Artificial intelligence solution implementation

Introduction

Appropriate decision-making whether to provide or not to provide loans to any individuals or business based on their personal data and credit history is very crucial to trade-off between profit and risk. Using artificial intelligence (AI) with machine learning (ML) models, the credit decision-making can be automated with excellent prediction accuracy in relatively less time compared to manual process. This report will demonstrate the implementation of AI for credit decision-making with an example using the open source application Waikato Environment for Knowledge Analysis (WEKA) version 3.8.6 developed by the University of Waikato, Hamilton, New Zealand.

Methodology

This demonstration of the ML project on credit decision-making will follow CRISP-DM methodology, which stands for Cross-Industry Standard Process for Data Mining (Frank et al., 2016).

Business understanding

The main aim in credit decisions is to curtail the risk involved in lending to parties with bad credit history. There are number of factors involved to analyse and classify the loan application as bad, which involves an in-depth review of the individual or business ability to repay the loan, evaluating their personal information, financial statements, employment and residential status, etc (CFI, 2022). In addition, the classification as bad have to have a proper justification and evidence that the results are unbiased.

Data understanding

In this report, the German credit open source dataset available from UCI Machine Learning Repository has been used to demonstrate AI implementation with WEKA. The link to the dataset is provided in the references section. This dataset has 20 attributes based on which about 1000 debtors are classified as good or bad, described in the below table.

Table 1: Attributes and description of credit dataset

Attribute	Description
checking_status	Checking the status of existing account of the applicant, in 4 categories (less than 0, between 0 and 200, more than 200, no checking)
duration	Duration of loan, in months
credit_history	Credit history of credit taken, in 5 categories (no credits, all paid, existing paid, delayed previously, critical/other existing credit)
purpose	Purpose of the credit
credit_amount	Credit amount requested by the applicant, numerical
savings_status	Status of savings account, in 5 categories (<100, between 100 and 500, between 500 to 1000, more than 1000, unknown)
Employment	Present employment duration, in 5 categories (unemployed, less than 1 year, between 1 and 4 years, between 4 and 7 years, more than 7 years)
installment_commitment	Installment rate (1,2,3,4)
personal_status	Personal status (married, single, divorced) and sex (male, female)
other_parties	Other debtors (Co-applicant, guarantor, none)
residence_since	Number of years in current residence (1,2,3,4)
property_magnitude	Property (real estate, life insurance, car, none)
age	Age in years

other_payment_plans	Other installment plans (banks, stores, none)
housing	Housing (rent, own, free)
existing_credits	Number of existing credits at applied bank (1,2,3,4)
job	Type of job (unskilled/skilled/highly qualified)
num_dependents	Number of persons dependent financially on the applicant
own_telephone	Telephone (yes,no)
foreign_worker	Foreign worker (yes,no)
class	Classifier (good/bad)

Data Preparation

Before ML modelling, data were checked for correct format that can be interpreted by the model, checked for any missing entries, each attribute checked for its variability and type whether categorical or numerical or binary. The data could be viewed or transformed in the data-preprocessing tab in WEKA application. WEKA files are required to be in attribute relation file format (arff). List view and graphical display of each attribute is presented in figure 1 and 2 respectively in appendix.

Feature selection: Selecting a small set of attributes from the given 20 attributes could possibly improve the quality of results. Feature selection can be done manually based on domain knowledge or automatically based on application computational methods such as the search and filter, the wrapper method, the embedded and the dimensionality reduction methods or a combination of both manual and automatic methods. In this report, three cases are presented: 1) Using all 20 attributes, 2) Manual selection and 3) Automatic selection. For manual selection, based on the knowledge, three attributes such as checking status, employment and existing credits were selected for classification, while for the automatic selection, the filter

‘Attribute selection’ was used which uses supervised search and filter method, filtered out three attributes such as checking status, duration and credit history. The manual and automatic selections of attributes in WEKA workspace are shown in figure 9 and 16, respectively.

Modelling

Using the above data, we can build a ML model using WEKA to predict the credit applications. The prediction model uses ML, statistics to learn the dependency of each feature to the outcome of the decision, predicts, and classifies as good or bad based on the learning. Xia et al. (2017) had investigated ML algorithms and found out that no single model works well in all scenarios. According to the research of Lessmann et al. (2015), the ensemble method was better than a single artificial intelligence and statistical method. As credit decision-making is a classification problem, decision tree method, J48 classifier was used, which is a C 4.5 algorithm able to handle both categorical and numerical data. For comparison, ensemble Random forest classifier and Support Vector Machine (SVM) with Sequential Minimal Optimization (SMO) algorithm were used.

Testing was carried out using two validation methods such as cross-validation with 10 folds and percentage split method with 90% training to 10% testing to investigate the difference between the two. In this demonstration, would use evaluation metrics to point out the difference in results with different classifiers.

Evaluation of model output

Confusion matrix is an important evaluation tool and is used to derive other performance metrics such as precision, recall, sensitivity, specificity and accuracy

(Raschka, 2018). In this report, the confusion matrix, precision and F-measure were used to evaluate different classifiers accuracy. The confusion matrix contains the following terms used to describe the performance of any classification model.

True positives (TP): Instances truly classified as good creditors.

True negatives (TN): Instances truly classified as bad creditors

False positives (FP): Instances falsely predicted as good creditors by the classifier, which are actually bad creditors.

False negatives (FN): Instances falsely predicted as bad creditors by the classifier, which are actually good creditors.

For a classifier to have good prediction accuracy, TP and TN have to have bigger values than FP and FN.

Precision is given by the ratio of TP to (TP+FP). Recall is given by the ratio TP to (TP+FN). The F-measure is the harmonic mean of precision and recall, used when comparing different models. For a classifier to be evaluated as ideal, would require a value of 1.000.

Table 2 shows the output of the three classifiers, J48, Random forest and SMO using all 20 attributes, manual and automatic feature selection methods and with cross-validation and percentage split methods of evaluation. It is clear from table 2 that the cross-validation method of evaluation works better compared to percentage split method, given that with 10% testing data, the number of instances for testing is only 100 for a total of 1000 instances. Although using percentage split method of testing, the SMO classifier produced 77% correctly classified instances with automatic feature selection method, its accuracy could not be acknowledged as limited number of instances using for testing (100 instances) for this dataset. It is also clear that using all 20 attributes gives better accuracy than using manual or automatic feature selection and the ensemble Random forest classifier has best prediction accuracy with 76.4% of the instances correctly classified, compared to other two classifiers J48 and SMO, for the given dataset. The screenshot of all the classifiers output in WEKA workspace are shown in figures in appendix.

Table 2: Output from WEKA workspace

Classifier	Feature selection	Testing	Classification (%)		Confusion matrix				Precision	F measure
			Correct	Incorrect	TP	FN	FP	TN		
Decision tree J48	All 20	Cross validation	70.5	29.5	588	112	183	117	0.687	0.692
		Percentage split	74	26	63	11	15	11	0.728	0.733
	Manual	Cross validation	69.1	30.9	677	23	286	14	0.606	0.595
		Percentage split	69	31	61	13	18	8	0.67	0.679
	Automatic	Cross validation	70.5	29.5	615	85	210	90	0.676	0.678
		Percentage split	74	26	58	16	10	16	0.761	0.748
Random forest	All 20	Cross validation	76.4	23.6	642	58	178	122	0.751	0.744
		Percentage split	76	24	63	11	13	13	0.754	0.757
	Manual	Cross validation	68.8	31.2	619	81	231	69	0.648	0.651
		Percentage split	70	30	64	10	20	6	0.661	0.674
	Automatic	Cross validation	70.2	29.8	577	123	175	125	0.688	0.693
		Percentage split	74	26	61	13	13	13	0.740	0.740
SVM, SMO	All 20	Cross validation	75.1	24.9	610	90	159	141	0.728	0.741
		Percentage split	75	25	59	15	10	16	0.767	0.757
	Manual	Cross validation	67.6	32.4	644	56	268	32	0.603	0.609
		Percentage split	68	32	57	17	15	11	0.688	0.684
	Automatic	Cross validation	71.7	28.3	664	36	247	53	0.689	0.659
		Percentage split	77	23	70	4	19	7	0.747	0.734

Deployment

The dataset required for the credit decision-making business problem would be similar to the dataset presented in this report, alike the 20 attributes providing applicant's personal information and credit history information. Hence, the same methodology could be carried out to implement ML methods for the business problem, credit decision-making. With appropriate data pre-processing, feature engineering methods and evaluation metrics, optimal ML algorithm can be chosen for the given problem.

Real-life problems

The common real-life data problems encountered are data imbalance, data sparseness, and missing data. In the training set the number of instances for good and bad are not always balanced, hence the classifier may find it difficult to learn from the sparse data for bad credit. Also missing data is inevitable and has significant impact on the output accuracy. Liu & Zhang (2021) had proposed a set of solutions from the perspectives of data imbalance processing and data preprocessing for classification models. For data imbalance, the K-means and SMOTE(Synthetic Minority Oversampling Technique) could be used to generate minority samples. By data preprocessing with dimensionality reduction algorithms and data missing processing and using a combination of GBDT2NN and Factorization Machines to integrate high-level feature combinations and low-level feature combinations could differentiate between good and bad creditors.

Conclusion

In conclusion, the feasibility of implementing AI for the identified business problem has been demonstrated with similar example from open source dataset with agreeable accuracy, using WEKA. Feature selection methods, choosing the right classifier and testing method were demonstrated for a given business problem, utilising the evaluation methods available. Control measures to mitigate the problems with real-life data were also discussed.

References

UCI Machine Learning Repository, German data set, available at: https://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data) (21 March 2017)

Liu, Z. & Zhang, Y. (2021) Credit evaluation with a data mining approach based on gradient boosting decision tree. *Journal of Physics: Conference Series* 1848(2021): DOI: https://doi:10.1088/1742-6596/1848/1/012034

CFI. (2022) Corporate Finance Institute. Available from: https://corporatefinanceinstitute.com/resources/knowledge/credit/credit-analysis-process/ [Accessed 30 May 2022].

Lessmann, S. , Baesens, B. , Seow, H.-V. & Thomas, L. C. (2015). Benchmarking state-of-theart classification algorithms for credit scoring: An update of re- search. *European Journal of Operational Research* 247: 124–136.

Raschka, S. (2018) Model evaluation, model selection, and algorithm selection in machine learning. arXiv preprint arXiv:1811.12808.

Xia, Y. et al. (2017) A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. *Expert Systems with Applications* 78: 225-241.

Frank, E., Hall, M.A & Witten, H. I (2016) The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques" Morgan Kaufmann, Fourth Edition, 2016.

Appendix: Screen shots of WEKA workspace

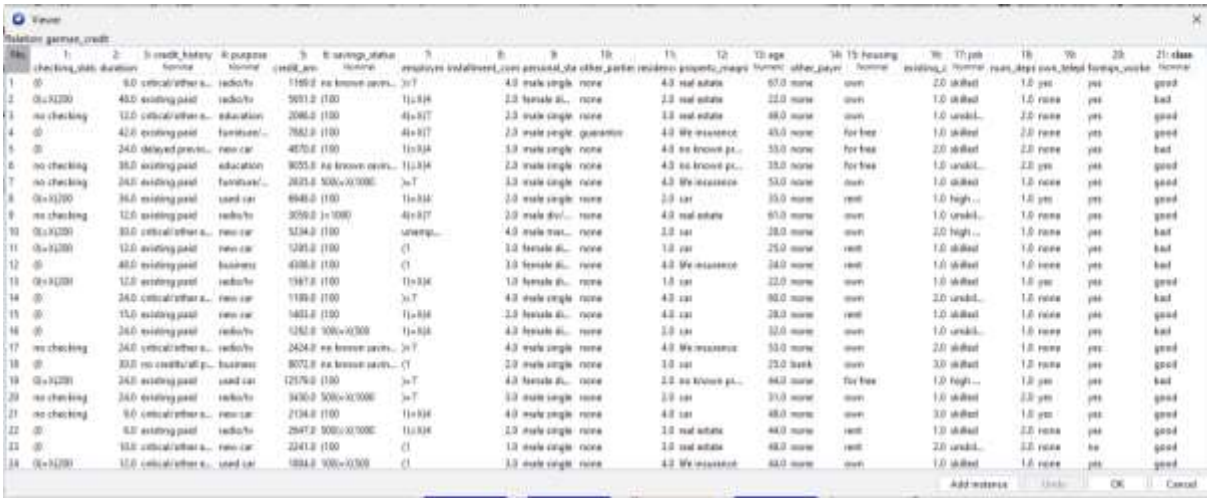

Figure 1: All attributes in the credit dataset

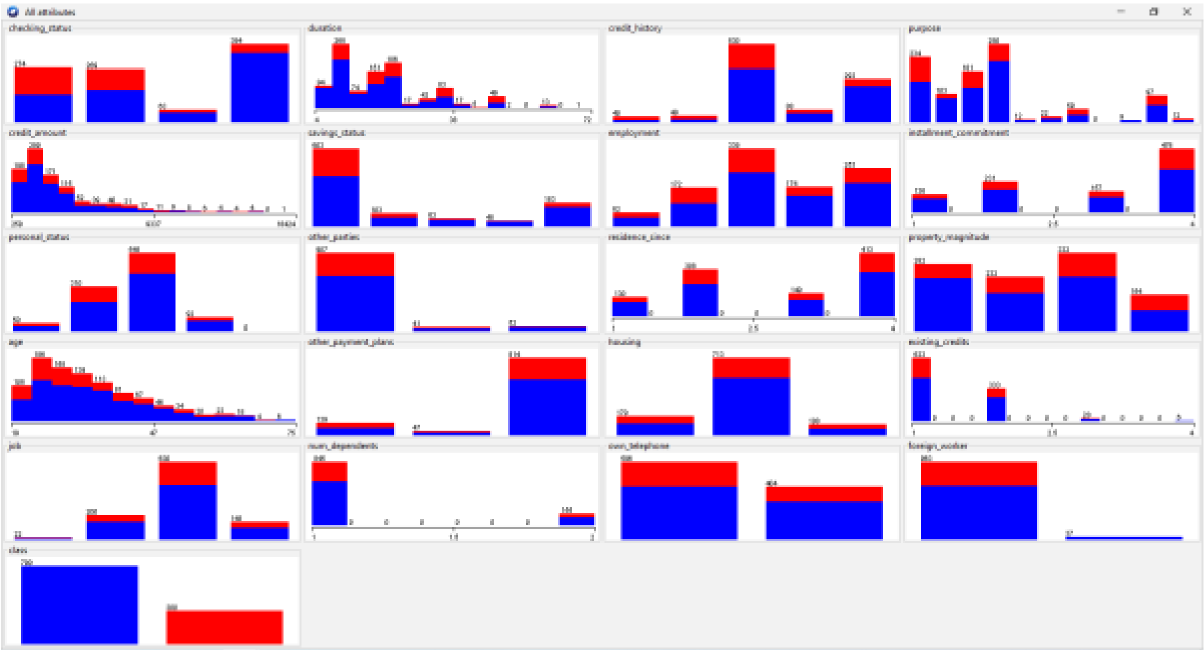

Figure 2: All attributes in the credit dataset, graphical display

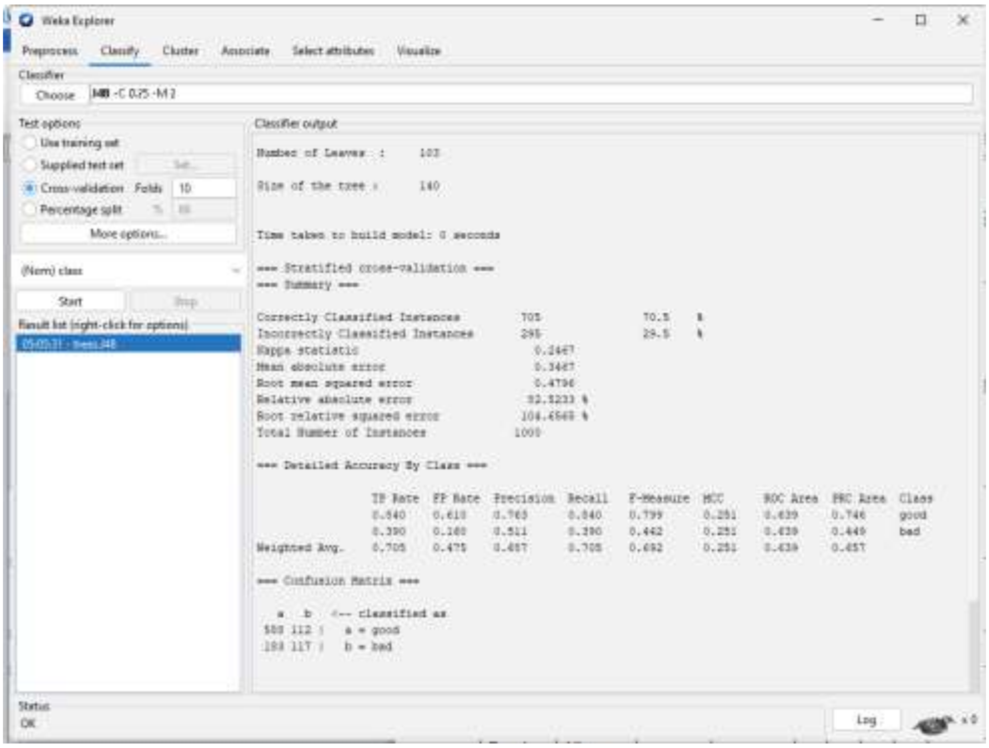

Figure 3: Output for J48 with cross-validation – all 20 attributes

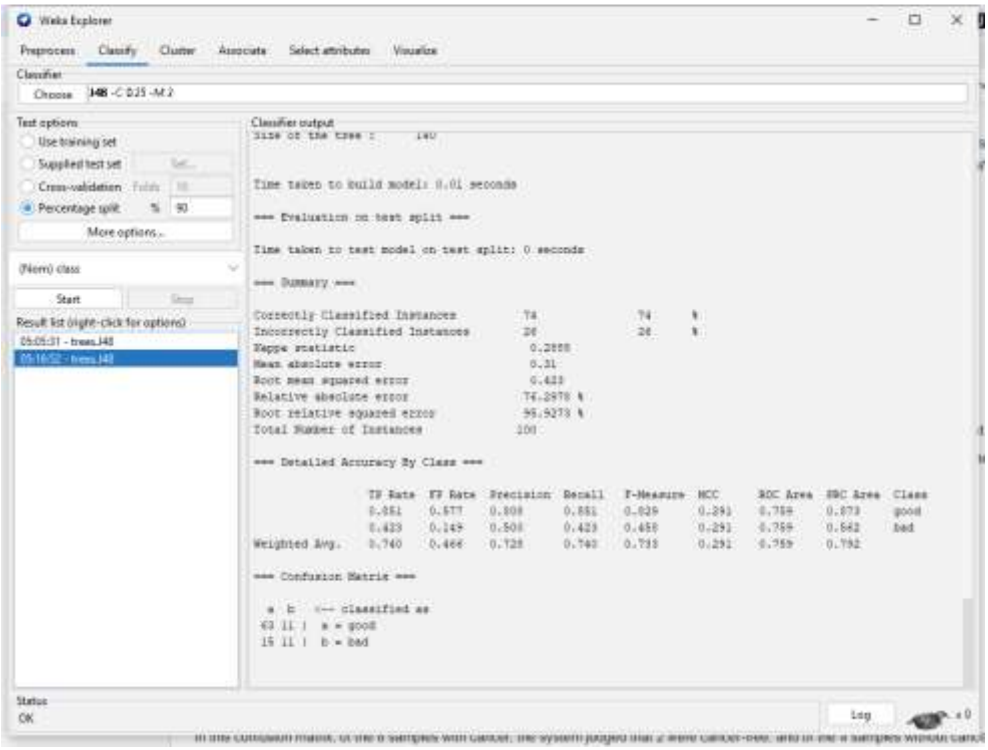

Figure 4: Output for J48 with percentage split – all 20 attributes

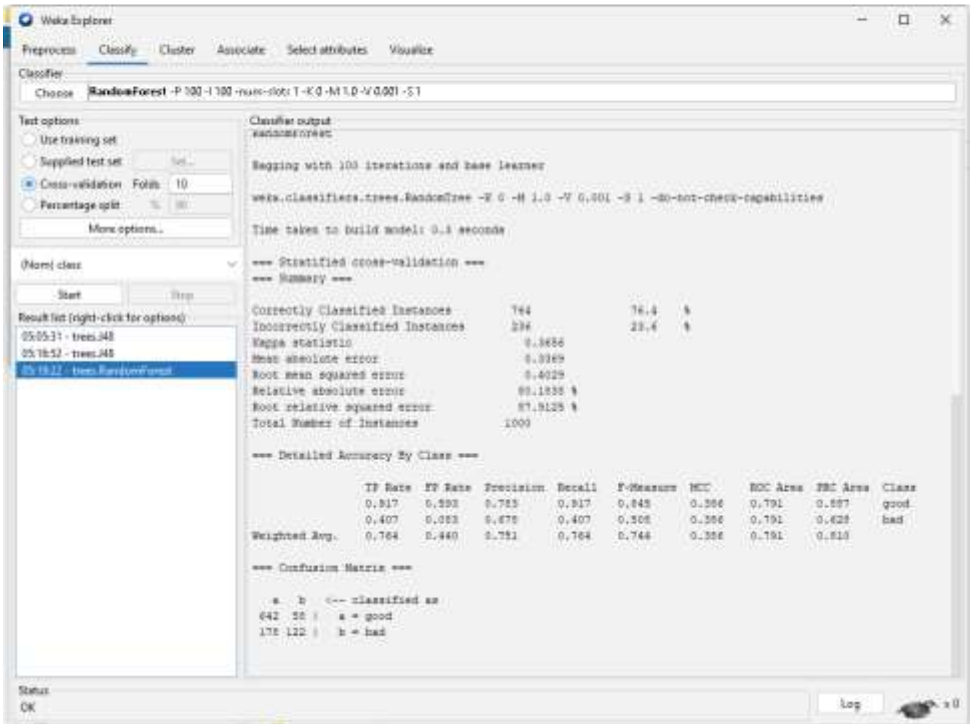

Figure 5: Output for Random forest with cross-validation – all 20 attributes

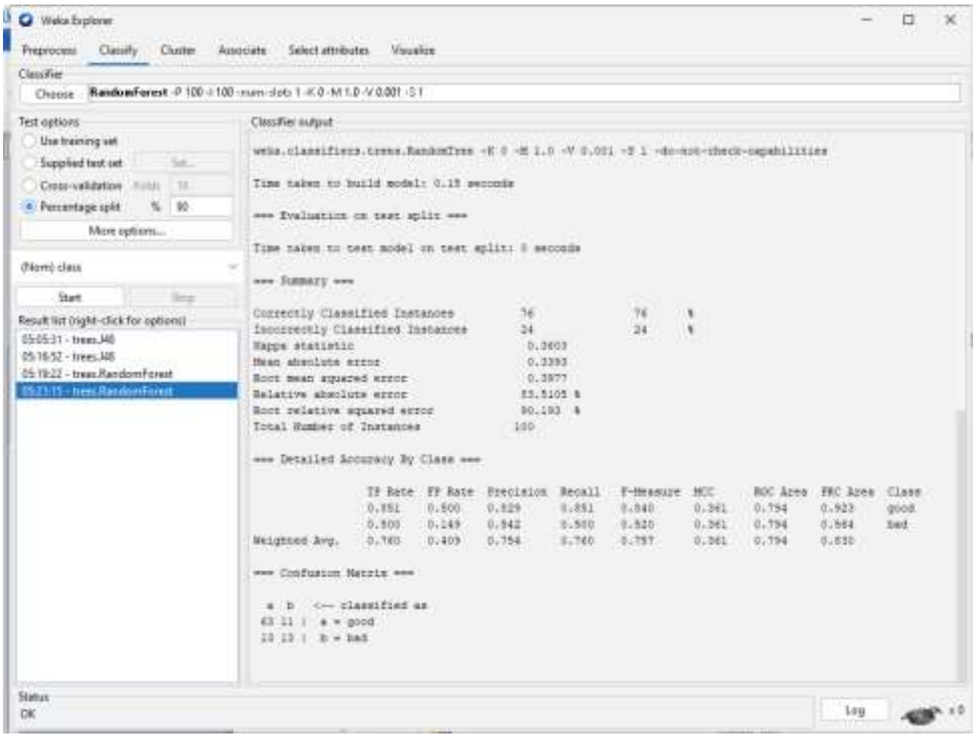

Figure 6: Output for Random forest with percentage split – all 20 attributes

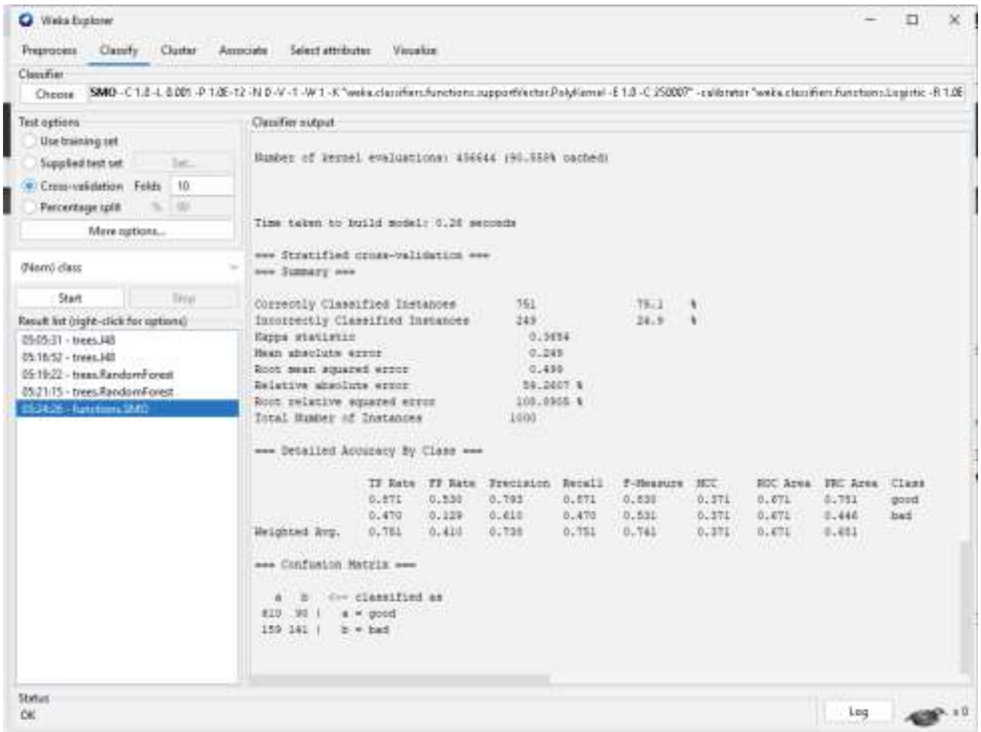

Figure 7: Output for SMO with cross-validation – all 20 attributes

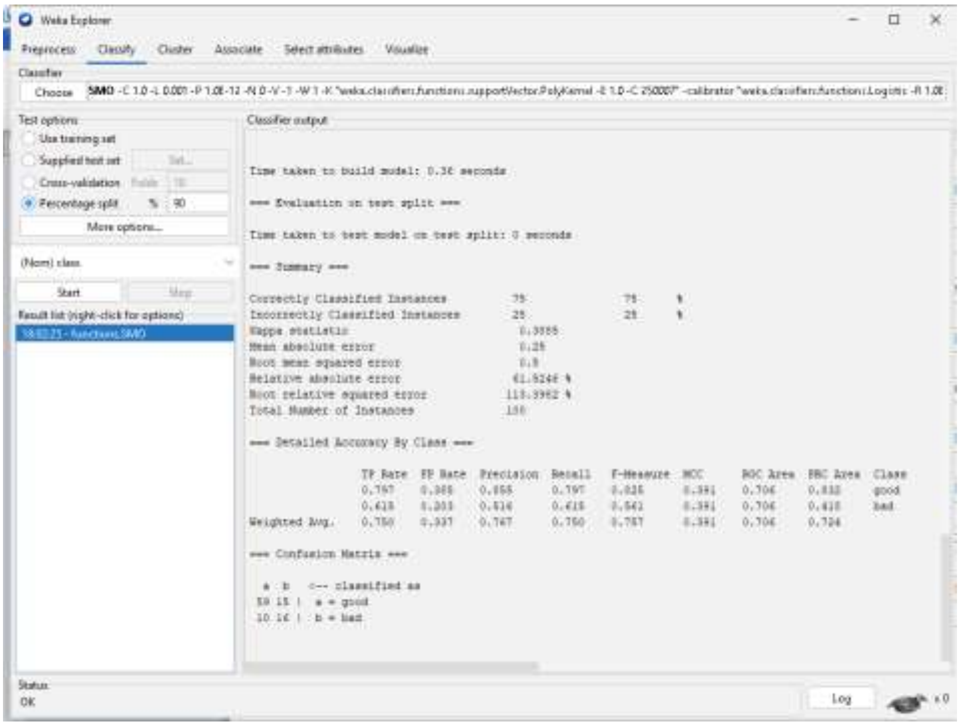

Figure 8: Output for SMO with Percentage split – all 20 attributes

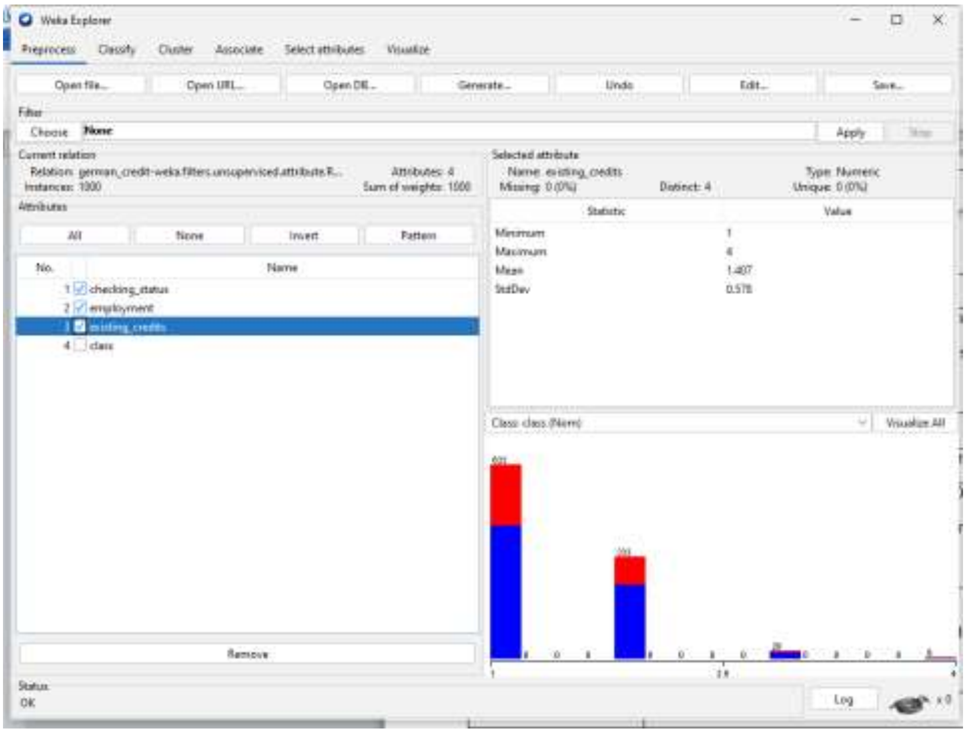

Figure 9: Feature selection - manual

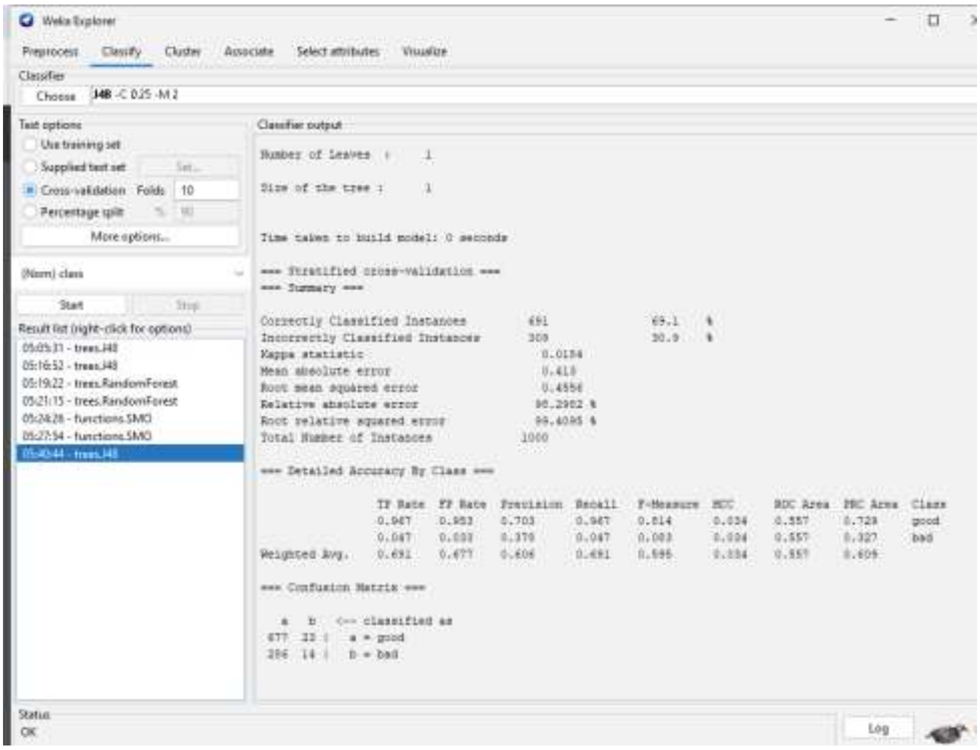

Figure 10: Output for J48 with cross-validation – manual feature selection

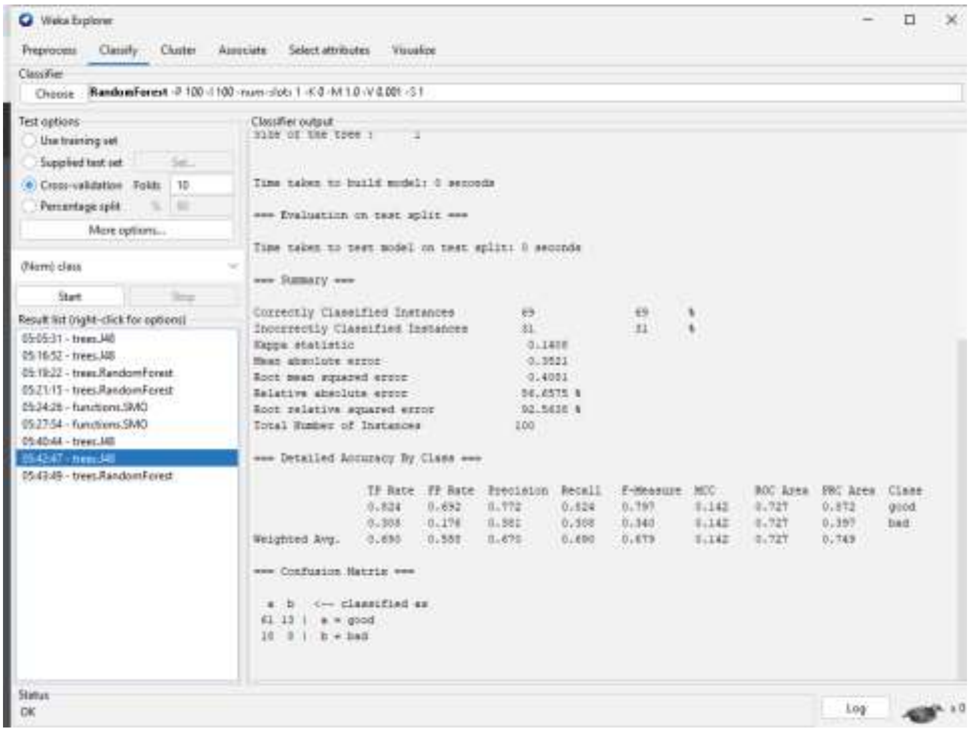

Figure 11: Output for J48 with percentage split – manual feature selection

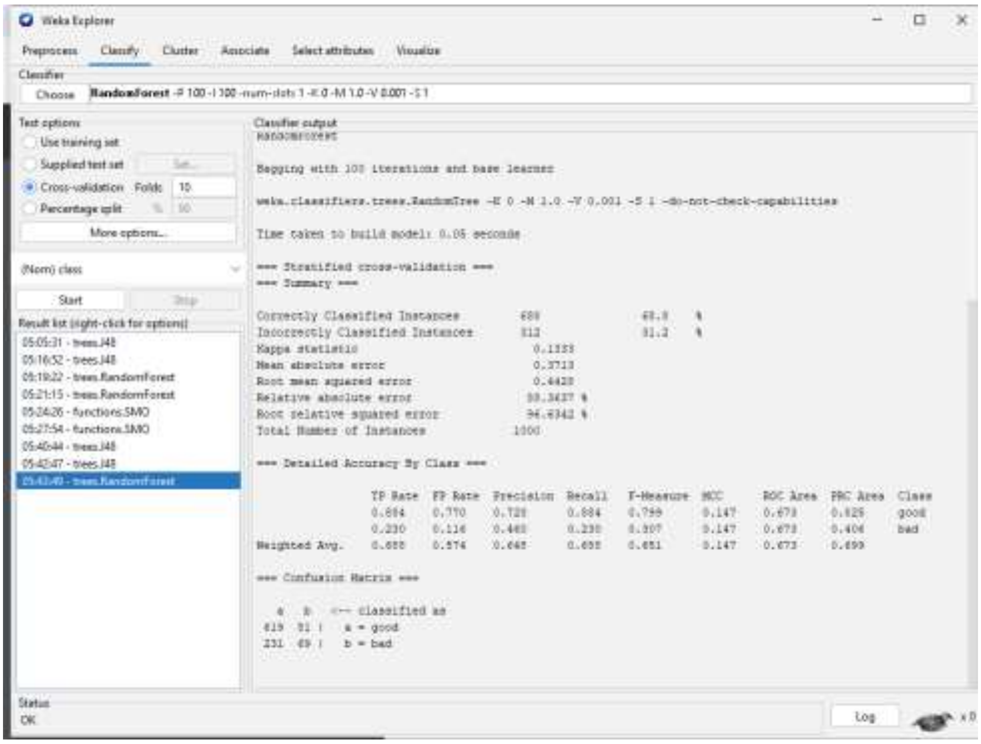

Figure 12: Output for Random forest with cross-validation – manual feature selection

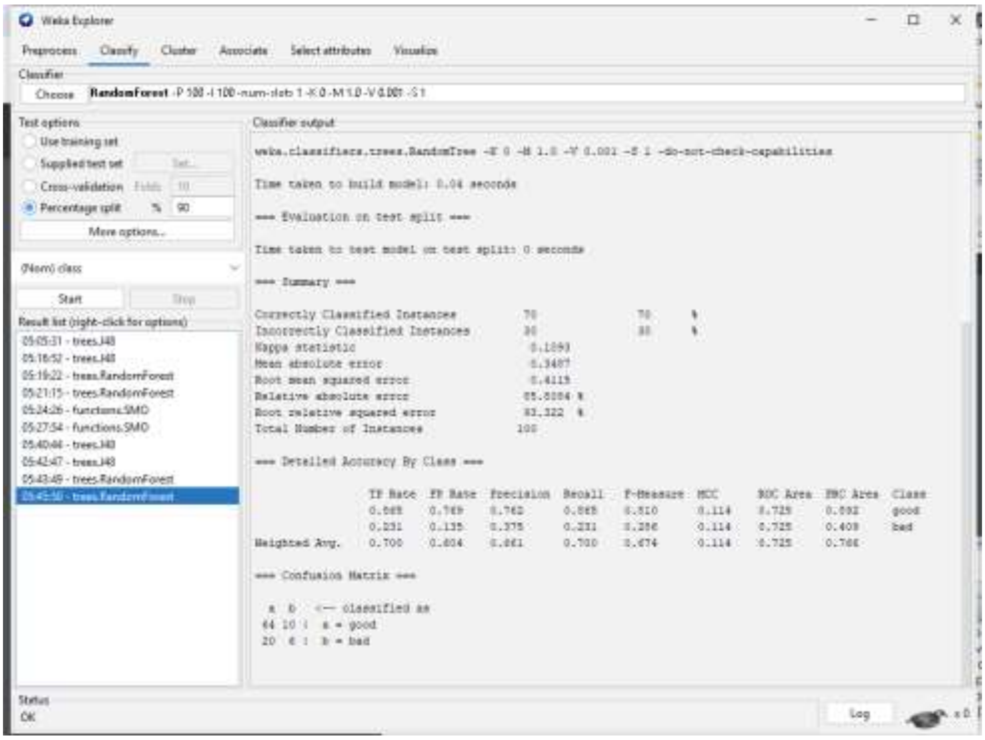

Figure 13: Output for Random forest with percentage split – manual feature selection

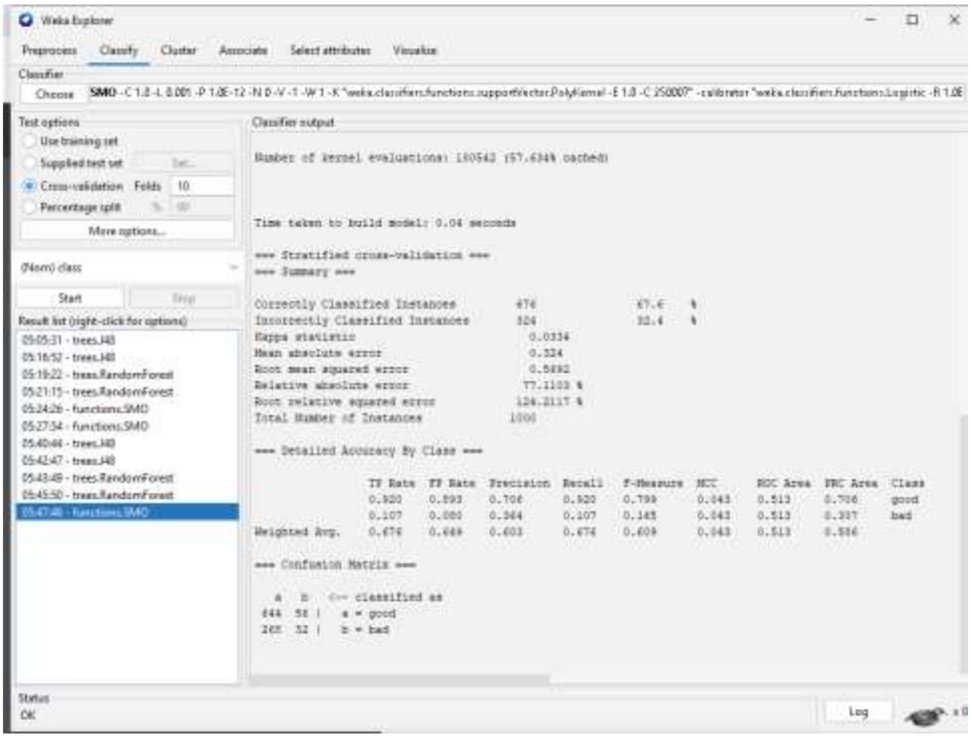

Figure 14: Output for SMO with cross-validation – manual feature selection

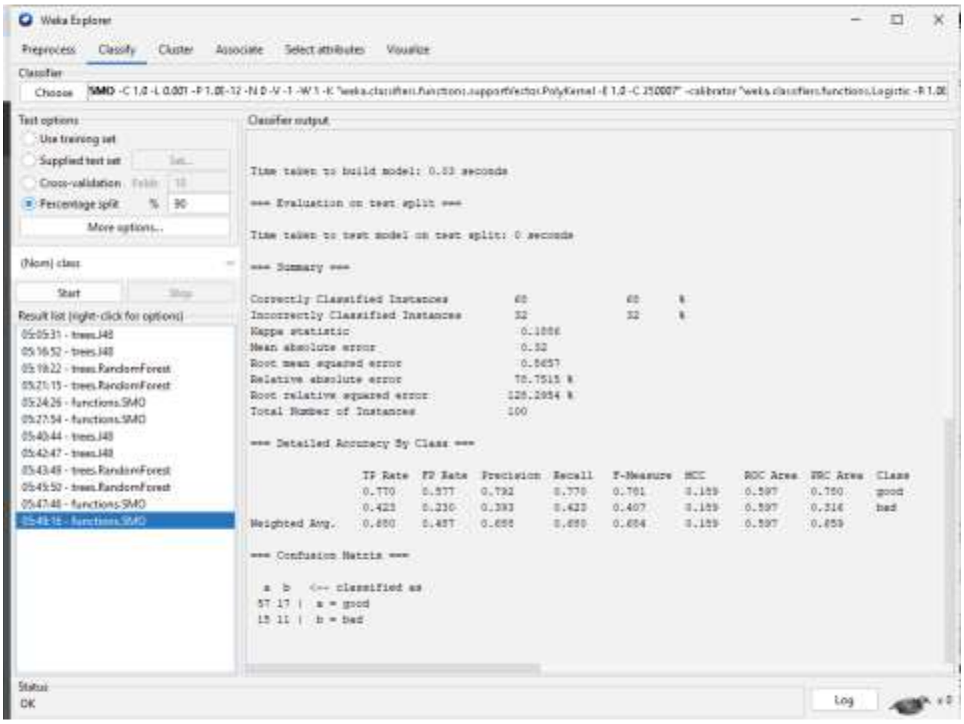

Figure 15: Output for SMO with Percentage split – manual feature selection

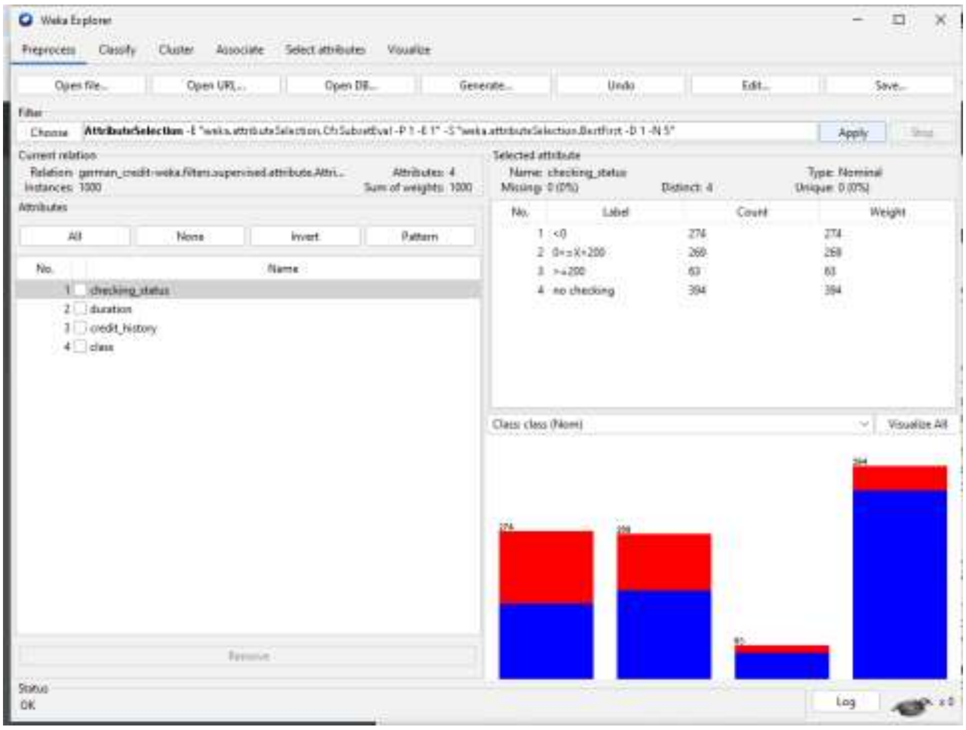

Figure 16: Feature selection - automatic

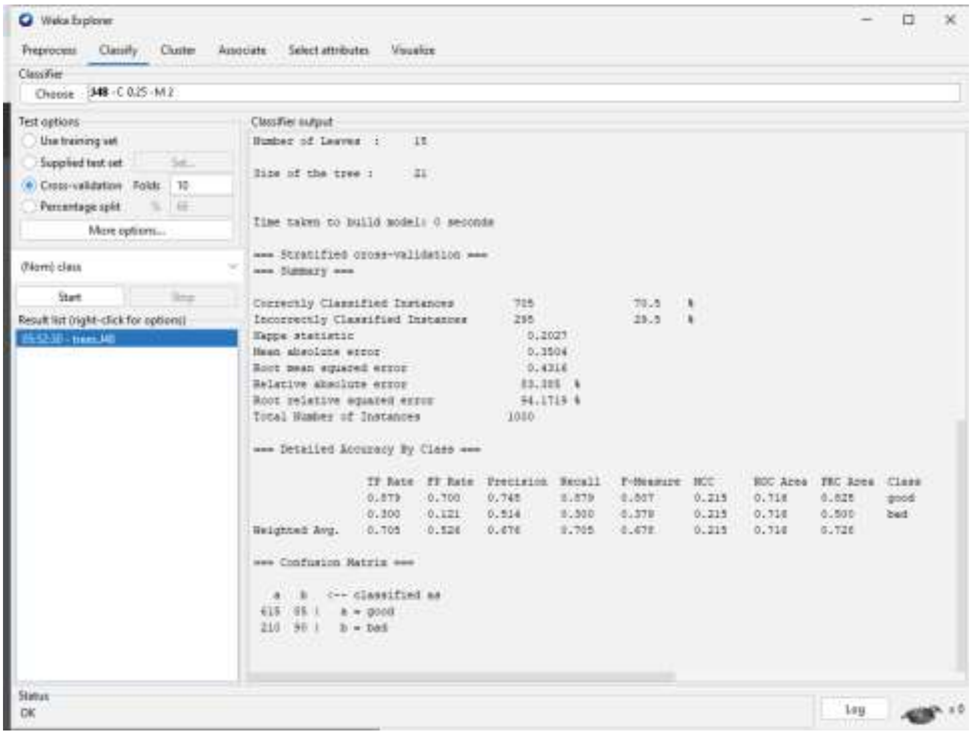

Figure 17: Output for J48 with Cross-validation – automatic feature selection

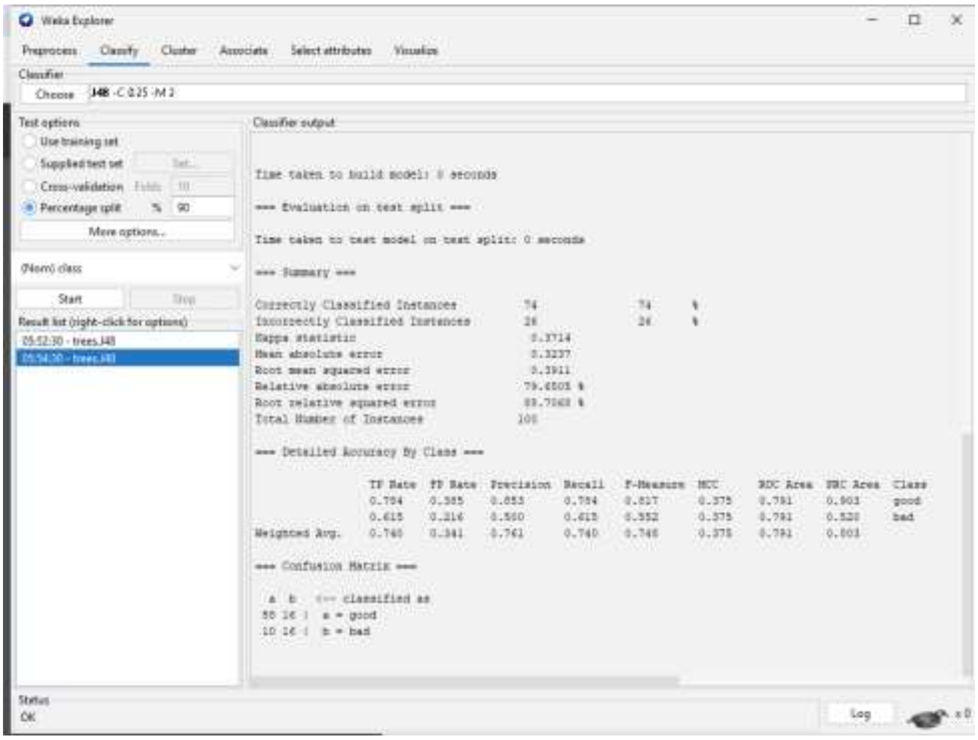

Figure 18: Output for J48 with Percentage split – automatic feature selection

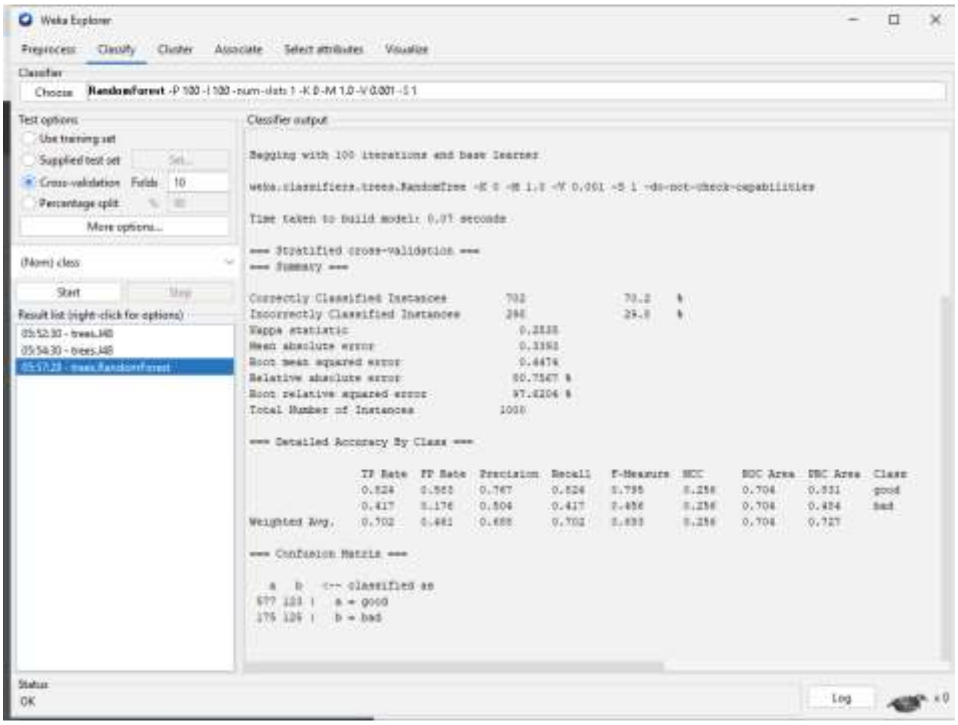

Figure 19: Output for Random forest with Cross-validation – automatic feature selection

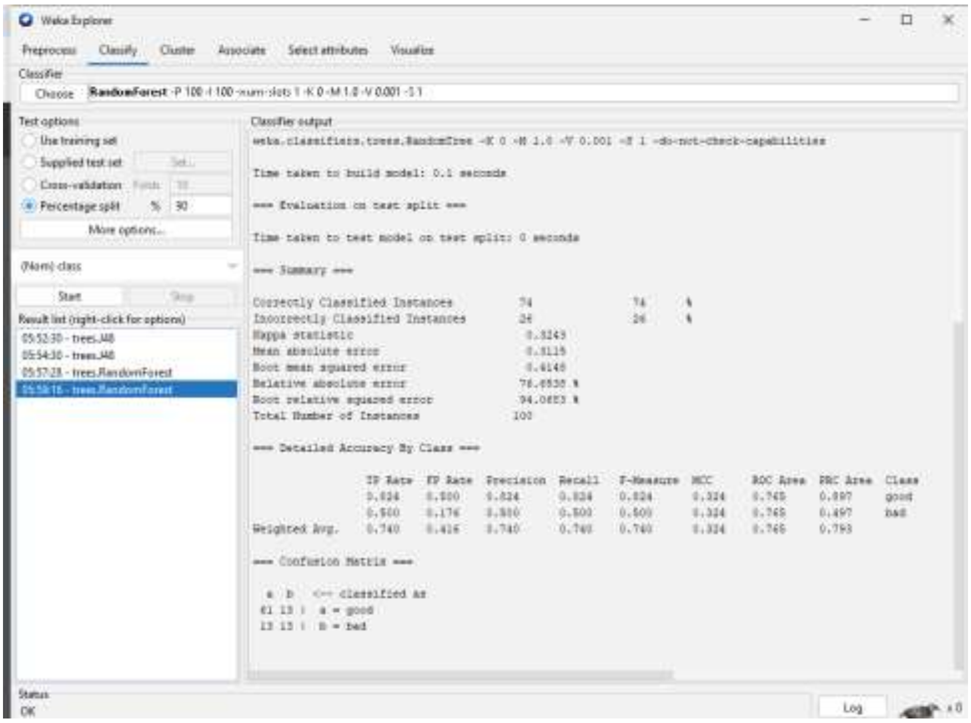

Figure 20: Output for Random forest with Percentage split – automatic feature selection

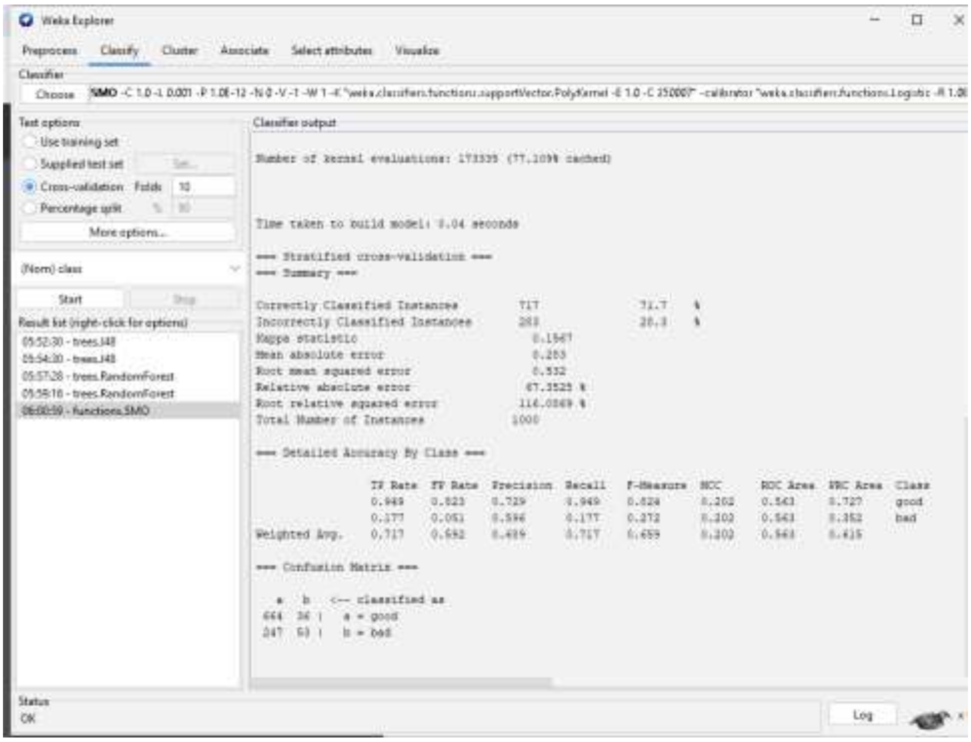

Figure 21: Output for SMO with Cross-validation – automatic feature selection

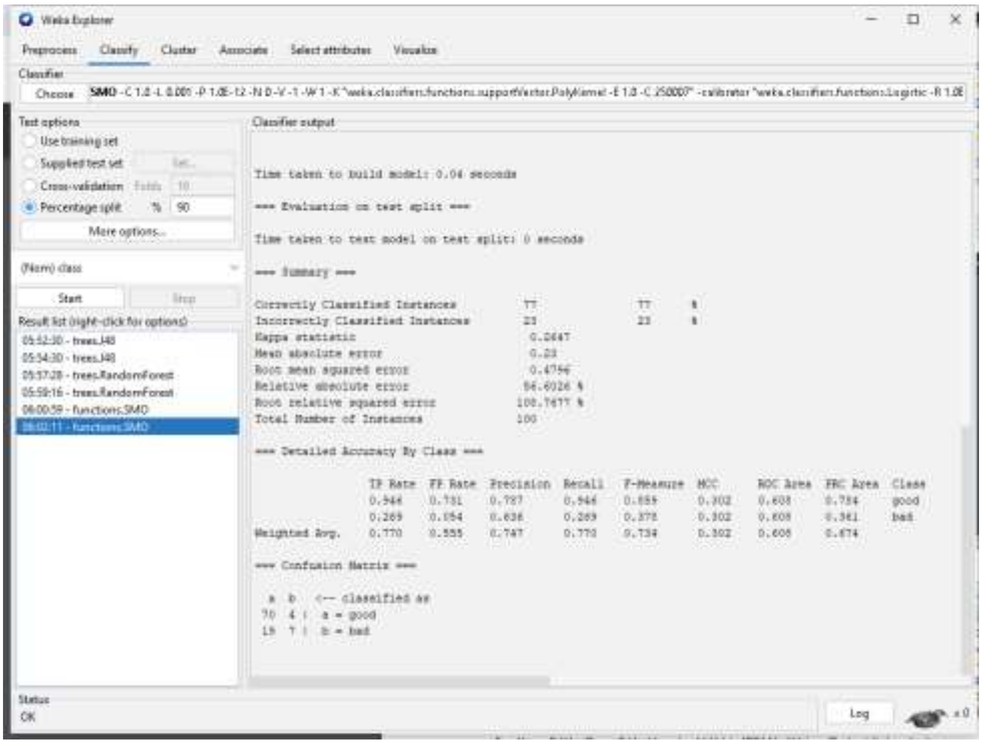

Figure 22: Output for SMO with Percentage split – automatic feature selection