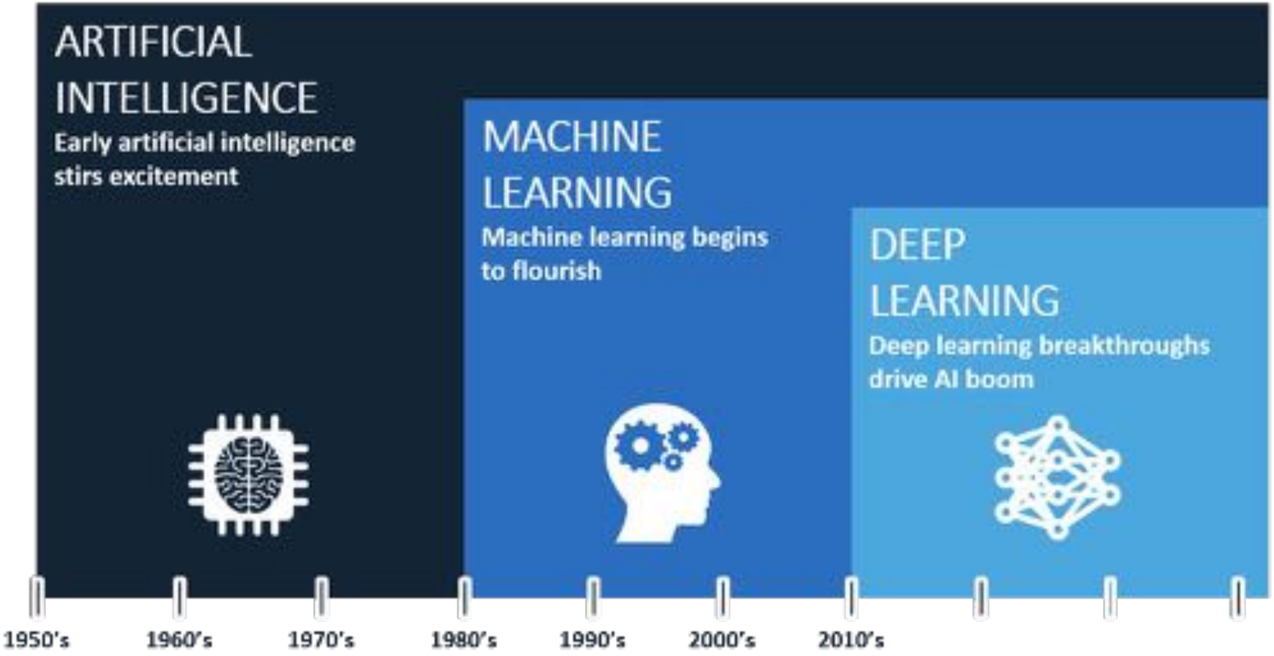
HyperLabelMe:
A web-platform for benchmarking remote sensing image classifiers

Jordi Muñoz-Marí, Luis Gómez-Chova, Ana B. Ruescas
Gonzalo Mateo-García,, Emma Izquierdo-Verdiguier
Manuel Campos Taberner, **Gustau Camps-Valls**

Image Processing Lab (IPL)
Universitat de València

Artificial intelligence, deep and machine learning

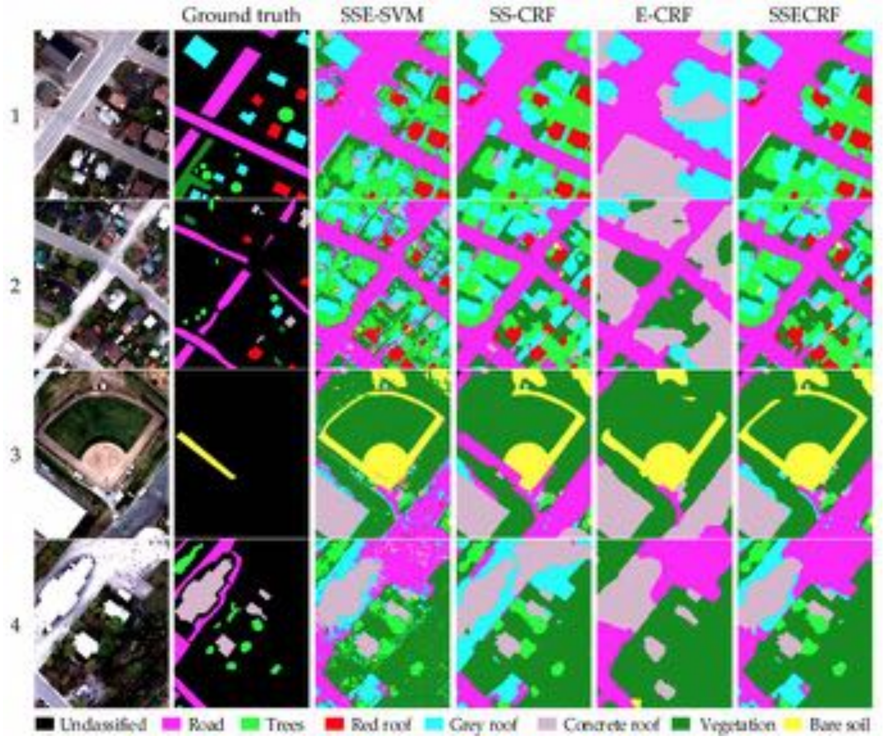

ARTIFICIAL INTELLIGENCE
Early artificial intelligence stirs excitement
MACHINE LEARNING
Machine learning begins to flourish
DEEP LEARNING
Deep learning breakthroughs drive AI boom
1950's
1960's
1970's
1980's
1990's
2000's
2010's

AI: Intelligence demonstrated by machines rather than humans or animals.

ML: Giving computers the skills to learn without explicit programming

DL: Is an ML subset, examining algorithms that learn and improve on their own.

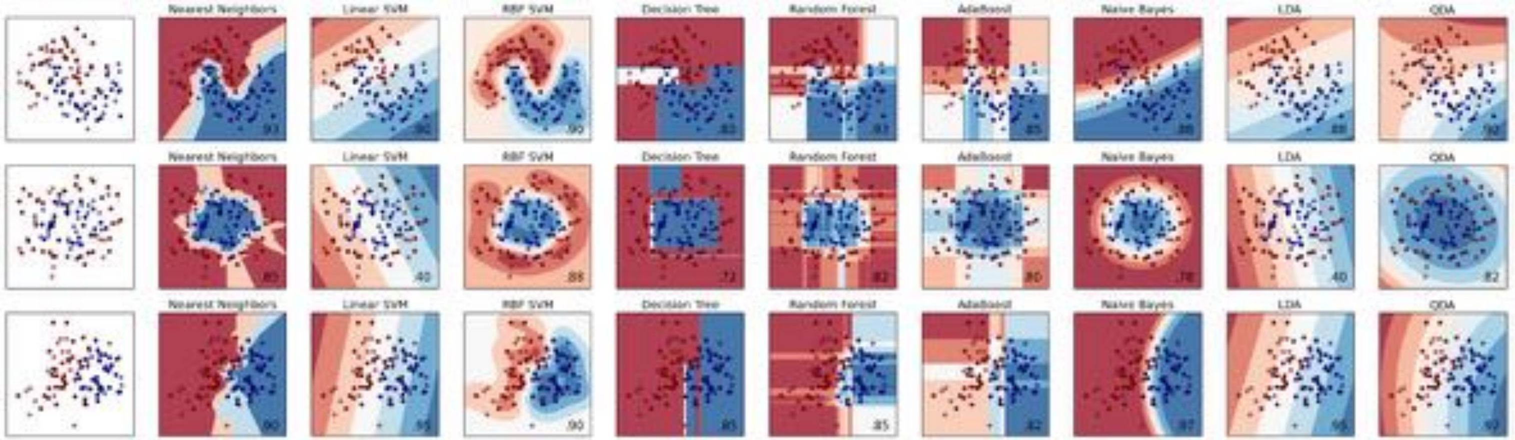
Remote sensing image classification



- **Goal:** "generate spatially explicit maps from remote sensing observations"
- **Machine learning**
  - Automates the process
  - Fast
  - Accurate
- **Many different classifiers**
  - SVM, RF, Boosting, Neural nets

Machine learning classifiers

- Computational cost?
- Accuracy?
- Easiness?
- Robustness to dimensionality?

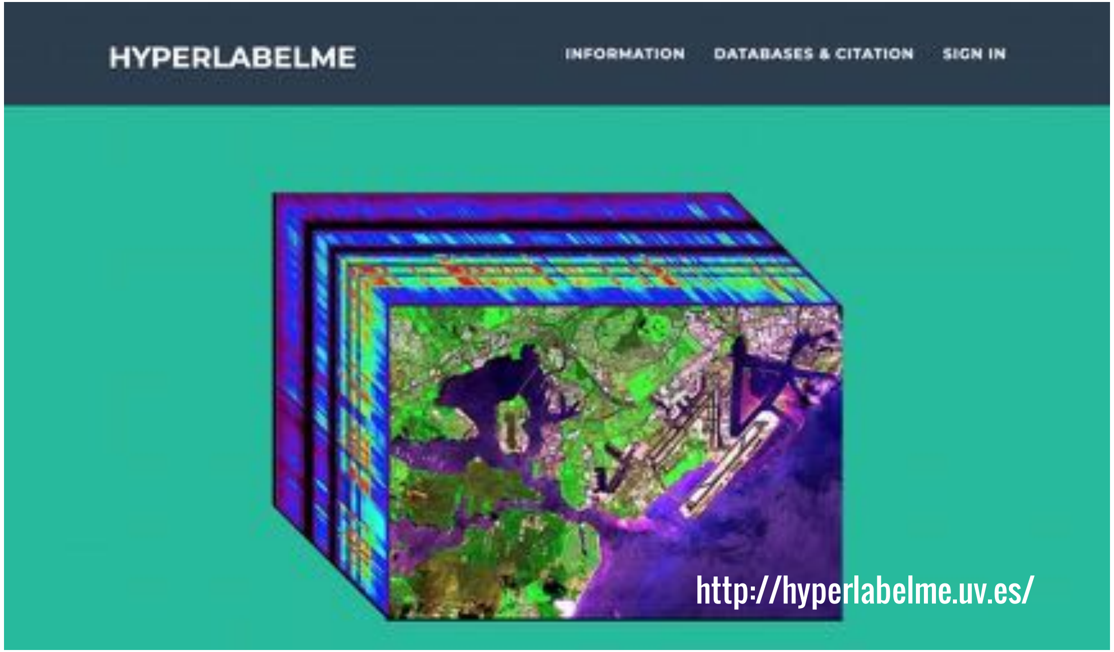


Hyperlabelme

http://hyperlabelme.uv.es/

Hyperlabelme:

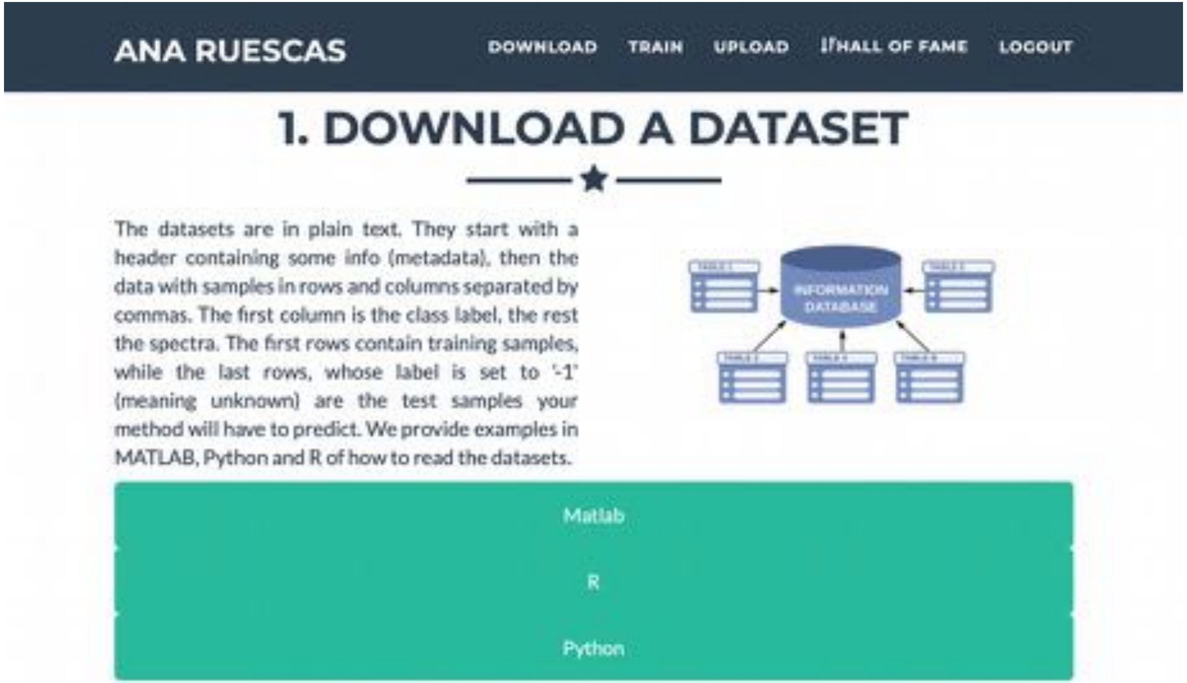
- A platform to evaluate machine learning classifiers
- Harmonized large set of labeled multispectral and hyperspectral images
  - Number of classes
  - Dimensionality
  - Noise sources and levels
- Scalable and modular



http://hyperlabelme.uv.es/

Download image datasets

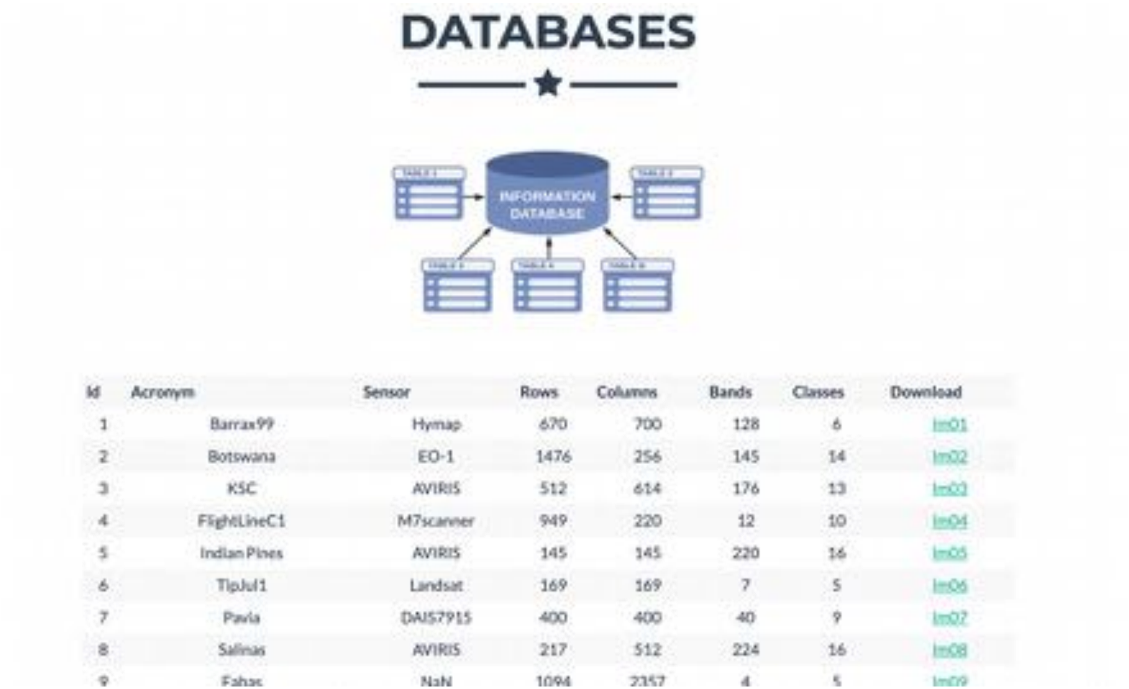
- HyperLabelMe runs a FAIR-use data policy
  - Data freely available to the public and scientific community
  - Download spectra and corresponding labels
  - Run your algorithms offline
  - Upload results
  - Get your scores



1. DOWNLOAD A DATASET

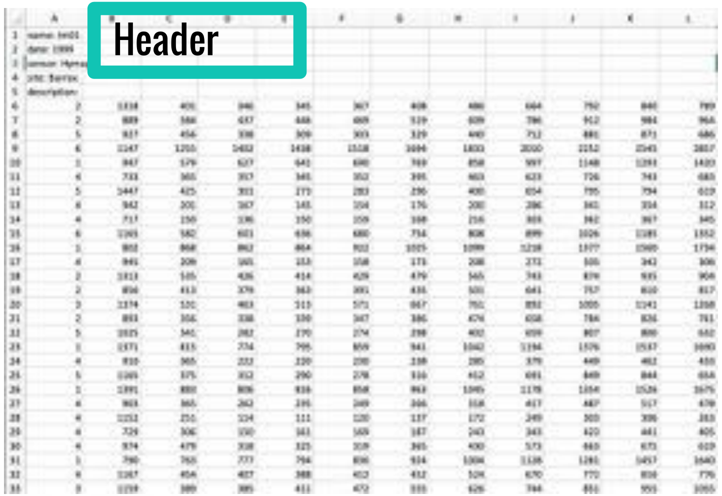
The datasets are in plain text. They start with a header containing some info (metadata), then the data with samples in rows and columns separated by commas. The first column is the class label, the rest the spectra. The first rows contain training samples, while the last rows, whose label is set to '-1' (meaning unknown) are the test samples your method will have to predict. We provide examples in MATLAB, Python and R of how to read the datasets.

Datasets are in plain text



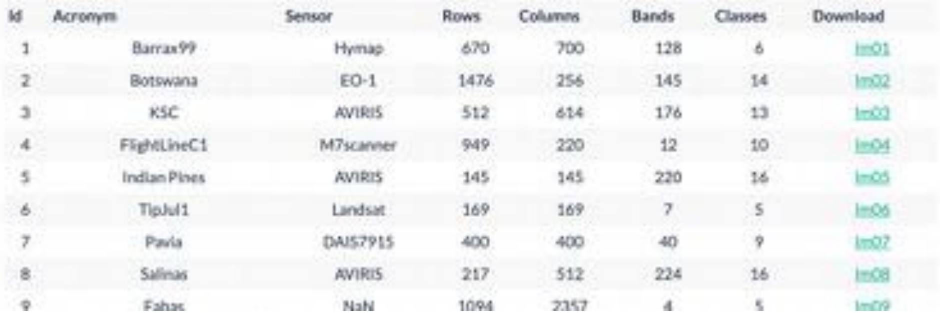
Id	Acronym	Sensor	Rows	Columns	Bands	Classes	Download
1	Barrax99	Hymap	670	700	128	6	Im01
2	Botswana	EO-1	1476	256	145	14	Im02
3	KSC	AVIRIS	512	614	176	13	Im03
4	FlightLineC1	M7scanner	949	220	12	10	Im04
5	Indian Pines	AVIRIS	145	145	220	16	Im05
6	TipJul1	Landsat	169	169	7	5	Im06
7	Pavia	DAIS7915	400	400	40	9	Im07
8	Salinas	AVIRIS	217	512	224	16	Im08
9	Fabas	NaN	1094	2357	4	5	Im09

Datasets are in plain text



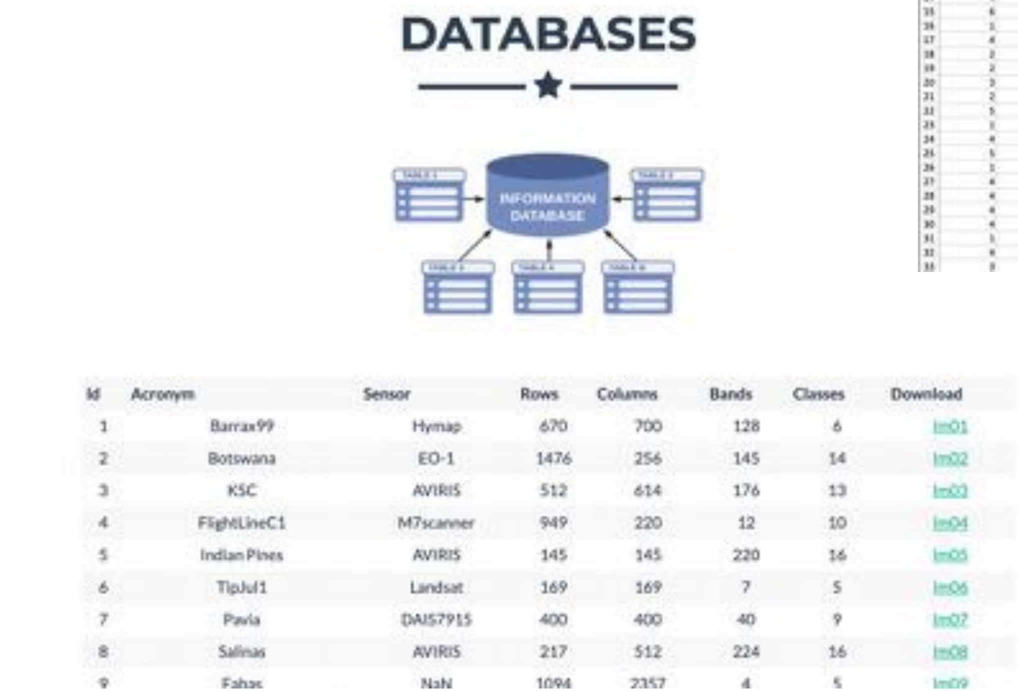
Header

name: Im01
date: 1999
sensor: Hymap
site: Barrax
description:



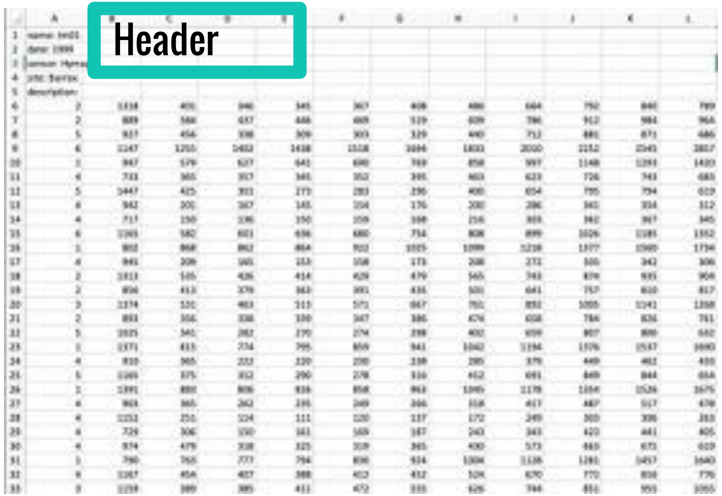
Id	Acronym	Sensor	Rows	Columns	Bands	Classes	Download
1	Barrax99	Hymap	670	700	128	6	Im01
2	Botswana	EO-1	1476	256	145	14	Im02
3	KSC	AVIRIS	512	614	176	13	Im03
4	FlightLineC1	M7scanner	949	220	12	10	Im04
5	Indian Pines	AVIRIS	145	145	220	16	Im05
6	TipJul1	Landsat	169	169	7	5	Im06
7	Pavia	DAIS7915	400	400	40	9	Im07
8	Salinas	AVIRIS	217	512	224	16	Im08
9	Fabas	NaN	1094	2357	4	5	Im09

Datasets are in plain text



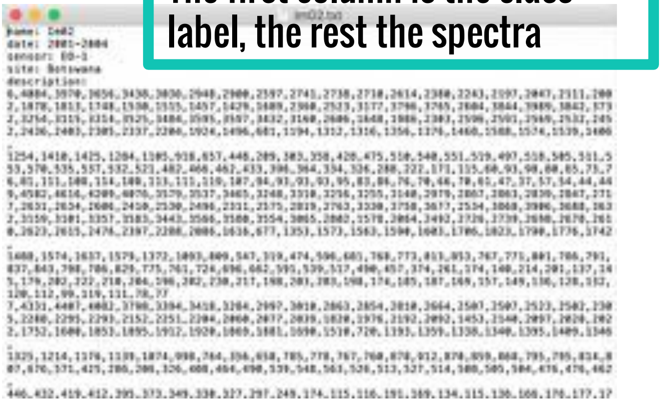
DATABASES

Id	Acronym	Sensor	Rows	Columns	Bands	Classes	Download
1	Barrax99	Hymap	670	700	128	6	Im01
2	Botswana	EO-1	1476	256	145	14	Im02
3	KSC	AVIRIS	512	614	176	13	Im03
4	FlightLineC1	M7scanner	949	220	12	10	Im04
5	Indian Pines	AVIRIS	145	145	220	16	Im05
6	TipJul1	Landsat	169	169	7	5	Im06
7	Pavia	DAIS7915	400	400	40	9	Im07
8	Salinas	AVIRIS	217	512	224	16	Im08
9	Fabas	NaN	1094	2357	4	5	Im09



Header

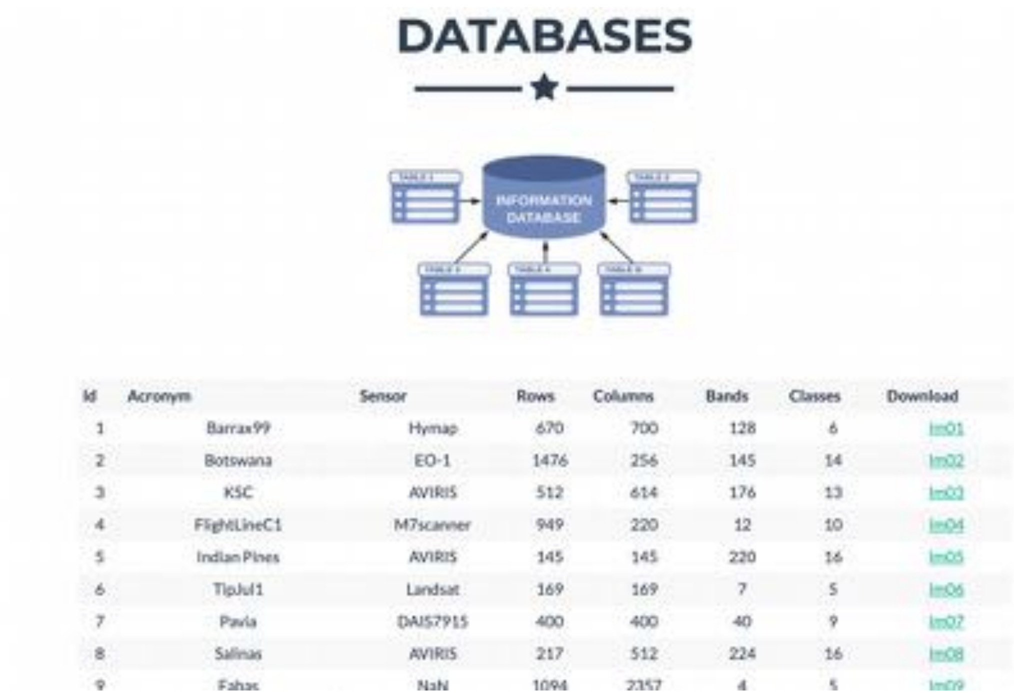
```
name: Im01
date: 1999
sensor: Hymap
site: Barrax
description:
[illegible]
```



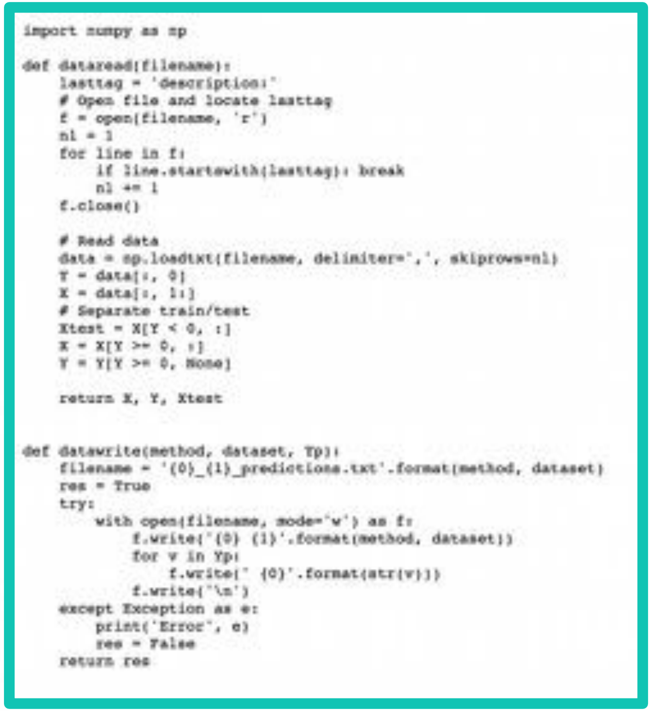
The first column is the class label, the rest the spectra

```
name: Im02
date: 2001-2004
sensor: EO-1
site: Botswana
description:
[illegible]
```

Easy to read



Id	Acronym	Sensor	Rows	Columns	Bands	Classes	Download
1	Barrax99	Hymap	670	700	128	6	Im01
2	Botswana	EO-1	1476	256	145	14	Im02
3	KSC	AVIRIS	512	614	176	13	Im03
4	FlightLineC1	M7scanner	949	220	12	10	Im04
5	Indian Pines	AVIRIS	145	145	220	16	Im05
6	TipJul1	Landsat	169	169	7	5	Im06
7	Pavia	DAIS7915	400	400	40	9	Im07
8	Salinas	AVIRIS	217	512	224	16	Im08
9	Fabas	NaN	1094	2357	4	5	Im09

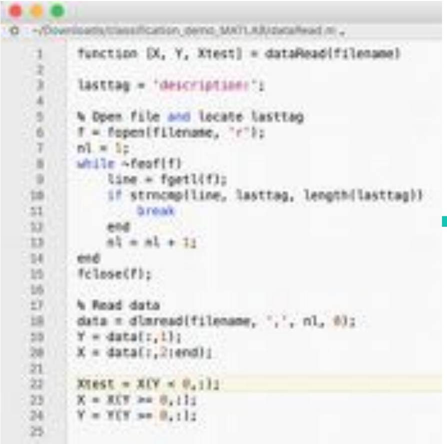


```
import numpy as np

def dataread(filename):
    lasttag = 'description:'
    # Open file and locate lasttag
    f = open(filename, 'r')
    nl = 1
    for line in f:
        if line.startswith(lasttag): break
        nl += 1
    f.close()

    # Read data
    data = np.loadtxt(filename, delimiter=',', skiprows=nl)
    Y = data[:, 0]
    X = data[:, 1:]
    # Separate train/test
    Xtest = X[Y < 0, :]
    X = X[Y >= 0, :]
    Y = Y[Y >= 0, None]

    return X, Y, Xtest


def datawrite(method, dataset, Yp):
    filename = '{0}_{1}_predictions.txt'.format(method, dataset)
    res = True
    try:
        with open(filename, mode='w') as f:
            f.write('{0} {1}'.format(method, dataset))
            for v in Yp:
                f.write(' {0}'.format(str(v)))
            f.write('\n')
    except Exception as e:
        print('Error', e)
        res = False
    return res
```

We provide examples in MATLAB, Python and R of how to read the datasets.

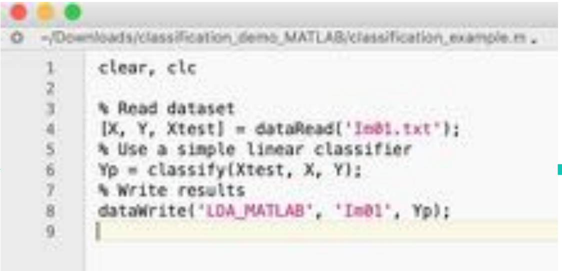
10
02
12
105
34
32
03
21
17
38

Train your models offline and upload your predictions



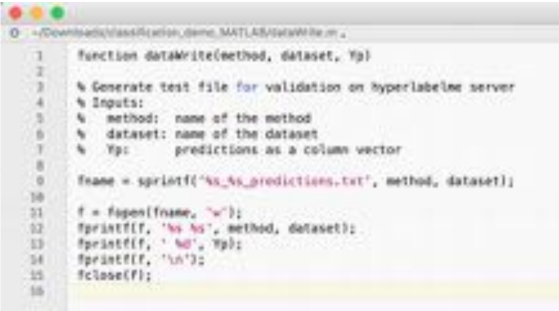
```
function [X, Y, Xtest] = dataRead(filename)

lasttag = 'description:';

% Open file and locate lasttag
f = fopen(filename, 'r');
nl = 1;
while ~feof(f)
    line = fgetl(f);
    if strncmp(line, lasttag, length(lasttag))
        break
    end
    nl = nl + 1;
end
fclose(f);

% Read data
data = dlmread(filename, ',', nl, 0);
Y = data(:,1);
X = data(:,2:end);

Xtest = X(Y < 0,:);
X = X(Y >= 0,:);
Y = Y(Y >= 0,:);
```

Read


~/Downloads/classification_demo_MATLAB/classification_example.m

```
clear, clc

% Read dataset
[X, Y, Xtest] = dataRead('Im01.txt');
% Use a simple linear classifier
Yp = classify(Xtest, X, Y);
% Write results
dataWrite('LDA_MATLAB', 'Im01', Yp);

```

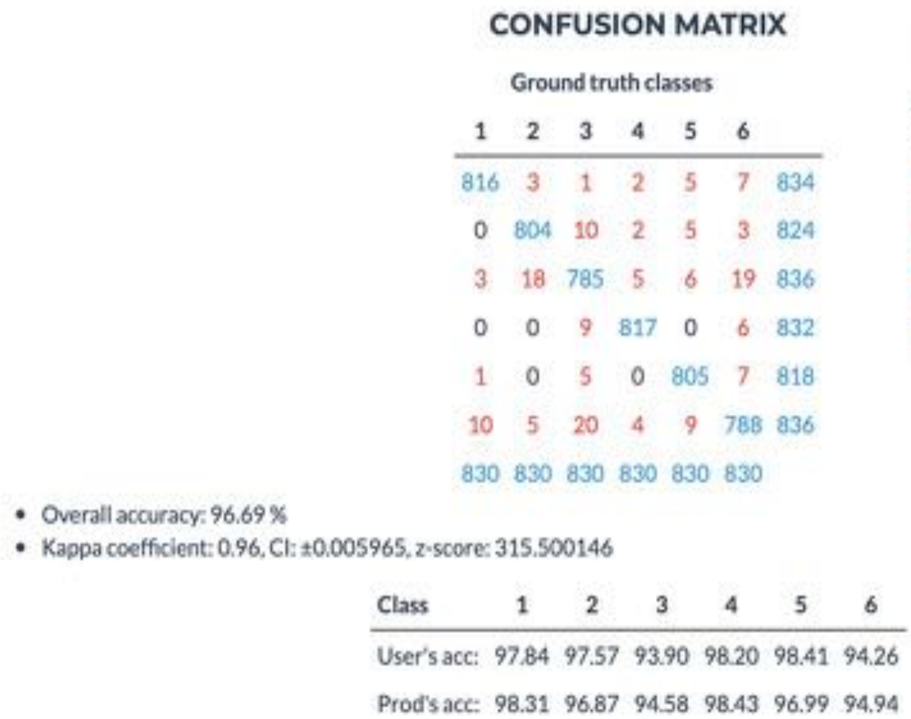
Train



```
function dataWrite(method, dataset, Yp)

% Generate test file for validation on hyperlabelme server
% Inputs:
%   method:  name of the method
%   dataset: name of the dataset
%   Yp:      predictions as a column vector

fname = sprintf('%s_%s_predictions.txt', method, dataset);

f = fopen(fname, 'w');
fprintf(f, '%s %s', method, dataset);
fprintf(f, ' %d', Yp);
fprintf(f, '\n');
fclose(f);
```

Save

Train your models offline and upload your predictions



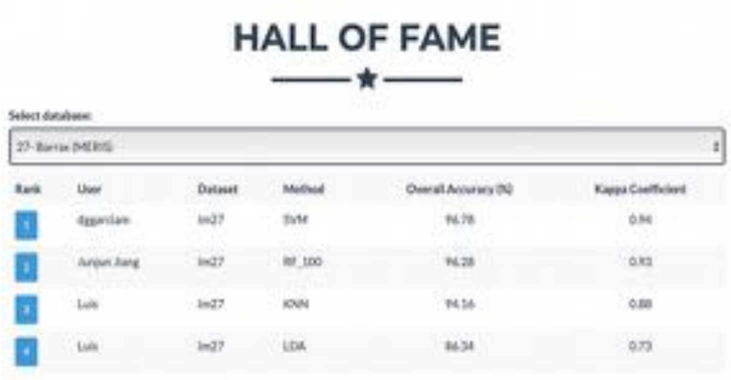
CONFUSION MATRIX

Ground truth classes

1	2	3	4	5	6	
816	3	1	2	5	7	834
0	804	10	2	5	3	824
3	18	785	5	6	19	836
0	0	9	817	0	6	832
1	0	5	0	805	7	818
10	5	20	4	9	788	836
830	830	830	830	830	830	

- Overall accuracy: 96.69 %
- Kappa coefficient: 0.96, CI: ±0.005965, z-score: 315.500146

Class	1	2	3	4	5	6
User's acc:	97.84	97.57	93.90	98.20	98.41	94.26
Prod's acc:	98.31	96.87	94.58	98.43	96.99	94.94

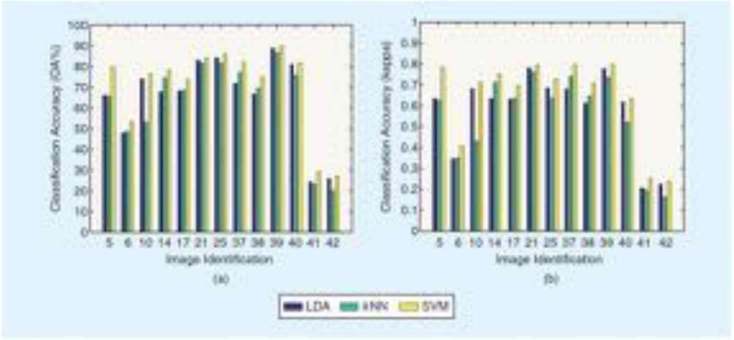


HALL OF FAME

Select database:

27- Barrax (MERIS)

Rank	User	Dataset	Method	Overall Accuracy (%)	Kappa Coefficient
1	dgarciam	Im27	SVM	96.78	0.96
2	Junjun Jiang	Im27	RF_100	96.28	0.93
3	Luis	Im27	KNN	94.16	0.88
4	Luis	Im27	LDA	86.34	0.73

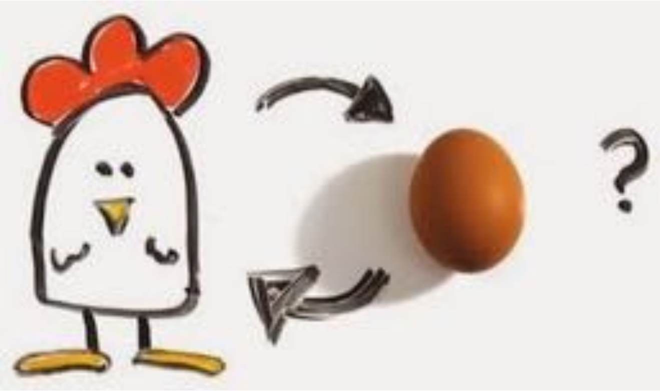


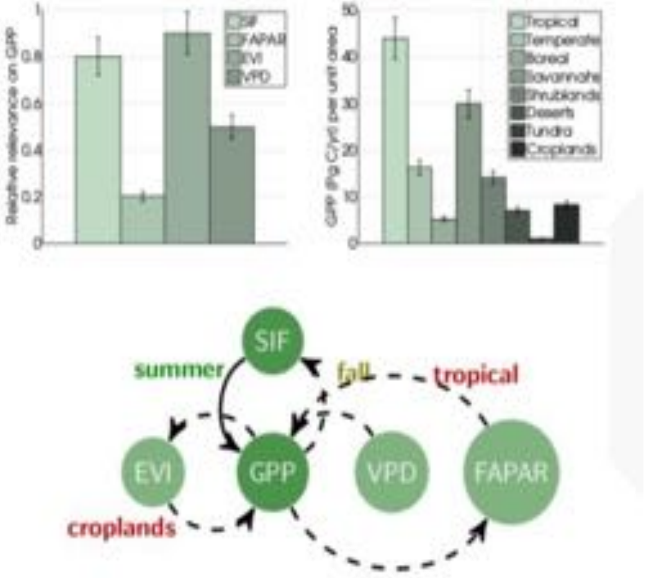
Causeme: understanding is harder than predicting!

http://causeme.uv.es/

Extension to causal discovery

- Learn what’s the cause and the effect from time series of variables
- Important implications in Earth science and climate





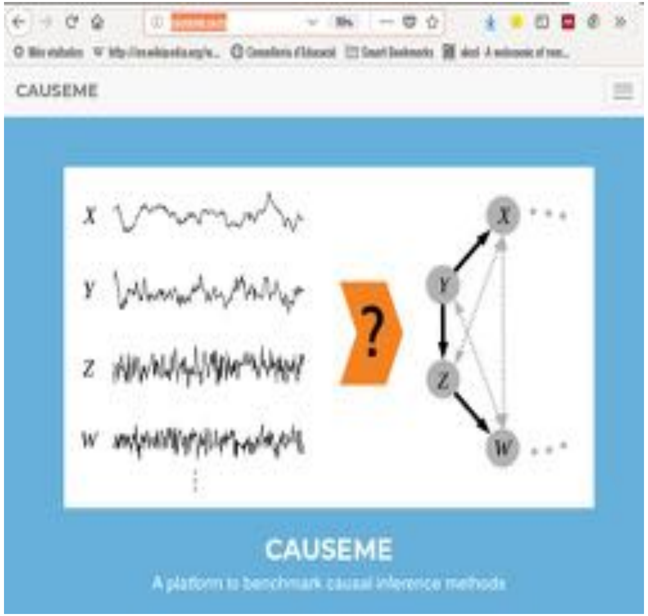
“Inferring causation from time series with perspectives in Earth system sciences”
Runge, Bathiany, Bollt, Camps-Valls, et al. Nat Comm (submitted), 2018.
“Causal Inference in Geoscience and Remote Sensing from Observational Data,”
Pérez-Suay and Camps-Valls, IEEE Trans. Geosc. Rem. Sens, 2018

Extension to causal discovery

- **CauseMe: http://causeme.uv.es**
  - Download time series with ground truth
  - Run your causal discovery algorithm offline
  - Upload your causal graph
  - Get your results!

“Inferring causation from time series with perspectives in Earth system sciences”
Runge, Bathiany, Bollt, Camps-Valls, et al. Nat Comm (submitted), 2018.
“Causal Inference in Geoscience and Remote Sensing from Observational Data,”
Pérez-Suay and Camps-Valls, IEEE Trans. Geosc. Rem. Sens, 2018

CauseMe: http://causeme.uv.es



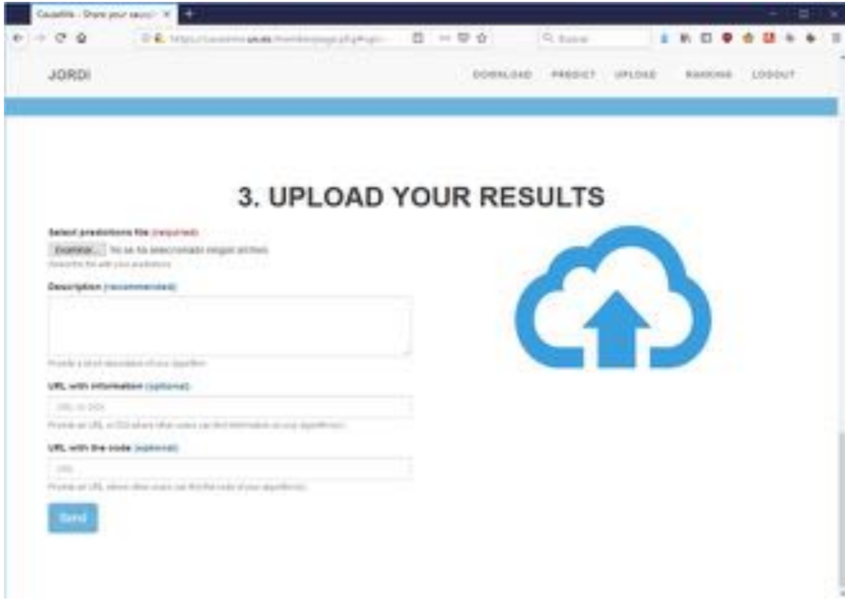
CAUSEME

CAUSEME

A platform to benchmark causal inference methods

Extension to causal discovery

- **CauseMe: http://causeme.uv.es**
  - Download time series with ground truth
  - Run your causal discovery algorithm offline
  - Upload your causal graph
  - Get your results!



3. UPLOAD YOUR RESULTS

Select predictions file (required)

Description (recommended)

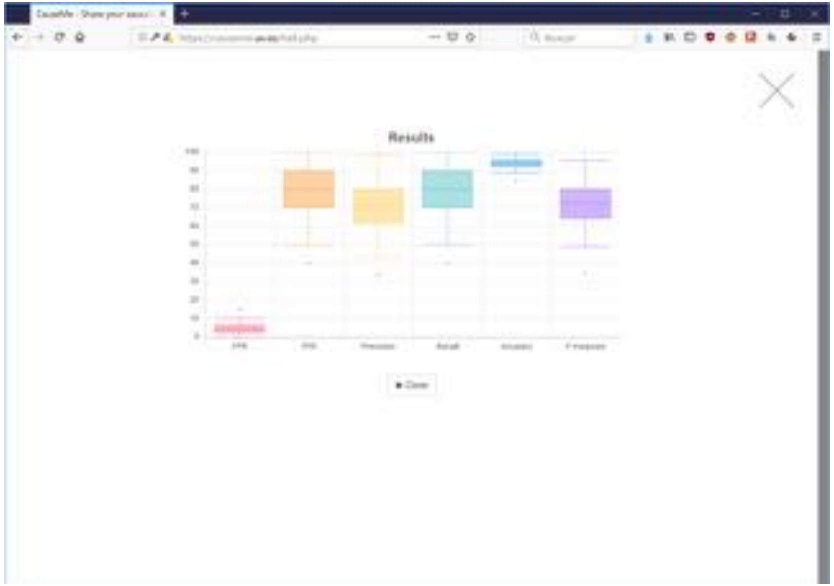
URL with information (optional)

URL with the code (optional)

“Inferring causation from time series with perspectives in Earth system sciences”
Runge, Bathiany, Bollt, Camps-Valls, et al. Nat Comm (submitted), 2018.
“Causal Inference in Geoscience and Remote Sensing from Observational Data,”
Pérez-Suay and Camps-Valls, IEEE Trans. Geosc. Rem. Sens, 2018

Extension to causal discovery

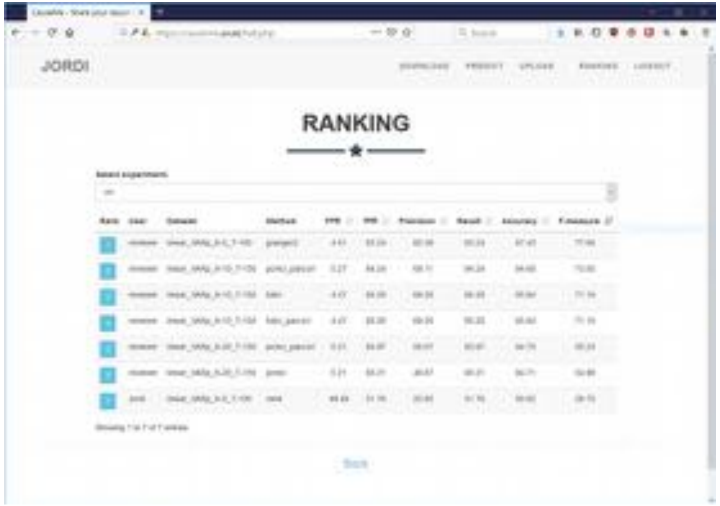
- **CauseMe: http://causeme.uv.es**
  - Download time series with ground truth
  - Run your causal discovery algorithm offline
  - Upload your causal graph
  - Get your results!



"Inferring causation from time series with perspectives in Earth system sciences"
Runge, Bathiany, Bollt, Camps-Valls, et al. Nat Comm (submitted), 2018.
"Causal Inference in Geoscience and Remote Sensing from Observational Data,"
Pérez-Suay and Camps-Valls, IEEE Trans. Geosc. Rem. Sens, 2018

Extension to causal discovery

- **CauseMe: http://causeme.uv.es**
  - Download time series with ground truth
  - Run your causal discovery algorithm offline
  - Upload your causal graph
  - Get your results!



"Inferring causation from time series with perspectives in Earth system sciences"
Runge, Bathiany, Bollt, Camps-Valls, et al. Nat Comm (submitted), 2018.
"Causal Inference in Geoscience and Remote Sensing from Observational Data,"
Pérez-Suay and Camps-Valls, IEEE Trans. Geosc. Rem. Sens, 2018

Conclusions

Conclusions

- **A fact:** Remote sensing needs machine learning
- **A question:** What's the best classifier for my specific task?
- **An observation:** Often the simpler is the better
- **Our aim:** Hyperlabelme is a simple web platform to benchmark classifiers
- Working towards extensions:
  - Spatial context
  - Spatio-temporal classifiers
- Some relatives:
  - Causal discovery in time series
  - Regression (parameter retrieval) algorithms
- **Please contribute!**

http://hyperlabelme.uv.es

http://causeme.uv.es

http://regressme.uv.es