

Introduction to Data Science 5

Overview

Optimisation of parameters in J48

Comparing (variants) of classifiers

Evaluation with the t-test

WEKA's Experimenter

J48

weka.gui.GenericObjectEditor

weka.classifiers.trees.J48

About

Class for generating a pruned or unpruned C4. [More](#) [Capabilities](#)

binarySplits

collapseTree

confidenceFactor

debug

minNumObj

numFolds

reducedErrorPruning

saveInstanceData

seed

subtreeRaising

unpruned

useLaplace

useMDLcorrection

[Open...](#) [Save...](#) [OK](#) [Cancel](#)

Information

NAME
weka.classifiers.trees.J48

SYNOPSIS
Class for generating a pruned or unpruned C4.5 decision tree. For more information, see Ross Quinlan (1993). C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA.

OPTIONS
debug -- If set to true, classifier may output additional info to the console.

minNumObj -- The minimum number of instances per leaf.

confidenceFactor -- The confidence factor used for pruning (smaller values incur more pruning).

binarySplits -- Whether to use binary splits on nominal attributes when building the trees.

seed -- The seed used for randomizing the data when reduced-error pruning is used.

numFolds -- Determines the amount of data used for reduced-error pruning. One fold is used for pruning, the rest for growing the tree.

saveInstanceData -- Whether to save the training data for visualization.

unpruned -- Whether pruning is performed.

subtreeRaising -- Whether to consider the subtree raising operation when pruning.

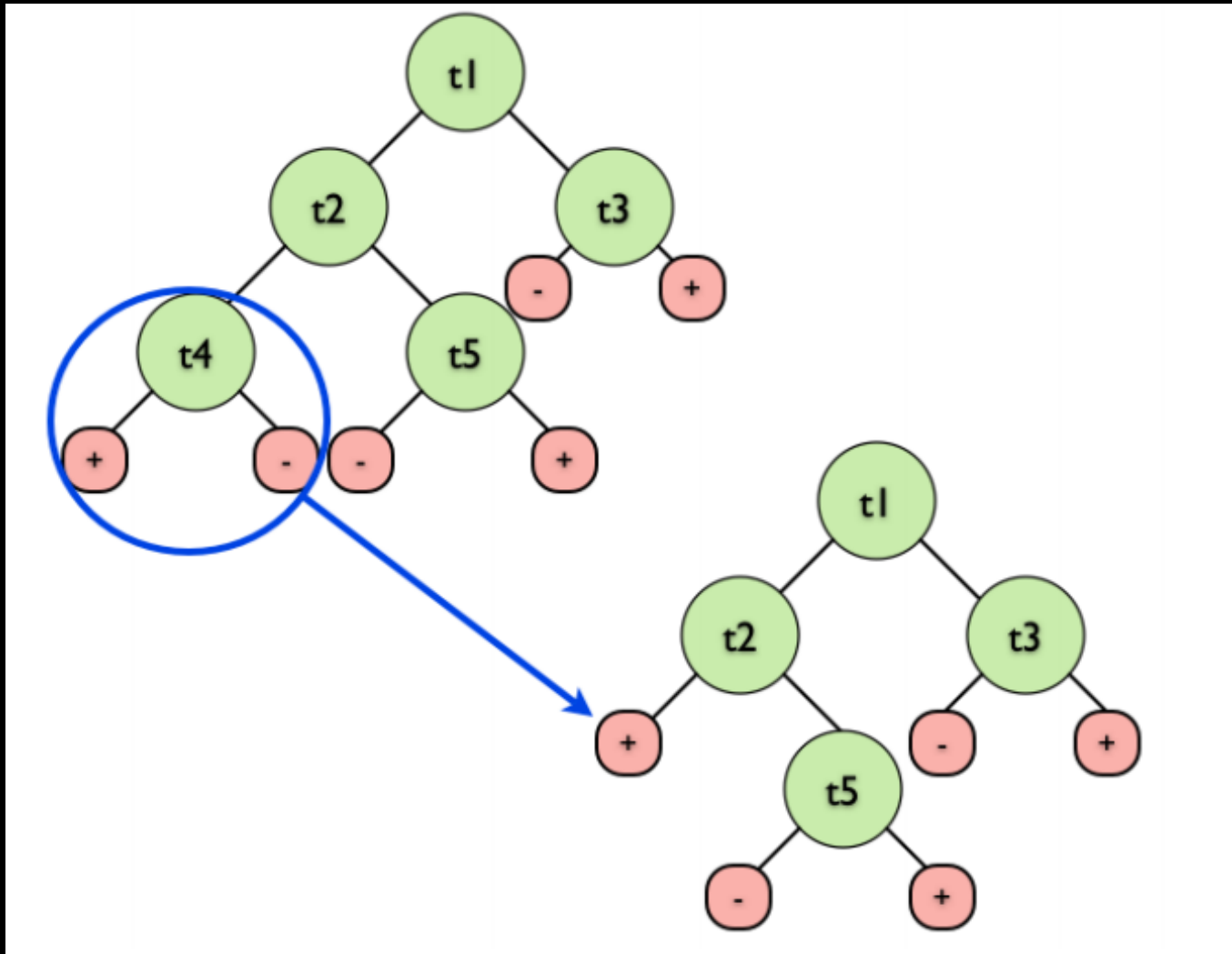
collapseTree -- Whether parts are removed that do not reduce training error.

useMDLcorrection -- Whether MDL correction is used when finding splits on numeric attributes.

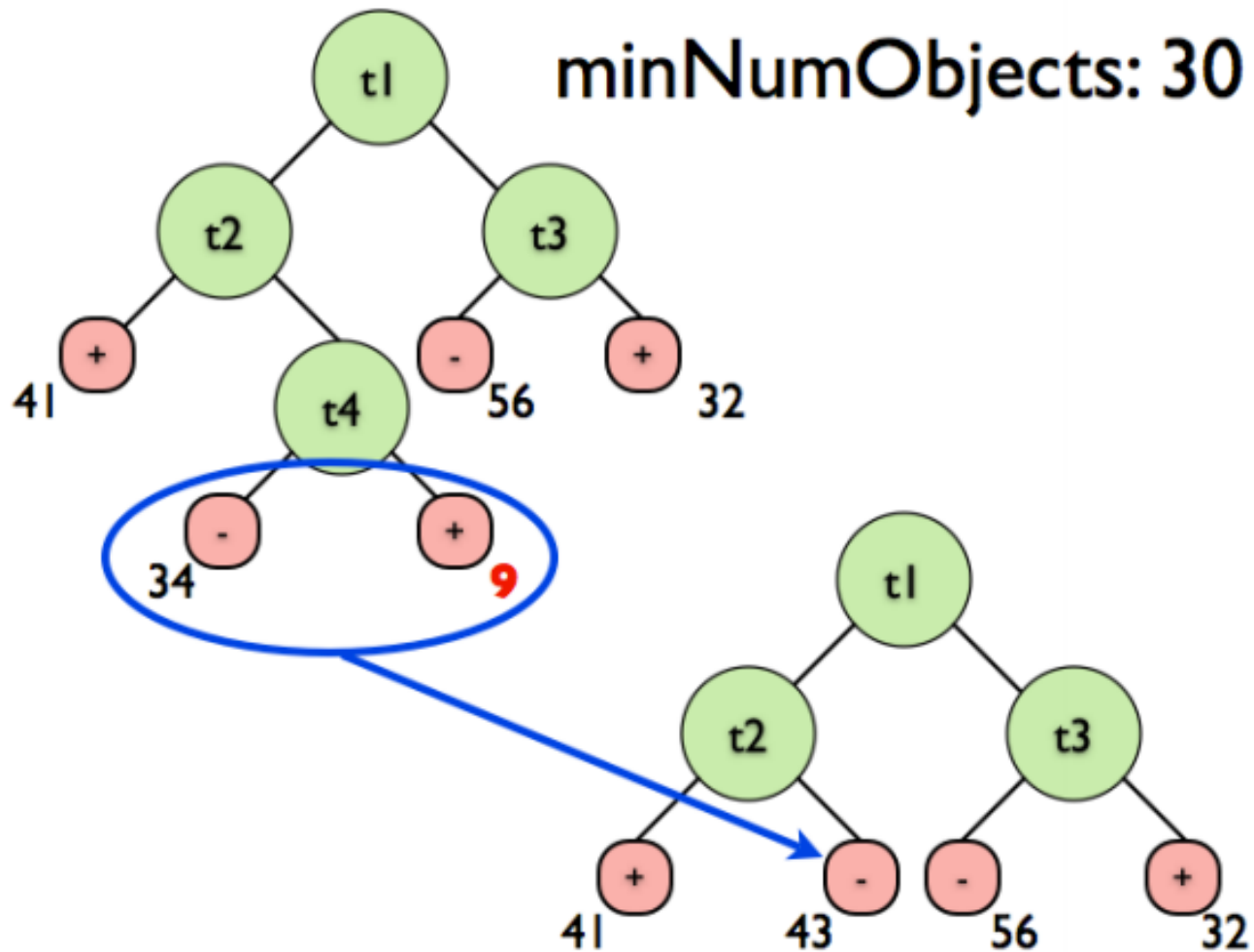
useLaplace -- Whether counts at leaves are smoothed based on Laplace.

reducedErrorPruning -- Whether reduced-error pruning is used instead of C4.5 pruning.

Pruning

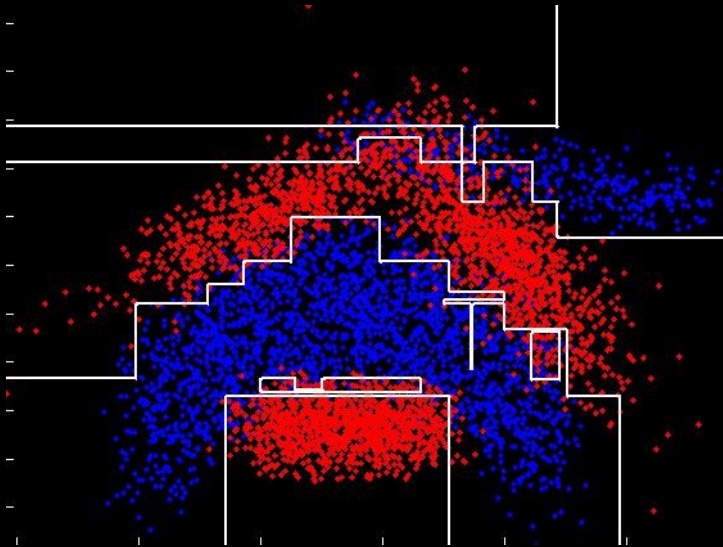


Pruning

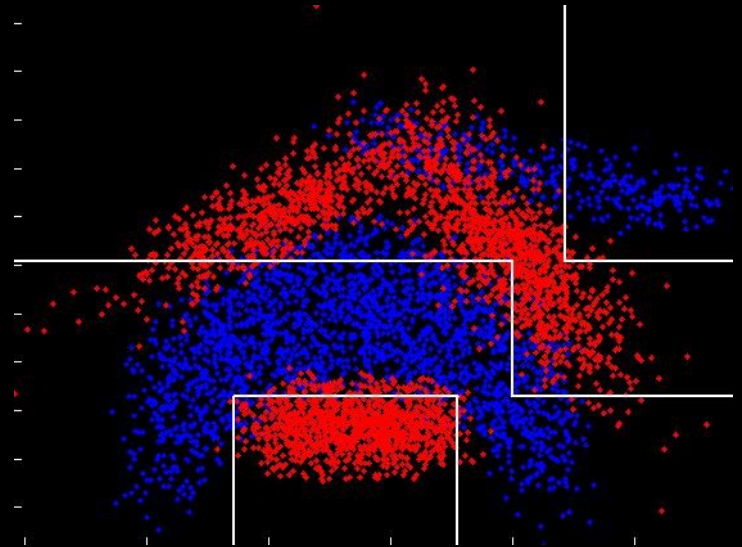


Pruning reduces the complexity of the decision tree

complex tree (unpruned)



Less complex tree (pruned)



Model complexity

A pruned decision tree is less complex, than an unpruned one

Less complex models tend to generalise better (= perform better on unseen data), provided that they are sufficiently complex to capture the structure of the data

kNN versus decision tree

In the kNN classifier, the k parameter tunes the complexity

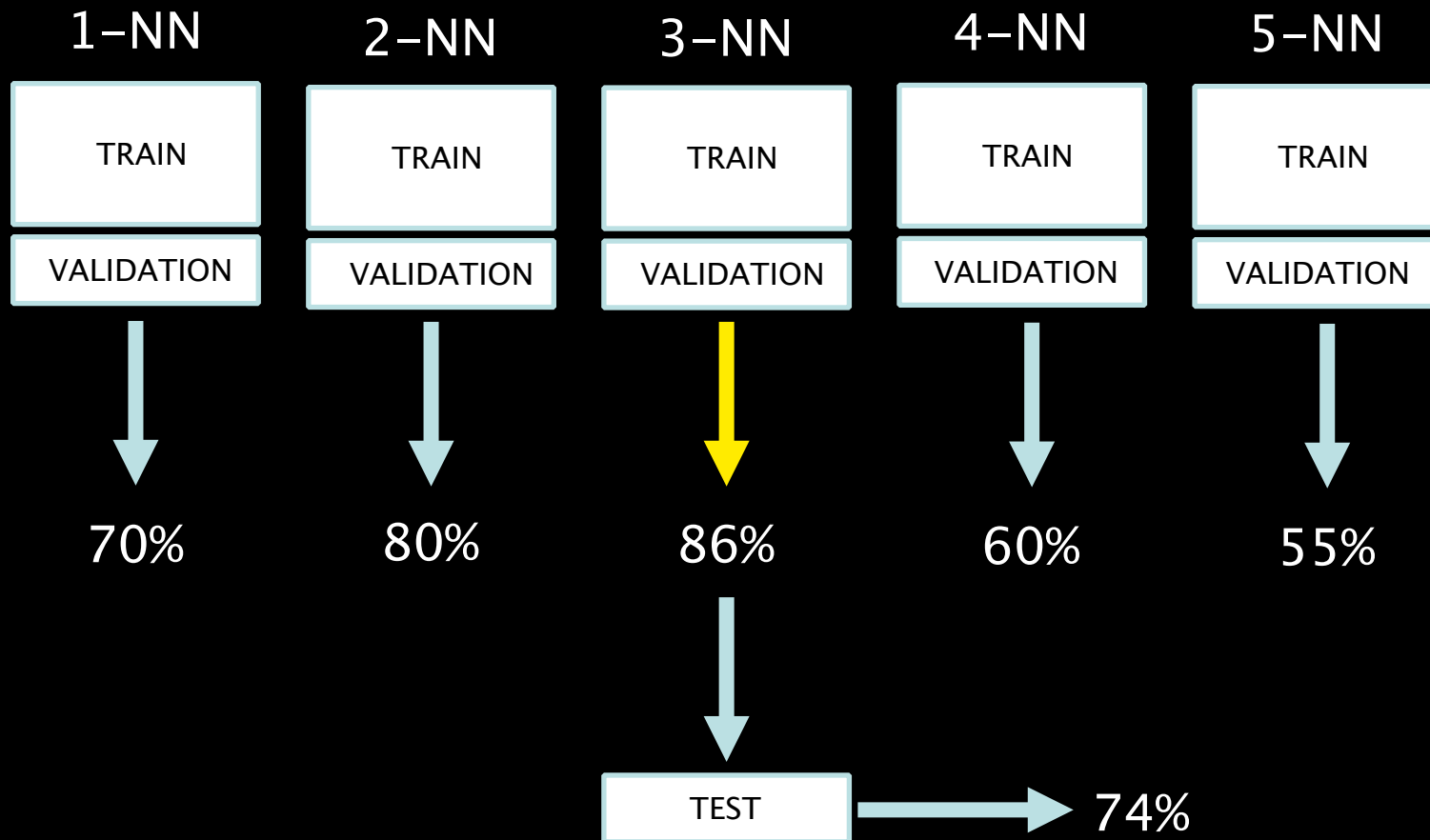
In the decision tree classifier, pruning tunes the complexity

Increasing k or pruning: less complexity

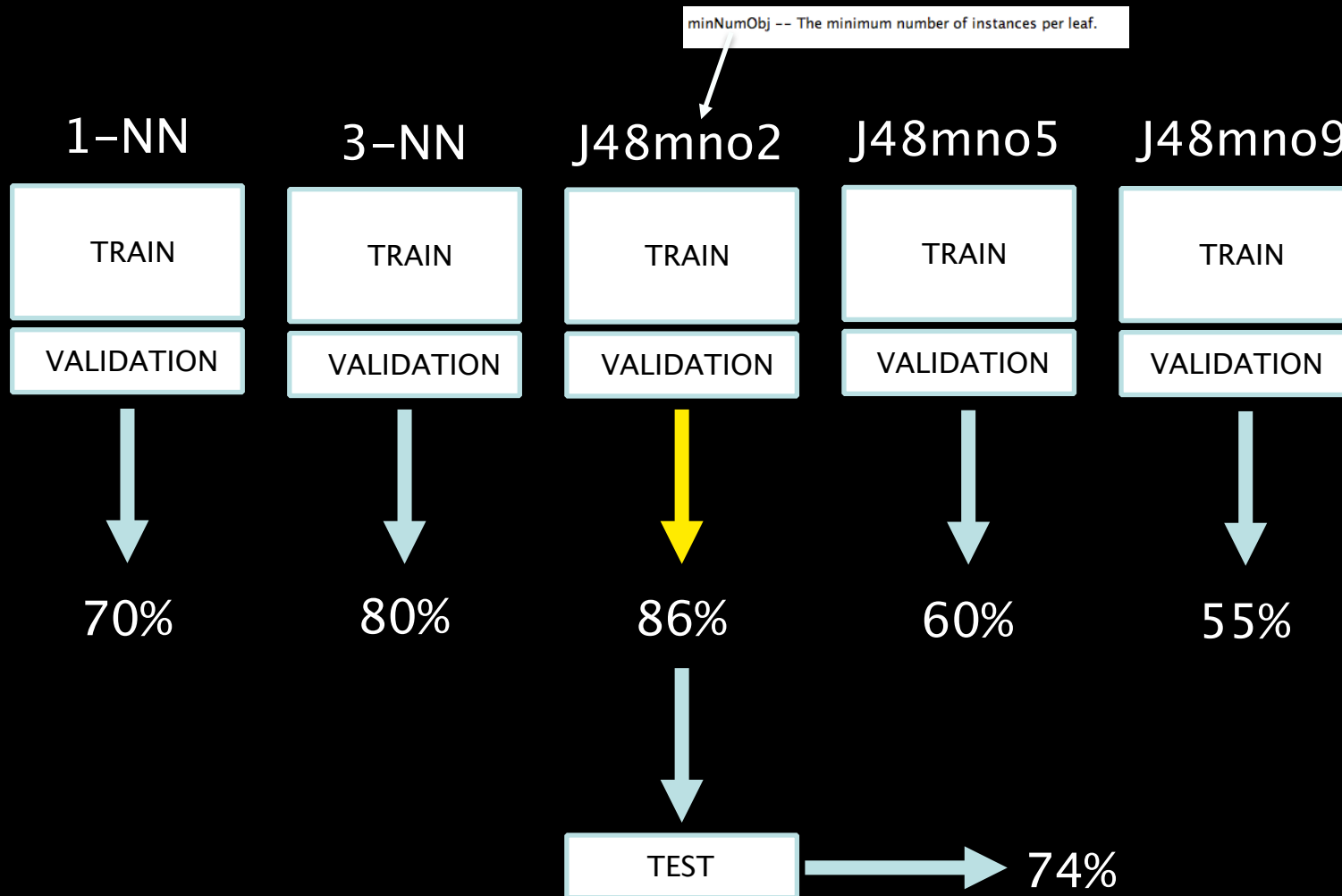
Decreasing k or pruning: more complexity

Comparing (variants) of classifiers

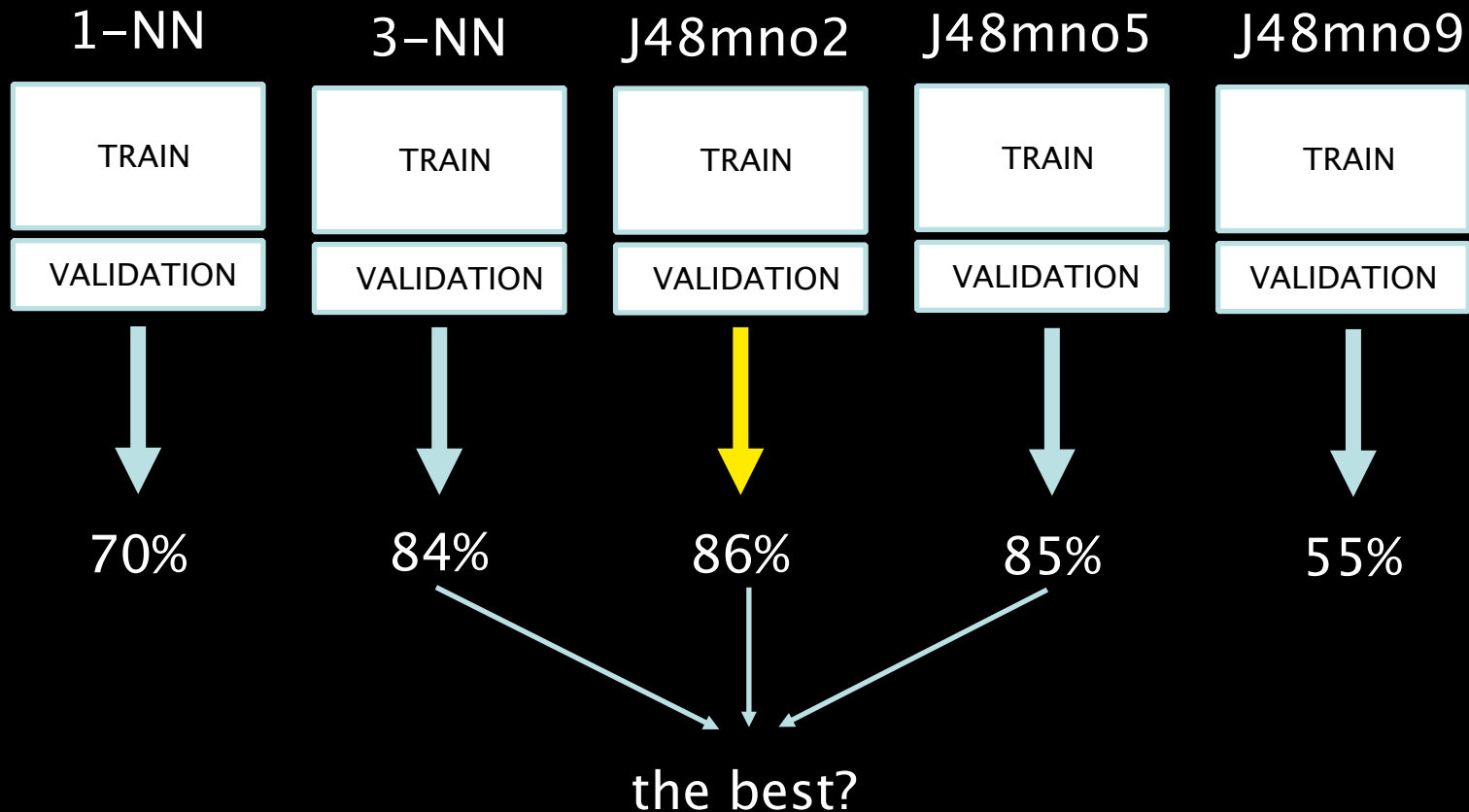
Parameter optimisation (example, see also p.149 WEKA book)



Model selection / parameter optimisation



Significant differences?



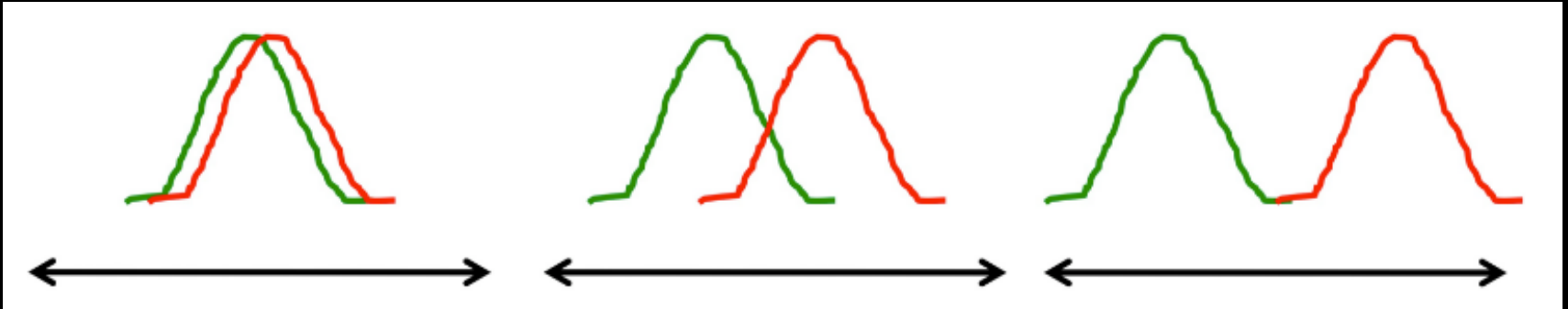
Validation performance is an average score

In case of 10-fold cross validation, it is an average of the scores over 10 folds

So, each validation performance has a standard deviation associated with it

To decide if two scores (averages) differ, you need to perform a statistical test

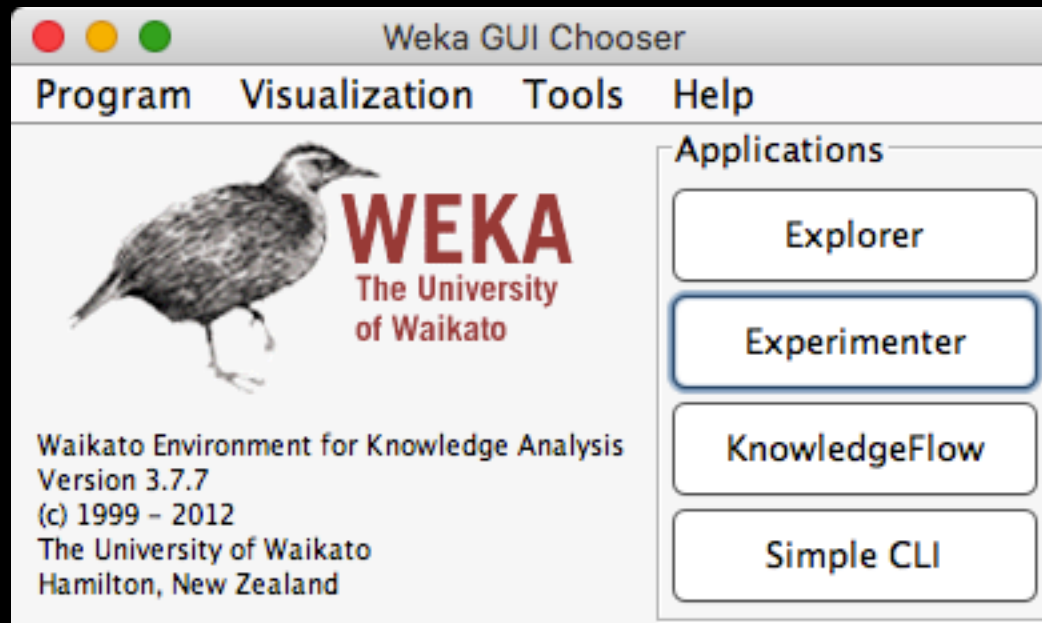
t-test and p-value



$p=0.05$ means: in 1 of 20 experiments you wrongly declare a difference to be significant

$p=p\text{value}$ means: in 1 of $1/p\text{value}$ experiments you wrongly declare a difference to be significant

WEKA Experimenter



Weka Experiment Environment

Setup Run Analyse

Experiment Configuration Mode: ☒ Simple ☐ Advanced

Open... Save... New

Results Destination

ARFF file Browse...

Experiment Type

Cross-validation

Number of folds: 10

☒ Classification ☐ Regression

Iteration Control

Number of repetitions: 10

☒ Data sets first ☐ Algorithms first

Datasets

Add new... Edit selected... Delete selected

☐ Use relative paths

/Users/ericpostma/Documents/Education/SDM2016/datasets/IsItClinton1000.arff

Up Down

Algorithms

Add new... Edit selected... Delete selected

ZeroR
IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A "weka.core.EuclideanDistance -R
J48 -C 0.25 -M 2

Load options... Save options... Up Down

Notes

In total 100 runs (10 x 10cv experiments) will be performed

Source

Got 300 results

File...

Database...

Experiment

Configure test

Testing with Paired T-Tester (correc...

Select rows and cols Rows Cols Swap

Comparison field Percent_correct

Significance 0.05

Sorting (asc.) by <default>

Test base Select

Displayed Columns Select

Show std. deviations ☒

Output Format Select

Perform test

Save output

Result list

12:09:50 - Available resultsets

12:10:02 - Percent_correct - rules.ZeroR " 4805554146586

12:10:16 - Percent_correct - rules.ZeroR " 4805554146586

Test output

Tester: weka.experiment.PairedCorrectedTTester
 Analysing: Percent_correct
 Datasets: 1
 Resultsets: 3
 Confidence: 0.05 (two tailed)
 Sorted by: -
 Date: 10/2/16 12:10 PM

Dataset	(1) rules.ZeroR ''	(2) lazy.IBk '-	(3) trees.J48 '
isClinton	(100) 50.00(0.00)	93.12(1.77) v	88.50(2.09) v
	(v/ /*)	(1/0/0)	(1/0/0)

Key:

(1) rules.ZeroR '' 48055541465867954
 (2) lazy.IBk '-K 1 -W 0 -A \\weka.core.neighboursearch.LinearNNSearch -A \\weka.core.EuclideanDistance -R first
 (3) trees.J48 '-C 0.25 -M 2' -217733168393644444

v = performs significantly better than the Test base
 * = performs significantly worse than the Test base

Required Reading

Bouckaert, R.R. & Frank, E. (2004). Evaluating the Replicability of Significance Tests for Comparing Learning Algorithms.
In H. Dai, R. Srikant, & C. Zhang (Eds.), Advances in Knowledge Discovery and Data Mining, Volume 3056 of the series Lecture Notes in Computer Science pp 3–12. Springer.

http://www.cs.waikato.ac.nz/~eibe/pubs/bouckaert_and_frank.pdf