Orange Widgets to Python

Task 1: Training Wheels

Open the [Task 1 Google Collaboratory Notebook] and add it to your Google Drive so you can edit it.

Follow the instructions in the notebook.

Task 2: From Orange to Python

The data for this task is contained in two csv files:

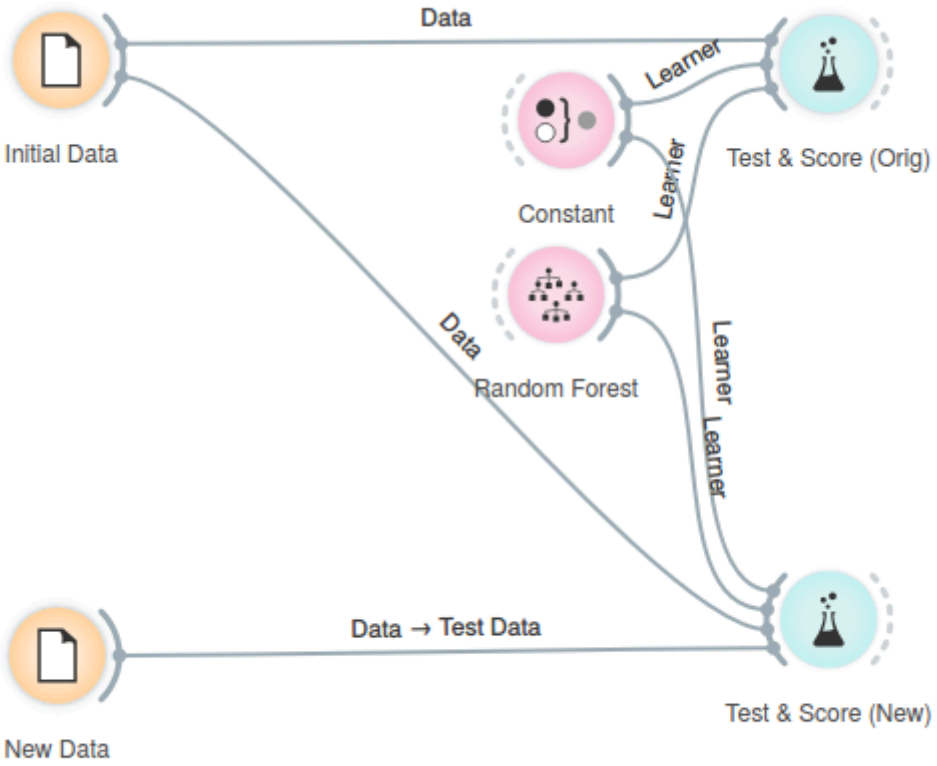
- Initial Data:
  `https://drive.google.com/uc?export=download&id=15djO0nRba6fpm0xtKGWBkU2Gf2AVaUqU`
- New Data:
  `https://drive.google.com/uc?export=download&id=17usoDKMyF7mO4EricY4qkpcVZCozTITg`

The data is comes from the same dataset as the demo. Although it has been slightly modified, the details that can be found with the original dataset are still valid:
http://archive.ics.uci.edu/ml/datasets/Wine+Quality

The flow you are required to implement in Python (in a new Google Collaboratory Notebook^) is as below and represents the case where we have trained and tested our model and come up with the parameters we want for the learners and then later on (say in a few months) we get some more data and can actually evaluate the generalized performance with a new, never seen before, data set.

^ Goto your google drive (google.drive.com) and then click "new", "more" then "Google Collaboratory"

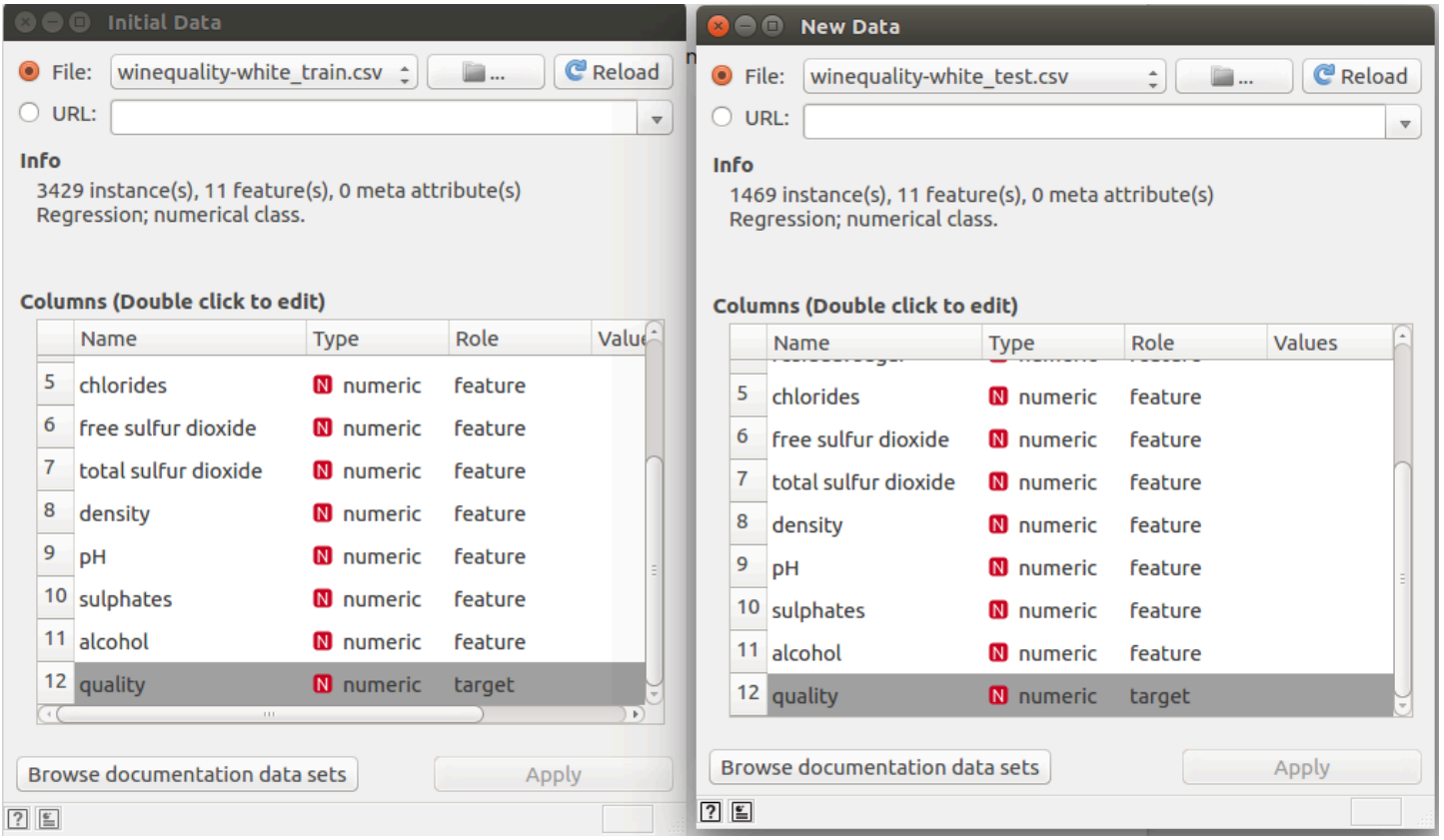
HINT: The csv files are in the format for Orange. Make sure you check to ensure you have loaded them correctly with pandas.read_csv - you'll probably need to read the document to work out how to make it work (take a look at the parameters *header* and *skiprows*)



Widgets have been configured as follows. Try and make your models as close as possible.

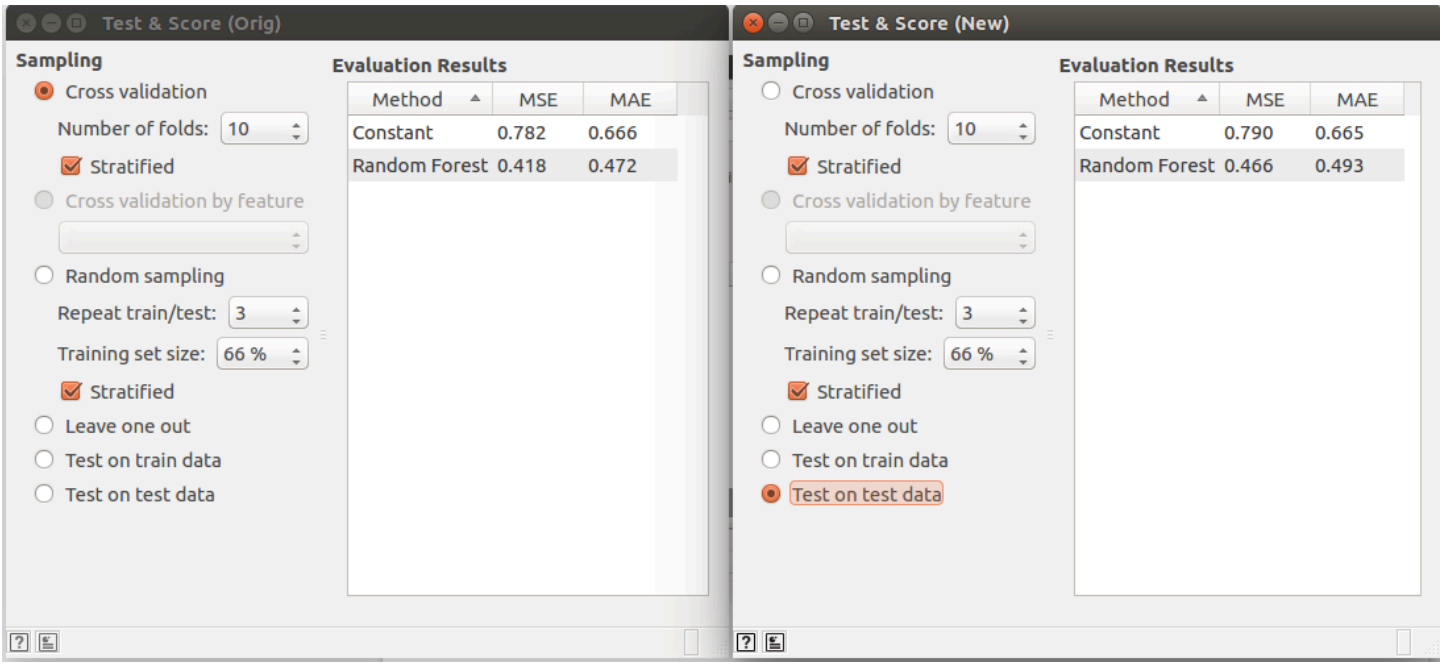
Initial Data

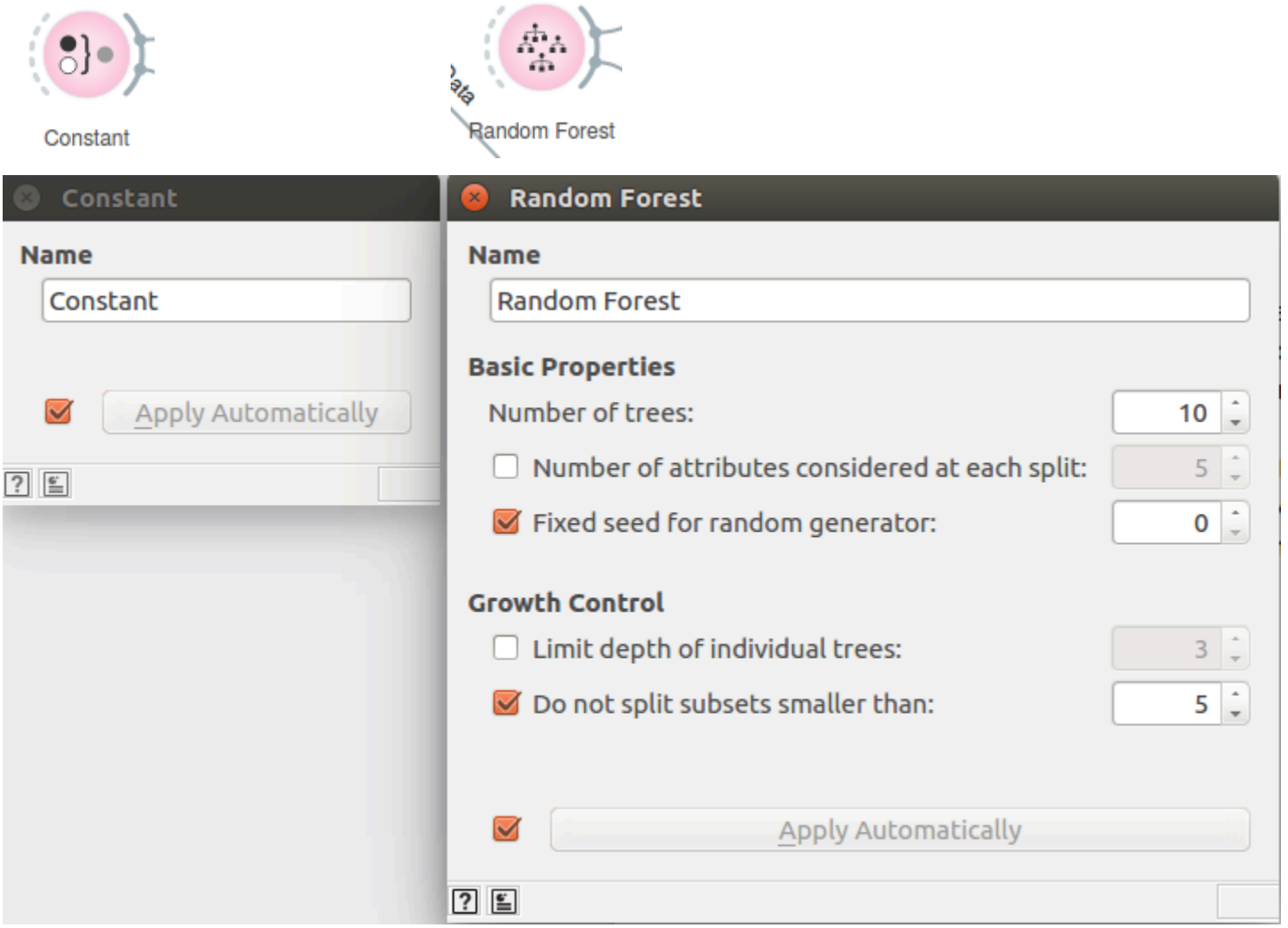
New Data



Test & Score (Orig)

Test & Score (New)




Constant
Random Forest
Constant
Name
Constant
Apply Automatically
Random Forest
Name
Random Forest
Basic Properties
Number of trees:
10
Number of attributes considered at each split:
5
Fixed seed for random generator:
0
Growth Control
Limit depth of individual trees:
3
Do not split subsets smaller than:
5
Apply Automatically

Task 3: From Task to Python

Task: Evaluate the performance of (at least) the following models for predicting whether bank notes are authentic (class = 1) or not (class = 0).

The dataset is available from: http://archive.ics.uci.edu/ml/datasets/banknote+authentication

HINT: Sketch out the sklearn workflow in advance.

Models to evaluate (feel free to read the docs and try more):

- Random Forest
- Logistic Regression
- Constant/Majority Learner