MEDNARODNA
PODIPLOMSKA ŠOLA
JOŽEFA STEFANA

INFORMATION AND COMMUNICATION TECHNOLOGIES
Master study programme

Data and Text Mining

Petra Kralj Novak

November 25, 2019

http://kt.ijs.si/petra_kralj/dmtm2.html

In previous episodes …

- 23-Oct-19
  - Data, data types
  - Interactive visualization (Orange)
  - Classification with decision trees (root, leaves, rules, entropy, info gain, TDIDT, ID3)
- 6-Nov-19
  - Classification: train – test (evaluate) - apply
  - Decision tree example (on blackboard)
  - Decision tree language bias (Orange workflow)
  - Homework:
    - InfoGain questions
    - Orange workflow
    - Reading “Classification and regression by randomForest”

Homework: InfoGain questions

- Construct an attribute with Information gain =1.
- Construct an attribute with Information gain =0.
- Compute the Information gain of the attribute “Person”.
- How would you compute the information gain of a numeric attribute.
- What would be the classification accuracy of the decision tree (on the previous slide) if we pruned it at the node „Astigmatic“?

Homework: Orange workflow

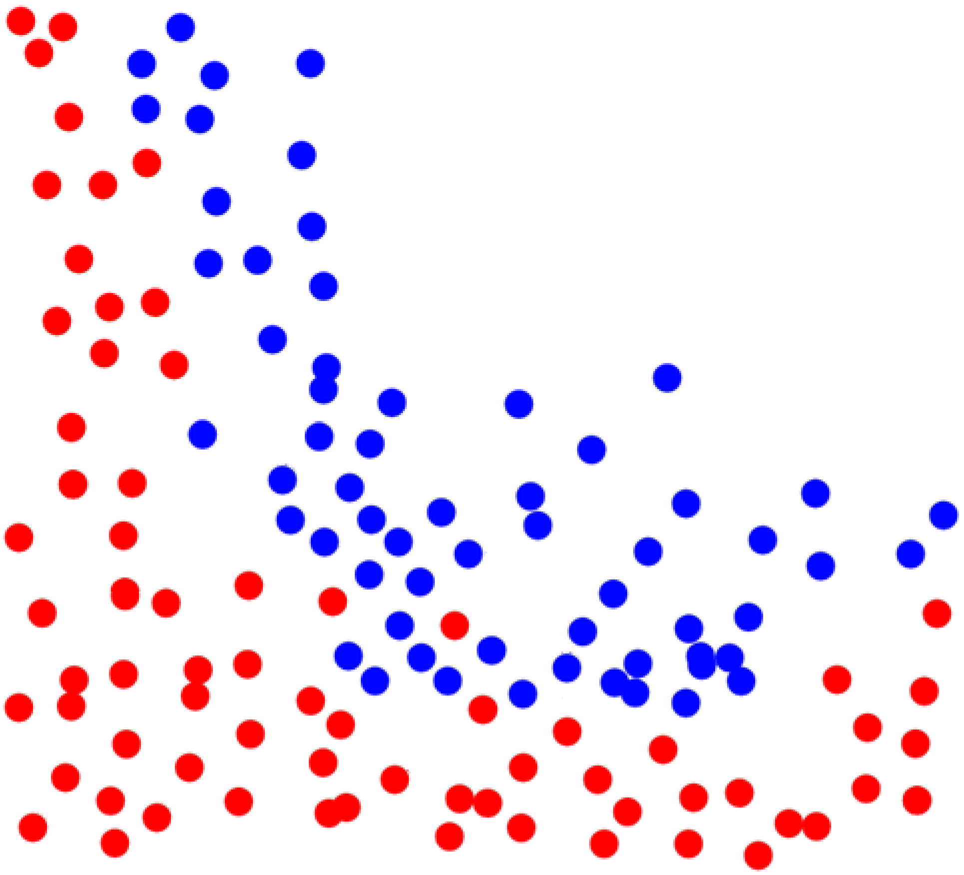
- Extend the workflow from the Lab exercise to use other ML algorithms:
  - Random forest
  - SVM with linear kernel
- Experiment with different random seeds (sample data with data sempler several times) and observe the stability of results of different algorithms in different runs.

Homework: Reading

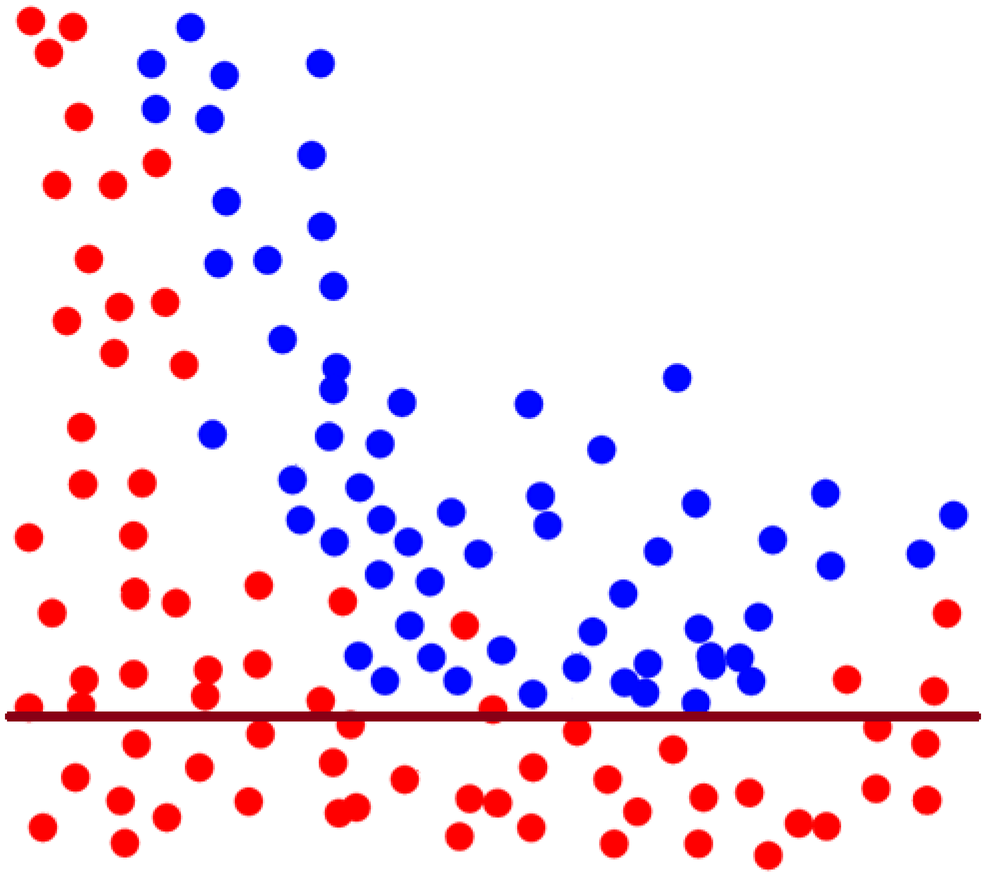
- Reading “Classification and regression by randomForest”
  - Ensemble learning: many classifiers and aggregate their results
  - Boosting
  - Bagging
  - Random forests
    - Bootstrap sample of the data
    - n_{tree} , m_{try}
    - Majority vote
    - OOB data, Out-of-bag
    - OOB estimate of the error rate
    - Variable importance
    - Proximity measure

Liaw, Andy, and Matthew Wiener: "Classification and regression by randomForest" R news 2.3 (2002): 18-22.

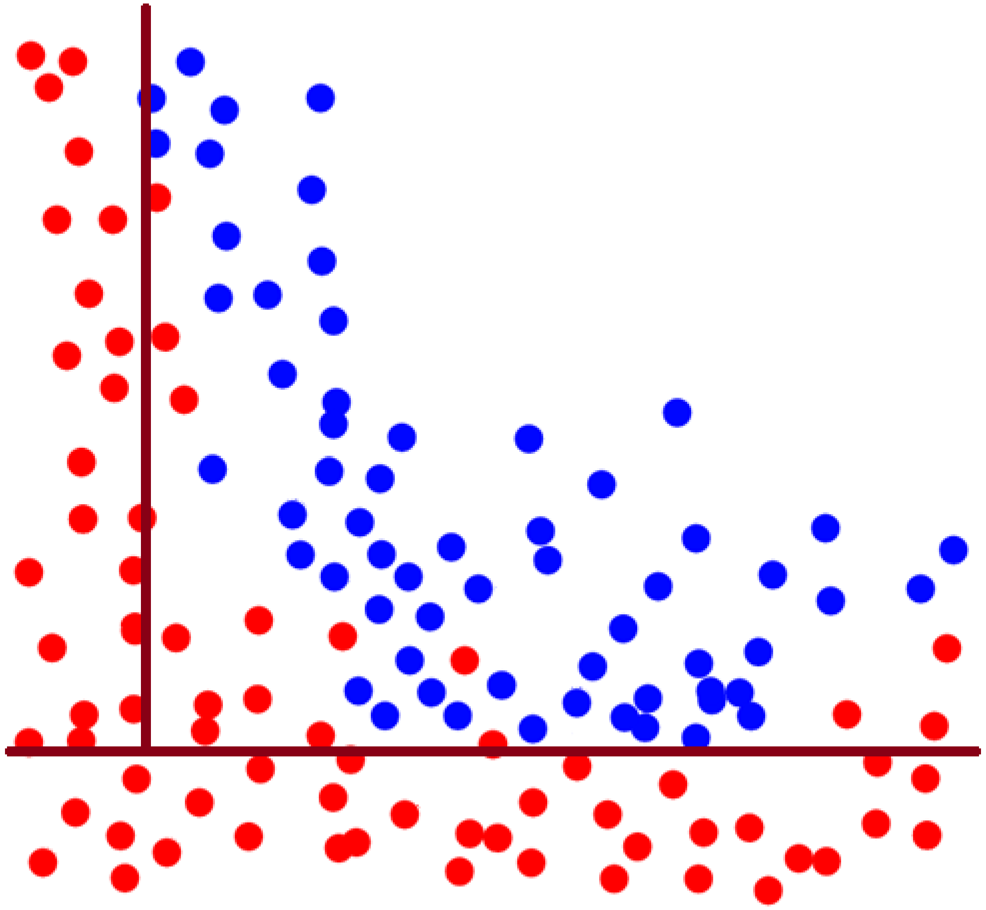
Separate the blue from the red



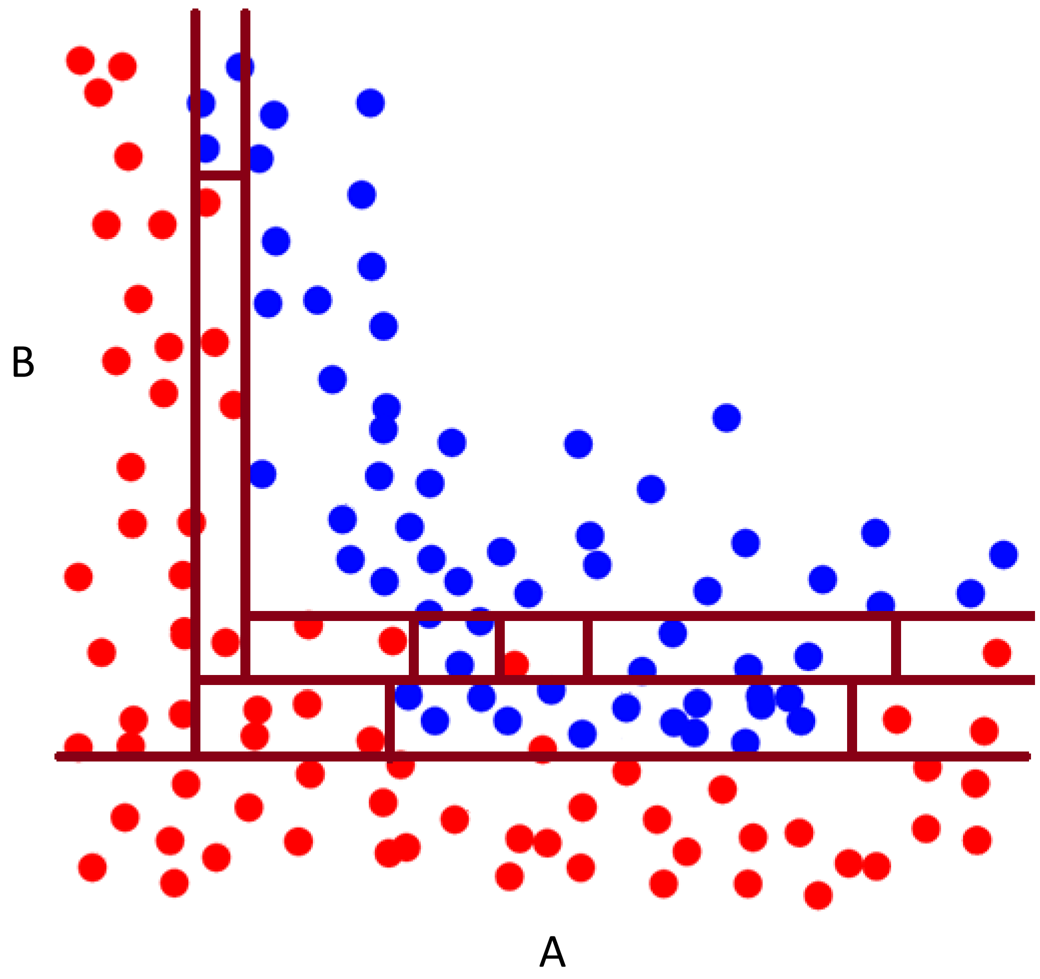
Decision trees …



Decision trees ...

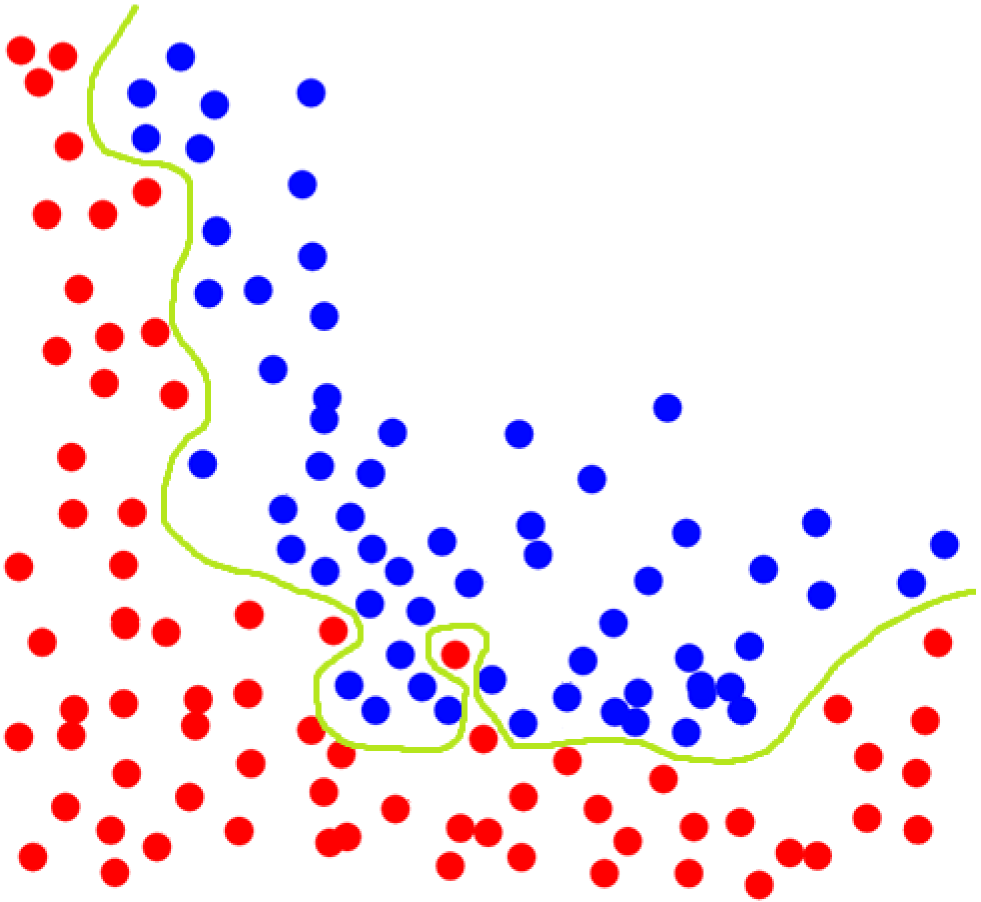


Decision trees …

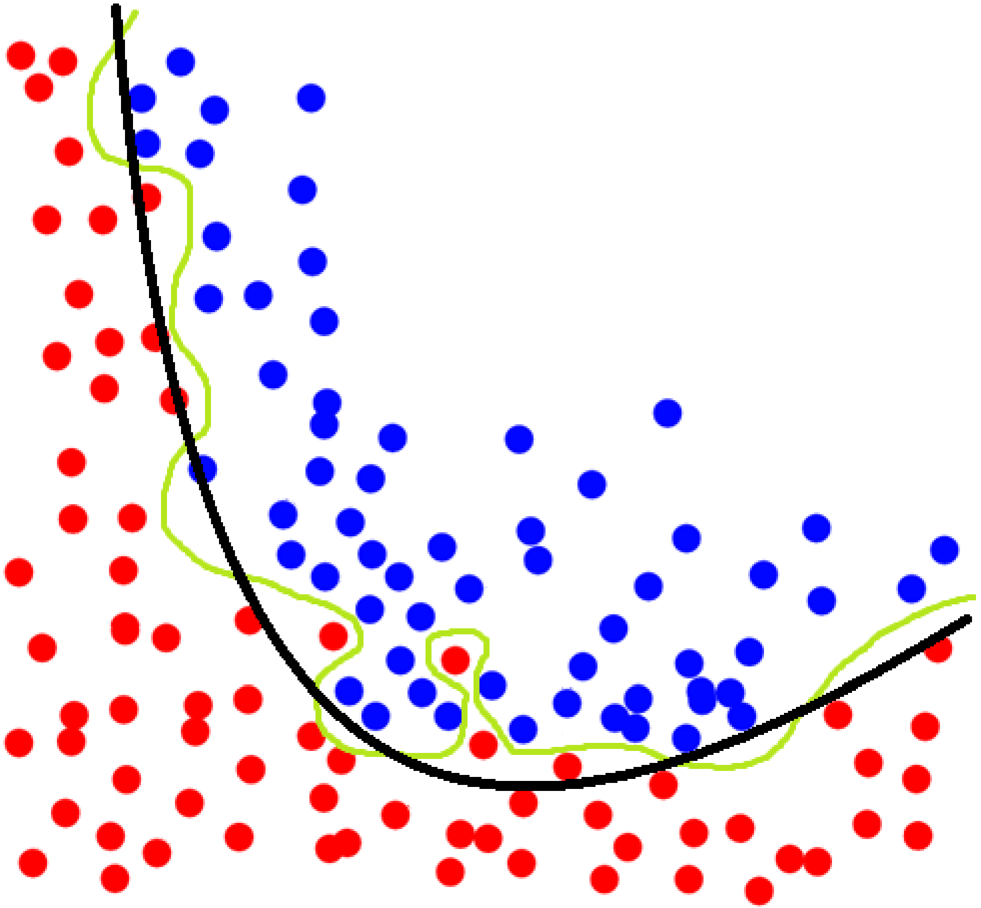


- Jezikovna pristranskost
  - Odločitvena drevesa imajo samo pogoje, kjer attribute primerjajo s konstantami (Samo vodoravne in navpične delitve, npr A > 1/4)
  - Odločitvena drevesa nimajo pogojev tipa A>B
- Ta model se pretirano prilagaja učni množici

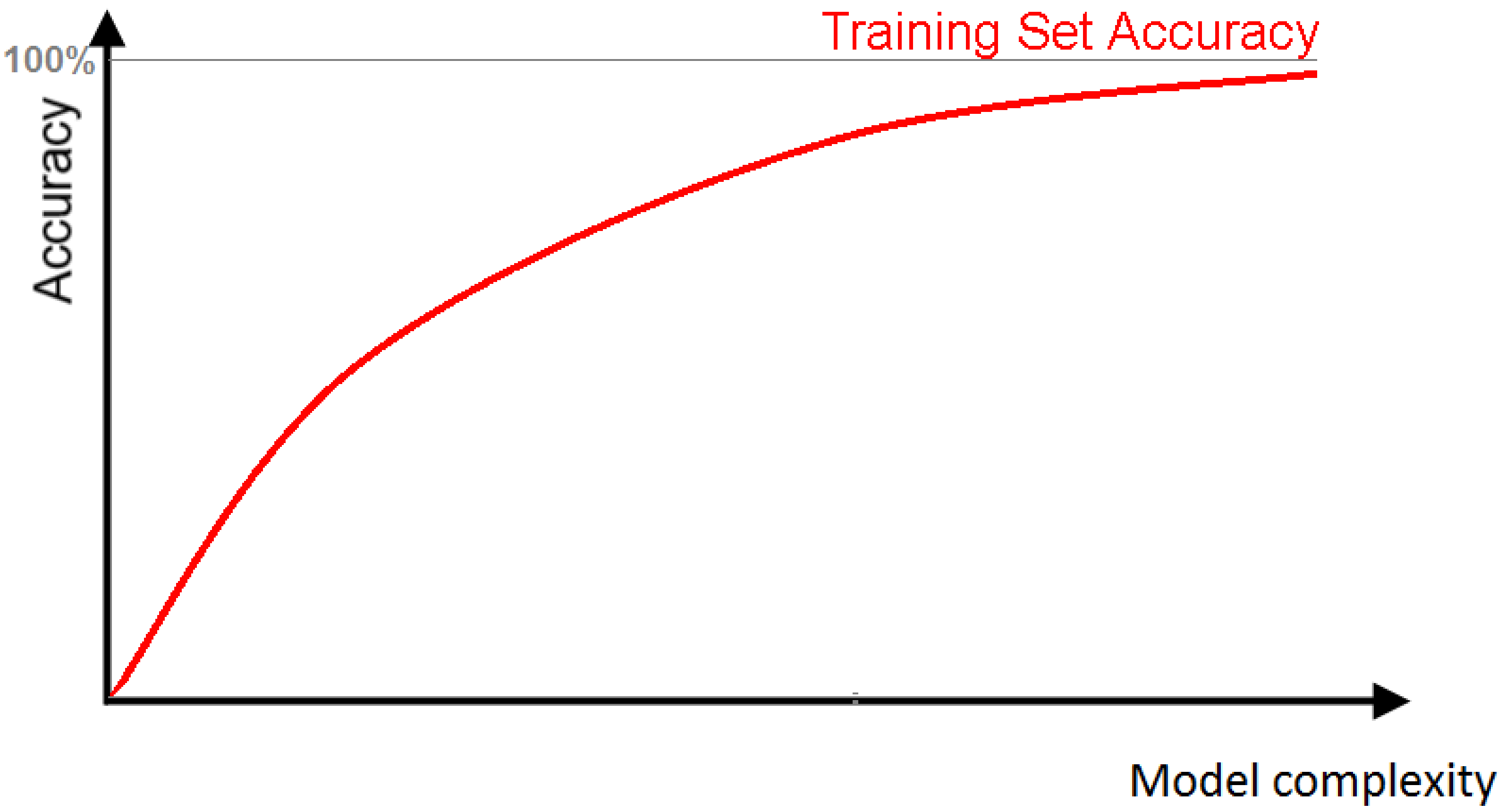
Other models overfit as well (e.g. SVM)



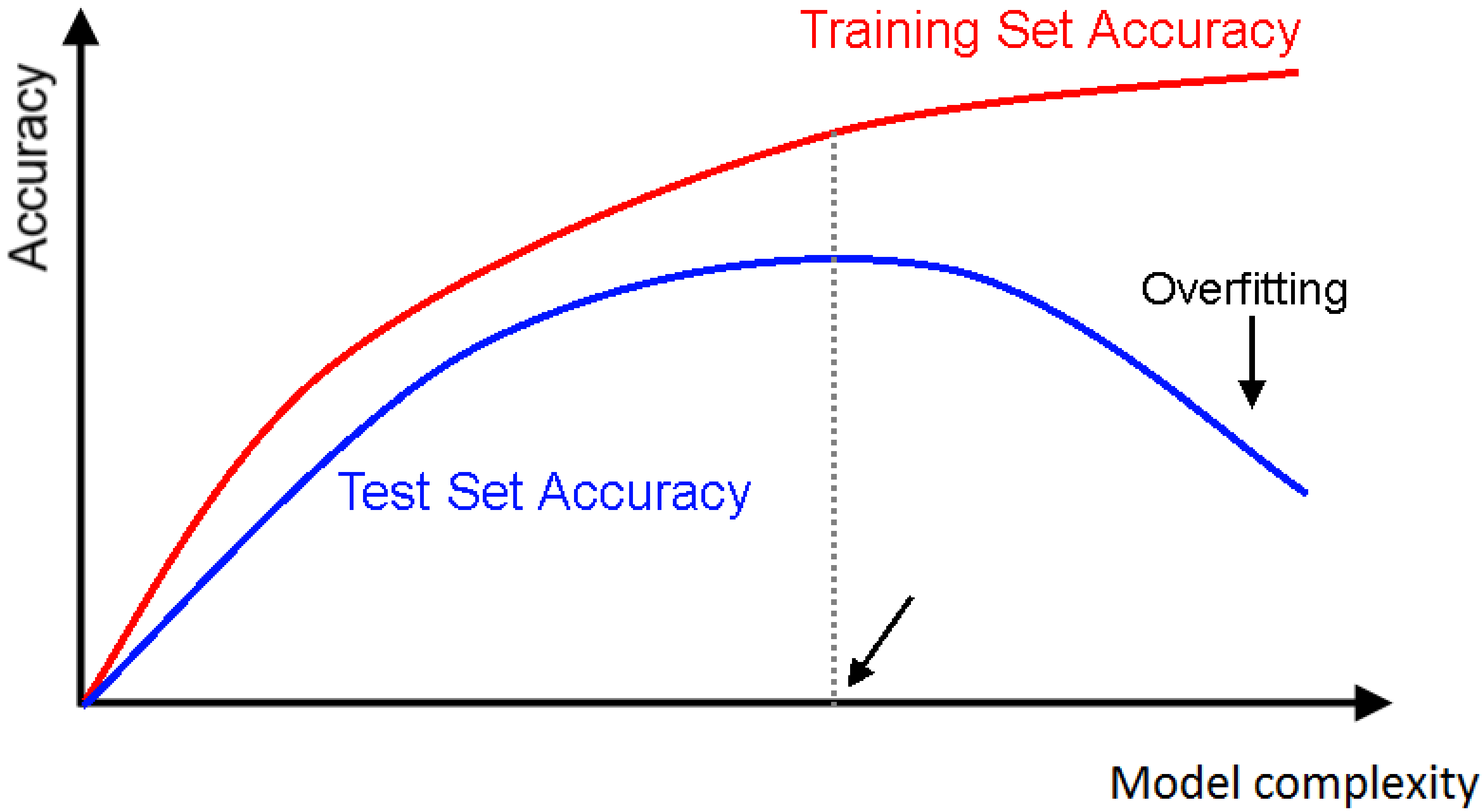
Other models overfit as well (e.g. SVM)



Model complexity and performance



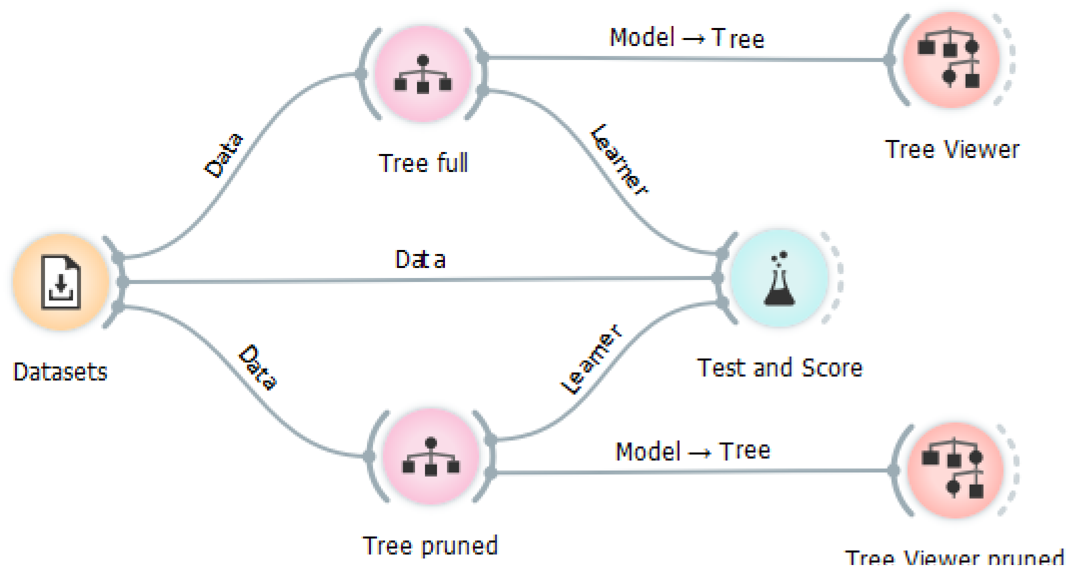
Performance on test set



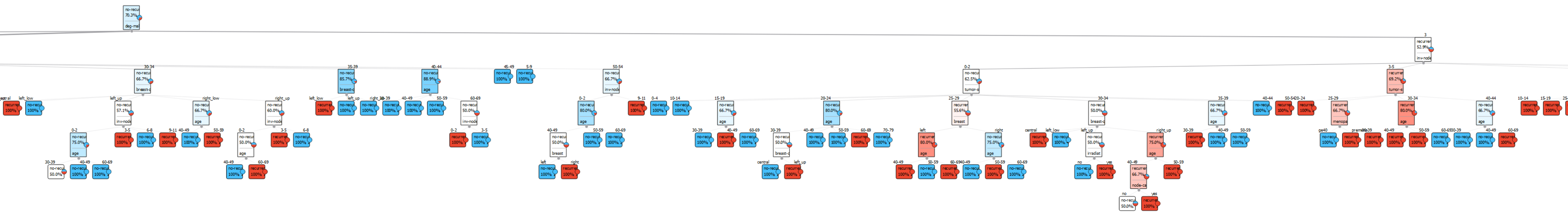
With training, the model fits to the training data

- Overfitting – the model fits to the noise in the data
- With regularization (e.g. decision tree pruning) we get a model that performs better on new data instances

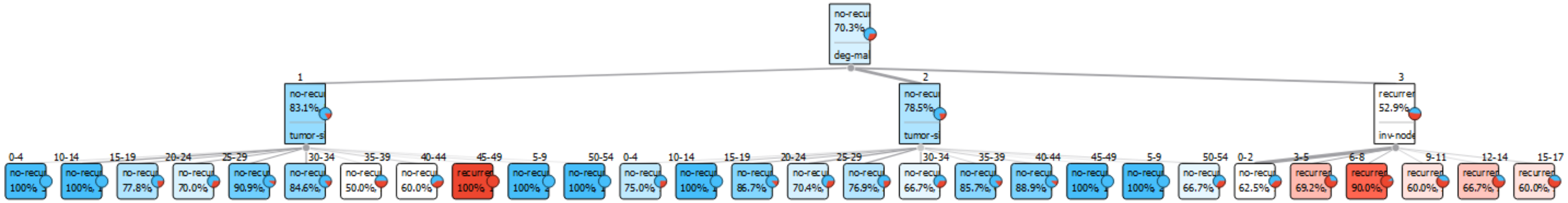
Overfitting example



- Dataset: Breast Cancer (1992)
- Full tree … CA = 0.661

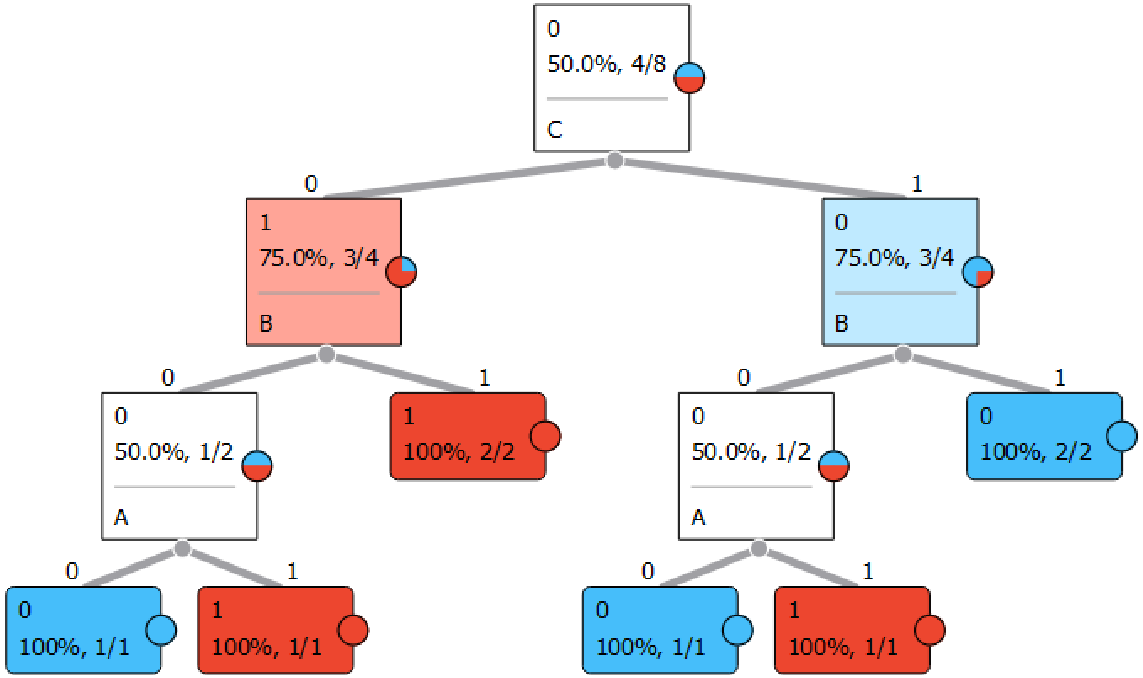


- Pruned tree (two levels) … CA = 0.710



Short-sightedness of decision trees

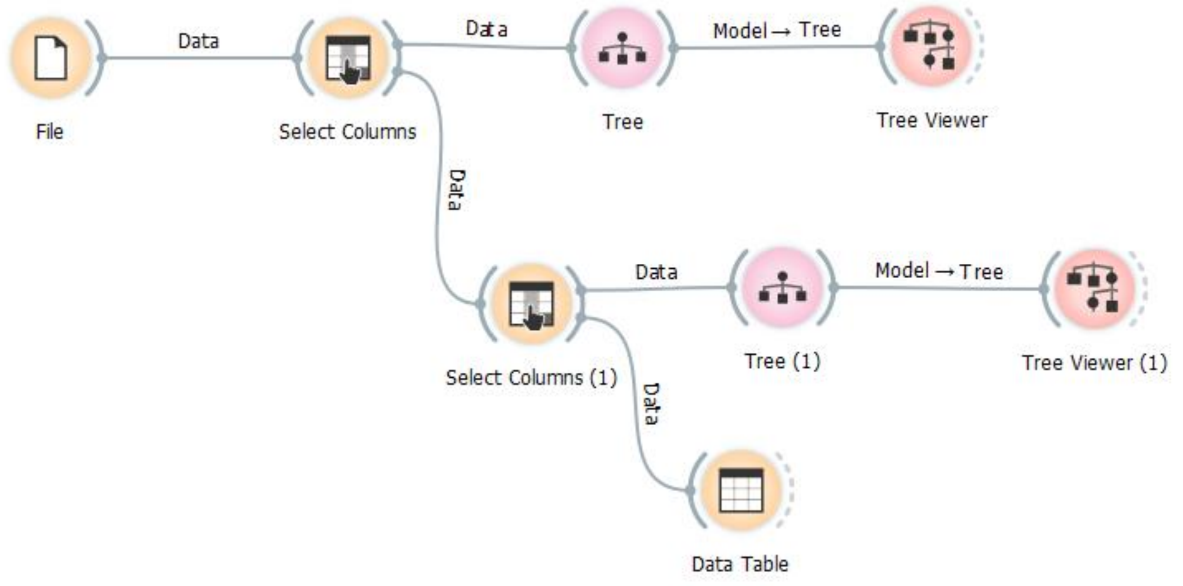
A	B	C	AxorB
1	1	1	0
1	1	1	0
1	0	1	1
1	0	0	1
0	1	0	1
0	1	0	1
0	0	1	0
0	0	0	0



Homework

1. Sketch the real decision tree model behind the data of the XOR example.
2. What happens if we remove the attribute “C”? Guess first, then use an Orange workflow and find out.

A	B	C	AxorB
1	1	1	0
1	1	1	0
1	0	1	1
1	0	0	1
0	1	0	1
0	1	0	1
0	0	1	0
0	0	0	0



!!!

Evaluation

How good is the model

Evaluation goal

- How good is the model

- Method
  - HOW we measure

- Measure
  - WHAT me measure

Test on a separate test set

Person	Age	Prescription	Astigmatic	Tear_Rate	**Lenses**
P1	young	myope	no	normal	**YES**
P2	young	myope	no	reduced	**NO**
P3	young	hypermetrope	no	normal	**YES**
P4	young	hypermetrope	no	reduced	**NO**
P5	young	myope	yes	normal	**YES**
P6	young	myope	yes	reduced	**NO**
P7	young	hypermetrope	yes	normal	**YES**
P8	young	hypermetrope	yes	reduced	**NO**
P9	pre-presbyopic	myope	no	normal	**YES**
P10	pre-presbyopic	myope	no	reduced	**NO**
P11	pre-presbyopic	hypermetrope	no	normal	**YES**
P12	pre-presbyopic	hypermetrope	no	reduced	**NO**
P13	pre-presbyopic	myope	yes	normal	**YES**
P14	pre-presbyopic	myope	yes	reduced	**NO**
P15	pre-presbyopic	hypermetrope	yes	normal	**NO**
P16	pre-presbyopic	hypermetrope	yes	reduced	**NO**
P17	presbyopic	myope	no	normal	**NO**
P18	presbyopic	myope	no	reduced	**NO**
P19	presbyopic	hypermetrope	no	normal	**YES**
P20	presbyopic	hypermetrope	no	reduced	**NO**
P21	presbyopic	myope	yes	normal	**YES**
P22	presbyopic	myope	yes	reduced	**NO**
P23	presbyopic	hypermetrope	yes	normal	**NO**
P24	presbyopic	hypermetrope	yes	reduced	**NO**

30% of examples are (randomly) selected for testing

Method: Test on a separate test set

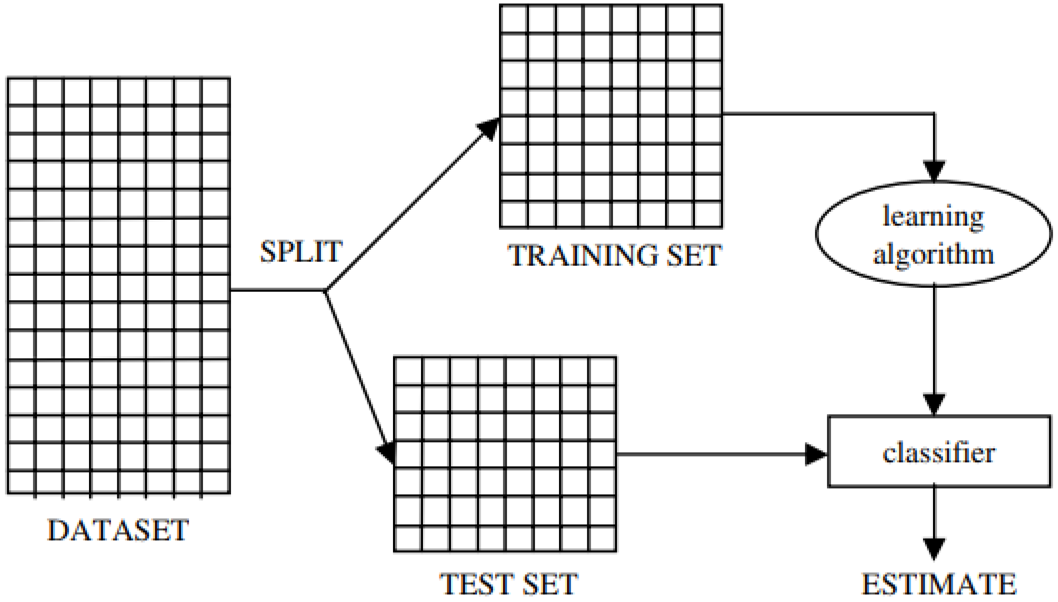


Figure from M. Bramer: [Principles of Data Mining (2007)]

Stratified sampling

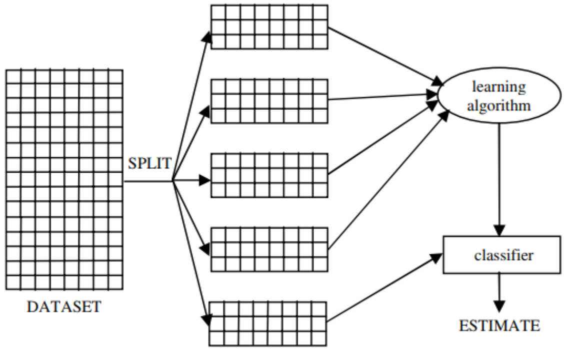
- Stratified sampling aims at splitting one data set so that each split are similar with respect to the target variable distribution.

Method: Random sampling

- Repeat several times „Test on a separate test set“ with different test set selections
- Compute the mean, variance on the results …
- The evaluation is more robust as it does not depend on a single random split

Method: *K*-fold cross validation

- Most commonly used in machine learning
- Split the dataset into *k* (disjunctive) subsets
- Repeat *k*-times:
    - Use a different subset for testing
    - Use all the other data for training
- Each example is in the test set just once



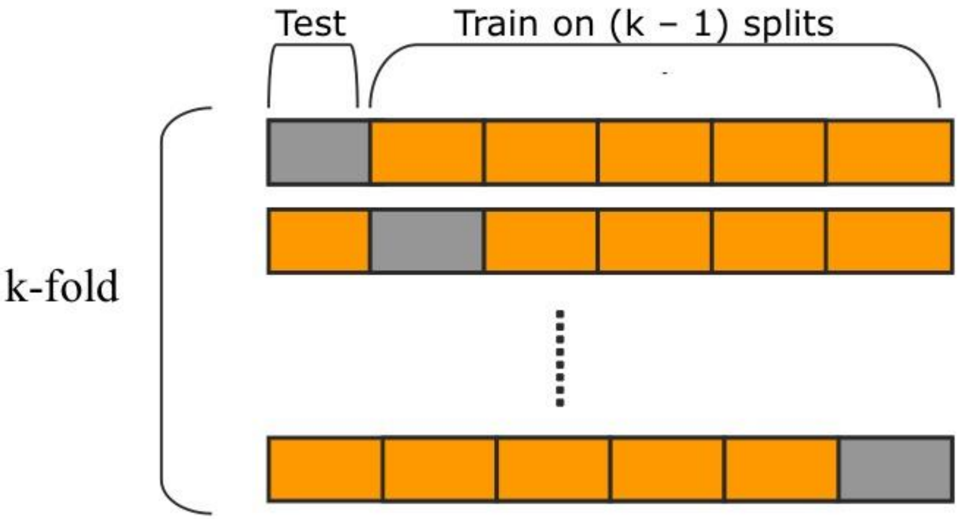
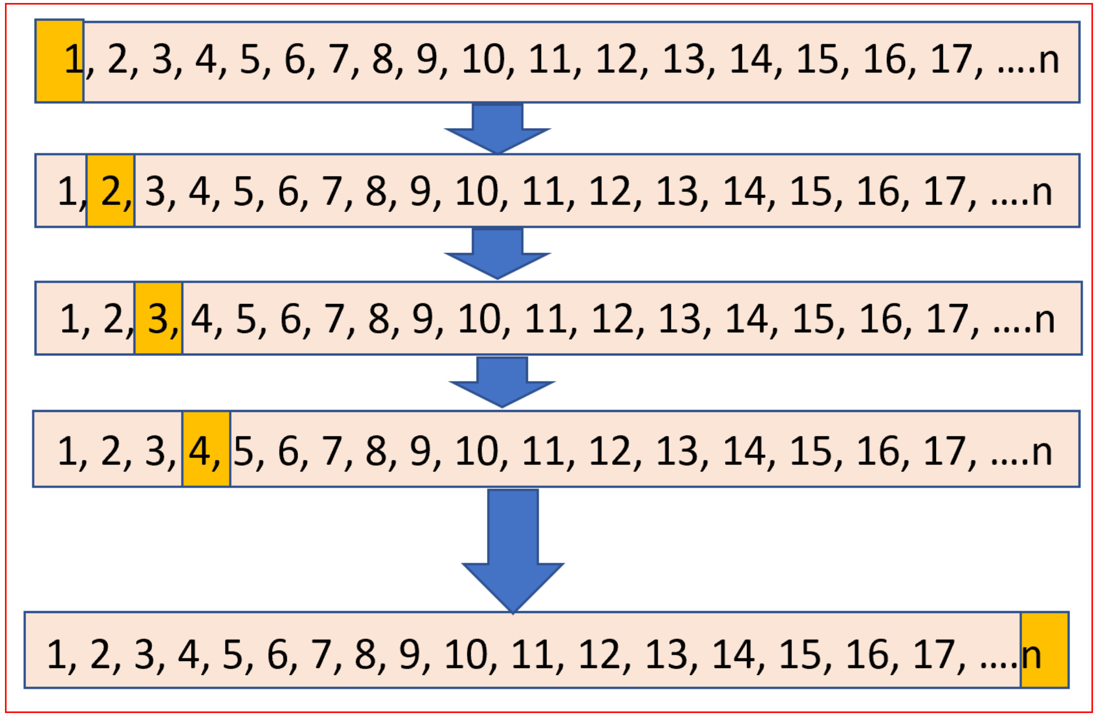


Figure from M. Bramer: [Principles of Data Mining (2007)]

Method: Leave one out (N-fold cross-validation)

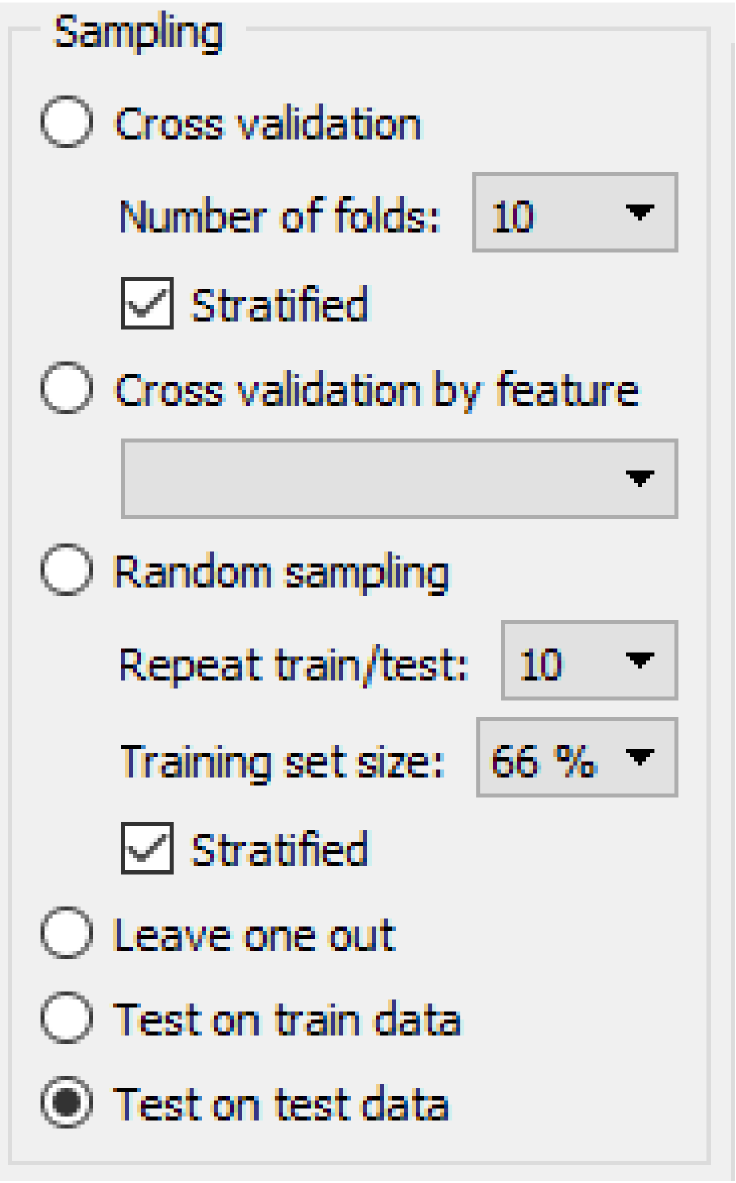
- For small datasets
- Similar to cross validation with test set size =1
- Repeat the training *N*-times if there is *N* examples in the dataset



Evaluation methods in Orange


Test & Score

- Cross validation
- Random sampling
- Leave one out
- Test on train data
- Test on test data



Questions

1. What do we get when testing on the training set?
2. Can we always get a 100% accuracy on the training set?
3. When do we use “leave-one-out”?
4. What is stratified sampling?

Classification quality measures

Confusion matrix (error matrix)

Breakdown of the classifier's performance, i.e. how frequently instances of class X were correctly classified as class X or misclassified as some other class.

Dataset: titanic

Actual \ Predicted	no	yes	Σ
no	1364	126	1490
yes	362	349	711
Σ	1726	475	2201

Dataset: car

Actual \ Predicted	unacc	acc	good	v-good	Σ
unacc	1154	54	2	0	1210
acc	94	276	14	0	384
good	0	44	22	3	69
v-good	0	25	0	40	65
Σ	1248	399	38	43	1728

Confusion matrix

- Matrix of correct and incorrect classifications
  - Rows are actual values
  - Columns are predicted values
  - Correct classifications are on the diagonal
  - We see what kind of mistakes does the classifier make.
  - If the classes are ordered, the errors far from the diagonal are heavier

Actual \ Predicted	unacc	acc	good	v-good	Σ
unacc	1154	54	2	0	1210
acc	94	276	14	0	384
good	0	44	22	3	69
v-good	0	25	0	40	65
Σ	1248	399	38	43	1728

Confusion matrix for two classes

The class we are interested in (e.g. fraud cases vs. normal, cancer patients vs. normal) is the „positive“ class.

		Predicted	
	Correct classification	Classified as	
		+	−
Actual	+	true positives	false negatives
	−	false positives	true negatives

TP: true positives
The number of positive instances that are classified as positive

FP: false positives
The number of negative instances that are classified as positive

FN: false negatives
The number of positive instances that are classified as negative

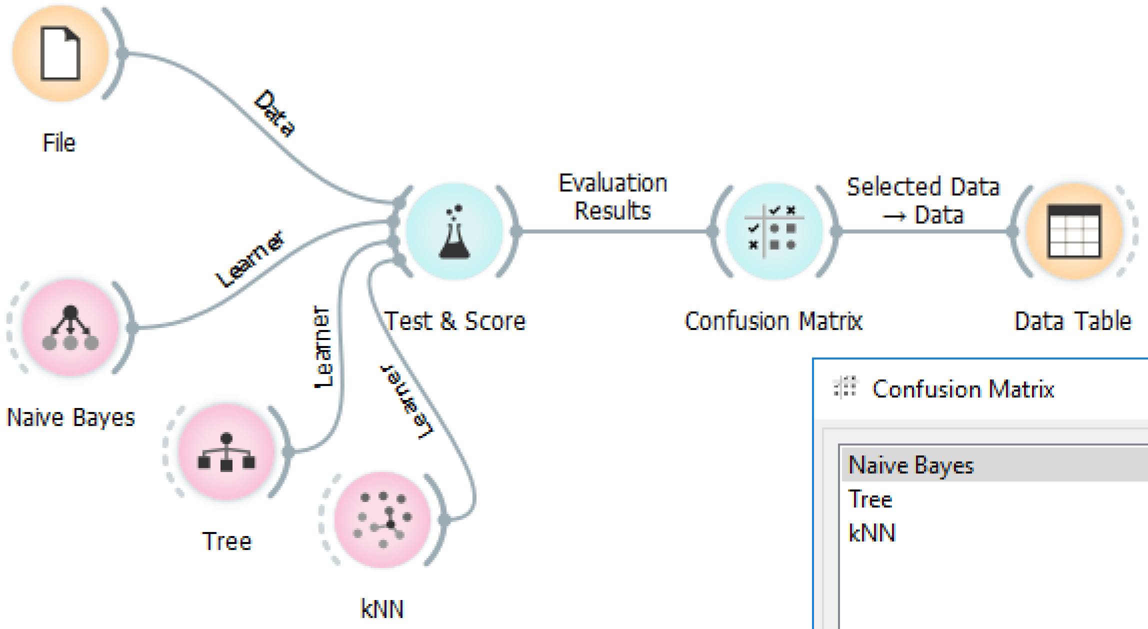
TN: true negatives
The number of negative instances that are classified as negative

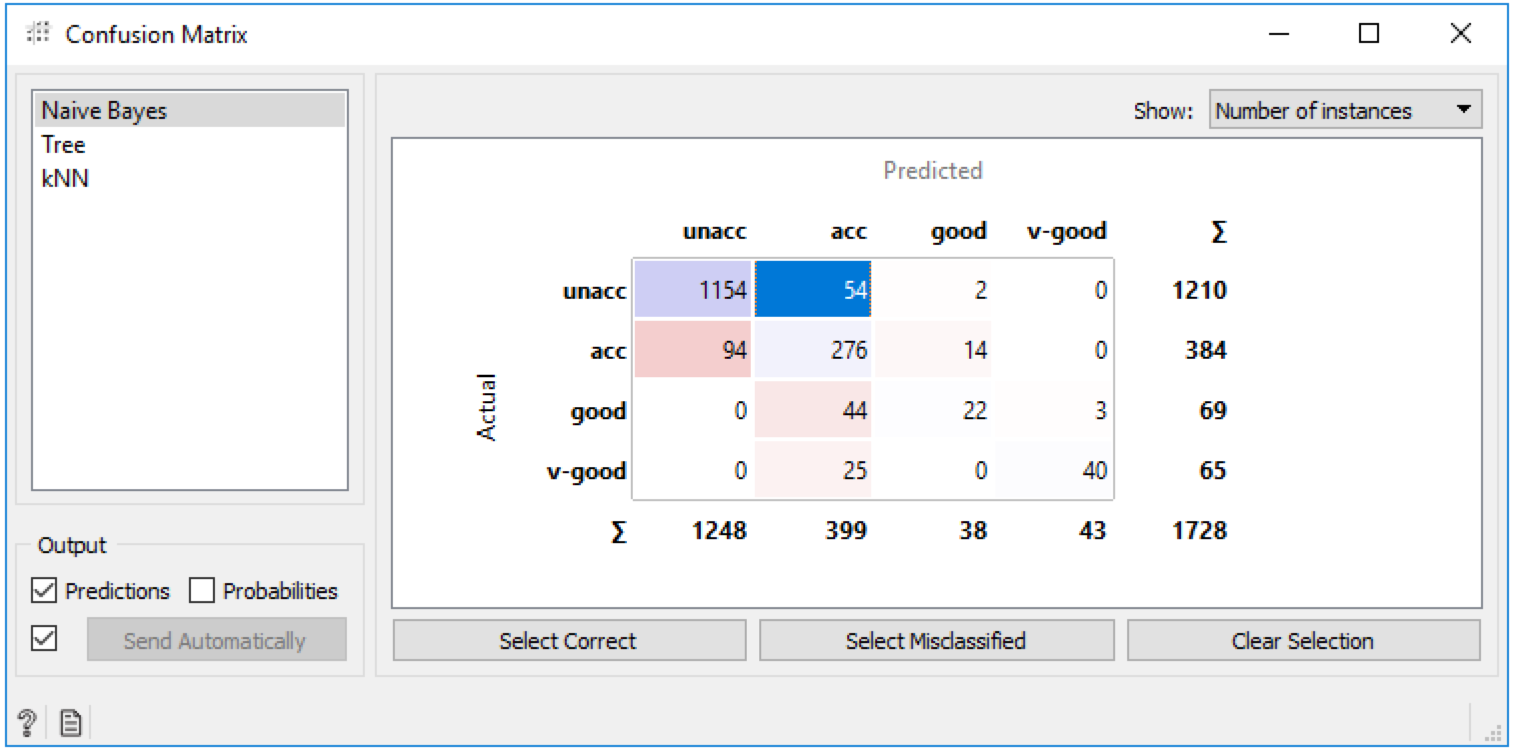
- Diagonal: correct classifications
- Outside: misclassifications
- Classification accuracy =

= |correct classifications| / |all examples|

= |correct classifications| / (|correct classifications| + |misclassifications|)

In Orange, the confusion matrix is interactive





Classification accuracy

- Percentage of correctly classified examples

Classification accuracy =
= |correct classifications| / |all examples|
= |correct classifications| / (|correct classifications| + |misclassifications|)

Exercise: Confusion matrix

Titanic

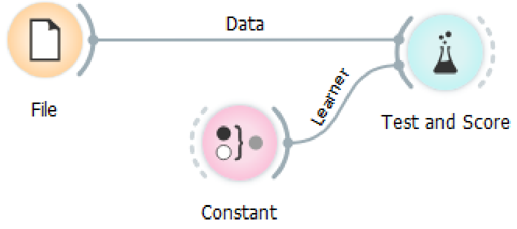
Actual \ Predicted	no	yes	Σ
no	1364	126	1490
yes	362	349	711
Σ	1726	475	2201

Car

Actual \ Predicted	unacc	acc	good	v-good	Σ
unacc	1154	54	2	0	1210
acc	94	276	14	0	384
good	0	44	22	3	69
v-good	0	25	0	40	65
Σ	1248	399	38	43	1728

	Titanic	Car
Number of examples		
Number of classes		
Number of examples in each class		
Number of examples classified in individual classes		
Number of misclassified examples		
Classification accuracy		

Majority class classifier (Constant)



Predicted

Actual		unacc	acc	good	v-good	Σ
	unacc	1154	54	2	0	1210
	acc	94	276	14	0	384
	good	0	44	22	3	69
	v-good	0	25	0	40	65
	Σ	1248	399	38	43	1728

Predicted

Actual		no	yes	Σ
	no	1364	126	1490
	yes	362	349	711
	Σ	1726	475	2201

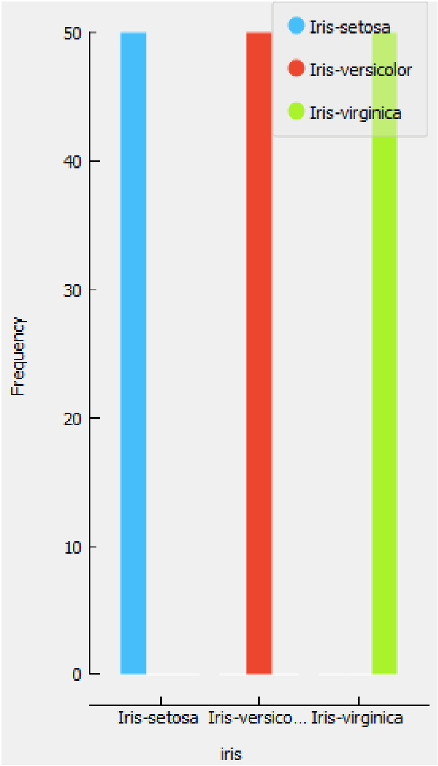
- What is the classification accuracy of a classifier that classifies all the examples in the majority class?
- Car: 70% Titanic: 68%

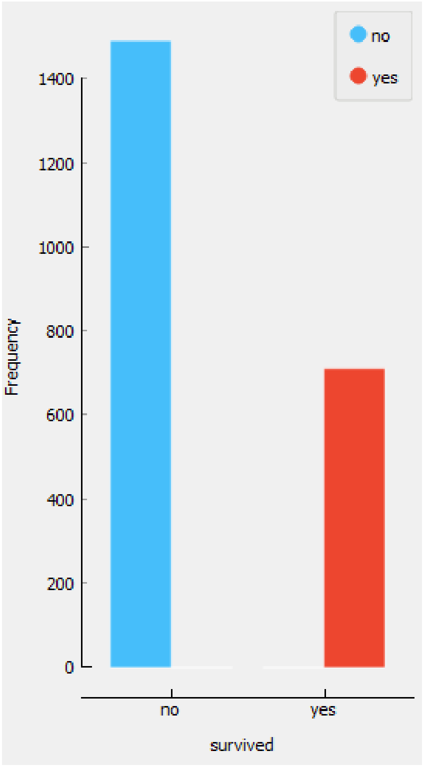
Question

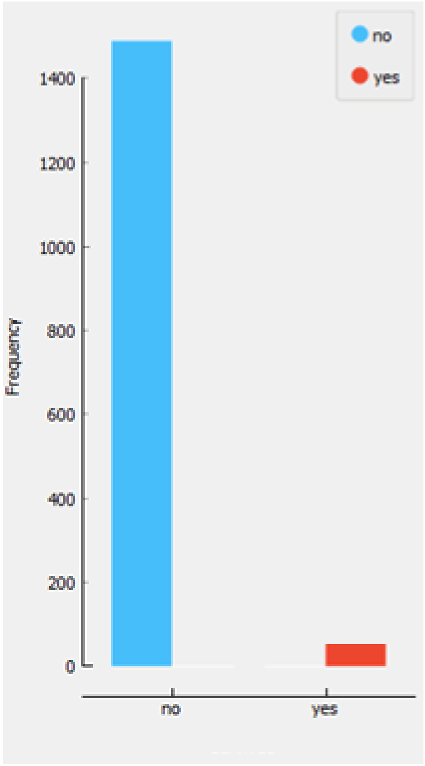
- When is classification accuracy “good”?

Imbalanced Data and Unequal Misclassification Costs

- Imbalanced dataset: One class is minority compared to the other(s)
  - The minority class is tipicaly the one of interest







Imbalanced Data and Unequal Misclassification Costs

- Imbalanced dataset: One class is minority compared to the other(s)
  - The minority class is usually the one of interest
- Unequal misclassification costs:
  - Some errors are more costly (have more severe consequences)
- Examples:
  - Intrusion detection
  - Credit card fraud
  - Screening tests (nuchal scan, Zora, Dora, Svit, …)

Exercise: Credit card fraud

*„FED report notes the fraud rate for debit and prepaid signature transactions in 2012 was approximately 4.04 basis points (bps), or about **four per every 10,000 transactions**.“*

- What is the classification accuracy of a classifier that classifies all the examples a „not fraudulent“?
  - Answer: 99.96%
- Can a classifier with classification accuracy of 97% be “better” then the one with classification accuracy 99.96%?

https://www.pymnts.com/in-depth/2014/a-tale-of-two-fraud-stats/

Exercise: Credit card fraud

Two confusion matrices for two classifiers

		Predicted		
		Fraud	**Not Fraud**	
Actual	**Fraud**	**0**	**4**	4
	Not fraud	**0**	**9996**	9996
		0	10000	

		Predicted		
		Fraud	**Not Fraud**	
Actual	**Fraud**	**4**	**0**	4
	Not fraud	**300**	**9696**	9996
		304	9696	

Classification accuracy

- CA = (0 + 99,96)/10000

  = 99,96%

- CA = (4 + 9696)/10000

  = 97,00%

The model with lower classification accuracy is better.

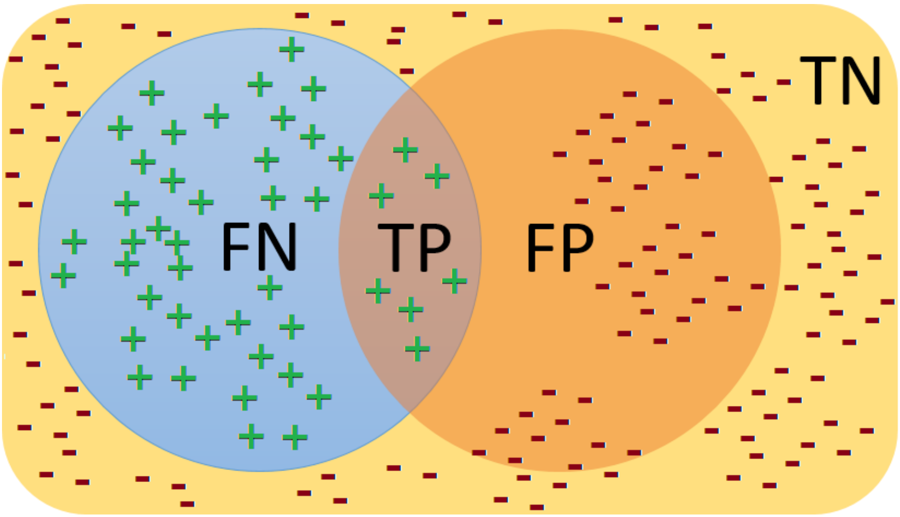
Precision and Recall

PRECISION

- Out of all the examples the classifier labeled as positive, what fraction were correct?

RECALL

- Out of all the positive examples there were, what fraction did the classifier pick up?



Precision & Recall

- Class-specific metrics
  - Precision (Positive Predictive Value)
    - Proportion of instances classified as positive that are really positive
  - Recall (True Positive Rate, TP Rate, Hit Rate, Sensitivity)
    - The proportion of positive instances that are correctly classified as positive
- Exercise: write down the formulas for precision and recall

		Predicted class		Total instances
		+	−	
Actual class	+	TP	FN	P
	−	FP	TN	N

Precision, Recall & F1

- Class-specific metrics
  - Precision (Positive Predictive Value)
    - Proportion of instances classified as positive that are really positive
  - Recall (True Positive Rate, TP Rate, Hit Rate, Sensitivity)
    - The proportion of positive instances that are correctly classified as positive
  - F1
    - Harmonic mean of precision and recall
    - Both precision and recall need to be high for F1 to be high

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

- We can average the metrics over the classes (macro average) or weigh them by the number of examples (micro average)

Precision, Recall, F1

		Predicted class +	Predicted class −	Total instances
Actual class	+	TP	FN	P
	−	FP	TN	N

True Positive Rate or Hit Rate or Recall or Sensitivity or TP Rate	TP/P	The proportion of positive instances that are correctly classified as positive	*Priklic*
Precision or Positive Predictive Value	TP/(TP+FP)	Proportion of instances classified as positive that are really positive	*Natančnost*
F1 Score	$(2 \times \text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$	A measure that combines Precision and Recall	*Mera F1*
Accuracy or Predictive Accuracy	$(\text{TP} + \text{TN})/(\text{P} + \text{N})$	The proportion of instances that are correctly classified	*Klasifikacijska točnost*

Homework: compute the precision, recall and F1 for both classifiers for the class Fraud

Two confusion matrices for two classifiers

		Predicted		
		Fraud	**Not Fraud**	
Actual	**Fraud**	**0**	**4**	4
	Not fraud	**0**	**9996**	9996
		0	10000	

		Predicted		
		Fraud	**Not Fraud**	
Actual	**Fraud**	**4**	**0**	4
	Not fraud	**300**	**9696**	9996
		304	9696	

For the class *Fraud*

- Precision=
- Recall=
- F1=

- Precision=
- Recall=
- F1=

Classification evaluation in Orange

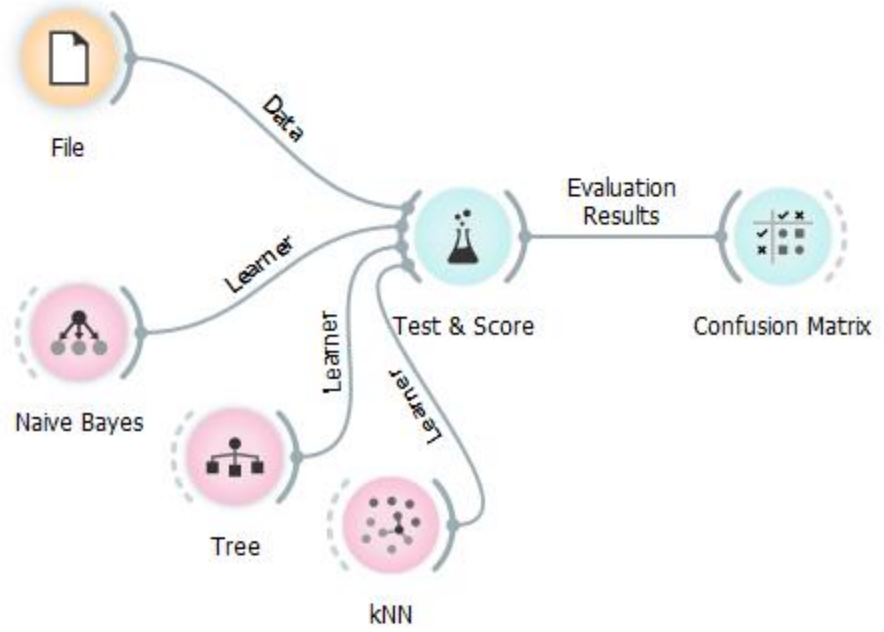
- AUC
  - Area under curve
  - AUROC
  - *Area under ROC curve*
- CA – classification accuracy

For a selected class or averaged over all classes (macro-average)

- F1 – harmonic mean of precision and recall
- Precision
- Recall

Evaluation Results

Method	AUC	CA	F1	Precision	Recall
kNN	0.951	0.845	0.823	0.835	0.845
Naive Bayes	0.971	0.863	0.858	0.859	0.863
Tree	0.991	0.951	0.951	0.951	0.951



Lab exercise

- Compare three evaluation methods
  - Train (70%) test (30%) split
  - Cross validation
  - Random sampling
- Test three models:
  - Decision trees
  - Random forest
  - Naïve Bayes classifier
- Metrics
  - Classification accuracy (CA)
  - Precision, Recall, F1 for selected class
  - Area under curve (AUC) *– more about this to come*
- Use the dataset „car“ from http://file.biolab.si/datasets/

Literature

- Max Bramer: Principles of data mining (2007)
    - 2. Introduction to Classification: Naive Bayes and Nearest Neighbour
    - 6. Estimating the Predictive Accuracy of a Classifier
    - 11. Measuring the Performance of a Classifier