



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=15dj00nRba6fpm0xtKGWBkU2Gf2AVaUqU>
- New Data:
<https://drive.google.com/uc?export=download&id=17usoDKMyF7mO4EricY4qkpcVZCozTITg>

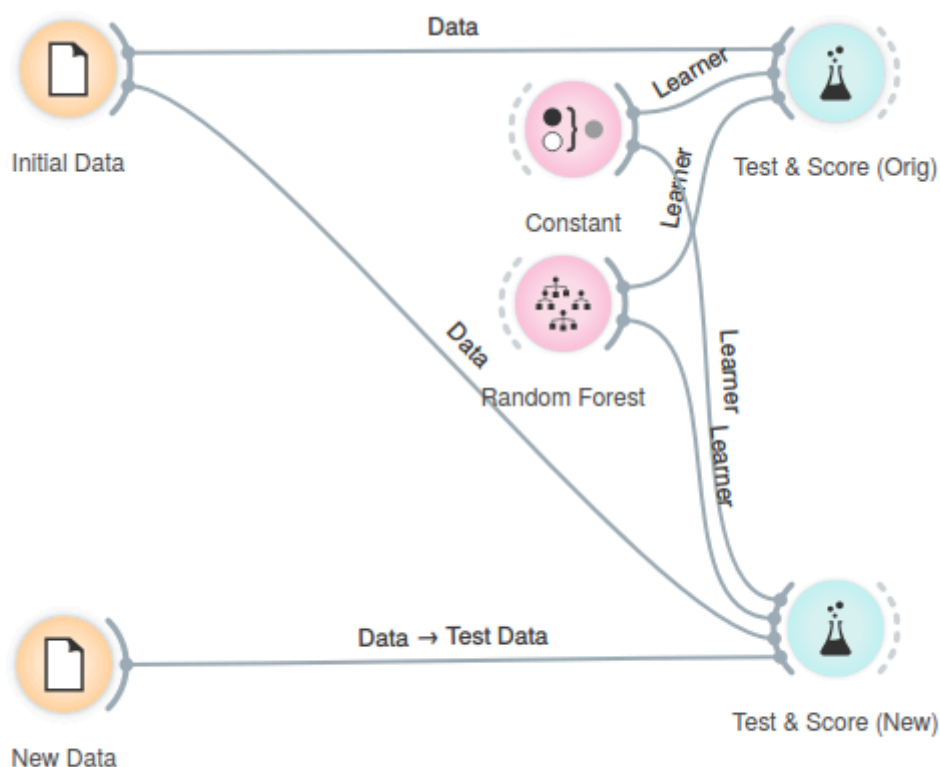
The data is comes from the same dataset as the demo. Although it has been slightly modified, the details that can be found with the original dataset are still valid:

<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

Initial Data

File: winequality-white_train.csv

URL:

Info
3429 instance(s), 11 feature(s), 0 meta attribute(s)
Regression; numerical class.

Columns (Double click to edit)

	Name	Type	Role	Value
5	chlorides	N numeric	feature	
6	free sulfur dioxide	N numeric	feature	
7	total sulfur dioxide	N numeric	feature	
8	density	N numeric	feature	
9	pH	N numeric	feature	
10	sulphates	N numeric	feature	
11	alcohol	N numeric	feature	
12	quality	N numeric	target	

New Data

File: winequality-white_test.csv

URL:

Info
1469 instance(s), 11 feature(s), 0 meta attribute(s)
Regression; numerical class.

Columns (Double click to edit)

	Name	Type	Role	Values
5	chlorides	N numeric	feature	
6	free sulfur dioxide	N numeric	feature	
7	total sulfur dioxide	N numeric	feature	
8	density	N numeric	feature	
9	pH	N numeric	feature	
10	sulphates	N numeric	feature	
11	alcohol	N numeric	feature	
12	quality	N numeric	target	



Test & Score (Orig)



Test & Score (New)

Test & Score (Orig)

Sampling
☒ Cross validation
Number of folds: 10
☒ Stratified
☐ Cross validation by Feature
☐ Random sampling
Repeat train/test: 3
Training set size: 66 %
☒ Stratified
☐ Leave one out
☐ Test on train data
☐ Test on test data

Evaluation Results

Method	MSE	MAE
Constant	0.782	0.666
Random Forest	0.418	0.472

Test & Score (New)

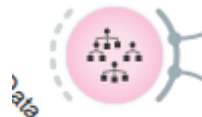
Sampling
☐ Cross validation
Number of folds: 10
☒ Stratified
☐ Cross validation by Feature
☐ Random sampling
Repeat train/test: 3
Training set size: 66 %
☒ Stratified
☐ Leave one out
☐ Test on train data
☒ Test on test data

Evaluation Results

Method	MSE	MAE
Constant	0.790	0.665
Random Forest	0.466	0.493



Constant



Random Forest

✕ Constant

Name

☒

?

✕ Random Forest

Name

Basic Properties
Number of trees:
☐ Number of attributes considered at each split:
☒ Fixed seed for random generator:

Growth Control
☐ Limit depth of individual trees:
☒ Do not split subsets smaller than:
☒

?

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