CONSISTENCY EVALUATION FOR EXPONENTIAL, SEQUENTIAL AND RANDOM SEARCH STRATEGIES USING FILTER APPROACH

Dr.R.Rajkumar

Department of Computer Science and Applications The Gandhigram Rural Institute Gandhigram, Dindigul, India

rajkumarbdu@gmail.com

Abstract—Feature selection can improve the accuracy and efficiency of the learning process. Some of the methods are based on the search of the features that allow the data set considered consistent. Many alternative analysis functions that are employed in feature selection can be categorized as measures like distance, information, dependence, consistency, and classifier error rate. This research work is proposed on consistency measures by using exponential, sequential and random based searching strategies. The thought behind these measures is to predict the concept or class value of its instances. The consistencies that will study and compare all methods are dealt with in this paper elaborately.

Keywords— consistency, inconsistency, features, exhaustive, heuristic, and random search.

1. INTRODUCTION

This Consistency measure uses an inconsistency rate which is computed by finding all patterns with the same values in all features and counting total number of patterns minus the largest among that pattern of the same class for each group. The rate is computed by finding the ratio of the sum of these counts by the number of instances in the dataset.

The consistency degree as the opposite value of inconsistency, the consistency defined as *consistency= 1- inconsistency*. Some search algorithms, require the measure being monotonic to get optimal or better performance.

The monotonic property requires that if the feature subset belongs in feature set that is ∀ *fs'*∈*fs*, *ConCal(fs',D)* where *ConCal* is calculate consistency rate for feature subset in data D.

Three different algorithms represent standard search strategies: exhaustive- *FOCUS-RRK*, heuristic- *SETCOVER-RRK*, and probabilistic- *LAS-RRK*. The evaluations criteria are taken as exponential based consistency measurement for *FOCUS-RRK* algorithmic rule, the sequential based consistency measurement for *SETCOVER-RRK* algorithmic rule, and random based consistency measurement for *LAS-RRK* algorithmic rule. The aim of this experiment is to compare all those values of the measures with accuracy achieved via filter approach.

2. FOCUS-RRK

A. *Exhaustive search*

The *FOCUS* is one among the earliest algorithms within machine learning. *FOCUS* starts with an empty set and carries out breadth-first search till it finds a minimal subset that predicts pure classes [10]. If the set has three features, the root is *(0, 0, 0),* its children are *(001), (010),* and *(100)* where a '*0*' means that the absence of the respective feature and '*1*' means that its presence in the feature subset. It is exhaustive search in nature and original works on binary and noise-free data. With some simple modification of *FOCUS*, *FOCUS-RRK* which will work on non-binary data with noise by applying the inconsistency rate in place of the original consistency measure [5].

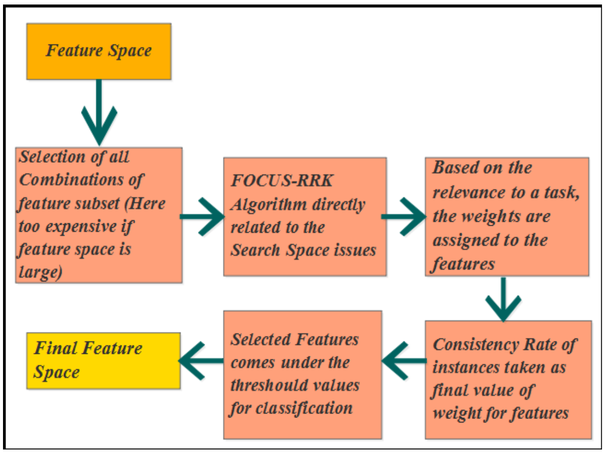
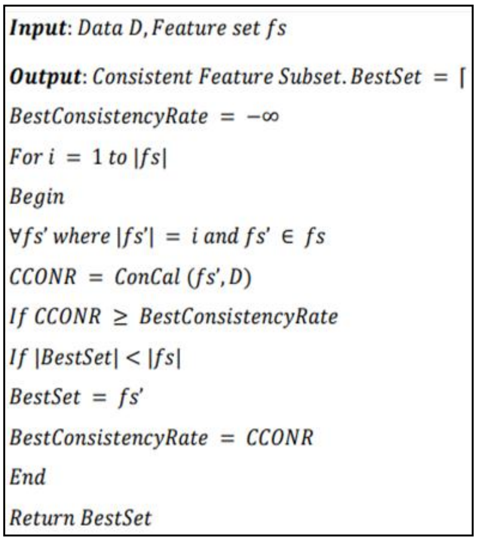

Feature Space
Selection of all Combinations of feature subset (Here too expensive if feature space is large)
FOCUS-RRK Algorithm directly related to the Search Space issues
Based on the relevance to a task, the weights are assigned to the features
Consistency Rate of instances taken as final value of weight for features
Selected Features comes under the threshould values for classification
Final Feature Space

Fig.2.1.Work flow of FOCUS-RRK Algorithm



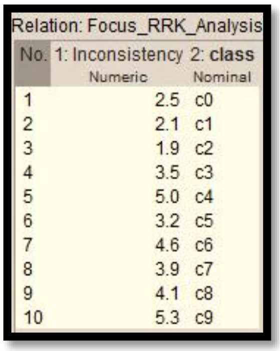
```
**Input**: Data $D$, Feature set $fs$
**Output**: Consistent Feature Subset. $BestSet = [$
$BestConsistencyRate = -\infty$
For $i = 1$ to $|fs|$
Begin
$\forall fs'$ where $|fs'| = i$ and $fs' \in fs$
$CCONR = ConCal\ (fs', D)$
If $CCONR \geq BestConsistencyRate$
If $|BestSet| < |fs|$
$BestSet = fs'$
$BestConsistencyRate = CCONR$
End
Return BestSet
```

Fig. 2.1.1 Pseudocode of FOCUS-RRK Algorithm

To be consistent and concise, denote *CCON* as the consistent count, *INC* as the inconsistent count, *CCONR* as the consistency rate, *INCR* as the inconsistency rate.

$$INCR = \sum_{i=1}^{h} INCi \div M$$

□□□□□□□□□□□□□□□□□



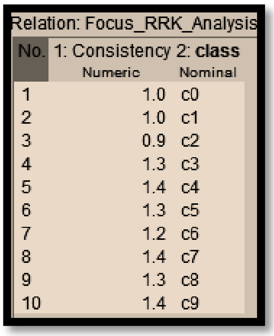
Relation: Focus_RRK_Analysis

No.	1: Inconsistency Numeric	2: class Nominal
1	2.5	c0
2	2.1	c1
3	1.9	c2
4	3.5	c3
5	5.0	c4
6	3.2	c5
7	4.6	c6
8	3.9	c7
9	4.1	c8
10	5.3	c9

Fig.2.2. Inconsistency Metrics for FOCUS-RRK

$$CCONR = \sum_{i=1}^{h} CCONi \div M$$

□□□□□□□□□□□□□□□



Relation: Focus_RRK_Analysis

No.	1: Consistency Numeric	2: class Nominal
1	1.0	c0
2	1.0	c1
3	0.9	c2
4	1.3	c3
5	1.4	c4
6	1.3	c5
7	1.2	c6
8	1.4	c7
9	1.3	c8
10	1.4	c9

Fig.2.3. Consistency Metrics for FOCUS-RRK

As *FOCUS-RRK* algorithm is exhaustive search it guarantees an optimal solution. However, a quick analysis will tell that *FOCUS-RRK's* time performance will deteriorate fast with increasing instances. This issue is directly associated with the size of the search space. The search space of *FOCUS-RRK* is closely related to the number of relevant features.

To overcome the redundancy issue, a new exponential search strategy approach has been incorporated within the *FOCUS-RRK* algorithm.

The efficiency of the algorithm is measured in terms of accurate classification. The accuracy of classification is measured in terms of Precision, Recall and F-Measures using Naïve Bayes classifier.

TABLE I. CONFUSION MATRIX FOR FOCUS-RRK

Algorithm	Precision	Recall	F-Measure	Accuracy (%)
FOCUS-RRK	0.823	0.8611	0.8415	84

In general, the smaller the search space of *FOCUS-RRK* and higher its efficiency. Otherwise, one requires more efficient techniques.

3. SETCOVER –RRK

B. Heuristic search

SetCover exploits the observation that the problem of finding the smallest set of consistent features is corresponding to 'covering' each pair of examples that have different class labels. Two instances with different class labels are said to be 'covered' when there exists at least one feature which takes different values for the two instances [8].

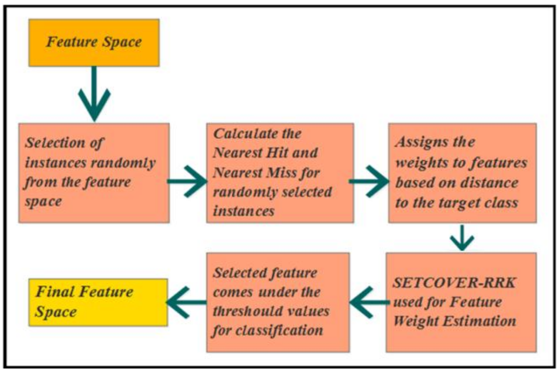
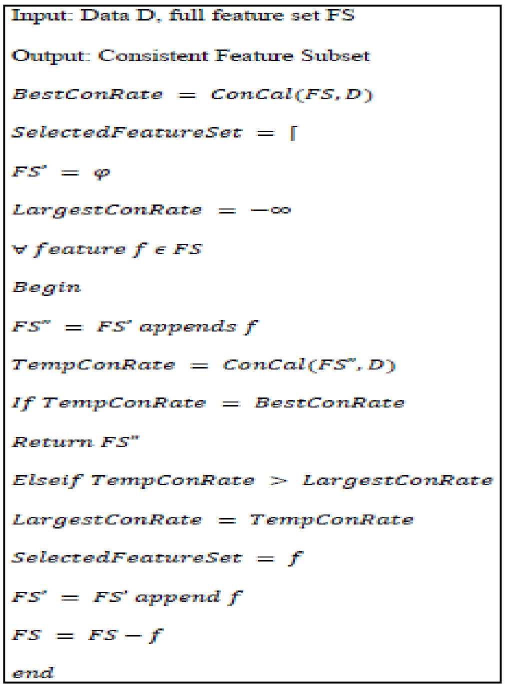
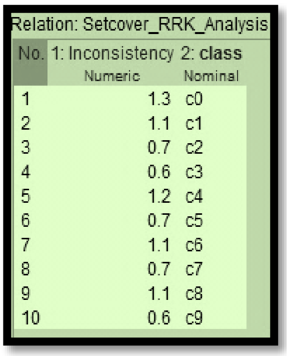

Feature Space
Selection of instances randomly from the feature space
Calculate the Nearest Hit and Nearest Miss for randomly selected instances
Assigns the weights to features based on distance to the target class
SETCOVER-RRK used for Feature Weight Estimation
Selected feature comes under the threshould values for classification
Final Feature Space

Fig.3.1 Work flow of SETCOVER-RRK Algorithm



```
Input: Data D, full feature set FS
Output: Consistent Feature Subset
$BestConRate = ConCal(FS, D)$
$SelectedFeatureSet = [$
$FS' = \varphi$
$LargestConRate = -\infty$
$\forall feature\ f \in FS$
$Begin$
$FS'' = FS'\ appends\ f$
$TempConRate = ConCal(FS'', D)$
$If\ TempConRate = BestConRate$
$Return\ FS''$
$Elseif\ TempConRate > LargestConRate$
$LargestConRate = TempConRate$
$SelectedFeatureSet = f$
$FS' = FS'\ append\ f$
$FS = FS - f$
$end$
```

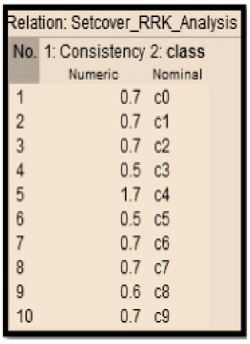
Fig. 3.1.1Pseudocode of SETCOVER-RRK Algorithm



Relation: Setcover_RRK_Analysis

No.	1: Inconsistency Numeric	2: class Nominal
1	1.3	c0
2	1.1	c1
3	0.7	c2
4	0.6	c3
5	1.2	c4
6	0.7	c5
7	1.1	c6
8	0.7	c7
9	1.1	c8
10	0.6	c9

Fig.3.2 Inconsistency Metrics for SETCOVER-RRK



Relation: Setcover_RRK_Analysis

No.	1: Consistency Numeric	2: class Nominal
1	0.7	c0
2	0.7	c1
3	0.7	c2
4	0.5	c3
5	1.7	c4
6	0.5	c5
7	0.7	c6
8	0.7	c7
9	0.6	c8
10	0.7	c9

Fig.3.3. Consistency Metrics for SETCOVER-RRK

The consistency criterion is restated by the expression that a feature set S is consistent if, for any pair of instances with different class labels, there is a feature in S that takes different values. Thus including a feature f in S 'covers' all those example pairs with different class labels on which f takes different values. Once all pairs are 'covered' is the resulting set S consistent.

In the report extensive experimental results which show that SETCOVER-RRK is fast, close to optimal, and deterministic. It works well for Social Network Dataset where features are rather independent of each other. It may, however, have a problem where features are correlated. This is often as a result of it selects the best feature in each iteration based on the number of instance pairs are covered. So, any feature that's most correlated to the class label is chosen initial. An example is the Social Network Dataset which has 51 instances which consist of 10 features. SetCover second selects the feature F10 due to the fact that among all the features. F10 covers the maximum number of instances (87%). Then it selects the features F1, F10; so, it selects the wrong subset *(F1, F10, F5, F8, F2)* overall.

The efficiency of an algorithmic rule is measured in terms of accurate classification. The accuracy of classification is measured in terms of Precision, Recall and F-Measures using Naïve Bayes classifier.

TABLE II. CONFUSION MATRIX FOR SETCOVER-RRK

Algorithm	Precision	Recall	F-Measure	Accuracy (%)
SET-COVERRRK	0.8407	0.9047	0.8714	86.6

So, we tend to found that exhaustive methods have an inherent drawback because they require large computational time. Heuristic methods such as SetCover, although very fast and accurate, will encounter issues if the data has highly correlated features. Hence, a new solution is required that avoids the problems of exhaustive and heuristic search. The probabilistic search could be a natural alternative.

4. LAS-RRK

C. Probabilistic search

LAS-RRK algorithm for feature subset selection can make probabilistic choices of subsets in search of an optimal set. Another similar type of algorithm is the Monte-Carlo algorithm in which it is often possible to reduce the error probability arbitrarily at the cost of a little increase in computing time. Proposed a probabilistic algorithm called LAS-RRK where probabilities of generating any subset are equal. LAS-RRK adopts the inconsistency rate as the evaluation measure. It generates feature subsets randomly with equal probability, and once a consistent feature subset is obtained that satisfies the threshold inconsistency rate. The size of generated subsets is fixed to the size of that subset, i.e., subsets of higher size are not evaluated any longer. This is based on the fact that inconsistency rate is monotonic, i.e., a superset of a consistent feature set is also consistent. LAS-RRK is fast in reducing the number of features and eliminates noisy features in the early stages and can produce optimal solutions.

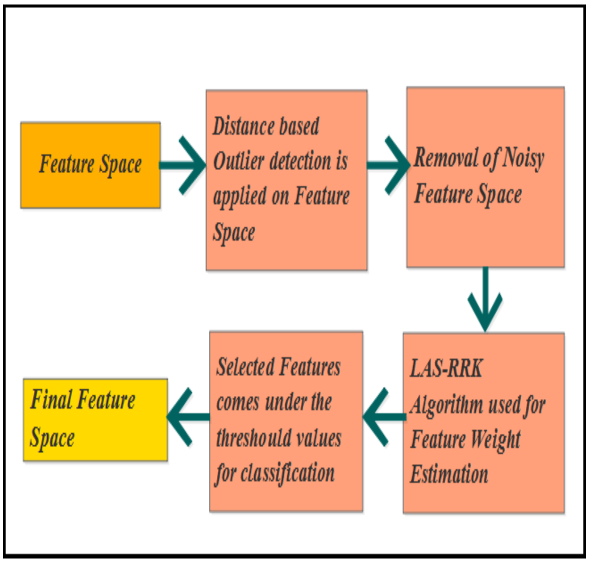
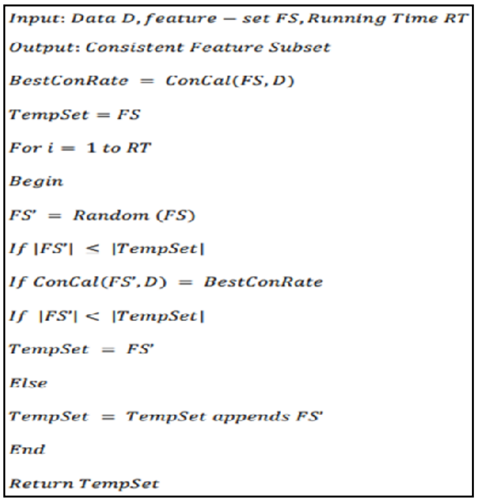

Feature Space
Distance based Outlier detection is applied on Feature Space
Removal of Noisy Feature Space
LAS-RRK Algorithm used for Feature Weight Estimation
Selected Features comes under the threshould values for classification
Final Feature Space

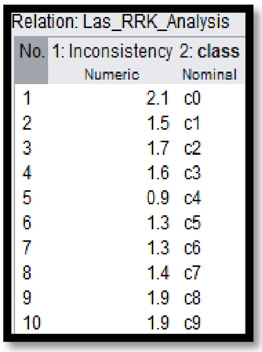
Fig. 4.1 Work Flow of LAS-RRK Algorithm



```
Input: Data D, feature – set FS, Running Time RT
Output: Consistent Feature Subset
BestConRate = ConCal(FS, D)
TempSet = FS
For i = 1 to RT
Begin
FS' = Random (FS)
If |FS'| < |TempSet|
If ConCal(FS', D) = BestConRate
If |FS'| < |TempSet|
TempSet = FS'
Else
TempSet = TempSet appends FS'
End
Return TempSet
```

Fig.4.1.1Pseudocode of LAS-RRK

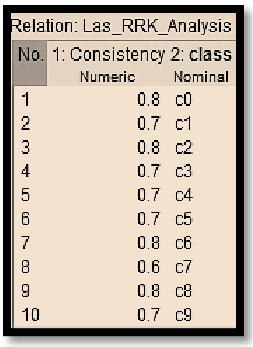
The solutions of equal size, the LAS-RRK algorithm produces a list of equal-sized feature subsets at the end.



Relation: Las_RRK_Analysis

No.	1: Inconsistency Numeric	2: class Nominal
1	2.1	c0
2	1.5	c1
3	1.7	c2
4	1.6	c3
5	0.9	c4
6	1.3	c5
7	1.3	c6
8	1.4	c7
9	1.9	c8
10	1.9	c9

Fig. 4.2 Inconsistency Metrics for LAS-RRK



Relation: Las_RRK_Analysis

No.	1: Consistency Numeric	2: class Nominal
1	0.8	c0
2	0.7	c1
3	0.8	c2
4	0.7	c3
5	0.7	c4
6	0.7	c5
7	0.8	c6
8	0.6	c7
9	0.8	c8
10	0.7	c9

Fig. 4.3 Consistency Metrics for LAS-RRK

TABLE III. CONFUSION MATRIX FOR LAS-RRK

Algorithm	Precision	Recall	F-Measure	Accuracy (%)
LAS-RRK	0.8909	0.9514	0.9201	91.9

5. COMPARATIVE ANALYSIS

TABLE IV. MEASURES FOR NAÏVE BAYES CLASSIFICATION

Algorithm	Precision	Recall	F-Measure	Accuracy (%)
FOCUS-RRK	0.823	0.8611	0.8415	84
SETCOVER-RRK	0.8407	0.9047	0.8714	86.6
LAS-RRK	0.8909	0.9514	0.9201	91.9

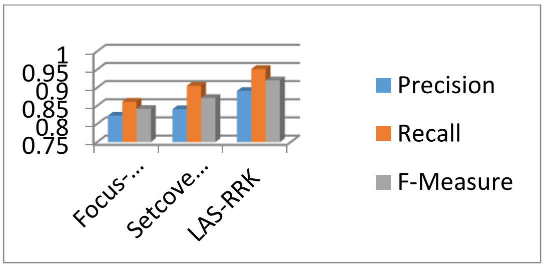


Fig. 5.1 Confusion Matrixes for Three Algorithms

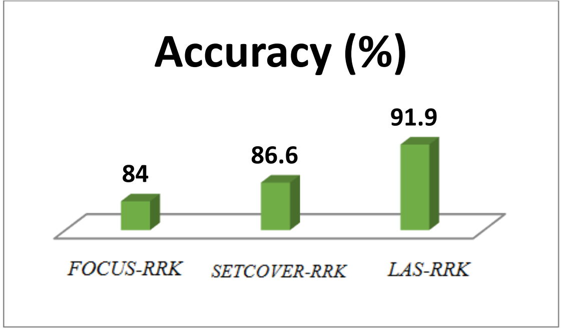


Fig. 5.2 Accuracy Percentages for Three Algorithms

The average accuracy of the proposed work for FOCUS-RRK is 84%, the SETCOVER-RRK is 86.6% and LAS-RRK is 91.9%.

6. CONCLUSION

This paper is aimed to carry out a study of consistency measure with different search strategies. The study of the consistency measure for FOCUS-RRK, SETCOVER-RRK, and

LAS-RRK can be used to remove redundant, irrelevant and noisy features [4]. There is a tendency to investigate different search strategies were investigated for consistency measure, like exhaustive, heuristic and probabilistic. Finally, all those strategies are compared over consistency measure. When compared with these three algorithms, the random search technique has achieved higher accuracy level.

7. REFERENCES

[1] R.Rajkumar “Behavior Analysis and Feature Selection in Online Social Network” IJSEAS, Volume-1, Issue-8,November 2015 ISSN: 2395-3470

[2] Jiliang Tang, HujiGao. Exploiting Homophily Effect for Trust Prediction, 2013.

[3] Harshali. D and Gangurde., Feature Selection using Clustering Approach for Big Data.ITCCE., 2014.

[4] Ghazi Fuad Khamis and Ramadas Naik.T., Identifying and removing irrelevant and redundant features in high dimension data using feature subset., 2: 2015.

[5] Rajkumar Ramasamy Focus-RRK Feature Selection Technique For Enhancing The Accuracy In Social Network Data, wjert, 2017, Vol. 3, Issue 1, 91 -96, ISSN 2454-695X.

[6] Sutha.k and Dr.Jebamalar Tamilselvi.J., A review of feature selection algorithms for data mining techniques., Volume no: 7- june-2015.

[7] Antonela tommasel., Integrating Social Network Structure in to Online Feature Selection IJCAI- 2016.

[8] R.Rajkumar and Dr. Anbuselvi Set Cover-RRK Feature Selection Technique For Enhancing The Accuracy In Social Network Data wjert, 2017, Vol. 3, Issue 1, 97 -102, ISSN 2454-695X

[9] Mohammed-Ali Abbasi Measuring User Credibility in Social Medid, 2012.

[10] Jiliang Tang, HujiGao. Exploiting Homophily Effect for Trust Prediction, 2013.