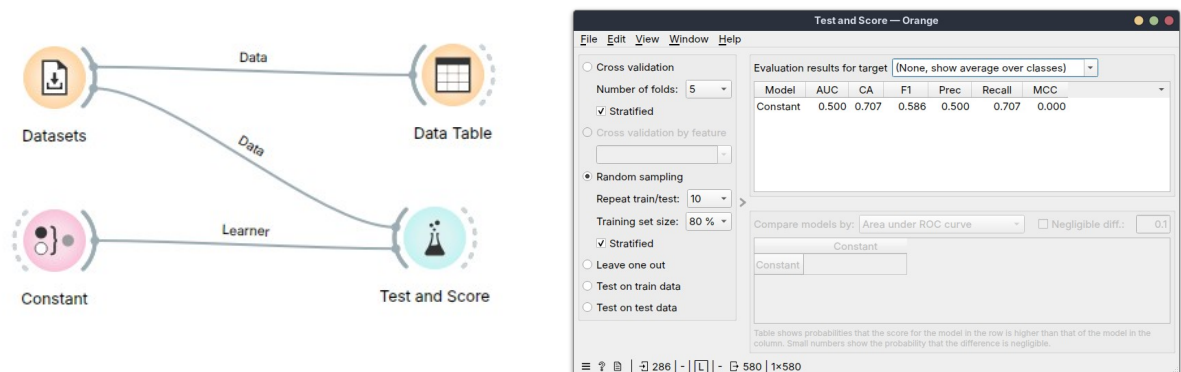


No code lab

Supervised learning

1. Use **Datasets** operator to load the *Breast cancer* dataset (since there are a few similar datasets, make sure you select the dataset with 286 instances). This breast cancer domain was obtained from the University Medical Centre, Institute of Oncology, Ljubljana, Slovenia. The detailed description of the dataset can be found here: <https://archive.ics.uci.edu/dataset/14/breast+cancer>
2. Send the data to the **Data Table** operator to get a first look at the data.
3. From [Model] section drag the **Constant** operator and put it inside the canvas. From [Evaluate] section drag the **Test and Score** operator and put it inside the canvas. **Constant** is the simplest possible classifier: it always predicts the most probable event (in this case, the recurrence of the tumor). Double-click on the **Test and Score** operator and make sure that you are using random sampling (left side of the window) with the *Training set size* set to 80%. As you can see, the classification accuracy (CA) is 70.7% (because this is exactly the percentage of cases with recurring tumor, all instances where no recurrence occurred are misclassified).



Hint: ask the instructor to explain what is cross validation, leave one out, and testing on train/test data.

Hint: ask the instructor to explain the metrics of precision, recall, and F1. Use the "Evaluation results for target" dropdown list to see the differences of these metrics for the two classes.

4. From [Model] select **CN2 Rule Induction** operator and send the dataset to the operator. Then, from [Visualize] section select **CN2 Rule Viewer** operator and send the output of the **CN2 Rule Induction** to **CN2 Rule Viewer**. Double-click on the **CN2 Rule Viewer** to see the simple rules discovered by the algorithm.

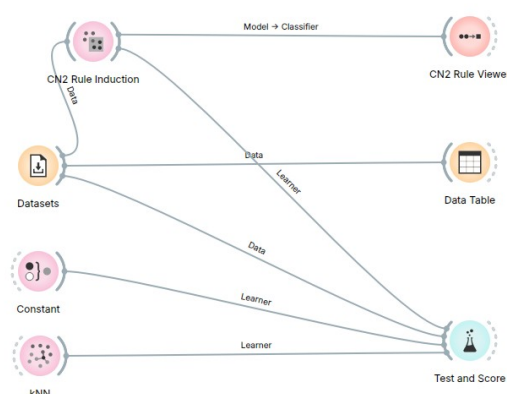
	IF conditions	THEN class	Distribution	Probabilities [%]	Quality	Length
0	age=20-29	recurrence=no-recurrence-events	[1, 0]	67 : 33	-0.00	1
1	tumor-size=5-9	recurrence=no-recurrence-events	[4, 0]	83 : 17	-0.00	1
2	tumor-size=10-14 AND age=40-49	recurrence=no-recurrence-events	[8, 0]	90 : 10	-0.00	2
3	tumor-size=10-14 AND age=50-59	recurrence=no-recurrence-events	[9, 0]	91 : 9	-0.00	2
4	tumor-size=10-14 AND age=60-69	recurrence=no-recurrence-events	[3, 0]	80 : 20	-0.00	2
5	tumor-size=0-4 AND age=30-39	recurrence=no-recurrence-events	[6, 0]	88 : 12	-0.00	2
6	tumor-size=10-14 AND menopause=ge40	recurrence=no-recurrence-events	[1, 0]	67 : 33	-0.00	2
7	tumor-size=10-14 AND inv-nodes=0-2	recurrence=recurrence-events	[0, 1]	33 : 67	-0.00	2
8	tumor-size=10-14	recurrence=no-recurrence-events	[6, 0]	88 : 12	-0.00	1
9	deg-malig=1 AND tumor-size=45-49	recurrence=recurrence-events	[0, 1]	33 : 67	-0.00	2
10	tumor-size=45-49	recurrence=no-recurrence-events	[2, 0]	75 : 25	-0.00	1
11	deg-malig=1 AND tumor-size=50-54	recurrence=no-recurrence-events	[1, 0]	67 : 33	-0.00	2
12	deg-malig=1 AND inv-nodes=3-5	recurrence=no-recurrence-events	[2, 0]	75 : 25	-0.00	2
13	deg-malig=1 AND inv-nodes=0-2	recurrence=recurrence-events	[0, 1]	33 : 67	-0.00	2

☒ Compact view
 Restore original order

28 | 286

The model applies the rules sequentially. First, it checks the age of the patient and if the the age is between 20 and 29, the model predicts no recurrence of the tumor (there is one such instance in the dataset). If the age is not within the range 20-29, the model checks the tumor size and if it is in the range 5-9 mm, again the model predicts no recurrence of the tumor. The procedure continues using next rules.

- Send the output of the **CN2 Rule Induction** to the **Test and Score** operator. Compare the quality of the new model with simple majority voting. What do you think is the biggest difference between these two models? (*hint: think about possible predictions of the model*)
- From [Model] section drag the **k-NN** operator and send it to the **Test and Score** operator. It is a simple algorithm which first finds k most similar instances (NN stands for nearest neighbors, the default value for k is 10), and classifies the current instance based on the majority voting among the neighbors. In other words, a patient will be classified as “recurrence-events” if the majority of the 10 most similar patients had “recurrence-events” as well. Double-click the **Test and Score** operator to see the accuracy metrics of all three classifiers. As you can see, our classification models are becoming better and better.

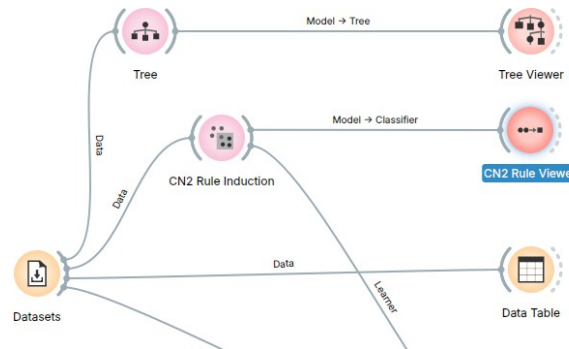


Evaluation results for target (None, show average over classes)							
Model	AUC	CA	F1	Prec	Recall	MCC	
Constant	0.500	0.707	0.596	0.500	0.707	0.000	
CN2 Rule Induction	0.578	0.647	0.644	0.641	0.647	0.134	
kNN	0.599	0.712	0.676	0.678	0.712	0.202	

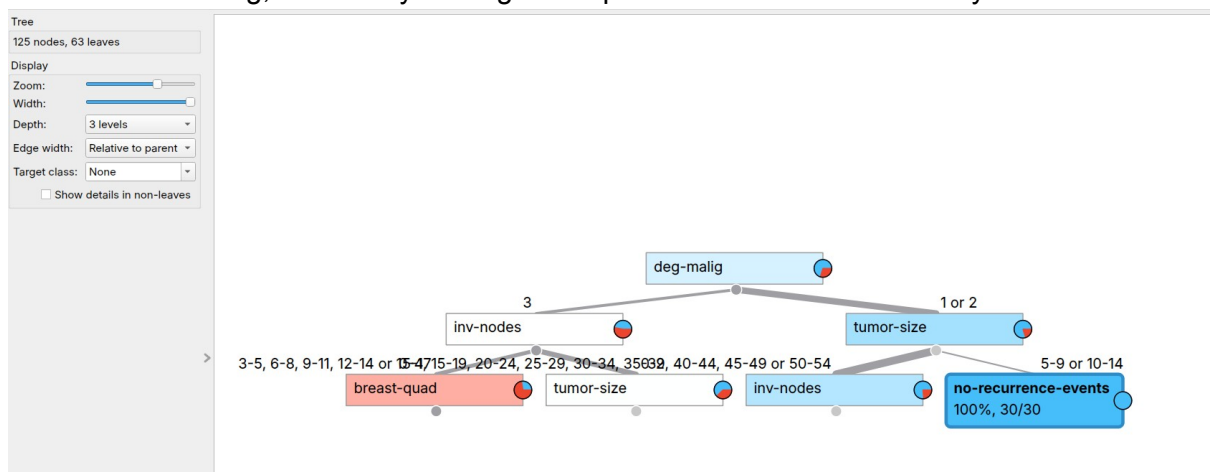
Compare models by: Area under ROC curve			
	Constant	CN2 Rule Induction	kNN
Constant			
CN2 Rule Induction			
kNN			

Table shows probabilities that the score for the model in the row is higher than that of the model in the column. Small numbers show the probability that the difference is negligible.

- Let us now turn to one of the most famous and most efficient classification methods: decision trees. In order to examine the structure of the tree, we will have to repeat the workflow that we have used to examine decision rules. First, from [Model] section drag the **Tree** operator. Send the data directly to the **Tree** operator, and send the output of the **Tree** operator to the **Tree Viewer** operator which you will find it in the [Visualize] section. This part of your workflow should be similar to the following:



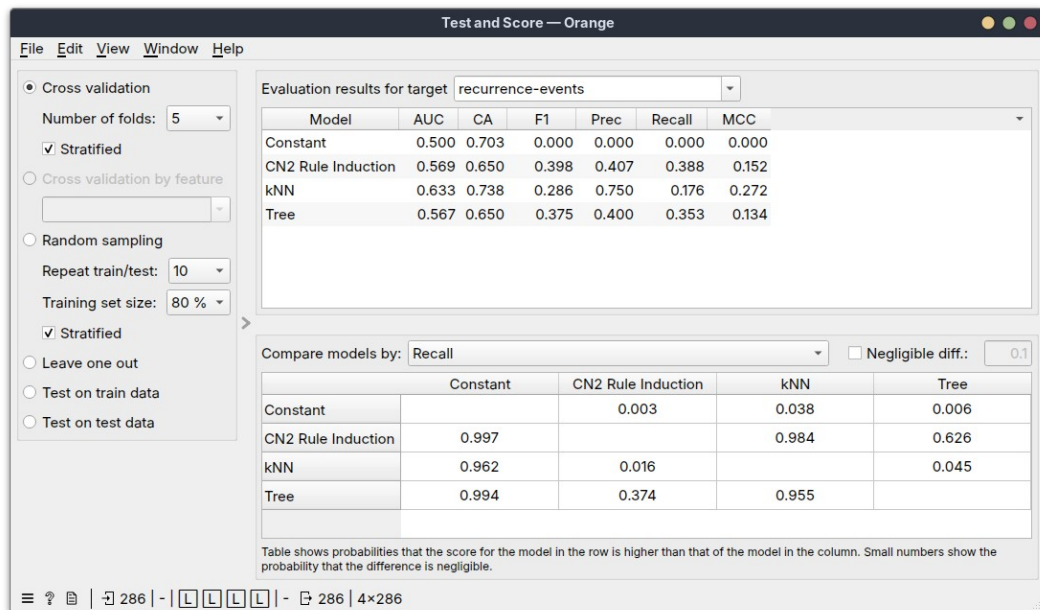
- Double-click on the **Tree Viewer** operator. The initial view of the tree may be too large and confusing, so start by limiting the depth of the tree to 3 levels only.



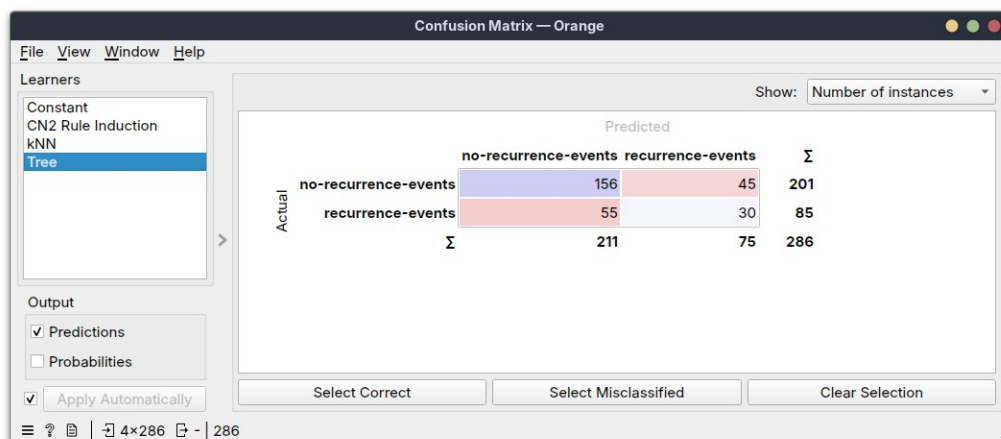
The tree consists of inner nodes (to the top) and leaves (at the bottom). The more blue a node is, the more it represents the part of the data where no-recurrence-events are present (i.e. the instances of patients who successfully completed the treatment). The little pie chart shows the distribution of recurrence/no-recurrence patients in each node. As we can see, the first decision the tree “makes” is based on the degree of malignancy, if it is 1 or 2, the patients travel to the right, if the degree of malignancy is 3, the patients travel to the left. Next, the tree checks for the tumor size, if it is between 5 and 14, the patient finishes in the right-most leaf (it has no further branches leading down), and the leaf consists of 30 patients who are all cured (100% no-recurrence-events). To see the full tree, either click on one of the gray dots to expand the tree down a particular branch, or remove the limit on the tree depth. In either case you may want to use Zoom and Width sliders in the left side of the window to better fit the tree to the window.

- Send the **Tree** operator to the **Test and Score** operator, double-click on the **Test and Score** and compare the effectiveness of the decision tree with previous models. Are we still getting a better model? How do results change if you switch from *Random Sampling*

to *Test on Train Data*? Observe, that if you use *Cross Validation*, the window will contain a new matrix comparing the models between each other. Each row says how probable it is that the given model will be better than the model described by a column. For instance, let us assume that I am most interested in the recall of the recurrence-events class (which means I want to make sure that my model finds as many instances of the recurring tumor as possible, and I don't care about false positives, i.e., making a mistake of predicting the recurrence of the tumor when in reality the patient is cured. The window below tells me that there is 96.2% chance that kNN will produce a better classifier than Constant, and only 37.4% chance that Tree model will be better than CN2 Rule Induction model.



- To better understand the structure of classification errors, drag the **Confusion Matrix** operator from the [Evaluate] section and connect its input to the output of the **Test and Score** operator. Double-click the **Confusion Matrix** to see the types of errors made by classifiers. Let us begin with the **Tree** model. The rows represent the true values from patient records, the columns represent model predictions. In the image below we can see that 156 patients with no-recurrence-events were correctly predicted as having no recurrence of the tumor, but 45 patients with no-recurrence-events were misclassified as having recurring tumor. The main diagonal (156+30) presents the correct model predictions, the red cells (55+45) represent mistakes of the classifier.



- Finally, drag the **Predictions** operator from the [Evaluate] section and send the results of the **Test and Score** operator to the **Predictions** operator. You will see the predictions and the probabilities of these predictions for all classifiers. Look at women who had menopause before the age of 40 and compare the predictions of different models for this group of patients.

The screenshot shows the 'Predictions - Orange' window. It contains a table with columns for classifiers (Constant, 2 Rule Induct, kNN, Tree, etc.), their performance metrics (Fold, age, menopause, tumor-size, inv-nodes), and a 'Target class' column. The table is sorted by menopause status, showing results for 'ge40' and 'premeno' groups. The bottom of the window shows 'Show performance scores' and 'Target class: (Average over classes)'.

Assignment

Use the **Datasets** operator to download the *Bank Marketing* dataset. Read the description of the dataset: <http://archive.ics.uci.edu/dataset/222/bank+marketing> Using learning tools try to do the following:

- Train a classifier to predict if a customer subscribed to a term deposit. Compare the performance of a few selected models (decision tree, k-nearest neighbors).
- Display the structure of the decision tree to interpret the algorithmic decisions made by the algorithm.
- Display the confusion matrix of a selected algorithm. Can you identify, what is the main weakness of the algorithm (false negatives or false positives)?