MEKA: A Multi-label Extension to WEKA

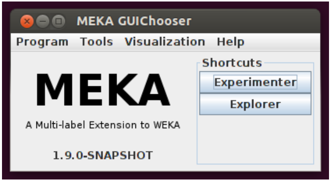
Jesse Read

ÉCOLE POLYTECHNIQUE
IP PARIS

Te Ipu o te Mahara
Artificial Intelligence Institute
THE UNIVERSITY OF WAIKATO

2021 International WEKA User Conference
November 26, 2021

MEKA: A Multi-label Extension to WEKA



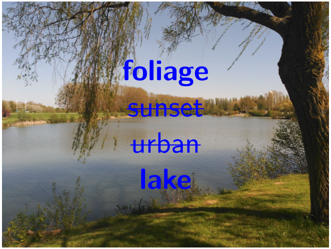
MEKA is a framework which adapts and wraps WEKA classifiers and methods to the problem of multi-label classification.

- Started at the Univ. of Waikato ca. 2008 from PhD research
- Peter Reutemann turned it into a proper software framework
- Published in JMLR-MLOSS in 2016[1]
- Additional input from Joerg Wicker (Univ. of Auckland), and others from the open-source community

[1] Jesse Read, Peter Reutemann, Bernhard Pfahringer, and Geoff Holmes. "MEKA: A Multi-label/Multi-target Extension to Weka". In: *Journal of Machine Learning Research* 17.21 (2016), pp. 1–5. URL: http://waikato.github.io/meka/

Multi-label Classification

Multi-label classification: *multiple* labels (decisions) per instance (multi-*class* classification = a single label per instance).

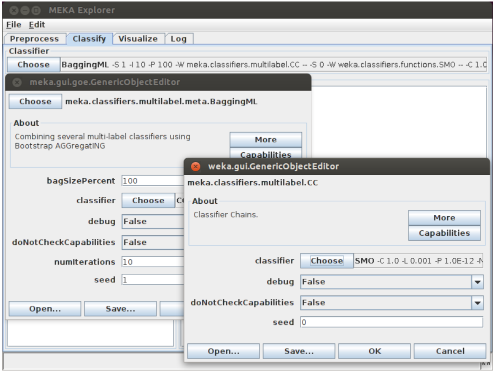


but it is much more than just labeling images! MEKA been used for: text/sentiment/image classification, RNA binding, bird song classification, call routing, wineinformatics, cancer diagnosis, neuroscience, recommender systems, time series forecasting, missing value analysis, . . .

Multi-label Classification via Problem Transformation

MEKA is designed around problem transformation methods:

1. Take a multi-label problem
2. Transform it into one or more single-label problems
3. Choose and apply WEKA classifiers
4. Evaluate and export results



Classifier: BaggingML -S 1 -I 10 -P 100 -W meka.classifiers.multilabel.CC -- -S 0 -W weka.classifiers.functions.SMO -- -C 1.0

meka.classifiers.multilabel.meta.BaggingML

About: Combining several multi-label classifiers using Bootstrap AGGregatING

bagSizePercent: 100
classifier: C
debug: False
doNotCheckCapabilities: False
numIterations: 10
seed: 1

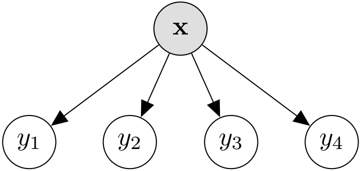
meka.classifiers.multilabel.CC

About: Classifier Chains.

classifier: SMO -C 1.0 -L 0.001 -P 1.0E-12 -N
debug: False
doNotCheckCapabilities: False
seed: 0

Transformation Example #1 – Binary Relevance

$\boldsymbol{X}$	Y_1	Y_2	Y_3	Y_4
$\boldsymbol{x}^{(1)}$	0	1	1	0
$\boldsymbol{x}^{(2)}$	1	0	0	0
$\boldsymbol{x}^{(3)}$	0	1	0	0
$\boldsymbol{x}^{(4)}$	1	0	0	1
$\boldsymbol{x}^{(5)}$	0	0	0	1
$\tilde{\boldsymbol{x}}$	?	?	?	?



The binary relevance method (BR transformation) $=$ *one binary classifier trained for each label*, i.e., independent models.

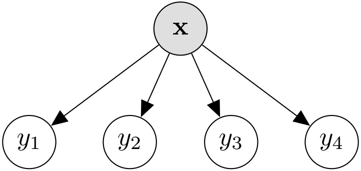
Transformation Example #1 – Binary Relevance

$\boldsymbol{X}$	Y_1
$\boldsymbol{x}^{(1)}$	0
$\boldsymbol{x}^{(2)}$	1
$\boldsymbol{x}^{(3)}$	0
$\boldsymbol{x}^{(4)}$	1
$\boldsymbol{x}^{(5)}$	0
$\tilde{\boldsymbol{x}}$	?

$\boldsymbol{X}$	Y_2
$\boldsymbol{x}^{(1)}$	1
$\boldsymbol{x}^{(2)}$	0
$\boldsymbol{x}^{(3)}$	1
$\boldsymbol{x}^{(4)}$	0
$\boldsymbol{x}^{(5)}$	0
$\tilde{\boldsymbol{x}}$	?

$\boldsymbol{X}$	Y_3
$\boldsymbol{x}^{(1)}$	0
$\boldsymbol{x}^{(2)}$	1
$\boldsymbol{x}^{(3)}$	0
$\boldsymbol{x}^{(4)}$	1
$\boldsymbol{x}^{(5)}$	0
$\tilde{\boldsymbol{x}}$	?

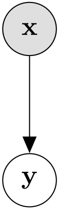
$\boldsymbol{X}$	Y_4
$\boldsymbol{x}^{(1)}$	1
$\boldsymbol{x}^{(2)}$	0
$\boldsymbol{x}^{(3)}$	1
$\boldsymbol{x}^{(4)}$	0
$\boldsymbol{x}^{(5)}$	0
$\tilde{\boldsymbol{x}}$	?



The binary relevance method (BR transformation) $=$ *one binary classifier trained for each label*, i.e., independent models.

Transformation Example #2 – Label Powerset Method

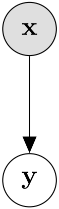
$\boldsymbol{X}$	Y_1	Y_2	Y_3	Y_4
$\boldsymbol{x}^{(1)}$	0	1	1	0
$\boldsymbol{x}^{(2)}$	1	0	0	0
$\boldsymbol{x}^{(3)}$	0	1	0	0
$\boldsymbol{x}^{(4)}$	1	0	0	1
$\boldsymbol{x}^{(5)}$	0	0	0	1
$\tilde{\boldsymbol{x}}$	?	?	?	?



The label powerset method (LP transformation) = *a single target multi-class classifier*. Labels are modeled together.

Transformation Example #2 – Label Powerset Method

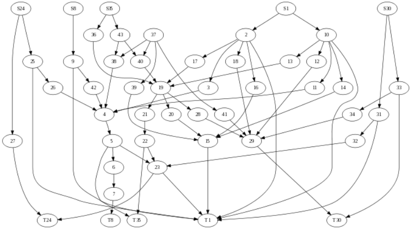
$\boldsymbol{X}$	Y
$\boldsymbol{x}^{(1)}$	0 1 1 0
$\boldsymbol{x}^{(2)}$	1 0 0 0
$\boldsymbol{x}^{(3)}$	0 1 0 0
$\boldsymbol{x}^{(4)}$	1 0 0 1
$\boldsymbol{x}^{(5)}$	0 0 0 1
$\tilde{\boldsymbol{x}}$	?



The label powerset method (LP transformation) = *a single target multi-class classifier*. Labels are modeled together.

Why Use MEKA (over just WEKA)? Reason #1

Things can get complicated:



MEKA contains 30–50 transformation methods (depending on your definition of a method);

- RAkEL: Random *k*-labeled subsets (LP-based)
- PS: Pruned sets (LP-based)
- CC: Classifier chains (BR-based)
- CDN: Conditional dependency networks (BR-based)
- . . .

Why Use MEKA (over just WEKA)? Reason #2

You made your predictions,

Test instance	Test labels	Prediction
$\tilde{\boldsymbol{x}}^{(1)}$	[1 0 1 0]	[1 0 0 1]
$\tilde{\boldsymbol{x}}^{(2)}$	[0 1 0 1]	[1 0 1 0]
$\tilde{\boldsymbol{x}}^{(3)}$	[1 0 0 1]	[1 0 0 1]
$\tilde{\boldsymbol{x}}^{(4)}$	[0 1 1 0]	[0 1 0 0]
$\tilde{\boldsymbol{x}}^{(5)}$	[1 0 0 0]	[1 0 0 1]

what is your ‘accuracy’?

MEKA contains over 20 evaluation metrics:

- Hamming score (evaluate each binary prediction separately)
- Exact match (prediction vector must match *exactly*)
- ...

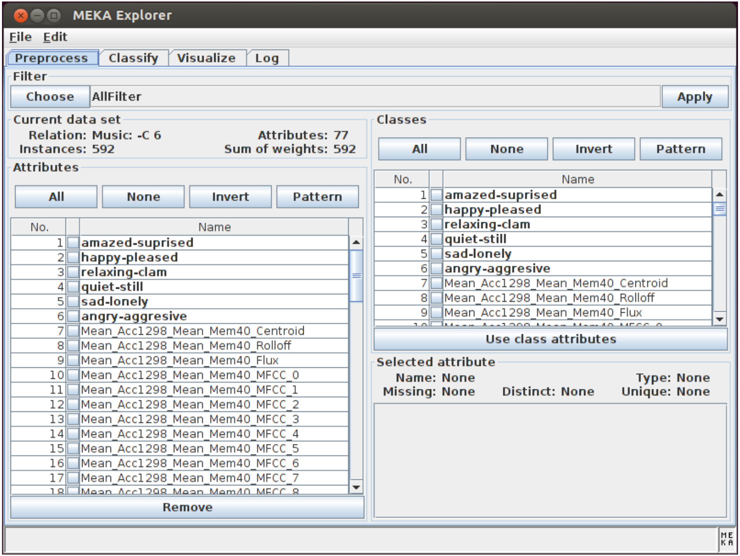
and threshold-tuning functionality (many classifiers implement `distributionForInstance`), e.g., [0.9,0.1,0.4,0.8].

Major implications depending on which metric you are targeting!

MEKA ARFF Files

```
% 'Music' dataset; normalised version.
@relation 'Music: -C 6'

@attribute amazed-suprised {0,1}
@attribute happy-pleased {0,1}
@attribute relaxing-clam {0,1}
@attribute quiet-still {0,1}
@attribute sad-lonely {0,1}
@attribute angry-aggresive {0,1}
@attribute MeanAcc1298MeanMem40Centroid numeric
@attribute MeanAcc1298MeanMem40Rolloff numeric
@attribute MeanAcc1298MeanMem40Flux numeric
@attribute BHLowPeakAmp numeric
@attribute BHLowPeakBPM numeric
@attribute BHHighPeakAmp numeric
@attribute BHSUM1 numeric
@attribute BHSUM2 numeric
@attribute BHSUM3 numeric

@data

0,1,1,0,0,0,0.132498,0.077848,0.229227,0.602629,0.512861,0.467404,0.529733,0.573498,0.592831
1,0,0,0,0,1,0.384281,0.355249,0.16719,0.853089,0.260577,0.332757,0.15393,0.519381,0.268043
0,1,0,0,0,1,0.541782,0.356491,0.152246,0.791142,0.228276,0.471278,0.378166,0.559905,0.279949
0,0,1,0,0,0,0.174288,0.243935,0.254326,0.438987,0.480346,0.472862,0.473128,0.777076,0.376471
0,0,0,1,0,0,0.347436,0.155448,0.100047,0.126026,0.45695,0.539992,0.301537,0.598898,0.36199
...
```

MEKA GUI



Filter: AllFilter

Current data set

Relation: Music: -C 6
Instances: 592
Attributes: 77
Sum of weights: 592

Attributes

No.		Name
1	□	**amazed-suprised**
2	□	**happy-pleased**
3	□	**relaxing-clam**
4	□	**quiet-still**
5	□	**sad-lonely**
6	□	**angry-aggresive**
7	□	Mean_Acc1298_Mean_Mem40_Centroid
8	□	Mean_Acc1298_Mean_Mem40_Rolloff
9	□	Mean_Acc1298_Mean_Mem40_Flux
10	□	Mean_Acc1298_Mean_Mem40_MFCC_0
11	□	Mean_Acc1298_Mean_Mem40_MFCC_1
12	□	Mean_Acc1298_Mean_Mem40_MFCC_2
13	□	Mean_Acc1298_Mean_Mem40_MFCC_3
14	□	Mean_Acc1298_Mean_Mem40_MFCC_4
15	□	Mean_Acc1298_Mean_Mem40_MFCC_5
16	□	Mean_Acc1298_Mean_Mem40_MFCC_6
17	□	Mean_Acc1298_Mean_Mem40_MFCC_7
18	□	Mean_Acc1298_Mean_Mem40_MFCC_8

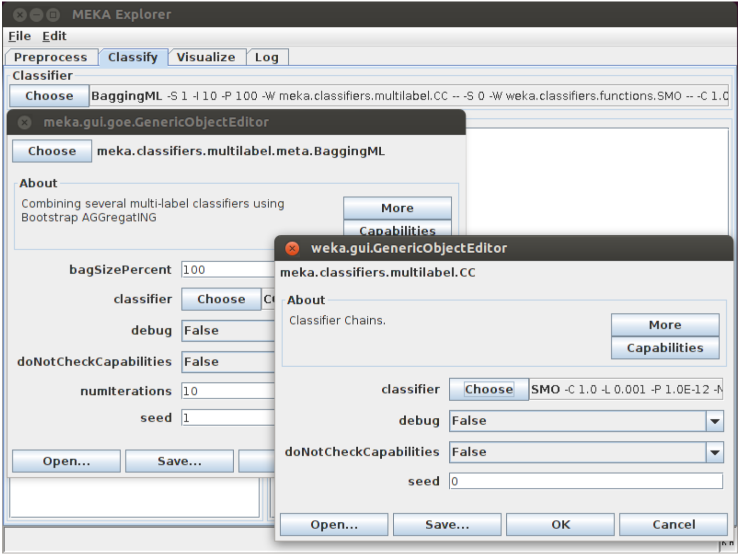
Classes

No.		Name
1	□	**amazed-suprised**
2	□	**happy-pleased**
3	□	**relaxing-clam**
4	□	**quiet-still**
5	□	**sad-lonely**
6	□	**angry-aggresive**
7	□	Mean_Acc1298_Mean_Mem40_Centroid
8	□	Mean_Acc1298_Mean_Mem40_Rolloff
9	□	Mean_Acc1298_Mean_Mem40_Flux
10	□	Mean_Acc1298_Mean_Mem40_MFCC_0

Selected attribute

Name: None
Missing: None
Distinct: None
Type: None
Unique: None

MEKA GUI



BaggingML -S 1 -I 10 -P 100 -W meka.classifiers.multilabel.CC -- -S 0 -W weka.classifiers.functions.SMO -- -C 1.0

meka.classifiers.multilabel.meta.BaggingML

Combining several multi-label classifiers using Bootstrap AGGregatING

bagSizePercent 100

classifier CC

debug False

doNotCheckCapabilities False

numIterations 10

seed 1

meka.classifiers.multilabel.CC

Classifier Chains.

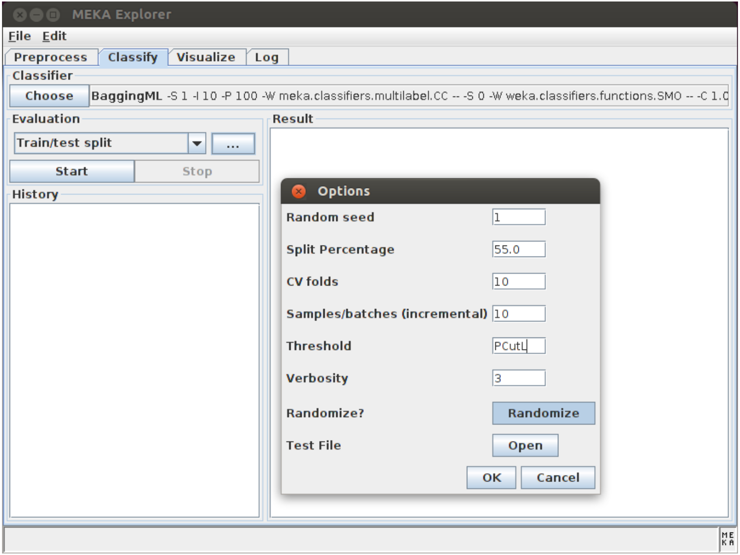
classifier SMO -C 1.0 -L 0.001 -P 1.0E-12 -N

debug False

doNotCheckCapabilities False

seed 0

MEKA GUI



BaggingML -S 1 -I 10 -P 100 -W meka.classifiers.multilabel.CC -- -S 0 -W weka.classifiers.functions.SMO -- -C 1.0

Train/test split

Random seed: 1

Split Percentage: 55.0

CV folds: 10

Samples/batches (incremental): 10

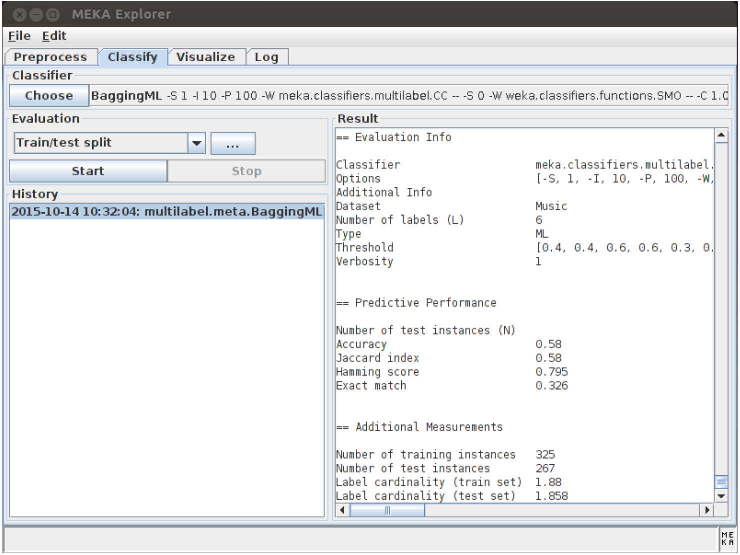
Threshold: PCutL

Verbosity: 3

Randomize? Randomize

Test File Open

MEKA GUI

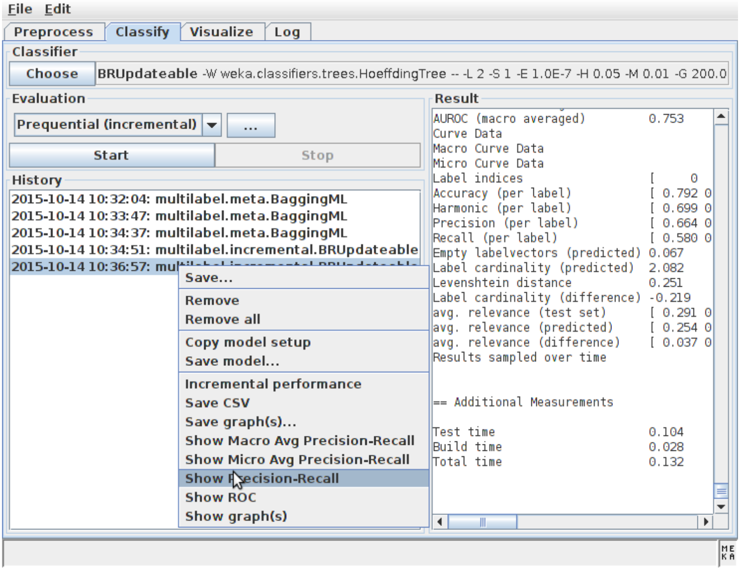


BaggingML -S 1 -I 10 -P 100 -W meka.classifiers.multilabel.CC -- -S 0 -W weka.classifiers.functions.SMO -- -C 1.0

2015-10-14 10:32:04: multilabel.meta.BaggingML

```
== Evaluation Info

Classifier                      meka.classifiers.multilabel.
Options                         [-S, 1, -I, 10, -P, 100, -W,
Additional Info
Dataset                         Music
Number of labels (L)            6
Type                            ML
Threshold                       [0.4, 0.4, 0.6, 0.6, 0.3, 0.
Verbosity                       1


== Predictive Performance

Number of test instances (N)
Accuracy                        0.58
Jaccard index                   0.58
Hamming score                   0.795
Exact match                     0.326


== Additional Measurements

Number of training instances    325
Number of test instances        267
Label cardinality (train set)   1.88
Label cardinality (test set)    1.858
```

MEKA GUI

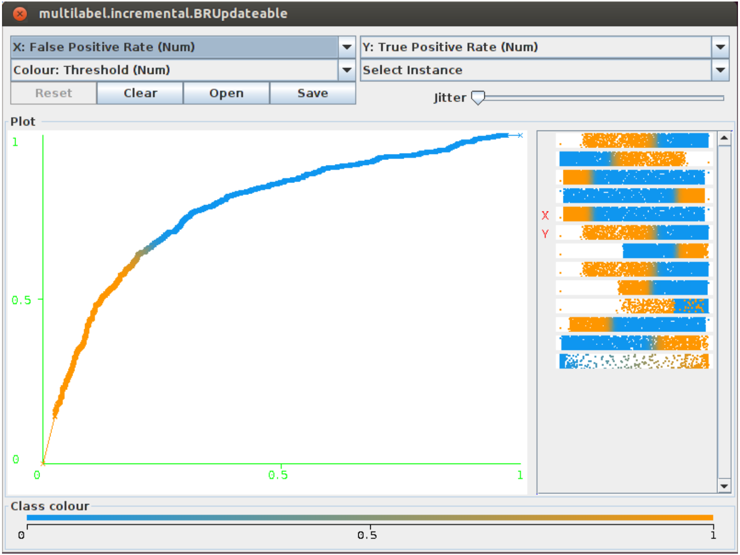


BRUpdateable -W weka.classifiers.trees.HoeffdingTree -- -L 2 -S 1 -E 1.0E-7 -H 0.05 -M 0.01 -G 200.0

2015-10-14 10:32:04: multilabel.meta.BaggingML
2015-10-14 10:33:47: multilabel.meta.BaggingML
2015-10-14 10:34:37: multilabel.meta.BaggingML
2015-10-14 10:34:51: multilabel.incremental.BRUpdateable
2015-10-14 10:36:57: mu

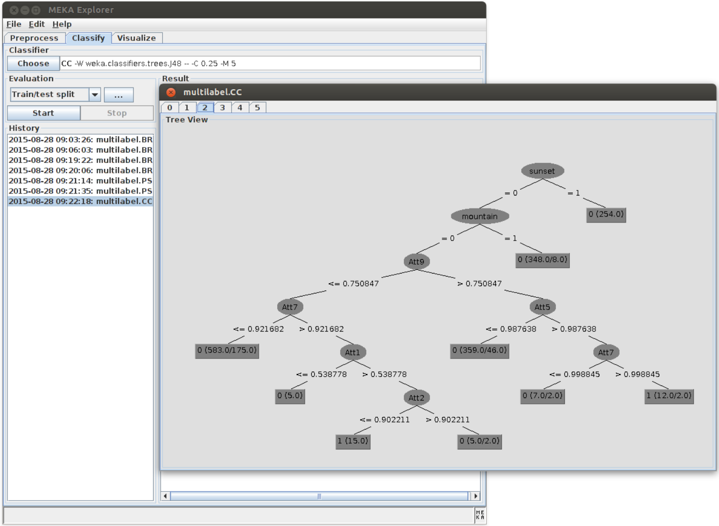
```
AUROC (macro averaged)          0.753
Curve Data
Macro Curve Data
Micro Curve Data
Label indices                   [     0
Accuracy (per label)            [ 0.792 0
Harmonic (per label)            [ 0.699 0
Precision (per label)           [ 0.664 0
Recall (per label)              [ 0.580 0
Empty labelvectors (predicted)  0.067
Label cardinality (predicted)   2.082
Levenshtein distance            0.251
Label cardinality (difference)  -0.219
avg. relevance (test set)       [ 0.291 0
avg. relevance (predicted)      [ 0.254 0
avg. relevance (difference)     [ 0.037 0
Results sampled over time


== Additional Measurements


Test time                       0.104
Build time                      0.028
Total time                      0.132
```

MEKA GUI



multilabel.incremental.BRUpdateable
X: False Positive Rate (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance
Reset
Clear
Open
Save
Jitter
Plot
1
0.5
0
0
0.5
1
X
Y
Class colour
0
0.5
1

MEKA GUI



CC -W weka.classifiers.trees.J48 -- -C 0.25 -M 5

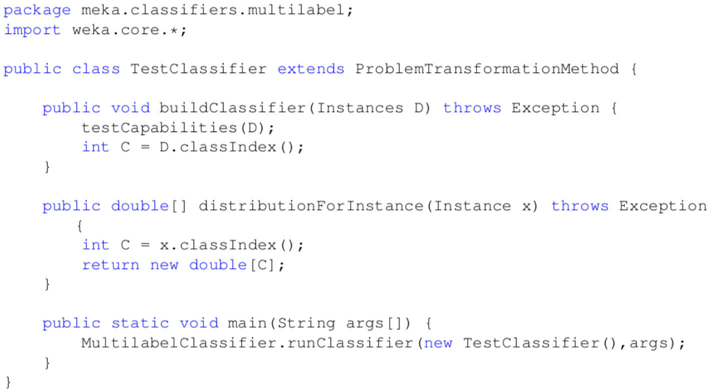
Train/test split

2015-08-28 09:03:26: multilabel.BR
2015-08-28 09:06:03: multilabel.BR
2015-08-28 09:19:22: multilabel.BR
2015-08-28 09:20:06: multilabel.BR
2015-08-28 09:21:14: multilabel.PS
2015-08-28 09:21:35: multilabel.PS
2015-08-28 09:22:18: multilabel.CC

MEKA CLI

```
java meka.classifiers.multilabel.meta.BaggingML -I 50 -P 100 \
  -t data/Enron.arff  \
  -output-debug-info -W meka.classifiers.multilabel.CC -- \
    -W weka.classifiers.functions.SMO
```

MEKA Classifiers



```
package meka.classifiers.multilabel;
import weka.core.*;

public class TestClassifier extends ProblemTransformationMethod {

    public void buildClassifier(Instances D) throws Exception {
        testCapabilities(D);
        int C = D.classIndex();
    }

    public double[] distributionForInstance(Instance x) throws Exception
       {
        int C = x.classIndex();
        return new double[C];
    }

    public static void main(String args[]) {
        MultilabelClassifier.runClassifier(new TestClassifier(),args);
    }
}
```

FAQ

Q#1: Can MEKA Handle Multi-target Classification?
(Multi-Label Multi-Class)
A: Yes – but note that your options for evaluation are more limited.

Q#2: Can MEKA Handle Multi-target Regression?
A: No – But this is a much more different problem; even if the 'conversion' is sometimes trivial, results are not as good (vs independent models).

Competition and Limitations

When *not* to use MEKA:

- You have 100,000s of labels and millions of training examples: Try distributed learning or deep learning instead
- You want to use `Python`: try Scikit-Learn, or Scikit Multi-Learn (contains wrappers to MEKA)
- Need to deal with data streams: MEKA has some support, but try a specialised framework like MOA or River (support for multi-label and multi-target)

Thank you!

MEKA Resources:

- A copy of these slides: `www.lix.polytechnique.fr/~jread/talks/MEKA.pdf`
- MEKA website: `http://waikato.github.io/meka/`
- GitHub `https://github.com/Waikato/meka/` (you'll find there a tutorial, documentation, etc.)

MEKA: A Multi-label Extension to WEKA

Jesse Read

ÉCOLE POLYTECHNIQUE
IP PARIS

Te Ipu o te Mahara
Artificial Intelligence Institute
THE UNIVERSITY OF WAIKATO

2021 International WEKA User Conference
November 26, 2021