



University of Minho
School of Engineering



Machine Learning with Knime

Similarity Based Systems

Perfil ML:FA@MiEI/4º ano - 1º Semestre

@MES/2º ano - 1º Semestre

Bruno Fernandes, Paulo Novais

31/10/2019

Contents

2

Decision Trees

Loops

Hands On

- The Learner-Predictor concept and Decision Trees
- Loops
- Tuning a Decision Tree
- Hands On

The Learner-Predictor Concept

3

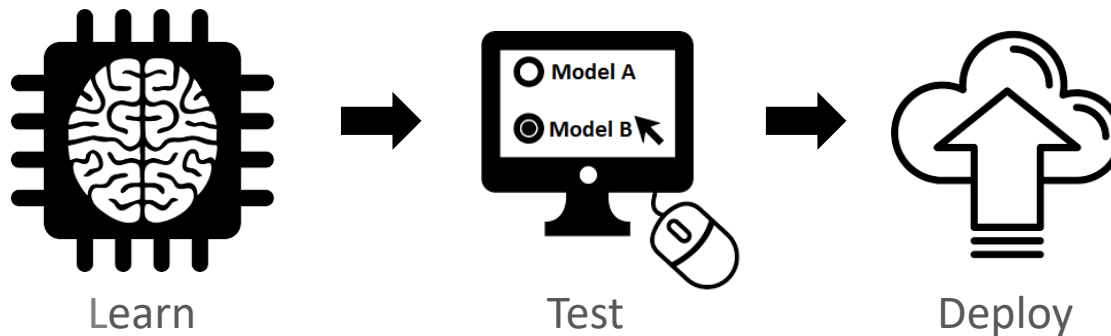
DECISION TREES

Loops

Hands On

Supervised algorithms imply a **learning phase** before applying the model to new data. But they also require a **testing phase** and a **tuning phase**!

In KNIME we implement supervised algorithms with **Learner** and **Predictor nodes**



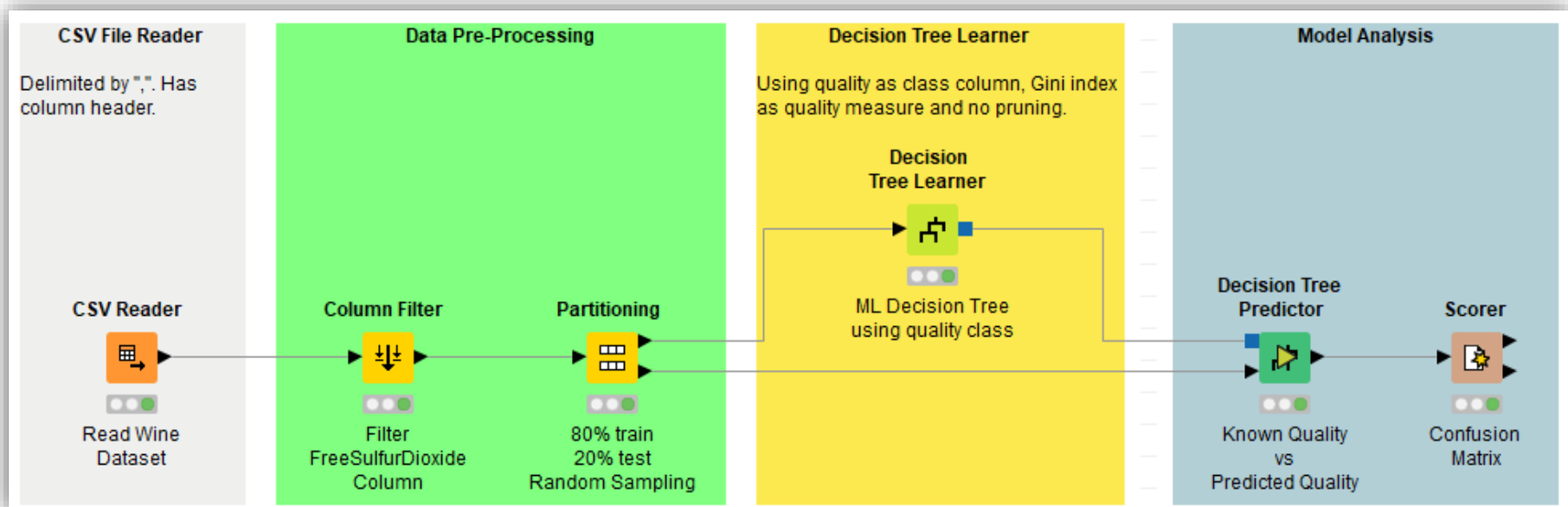
Learner-Predictor

4

DECISION TREES

Loops

Hands On



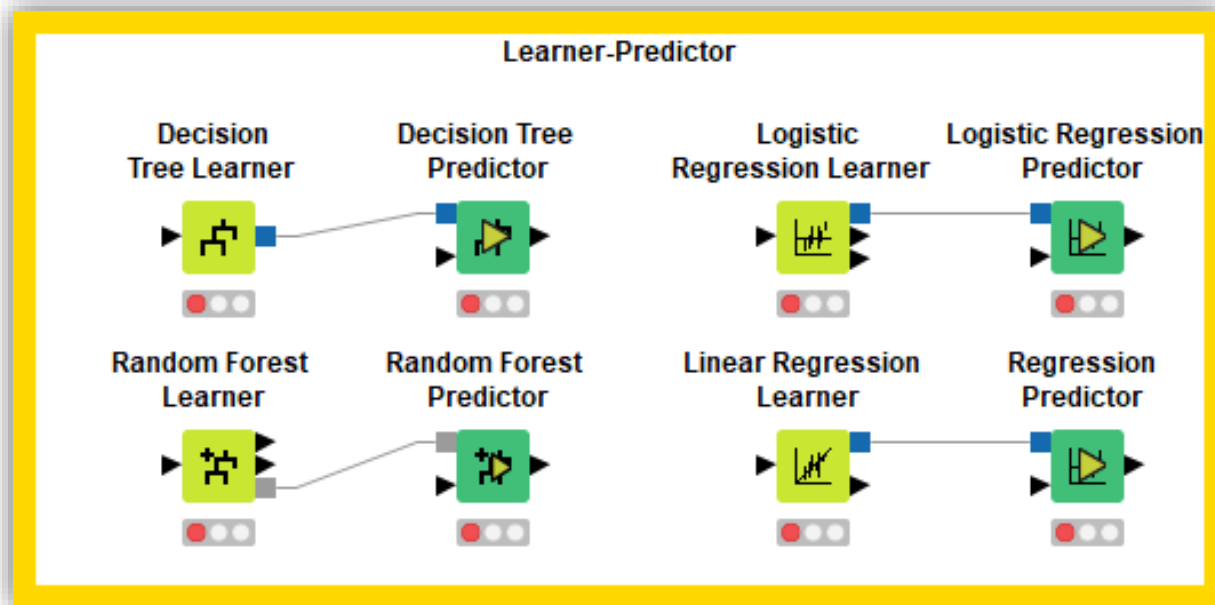
Some Learner-Predictor Nodes

5

DECISION TREES

Loops

Hands On



Decision Tree Learner Settings

6

DECISION TREES

Loops

Hands On

Dialog - 2:8:4 - Decision Tree Learner (ML D...)

File

Options PMMLSettings Flow Variables Memory Policy

General

Class column **S** quality

Quality measure Gini index

Pruning method No pruning

☒ Reduced Error Pruning

Min number records per node 10

Number records to store for view 10,000

☒ Average split point

Number threads 4

☒ Skip nominal columns without domain information

Root split

☐ Force root split column

Root split column **D** alcohol

Binary nominal splits

☐ Binary nominal splits

Max #nominal 10

☐ Filter invalid attribute values in child nodes

OK Apply Cancel ?

Node Description

Dialog Options

Class column

To select the target attribute. Only nominal attributes are allowed

Quality measure

To select the quality measure according to which the split is calculated. Available are the "Gini Index" and the "Gain Ratio".

Pruning method

Pruning reduces tree size and avoids overfitting which increases the generalization performance, and thus, the prediction quality (for predictions, use the "Decision Tree Predictor" node). Available is the "Minimal Description Length" (MDL) pruning or it can also be switched off.

Reduced Error Pruning

If checked (default), a simple pruning method is used to cut the tree in a post-processing step: Starting at the leaves, each node is replaced with its most popular class, but only if the prediction accuracy doesn't decrease. Reduced error pruning has the advantage of simplicity and speed.

Min number records per node

To select the minimum number of records at least required in each node. If the number of records is smaller or equal to this number the tree is not grown any further. This corresponds to a stopping criteria (pre pruning).

Number records to store for view

To select the number of records stored in the tree for the view. The records are necessary to enable

Loop Nodes

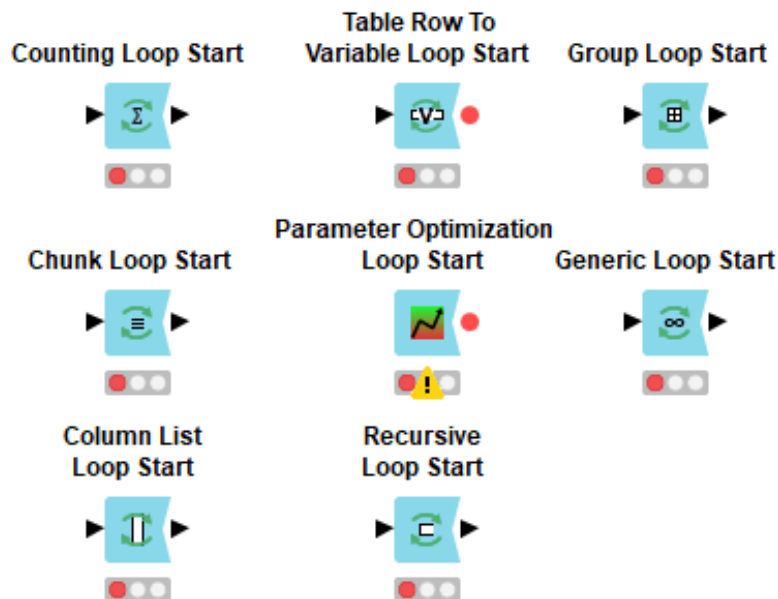
7

Decision Trees

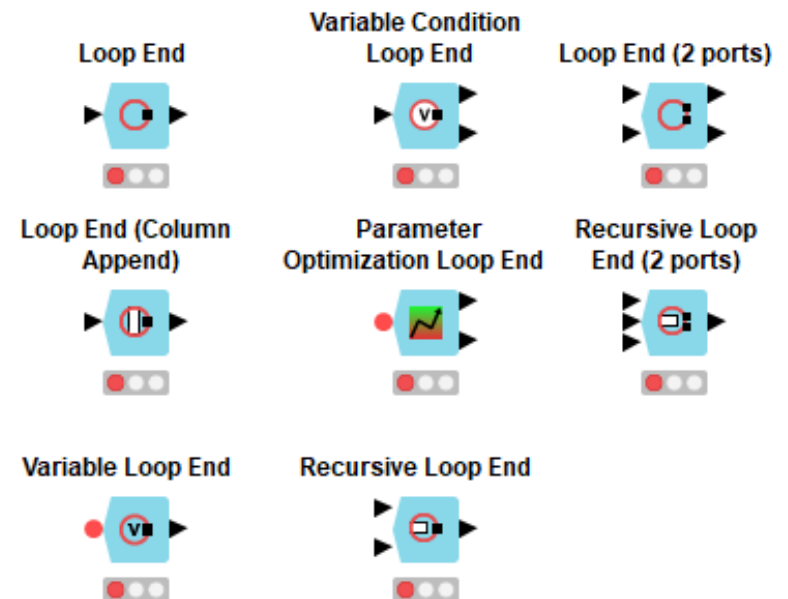
LOOPS

Hands On

Loop Start Nodes



Loop End Nodes



Install **Knime Optimization Extension**

Tuning Numeric Parameters

8

Decision Trees

LOOPS

Hands On

Dialog - 0:4 - Decision Tree Learner (ML Decision Tree)

File

Options PMMLSettings Flow Variables Memory Policy

General

Class column **S** quality ▾

Quality measure Gini index ▾

Pruning method No pruning ▾

☒ Reduced Error Pruning

Min number records per node 10 ▴ ▾

Number records to store for view 10.000 ▴ ▾

☒ Average split point

Number threads 4 ▴ ▾

☒ Skip nominal columns without domain information

Root split

☐ Force root split column

Root split column **D** alcohol ▾

Binary nominal splits

☐ Binary nominal splits

Max #nominal 10 ▴ ▾

OK Apply Cancel ?

Tuning Numeric Parameters

Parameter Optimization Loop Nodes

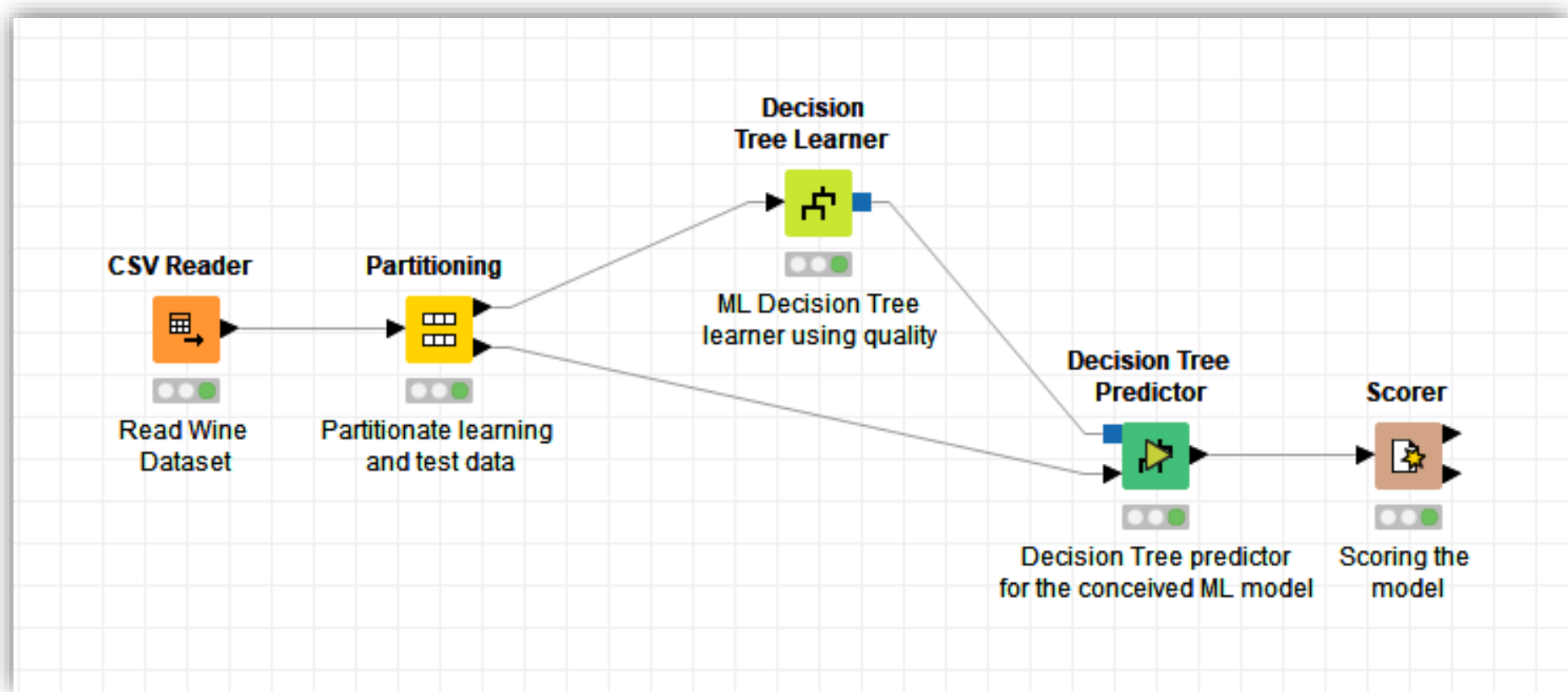
9

Decision Trees

LOOPS

Hands On

Optimize the **value** of some parameters with respect to a **cost function**



Tuning Numeric Parameters

Parameter Optimization Loop Nodes

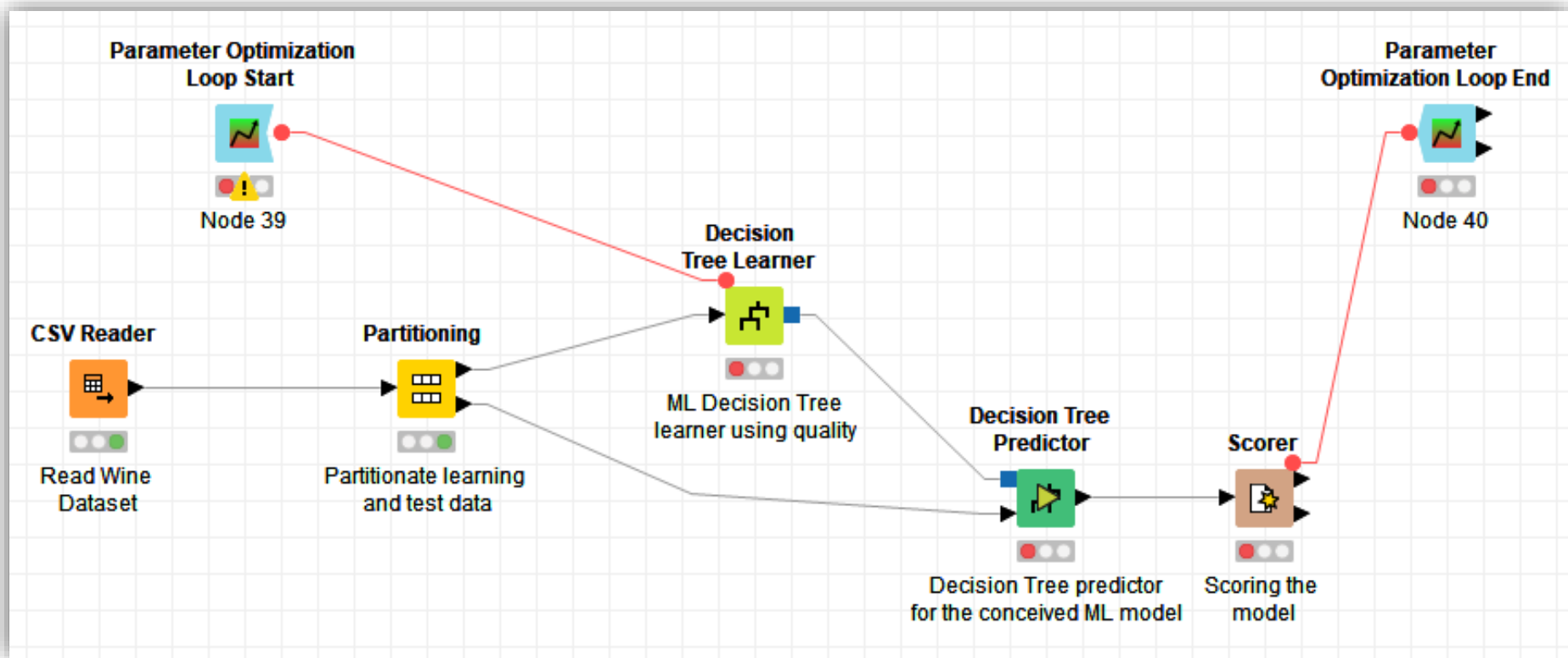
10

Decision Trees

LOOPS

Hands On

Optimize the **value** of some parameters with respect to a **cost function**



Tuning Numeric Parameters

Parameter Optimization Loop Start

11

Decision Trees

LOOPS

Hands On

The image shows a workflow diagram on the left and a dialog box on the right.

Workflow Diagram:

- CSV Reader:** Labeled "Read Wine Dataset".
- Partitioning:** Labeled "Partitionate learning and test data".
- Decision Tree:** Labeled "ML Decision learner".
- Parameter Optimization Loop Start:** A node labeled "Node 39" with a red line connecting it to the Partitioning node.

Dialog - 0:39 - Parameter Optimization Loop Start:

File

Standard settings | Flow Variables | Memory Policy

Parameter	Start value	Stop value	Step size	Integer?	
stoppingCriteria	2	10	1	<input checked="" type="checkbox"/>	

+ Add new parameter

Search strategy: BruteForce (dropdown menu showing BruteForce and Hillclimbing)

OK Apply Cancel ?

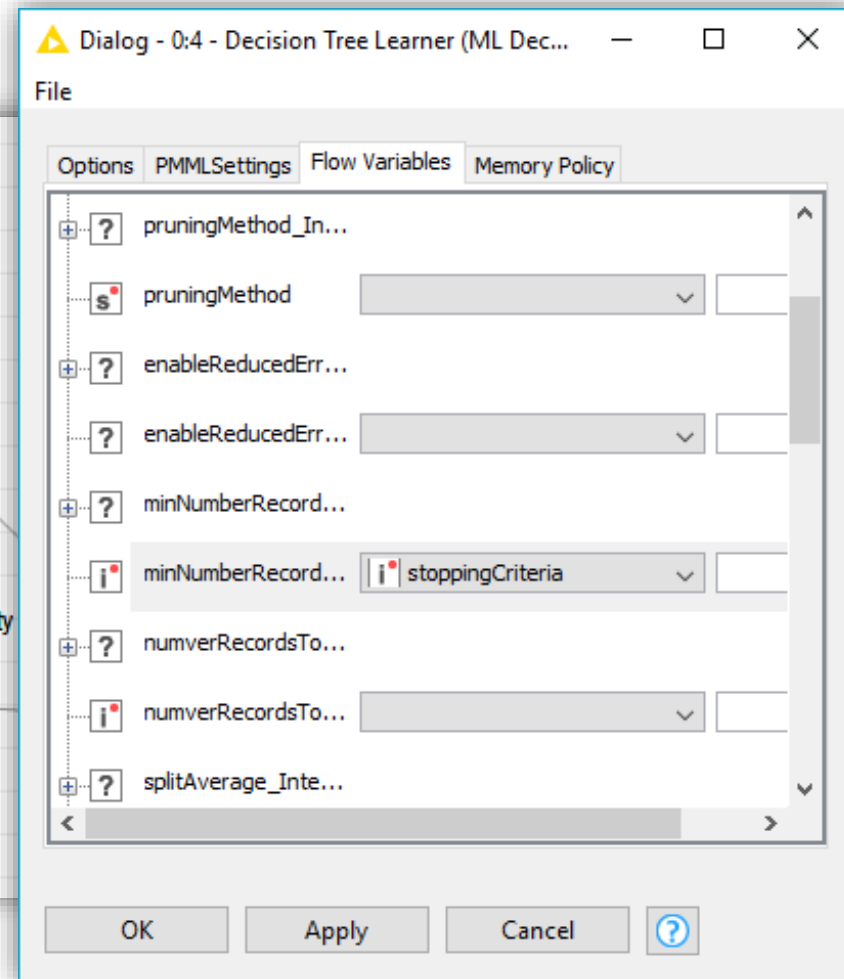
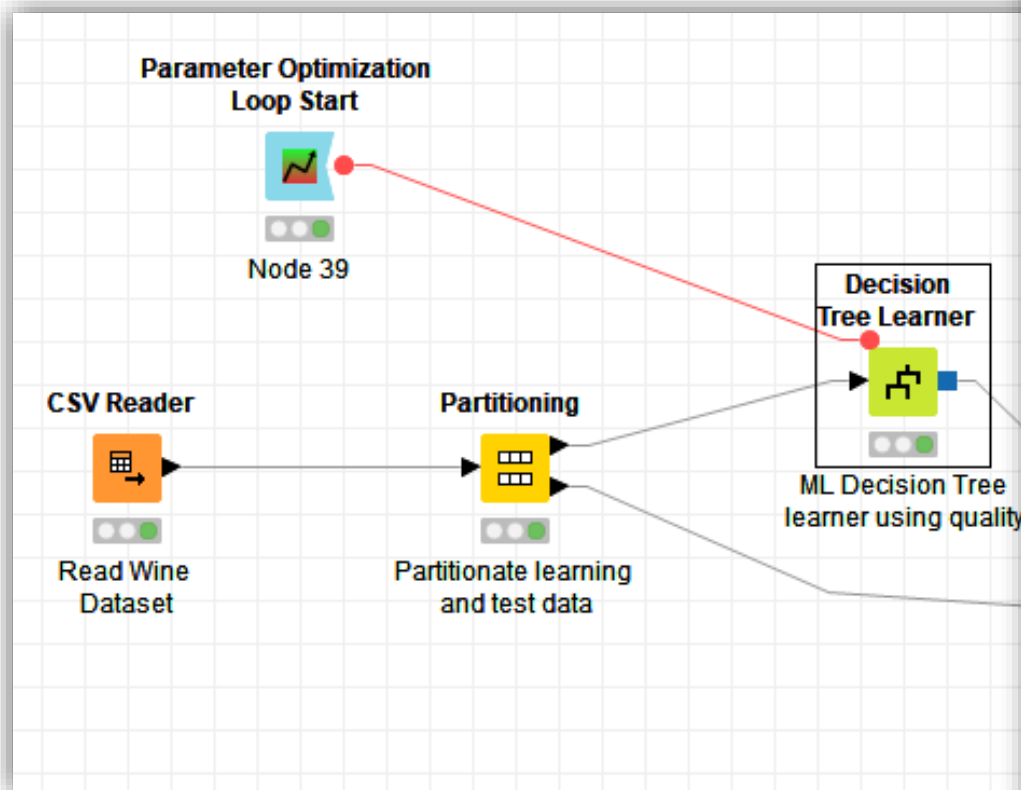
Tuning Numeric Parameters Using the variables

12

Decision Trees

LOOPS

Hands On



Tuning Numeric Parameters

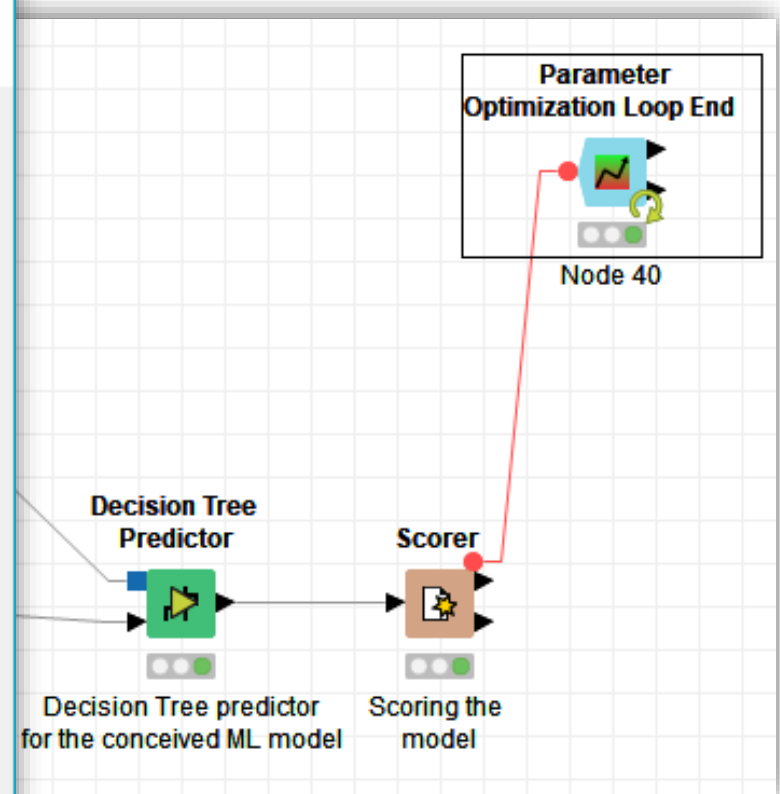
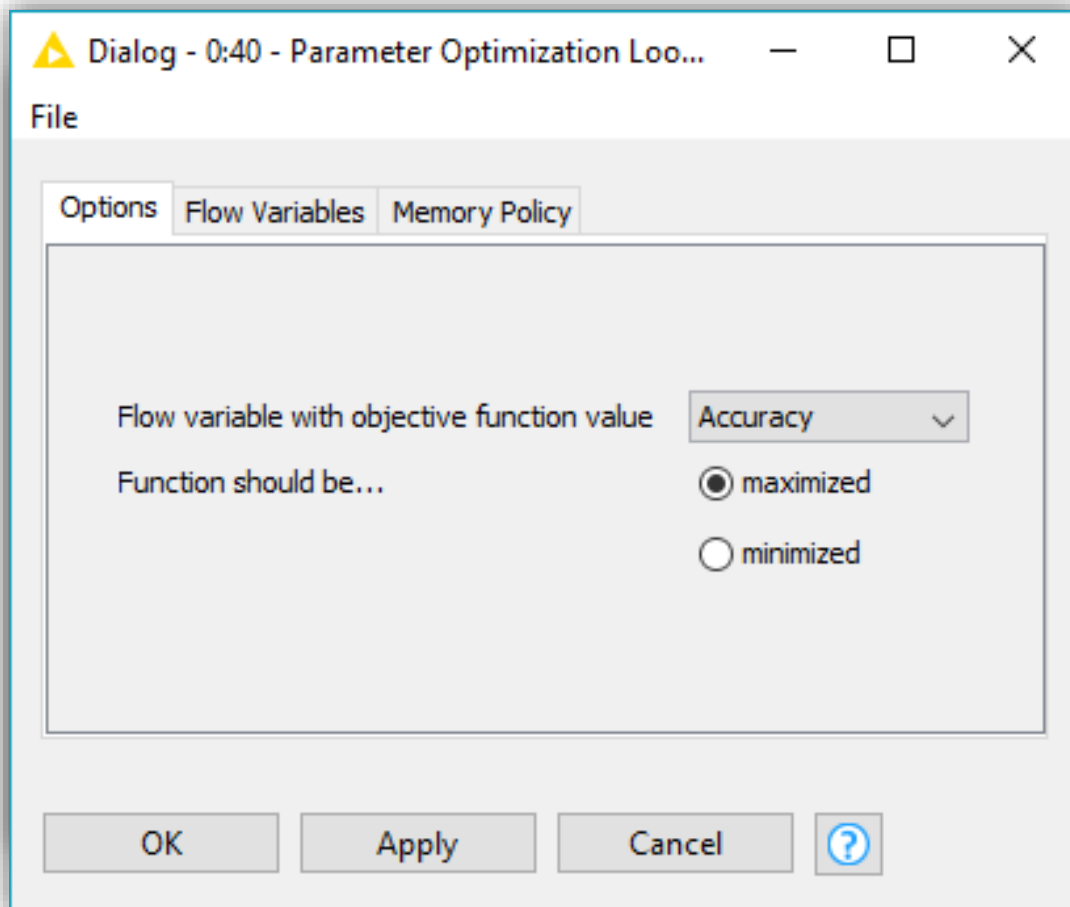
Parameter Optimization Loop End

13

Decision Trees

LOOPS

Hands On



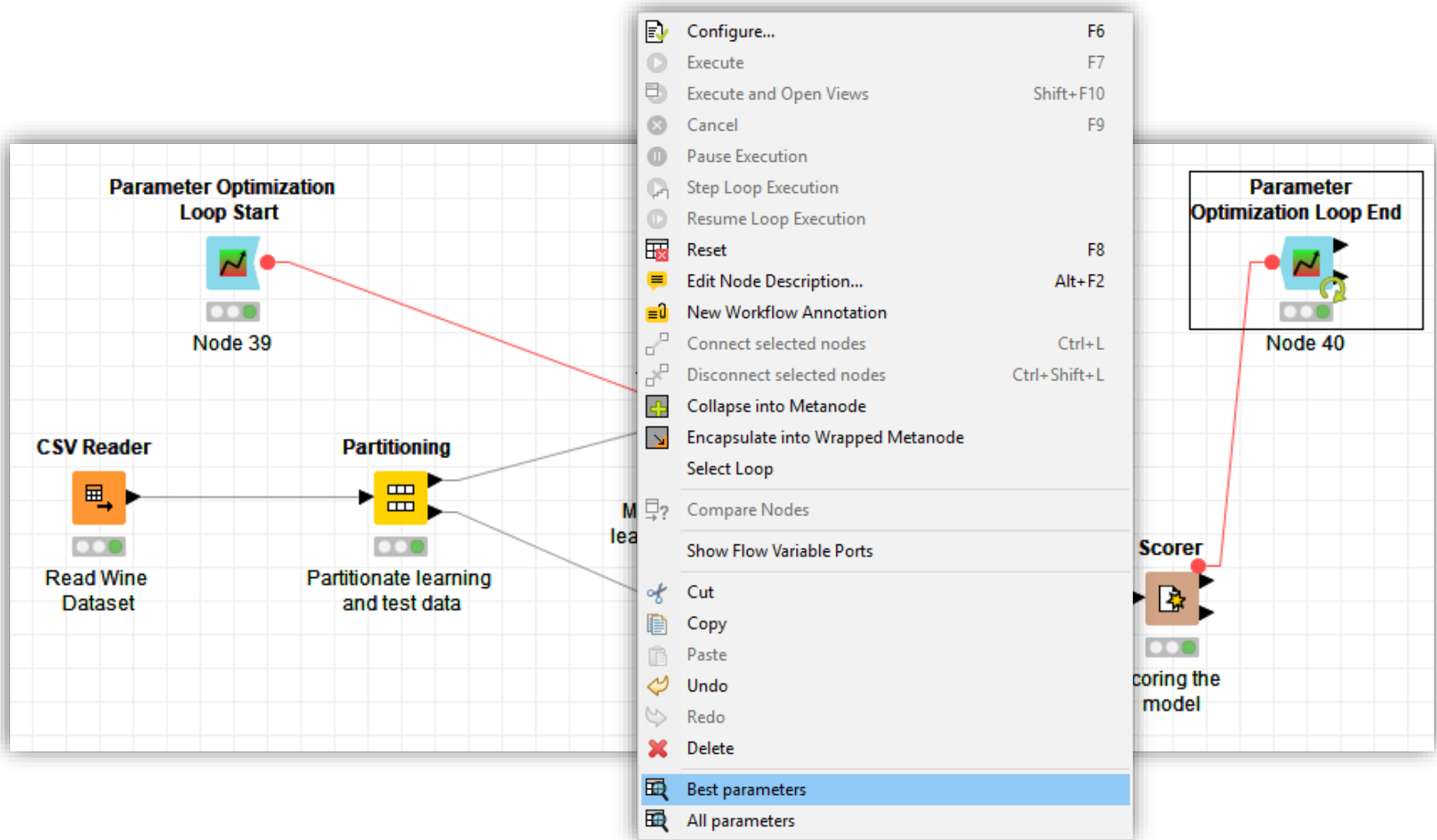
Tuning Numeric Parameters Obtained Results

14

Decision Trees

LOOPS

Hands On



Tuning Numeric Parameters

Obtained Results

15

Decision Trees

LOOPS

Hands On

Best parameters - 0:40 - Parameter Op... — □ ×

File Hilite Navigation View

Table "default" - Rows: 1 Spec - Columns: 2 Properties Flow Variables

Row ID	I stoppin...	D Objecti...
Best parameters	2	0.619

All parameters - 0:40 - Parameter Opti... — □ ×

File Hilite Navigation View

Table "default" - Rows: 9 Spec - Columns: 2 Properties Flow Variables

Row ID	I stoppin...	D Objecti...
Row0	2	0.619
Row1	3	0.606
Row2	4	0.597
Row3	5	0.597
Row4	6	0.581
Row5	7	0.562
Row6	8	0.566
Row7	9	0.569
Row8	10	0.562

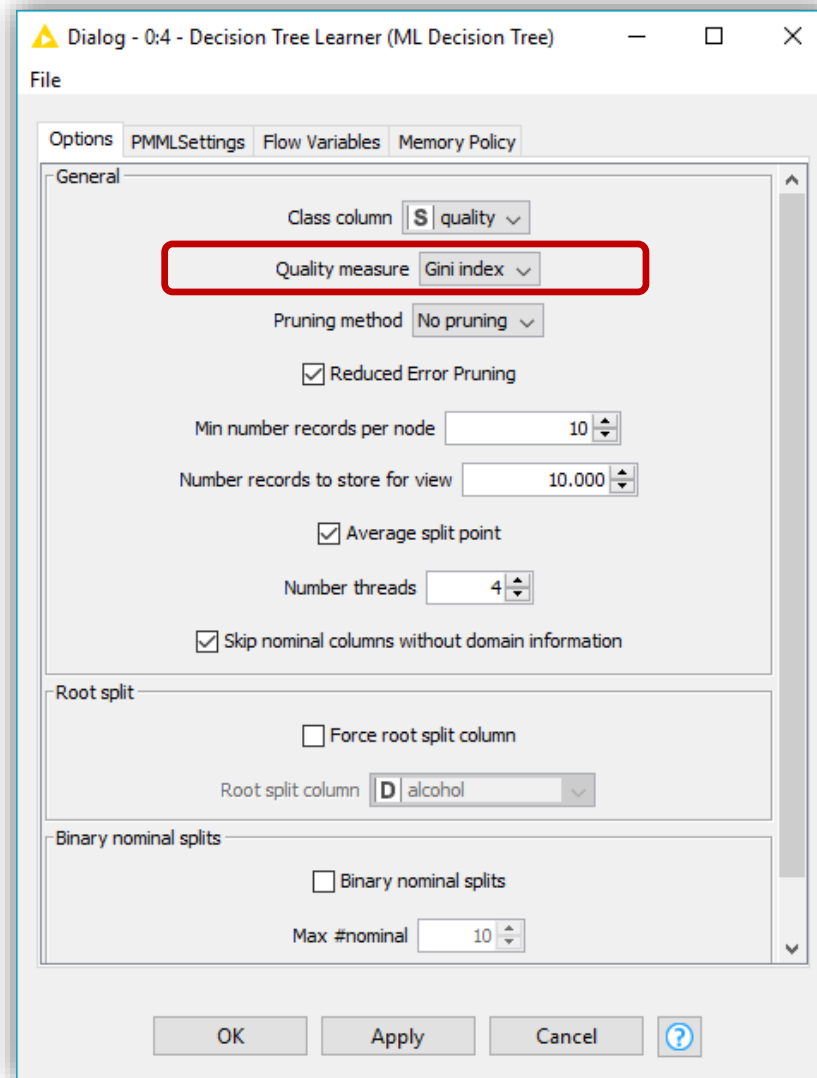
Tuning Nominal Parameters

16

Decision Trees

LOOPS

Hands On



Tuning Nominal Parameters

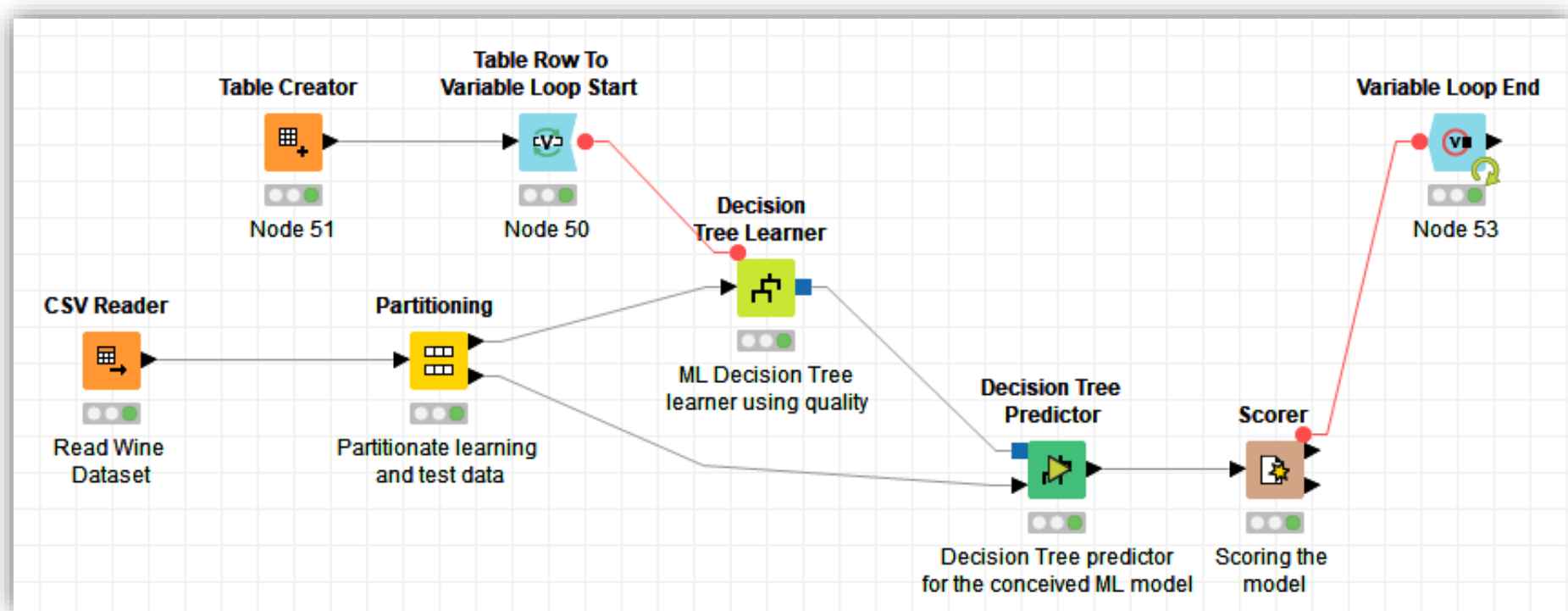
Table Row to Variable Loop

17

Decision Trees

LOOPS

Hands On



Tuning Nominal Parameters

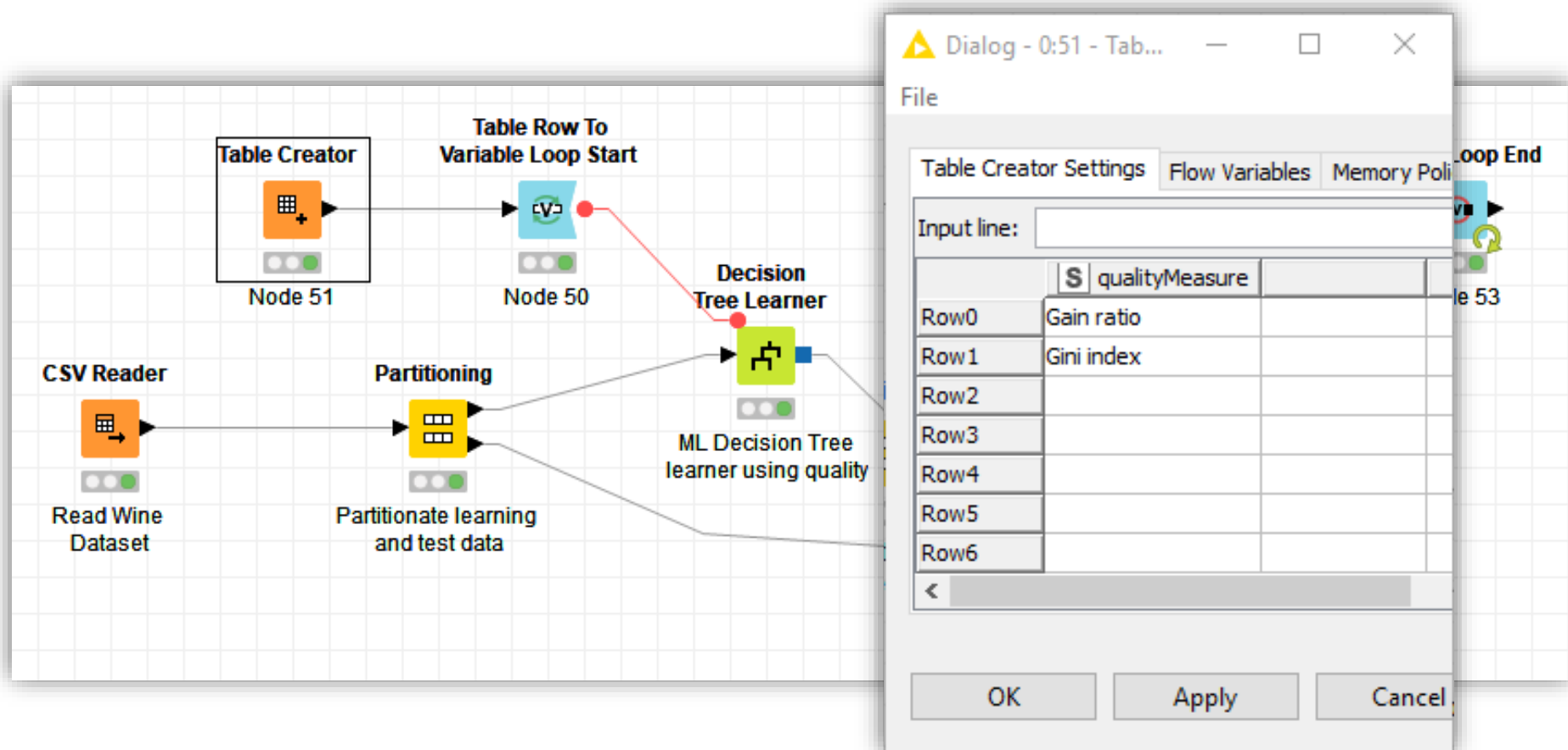
Parameter Possible Values

18

Decision Trees

LOOPS

Hands On



Tuning Nominal Parameters

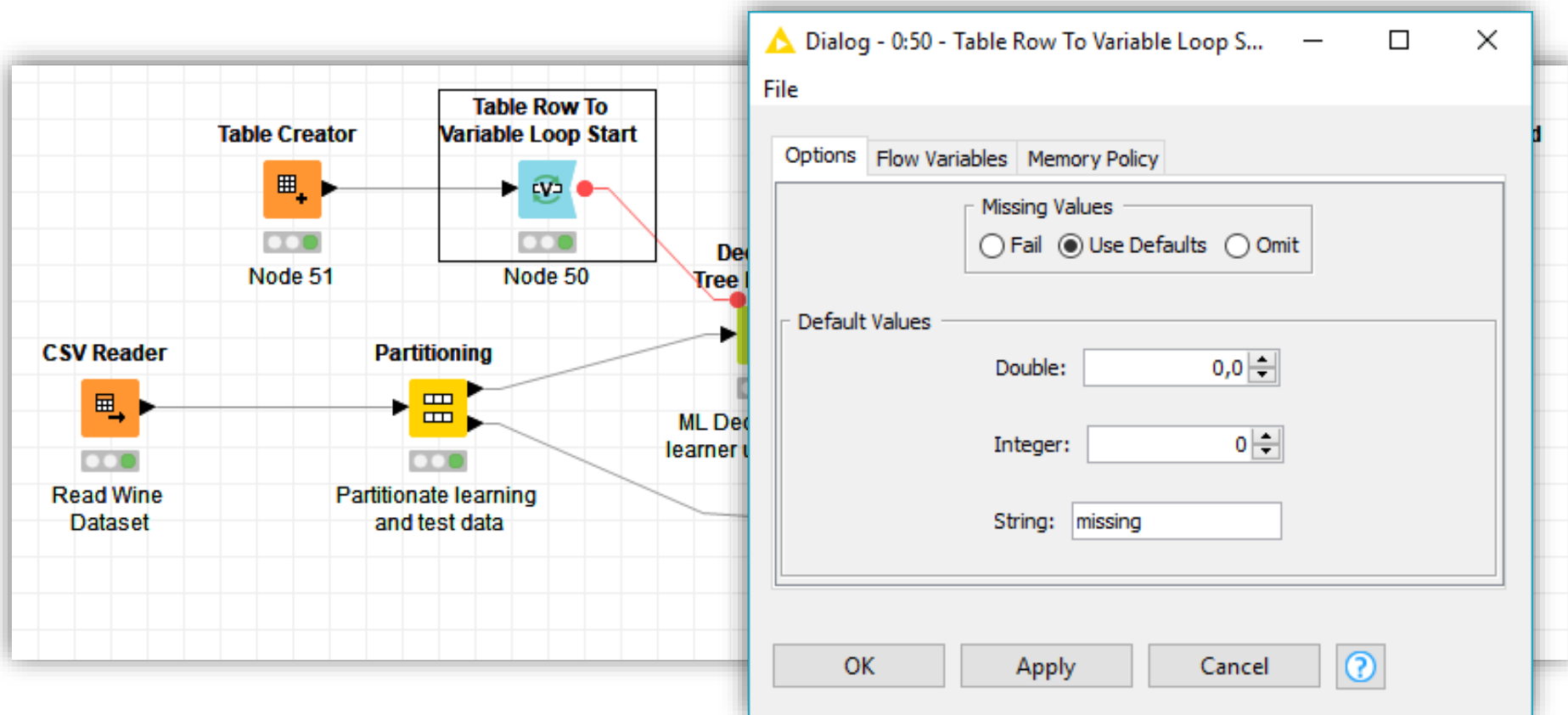
Table Row to Variable Loop Start

19

Decision Trees

LOOPS

Hands On



Tuning Nominal Parameters

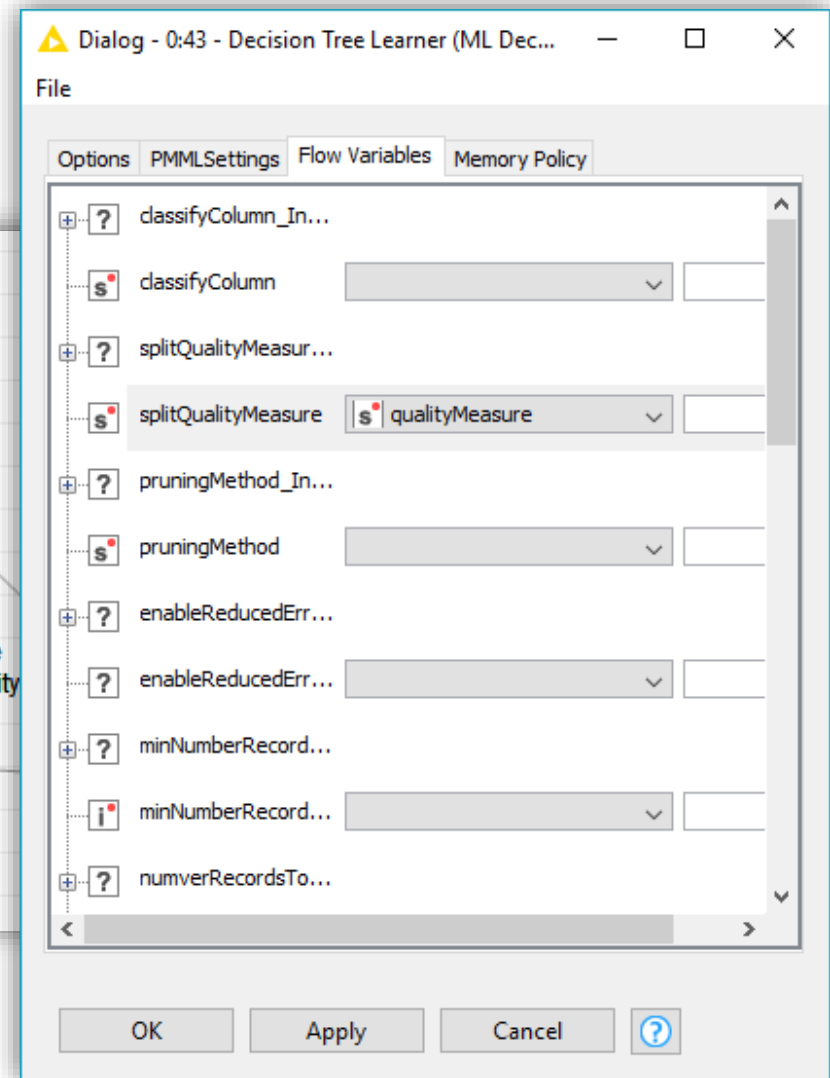
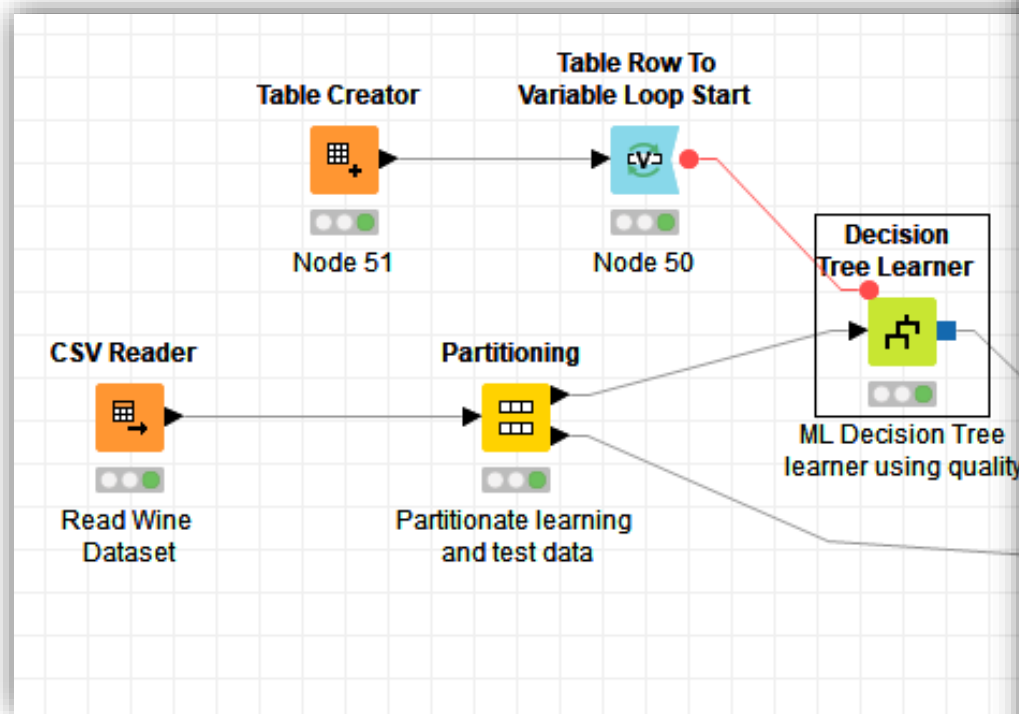
Using the variables

20

Decision Trees

LOOPS

Hands On



Tuning Nominal Parameters

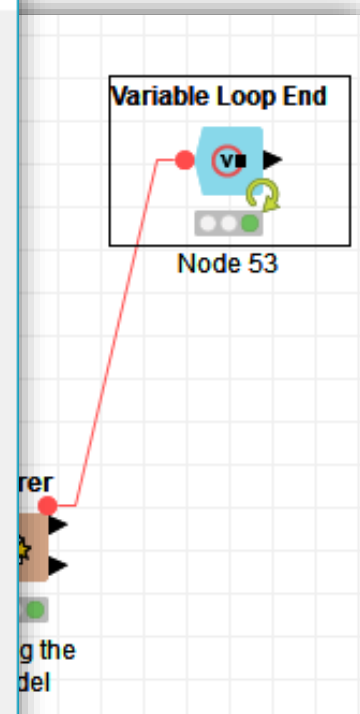
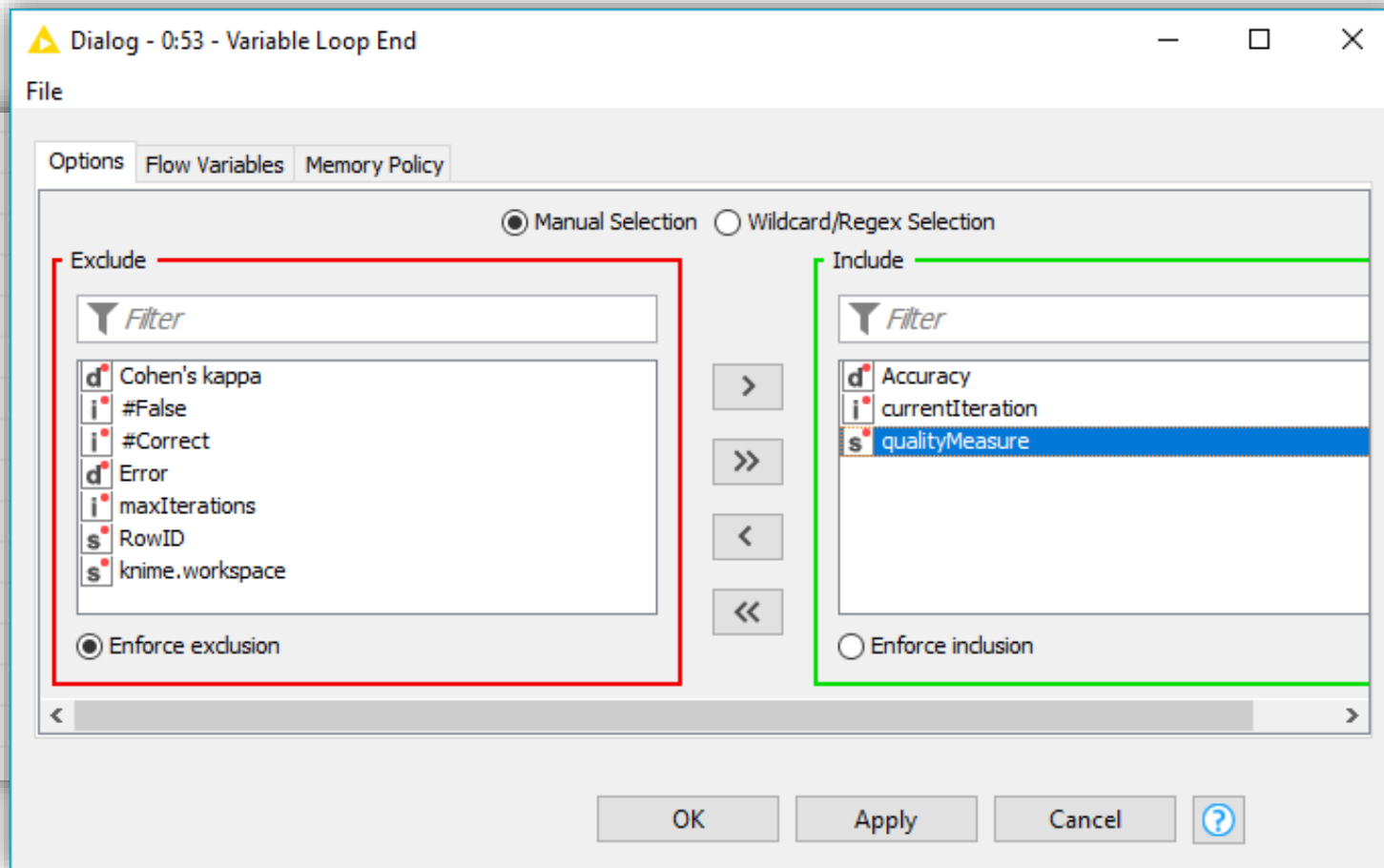
Variable Loop End

21

Decision Trees

LOOPS

Hands On



Tuning Nominal Parameters

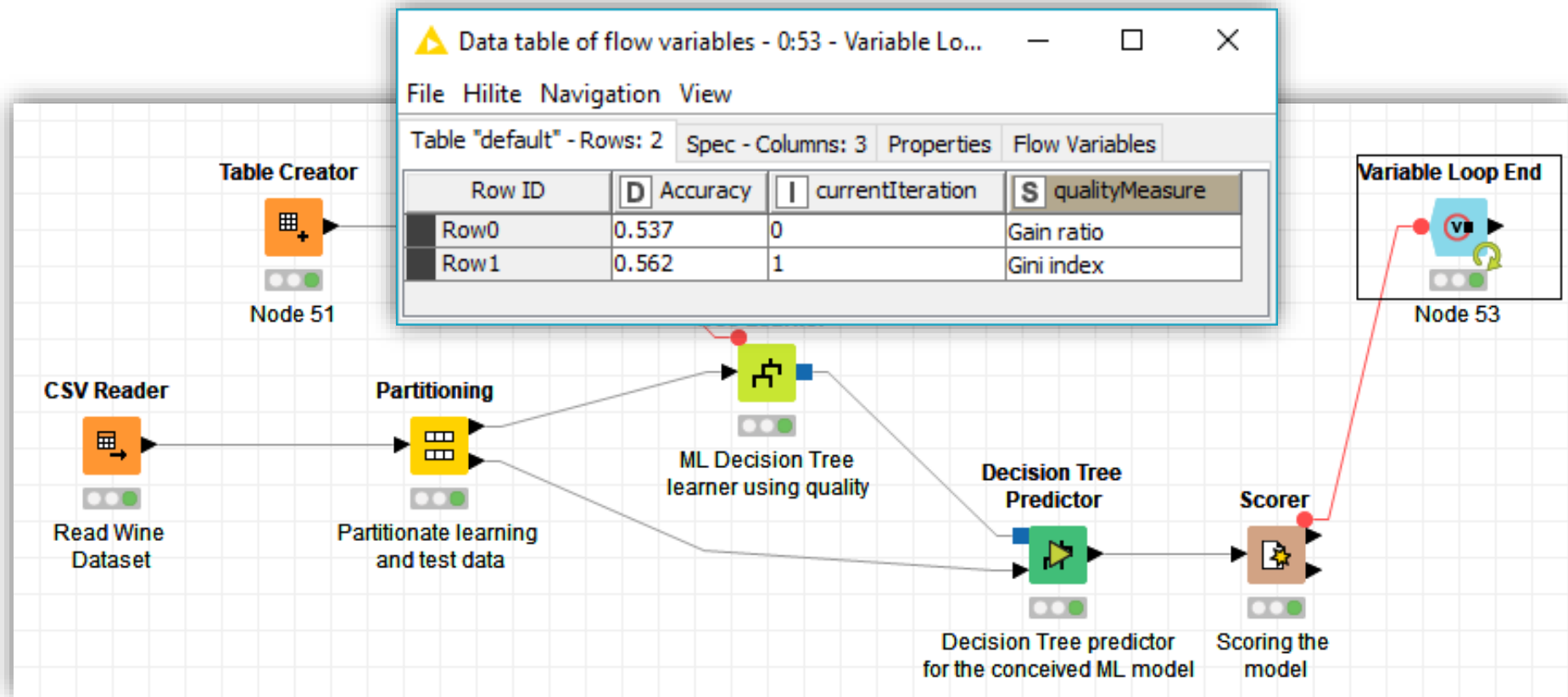
Obtained Results

22

Decision Trees

LOOPS

Hands On



Tuning Several Nominal Parameters

23

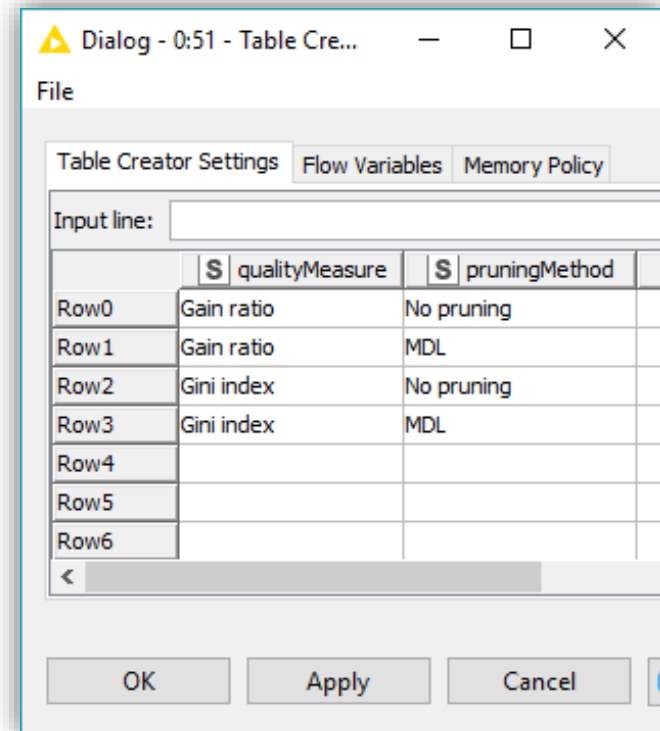
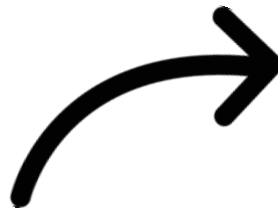
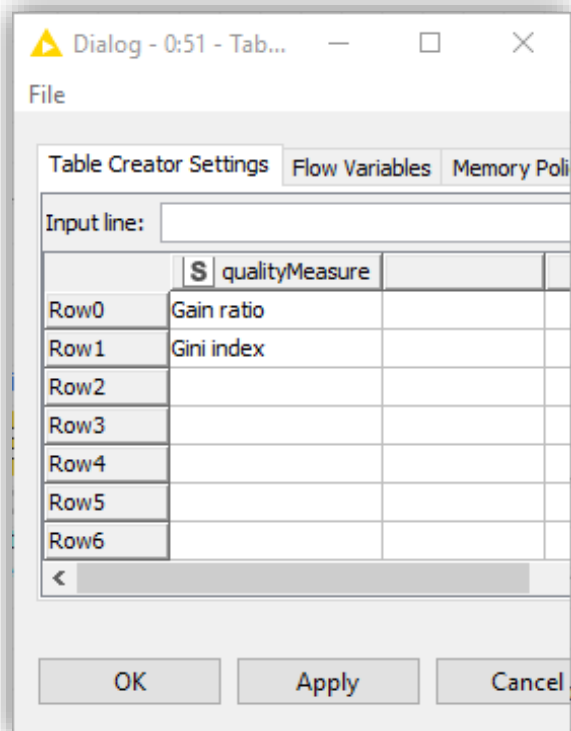
Decision Trees

LOOPS

Hands On

What If we want to **tune several nominal parameters** at once?

- Nested loops
- Specify all possible combinations in a table



Hands On

24

Decision Trees

Loops

HANDS ON

The screenshot displays the KNIME Analytics Platform interface. The top menu bar includes File, Edit, View, Node, and Help. Below the menu is a toolbar with various icons for file operations, workflow management, and execution. The main workspace is a large grid where a workflow can be built. On the left side, there are three panels: the 'KNIME Explorer' showing a project tree with folders like 'My-KNIME-Hub', 'EXAMPLES', 'LOCAL', and 'KNIME_Aula_Exercicio'; the 'Workflow Coach' with a list of 'Recommended Nodes' including File Reader, CSV Reader, Excel Reader, Table Creator, Database Reader, Table Reader, List Files, and Database Connection; and the 'Node Repository' with categories like IO, Manipulation, Views, Analytics, DB, Other Data Types, Structured Data, Scripting, Tools & Services, Workflow Control, Workflow Abstraction, and Reporting. On the right side, there is a 'Quick Access' panel with a search bar labeled 'Search workflows, nodes, and more...'. At the bottom, there is a 'Console' and 'Outline' panel. A large, bold, black 'HANDS ON' watermark is diagonally placed across the center of the workspace.