

```
import pandas as pd
from sklearn.datasets import make_multilabel_classification
from sklearn.multioutput import MultiOutputClassifier, ClassifierChain
from atom import ATOMClassifier
```

```
# Create data
X, y = make_multilabel_classification(n_samples=300, n_classes=3,
random_state=1)
```

```
# Note that for multioutput tasks, you must specify the `y` keyword
atom = ATOMClassifier(X, y=y, verbose=2, random_state=1)
atom.multioutput = MultiOutputClassifier
atom.run(models='RF')
```

```
<< ===== ATOM ===== >>
```

```
Algorithm task: multilabel classification.
```

```
Dataset stats ===== >>
```

```
Shape: (300, 23)
```

```
Train set size: 240
```

```
Test set size: 60
```

```
-----
```

```
Memory: 55.33 kB
```

```
Scaled: False
```

```
Outlier values: 29 (0.5%)
```

```
Training ===== >>
```

```
Models: RF
```

```
Metric: average_precision
```

```
Results for RandomForest:
```

```
Fit -----
```

```
Train evaluation --> average_precision: 1.0
```

```
Test evaluation --> average_precision: 0.6468
```

```
Time elapsed: 0.300s
```

```
-----
```

```
Total time: 0.300s
```

```
Final results ===== >>
```

```
Total time: 0.302s
```

```
-----
```

```
RandomForest --> average_precision: 0.6468 ~
```

```
# Note that for multioutput tasks, you must specify the `y` keyword
atom1 = ATOMClassifier(X, y=y, verbose=2, random_state=1)
atom1.multioutput = ClassifierChain
atom1.run(models='RF')
```

```
<< ===== ATOM ===== >>
Algorithm task: multilabel classification.
```

```
Dataset stats ===== >>
```

```
Shape: (300, 23)
Train set size: 240
Test set size: 60
```

```
-----
Memory: 55.33 kB
Scaled: False
Outlier values: 29 (0.5%)
```

```
Training ===== >>
```

```
Models: RF
Metric: average_precision
```

```
Results for RandomForest:
```

```
Fit -----
```

```
Train evaluation --> average_precision: 1.0
Test evaluation --> average_precision: 0.6468
Time elapsed: 0.305s
```

```
-----
Total time: 0.305s
```

```
Final results ===== >>
```

```
Total time: 0.306s
-----
RandomForest --> average_precision: 0.6468 ~
```

```
# Note that for multioutput tasks, you must specify the `y` keyword
atom2 = ATOMClassifier(X, y=y, verbose=2, random_state=1)
atom2.multioutput = None
atom2.run(models='RF')
```

```
<< ===== ATOM ===== >>
Algorithm task: multilabel classification.
```

```
Dataset stats ===== >>
Shape: (300, 23)
Train set size: 240
Test set size: 60
-----
Memory: 55.33 kB
Scaled: False
Outlier values: 29 (0.5%)

Training ===== >>
Models: RF
Metric: average_precision

Results for RandomForest:
Fit -----
Train evaluation --> average_precision: 1.0
Test evaluation --> average_precision: 0.6468
Time elapsed: 0.393s
-----
Total time: 0.393s

Final results ===== >>
Total time: 0.396s
-----
RandomForest --> average_precision: 0.6468 ~
```