

A Comparative Experiment Study on Supervised Classifier SMO – Support Vector Classifier and Unsupervised Hierarchical Cluster for Chronic Kidney Disease Data Statistics

Dr. Rajesh S. Walse¹, Dr. Pawan S. Wasnik², Dr. Hemant S. Mahalle³

¹Department of Dairy Business Management, College of Dairy Technology, Warud (Pusad), Animal and Fishery Sciences University, Nagpur, Maharashtra, India

²Department of Computer Science, New Model Degree College, Hingoli, S.R.T.M. University, Nanded, Maharashtra, India

³Department of Computer Science, Shri Vitthal Rukhmini College, Sawana, Tal, Mahagaon, Dist, Yavatmal, Sant Gadge Baba Amaravati University, Amaravati, Maharashtra, India

ABSTRACT

The researcher using a classification method for the comparative study of chronic kidney patient analysis of data for Supervised with SMO – SVM and Unsupervised Hierarchical Clustering function. Now we are proposed best model by applying the Chronic kidney disease data contains 25 attributes and 400 instances including class, First, we are applying Supervised- classify – SMO- SVM- One Attributes – htnVs Class, Applying Cross- validation from 4 to 15 + 70% split, Calibrator: Logistic and Kernel: Polynomial, and found the result, the summary of classifier model value of ROC area for CV fold using 4 is 0.794 and the weighted average is same for all CKD, not-CKD class value is also 0.794, the accuracy of correctly classified instances 74.25 % as well as the result of a Confusion matrix is same for all Cross-Validation Folds from 5 to 15, the value of ROC area CCI is same for all CV folds from 5 to 15.

And similarly, the researcher is using Second, Step-2 by applying two attributes (htn, appetVs Class),– Two Attribute with Supervised- classify – SMO- SVM- Applying Cross-validation folds from 4 to 15 + 70% split, Calibrator: Logistic and Kernel: Polynomial function to check further prediction and trying to increase the better accuracy as compared to the earlier model, during practical research found the result the summary of classifier model value of ROC area is by using CV folds is 4 the value of ROC Area is 0.868 and CV fold is 5 the result of ROC area is 0.844. The accuracy of the result of summary classifier model by applying CV Fold is 4 and the accuracy of correctly classified instances is 83.33 %, the researcher also checks the CCI accuracy by increasing CV fold from 5 to 15, but the result is not getting the higher accuracy, as well as the value of Confusion matrix, is same for all Cross-Validation Folds from 6 to 15, the value of ROC area CCI is same for all CV folds from 6 to 15.

Similarly, research also trying to increased accuracy and allying Third, Step-3 (24-Full Vs Class) – by using Full Attribute, Supervised- classify – SMO- SVM- Full 24 Attributes – 24 Attributes Vs Class (CKD and not-CKD), Applying Cross-validation from 4 to 15 + 70% split, Calibrator: Logistic and Kernel: Polynomial

function and found the result of accuracy by the class value of ROC Area is near to value 1 i.e 0.988 with accepting the value of CV fold is 4,7,11,12,13 and the another highest accuracy value of ROC Area is 0.990 which is equal to value 1.000 with accepting the value of CV folds is 5,6,8,9,10,14,15 which is the highest accuracy, similarly the value of CCI is

98.50 % for CV fold is 4, 7, 11, 12 & 13 & 98.75 % for CV folds are 5,6,8,9,10,14,15, the result is same for confusion matrix also by using same CV folds.

Again researcher is also comparing the result with unsupervised, hierarchical clustering algorithm by applying full attributes i.e of 24 attributes with 400 instances, also found the best and accurate result and highest accuracy is the prediction of hierarchical algorithm model.

Therefore, found the best model prediction for Supervised- SMO in WEKA on the basis of three test using calibrator of logistic and kernel using polynomial function by applying Cross Validation Folds from 4 to 15(Train on a portion of the data and test on the remainder) with 70% split, and final prediction is, increasing the no of attributes the accuracy of Correctly Classified Instances (CCI), ROC Area value and Confusion matrix value increased. Similarly research is also found to confirm predict the result of Unsupervised Hierarchical Clustering algorithm by applying full attributes also confirm prediction is increasing the number of Clusters from 2,3,4 and 5 for both Euclidean and Manhattan Function, the accuracy of result in terms of better and accurate clusters found. The adopted methodology clears the process of practical.

Keywords: Data Mining, Classification, Clustering, SMO, SVM, Calibrator, Logistic, Kernel, Polynomial, Euclidean, Manhattan, WEKA, CKD.

I. INTRODUCTION

INTRODUCTION

to Now the computer science techniques like optimized association rule mining techniques is using for improved Genetic Algorithms data mining and machine learning are used to study the power of various parameters and make predictions of the based on different data sets. Data mining techniques is the process of identifying the hidden patterns from the big and tedious data. This may provide a vital role in the decision making for large data, not only agriculture but also health-related problems. Bharara et al. [3] reviewed to extract for business operations using Data Mining techniques. Ariff et al. [2] studied RFID based systematic livestock health management system. Jinyin [7] performed a novel cluster center is the fast determination clustering algorithm. Dilli Arasu and Thirumalaiselvi [1] dealt for novel imputation techniques for the effective type of predictions of kidney disease patients. Zou Chuan et al. [4] performed an applied study of Guangdong provincial hospital of traditional Chinese treatment. Guangzhou and explore clustering analysis for syndrome evolution peritoneal dialysis patients. Kunwar et al. [9] studied and analyzed Chronic in terms of permanent Kidney Disease harnessing of data mining for classification techniques. Anh Luong [5] applied K- Means

Approach to Clustering disease Progressions. Sabri [6] used data mining techniques for segmenting customers' information. Kumar and Lhatri [10] used WEKA is used for medical related data classification and to find early disease prediction. Khanna [10], NCBI [12] performed a study on the economics of Dialysis in India. J Nephrol [13] studied the occurrence of chronic kidney disease in India, and where are we heading? Uboltham et al. [11] performed a diagnostic study of acute kidney injury using the KDIGO guideline approach. This paper Experiment has carried out on chronic kidney disease patient based on their relationship attributes, nowadays chronic kidney disease patient in India is increased day by day because of their eating habit and other health issues. Still, from the last ten years, CKD patient numbers it is increased tremendously Indian Journal of Nephrology et al. [12], therefore, in future this kind of research which will be helpful to the doctors or medical industry for prediction of CKD and not CKD patient based on their other health parameters, to minimize the growth rate of CKD patients and to control further damages of the kidney. Data mining plays an active role in predicting future kidney-related health problems. In this research paper, three algorithms it has been analyzed one is NB Classifier, J48, and Random Forest Decision Tree. Data cleaning in DM is used to removal of noise and inconsistent data with data integration technique with the combination of multiple types of data. To evaluate the data, we have used secondary data and it is retrieved from UCI machine learning repository [14]. Jnephrol [13], with increasing life period and the frequency of lifestyle disease, the US has seen a 30% considerable growth in the widespread presence of CKD in the last decade. Unfortunately, from India, there is no longitudinal study and limited data on the incidence of CK. At present, the living standard of the people and the daily consumption of food are adversely affecting their health, especially their everyday living, which is increasing the number of kidney diseases in India every day. His anatomy also depended on the diet of people 40 years ago or older, but today, kidney disease is not only limited to people with diabetes or hypertension, but it has many causes. Chemical cereals, vegetables, fruits are the result of all these things, This is our daily food, and where the result is not where we are on the kidney, so the loss of kidney function slowly and then become kidney failure, such things are growing. According to reference Jnephrol [13], unfortunately, from India, there is no longitudinal study of CKD and limited data. So because of all of the above, we have tried to analyze acute kidney disease by using Naïve Bayes, decision tree J48, and random forest algorithm unprocessed learning technique. Indeed, the purpose of our research is to use our research to analyze kidney disease or whether it can cause kidney disease in the future.

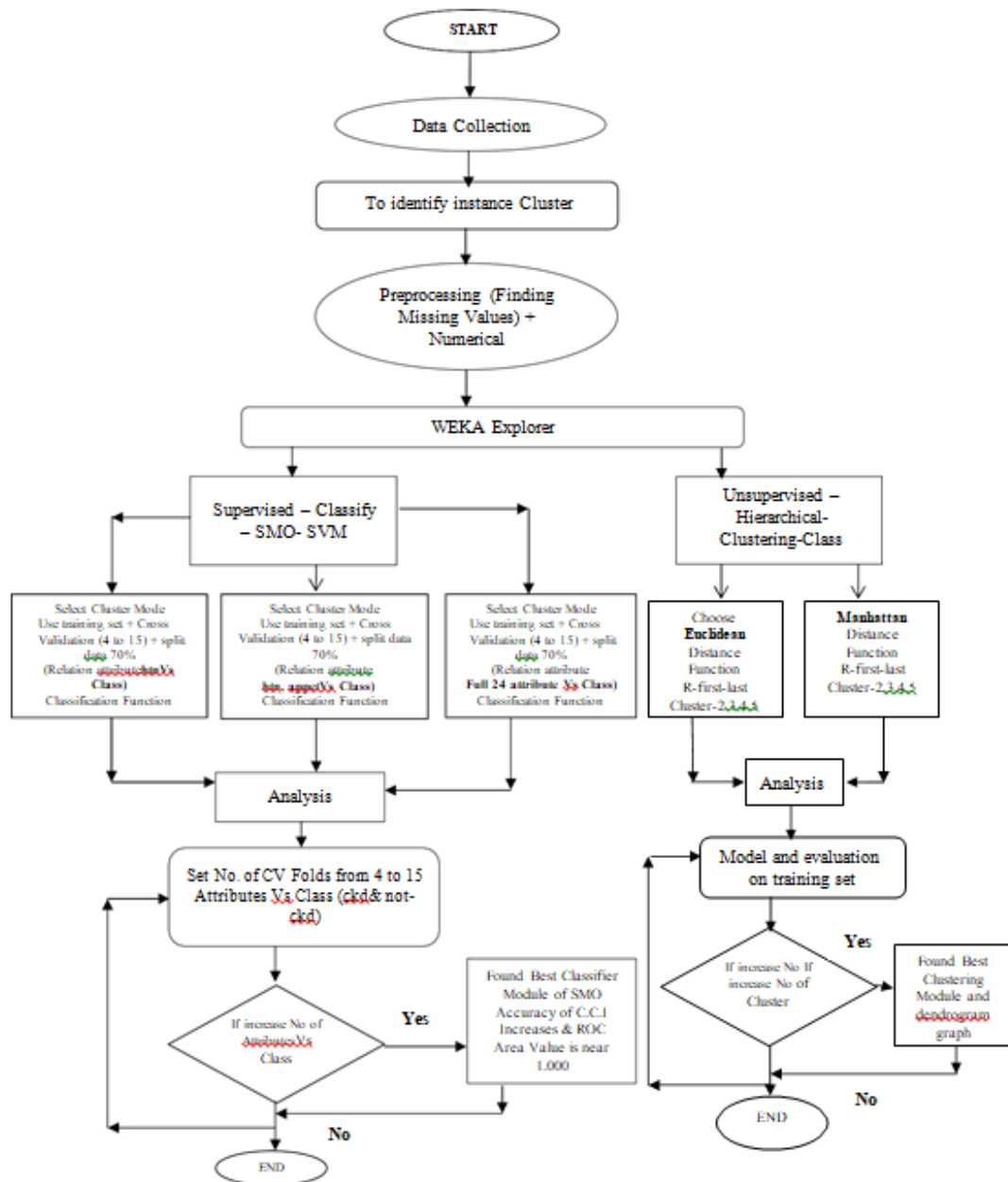


Figure 1: The research work flow

Figure 1: Adopted methodology flowchart

II. RESULT AND DISCUSSION

We are using a chronic kidney failure disease dataset, in dataset training database perfection for the Supervised SMO- SVM function and Unsupervised Hierarchical clustering techniques, and select some parameters 1. RBC count 2. Hyper-tension (BP) 3.Diabetes M. 4.Coronary disease 5.Appetite 6.Pedal Edema 7. Anemia, we are using WEKA tool for classifying and Clustering algorithms of data using Logistic and Polynomial function as well as Euclidean and Manhattan distance function.

The clinical data of kidney disease of 400 records considered for analysis has taken from the standard Machine Learning website. The data obtained after cleaning and removing missing values for further analysis, the data

contains 25 attributes in the dataset with class (CKD and Not-CKD) and Class distribution is (63% for CKD and 37% for not CKD).

III. RESULT & ANALYSIS

The result of experiment is to be compared of SMO Classification – SVM algorithm with Hierarchical Clustering algorithm are made established on the basis of performance in terms of high accuracy with a minimum period processing. The following algorithm is to analyze through data; the results and analysis of all three algorithms are as follows.

Supervised – Classifier – SMO–SVM- One , Two and Full Attributes Test Results using Calibrator: Logistic and Kernel : Polynomial

Step-1

Step-1 (htnVs Class) – One Attribute

Supervised- classify – SMO- SVM- 2 Attributes – htnVs Class Applying Cross validation from 4 to 15 + 70% split Calibrator: Logistic and Kernel: Polynomial

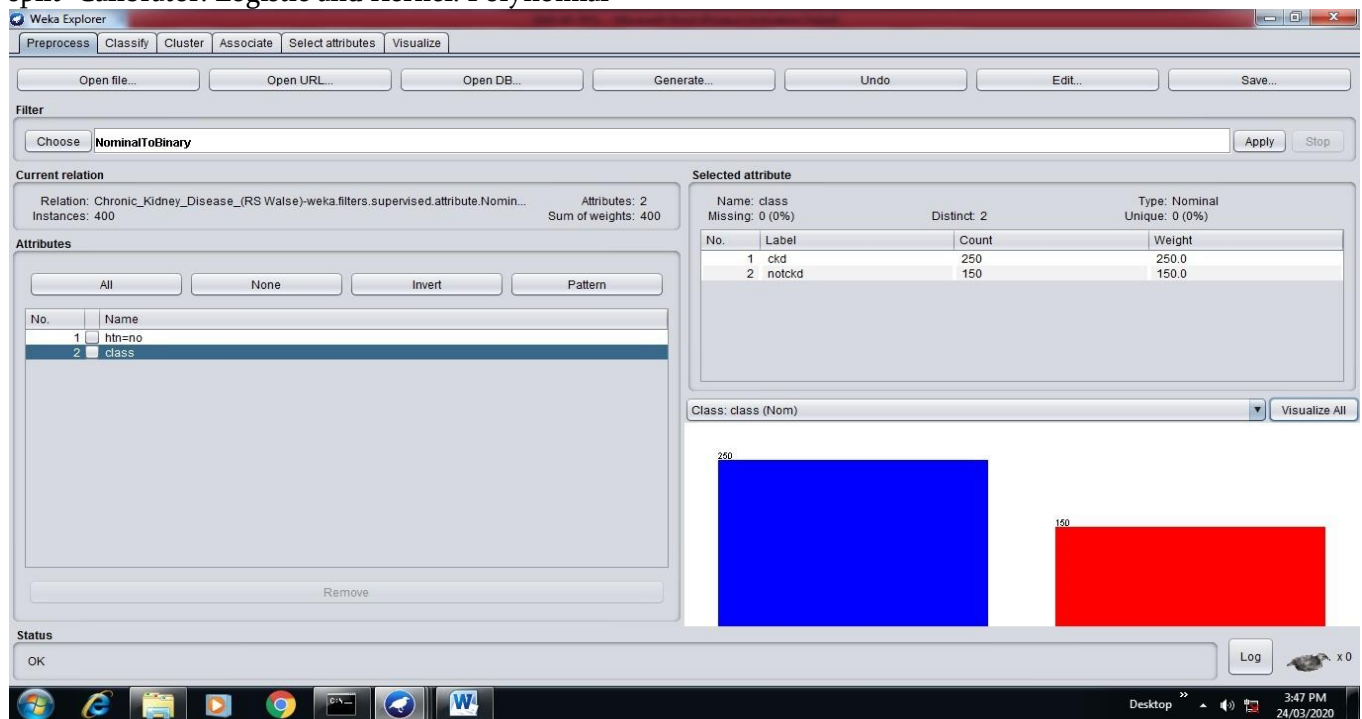


Figure: 2 Preprocess of imported data in WEKA

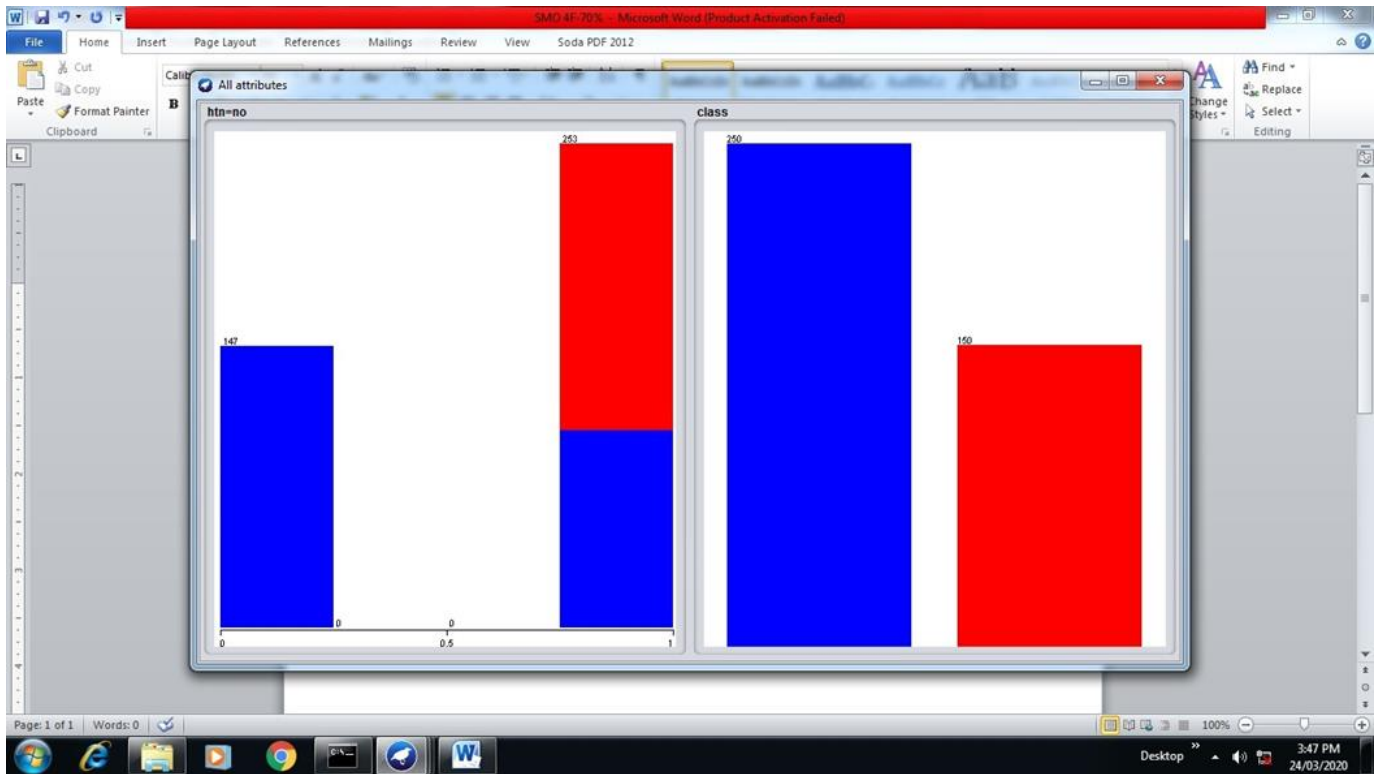


Figure: 3Histogram of data set in WEKA

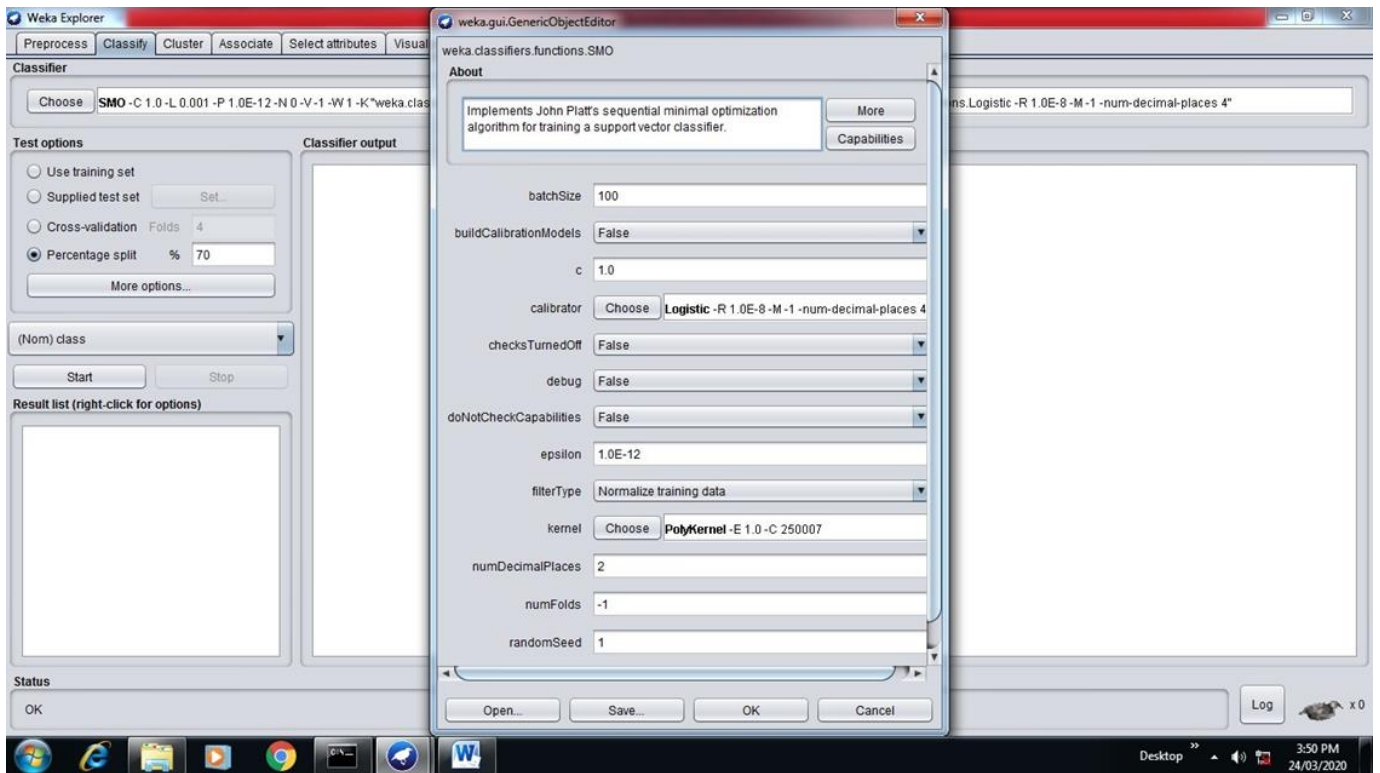


Figure: 4SMO – applying sequential minimal optimization algorithm for training a support vector classifier

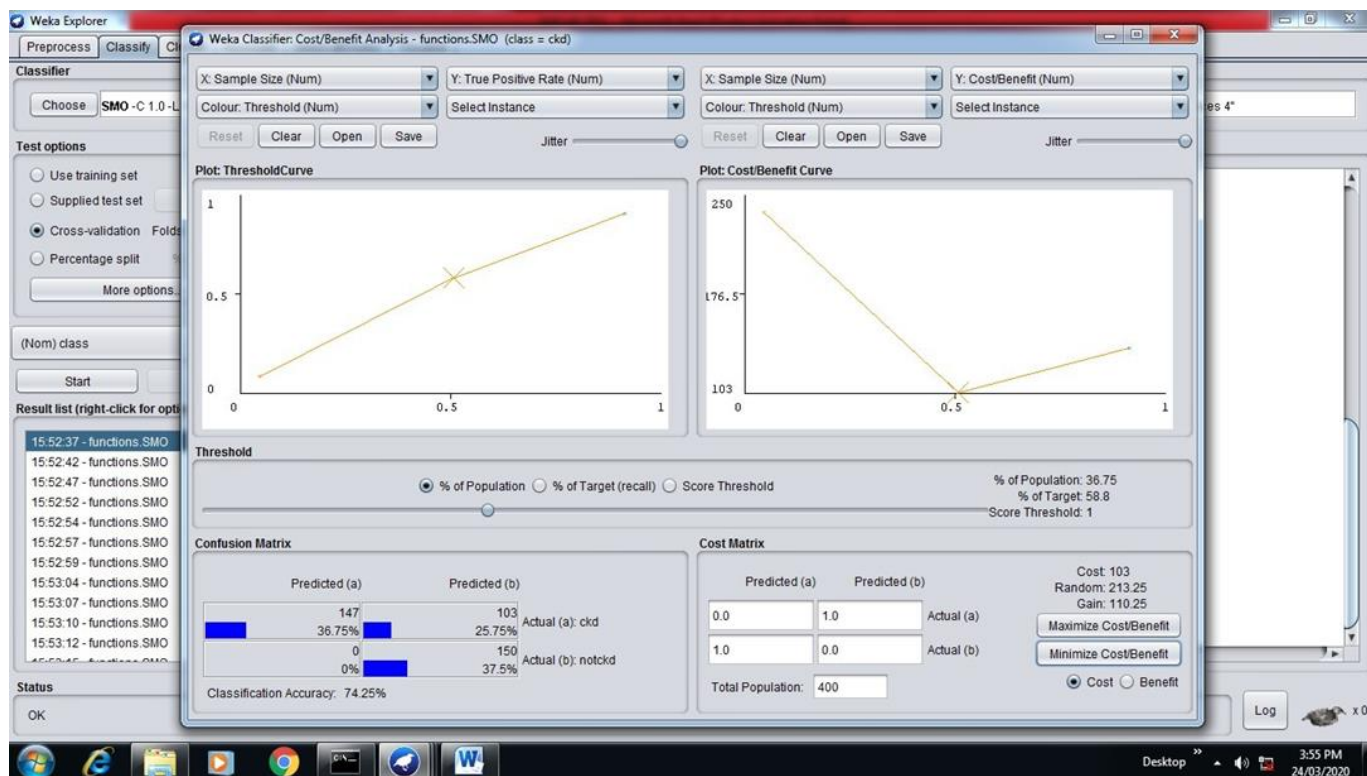


Figure: 5 Weka classifier : Cost / Benefit analysis: Function SMO (class=Ckd)

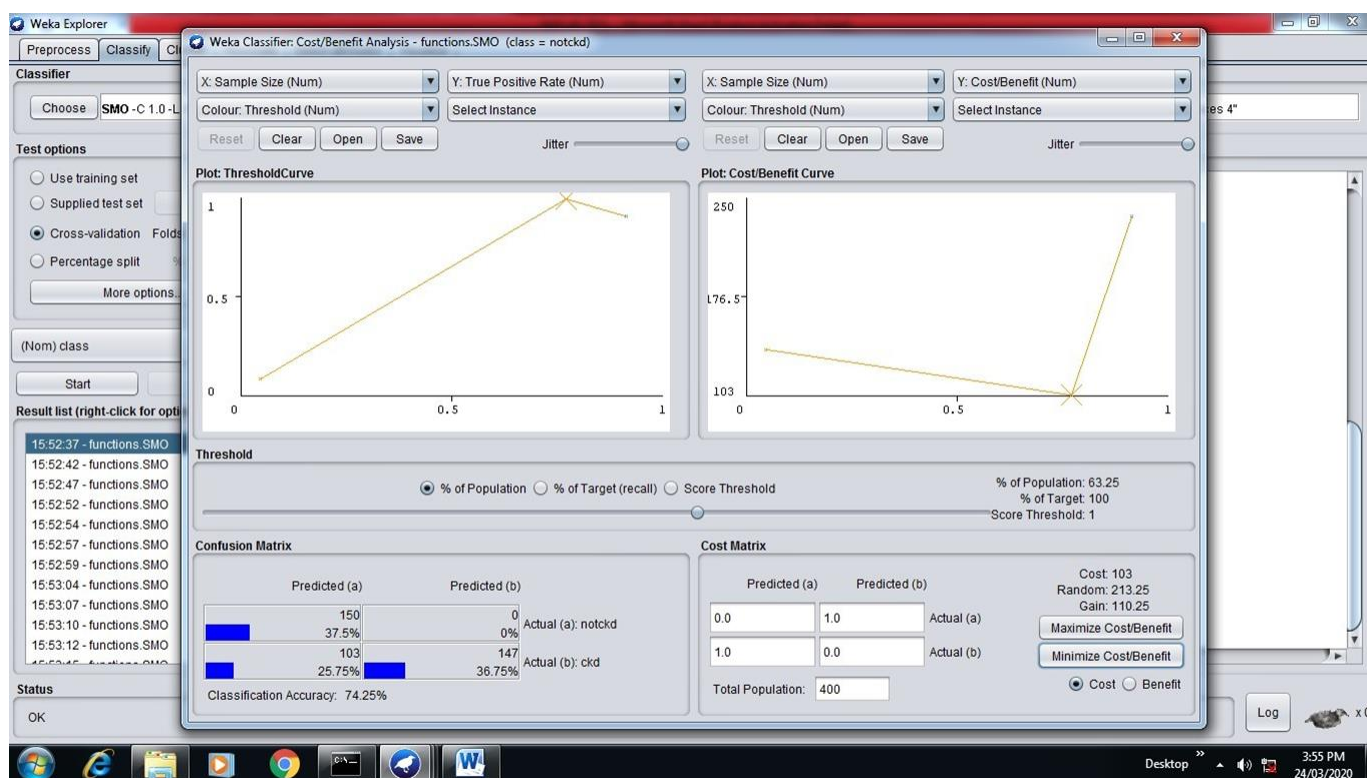


Figure: 6 Weka classifier : Cost / Benefit analysis: Function SMO (class=Not-Ckd)

Supervised – Classifier – SMO Function Step-1

Step-1 (htnVs Class) – One Attribute

Supervised- classify – SMO- SVM- 2 Attributes – htnVs Class Applying Cross validation from 4 to 15 + 70% split Calibrator: Logistic and Kernel: Polynomial

=== Classifier model (full training set) ===, SMO ,Kernel used:Linear Kernel: $K(x,y) = \langle x,y \rangle$

Classifier for classes: ckd, notckd BinarySMO, Machine linear: showing attribute weights, not support vectors.

2 * (normalized) htn=no, -1

COMPARATIVE STUDY SUPERVISED OF SMO, CALIBRATOR: LOGISTIC AND KERNEL: POLYNOMICAL USING CROSS VALIDATION FOLD 4 TO 15 WITH 70% OF SPLIT CLASSIFIER

Table 1: Detailed accuracy by Class for htnVs class

Cross Validation Fold	Class	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area
4	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
5	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
6	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
7	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
8	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
9	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
10	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
11	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
12	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593

	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
13	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
14	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
15	CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751

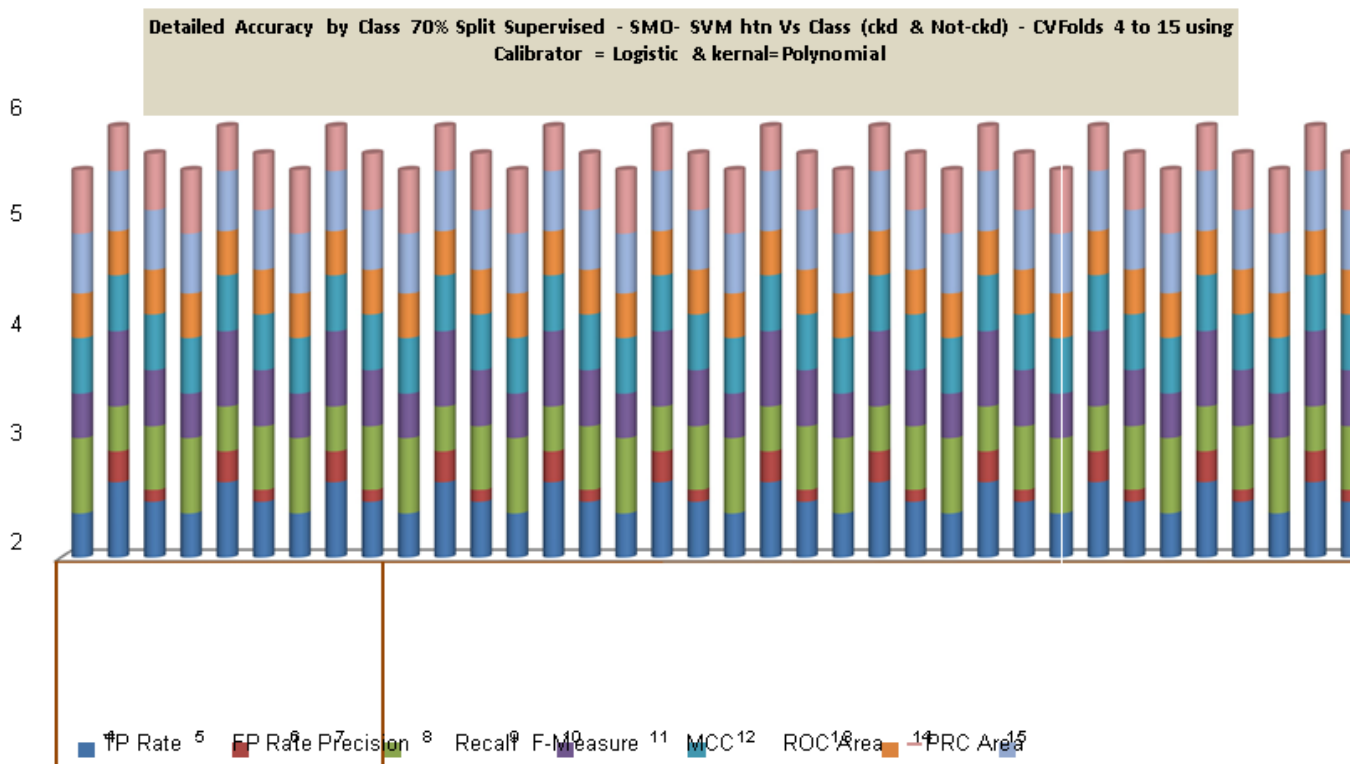


Figure: 7detailed accuracy by class Supervised SMO-SVM attributehtnVs class with CV folds 4 to 15 using Calibrator and Kernel function.

Table 2: Summary of Classifier model (full training set) for htnVs Cla s

1. Test Mode: split 70% , 2. Total Number of Instances=400

Sr. No.	Particulars	4	5	6	7	8	9	10	11	12	13	14	15

1	Correctly Classified Instances	74.25	74.25	74.25	74.25	74.25	74.25	74.25	74.25	74.25	74.25	74.25	74.25
2	Incorrectly Classified Instances	25.75	25.75	25.75	25.75	25.75	25.75	25.75	25.75	25.75	25.75	25.75	25.75
3	Kappa statistic	0.517	0.517	0.517	0.517	0.517	0.517	0.517	0.517	0.517	0.517	0.517	0.517
4	Mean absolute error	0.2575	0.2575	0.2575	0.2575	0.2575	0.2575	0.2575	0.2575	0.2575	0.2575	0.2575	0.2575
5	Root mean squared error	0.5074	0.5074	0.5074	0.5074	0.5074	0.5074	0.5074	0.5074	0.5074	0.5074	0.5074	0.5074
6	Relative absolute error	54.90	54.916	54.911	54.9098	54.91	54.9108	54.9131	54.9105	54.9107	54.9101	54.9114	54.9128
7	Root relative squared error	104.81	104.87	104.85	104.8123	104.8121	104.8128	104.8172	104.8114	104.8113	104.81098	104.8122	104.8158

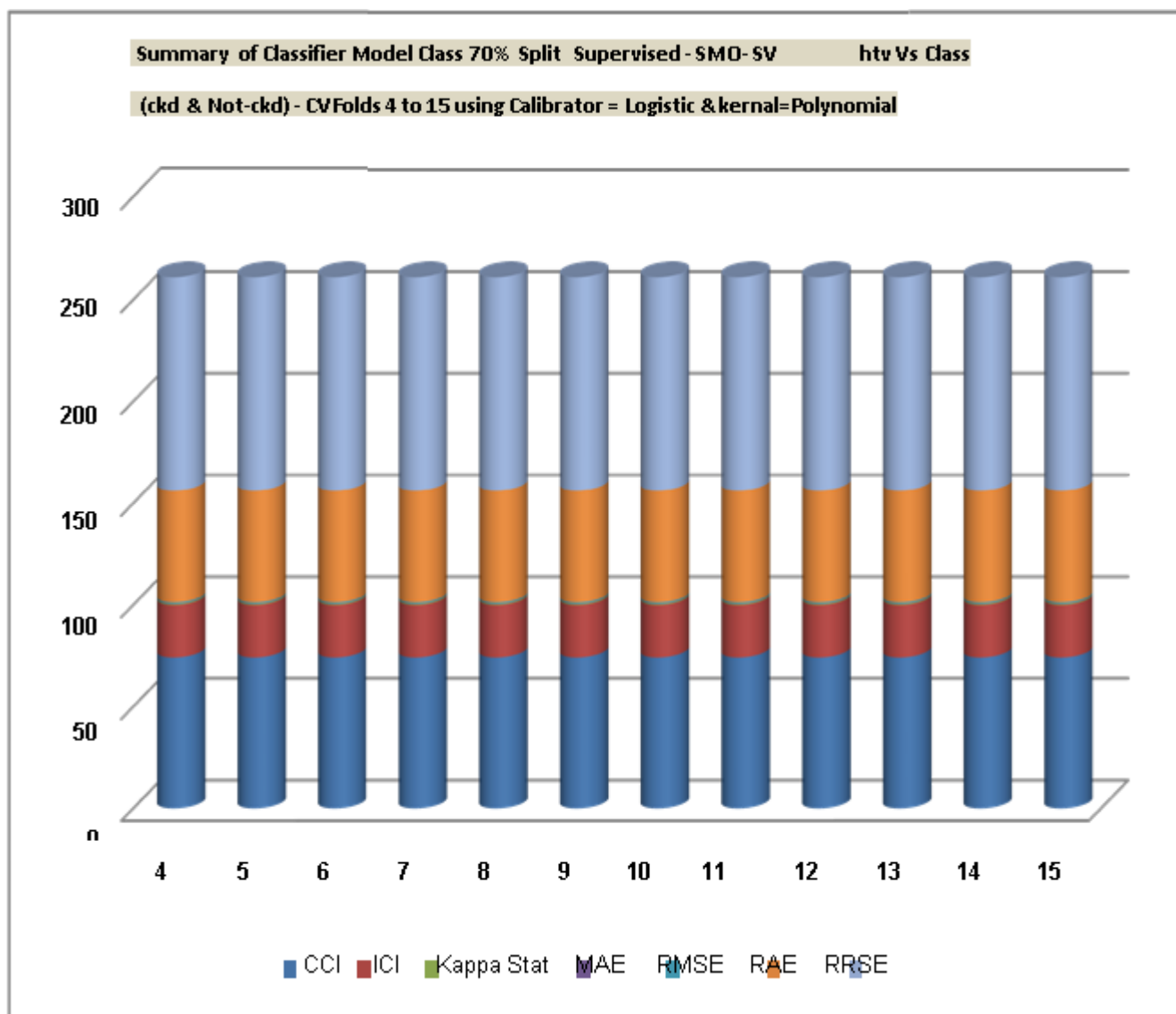


Figure: 8 Summary of classifier model class SMO-SVM attribute htnVs class with CV folds 4 to 15 using Calibrator and Kernel function.

===== CONFUSION MATRIX =====

The above result for all Cross Validation Folds from CVF = 4 to CVF=15 is the same
 Table 3: Confusion matrix (full training set) for htnVs Class

Sr. No.	CVF	Predicted (a)	Predicted (b)	< - Classified as
1	4	147	103	a = ckd
		0	150	b = not-ckd
2	5	147	103	a = ckd
		0	150	b = not-ckd
3	6	147	103	a = ckd
		0	150	b = not-ckd

4	7	147	103	a = ckd
		0	150	b = not-ckd
5	8	147	103	a = ckd
		0	150	b = not-ckd
6	9	147	103	a = ckd
		0	150	b = not-ckd
7	10	147	103	a = ckd
		0	150	b = not-ckd
8	11	147	103	a = ckd
		0	150	b = not-ckd
9	12	147	103	a = ckd
		0	150	b = not-ckd
10	13	147	103	a = ckd
		0	150	b = not-ckd
11	14	147	103	a = ckd
		0	150	b = not-ckd
12	15	147	103	a = ckd
		0	150	b = not-ckd

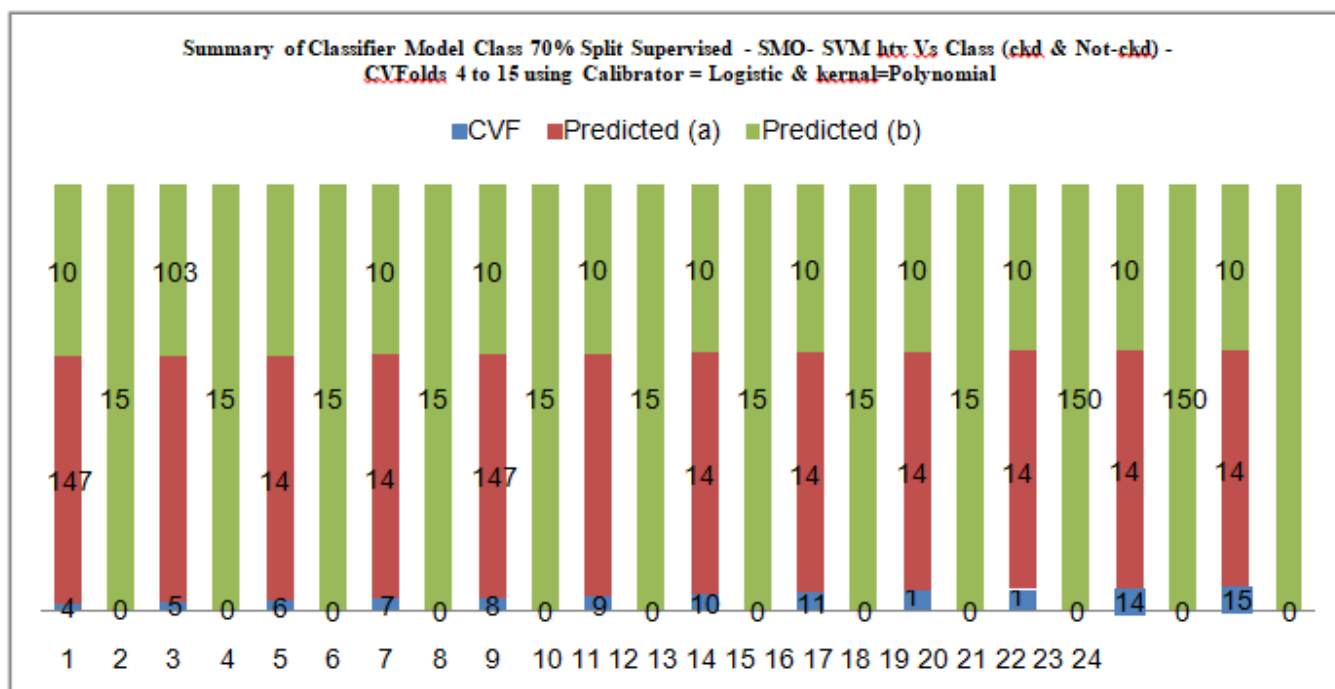


Figure: 9Confusion matrix SMO-SVM attribute htnVs class with CV folds 4 to 15 using Calibrator and Kernel function

Supervised- classify – SMO- SVM- 2 Attributes – htn, appetVs Class Applying Cross validation from 4 to 15 + 70% split

Calibrator: Logistic and Kernel: Polynomial

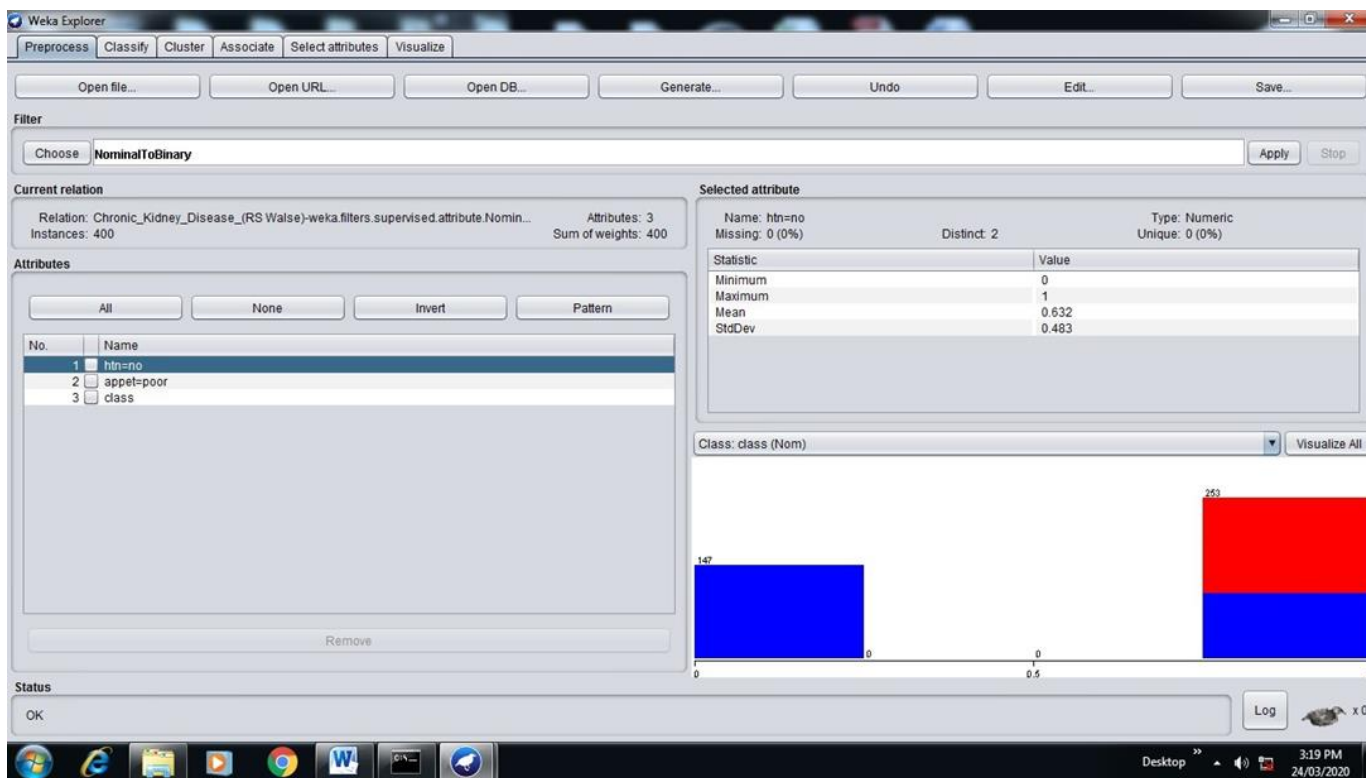


Figure: 10 Preprocess of imported data in WEKA selected two attributes

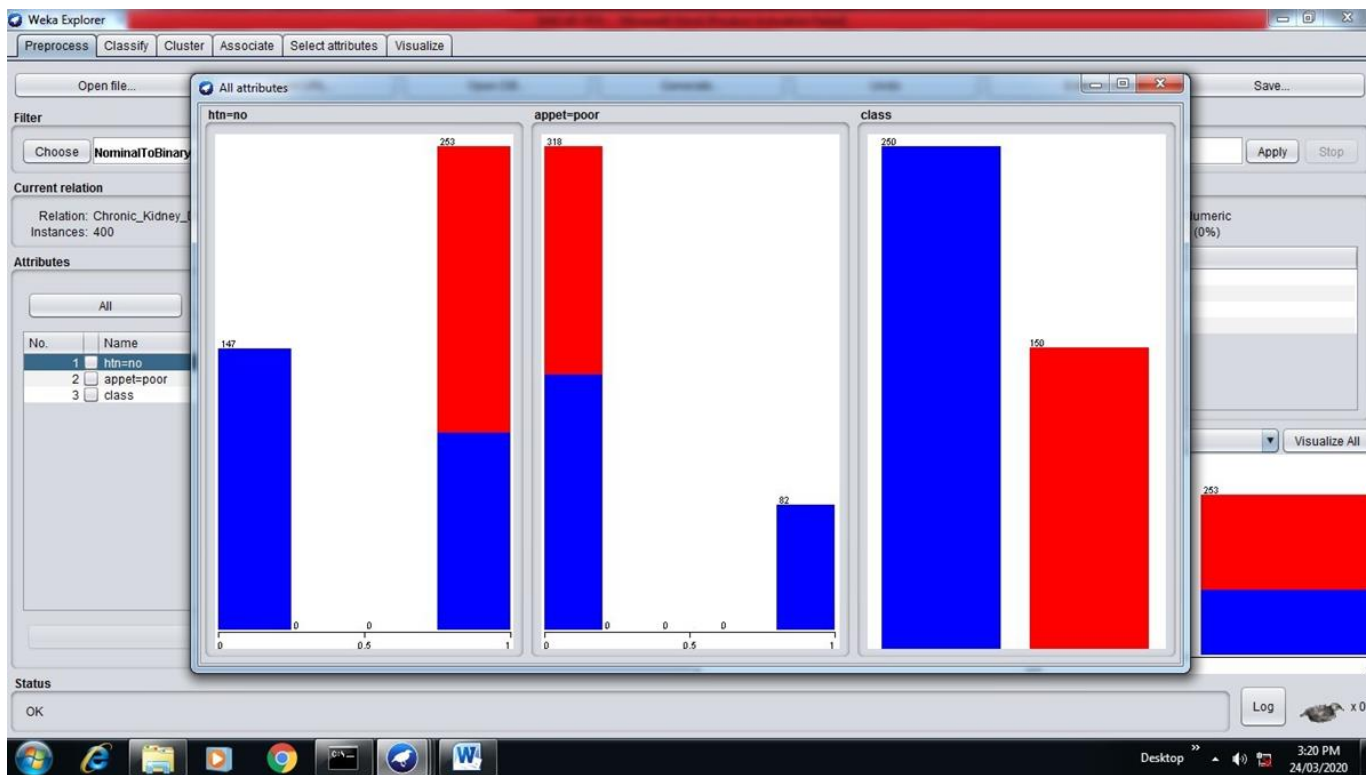


Figure: 11 Histogram of Two attributes Vs Class

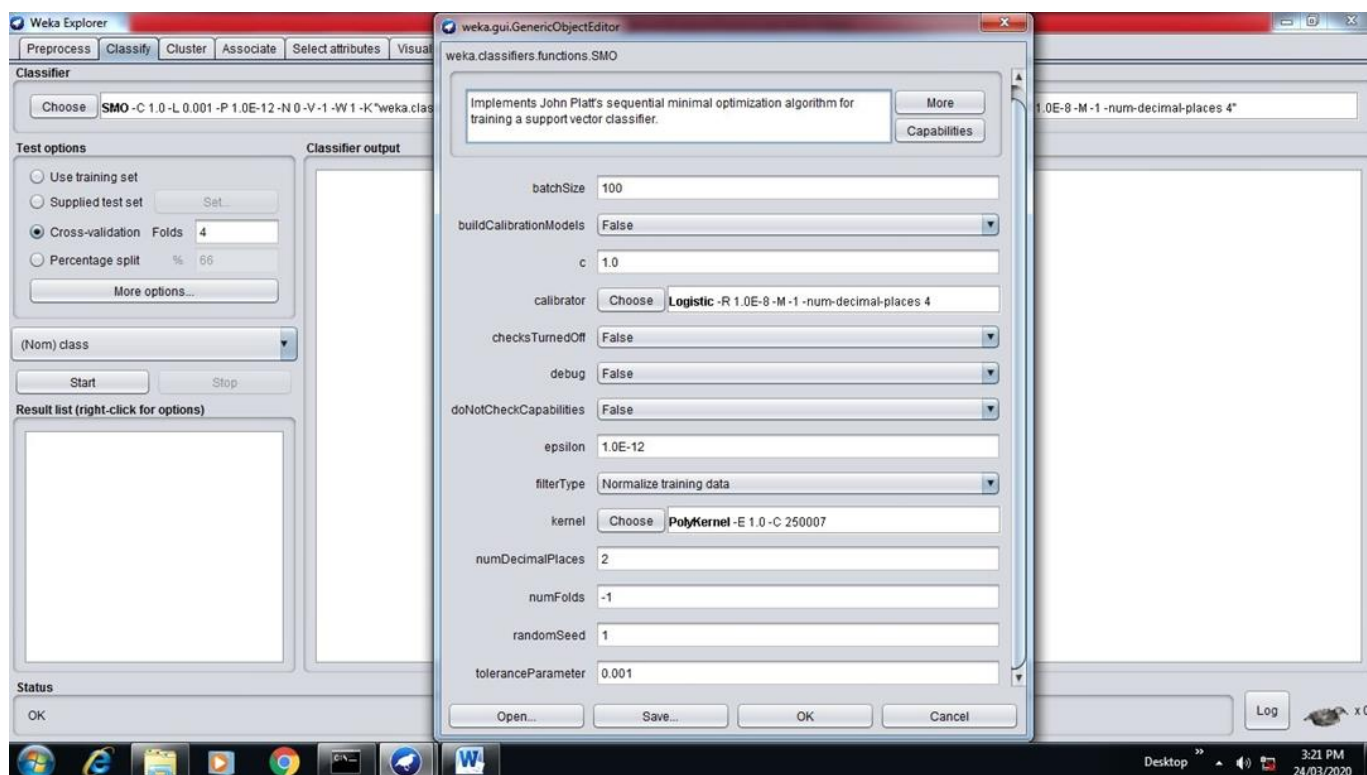


Figure: 12 SMO algorithm with properties of Logistic and Polynomial function

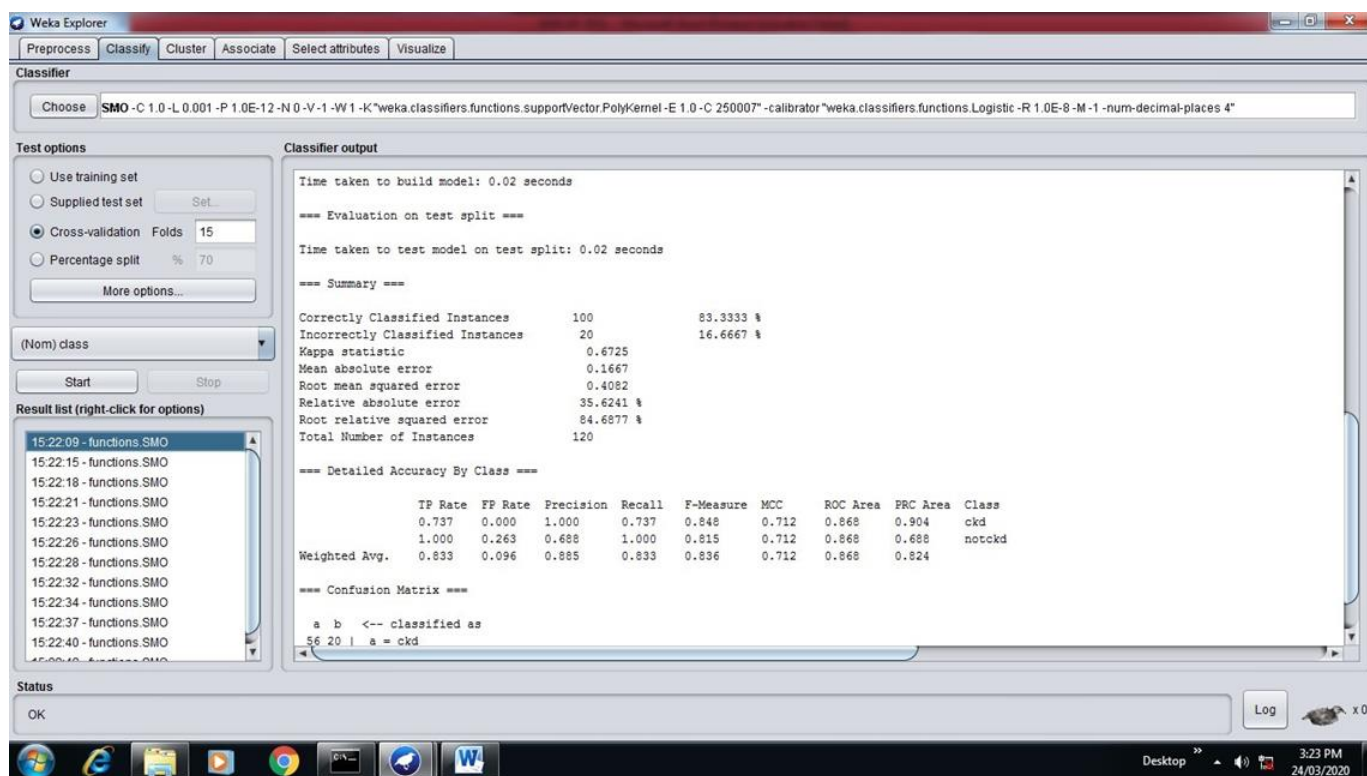


Figure: 13 Output SMO classifier with two attributes with class

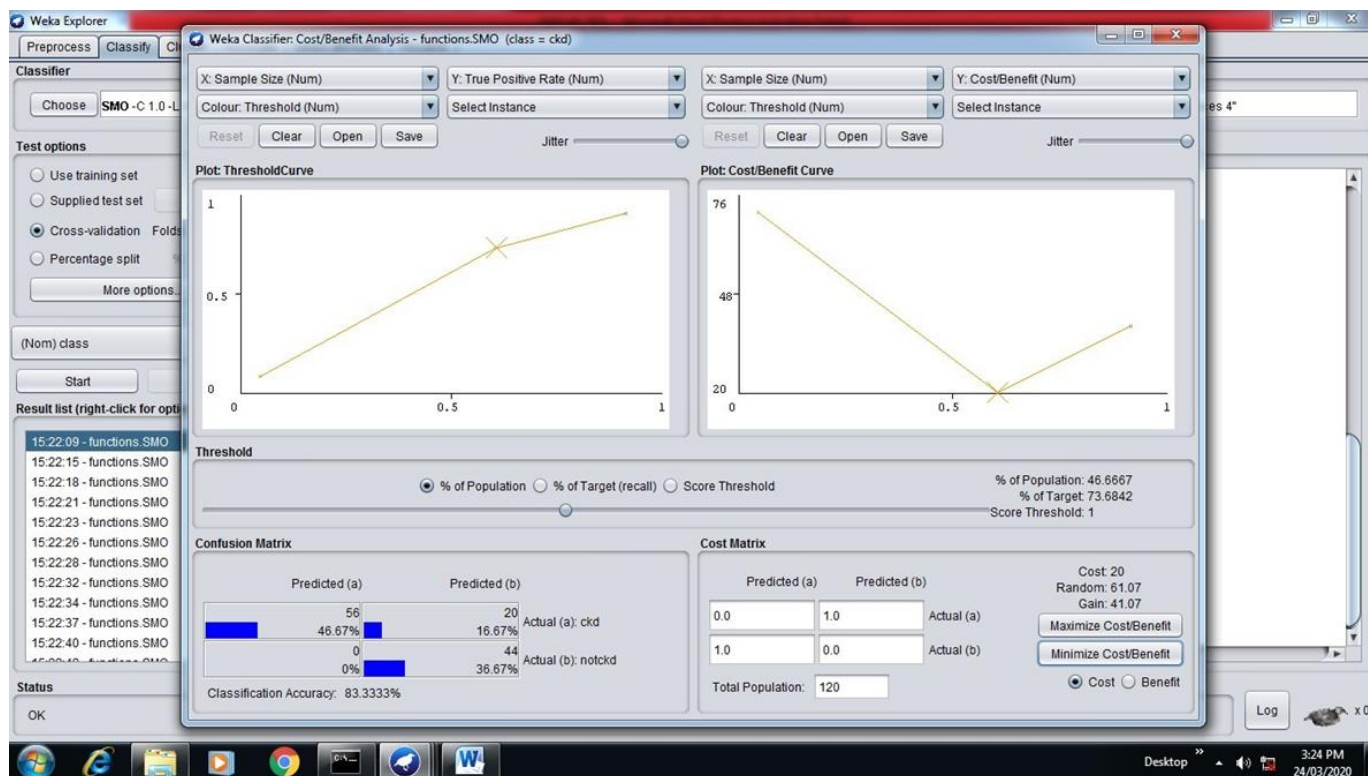


Figure: 14 Weka classifier: cost / benefit analysis – function SMO class as ckd

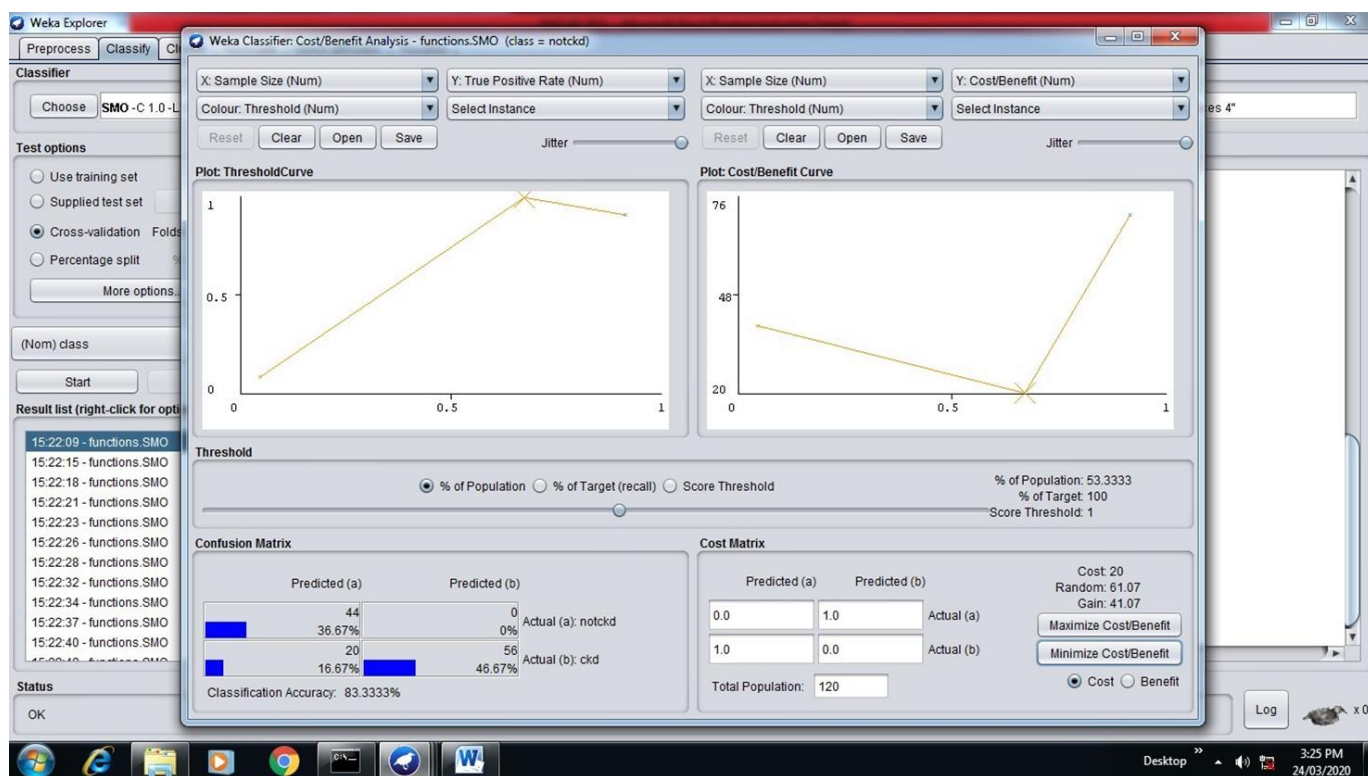


Figure: 15 Weka classifier: cost / benefit analysis – function SMO class as not-ckd

Attributes: 3, htn=noappet=poorclass Test mode: split 70.0% train, remainder test
 === Classifier model (full training set) ===, SMO , Kernel used: Linear Kernel: $K(x,y) = \langle x,y \rangle$
 Classifier for classes: ckd, notckd, BinarySMO , Machine linear: showing attribute weights, not support vectors.
 2 * (normalized) htn=no

+ -2 * (normalized) appet=poor
- 1

COMPARATIVE STUDY SUPERVISED OF SMO, CALIBRATOR: LOGISTIC AND KERNEL: POLYNOMICAL USING CROSS VALIDATION FOLD 4 TO 15 WITH 70% OF SPLIT CLASSIFIER

Table 4:Detailed accuracy by Class with two attributes

Cross Validation Fold	Class	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area
4	CKD	0.737	0.000	1.000	0.737	0.848	0.712	0.868	0.904
	Not-CKD	1.000	0.263	0.688	1.000	0.815	0.172	0.868	0.688
	Weight Avg	0.833	0.096	0.886	0.833	0.836	0.172	0.868	0.824
5	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.799
6	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
7	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
8	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
9	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
10	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
11	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
12	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
13	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
14	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883
	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794
	CKD	0.688	0.000	1.000	0.688	0.815	0.673	0.844	0.883

15	Not-CKD	1.000	0.312	0.658	1.000	0.794	0.673	0.844	0.658
	Weight Avg	0.805	0.117	0.872	0.805	0.807	0.673	0.844	0.794

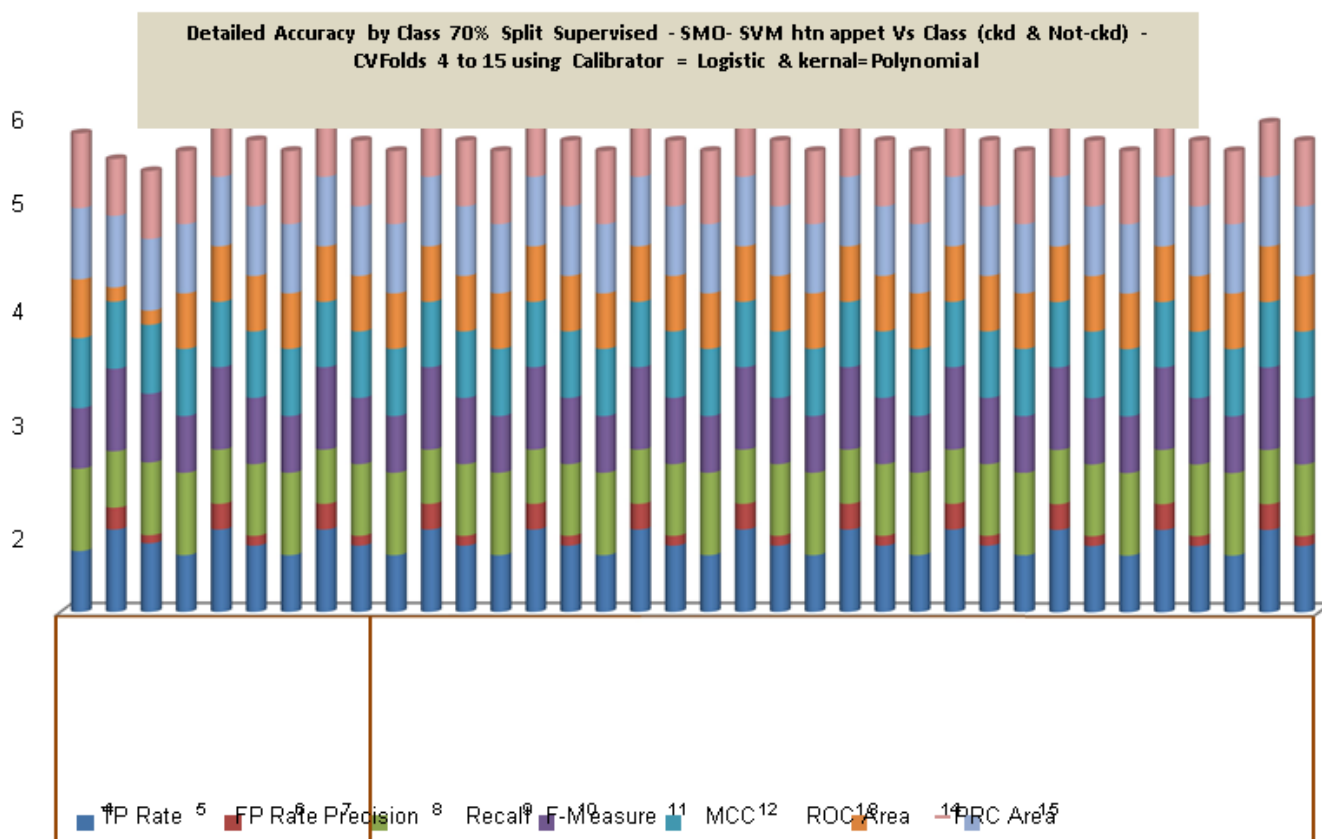


Figure: 16 Detailed accuracy by class with two attributes, CVFolds 4 to 15

Table 5: Summary of Classifier model (full training set) with two attributes

1. Test Mode: split 70%

Sr. No.	Particulars	4	5	6	7	8	9	10	11	12	13	14	15
1	Time taken to test model	0.02	0.03	0.01	0	0.02	0	0.03	0.01	0	0.02	0	0

2	Correctly Classified Instances	83.33	80.50	80.50	80.50	80.50	80.50	80.50	80.50	80.50	80.50	80.50	80.50
3	Incorrectly Classified Instances	16.66	19.5	19.5	19.5	19.5	19.5	19.5	19.5	19.5	19.5	19.5	19.5
4	Kappa statistic	0.6725	0.6232	0.6232	0.6232	0.6232	0.6232	0.6232	0.6232	0.6232	0.6232	0.6232	0.6232
5	Mean absolute error	0.1667	0.195	0.195	0.195	0.195	0.195	0.195	0.195	0.195	0.195	0.195	0.195
6	Root mean squared error	0.4082	0.4416	0.4416	0.4416	0.4416	0.4416	0.4416	0.4416	0.4416	0.4416	0.4416	0.4416
7	Relative absolute error	35.6241	41.5828	41.5828	41.5828	41.5828	41.5828	41.5828	41.5828	41.5828	41.5828	41.5828	41.5828
8	Root relative squared error	84.6877	91.2139	91.2139	91.2139	91.2139	91.2139	91.2139	91.2139	91.2139	91.2139	91.2139	91.2139

9	Total No. of Instances	120	40	40	40	400	400	400	400	400	400	400	400	400
---	------------------------	-----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----

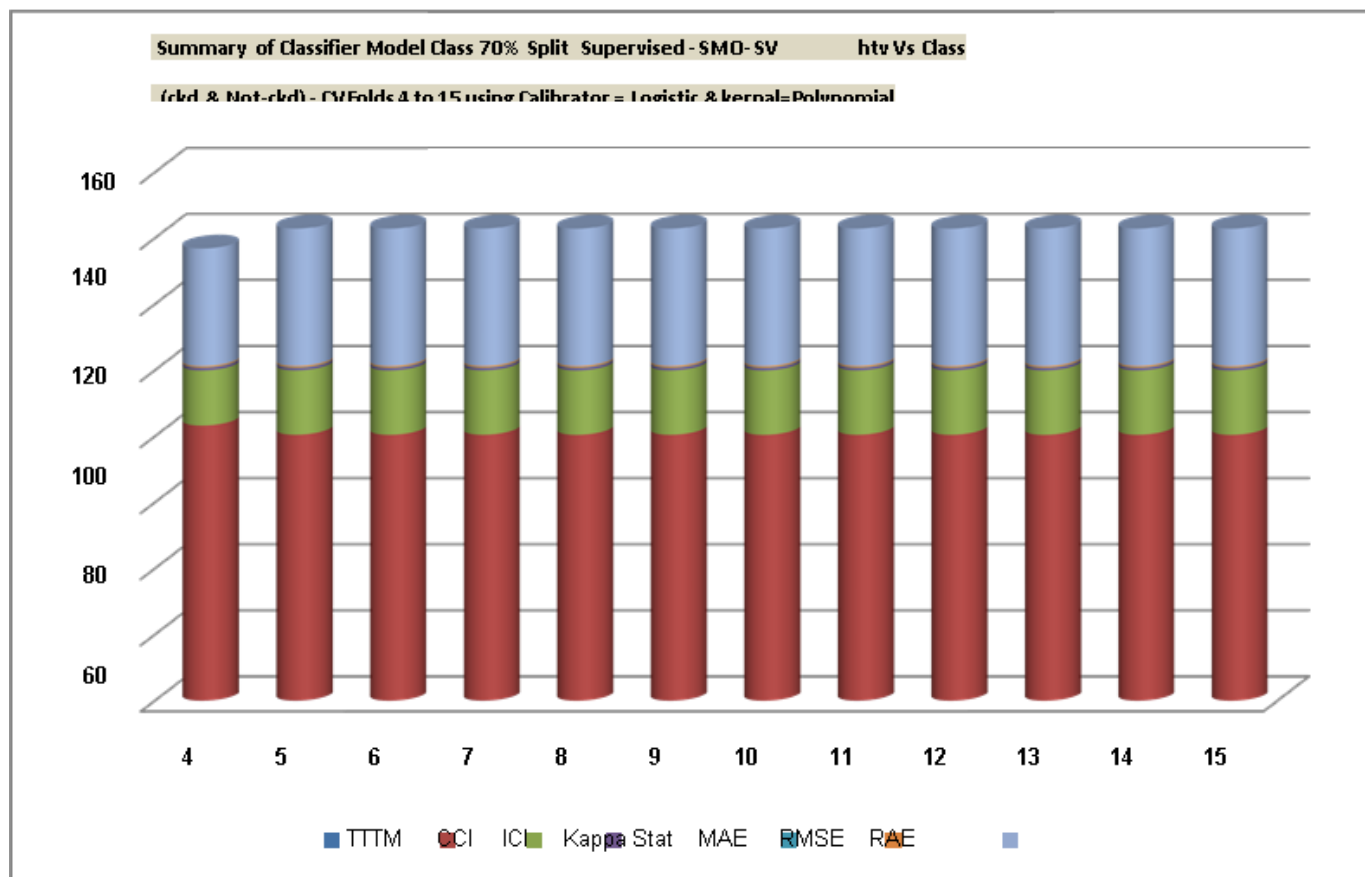


Figure: 17 Summary of Classifier Model with two attributes, CVFolds 4 to 15

===== CONFUSION MATRIX =====

Table 6: Values of Confusion matrix with two attributes

Sr. No.	CVF	Predicted (a)	Predicted (b)	< - Classified as
1	4	147	103	a = ckd
		0	150	b = not-ckd
2	5	172	78	a = ckd
		0	150	b = not-ckd
3	6	172	78	a = ckd
		0	150	b = not-ckd
4	7	172	78	a = ckd
		0	150	b = not-ckd
5	8	172	78	a = ckd
		0	150	b = not-ckd
6	9	172	78	a = ckd
		0	150	b = not-ckd
7	10	172	78	a = ckd

		0	150	b = not-ckd
8	11	172	78	a = ckd
		0	150	b = not-ckd
9	12	172	78	a = ckd
		0	150	b = not-ckd
10	13	172	78	a = ckd
		0	150	b = not-ckd
11	14	172	78	a = ckd
		0	150	b = not-ckd
12	15	172	78	a = ckd
		0	150	b = not-ckd

The above result for all Cross Validation Folds from CVF = 4 to CVF=15 is the same

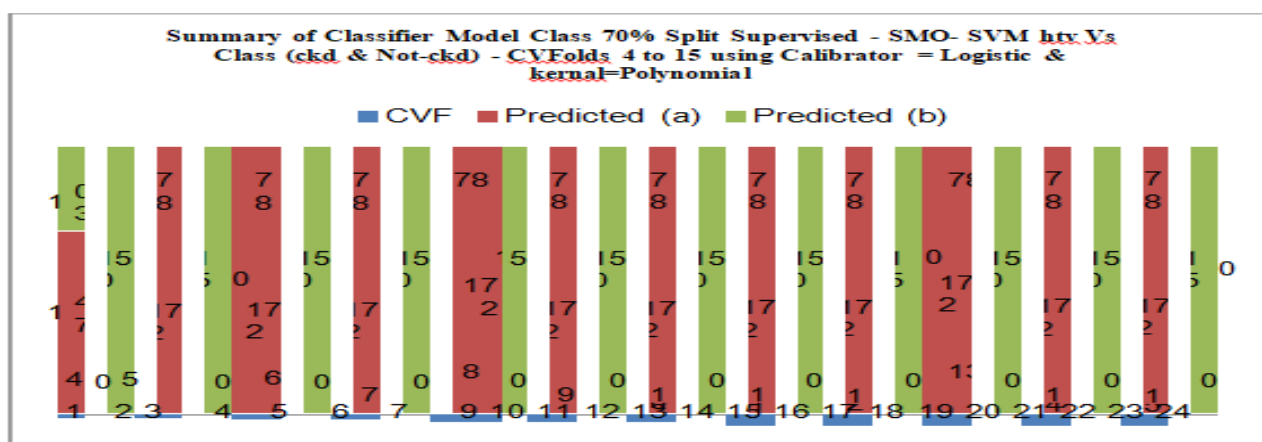


Figure: 18 Summary of Classifier Model with two attributes, CVFolds 4 to 15 Step-3
 Step-3 (24-Full Vs Class) – Full Attribute
 Supervised- classify – SMO- SVM- Full 24 Attributes – 24 Attributes Vs Class (ckd and not-ckd) Applying Cross validation from 4 to 15 + 70% split
 Calibrator: Logistic and Kernel: Polynomial

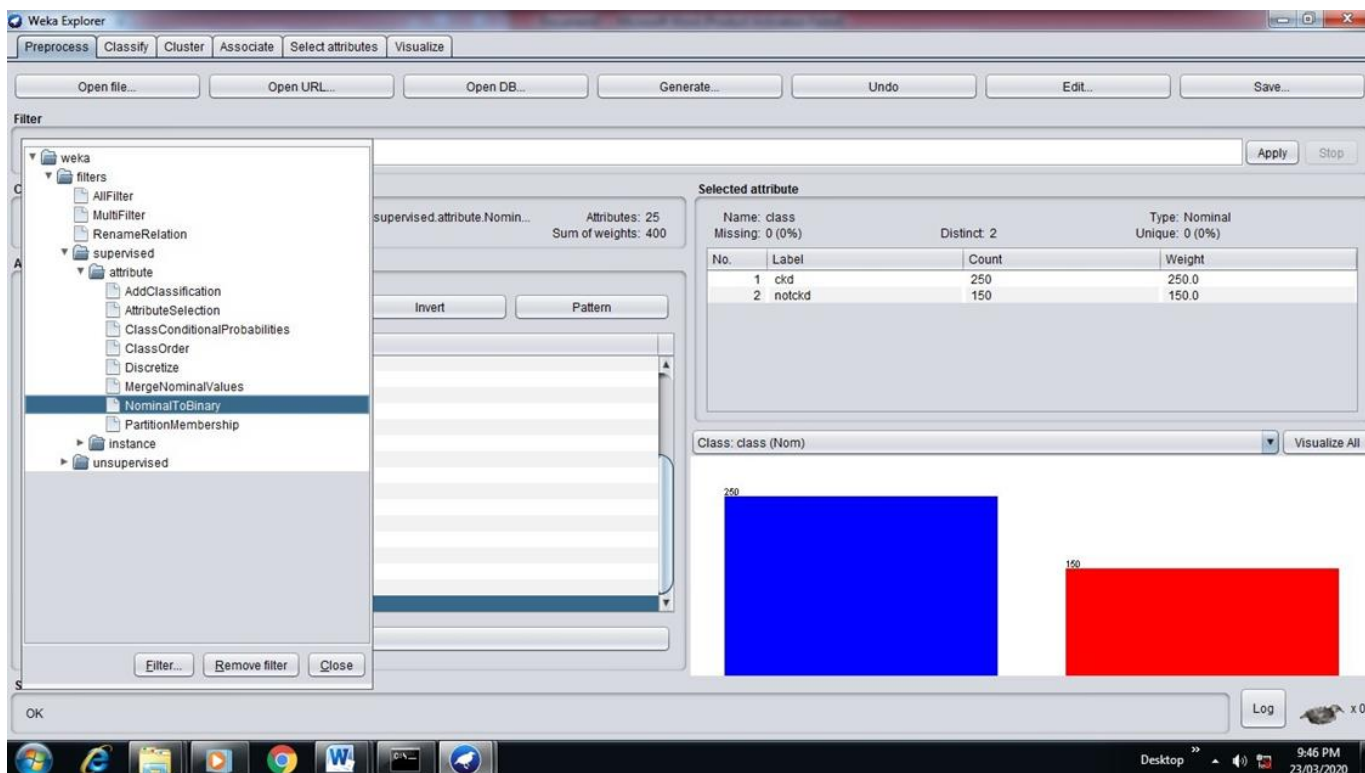


Figure: 19 Preprocess of data filter attributes with Nominal to binary full attributes

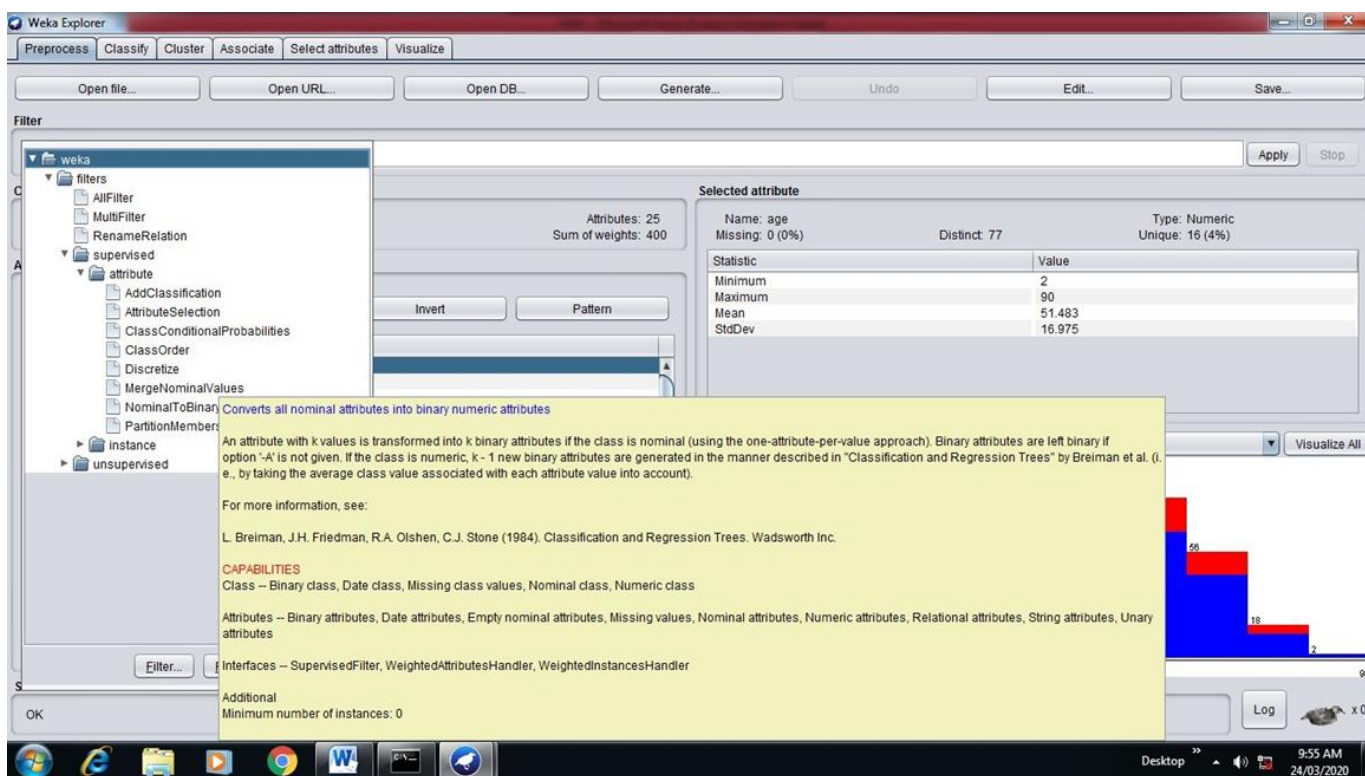


Figure: 20 Convert all nominal attributes into binary numeric attributes

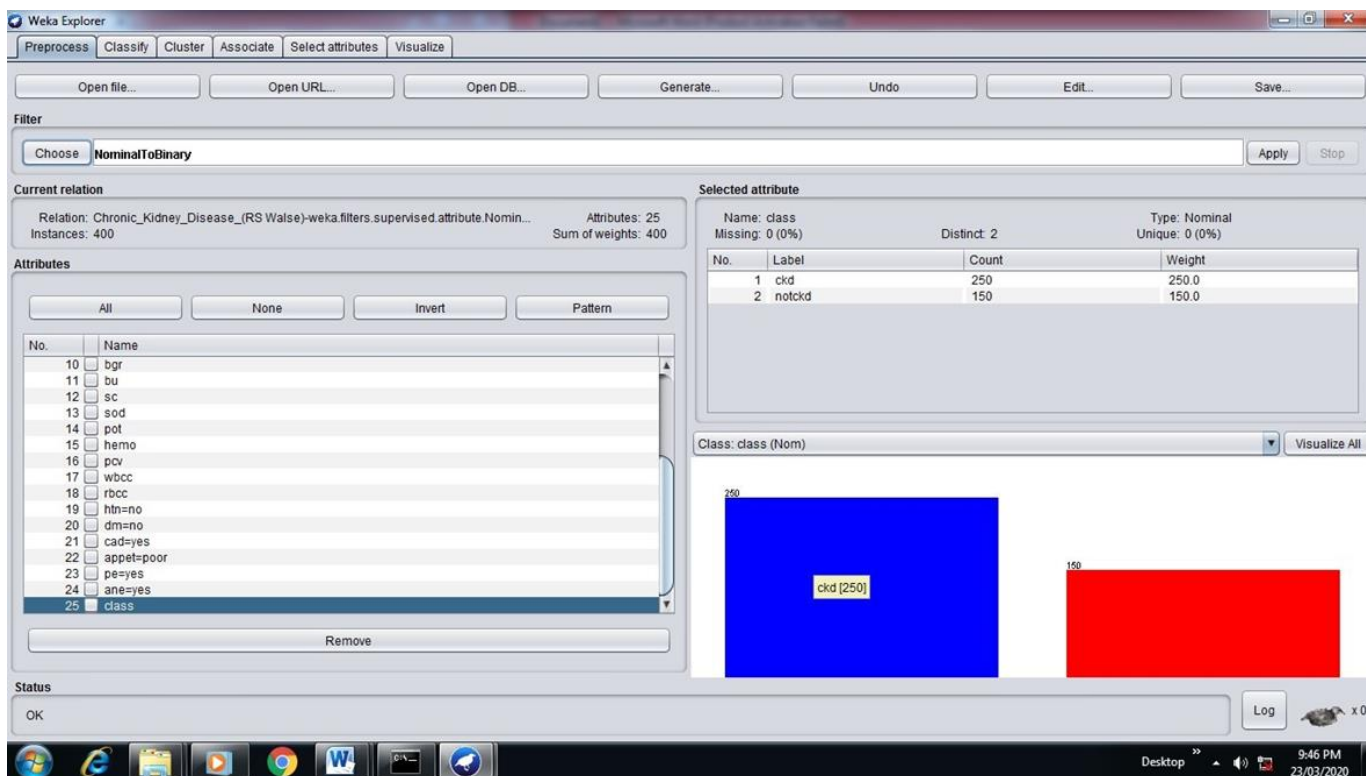


Figure: 21 Convert all nominal attributes in to binary numeric attributes

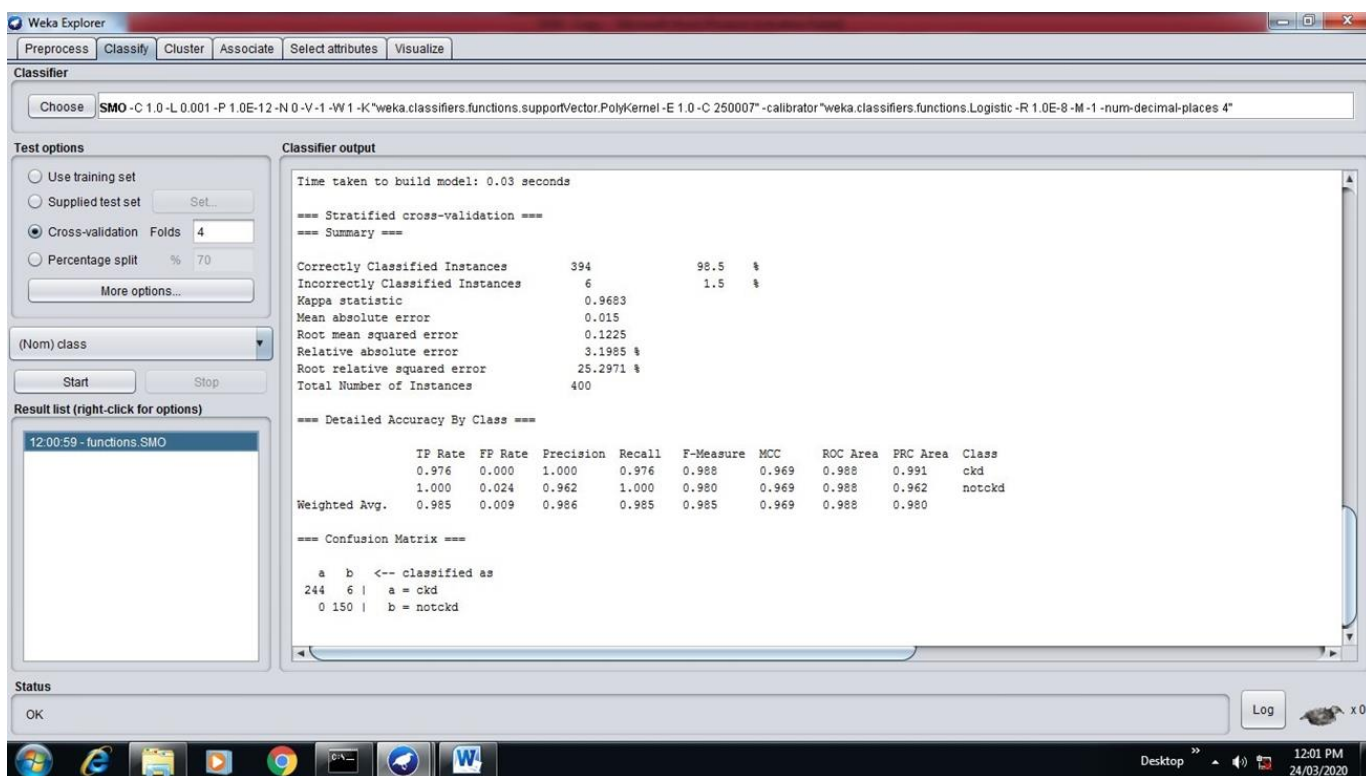


Figure: 22 Result of SMO classifier with Full attribute Vs Class

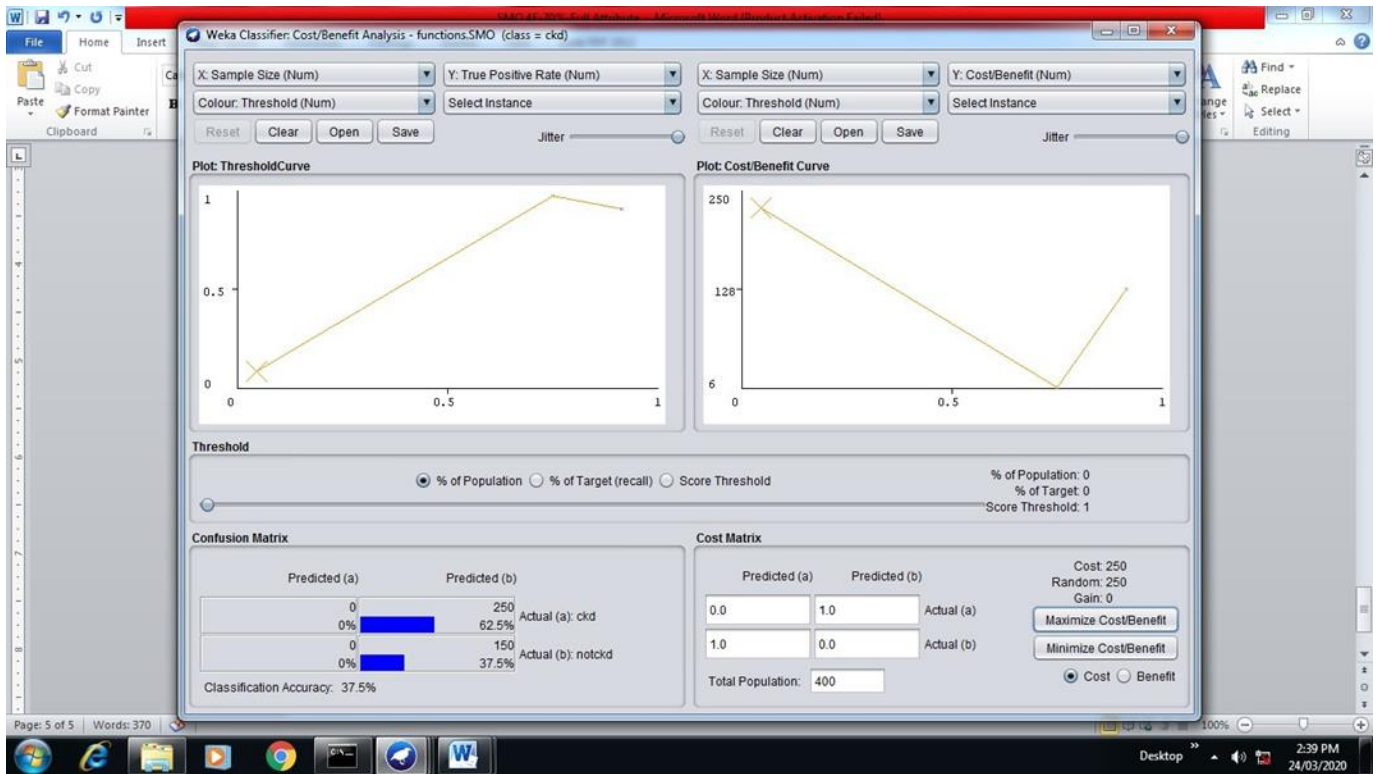


Figure: 23 Weka classifier cost benefit function smo class-ckd Full attribute Vs Class

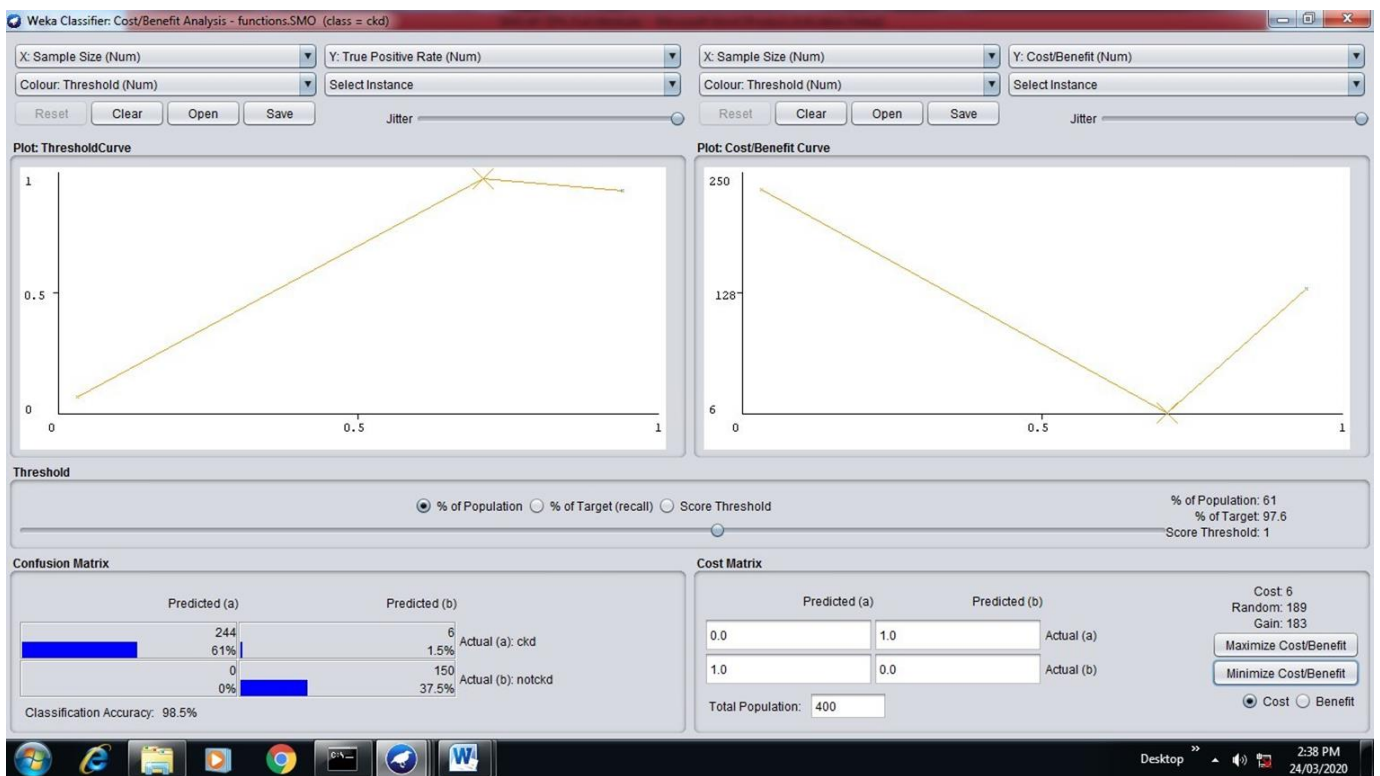


Figure: 24 Weka classifier cost benefit function smo class-Not-Ckd Full attribute Vs Class

=== Classifier model (full training set) ===, SMO ,Kernel used:Linear Kernel: $K(x,y) = \langle x,y \rangle$
 Classifier for classes: ckd, notckd BinarySMO, Machine linear: showing attribute weights, not support vectors.
 $2 * (\text{normalized}) \text{htn} = \text{no}, -1$

COMPARATIVE STUDY SUPERVISED OF SMO, CALBRATOR: LOGISTIC AND KERNEL: POLYNOMICAL USING CROSS VALIDATION FOLD 4 TO 15 WITH 70% OF SPLIT CLASSIFIER

Table 7:Detailed accuracy by Class with full attributes Vs Class (ckd-not-ckd)

Cross Validation Fold	Class	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area
4	CKD	0.976	0.000	1.000	0.976	0.988	0.969	0.988	0.991
	Not-CKD	1.000	0.024	0.962	1.000	0.980	0.969	0.988	0.962
	Weight Avg	0.985	0.009	0.986	0.985	0.985	0.969	0.998	0.980
5	CKD	0.980	0.000	1.000	0.980	0.990	0.974	0.990	0.992
	Not-CKD	1.000	0.020	0.968	1.000	0.984	0.974	0.990	0.968
	Weight Avg	0.988	0.008	0.988	0.988	0.988	0.974	0.990	0.983
6	CKD	0.980	0.000	1.000	0.980	0.990	0.974	0.990	0.992
	Not-CKD	1.000	0.020	0.968	1.000	0.984	0.974	0.990	0.968
	Weight Avg	0.988	0.008	0.988	0.988	0.988	0.974	0.990	0.983
7	CKD	0.976	0.000	1.000	0.976	0.988	0.969	0.988	0.991
	Not-CKD	1.000	0.024	0.962	1.000	0.980	0.969	0.988	0.962
	Weight Avg	0.985	0.009	0.986	0.985	0.985	0.969	0.998	0.980
8	CKD	0.980	0.000	1.000	0.980	0.990	0.974	0.990	0.992
	Not-CKD	1.000	0.020	0.968	1.000	0.984	0.974	0.990	0.968
	Weight Avg	0.988	0.008	0.988	0.988	0.988	0.974	0.990	0.983
9	CKD	0.980	0.000	1.000	0.980	0.990	0.974	0.990	0.992
	Not-CKD	1.000	0.020	0.968	1.000	0.984	0.974	0.990	0.968
	Weight Avg	0.988	0.008	0.988	0.988	0.988	0.974	0.990	0.983
10	CKD	0.980	0.000	1.000	0.980	0.990	0.974	0.990	0.992
	Not-CKD	1.000	0.020	0.968	1.000	0.984	0.974	0.990	0.968
	Weight Avg	0.988	0.008	0.988	0.988	0.988	0.974	0.990	0.983
11	CKD	0.976	0.000	1.000	0.976	0.988	0.969	0.988	0.991
	Not-CKD	1.000	0.024	0.962	1.000	0.980	0.969	0.988	0.962
	Weight Avg	0.985	0.009	0.986	0.985	0.985	0.969	0.998	0.980
12	CKD	0.976	0.000	1.000	0.976	0.988	0.969	0.988	0.991
	Not-CKD	1.000	0.024	0.962	1.000	0.980	0.969	0.988	0.962
	Weight Avg	0.985	0.009	0.986	0.985	0.985	0.969	0.998	0.980

13	CKD	0.976	0.000	1.000	0.976	0.988	0.969	0.988	0.991
	Not-CKD	1.000	0.024	0.962	1.000	0.980	0.969	0.988	0.962
	Weight Avg	0.985	0.009	0.986	0.985	0.985	0.969	0.998	0.980
14	CKD	0.980	0.000	1.000	0.980	0.990	0.974	0.990	0.992
	Not-CKD	1.000	0.020	0.968	1.000	0.984	0.974	0.990	0.968
	Weight Avg	0.988	0.008	0.988	0.988	0.988	0.974	0.990	0.983
15	CKD	0.980	0.000	1.000	0.980	0.990	0.974	0.990	0.992
	Not-CKD	1.000	0.020	0.968	1.000	0.984	0.974	0.990	0.968
	Weight Avg	0.988	0.008	0.988	0.988	0.988	0.974	0.990	0.983

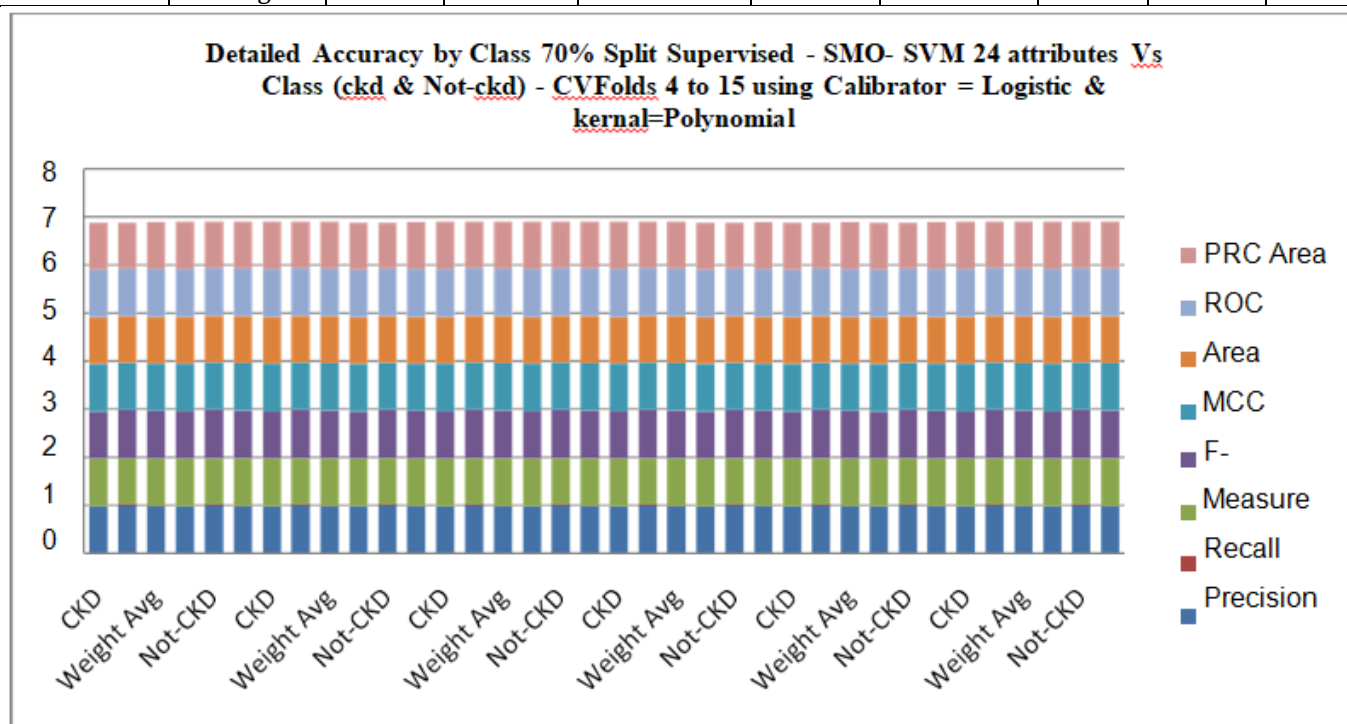


Figure: 25 Detailed accuracy by class Full attribute Vs Class

Table 8: Summary of Classifier model (full training set) with full attributes

1. Test Mode: split 70% , 2. Total Number of Instances=400

Sr. No	Particulars	4	5	6	7	8	9	10	11	12	13	14	15
1	Time taken to build model	0.03	0.03	0.03	0.03	0.02	0.02	0.03	0.03	0.03	0.02	0.03	0.03
2	Classified Instances Correctly	98.50	98.75	98.75	98.50	98.75	98.75	98.75	98.50	98.50	98.50	98.75	98.75
3	Classified Instances Incorrectly	1.50	1.25	1.25	1.50	1.25	1.25	1.25	1.50	1.50	1.50	1.25	1.25
4	Classified Instances Kappa statistic	0.9683	0.973	0.9735	0.968	0.9735	0.973	0.973					
5	Mean absolute error	0.015	5	0.0125	0.015	0.0125	5	5					
6	Root mean squared error	3.1985	2.665	2.6656	3.198	2.6655	2.665	2.665	3.1987	3.1987	3.1986	2.6656	2.6657
8	Relative absolute error	25.297	23.09	23.093	25.29	23.092	23.099	23.099	25.296	25.296	25.296	23.092	23.093

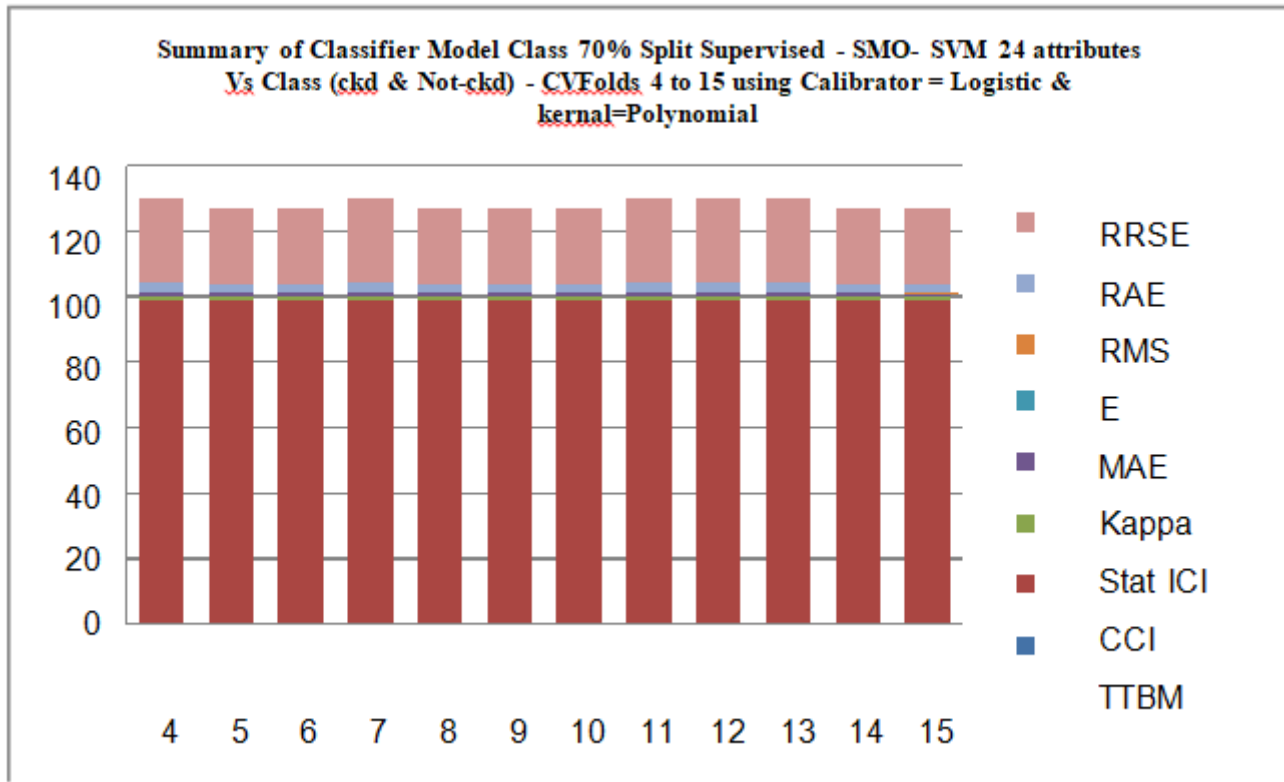


Figure: 26 Summary of Classifier Model with Full attribute Vs Class

===== CONFUSION MATRIX =====

The above result for all Cross Validation Folds from CVF = 4 to CVF=15 is the same

Table 9: Predicted values of Confusion matrix full attributes Vs Class (ckd-not-ckd)

Sr. No.	CVF	Predicted (a)	Predicted (b)	< - Classified as
1	4	244	6	a = ckd
		0	150	b = not-ckd
2	5	245	5	a = ckd

		0	150	b = not-ckd
3	6	245	5	a = ckd
		0	150	b = not-ckd
4	7	244	6	a = ckd
		0	150	b = not-ckd
5	8	245	5	a = ckd
		0	150	b = not-ckd
6	9	245	5	a = ckd
		0	150	b = not-ckd
7	10	245	5	a = ckd
		0	150	b = not-ckd
8	11	244	6	a = ckd
		0	150	b = not-ckd
9	12	244	6	a = ckd
		0	150	b = not-ckd
10	13	244	6	a = ckd
		0	150	b = not-ckd
11	14	245	5	a = ckd
		0	150	b = not-ckd
12	15	245	5	a = ckd
		0	150	b = not-ckd

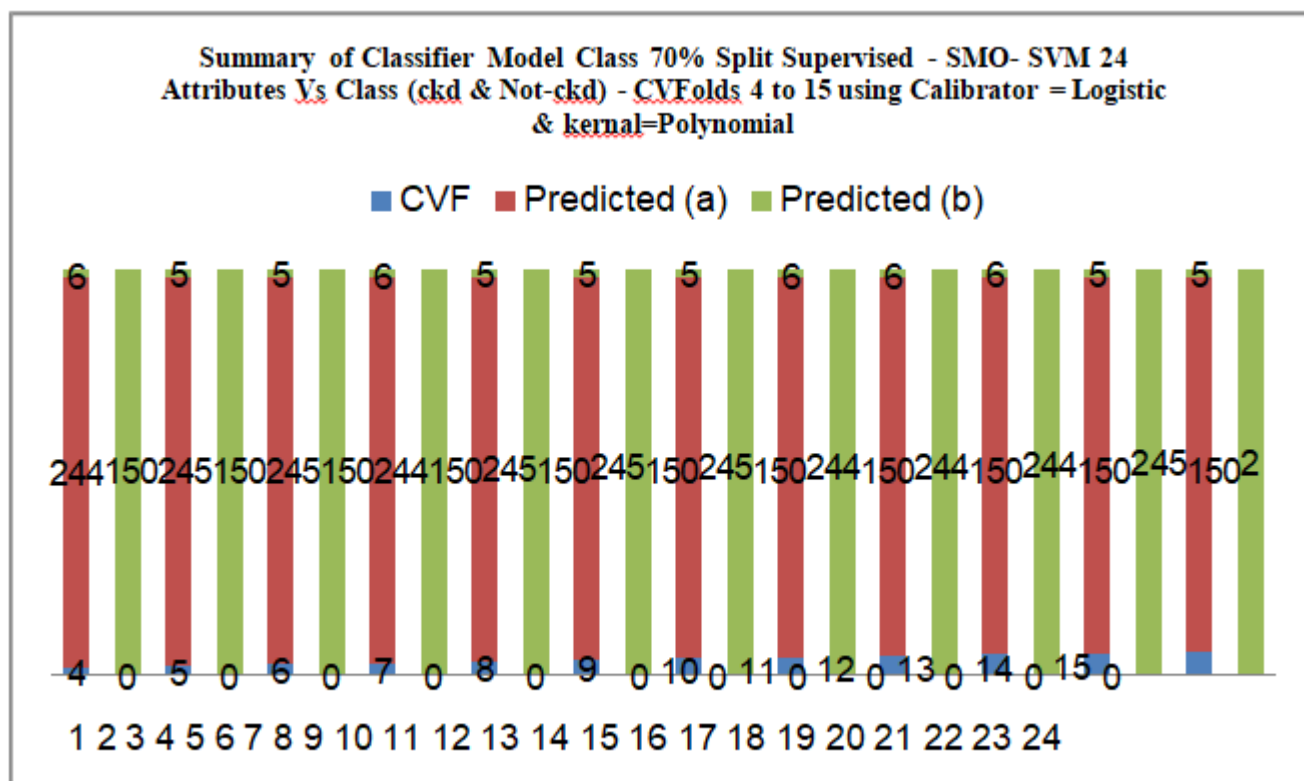


Figure: 27 Summary of Classifier Model with Full attribute Vs Class

Unsupervised – Clustering – Hierarchical –Euclidean & Manhattan Distance Function using No of Clusters 2,3,4& 5 test results: Full Attributes

Step 1: Using Euclidean Distance function No of Clusters 2,3,4& 5

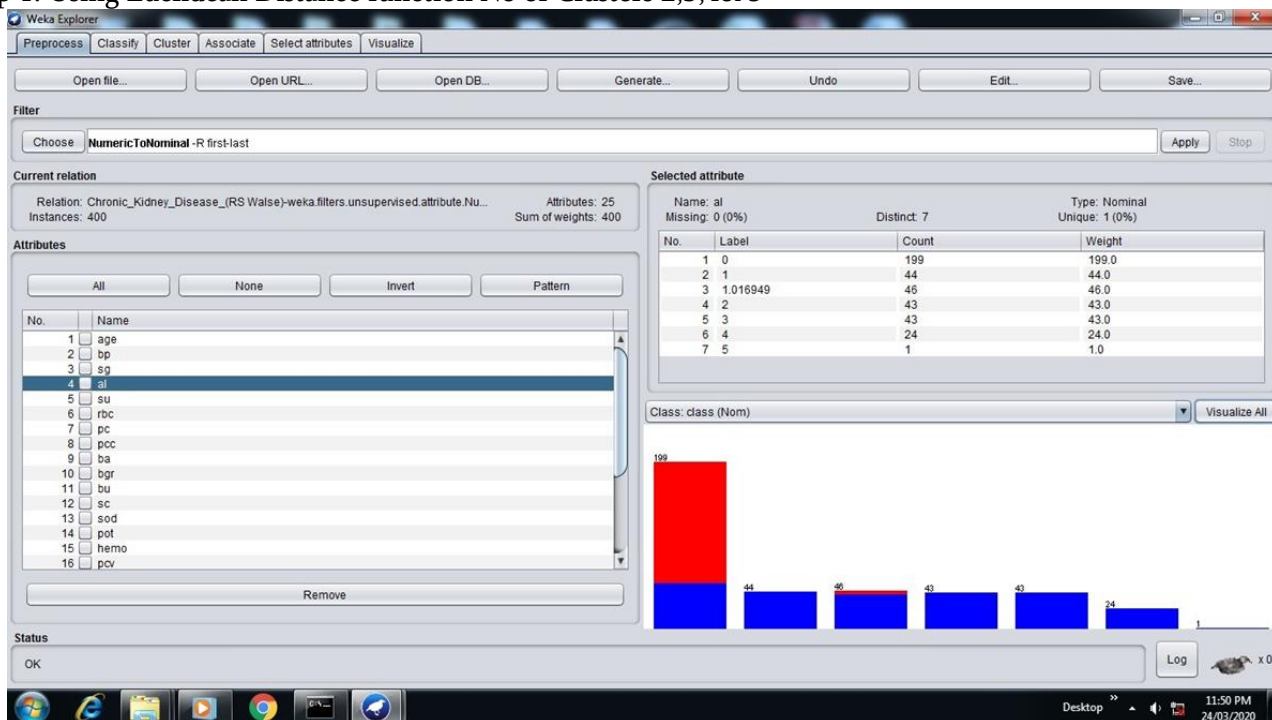


Figure: 28Preprocess Full attribute Vs Class

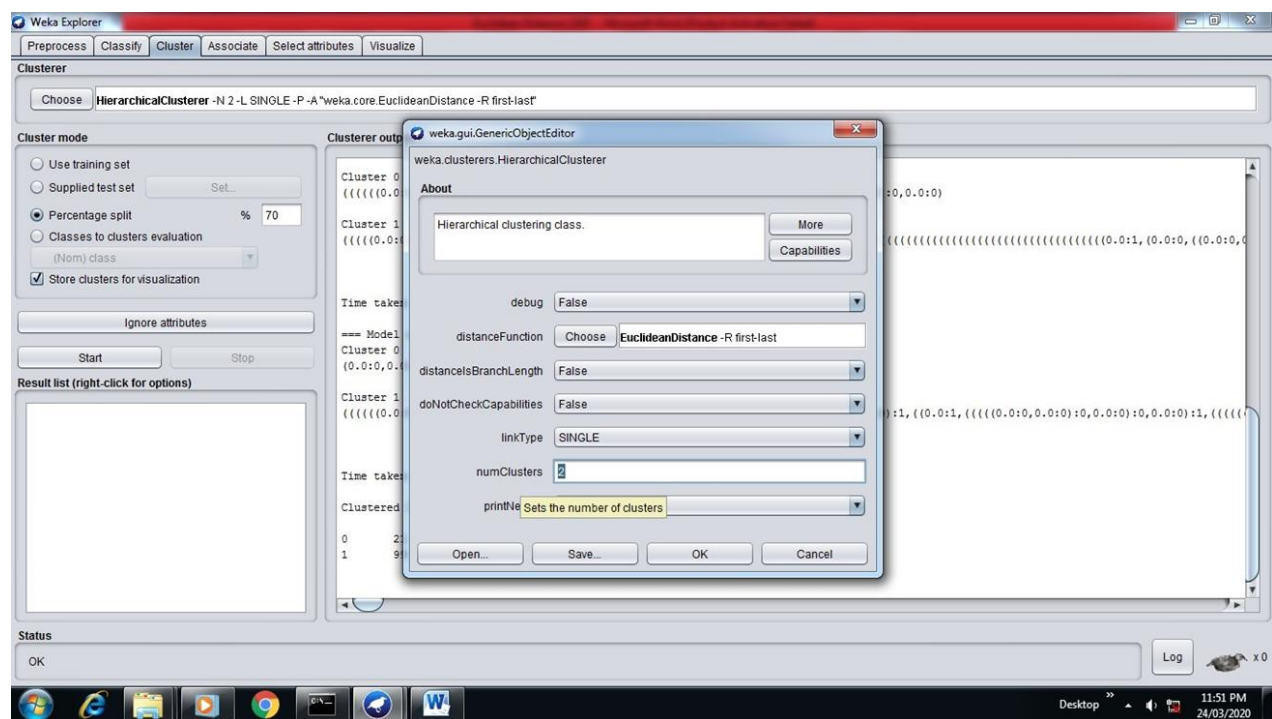


Figure: 29Hierarchical clustering class with Euclidean function of Full attribute Vs Class with no of cluster 2,3,4& 5

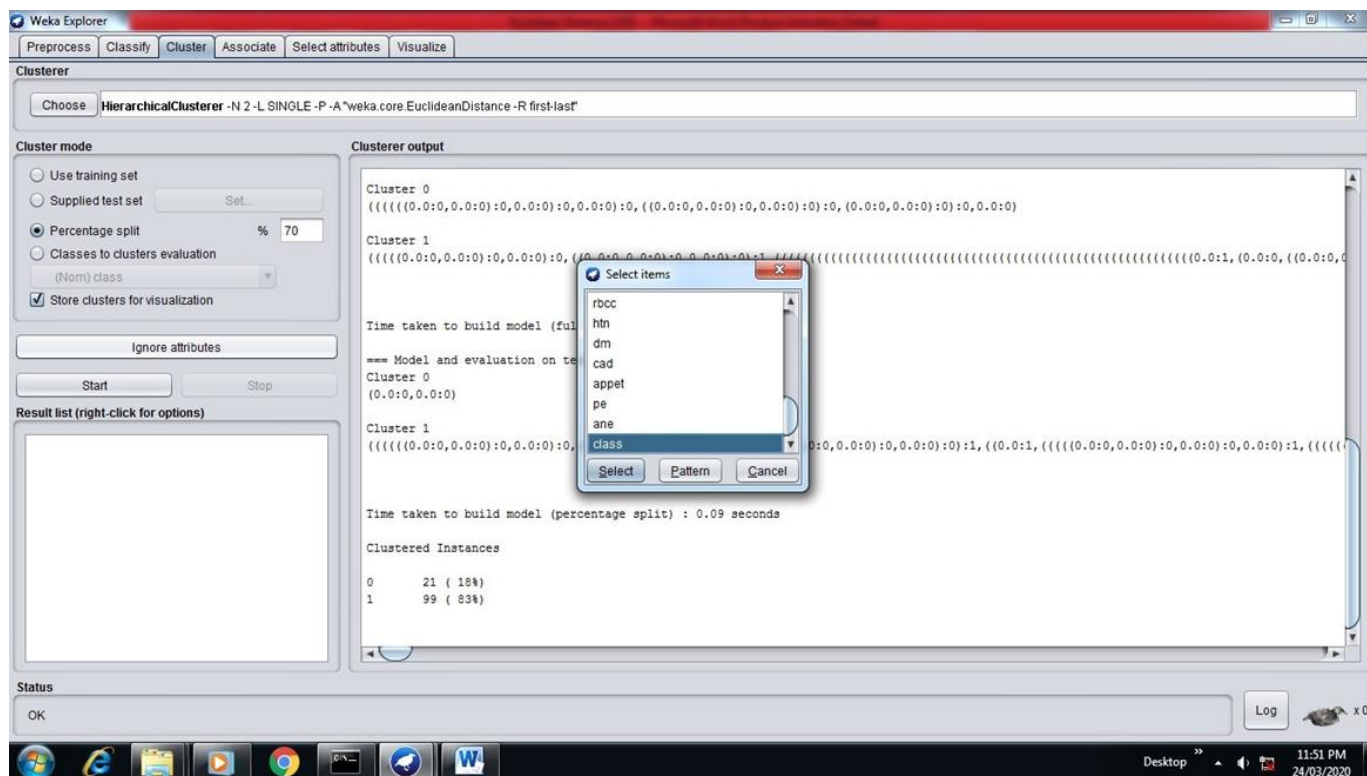


Figure: 30No of 02 cluster of hierarchical clustering class clustering output no of cluster 2,3,4& 5 STEP 1:

(Euclidean Distance) = 2 Cluster = Full Attributes

==== Run information ====

Scheme: weka.clusterers.HierarchicalClusterer -N 2 -L SINGLE -P -A "weka.core.EuclideanDistance -R first- last"

Relation: Chronic_Kidney_Disease_(RS Walse)-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst- last

Instances: 400

Attributes: 25 age

bp sg al su rbc pc pcc ba bgr bu sc sod pot

hemo pcv wbcc rbcc htn dm cad appet pe ane

Ignored:

class

Test mode: Classes to clusters evaluation on training data

==== Clustering model (full training set) ==== Cluster 0

Time taken to build model (full training data) : 0.73 seconds

==== Model and evaluation on training set ==== Clustered Instances

0 399 (100%)

1 1 (0%)

Class attribute: class Classes to Clusters:

0 1 <-- assigned to cluster

```

249 1 | ckd
150 0 | notckd
Cluster 0 <-- ckd Cluster 1 <-- No class
Incorrectly clustered instances :      151.0  37.75 %
STEP 2:
(Euclidean Distance) = 3 Cluster = Full Attributes
Time taken to build model (full training data) : 0.73 seconds
=== Model and evaluation on training set === Clustered Instances
0      398 (100%)
1       1 ( 0%)
2       1 ( 0%)

```

Class attribute: class Classes to Clusters:

```

0 1 2 <-- assigned to cluster
248 1 1 | ckd
150 0 0 | notckd

Cluster 0 <-- ckd Cluster 1 <-- No class Cluster 2 <-- No class

Incorrectly clustered instances :      152.0  38      %
STEP 3:

```

```

(Euclidean Distance) = 4 Cluster = Full Attributes

Time taken to build model (full training data) : 0.7 seconds

=== Model and evaluation on training set === Clustered Instances
0      397 ( 99%)
1       1 ( 0%)
2       1 ( 0%)
3       1 ( 0%)

```

Class attribute: class Classes to Clusters:

```

0 1 2 3 <-- assigned to cluster
247 1 1 1 | ckd
150 0 0 0 | notckd

Cluster 0 <-- ckd Cluster 1 <-- No class

Cluster 2 <-- No class Cluster 3 <-- No class

Incorrectly clustered instances :      153.0  38.25 %
STEP 4:

```

(Euclidean Distance) = 5 Cluster = Full Attributes

Time taken to build model (full training data) : 0.69 seconds

=== Model and evaluation on training set === Clustered Instances

```
0      396 ( 99%)
1       1 (  0%)
2       1 (  0%)
3       1 (  0%)
4       1 (  0%)
```

Class attribute: class Classes to Clusters:

```
0 1 2 3 4      <-- assigned to cluster
246 1 1 1      1 | ckd
150 0 0 0      0 | notckd
```

Cluster 0 <-- ckd Cluster 1 <-- No class Cluster 2 <-- No class Cluster 3 <-- No class Cluster 4 <-- No class

Incorrectly clustered instances : 154.0 38.5 %

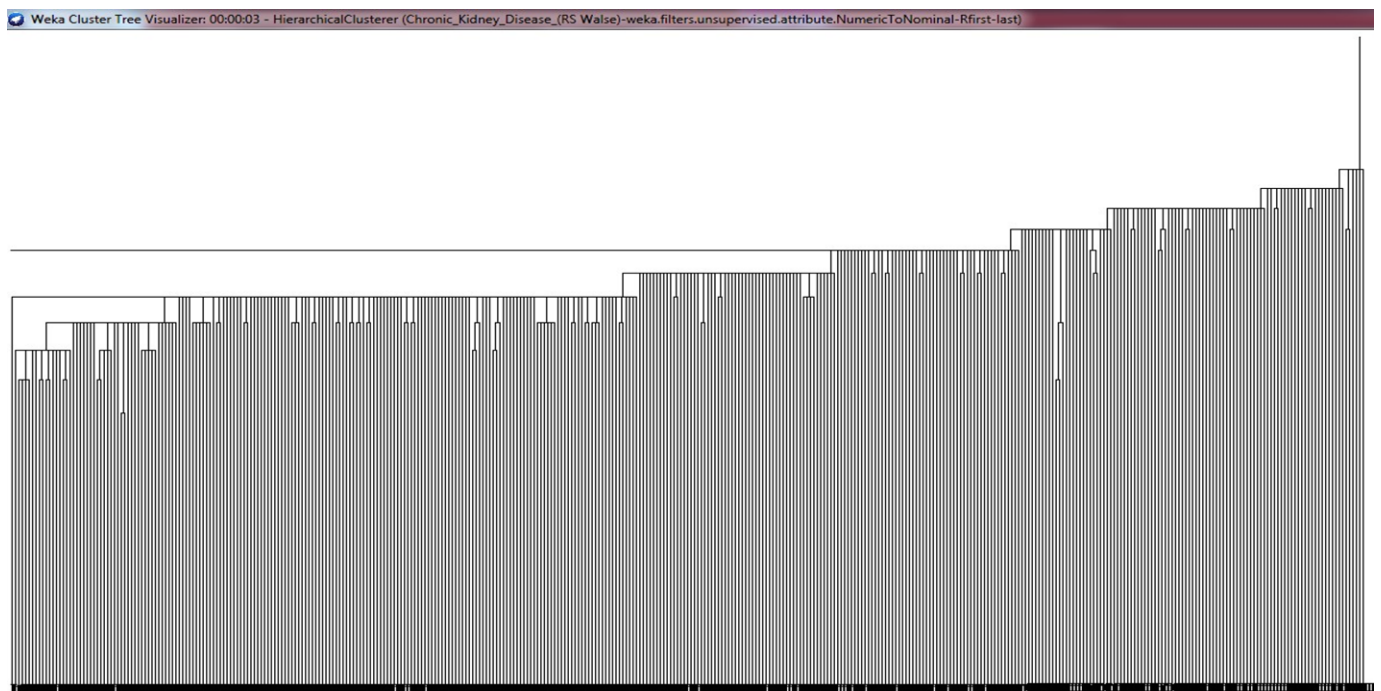


Figure: 31 Hierarchical clustering with Manhattan distance Function of Dendrogram

Figure 3: Performance accuracy by Class and Confusion matrix by decision tree.

Step 2: Using Manhattan Distance function No of Clusters 2,3,4& 5

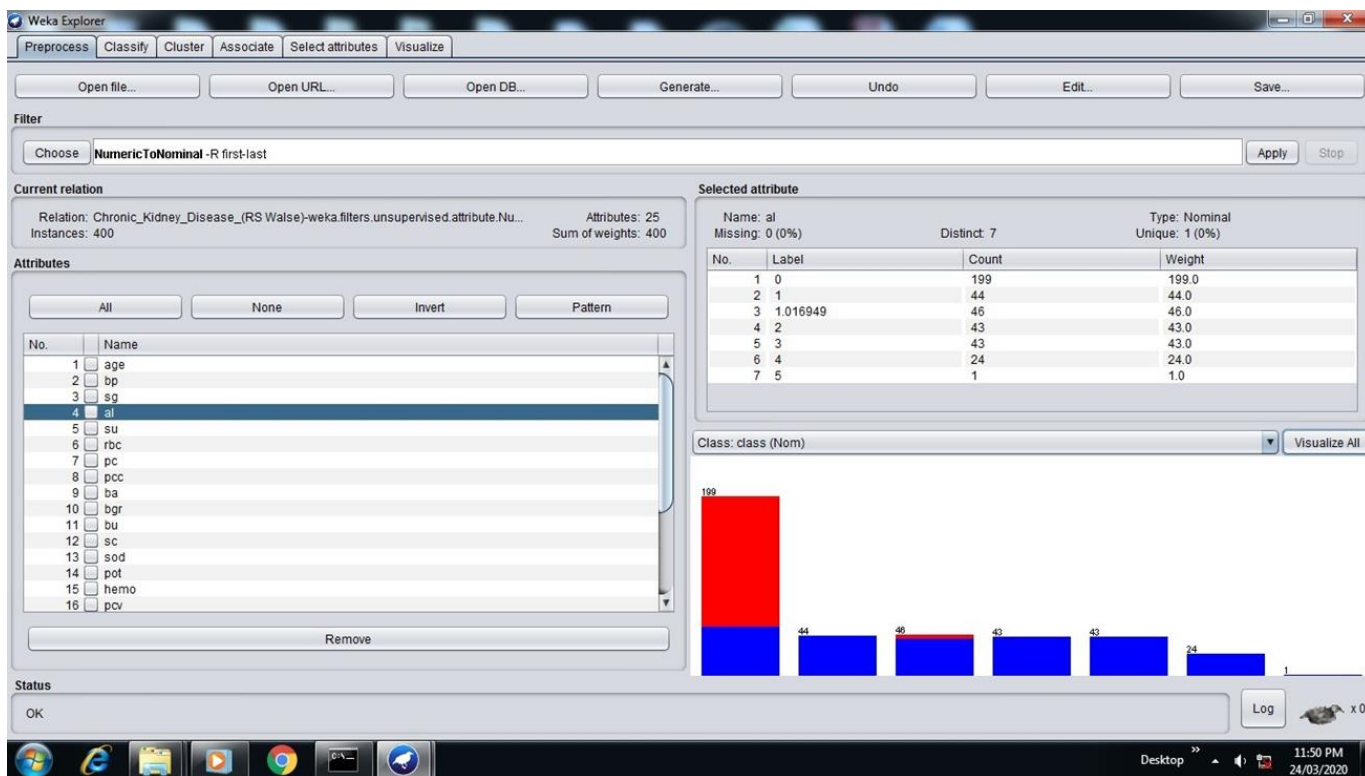


Figure: 32Preprocess data set with full attributes for Manhattan Distance function

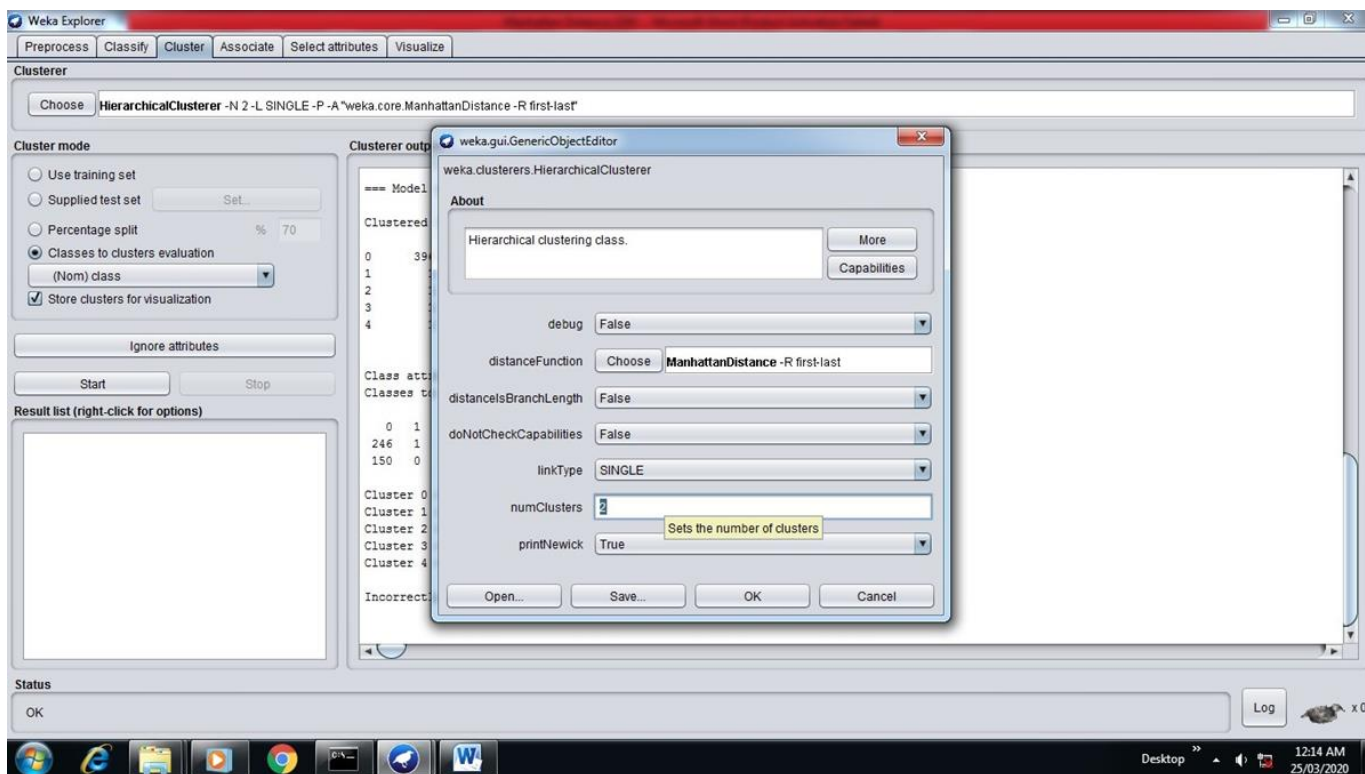


Figure: 33Hierarchical Clustering class with Manhattan Distance function for no of cluster 2,3,4& 5

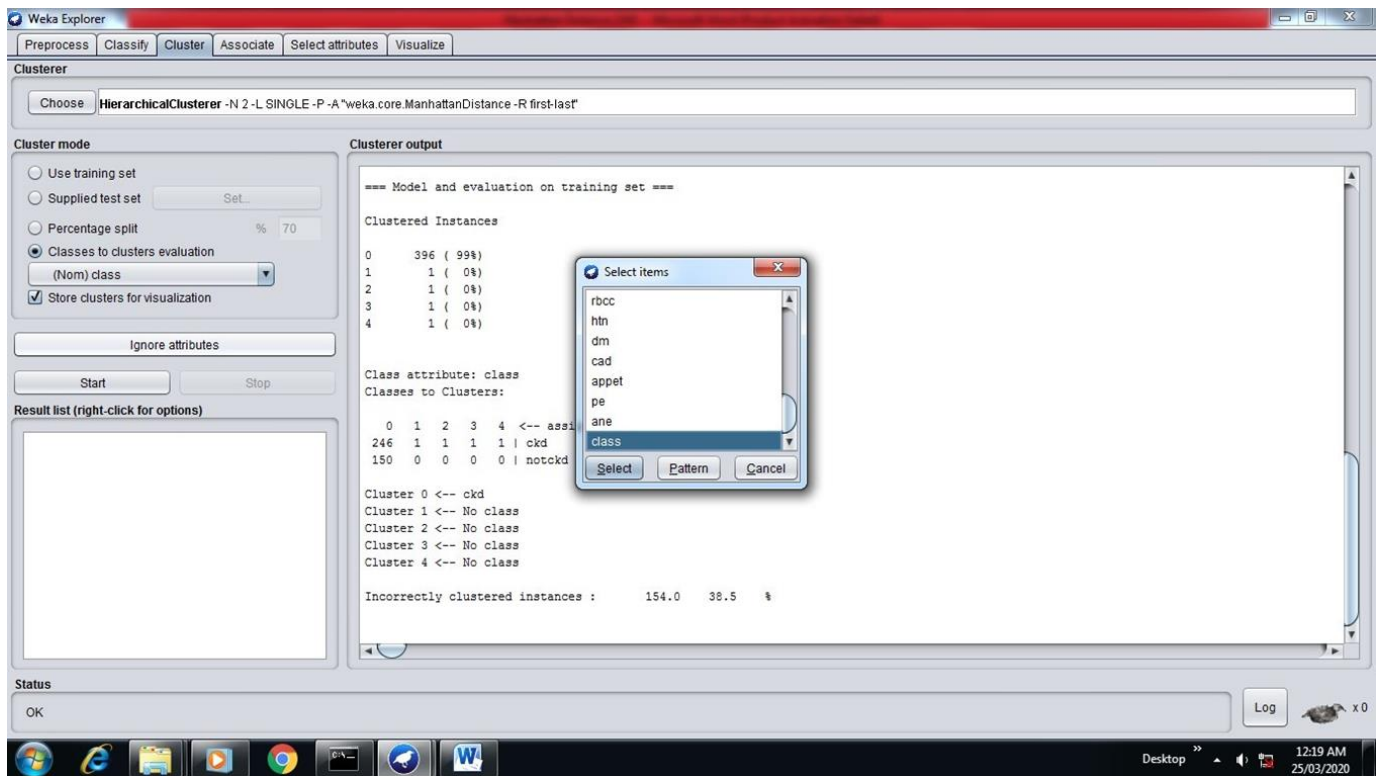


Figure: 34 Result of Hierarchical Clustering class with Manhattan Distance function for no of cluster 2,3,4& 5

STEP 1:

(Manhattan Distance) = 2 Cluster = Full Attributes

=== Run information ===

Scheme: weka.clusterers.HierarchicalClusterer -N 2 -L SINGLE -P -A "weka.core.ManhattanDistance -R first- last"

Relation: Chronic_Kidney_Disease_(RS Walse)-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst- last

Instances: 400

Attributes: 25

Time taken to build model (full training data) : 0.72 seconds

=== Model and evaluation on training set === Clustered Instances

0 399 (100%)

1 1 (0%)

Class attribute: class Classes to Clusters:

0 1 <-- assigned to cluster

249 1 | ckd

150 0 | notckd

Cluster 0 <-- ckd Cluster 1 <-- No class

Incorrectly clustered instances : 151.0 37.75 %

STEP 2:

(Manhattan Distance) = 3 Cluster = Full Attributes

Time taken to build model (full training data) : 0.71 seconds

=== Model and evaluation on training set === Clustered Instances

0 398 (100%)

1 1 (0%)

2 1 (0%)

Class attribute: class Classes to Clusters:

0 1 2 <-- assigned to cluster

248 1 1 | ckd

150 0 0 | notckd

Cluster 0 <-- ckd Cluster 1 <-- No class Cluster 2 <-- No class

Incorrectly clustered instances : 152.0 38 %

STEP 3:

(Manhattan Distance) = 4 Cluster = Full Attributes

Time taken to build model (full training data) : 0.73 seconds

=== Model and evaluation on training set === Clustered Instances

0 397 (99%)

1 1 (0%)

2 1 (0%)

3 1 (0%)

Class attribute: class Classes to Clusters:

0 1 2 3 <-- assigned to cluster

247 1 1 1 | ckd

150 0 0 0 | notckd

Cluster 0 <-- ckd Cluster 1 <-- No class Cluster 2 <-- No class Cluster 3 <-- No class

STEP 4:

(Manhattan Distance) = 5 Cluster = Full Attributes

=== Model and evaluation on training set === Clustered Instances

0 396 (99%)

1 1 (0%)

2 1 (0%)

3 1 (0%)

4 1 (0%)

Class attribute: class Classes to Clusters:

0 1 2 3 4 <-- assigned to cluster 246 1 1 1 1 | ckd

150 0 0 0 0 | notckd

Cluster 0 <-- ckd Cluster 1 <-- No class Cluster 2 <-- No class Cluster 3 <-- No class Cluster 4 <-- No class

Incorrectly clustered instances : 154.0 38.5 %

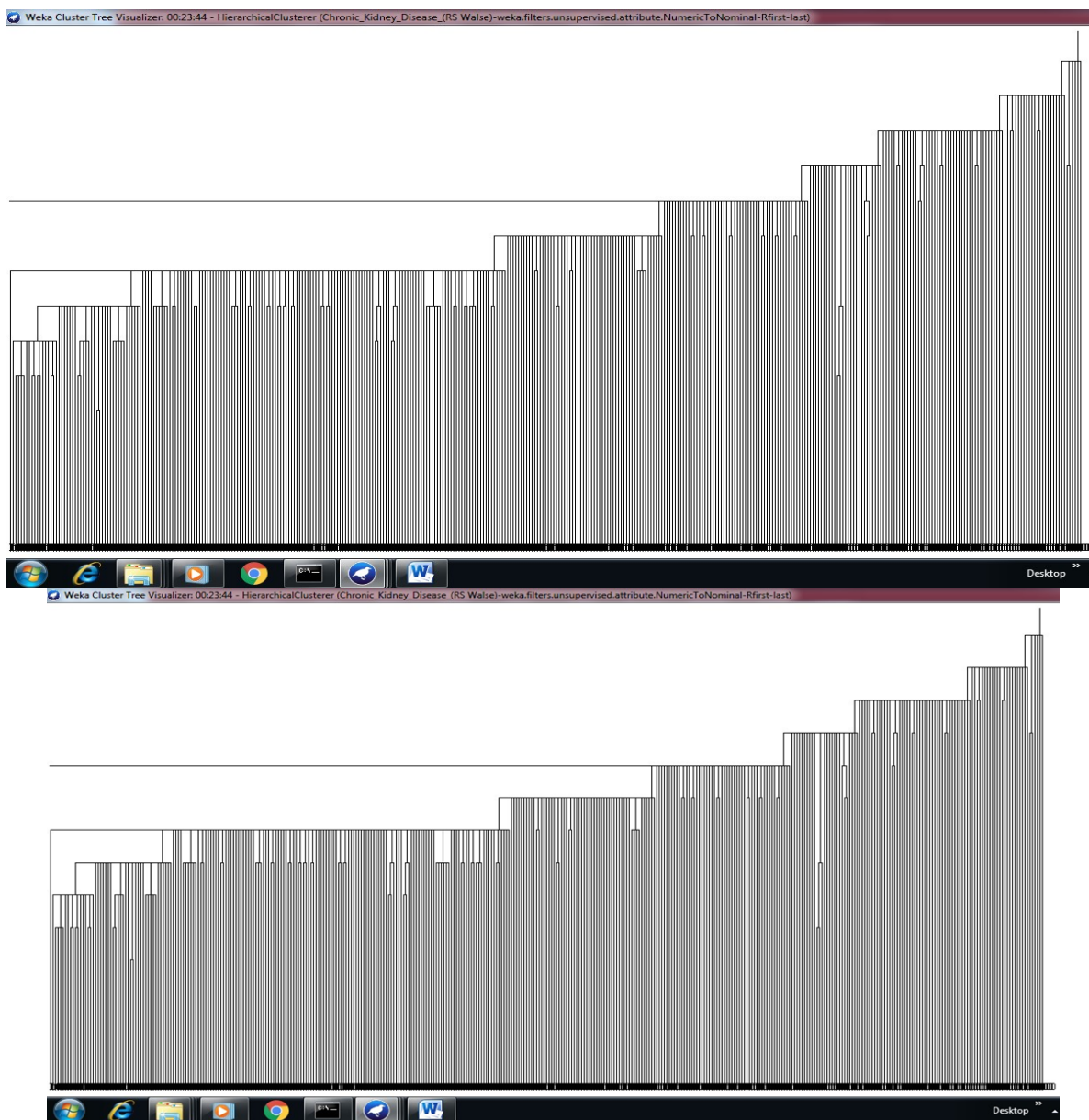


Figure: 35 Result of Hierarchical Clustering class with Dendrograph Figure 35 What shows ?

IV. RESULTS AND DISCUSSION

{a}Comparative statement of Supervised SMO-SVM- Classifier with one, two & full attributes with 70% split using Calibrator: Logistic and Kernel= Polynomial function

The research, the kidney dataset processed with different attributes (25), which contains 400 rows, i.e., instances and 25 attributes, means columns. The researcher has select every attributes to displays type of attributes, the type means Nominal, how many missing values present in the data set for each attribute viz. instances, how many distinct values are present in the dataset, distinct means different values, if we select attribute is shown Nom- in front of attribute- nom means nominal type. The numeric data gives summary of the overall data in the form of descriptive statistics. Also, if data is qualitative then it treated as an attribute class and it shows in the form of label count and its weight in the form of true/false or yes/no.

Table 10: Correctness by Class values

Table 10: Correctness by Class values

No of Attributes	Class	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area
*CVF=4	01 CKD	0.588	0.000	1.000	0.588	0.741	0.590	0.794	0.845
	(htnVs Not-CKD	1.000	0.412	0.593	1.000	0.744	0.590	0.794	0.593
	Class) Weight Avg	0.743	0.155	0.847	0.743	0.742	0.590	0.794	0.751
**CVF	02 CKD	0.737	0.000	1.000	0.737	0.848	0.712	0.868	0.904
	appetV Class) Weight Avg	0.833	0.096	0.886	0.833	0.836	0.172	0.868	0.824
24	CKD	0.980	0.000	1.000	0.980	0.990	0.974	0.990	0.992

- * Found max and same accuracy for 5 to 15 CVF
- ** Found max and same accuracy for 5 to 15 CVF
- *** 5,6,8,9,10,14,15 found max and same accuracy CVF

Table 10 shows the correctness by Class values Supervised- classify – SMO- SVM- with attributes 2,3 & Full Attributes – Applying Cross validation fold from 4 to 15 with 70% split, and applying the Calibrator: Logistic and Kernel: Polynomial function and it shows the comparative study of the result and found the accurate prediction value, therefore research perfectly state that the increasing no of cross validation folds with respective increasing attributes the accuracy of the correctness of the class values increases shows in the form of ROC Area CVF=4 ROC Area = 0.794 using one attribute , CVF=4 ROC Area = 0.868 using two attributes and CVF=5 ROC Area=0.990. Hence we found best and Novel Predictive Module with the help of SMO- SVM- Classifier Class Module on the basis of Calibrator and Kernel function.

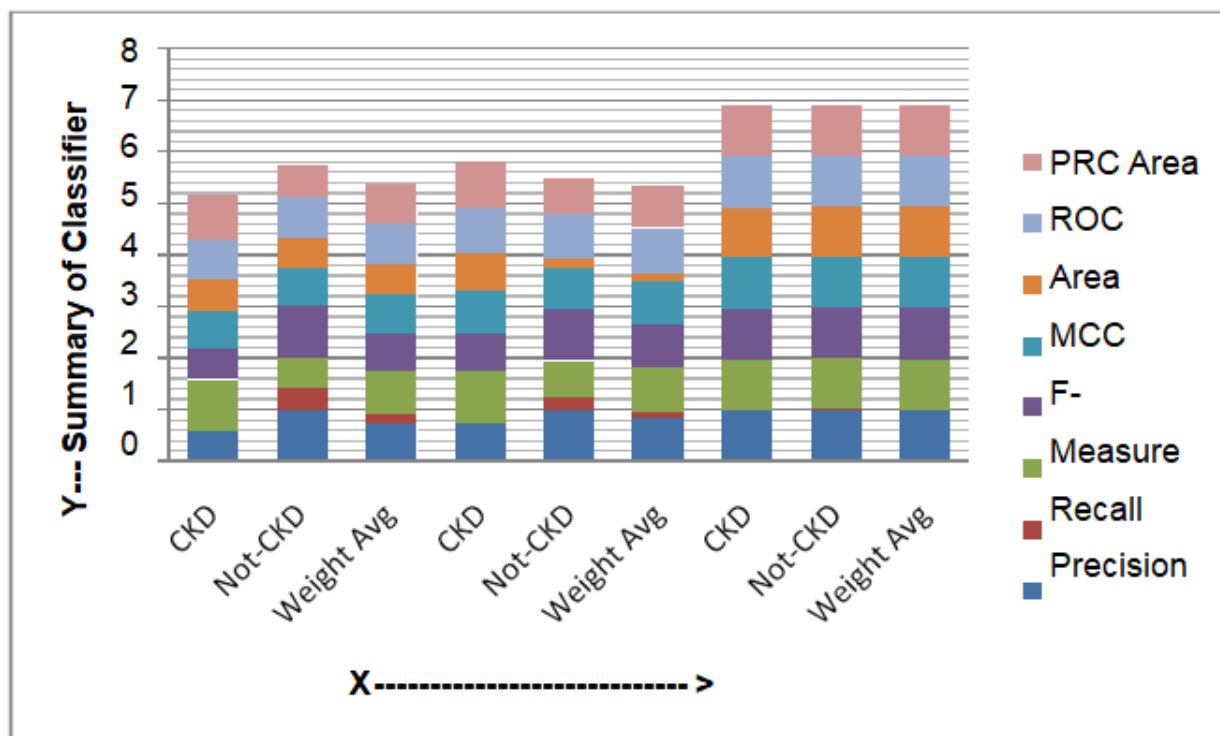


Figure: 36 Comparative study of Correctness by class values of SMO-SVM- Classifier with 1,2 and full attributes with 70% and CVF from 4 to 15.

Table 11: Summary of Classifier model (Train set data)

Sr. No.	Particulars	01	02	24
		(htnVs Class) CV Folds 4 Found max and same accuracy for 5 to 15 CVF	(htn, appetVs Class) CV Folds 4 Found max and same accuracy for 5 to 15 CVF	(Full Vs Class) CV Folds 5,6,8,9,10,14,15 found max and same accuracy CVF
1	Test mode: 70% train, 30% test	70.0% train	70.0% train	70.0% train
2	Correctly Classified Instances	74.25	83.33	98.75
3	Incorrectly Classified Instances	25.75	16.66	1.25
4	Kappa statistic	0.517	0.6725	0.9735
5	Mean absolute error	0.2575	0.1667	0.0125
6	Root mean squared error	0.5074	0.4082	0.1118
7	Relative absolute error	54.90	35.6241	2.6656
8	Root relative squared error	104.81	84.6877	23.094
9	Total Number of Instances	400	400	400

(CVF= Cross Validation Folds)

Table 11 shows the Summary of classifier model with train set data, of SMO- classifier by applying one, two and full 25 attributes to test the result and getting some of the useful and accurate result, so as to use our research to further researcher for further prediction of data and effectively useful of the SMO – SVM classifier algorithm, further table 11 shows the how accuracy will be increased on the basis of different parameters like Cross Validation Folds just testing from 4 to 15, accuracy of the result is higher as compared to the earlier result in the form of Correctly Classifier Instances is 74.25 % , 83.33% and 98.75% respectively. Therefore also found the best Novel Summary of Classified module with highest accuracy of CCI.

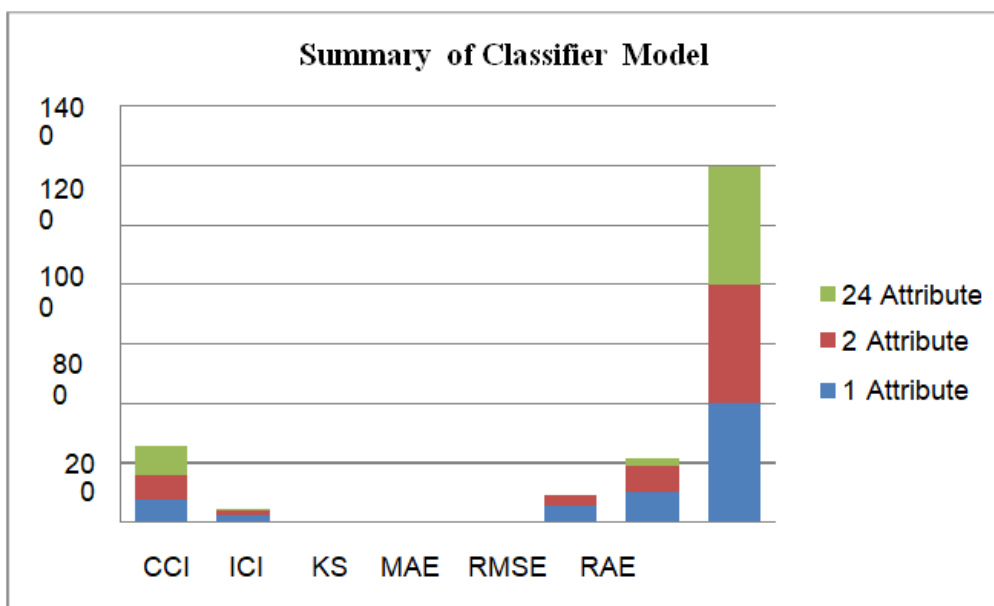


Figure: 37Comparative study of Summary of Classifier Module of SMO-SVM- Classifier with 1,2 and full attributes with 70% and CVF from 4 to 15.

Table 12:Comparative statement of Confusion Matrix SMO-SVM- Classifier with one, two & full attributes with 70% split using Calibrator: Logistic and Kernel= Polynomial function

Sr. No.	No of Attributes	CVF	Predicted (a)	Predicted (b)	< - Classified as
1	One Attribute htnvs Class	4-15	147	103	a = ckd
			0	150	b = not-ckd
2	Two Attributes Htn, appetVs Class	4	147	103	a = ckd
			0	150	b = not-ckd
		5-15	172	78	a = ckd
			0	150	b = not-ckd
3	Full Attributes 24 Vs Class	4,7,11,12,13	244	6	a = ckd
			0	150	b = not-ckd
		5,6,8,9,10,14,15	245	5	a = ckd
			0	150	b = not-ckd

(CVF = Cross Validation Folds)

Table 12 shows the Comparative statement of Confusion Matrix SMO-SVM- Classifier with one, two & full attributes with 70% split using Calibrator: Logistic and Kernel= Polynomial function, it plot the threshold Curve and Cost benefit curve for class ckd and Not-ckd in the form of % population of confusion matrix and cost matrix. Confusion matrix of predicted a & b also cost matrix a & b with the random and gain value. The researcher also find the research output is using full length of attributes it creates minimum Confusion matrix with accurate predicted class value with minimum no of iterations of clusters.

{b}Comparative statement of Un-Supervised Hierarchical Clustering algorithm using Euclidean and Manhattan distance function for the no of cluster 2,3,4 & 5.

Comparative statement of the Hierarchical clustering algorithm using Euclidean nad Manhattan distance function result with accuracy.

□ Euclidean Distance:

=== Run information ===

Scheme: weka.clusterers.HierarchicalClusterer -N 2 -L SINGLE -P -A "weka.core.EuclideanDistance -R first-last"
 Relation:Chronic_Kidney_Disease_(RS Walse)-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last

□ Manhattan Distance:

=== Run information ===

Scheme:weka.clusterers.HierarchicalClusterer -N 2 -L SINGLE -P -A "weka.core.ManhattanDistance -R first-last"
 Relation:Chronic_Kidney_Disease_(RS Walse)-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last
 === Clustering model (full training set) ===

Table 13:Comparative statement of Unsupervised Hierarchical Clustering algorithm using Euclidean and Manhattan Distance Function of full attributes with 70% split. (Cluster 2,3,4& 5)

Sr. No.	No. of Cluster	Euclidean Distance		Manhattan Distance			
		Clustered Instances	Classes to Clusters	Clustered Instances	Classes to Clusters		
1	2	0	399(100)%	249 1 ckd	0	399(100)%	249 1 ckd
		1	1 (0%)	150 0 notckd	1	1 (0%)	150 0 notckd
2	3	0	398	248 1 1 ckd	0	398	248 1 1 ckd
		1	(100%)	150 0 0 notckd	1	(100%)	150 0 0 notckd
		2	1 (0%)		2	1 (0%)	
		1	1 (0%)		1	1 (0%)	
		0	397 (99%)		0	397 (99%)	
3	4	1	1 (0%)	247 1 1 1 ckd	1	1 (0%)	247 1 1 1 ckd
		2	1 (0%)	150 0 0 0 notckd	2	1 (0%)	150 0 0 0 notckd
		3	1 (0%)		3	1 (0%)	
		0	396 (99%)		0	396 (99%)	
4	5	1	1 (0%)	246 1 1 1 1 ckd	1	1 (0%)	246 1 1 1 1 ckd
		2	1 (0%)	150 0 0 0 0 notckd	2	1 (0%)	150 0 0 0 0 notckd
		3	1 (0%)		3	1 (0%)	
		4	1 (0%)		4	1 (0%)	

Table 13 describes the researcher also compare the result by using Unsupervised Hierarchical Clustering algorithm of Chronic Kidney Disease data with the support of Euclidean and Manhattan Distance clustering function with the help of 70% Split by setting the properties No of Clusters from 2,3,4, & 5 of Euclidean and Manhattan distance function. And finally research found the best and accurate prediction is the, Increasing the Number of Clusters of full attributes of Chronic Kidney Disease data accuracy of creating no of clusters as well as classes to Clusters increases. One more main comparative results also found, both the Euclidean and Manhattan distance function output result is also same for No of clusters 2,3,4& 5.

V. CONCLUSION

The CKD data is analyzed and predicted for diagnosed patients using data mining supervised classifiers and unsupervised clustering algorithm of SMO – SVM algorithms and Hierarchical algorithms respectively. The performance of these algorithms is compared using Weka tools. The final obtained result shows that the both the Classification and Clustering algorithm Novel discovered module is the best truthful classifier with 98.75% accuracy. i.e. Correctly Classified Instances as compared to earlier results by applying one, two attributes by applying Calibrator: Logistic and Kernel: Polynomial function for CV folds is 5,6,8,9,10,14,15, the result is also same for confusion matrix also by using same CV folds. Therefore, found the best model prediction for Supervised- SMO in WEKA on the basis of three test using calibrator of logistic and kernel using polynomial function by applying Cross Validation Folds from 4 to 15 (Train on a portion of the data and test on the remainder) with 70% split, and final prediction is, increasing the no of attributes the accuracy of Correctly Classified Instances (CCI), ROC Area value and Confusion matrix value increased. Similarly research is also found to confirm predict the result of Unsupervised Hierarchical Clustering algorithm by applying full attributes also confirm prediction is increasing the number of Clusters from 2,3,4 and 5 for both Euclidean and Manhattan Function, the accuracy of result in terms of better and accurate clusters found. The adopted methodology clears the process of practical. For research work, some of the attributes were measured RBC count, HP, Diabetes Mellitus, CAD, Appetite, Pedal Edema, Anemia, etc. Now future, this kind of research which will be helpful to the doctors or medical industry for prediction of CKD and not CKD patient based on their other health parameters, to minimize the growth rate of CKD patients and to control further damages of the kidney. Data mining plays an active role in predicting future kidney-related health problems. In this paper, these algorithms it has been analyzed. We have tried to analyze chronic kidney disease by using SMO- SVM classifier algorithm unprocessed learning technique with Calibrator: Logistic and Kernel: Polynomial function. Indeed, the purpose of our research is to use our research to analyze kidney disease, or whether it can cause kidney disease in the future, this will allow kidney patients who are currently in a state to know what caused the kidney disease, and those who have not had a kidney disease will see if they can develop kidney disease in the future, so they will not need to perform additional tests and save money.

VI. ACKNOWLEDGEMENT

Authors are grateful to the UCI ML repository is provided that all types of necessary database WEKA for providing such a reliable tool to extract and analyze knowledge from the database.

VII. REFERENCES

- [1]. Arasu, S. Dilli, and R. Thirumalaiselvi. "A novel imputation method for effective prediction of coronary Kidney disease." In 2017 2nd International Conference on Computing and Communications Technologies (ICCT), pp. 127-136. IEEE, 2017.
- [2]. Ariff, M. H., I. Ismarani, and N. Shamsuddin. "RFID based systematic livestock health management system." In 2014 IEEE Conference on Systems, Process and Control (ICSPC 2014), pp. 111-116. IEEE, 2014.
- [3]. Bharara, Sanyam, A. SaiSabitha, and AbhayBansal. "A review on knowledge extraction for business operations using data mining." In 2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence, pp. 512-518. IEEE, 2017.
- [4]. Chuan, Zou, Tang Ying, Bai Li, ZengYuqun, and Lu Fuhua. "Application of clustering analysis to explore syndrome evolution law of peritoneal dialysis patients." In 2013 IEEE International Conference on Bioinformatics and Biomedicine, pp. 23-26. IEEE, 2013.
- [5]. Due ThanhAnhLuong, Dept of the computer. Sci&Engi, uni, at Buffalo Buffalo, NY, USAA K- Means Approach to Clustering disease Progressions IEEE Keywords, Sep 2017
- [6]. Güllüoğlu, SabriSerkan. "Segmenting customers with data mining techniques." In 2015 Third International Conference on Digital Information, Networking, and Wireless Communications (DINWC), pp. 154-159. IEEE, 2015.
- [7]. Jinyin, Chen, et al. "A novel cluster center fast determination clustering algorithm." *Applied Soft Computing* 57 (2017): 539-555.
- [8]. Khanna, Umesh. "The economics of dialysis in India." *Indian journal of nephrology* 19, no. 1 (2009): 1.
- [9]. Kunwar, Veenita, et al. "Chronic Kidney Disease analysis using data mining classification techniques." 2016 6th International Conference-Cloud System and Big Data Engineering (Confluence). IEEE, 2016.
- [10]. Narander Kumar, SabitaKLhatri, Department of Computer, 3rd IEEE International Conference on Computational Intelligence and Communication Technology (IEEE-CICT 2017) "Implementing WEKA for medical data classification and early disease prediction 978-1-50, 2017
- [11]. Uboltham, Issariya, NakornthipPrompoon, and Wirichada Pan-Ngum. "AKIHelper: Acute kidney injury diagnostic tool using KDIGO guideline approach." In 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), pp. 1-6. IEEE, 2016.
- [12]. Khanna, Umesh. "The economics of dialysis in India." *Indian journal of nephrology* 19, no. 1 (2009): 1.
- [13]. Varma, P. P. "Prevalence of chronic kidney disease in India-Where are we heading?." *Indian journal of nephrology* 25, no. 3 (2015): 133.
- [14]. https://archive.ics.uci.edu/ml/datasets/Chronic_Kidney_Disease