MIS772
Predictive Analytics

Workshop: Advanced Classification
Classification with cross-validation, and ensembles

AACSB ACCREDITED

EFMD EQUIS ACCREDITED

DEAKIN BUSINESS SCHOOL

Workshop Plan

Objectives:
The task is to improve previously developed predictive model classifying all Danish AirBnB rental properties into "cheap" (price/night < $100) and "expensive".

Data Set:

AirBNB-DK.csv (From Unit Website)

Acknowledgements: *http://tomslee.net*

Method:
Attend the workshop, follow the tutor's demo and instructions, take notes. Note that the class and online seminar will be recorded and their videos linked to the CloudDeakin topic for later access and study.

1 **Previous workshop – be ready**
   (a) Load the data
   (b) Prepare data and explore
   (c) Split data for model training and validation
   (d) Add k-NN/DT model and apply
   (e) Measure the model performance

2 **Cross-validate the model**
   (a) Load the previous classification process
   (b) Replace data split with cross-validation (k-fold)
   (c) Add k-NN/DT or other classifiers, apply and performance inside the cross-validation
   (d) Run and investigate the performance, save!

3 **Experiment with model parameters**
   (a) Experiment with model parameters and save best model

4 **Experiment with ensembles**
   (a) Replace single classifier with ensemble
   (b) Explore performance

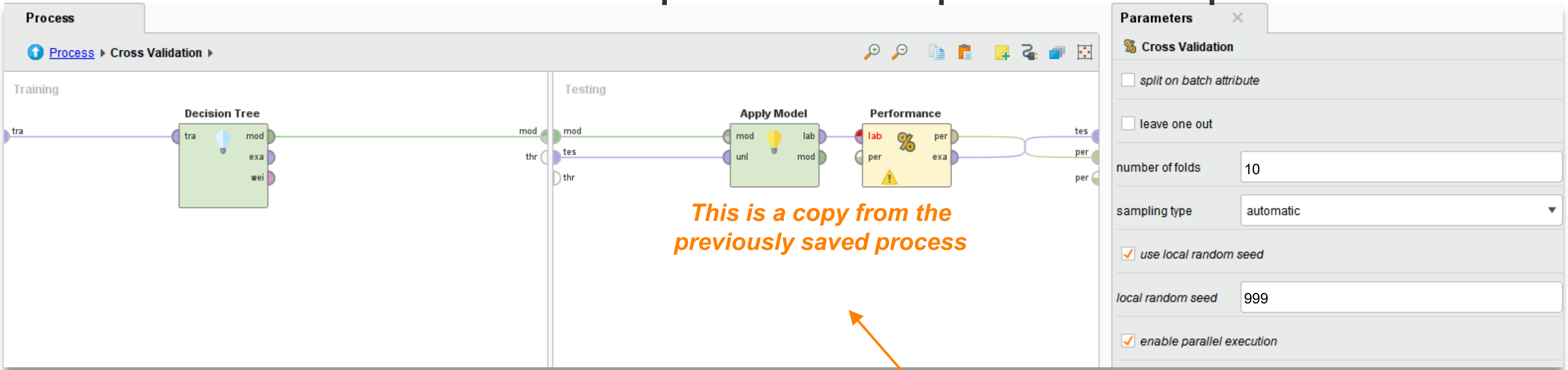
Cross-Validation

We will start by opening the process saved in the last workshop. Replace “Split Data” with “Cross Validation” and relocate the model, its application and performance operators appropriately. Use CV 3 folds. Use DT pruned to max depth 5. Run and explore the results.

We will extend the previous process by adding cross-validation.

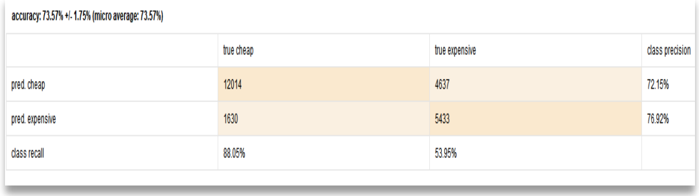
What results does it produce?

How does it work?



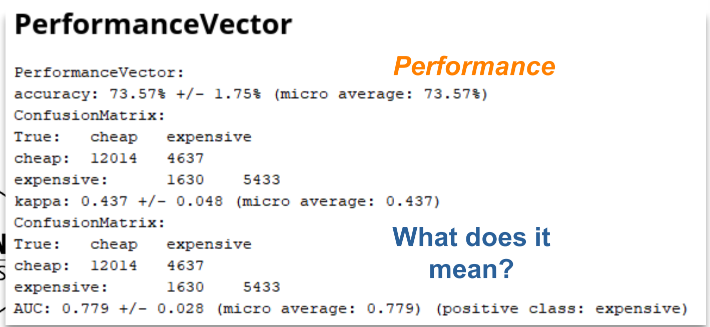
split on batch attribute ☐
leave one out ☐
number of folds: 10
sampling type: automatic
use local random seed ☑
local random seed: 999
enable parallel execution ☑

This is a copy from the previously saved process



accuracy: 73.57% +/- 1.75% (micro average: 73.57%)

	true cheap	true expensive	class precision
pred. cheap	12014	4637	72.15%
pred. expensive	1630	5433	76.92%
class recall	88.05%	53.95%	

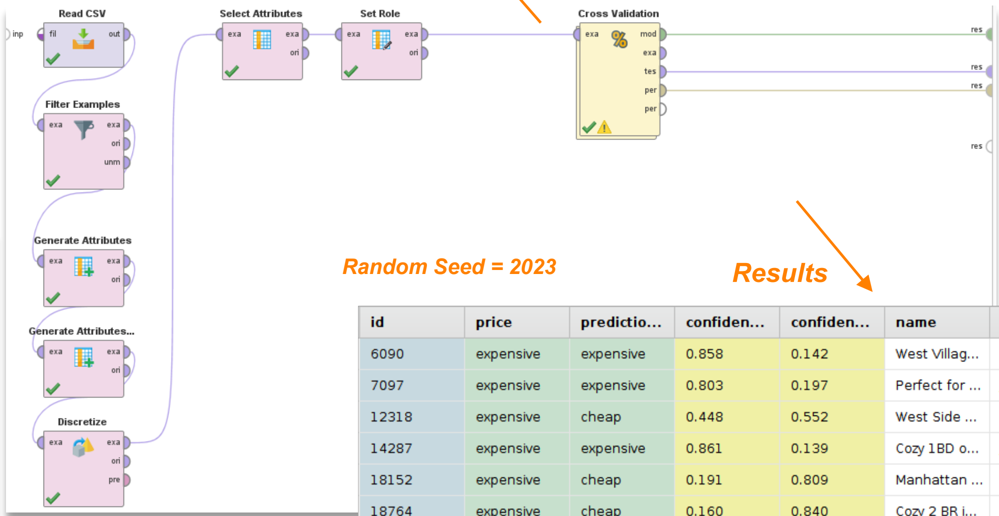


PerformanceVector

```
PerformanceVector:
accuracy: 73.57% +/- 1.75% (micro average: 73.57%)
ConfusionMatrix:
True:   cheap   expensive
cheap:  12014   4637
expensive:      1630    5433
kappa: 0.437 +/- 0.048 (micro average: 0.437)
ConfusionMatrix:
True:   cheap   expensive
cheap:  12014   4637
expensive:      1630    5433
AUC: 0.779 +/- 0.028 (micro average: 0.779) (positive class: expensive)
```

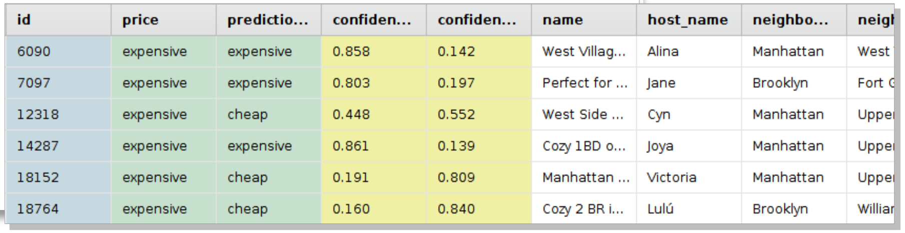
Performance

What does it mean?



Random Seed = 2023

Results



id	price	predictio...	confiden...	confiden...	name	host_name	neighbo...	neigl
6090	expensive	expensive	0.858	0.142	West Villag...	Alina	Manhattan	West
7097	expensive	expensive	0.803	0.197	Perfect for ...	Jane	Brooklyn	Fort C
12318	expensive	cheap	0.448	0.552	West Side ...	Cyn	Manhattan	Uppe
14287	expensive	expensive	0.861	0.139	Cozy 1BD o...	Joya	Manhattan	Uppe
18152	expensive	cheap	0.191	0.809	Manhattan ...	Victoria	Manhattan	Uppe
18764	expensive	cheap	0.160	0.840	Cozy 2 BR i...	Lulú	Brooklyn	Willia

What can you say about the model cross-validated vs holdout performance? Which would you choose?

DEAKIN BUSINESS SCHOOL

How can you improve the model's performance, given your objective?

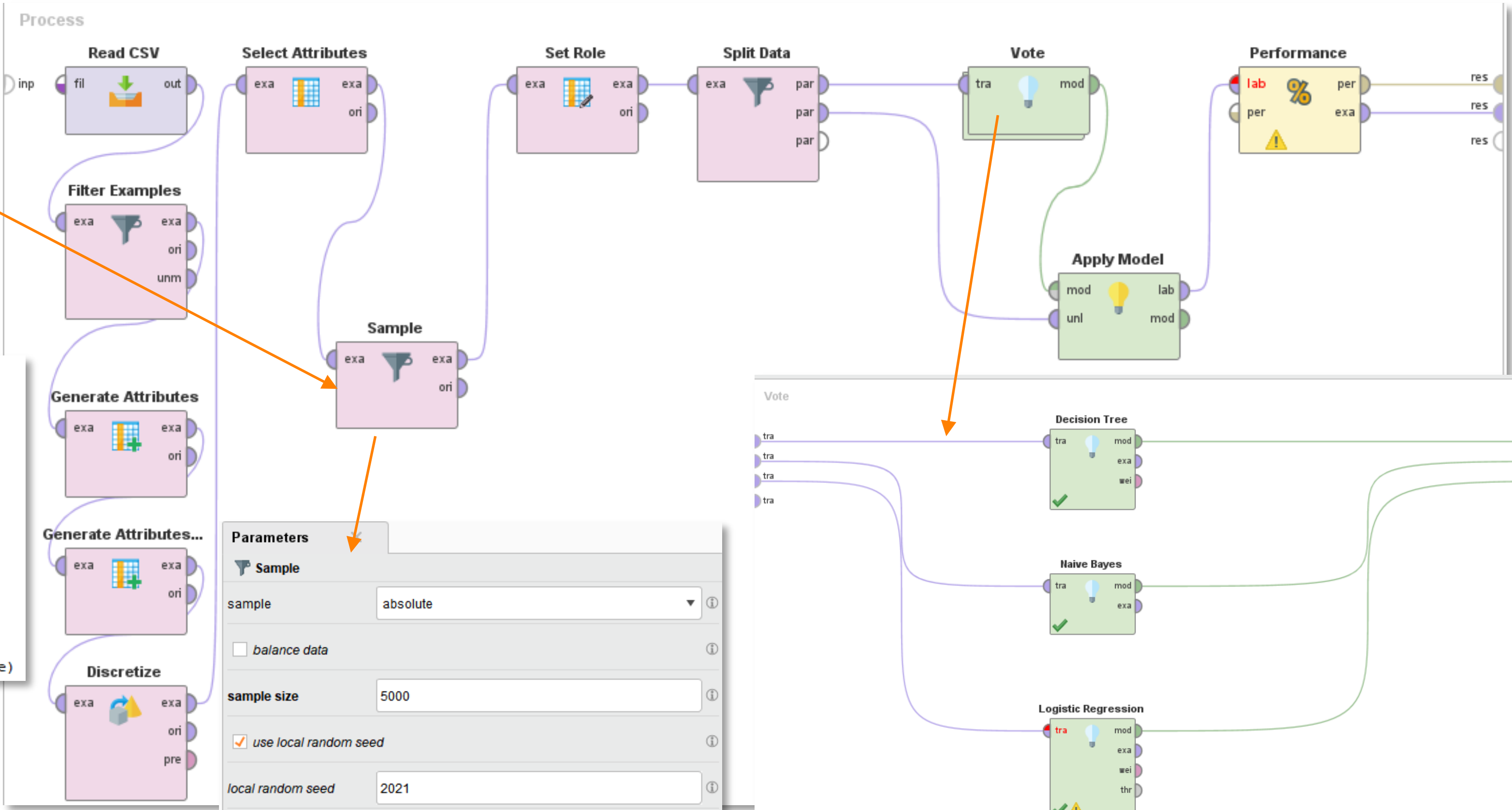
Will the process work with the current selection of attributes (bedrooms, room_type, etc.) if you use Logistic Regression? If not, what are our options?

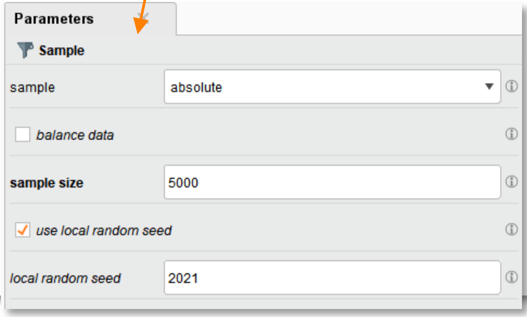
Try different classifiers, e.g. DT, k-NN, Naive Bayes, Logistic Regression
Try different parameters for the classifiers (e.g., different values of k, different pruning parameters)
Or…try an ensemble model!

Ensembles:

Replace single classifier with an ensemble model comprising multiple base-classifiers (e.g., DT, Naïve Bayes & Logistic Regression) and use Voting to aggregate prediction. Compare ensemble performance against previous single-models.

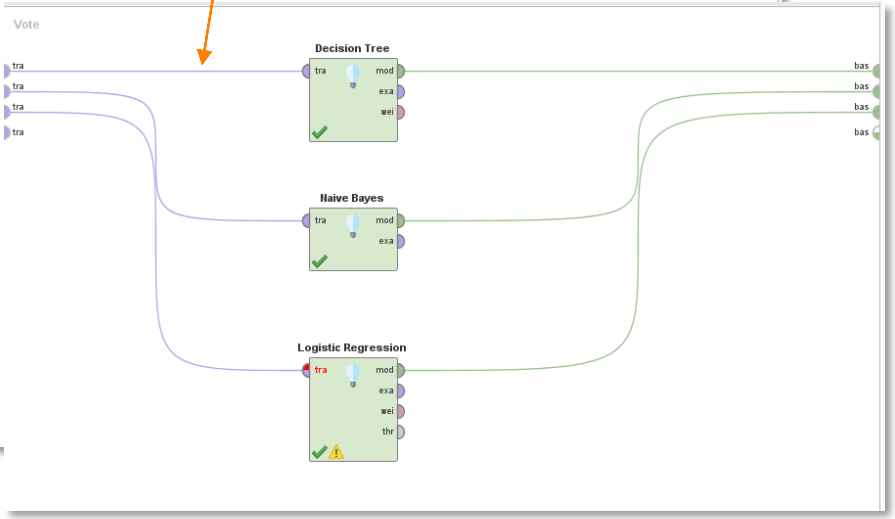
Note: Down sample the dataset (use Sample operator with 5000 samples, fix local random seed) as the full dataset might take a long time to run.

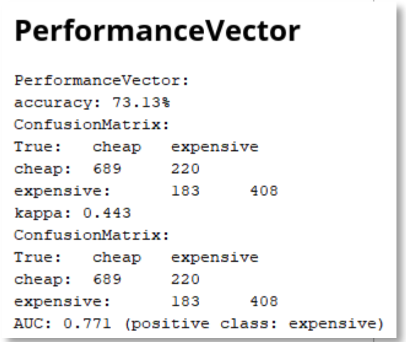



Parameters

Sample

sample	absolute
balance data	☐
sample size	5000
use local random seed	✓
local random seed	2021




PerformanceVector

```
PerformanceVector:
accuracy: 73.13%
ConfusionMatrix:
True:     cheap    expensive
cheap:    689      220
expensive:         183      408
kappa: 0.443
ConfusionMatrix:
True:     cheap    expensive
cheap:    689      220
expensive:         183      408
AUC: 0.771 (positive class: expensive)
```

Note: Performance based on sample of 5000