PROJECT 1: DATA ANALYSIS (MUSHROOM)

4IZ451 - Knowledge Discovery in Databases, Prof. Ing. Petr Berka

Author:

Prague

Contents

1. Introduction

The purpose of this coursework is to analyze “Mushroom” dataset by using 4 different data-mining software tools, namely Weka, RapidMiner, SAS Enterprise Miner and IBM SPSS Modeler. Main aim of analyzing dataset is to identify whether mushroom is safe to eat or poisoned.

1.1 Data description

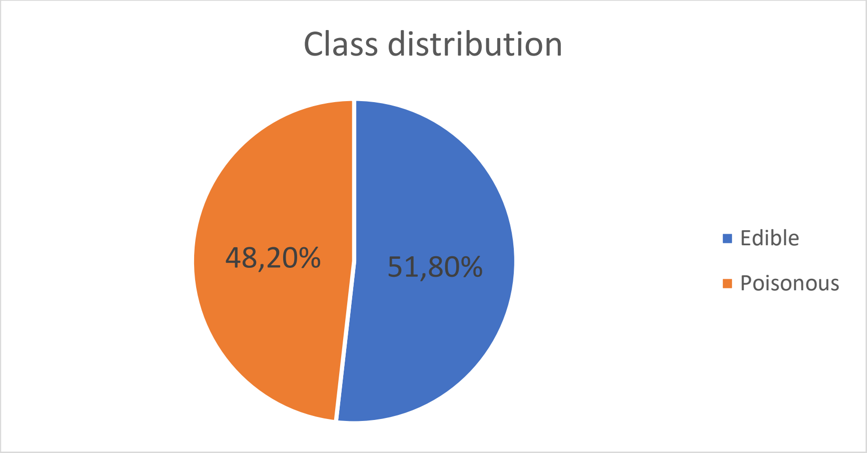
Number of Instances: 8124

Number of Attributes: 22 (all nominally valued)

Missing Attribute Values: 2480

Class Distribution:

- Edible: 4208 (51.8%)
- Poisonous: 3916 (48.2%)



1.2 Attributes

#	Abbreviation	Description
1	cap-shape	bell=b,conical=c,convex=x,flat=f, knobbed=k,sunken=s
2	cap-surface	fibrous=f,grooves=g,scaly=y,smooth=s
3	cap-color	brown=n,buff=b,cinnamon=c,gray=g,green=r, pink=p,purple=u,red=e,white=w,yellow=y
4	bruises?	bruises=t,no=f
5	odor	almond=a,anise=l,creosote=c,fishy=y,foul=f, musty=m,none=n,pungent=p,spicy=s
6	gill-attachment	attached=a,descending=d,free=f,notched=n
7	gill-spacing	close=c,crowded=w,distant=d
8	gill-size	broad=b,narrow=n
9	gill-color	black=k, brown=n, buff=b, chocolate=h, gray=g, green=r, orange=o, pink=p, purple=u, red=e, white=w, yellow=y
10	stalk-shape	enlarging=e, tapering=t
11	stalk-root	bulbous=b, club=c, cup=u, equal=e, rhizomorphs=z, rooted=r, missing=?
12	stalk-surface-above-ring	ibrous=f, scaly=y, silky=k, smooth=s
13	stalk-surface-below-ring	ibrous=f, scaly=y, silky=k, smooth=s

14	stalk-color-above-ring	brown=n, buff=b, cinnamon=c, gray=g, orange=o, pink=p, red=e, white=w, yellow=y
15	stalk-color-below-ring	brown=n, buff=b, cinnamon=c, gray=g, orange=o, pink=p, red=e, white=w, yellow=y
16	veil-type	partial=p, universal=u
17	veil-color	brown=n, orange=o, white=w, yellow=y
18	ring-number	none=n, one=o, two=t
19	ring-type	cobwebby=c, evanescent=e, flaring=f, large=l, none=n, pendant=p, sheathing=s, zone=z
20	spore-print-color	black=k, brown=n, buff=b, chocolate=h, green=r, orange=o, purple=u, white=w, yellow=y
21	population	abundant=a, clustered=c, numerous=n, scattered=s, several=v, solitary=y
22	habitat	grasses=g, leaves=l, meadows=m, paths=p, urban=u, waste=w, woods=d

Table 1. Attributes representing board position

1.3 Data pre-processing & preparation

All attributes and instances of the dataset, as well as class attribute was ready for processing information. So, no data pre-processing actions executed before analysis. It also should be noted that, dataset contains missing values in amount: 2480 of them (denoted by "?"), all for attribute #11. Which was not replaced or changed while analyzing the data. Missing values were considered and processed by data-mining tools.

Data partition and class attributes selection was done using options (nodes or process steps) of data-mining tools.

Finally, dataset was converted from “.arff” to “.csv”, in order to be able to analyze data in RapidMiner, IBM SPSS Modeler and SAS Enterprise Miner.

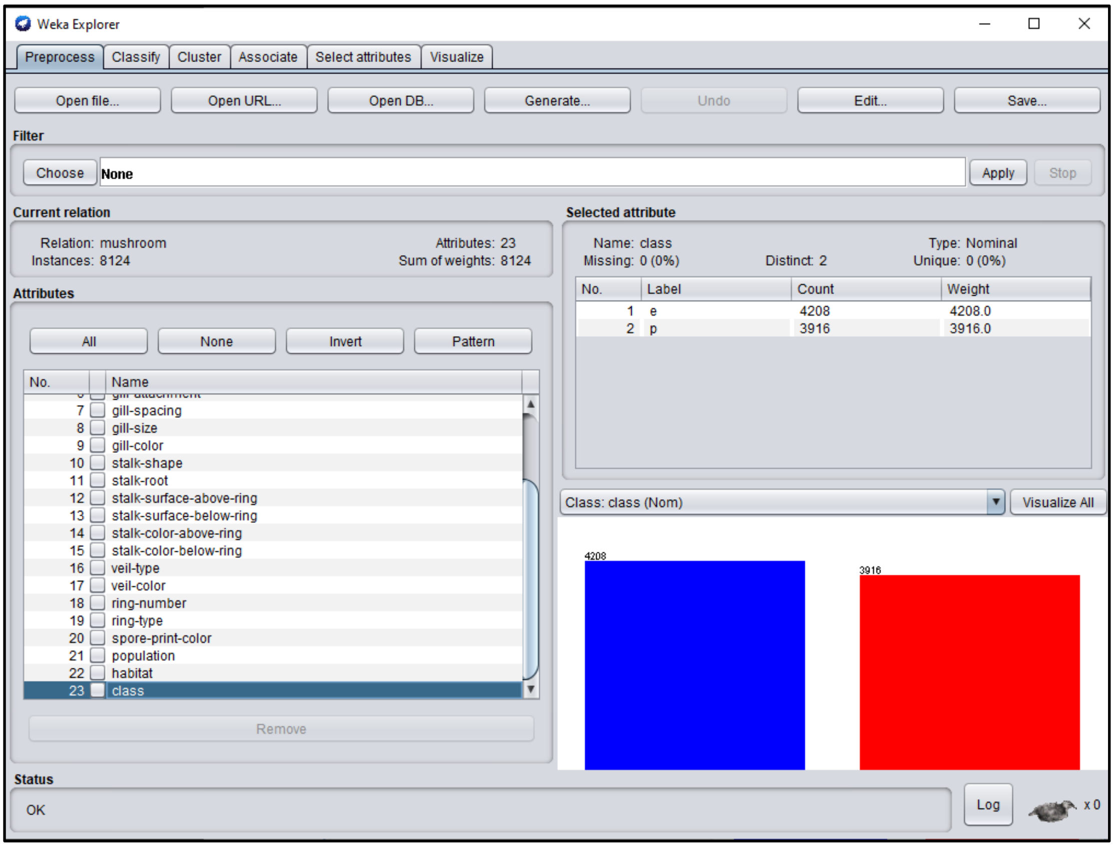
2. Datamining with Weka 3.8

Weka is a collection of machine learning algorithms for data mining tasks. It contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization.

Weka is open source software issued under the GNU General Public License, developed by Machine Learning Group at the University of Waikato, New Zealand. [1]

2.1 Modelling

First step in order to be able to process modelling algorithms is the data import and explorer tool. Successful data import and selecting of target attribute can be seen on figure 1.



Filter: None

Current relation

Relation: mushroom | Attributes: 23
Instances: 8124 | Sum of weights: 8124

Selected attribute

Name: class | Type: Nominal
Missing: 0 (0%) | Distinct: 2 | Unique: 0 (0%)

No.	Label	Count	Weight
1	e	4208	4208.0
2	p	3916	3916.0

Attributes

No.	Name
6	gill-attachment
7	gill-spacing
8	gill-size
9	gill-color
10	stalk-shape
11	stalk-root
12	stalk-surface-above-ring
13	stalk-surface-below-ring
14	stalk-color-above-ring
15	stalk-color-below-ring
16	veil-type
17	veil-color
18	ring-number
19	ring-type
20	spore-print-color
21	population
22	habitat
23	class

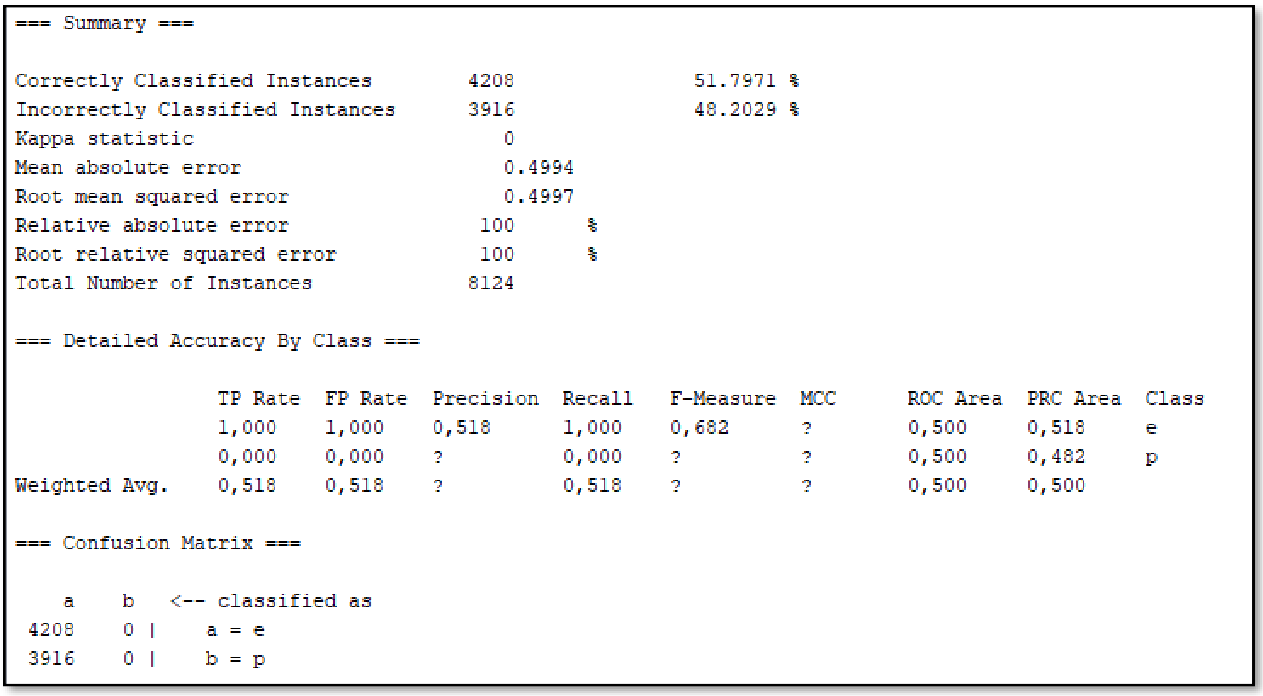
Class: class (Nom)

Status
OK

Figure 1 - Zero Rule

Zero R or zero rule is the elementary classification technique which ignores all predictors and relies on the class attribute. In other words, Zero R classifier just forecast the general attribute. It is useful for defining a baseline efficiency as a reference point for other classification methods.

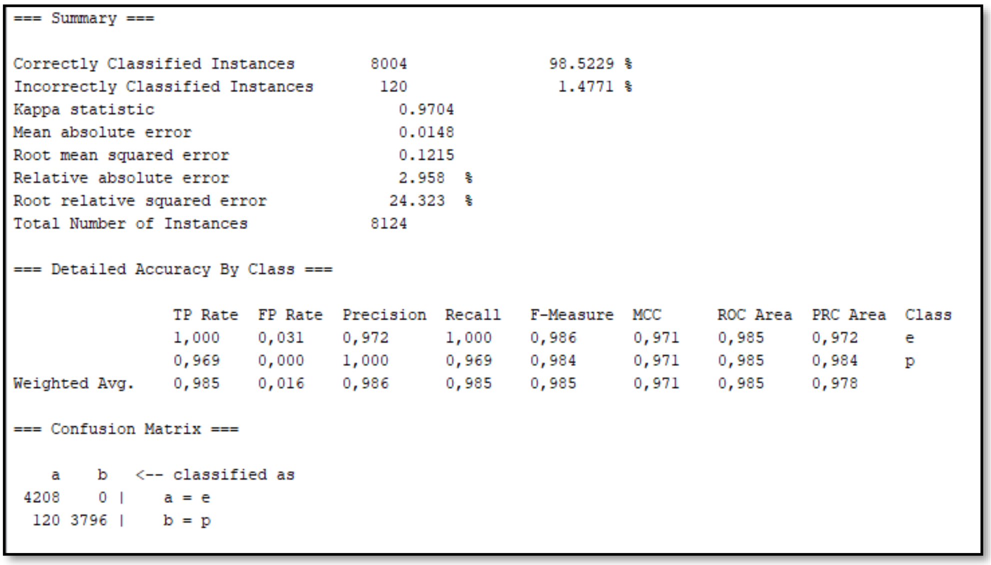
2.1.1 Zero Rule



```
=== Summary ===

Correctly Classified Instances        4208               51.7971 %
Incorrectly Classified Instances      3916               48.2029 %
Kappa statistic                          0
Mean absolute error                      0.4994
Root mean squared error                  0.4997
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances             8124

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    1,000    0,518      1,000    0,682      ?        0,500     0,518     e
                 0,000    0,000    ?          0,000    ?          ?        0,500     0,482     p
Weighted Avg.    0,518    0,518    ?          0,518    ?          ?        0,500     0,500

=== Confusion Matrix ===

    a    b   <-- classified as
 4208    0 |    a = e
 3916    0 |    b = p
```

2.1.2 One Rule

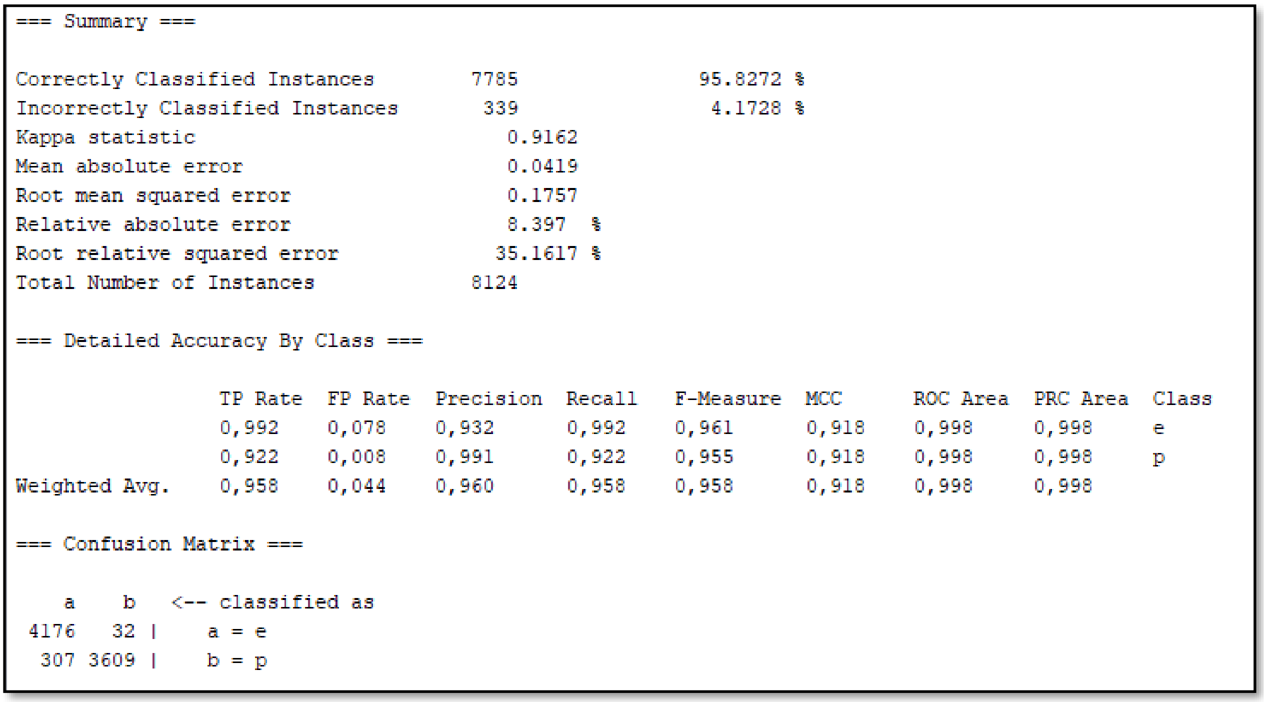
One Rule algorithm is slightly more complex than the Zero Rule – it chooses the best attribute in dataset to predict the target.



```
=== Summary ===

Correctly Classified Instances        8004               98.5229 %
Incorrectly Classified Instances       120                1.4771 %
Kappa statistic                          0.9704
Mean absolute error                      0.0148
Root mean squared error                  0.1215
Relative absolute error                  2.958  %
Root relative squared error             24.323  %
Total Number of Instances             8124

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    0,031    0,972      1,000    0,986      0,971    0,985     0,972     e
                 0,969    0,000    1,000      0,969    0,984      0,971    0,985     0,984     p
Weighted Avg.    0,985    0,016    0,986      0,985    0,985      0,971    0,985     0,978

=== Confusion Matrix ===

    a    b   <-- classified as
 4208    0 |    a = e
  120 3796 |    b = p
```

2.1.3 Naive Bayes

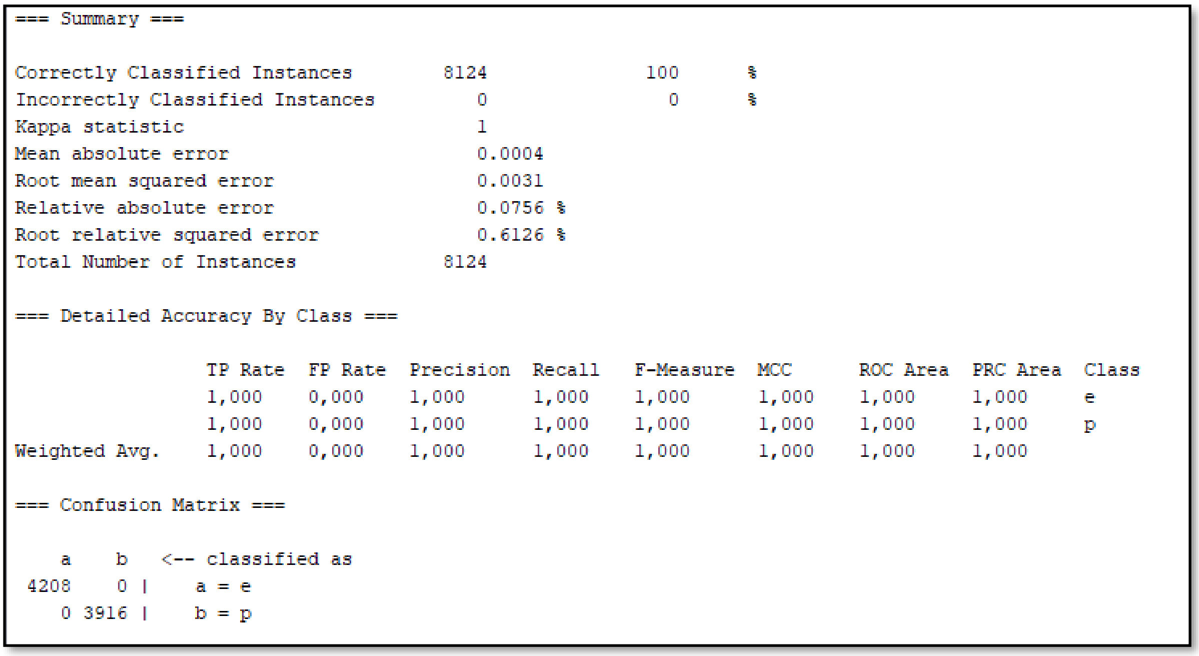
It is a classification technique based on Bayes’ Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. [2]



```
=== Summary ===

Correctly Classified Instances        7785               95.8272 %
Incorrectly Classified Instances       339                4.1728 %
Kappa statistic                          0.9162
Mean absolute error                      0.0419
Root mean squared error                  0.1757
Relative absolute error                  8.397  %
Root relative squared error             35.1617 %
Total Number of Instances             8124

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,992    0,078    0,932      0,992    0,961      0,918    0,998     0,998     e
                 0,922    0,008    0,991      0,922    0,955      0,918    0,998     0,998     p
Weighted Avg.    0,958    0,044    0,960      0,958    0,958      0,918    0,998     0,998

=== Confusion Matrix ===

    a    b   <-- classified as
 4176   32 |    a = e
  307 3609 |    b = p
```

2.1.4 Random Forrest

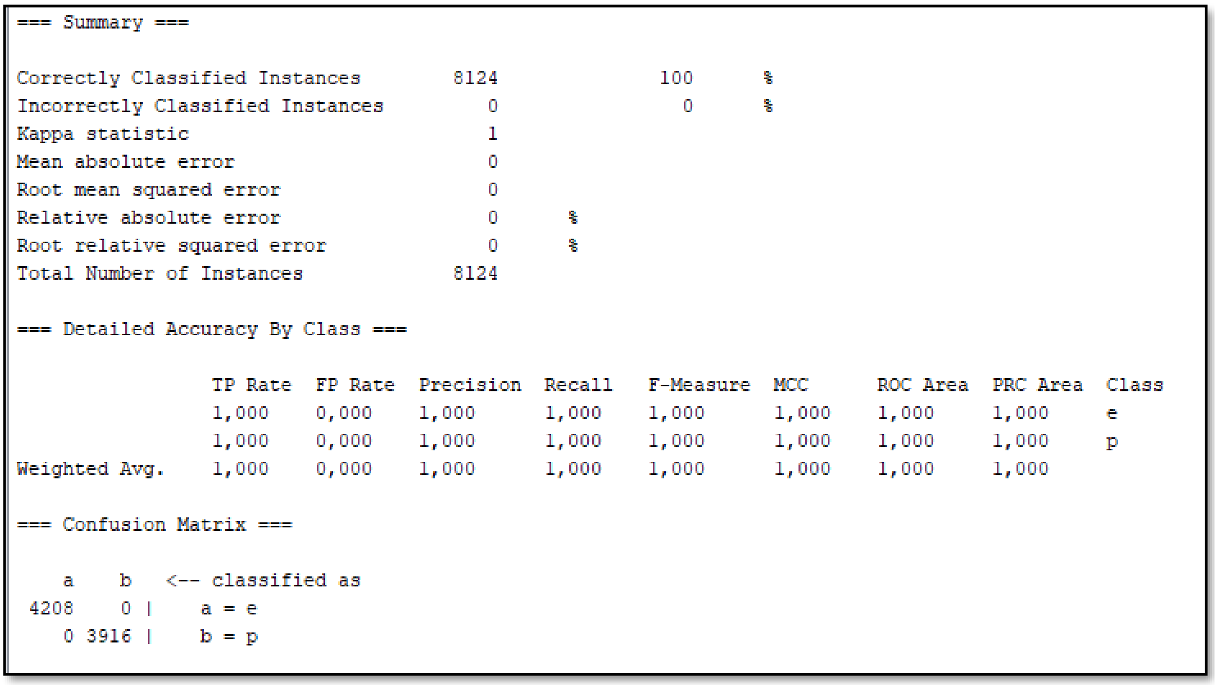
Random forest classifier creates a set of decision trees from randomly selected subset of training set. It then aggregates the votes from different decision trees to decide the final class of the test object.



```
=== Summary ===

Correctly Classified Instances        8124              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0.0004
Root mean squared error                  0.0031
Relative absolute error                  0.0756 %
Root relative squared error              0.6126 %
Total Number of Instances             8124

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     e
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     p
Weighted Avg.    1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000

=== Confusion Matrix ===

    a    b   <-- classified as
 4208    0 |    a = e
    0 3916 |    b = p
```

2.1.5 J48 Decision tree

Decision tree generates model in the form of a tree pattern. Decision tree is developed step-by-step by breaking down a data into subsets. The last outcome of this algorithm is a tree with decision branches and leaf points. Decision trees can be used for analyzing categorical and numerical datasets.



```
=== Summary ===

Correctly Classified Instances        8124              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances             8124

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     e
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     p
Weighted Avg.    1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000

=== Confusion Matrix ===

    a    b   <-- classified as
 4208    0 |    a = e
    0 3916 |    b = p
```

2.2 Results

Algorithm	Correctly classified	Success rate
Zero Rule	4208	51.79%
One Rule	8004	98.52%
Naive Bayes	7785	95.82%
Random Forrest	8124	100%
J48	8124	100%

The best algorithms were decision trees – namely decision forest and J48 with 100% success rate. It is quite interesting that simple classification algorithm such as One Rule present good success rate of 98.52%. Naive Bayes provided 95,82% success rate, while, Zero Rule classification algorithm is unusable for this dataset because it presented only 51.79%.

3 Rapid Miner

RapidMiner is a software platform developed for machine learning, data mining, text mining, predictive analytics and business analytics. It is used for business and industrial applications as well as for research, education, training, rapid prototyping, and application development and supports all steps of the data mining process including results visualization, validation and optimization. [3]

3.1 Modelling

RapidMiner provides users with highly intuitive modelling process window that covers all data-mining steps. Firstly, users have to specify the class attribute – in RapidMiner called label, secondly multiply the data stream for validation boxes and in the end – very intuitively specify what will be included in final report by connecting “boxes” with the output column. The filter operator was added because after reading CSV file the class attribute had 3 missing values which was not in original dataset.
The project setup can be seen on figure 2.

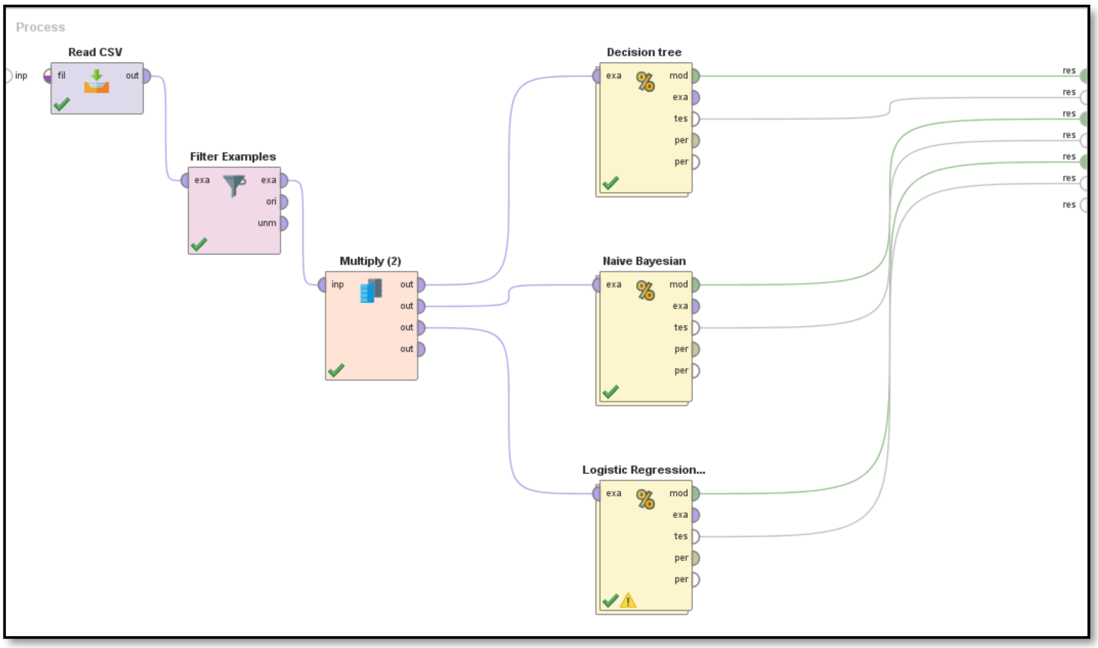


Figure 2 - Rapid Miner project setup

3.1.1 Decision Tree

The design of Decision tree validation box can be seen on figure 3.

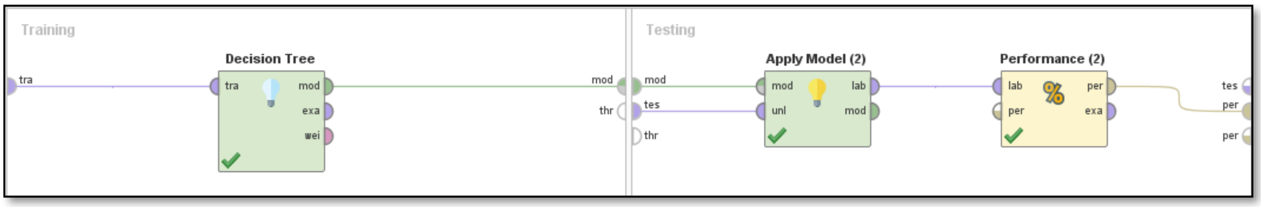
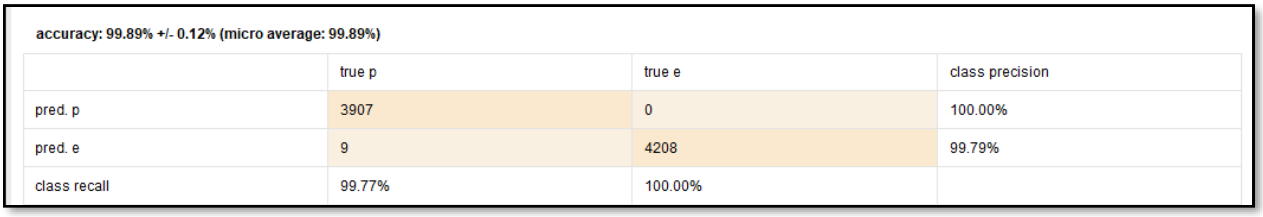


Figure 3 - RapidMiner Decision tree validation box in detail



accuracy: 99.89% +/- 0.12% (micro average: 99.89%)

	true p	true e	class precision
pred. p	3907	0	100.00%
pred. e	9	4208	99.79%
class recall	99.77%	100.00%	

3.1.2 Naive Bayes

The modelling process of the Naive Bayes is the same as in case of Decision tree algorithm and can be seen on figure 4.

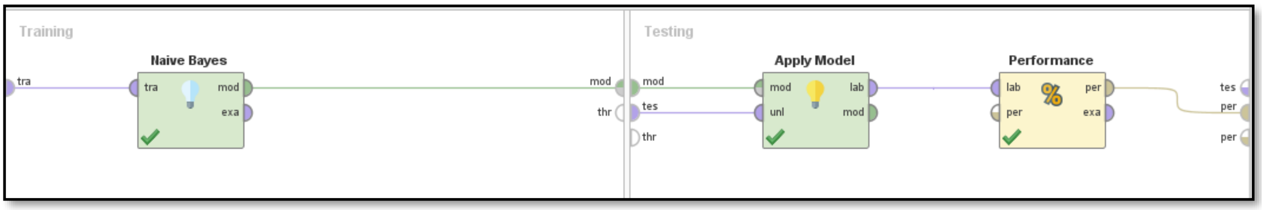
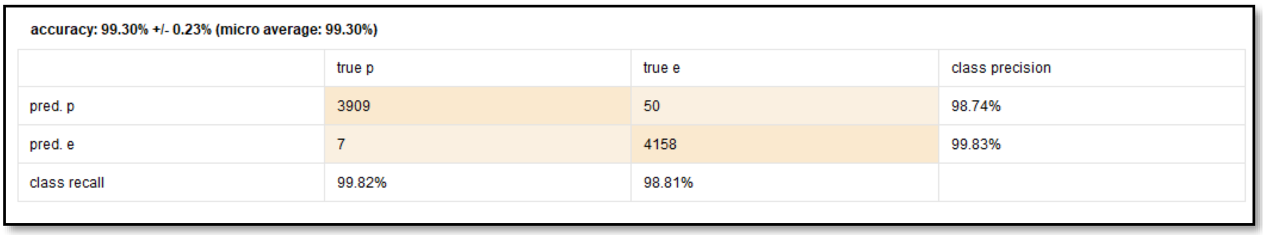


Figure 4 - RapidMiner Naive Bayes validation box in detail



accuracy: 99.30% +/- 0.23% (micro average: 99.30%)

	true p	true e	class precision
pred. p	3909	50	98.74%
pred. e	7	4158	99.83%
class recall	99.82%	98.81%	

3.1.3 Logistic Regression

The modelling process of Logistic Regression algorithm is the same as in case of Naive Bayes algorithm – the inner part of the validation box can be seen on figure 5.

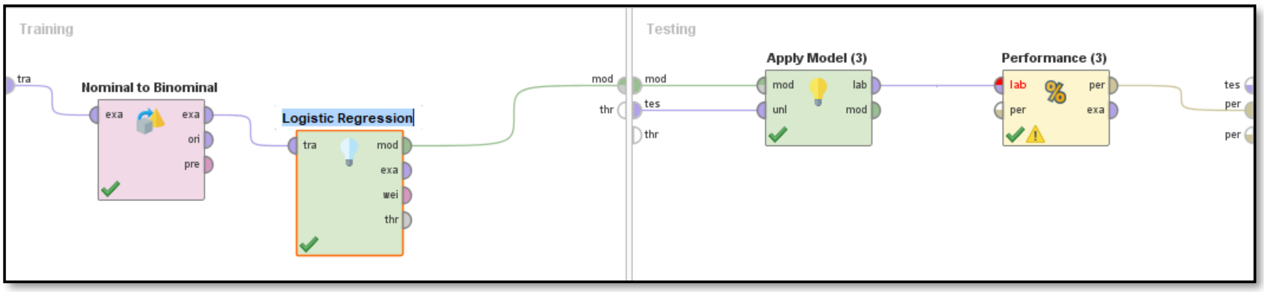
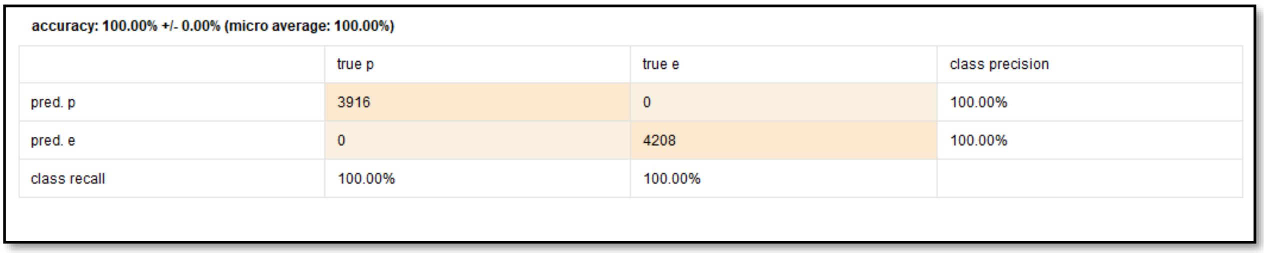


Figure 5 - Logistic Regression validation box in detail



accuracy: 100.00% +/- 0.00% (micro average: 100.00%)

	true p	true e	class precision
pred. p	3916	0	100.00%
pred. e	0	4208	100.00%
class recall	100.00%	100.00%	

3.2 Results

Algorithm	Correctly classified	Success rate
Decision Tree	8115	99.89
Naive Bayes	8067	99.3%
Logistic Regression	8124	100%

As we can see the Logistic Regression present the best result of 100% success rate.

4 IBM SPSS Modeler

IBM SPSS Modeler is an extensive predictive analytics platform that is designed to bring predictive intelligence to decisions made by individuals, groups, systems and the enterprise. By providing a range of advanced algorithms and techniques that include text analytics, entity analytics, decision management and optimization. [4]

4.1 Modelling

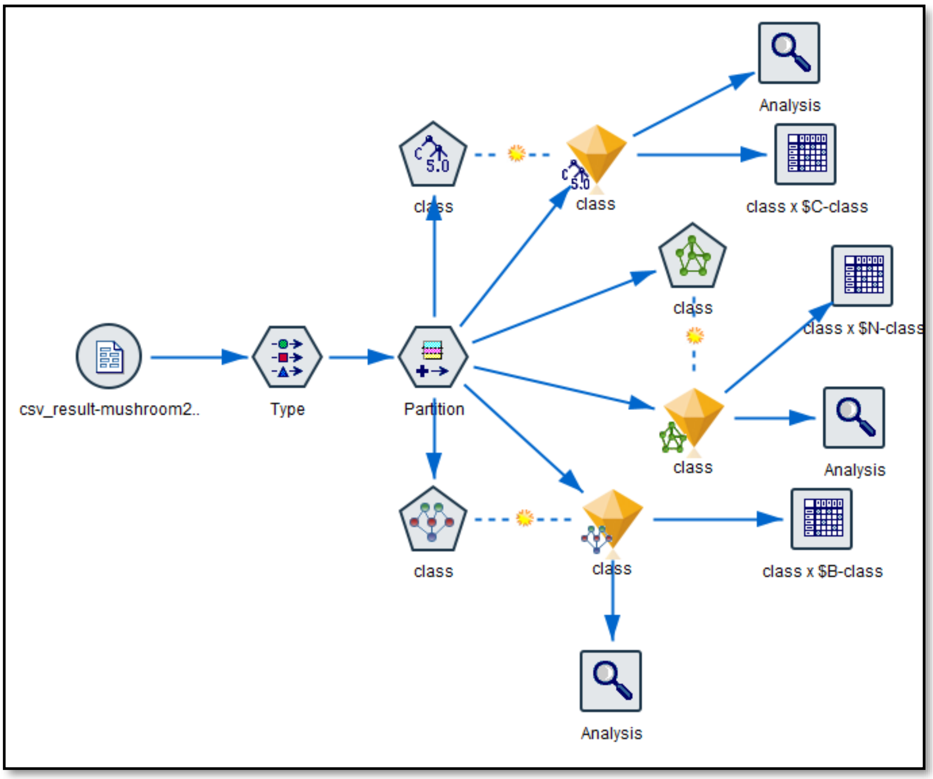
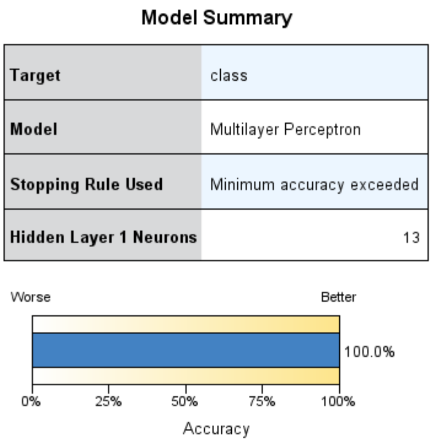


Figure 6 - IBM SPSS Modeler project setup

4.1.1 Neural Net

IBM SPSS Modeler enables users to build neural network model from nominal data, results can be seen in the result window and on the figure 7 and figure 8.



Model Summary

Target	class
Model	Multilayer Perceptron
Stopping Rule Used	Minimum accuracy exceeded
Hidden Layer 1 Neurons	13

Figure 7 - IBM SPSS Neural network model summary

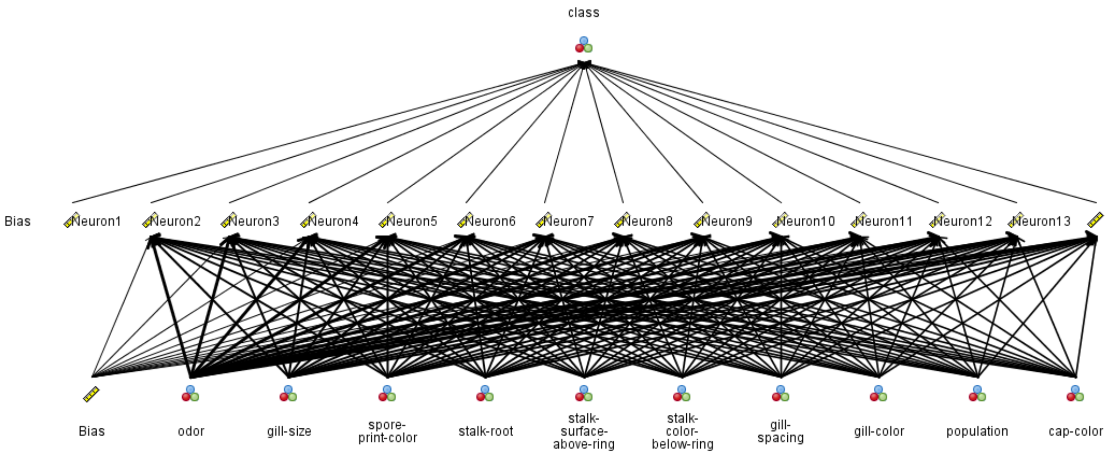
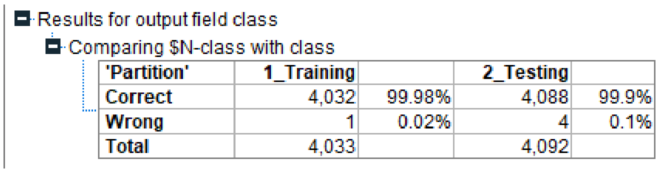


Figure 8 - IBM SPSS Neural network visualization



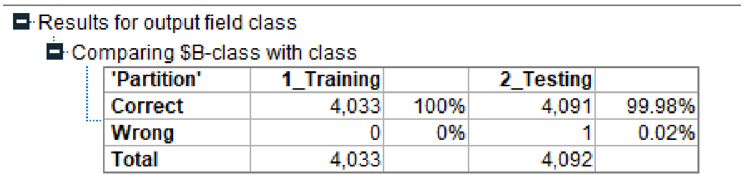
Results for output field class

Comparing $N-class with class

'Partition'	1_Training		2_Testing	
Correct	4,032	99.98%	4,088	99.9%
Wrong	1	0.02%	4	0.1%
Total	4,033		4,092	

4.1.2 Bayes Net

Second model, that can be used for modelling this dataset in SPSS is Bayes Net. The software tool gives interesting visualizations that can be further enhanced – as it is shown on figure 9.



Results for output field class

Comparing $B-class with class

'Partition'	1_Training		2_Testing	
Correct	4,033	100%	4,091	99.98%
Wrong	0	0%	1	0.02%
Total	4,033		4,092	

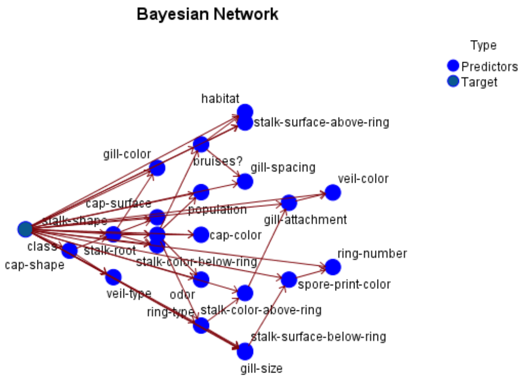


Figure 9 - IBM SPSS Bayes Net visualization

4.1.3 C5.0

The third model, that can be used for modelling dataset is C5.0 as it shown in figure 10.

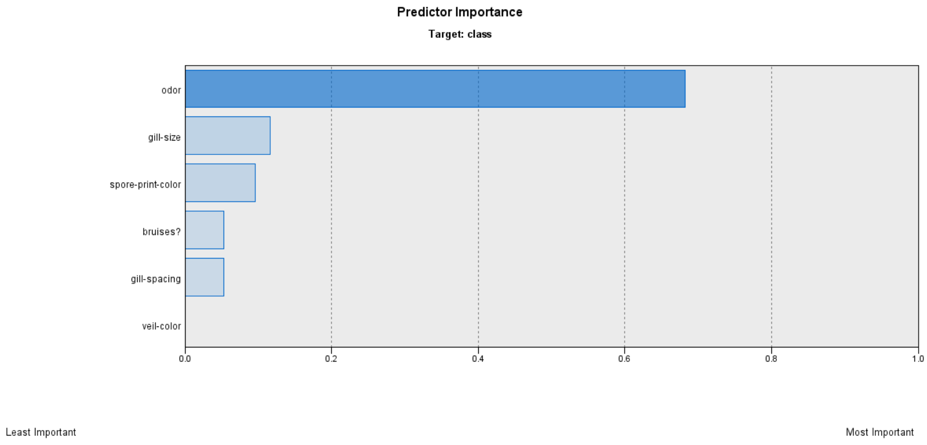
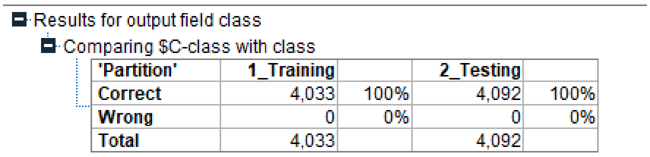


Figure 10 - IBM SPSS C5.0 visualization



Results for output field class

Comparing $C-class with class

'Partition'	1_Training		2_Testing	
Correct	4,033	100%	4,092	100%
Wrong	0	0%	0	0%
Total	4,033		4,092	

4.2 Results

C5.0 algorithm showed the best result.

Algorithm	Correctly classified	Success rate
Neural Net	4088	99.9%
Bayes Net	4091	99.98%
C5.0	4092	100%

5 Conclusion

The aim of this project was fulfilled – the dataset mushroom has been analyzed by using 4 different data mining software tools as described in previous chapters.

It is noticeable that Regression algorithm which was used in Rapid Miner and SaS Enterprise Miner showed 100% result. I would recommend SAS Enterprise Miner to other data analysts if the summary of the data should be provided immediately. Because using this tool it will take few minutes. But in case of working on data more rigorous then I would suggest the Rapid Miner. Because there is a lot of tools and operators.

6 References

[1] "Udemy.com," DATAhill Solutions Srinivas Reddy, 3 2019. [Online]. Available: https://www.udemy.com/course/weka-data-mining-with-open-source-machine-learning-tool/. [Accessed 14 12 2019].

[2] SUNIL RAY, "analyricsvudhya.com," 11 09 2017. [Online]. Available: https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/. [Accessed 14 12 2019].

[3] "business-iq.net," 2 10 2015. [Online]. Available: https://business-iq.net/articles/1115-en-rapidminer-is-a-data-mining-tool-with-exceptional-performance?v=cloudtech. [Accessed 14 12 2019].

[4] "www-01.ibm.com," IBM United States Software, 15 09 2015. [Online]. Available: https://www-01.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/1/897/ENUS215-321/index.html&request_locale=en. [Accessed 14 12 2019].

[5] Abie Reifer, "searchbusinessanalytics.techtarget.com," DecisionWorx, 31 07 2017. [Online]. Available: https://searchbusinessanalytics.techtarget.com/feature/How-SAS-Enterprise-Miner-simplifies-the-data-mining-process. [Accessed 14 12 2019].

7 Table of figures