified sampling in steps 1, 2, and 3. Since all the datasets are for classification, it's important to have stratified splits for maintaining consistency of data balance in the classes of the response variable. After this, different fixed thresholds are calculated based on the number of features in the dataset which is then subjected to feature selection

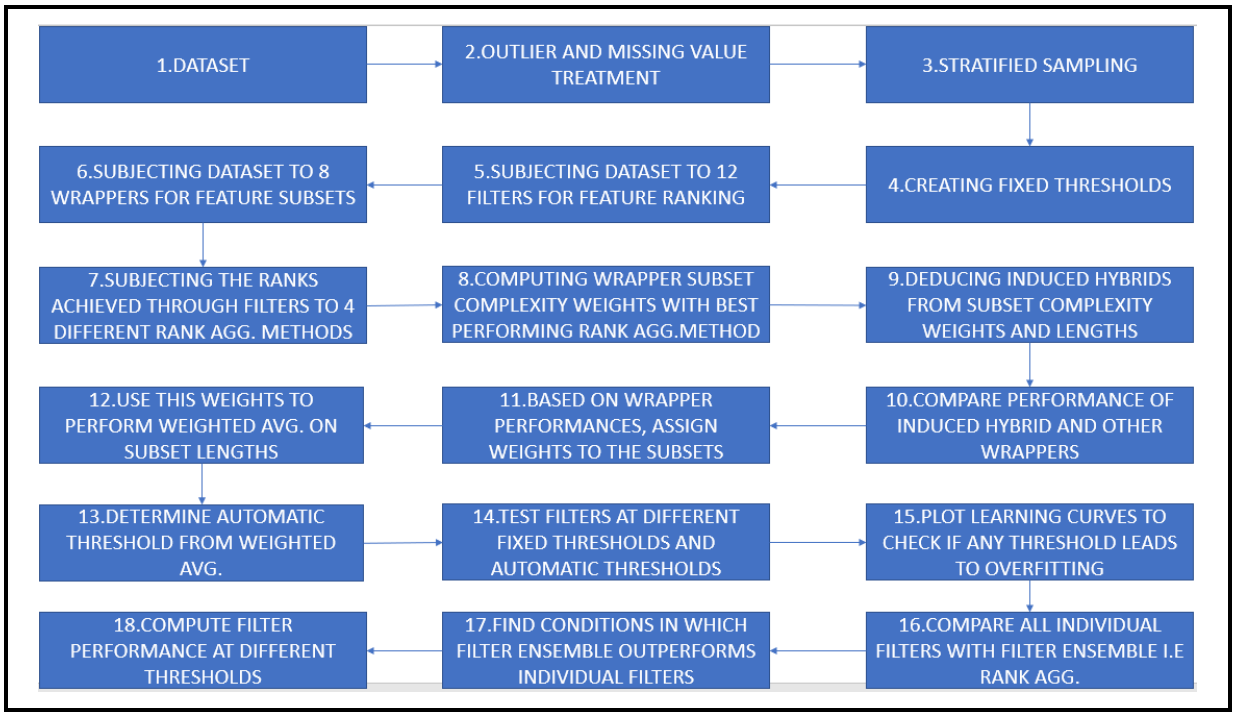


Figure 7: IMPLEMENTATION DESIGN

techniques through 12 filters and 8 wrapper methods as mentioned in steps 4, 5, and 6. The ranks of features achieved through the 12 filters are combined using 4 different rank aggregation methods to ensure perfect ensembling in step 7. Out of this 4 methods, the best ensemble is chosen to assign complexity weights on the subsets achieved through 8 wrapper methods in step 8. Using this complexity weights and subset length, the novel approach of induced hybrids is performed in step 9. This hybrid model is the 1 /8 wrappers that is able to balance relevancy along with redundancy as opposed to other 7 wrappers that are just tackling redundancies. This induced hybrid wrapper is later compared in step 10, with performances of other wrappers in order to validate the applicational reliability on induced hybrids. Following this process, the subsets are assigned with weights based on their respective performance in step 11. Better the performance, higher is the weight assigned. With the help of this weights and subset lengths, the weighted average of the subsets is calculated to finalize on the automatic thresholds in steps 12 and 13. Based on this automatic threshold and other fixed thresholds, the filter methods get tested for performance and overfitting to find the optimum set of filters and the best performing thresholds in step 14. After achieving all the thresholds, a total upto 14 graphs per dataset were plotted to check if any overfitting occurs at any particular threshold in step 15. These graphs are termed as the learning curves since they help in understanding the model performance on the training set as well as on the test set. The individual filters are also compared with the filter ensembles to find the conditions in which ensemble is preferred as opposed to usage of individual filter methods for feature ranking in steps 16 and 17. Finally, the threshold wise performance testing is executed to find dependencies of filters and thresholds on the 10 different datatypes in step 18.
The combined results achieved through above-mentioned process helps in explaining the 6 research questions which are to find the best filters at different thresholds, to find conditions which require filter ensembles, to find if automatic threshold outperforms fixed

thresholds without overfitting, to find best wrappers, to find if induced hybrids outperform other wrappers and to deduce the automatic thresholds.

5 Evaluation

Throughout the course of this study, a variety of observations were made and recorded. Some of these observations are already acknowledged in different researches and can be used in proving the fact that certain feature selection techniques that were used in the past may become the bygones in future world of big data. These observations include the inefficiency of wrapper methods like exhaustive search to handle data with higher dimensionality in acceptable computational time, failure of wrappers like SVM-RFE to handle more than 40 classes in classification, incompetence of filters like fscaret in accrediting ranks to every feature for high dimensional data, inability of filters like from caret package to handle high dimensionality and an overall heavy computation time required for wrappers to construct feature subset for wide datasets.
In conjunction with these familiar observations, it was also observed that the novel approaches of automatic thresholds and induced hybrid worked perfectly with help of the rank aggregation methods used. General findings of the study include the failure of the stability selection of rank aggregation method in assigning unique ranks to every feature which made this method obsolete for datasets with higher dimensions along with some fixed thresholds. Hence, this method was not able to find its place in the final results.
In order to make the results more understandable, they have been summarized in a way of situational conclusions or limited applications so that they can be benefited in certain situations more than the usual ones. For instance, certain filters, wrappers, thresholds and filter ensembles are more suited and suggested for particular group of datasets as compared to others. The results compiled through this study helps in finding such dependencies and reliabilities. These outcomes are summarized from the findings elucidated in figure 9, figure 10, table 8, table 9, table 10, table 11, table 12, table 13, table 14, table 15, table 16, table 17, table 18, and table 19 that are acknowledged in the the AppendixA, AppendixB and AppendixC of this paper. Specific study findings are outlined in the next subsections.

5.1 Essential information to interpret the results

In order to make the results more fathomable, datasets referred for this study are compiled in groups as mentioned in table 2 along with the abbreviations mentioned in the table 3 and table 4 used for compiling all the filter and wrapper methods used.

The datasets with feature size 1 to 10 has been grouped under group 1 followed by feature size of 11-100 in group 2, feature size of 101-200 in group 3, feature size of 201-400 in group 4, feature size of 401-600 in group 5, feature size of 601 and more in group 6. These groups were created to forge a guideline for any analyst to decide over the preprocessing steps based on the datasets.

NO. OF FEATURES	GROUP NAME
1-10	G1
11-100	G2
101-200	G3
201-400	G4
401-600	G5
601+	G6

Table 2: DATA TYPES BASED ON NUMBER OF FEATURES

FILTER	ABBREVIATION	FILTER	ABBREVIATION
Boruta	B	OneR	O
Fscaret	FS	Random Forest(Accuracy)	R1
Chi-Square	CH	Random Forest(Impurity)	R2
Information Gain	I	RelieF	R
Gain Ratio	G	Caret	C
Symmetric Uncertainty	S	max relevant min redundant	M
Separated Rank Agg.	SR	All Rank Agg.	AR

Table 3: FILTER ABBREVIATIONS

FILTER	ABBREVIATION	FILTER	ABBREVIATION
Best First Search	BFS	Hill Climb Search	HCS
Exhaustive Search	ES	correlation feature selection	CFS
Forward Search	FS	Consistency	CONS
Backward Search	BS	SVM-RFE	SVM-RFE

Table 4: WRAPPER ABBREVIATIONS

5.2 Filters and Ensembles

The upcoming table 5 can be referred to answer one of the research questions of finding the best filter method for a particular type of dataset.

Table 5 shows the dataset groups and the best performing filters at different thresholds in these groups. This answers the first research question of finding best filters at different thresholds, that this study was focused on solving. The filters that failed to assign ranks to the features were exempted from the study. This failed filters are mentioned in the tables in AppendixB section.If the dataset lie in the groups, any of the mentioned filters in that group can be used for feature selection at that threshold level. It can be seen that the automatic threshold has ensembled filters (Rank Aggregation) as the best.For instance, at automatic thresholds data groups like G1, G2, G4, and G5 are not suited for using individual filters since filter ensembles are suggested. In these situations, using more than a single individual filter and forming ensemble of filters through rank

aggregation can deliver better results. This can guide the user about the applicational suitability of individual filters or ensemble filters best suited in that scenario. It can be presumed that in situations where the dataset lies within the mentioned group with no suggestion to use only ensemble of filter methods like the highlighted entries in table 5, any of the best performing individual filter can be used that is suitable for the analyst. For instance, it can be seen that the filter method oneR is suitable to be used at log2 thresholds for dataset belonging to groups G1, G2, G3, and G6. For situations like 10 percent thresholds in group G2, the number of choices is more to choose between filters. In such case, the study recommends using any of the suggested filters as per the user's ease.

Groups	Log2	10	25	50	automatic
1-10(G1)	O	B,R,SR,AR	O	R	ALL,SR,AR
11-100(G2)	O,R1,C,G	CH,I,S,O,R1,R2,G,C	S,C,R2	R,G,AR,CH	G,S,AR
101-200(G3)	O	O	B,R2	B	B,R1
201-400(G4)	C	R2,I	B	R2,AR	R2,AR
401-600(G5)	**SR**	CH,I,G,S,O	FS,AR	B	**AR**
601+(G6)	O,R2	R2	B,R1	R1,AR	R1,R2

Table 5: BEST FILTERS

Groups	Log2	10	25	50	automatic
1-10(G1)		SR,AR			**SR,AR**
11-100(G2)				AR	**AR**
101-200(G3)					
201-400(G4)				AR	**AR**
401-600(G5)	SR		AR		**AR**
601+(G6)				AR	

Table 6: SITUATIONS TO USE FILTER ENSEMBLES

Table 6 describes the conditions in which separated and all rank aggregation methods for filter ensembles is to be used.It guides the conditions in which ensemble of filters should be used which answers the second research question that the study is focused on solving. This table forms the sub-table for table 5. It highlights the dataset groups that required filter ensembles for optimum performance at different thresholds. for instance, If dataset lies in G5 and business demands 25 percent thresholding then All Rank Aggregation method can be used to get the best result. As seen in the highlighted entries in table 6 and the discussion made earlier, the novel approach of automatic threshold introduced in the study demands filter ensembles for optimum performance for 4 out of 6 dataset groups namely G1, G2, G4, and G5. It was also observed that amongst all the datasets, the ensemble method of separated rank aggregation didn't cause any kind of overfitting which commends about its perfection in suggesting most relevant and useful features in their order of their importance.

5.3 Wrappers and Hybrids

Table 7 guides the users about the best wrappers that can be used at different dataset groups. This result helps in answering the fourth research question of finding the best wrapper method, that the study aims to find. These results are extremely co-dependent on the number of features present in the dataset. Since wrapper methods are mostly search methods, there time of process execution exceeds the time lines. Because of this, some wrappers like exhaustive search that can take long computational time at different situations and for different data types were specifically removed from evaluation. The exempted wrappers are mentioned in the AppendixC section. Besides this, table 7 provides an amazing insight and guide to use different wrappers based on the dataset group.

1-10(G1)	11-100(G2)	101-200(G3)	201-400(G4)	401-600(G5)	601+(G6)
BFS	**CFS**,HCS	CONS	**CFS**	**FS**	**FS**,BS

Table 7: BEST WRAPPERS

From highlighted entries in table 7, it can be said that the CFS wrapper performed best for 3/10 datasets with 2 in G2 and 1 in G4 and FS wrapper for 2/10 datasets. So, it can be suggested that when dataset lies in G2 and G4, the CFS wrapper should be used for creating feature subsets and if the dataset lies in G5 and G6, the FS wrapper should be used.
Along with finding the best wrappers, the study also aims at testing the novel approach of induced hybrids to find out best wrappers that not only are able to tackle redundancy but also handle relevancy. This forms the fifth research question of finding if induced hybrids are superior to other wrappers, introduced in the study. Through imputation of achieved ranks through best performing ensembles, different performance weights were assigned to the features. As different wrapper suggested different subsets, these weights were used to find the subset complexity weights which helped in finding the best subsets that had features in their subset which had higher ranks of performance and hence handled relevancy. It was observed that the induced hybrid wrappers outperformed other wrappers for 4 out of 6 dataset groups namely G2, G3, G4, and G6. The most common wrappers that were deduced as the induced hybrids were CFS and HCS which combined outperformed 6 out of 10 datasets. This observation when compared with table 7 it can be said that the wrappers that were deduced as the induced hybrids managed to be the best or the second-best wrappers as compared to all the other wrappers. This observation confirmed the ideology that wrappers which can handle redundancy along with relevancy can outshine the other wrappers. It was also calculated that if CFS wrapper was deduced as the induced hybrid for a particular dataset then 67 percent of the times it will outperform other wrapper methods. Thus, finding the novel approach to be successful in its assumptions.

5.4 Automatic thresholds and Overfitting

After evaluating the performance of all wrappers, automatic thresholds were designed for all the 10 datasets as explained in the methodology figure 4. This answered the sixth

question of the study about deducing automatic thresholds. After achieving this automatic thresholds, they were used with the fixed thresholds to limit the relevant features from the filters. Out of all the fixed thresholds and the automatic thresholds created for every dataset, the performance of automatic thresholds was much better than the performance of fixed thresholds. 8 out of 10 datasets had their automatic thresholds as the best threshold without causing any overfitting. Thus, it was concluded that automatic thresholds were able to give better accuracies for 80 percent of the times as compared to the fixed thresholds which proved the success of the considered novel approach. These observation helped in answering the third research question that this study strives to find. It was also noted that automatic threshold not only outshined the fixed threshold but also reduced the model complexity which made the model easier to comprehend.
Overfittings were evaluated using the technique of learning curves. In learning curve plots, the training and testing sets were evaluated with the fitted models and their accuracies were plotted. This plot helped in understanding if the achieved accuracies show opposing patterns to one another or not. Overfitting for a dataset can be understood from the difference in such patterns. For instance, if the model performance seems to increase for the train sets but decrease for the test sets then it can be said that the model is overfitting. Learning curves helps in understanding if the model complexity used in the predictive modelling is suitable for the new data or not. This quality of the learning curves was used in this study to ascertain that the novel approach of automatic thresholds holds good in offering better accuracy along with maintaining the fact that the model does not overfit in new data. This step was performed to get assurance on the applicational reusability of the automatic thresholds for unfamiliar data. Along with this, it was also detected that the fixed thresholds like 50 percent was responsible for overfitting in most of the scenarios which had issues related to model complexity and model fitting. Graphs mentioned in AppendixA shows the overfitting scenarios that were perceived during this study.

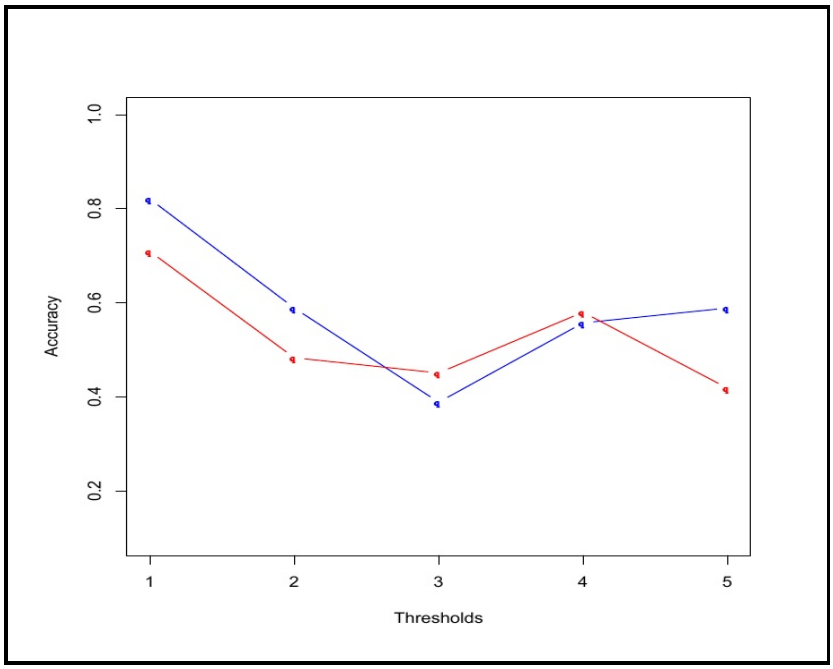


Figure 8: Dataset 8

In the mentioned graphs in figure 9 and figure 10 in AppendixA and figure 8, the X-axis represent the threshold values that were used in the study. These threshold values are represented on the scale as 1 for LOG2, 2 for 10 percent, 3 for 25 percent, 4 for automatic thresholds and 5 for 50 percent threshold. The Y-axis represent the accuracies achieved by different filters for their respective train and test sets. The legends used in the graphs include blue color for training sets and red color for testing sets.Each of these graphs justify the claim that 50 percent fixed threshold lead to maximum overfits in this study for different datasets namely dataset 3(Madelon), 6(Ionosphere), 7(musk, clean) and 8(Voice). The difference in line trajectory of increment and decrement of blue and red lines respectively at 50 percent threshold in each of this graphs indicate overfits. For instance figure 8 for dataset 8 shows how thresholds upto automatic threshold i.e 4 on the X axis work optimally but overfits at 50 percent i.e 5 on X axis causing the training(blue) accuracy to surge and testing(red) accuracy to drop. Similar to this, all the other learning curve graphs mentioned in AppendixA show analogous behaviour and validate the reliability of automatic thresholds from the standpoint of model performance, complexity and overfitting.

5.5 Discussion and Findings

Through the span of this study, the main focus was kept on achieving better filter ensemble limited by better thresholds and finding the conditions in which using ensembles is recommended. Research by Seijo-Pardo et al. (2017) worked primarily on a comparative study with homogeneous, heterogeneous ensembles and there comparison with individual filters. It was observed that in some cases of heterogeneous ensembles, the individual methods outperformed the ensembles. But this cases where not summarized according to the fixed thresholds chosen and number of features because of which ensemble dependencies on datasets were not understood. The study also used 5 different filters which were just ensembled with single ensemble method and restricted through only fixed thresholds. This experiments were performed on a set of 7 datasets with feature size randomly selected. As opposed to this, the undertaken study was performed on 10 datasets with steadily surging number of features, with 12 filters and 4 different ensemble methods. The performance of the ensembles were summarized as mentioned in results and AppendixA and AppendixB along with successful implementation of a novel approach of automatic thresholds as opposed to only fixed thresholds. This automatic thresholds were designed through use of 4 different wrapper methods which formed the unique aspect of the study as compared to its benchmark research by Seijo-Pardo et al. (2017). Apart from taking the motivation from this benchmark research, this study was able to achieve results that were developed on broader grounds in terms of datasets chosen, variety in filter methods, introduction of automatic thresholds and variety of wrappers with design of induced hybrids. Because of this broader consideration, this study really puts on a progressive shift in feature selection process and enhance certain factors mentioned by Seijo-Pardo et al. (2017).

The achieved results when summarized were able to provide evidence for the following claims. This claims forms the novel part of this study which differs from the benchmark study

1. Weighted average of 6 wrappers created the best automatic thresholds that outperformed all the other fixed thresholds

2. The novel approach of automatic thresholds outperformed the fixed threshold for 80 percent of the times and were not seen to lead to any kind of overfitting for the datasets from all the groups
3. Fixed threshold of 50 percent i. e 0.5*N, N is no. of features were more likely to cause overfitting as compared to other thresholds as displayed by the learning curve plots. This result helps in choosing the thresholds while fitting the model and creates a benchmark rule to choose features not only through achieved accuracies but also through model complexities. Thus, making models easy to comprehend
4. Separated Rank Aggregation did not cause any overfitting for any of the datasets which prove its reliability in assigning deserving ranks to the relevant features. This ensemble method formed the part of one of the unique approaches used in this study.
5. At automatic thresholds, Ensemble of Filters works best for 67 percent of the times as compared to individual filters. This provides a good guiding principle for the use of automatic thresholds and filter ensembles
6. The novel approach of induced Hybrid was the best or the second best performing wrapper for 60 percent of the times. This ensures that the ideology used behind this approach to find wrappers that can tackle relevancy along with redundancy can outperform wrappers who can only tackle redundancy
7. At groups G2 G4 there is a 60 percent probability that CFS is the best wrapper and at groups G5 and G6 there is a 67 percent probability that FS is the best wrapper. This creates a guiding principle for the use of wrappers based on the feature size of a dataset.
8. If Induced Hybrid was deduced to be CFS then there is a 67 percent probability that it will be the best wrapper. These results confirm that CFS wrappers can handle relevancy and redundancy better than the other wrappers.

6 Conclusion and Future work

Throughout the study, the experiments were focused on achieving the best performing feature selection techniques and creating an environment that can amplify their competence in finding the best features. It was noticed that the filters and wrappers, when used together, can help in contrasting and improving each other's abilities to detect features. The achieved conditions suitable for using rank aggregation and filter ensembles definitely helped to optimize the relevancy of the features. The novel approach of induced hybrids was successful in identifying the ability of wrappers to tackle relevancy along with redundancy. The novel approach of automatic thresholds was successful in achieving maximum model usefulness along with handling model complexity without causing overfitting. The learning curve plots introduced in the study helps in understanding the thresholds that caused overfitting and hence offers a benchmark to choose the number of features. So, to conclude the study results, it can be said that
a.filter ensembles, when used with automatic thresholds, can give the best results in feature selection by filters through deploying less complex models without overfitting and
b.induced hybrids identified from a set of wrappers for a particular dataset when used for feature subsets can give best results as compared to other wrappers.
The novel approaches of automatic thresholds and induced hybrids, used in the study proved to work perfectly and optimally as compared to the traditional approaches of feature selection along with providing the conditions in which use of filter ensembles is recommended as opposed to the use of individual filters. Thus, the study sturdily prom-

ises the process of feature engineering through a guided path of execution with assured results.
Due to the extent of the study and vastness of the topic, the scope for future work is enormous which can be open to many different approaches. The undertaken study only deals with the datasets limited to classification and not regression. Hence the study can be extended to feature selection in regression environments. The number of datasets clubbed per groups used in the study can be increased to achieve more concrete results with situational analysis. The variety in fixed thresholds can be increased to find more overfitting thresholds between the automatic thresholds and the 50 percent thresholds. All the used methods in this study were running on 2 cores of the machine which affected the computing times of the methods and hence parallel computing can be used in order to understand the performance of methods in parallel processing environments. The ensemble methods used in the study is limited to the use of aggregation methods and hence there is a huge scope for using stacking, bagging and other ensemble methods. The novel approaches introduced in this study makes use of different methods making it more dependable but their ideology has been proved in form of the achieved results. Hence, they can be compiled to form more robust and elegant statistical algorithms that can adhere to the ideology and make the feature selection process easier to implement. Thus, the research can be continued in endless ways to add one step closer to create flawless pre-processing through feature selection in the analytics domain.

7 Acknowledgements

The success of the study can be credited to my wonderful supervisors Dr. Pramod Pathak, Dr. Paul Stynes, and Dr. Dympna 0'Sullivan. Without there guidance and valuable suggestions, this research would not have been possible. Their advice and kind gestures have helped me stay motivated throughout the course of this study and I owe all the successful results achieved in this study to them. Thank you very much!
Furthermore, I will take this opportunity to express my gratitude to my parents without whose support this masters study would not be possible. Thank you for the blessings!

References

Brancotte, B., Yang, B., Blin, G., Cohen-Boulakia, S., Denise, A. and Hamel, S. (2015). Rank aggregation with ties: Experiments and analysis, *Proceedings of the VLDB Endowment*, Vol. 11, pp. 1202–1213.

Canedo, V. B., Maroño, N. S. and Betanzos, A. A. (2015). Recent advances and emerging challenges of feature selection in the context of big data, *Knowledge-Based Systems*, Vol. 86, pp. 33–45.

Chandrashekar, G. and Sahin, F. (2014). A survey on feature selection methods, *Computers and Electrical Engineering*, Vol. 40, pp. 16–28.

Chatterjee, S., Mukhopadhyay, A. and Bhattacharyya, M. (2018). A weighted rank aggregation approach towards crowd opinion analysis, *Knowledge-Based Systems*, Vol. 149, pp. 47–60.

Dittman, D. J., Khoshgoftaar, T. M., Wald, R. and Napolitano, A. (2013). Classification performance of rank aggregation techniques for ensemble gene selection, *FLAIRS 2013 - Proceedings of the 26th International Florida Artificial Intelligence Research Society Conference*, pp. 420–425.

Hancer, E., Xue, B. and Zhang, M. (2018). Differential evolution for filter feature selection based on information theory and feature ranking, *Knowledge-Based Systems*, Vol. 140, pp. 103–119.

Hoque, N., Bhattacharyya, D. and Kalita, J. (2014). Mifs-nd: A mutual information-based feature selection method, *Expert Systems with Applications*, Vol. 41(14), pp. 6371–6385.

Li-Yeh Chuang, C.-H. K. and Yang, C.-H. (2008). A hybrid both filter and wrapper feature selection method for microarray classification, *IAENG*.

Liu, X. Y., Liang, Y., W, S., Yang, Z. Y. and Ye, H. S. (2018). A hybrid genetic algorithm with wrapper-embedded approaches for feature selection, *IEEE Access*.

Morán-Fernández, L., Bolón-Canedo, V. and Alonso-Betanzos, A. (2017). Centralized vs. distributed feature selection methods based on data complexity measures, *Knowledge-Based Systems*, Vol. 117, pp. 27–45.

Nakariyakul, S. (2018). High-dimensional hybrid feature selection using interaction information-guided search, *Knowledge-Based Systems*, Vol. 140, pp. 1–14.

Nassif, A., Azzeh, M. and Banitaan, S. (2017). Robust rank aggregation method for case-base effort estimation, *Canadian Conference on Electrical and Computer Engineering*, Vol. 7946617.

Prati, R. (2012). Combining feature ranking algorithms through rank aggregation, *Proceedings of the International Joint Conference on Neural Networks*.

Rodriguez-Galiano, V. F., Luque-Espinar, J. A., Chica-Olmo, M. and Mendes, M. P. (2018). Feature selection approaches for predictive modelling of groundwater nitrate pollution: An evaluation of filters, embedded and wrapper methods, *Science of the Total Environment*, Vol. 624, pp. 661–672.

Rodríguez, J. P., Dharo-García, A., del Castillo, J. A. R. and Pedrajas, N. G. (2018). A general framework for boosting feature subset selection algorithms, *Information Fusion*, Vol. 44, pp. 147–175.

Seijo-Pardo, B., Canedo, V. B. and Betanzos, A. A. (2019). On developing an automatic threshold applied to feature selection ensembles, *Information Fusion*, Vol. 45, pp. 227–245.

Seijo-Pardo, B., Porto-Díaz, I., Bolón-Canedo, V. and Alonso-Betanzos, A. (2017). Ensemble feature selection: Homogeneous and heterogeneous approaches, *Knowledge-Based Systems*, Vol. 118, pp. 124–139.

Smetannikov, I., Deyneka, A. and Filchenkov, A. (2017). Meta learning application in rank aggregation feature selection, *Proceedings - 2016 3rd International Conference on Soft Computing and Machine Intelligence, ISCMI 2016*, Vol. 8057451, pp. 120–123.

Vora, S. and Yang, H. (2018). A comprehensive study of eleven feature selection algorithms and their impact on text classification, *Proceedings of Computing Conference*, pp. 440–449.

Wang, H., Khoshgoftaar, T. F. and Napolitano, A. (2010). A comparative study of ensemble feature selection techniques for software defect prediction.

Wang, Y. and Feng, L. (2018). Hybrid feature selection using component co-occurrence based feature relevance measurement, *Expert Systems with Applications*, Vol. 102, pp. 83–99.

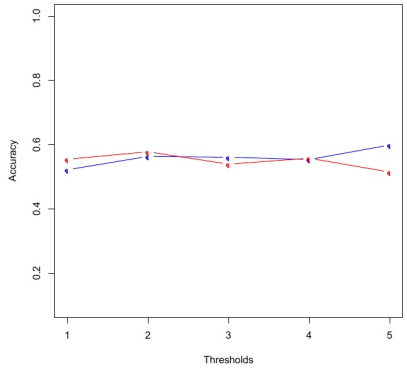
Yoon, H., Yang, K. and Shahabi, C. (2005). Feature subset selection and feature ranking for multivariate time series, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17(9), pp. 1186–1198.

Yun, Y. H., Deng, B. C., Cao, D. S., Wang, W. T. and Liang, Y. Z. (2016). Variable importance analysis based on rank aggregation with applications in metabolomics for biomarker discovery, *Analytica Chimica Acta*, Vol. 911, pp. 27–34.

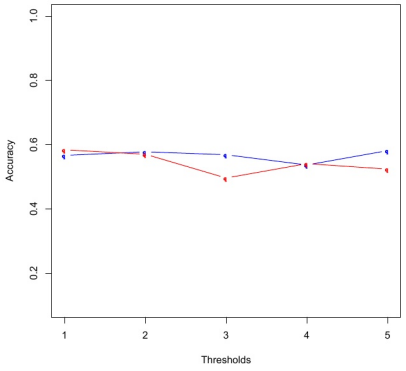
Zainudin, M. N. S., Sulaiman, M. N., Mustapha, N., Perumal, T., Nazri, A. S. A., Mohamed, R. and Manaf, S. A. (2017). Feature selection optimization using hybrid relief-f with self-adaptive differential evolution, *International Journal of Intelligent Engineering and Systems*, Vol. 10(2), pp. 21–29.

Zhang, C., Vinyals, O., Munos, R. and Bengio, S. (2018). A study on overfitting in deep reinforcement learning, *arXiv:1804.06893 [cs.LG]*.

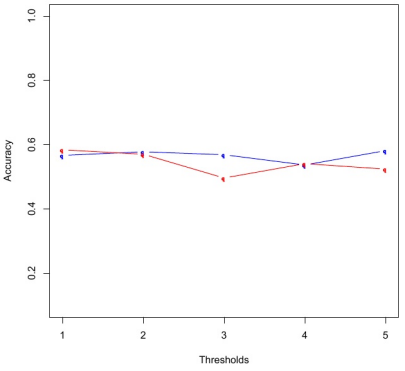
A Learning Curve Graphs



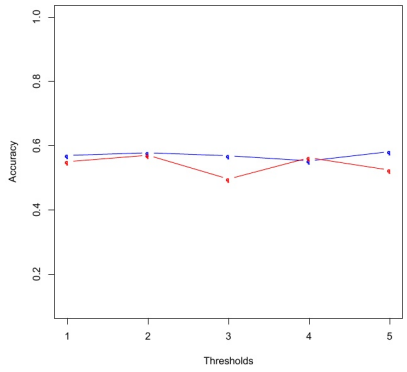
(a) DATASET3



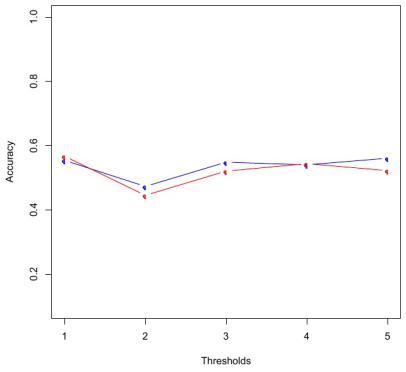
(b) DATASET3



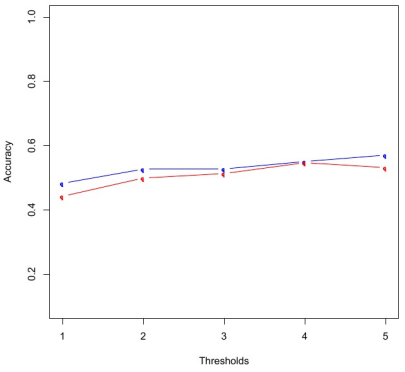
(c) DATASET3



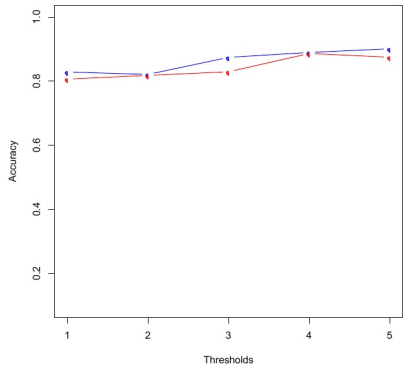
(d) DATASET3



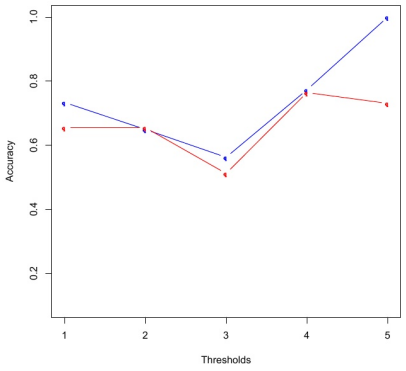
(e) DATASET3



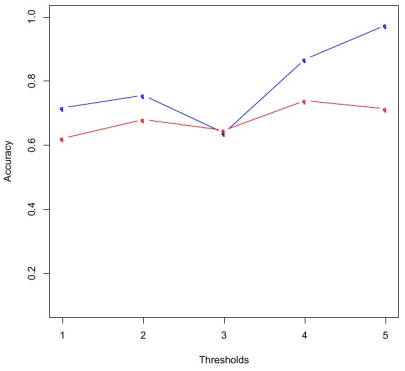
(f) DATASET3



(g) DATASET6

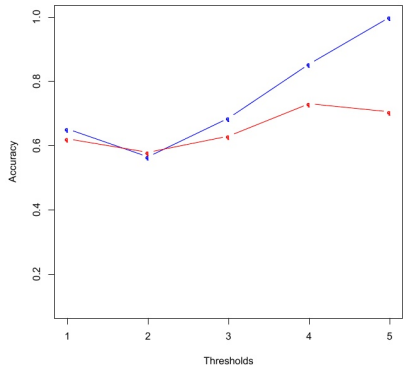


(h) DATASET7

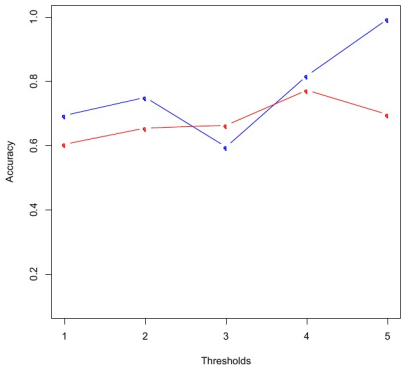


(i) DATASET7

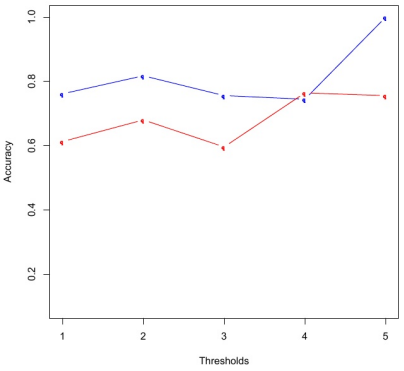
Figure 9: LEARNING CURVES FOR DATASET 3,6 AND 7



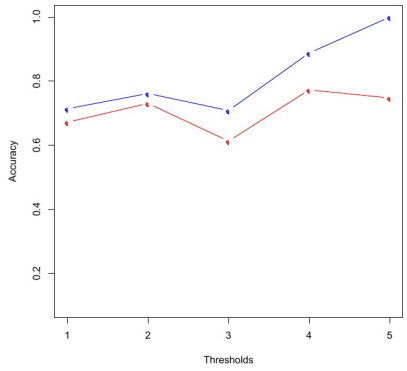
(a) DATASET7



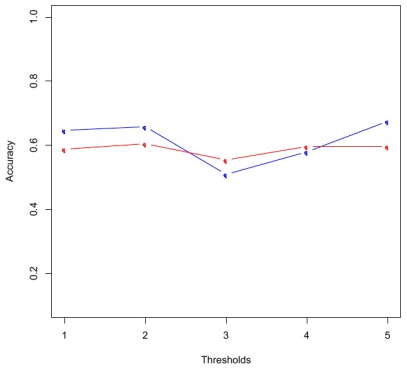
(b) DATASET7



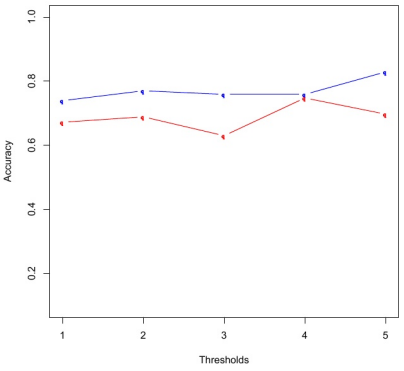
(c) DATASET7



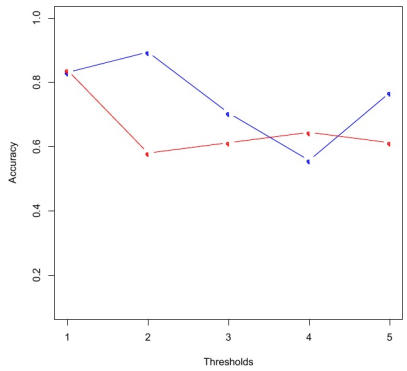
(d) DATASET7



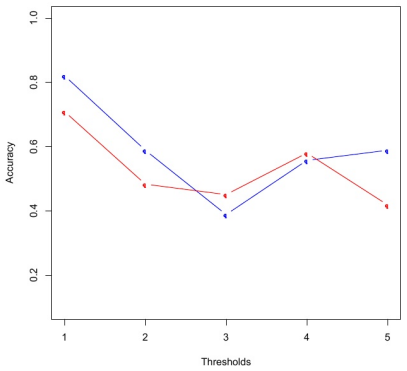
(e) DATASET7



(f) DATASET7



(g) DATASET8



(h) DATASET8

Figure 10: LEARNING CURVES FOR DATASET 7 AND 8

B Filter accuracies

Data	SR	B	FS	CH	I	G	S
1	0.5080645	0.5080645	0.4301075	0.4301075	0.4516129	0.4301075	0.4516129
2	0.7982609	0.8886957	0.6808696	0.8591304	0.8591304	0.8782609	0.8765217
3	0.6107692	0.4046154	0.5600000	0.4553846	0.4553846	0.4553846	0.4553846
4	0.6582583	0.6582583	FAILED	0.6582583	0.6582583	0.6582583	0.6582583
5	0.5131119	0.4606643	FAILED	0.2045455	0.3706294	0.1853147	0.4903846
6	0.8295455	0.8409091	0.8295455	0.8977273	0.9090909	0.7954545	0.8295455
7	0.5966387	0.7226891	0.6554622	0.6722689	0.6806723	0.6470588	0.6806723
8	0.4838710	0.7741935	0.6774194	0.5483871	0.7096774	0.7741935	0.7096774
9	0.5116279	0.5891473	0.2868217	0.5116279	0.3875969	0.4263566	0.3565891
10	0.3518519	0.6185185	0.5592593	0.5185185	0.6296296	0.5296296	0.6185185

Table 8: FILTER ACCURACIES AT LOG2

Data	O	R1	R2	R	C	M	AR
1	0.5672043	0.5080645	0.5080645	0.516129	0.3521505	0.3252688	0.5080645
2	0.8756522	0.8904348	0.8756522	0.8095652	0.3643478	0.6347826	0.8773913
3	0.5846154	0.4553846	0.4553846	0.5415385	0.5492308	0.5476923	0.5400000
4	0.6582583	0.6582583	0.6582583	0.6582583	FAILED	0.6582583	0.6582583
5	0.5664336	0.4589161	0.5620629	0.3741259	FAILED	0.3540210	0.5174825
6	0.7613636	0.8181818	0.8409091	0.8181818	0.8636364	0.7613636	0.8181818
7	0.6722689	0.7310924	0.7226891	0.6050420	0.6890756	0.5798319	0.6974790
8	0.5806452	0.7741935	0.5806452	0.7096774	0.8064516	0.7419355	0.5806452
9	0.6201550	0.5426357	0.6124031	0.5116279	FAILED	0.2093023	0.4573643
10	0.5629630	0.6111111	0.6444444	0.3740741	FAILED	FAILED	0.5851852

Table 9: FILTER ACCURACIES AT LOG2

Data	SR	B	FS	CH	I	G	S
1	0.4086022	0.4086022	0.3279570	0.4086022	0.4086022	0.3118280	0.4086022
2	0.7982609	0.8886957	0.6808696	0.8591304	0.8591304	0.8782609	0.8765217
3	0.4938462	0.4800000	0.4569231	0.5569231	0.5569231	0.5569231	0.5569231
4	0.6582583	0.6582583	FAILED	0.6582583	0.6582583	0.6582583	0.6582583
5	0.8688811	0.8776224	FAILED	0.7307692	0.7674825	0.6809441	0.7517483
6	0.8181818	0.8068182	0.8295455	0.8522727	0.8977273	0.7954545	0.8295455
7	0.5378151	0.6890756	0.7142857	0.6554622	0.7058824	0.6302521	0.6386555
8	0.6451613	0.6774194	0.8387097	0.7419355	0.8709677	0.7096774	0.7096774
9	0.4883721	0.5891473	0.3178295	0.5658915	0.5271318	0.4263566	0.3720930
10	0.7259259	0.9111111	0.8000000	0.8518519	0.8555556	0.8518519	0.8555556

Table 10: FILTER ACCURACIES AT 10

Data	O	R1	R2	R	C	M	AR
1	0.4086022	0.4086022	0.4086022	0.4086022	0.3279570	0.3279570	0.4086022
2	0.8756522	0.8904348	0.8756522	0.8095652	0.3643478	0.6347826	0.8773913
3	0.5569231	0.4969231	0.5200000	0.4984615	0.5538462	0.5476923	0.5446154
4	0.6582583	0.6582583	0.6582583	0.6582583	FAILED	0.6582583	0.6582583
5	0.8452797	0.8715035 0	.8802448	0.7762238	FAILED	0.6949301	0.8216783
6	0.7386364	0.8068182	0.8409091	0.8295455	0.7954545	0.7272727	0.8068182
7	0.7058824	0.6806723	0.7226891	0.5630252	0.7394958	0.5630252	0.7058824
8	0.6774194	0.7419355	0.8709677	0.7419355	0.3548387	0.5483871	0.4516129
9	0.6744186	0.5891473	0.5891473	0.4806202	FAILED	0.2248062	0.5503876
10	0.8555556	0.9037037	0.9185185	0.8333333	FAILED	FAILED	0.8777778

Table 11: FILTER ACCURACIES AT 10

Data	SR	B	FS	CH	I	G	S
1	0.4758065	0.4758065	0.3521505	0.4086022	0.4301075	0.4086022	0.4301075
2	0.8573913	0.6956522	0.6747826	0.4860870	0.5617391	0.9017391	0.9130435
3	0.5061538	0.5169231	0.5723077	0.5430769	0.5430769	0.5430769	0.5430769
4	0.6582583	0.6582583	FAILED	0.6582583	0.6582583	0.6582583	0.6582583
5	0.9230769	0.9326923	FAILED	0.8951049	0.8933566	0.8784965	0.8846154
6	0.8181818	0.8522727	0.7840909	0.8295455	0.9090909	0.8522727	0.8636364
7	0.6134454	0.5546218	0.5462185	0.5798319	0.6974790	0.6974790	0.6974790
8	0.5483871	0.8709677	0.4838710	0.7419355	0.7419355	0.7419355	0.7419355
9	0.6744186	0.6976744	0.4883721	0.6589147	0.6589147	0.4496124	0.6589147
10	0.8148148	0.8925926	0.8000000	0.8703704	0.8703704	0.8703704	0.8703704

Table 12: FILTER ACCURACIES AT 25

Data	O	R1	R2	R	C	M	AR
1	0.5000000	0.4758065	0.4758065	0.4274194	0.3521505	0.3252688	0.4301075
2	0.3869565	0.5573913	0.8382609	0.8443478	0.4939130	0.5965217	0.8078261
3	0.5430769	0.5261538	0.5369231	0.4830769	0.5523077	0.5246154	0.5569231
4	0.6582583	0.6582583	0.6582583	0.6582583	FAILED	0.6582583	0.6582583
5	0.8994755	0.9326923	0.9300699	0.9117133	FAILED	0.8522727	0.9213287
6	0.8068182	0.8522727	0.8409091	0.8295455	0.8750000	0.7386364	0.8636364
7	05882353	0.6890756	0.7226891	0.5882353	0.6218487	0.6218487	0.6722689
8	0.5806452	0.5806452	0.5161290	0.2580645	0.5483871	0.4838710	0.6129032
9	0.6821705	0.6744186	0.6976744	0.6201550	FAILED	0.5348837	0.6744186
10	0.8703704	0.9407407	0.9296296	0.8407407	FAILED	FAILED	0.8851852

Table 13: FILTER ACCURACIES AT 25

Data	SR	B	FS	CH	I	G	S
1	0.5241935	0.5241935	0.5403226	0.5080645	0.5241935	0.5080645	0.5241935
2	0.5913043	0.7130435	0.6747826	0.6121739	0.5886957	0.9226087	0.6313043
3	0.5476923	0.5584615	0.4953846	0.5230769	0.5230769	0.5230769	0.5230769
4	0.6582583	0.6582583	FAILED	0.6582583	0.6582583	0.6582583	0.6582583
5	0.9510490	0.9510490	FAILED	0.9466783	0.9440559	0.9431818	0.9475524
6	0.8295455	0.8750000	0.8068182	0.8636364	0.8636364	0.8750000	0.8863636
7	0.7478992	0.8067227	0.8067227	0.8151261	0.7058824	0.7731092	0.8067227
8	0.6129032	0.6451613	0.6129032	0.5806452	0.5806452	0.5806452	0.5806452
9	0.7054264	0.7209302	0.4883721	0.6821705	0.6821705	0.6976744	0.6821705
10	0.8518519	0.9333333	0.8000000	0.9074074	0.9074074	0.9074074	0.9074074

Table 14: FILTER ACCURACIES AT 50

Data	O	R1	R2	R	C	M	AR
1	0.5698925	0.5241935	0.5241935	0.5752688	0.4166667	0.3306452	0.5241935
2	0.7226087	0.6634783	0.5721739	0.5695652	0.6582609	0.6669565	0.6260870
3	0.5230769	0.5230769	0.5246154	0.5138462	0.5384615	0.5415385	0.5015385
4	0.6582583	0.6582583	0.6582583	0.6582583	FAILED	0.6582583	0.6582583
5	0.9493007	0.9545455	0.9510490	0.9344406	FAILED	0.9222028	0.9466783
6	0.8522727	0.8750000	0.8636364	0.8409091	0.8750000	0.8522727	0.8636364
7	0.7394958	0.7394958	0.7731092	0.6386555	0.7478992	0.6890756	0.7394958
8	0.6451613	0.6774194	0.8064516	0.6129032	0.5483871	0.5483871	0.6774194
9	0.6744186	0.7131783	0.7131783	0.6821705	FAILED	0.6434109	0.6666667
10	0.9074074	0.9333333	0.9296296	0.8851852	FAILED	FAILED	0.9407407

Table 15: FILTER ACCURACIES AT 50

Data	SR	B	FS	CH	I	G	S
1	0.6048387	0.6048387	0.6048387	0.6048387	0.6048387	0.6048387	0.6048387
2	0.8886957	0.8452174	0.6747826	0.7756522	0.7756522	0.9069565	0.5947826
3	0.5461538	0.5369231	0.4615385	0.5600000	0.5600000	0.4307692	0.4307692
4	0.6582583	0.6582583	FAILED	0.6582583	0.6582583	0.6582583	0.6582583
5	0.6660839	0.5533217	FAILED	0.2386364	0.6101399	0.3068182	0.4965035
6	0.8295455	0.8750000	0.8068182	0.8636364	0.8636364	0.8750000	0.8863636
7	0.5966387	0.7142857	0.6974790	0.6134454	0.7394958	0.6638655	0.7899160
8	0.5161290	0.7741935	0.4838710	0.4838710	0.4838710	0.4838710	0.4838710
9	0.6899225	0.7054264	0.4883721	0.6744186	0.6589147	0.6434109	0.6511628
10	0.8222222	0.9111111	0.8000000	0.9000000	0.9000000	0.9000000	0.9000000

Table 16: FILTER ACCURACIES AT AUTOMATIC THRESHOLD

Data	O	R1	R2	R	C	M	AR
1	0.6048387	0.6048387	0.6048387	0.6048387	0.6048387	0.6048387	0.6048387
2	0.3695652	0.7782609	0.5356522	0.8634783	0.5678261	0.7852174	0.8652174
3	0.6061538	0.4307692	0.6292308	0.5738462	0.3861538	0.4446154	0.6030769
4	0.6582583	0.6582583	0.6582583	0.6582583	FAILED	0.6582583	0.6582583
5	0.6354895	0.6529720	0.6748252	0.4886364	FAILED	0.4055944	0.6486014
6	0.8522727	0.8750000	0.8636364	0.8409091	0.8750000	0.8522727	0.8636364
7	0.6554622	0.7310924	0.7815126	0.7142857	0.7647059	0.7058824	0.7899160
8	0.6129032	0.5806452	0.8709677	0.7419355	0.7419355	0.6129032	0.8387097
9	0.6821705	0.7054264	0.6899225	0.6589147	FAILED	0.6511628	0.6356589
10	0.9000000	0.9370370	0.9296296	0.8851852	FAILED	FAILED	0.9222222

Table 17: FILTER ACCURACIES AT AUTOMATIC THRESHOLD

C Wrapper accuracies

Data	BFS	ES	FS	BS
1	0.6075269	0.6048387	0.6075269	0.6048387
2	0.6669565	EXEMPTED	0.8808696	0.7208696
3	0.5307692	EXEMPTED	0.5615385	EXEMPTED
4	0.6582583	EXEMPTED	0.6582583	0.6582583
5	EXEMPTED	EXEMPTED	0.7858392	EXEMPTED
6	0.8522727	EXEMPTED	0.8409091	0.9204545
7	0.6722689	EXEMPTED	0.5630252	0.8823529
8	0.5483871	EXEMPTED	0.6129032	0.6129032
9	0.5503876	EXEMPTED	0.6434109	0.6821705
10	0.7555556	EXEMPTED	0.762963	0.9222222

Table 18: WRAPPER ACCURACIES

Data	HCS	CFS	CONS	SVM-RFE
1	0.6048387	0.5241935	0.6075269	0.6075269
2	0.9008696	0.926087	0.6434783	0.8773913
3	EXEMPTED	0.5153846	0.5523077	EXEMPTED
4	0.6582583	0.6582583	0.6582583	0.6582583
5	EXEMPTED	EXEMPTED	0.7325175	EXEMPTED
6	0.875	0.8977273	0.8522727	0.8409091
7	0.8067227	0.6890756	0.6386555	0.5462185
8	0.516129	0.8387097	0.483871	0.516129
9	0.7054264	0.6899225	0.7054264	EXEMPTED
10	0.8407407	0.7555556	EXEMPTED	0.1185185

Table 19: WRAPPER ACCURACIES