Data Mining In Practice

Lab 3

Silesian University of Technology

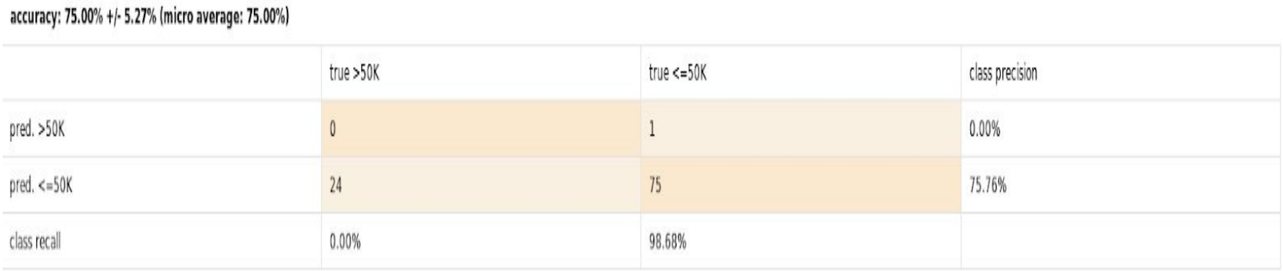
Anastas Iiti

Ankit Rathi

1. K- nearest Neighbour -- The K-nearest neighbour algorithm compare unknown example based on k training example(i.e the one who are nearest neighbour of the unknown example.

   It assumes that the similar things are nearby to one another.

Accuracy: 75.00% +/- 5.27% (micro average: 75.00%) - percentage of correct predictions
classification_error: 25.00% +/- 5.27% (micro average: 25.00%) - percentage of incorrect predictions.
kappa: 0.179 +/- 0.052 (micro average: 0.180) - The kappa statistics for the classification. It is generally thought to be a more robust measure than simple percentage correct prediction calculation since it takes into account the correct prediction occurring by chance.



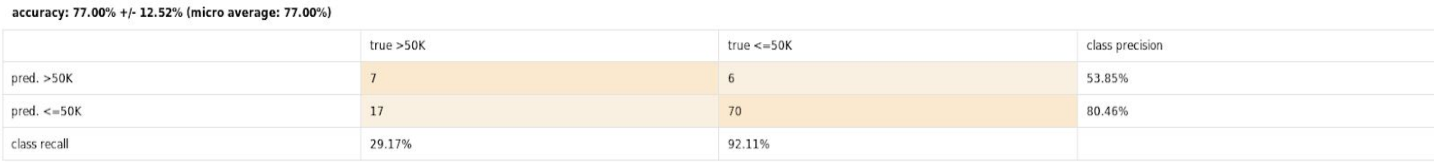
accuracy: 75.00% +/- 5.27% (micro average: 75.00%)

	true >50K	true <=50K	class precision
pred. >50K	0	1	0.00%
pred. <=50K	24	75	75.76%
class recall	0.00%	98.68%	

2. Naive Bayes -- The fundamental assumption of Naive Bayes is that, given the valueof the label (the class), the value of any Attribute is independent of the value of any other Attribute. Strictly speaking, this assumption is rarely true (it's "naive"!), but experience shows that the Naive Bayes classifier often works well. The independence assumption vastly simplifies the calculations needed to build the Naive Bayes probability model.

A Naive Bayes classifier is a probabilistic machine learning model that’s used for classification task. The crux of the classifier is based on the Bayes theorem.

$p(A|B) = P(B|A)/P(A)/P(B)$



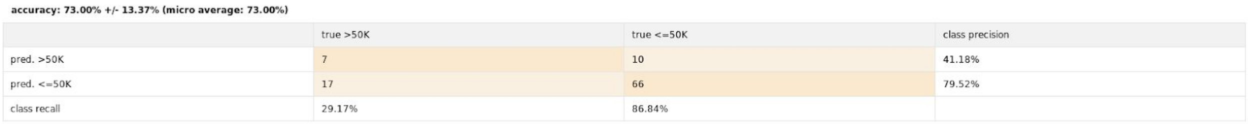
accuracy: 77.00% +/- 12.52% (micro average: 77.00%)

	true >50K	true <=50K	class precision
pred. >50K	7	6	53.85%
pred. <=50K	17	70	80.46%
class recall	29.17%	92.11%	

accuracy: 77.00% +/- 12.52% (micro average: 77.00%)

classification_error: 23.00% +/- 2.45% (micro average: 23.00%)
kappa: 0.509 +/- 0.069 (micro average: 0.509)

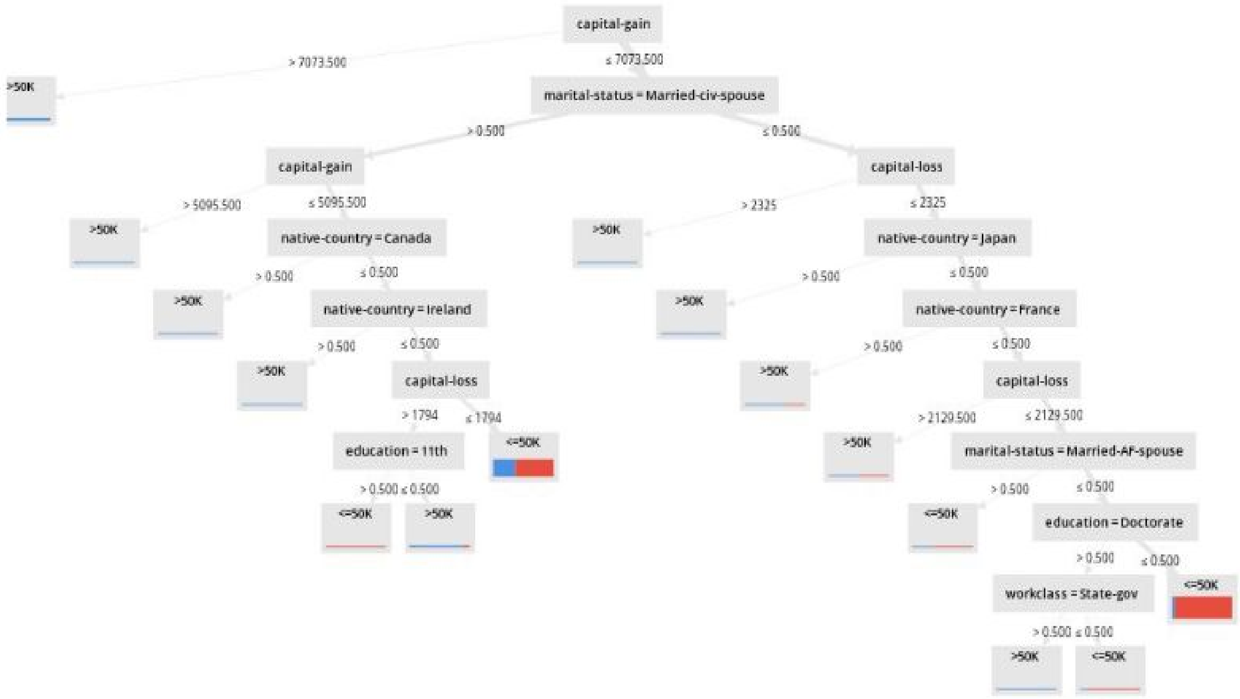
3. Decision Trees -- This Operator generates a decision tree model, which can be used for classification and regression.



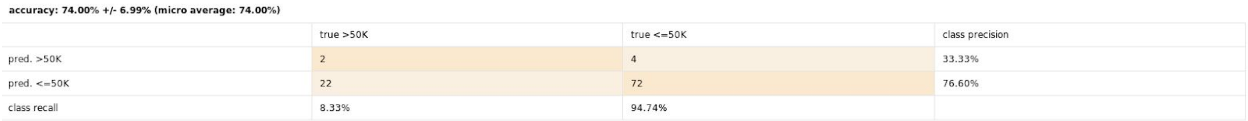
accuracy: 73.00% +/- 13.37% (micro average: 73.00%)

	true >50K	true <=50K	class precision
pred. >50K	7	10	41.18%
pred. <=50K	17	66	79.52%
class recall	29.17%	86.84%	

accuracy: 73.00% +/- 13.37% (micro average: 73.00%)
classification_error: 27.00% +/- 13.37% (micro average: 27.00%)
kappa: 0.411 +/- 0.135 (micro average: 0.422)



4. Rule Induction -- This operator learns a pruned set of rules with respect to the information gain from the given ExampleSet.

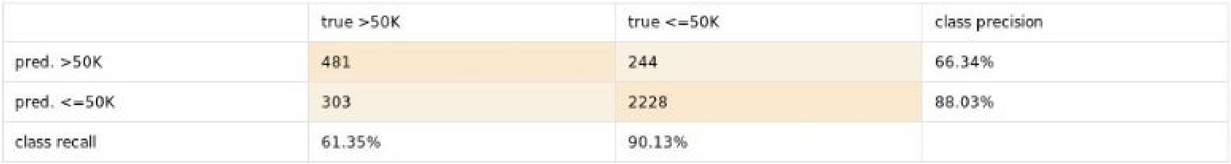


accuracy: 74.00% +/- 6.99% (micro average: 74.00%)

	true >50K	true <=50K	class precision
pred. >50K	2	4	33.33%
pred. <=50K	22	72	76.60%
class recall	8.33%	94.74%	

accuracy: 74.00% +/- 6.99% (micro average: 74.00%)
classification_error: 26.00% +/- 6.99% (micro average: 26.00%)
kappa: 0.456 +/- 0.067 (micro average: 0.457)

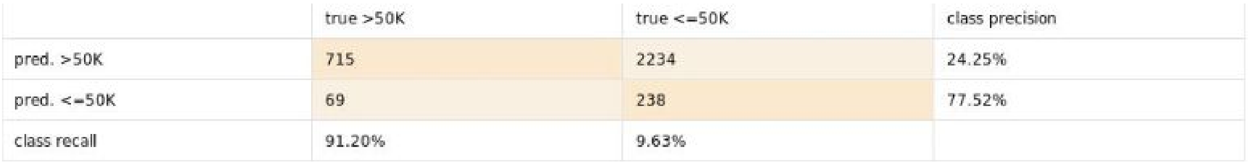
5. SVM -- According to the SVM algorithm we find the points closest to the line from both the classes.These points are called support vectors. Now, we compute the distance between thE Line and the support vectors. This distance is called the margin. Our goal is to maximize the margin. The hyperplane which the margin is maximum is the optimal hyperplane.



	true >50K	true <=50K	class precision
pred. >50K	481	244	66.34%
pred. <=50K	303	2228	88.03%
class recall	61.35%	90.13%	

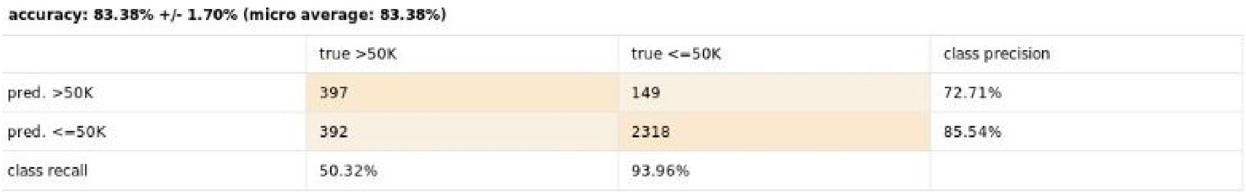
accuracy: 83.20% +/- 1.87% (micro average: 83.20%)
classification_error: 16.80% +/- 1.87% (micro average: 16.80%)
kappa: 0.528 +/- 0.055 (micro average: 0.528)

6. Artificial neural network -- This operator learns a linear classifier called Single Perceptron which finds separating hyperplane (if existent). This operator cannot handle polynominal attributes.
accuracy: 29.26% +/- 16.45% (micro average: 29.27%)
classification_error: 70.74% +/- 16.45% (micro average: 70.73%)
kappa: 0.012 +/- 0.038 (micro average: 0.004)



	true >50K	true <=50K	class precision
pred. >50K	715	2234	24.25%
pred. <=50K	69	238	77.52%
class recall	91.20%	9.63%	

7. Linear Regression --
accuracy: 83.38% +/- 1.70% (micro average: 83.38%)
classification_error: 16.62% +/- 1.70% (micro average: 16.62%)
kappa: 0.495 +/- 0.048 (micro average: 0.495)



accuracy: 83.38% +/- 1.70% (micro average: 83.38%)

	true >50K	true <=50K	class precision
pred. >50K	397	149	72.71%
pred. <=50K	392	2318	85.54%
class recall	50.32%	93.96%	

Method	Accuracy
k-Nearest Neighbours(knn)	75.00% +/- 5.27%
Naive Bayes	77.00% +/- 12.52%
Decision Trees	73.00% +/- 13.37%
Rule Induction	74.00% +/- 6.99%
SVM	83.20% +/- 1.87%
Artificial Neural Network	29.26% +/- 16.45%
Linear Regression	83.38% +/- 1.70%