Machine learning for scientific workflows
Managing the data science process

Balázs Kégl

Head of AI research

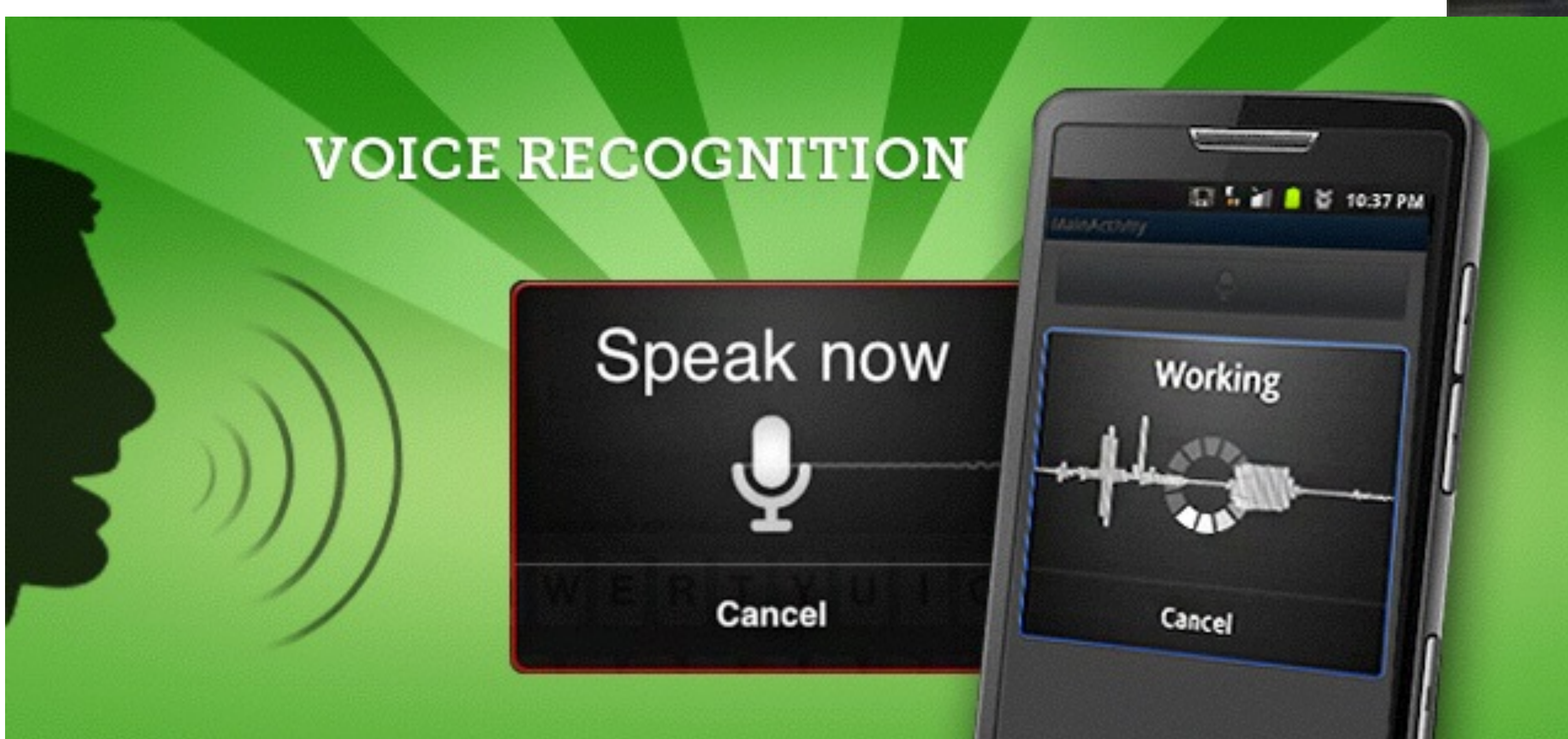
Huawei Paris

Who am I?

Balázs Kégl

- Senior researcher CNRS
  - machine learning (20+ years)
    interfacing with particle physics (10+ years)
- Head of the Paris-Saclay Center for Data Science
  - interfacing with biology, economy, climatology, chemistry, etc. (4 years)
  - industrial ML projects (4 years)
- Head of AI research, Huawei Paris
  - interfacing with telecom engineering applications (1 year)

AI: Higly visible breakthroughs



Why is the adoption of AI **so slow**?

THE HUMAN FACTOR

- Adopting AI will change the **way we work**
  - both the AI "consumer"
  - and the AI developer (engineers and data scientists)
- We have excellent tools to solve problems, but not very **little know-how to manage the process**
- Designing interfaces
  - formal **APIs**
  - **human/human** communication
  - **human/AI** communication
  - **AI/AI** communication

Outline

- Machine learning for reusable scientific workflows
  - use cases
  - examples
- Managing the data science process: the RAMP framework
  - roles and tasks
  - building the workflow: who does what
  - what is a predictive workflow, what are the parametrizable components
  - how to make data scientists efficient

Why is data science process management relevant in research?

- Typical **applied** research project
  - take an **existing domain-scientific or industrial problem** (e.g., galaxy deblending)
  - scan **literature**
  - install/develop **experimental environment**
  - **collect data** and **establish benchmark**
  - apply **existing ML solution**
  - optionally fine tune, explore a **small number** of alternatives
  - show that the **ML solution is better than the classical "manual" solution** on your **own benchmark**
  - publish

université PARIS-SACLAY
Paris-Saclay Center for Data Science

Takes **years**, typically

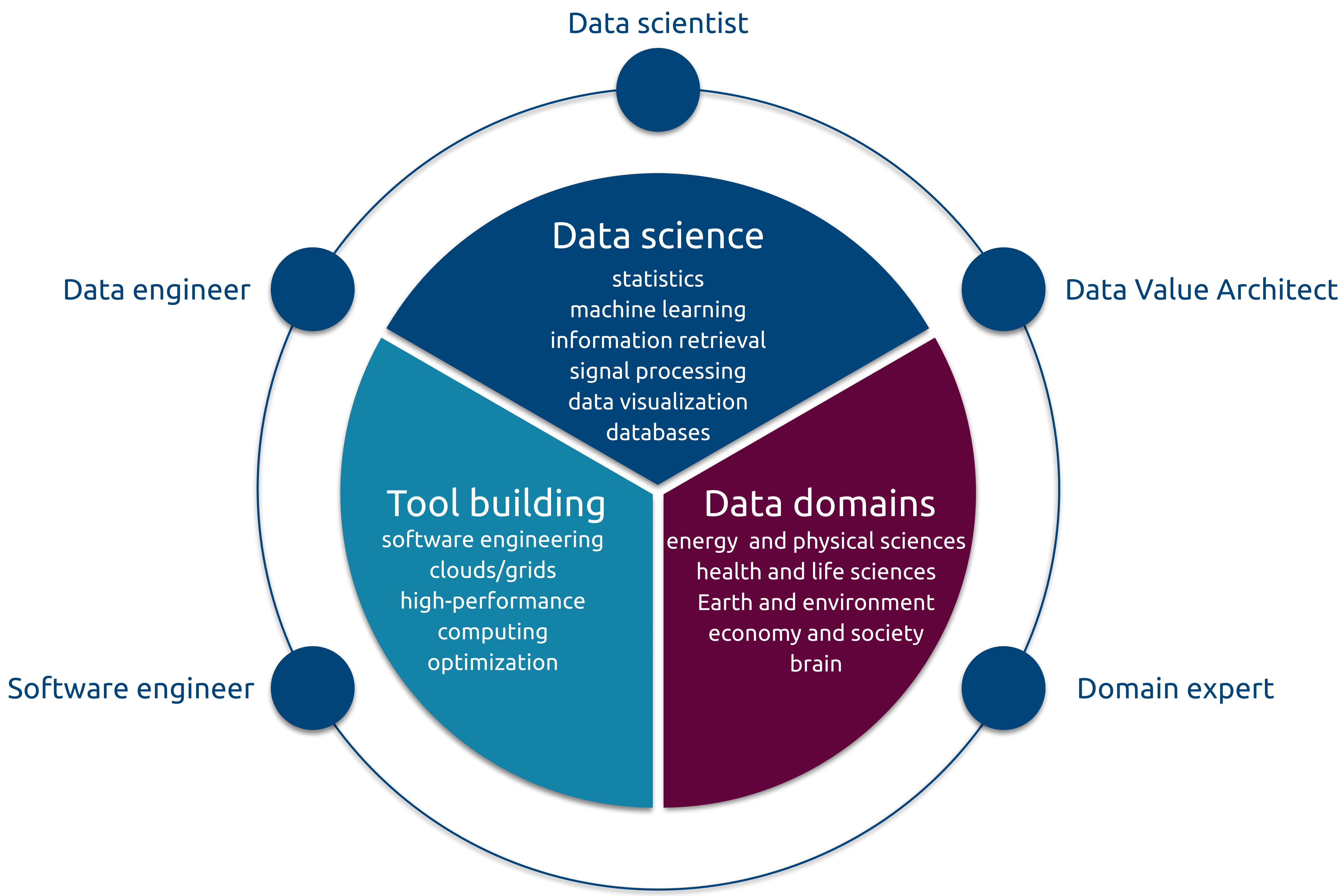
How to accelerate experimental projects?

How to explore a large number of ML solutions in a short time?

How to make solutions not only reproducible but also reusable?

THE DATA SCIENCE ECOSYSTEM

https://medium.com/@balazskegl



ML USE CASES IN SCIENCES

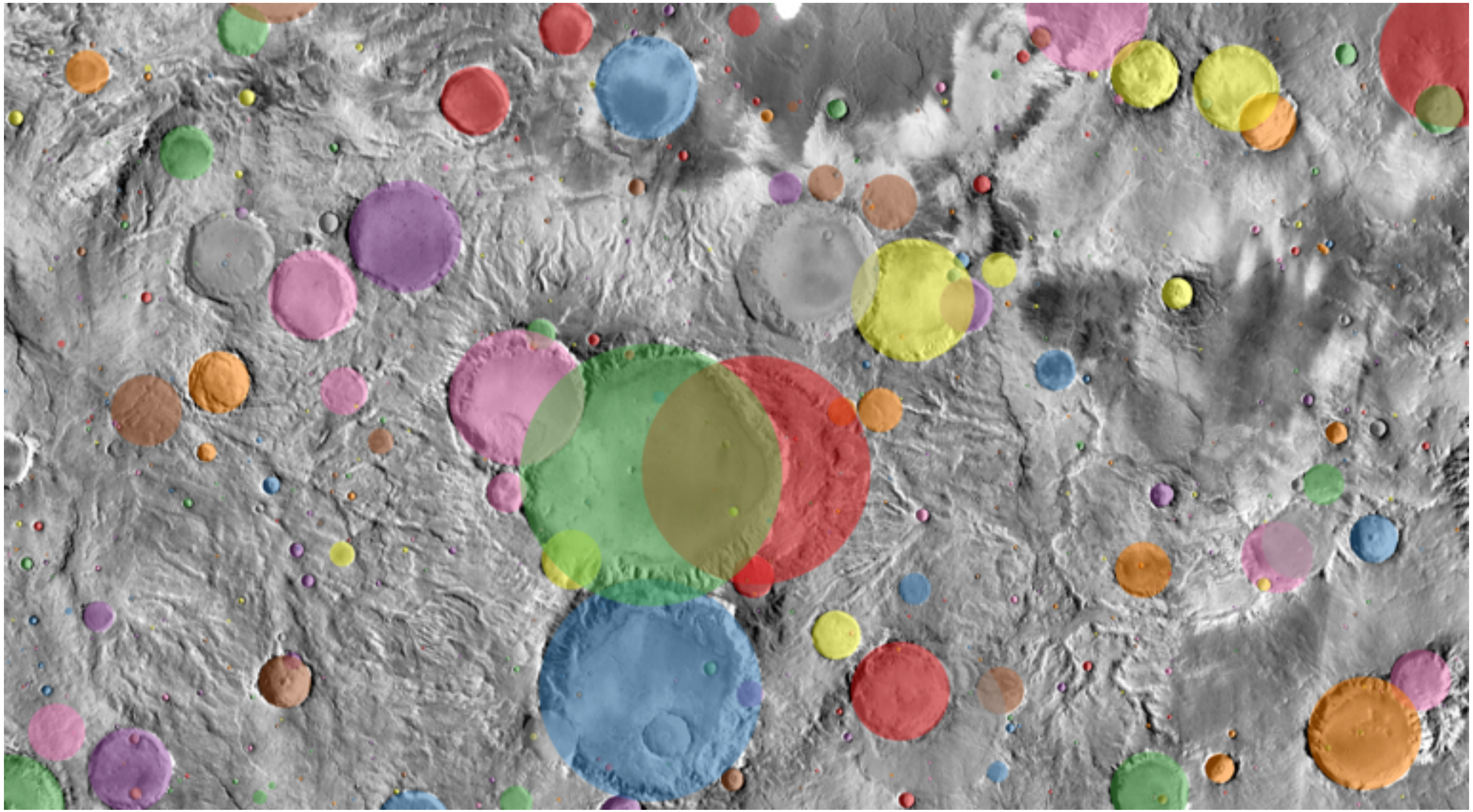
https://www.ramp.studio/problems

- Data collection: replace human or algorithmic collector or annotator
  - label insect photos, detect Mars craters, detect particle tracks
- Inference: to invert the generative model
  - "predict" a particle, detect an anomaly, infer a parameter y from observation x
- Generation, model reduction: to replace expensive simulations
  - "learn" a physics simulation or an agent based micro-economical model with a neural net
- Hypothesis generation: to "replace" theoreticians
  - learn, represent structural knowledge and generate novelty in model space, e.g., molecule generation in drug discovery

Data collection

Detecting Mars craters

- collaboration with planetary geologists at Paris-Saclay
- new metrics and workflow
- great benchmark for detection in satellite imagery



Detecting Mars craters

Saclay MSc students competing and learning from each other while benchmarking state of the art deep learning detection algorithms

rank	move	team	submission	ap	train time [s]	test time [s]
1	+1	felix.larrouy	m-rcnn3	0.584	5947	2009
2	-1	nicolas.toussaint	thanks_felix	0.577	6524	1964
3	-	haquang.le	yolo_v3	0.567	35400	1793
4	-	clement.hardy	mask2	0.565	12130	1512
5	+1	alann.cheral	hello_world_7	0.538	6039	2075
6	+4	guillaume.fradet	takeoff_COCO_augment	0.519	11824	2098
7	-2	sidali.hamideche	eighth	0.499	27407	285
8	-1	manon.cesaire	mrcnn_3	0.476	11734	1463
9	-	hao.liu	NASA-V	0.434	16140	200
10	-2	enzo.terreau	ssd_class_1	0.408	2551	125


Inference

Learning to discover: the Higgs boson machine learning challenge

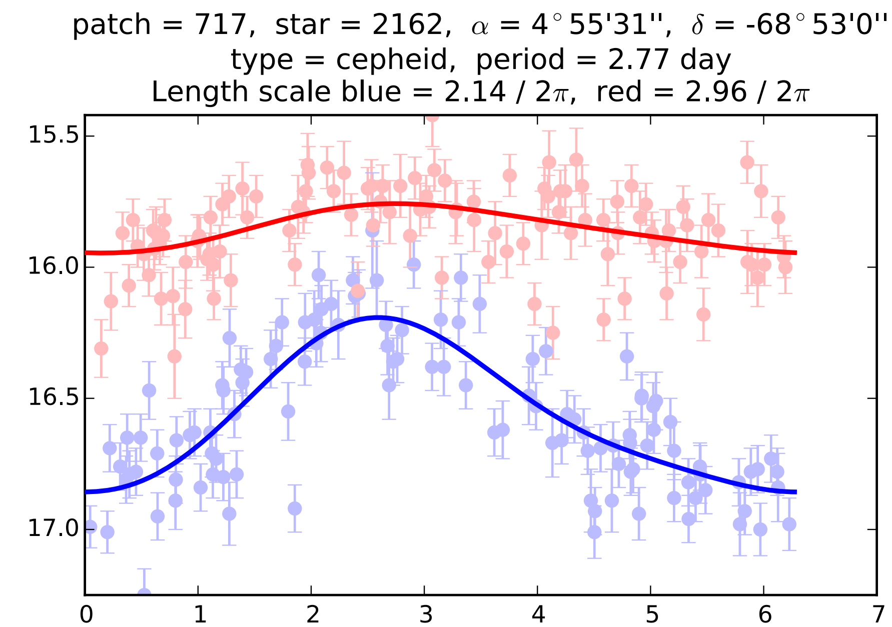
Claire Adam-Bourdarios[a], Glen Cowan[b], Cécile Germain[c], Isabelle Guyon[d], Balázs Kégl[a,c], David Rousseau[a]

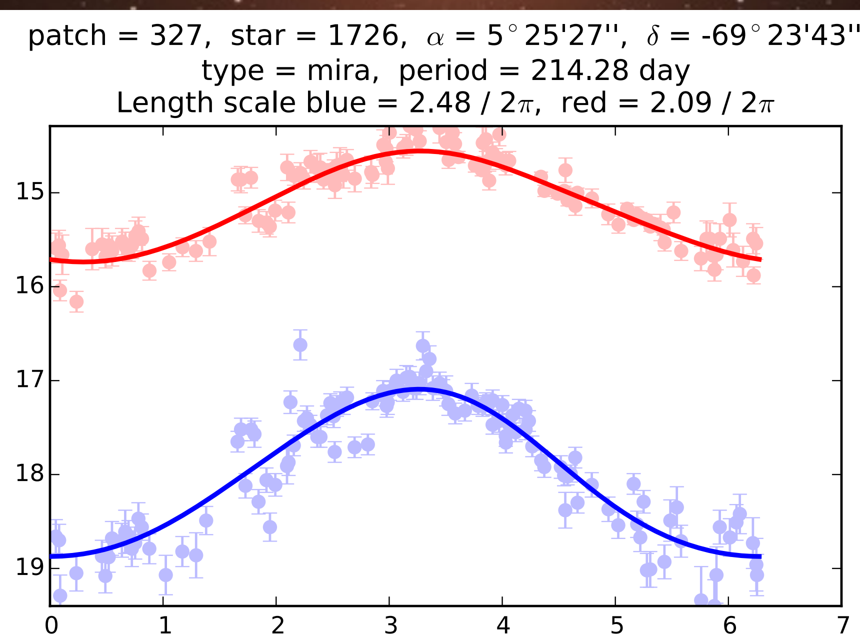
[a] LAL, IN2P3/CNRS & University Paris-Sud, France
[b] Physics Department, Royal Holloway, University of London, UK
[c] TAO team, INRIA & LRI, CNRS & University Paris-Sud, France
[d] ChaLearn

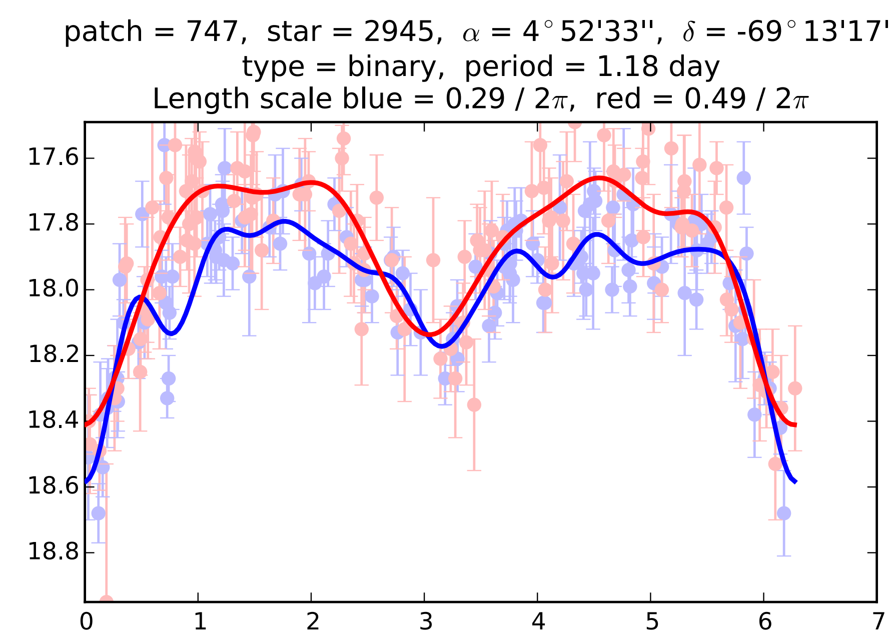
21 July 2014, version 1.8

Classifying variable stars

- collaboration with **astrophysicists at Paris-Saclay**
- variable-length **functional data**

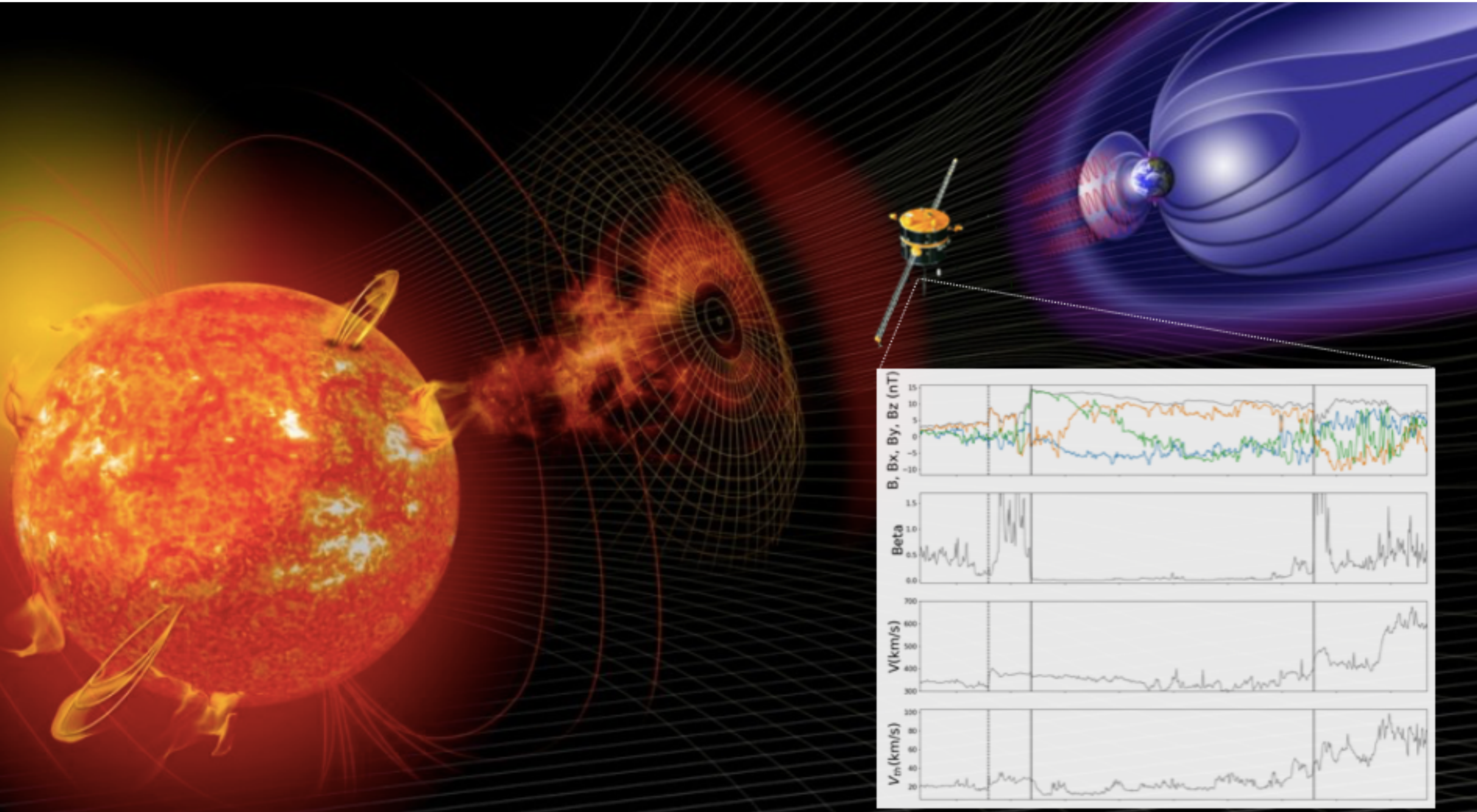




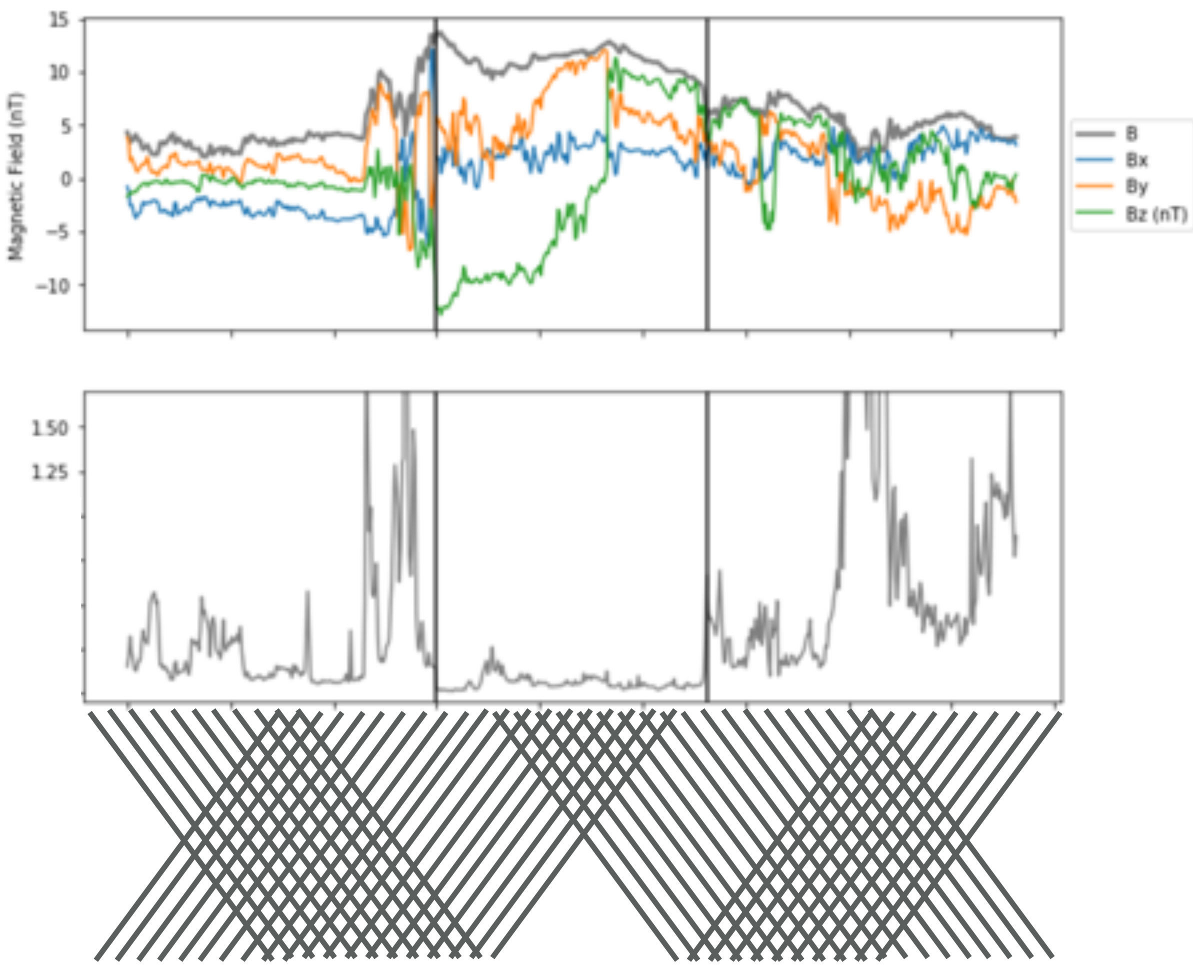


Detecting solar storms

- collaboration with **plasma physicists at Paris-Saclay**
- multi time series detection



DETECTING SOLAR STORMS



feature extractor

x

(a fixed length feature vector at each time step)

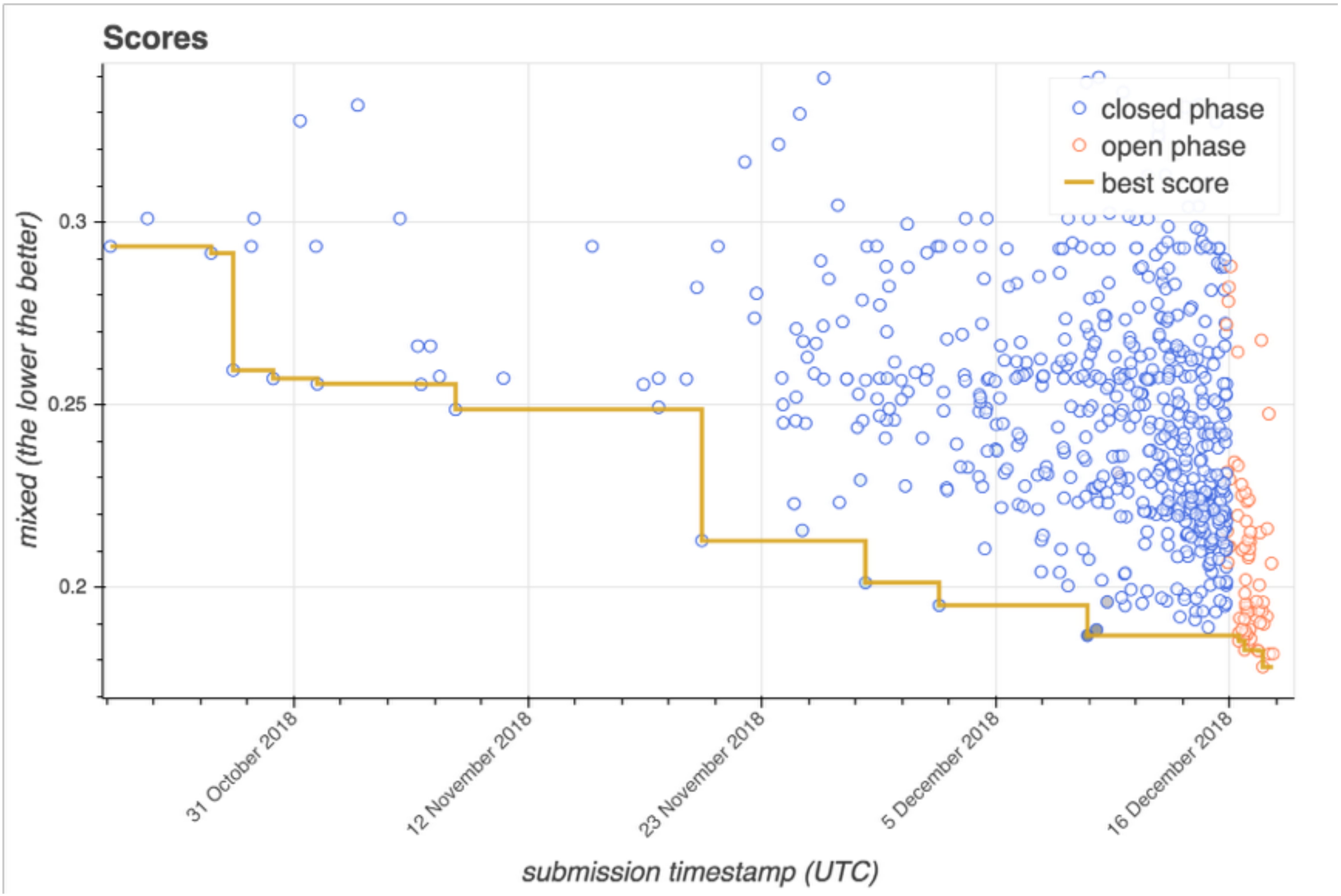
classifier

y_{pred}

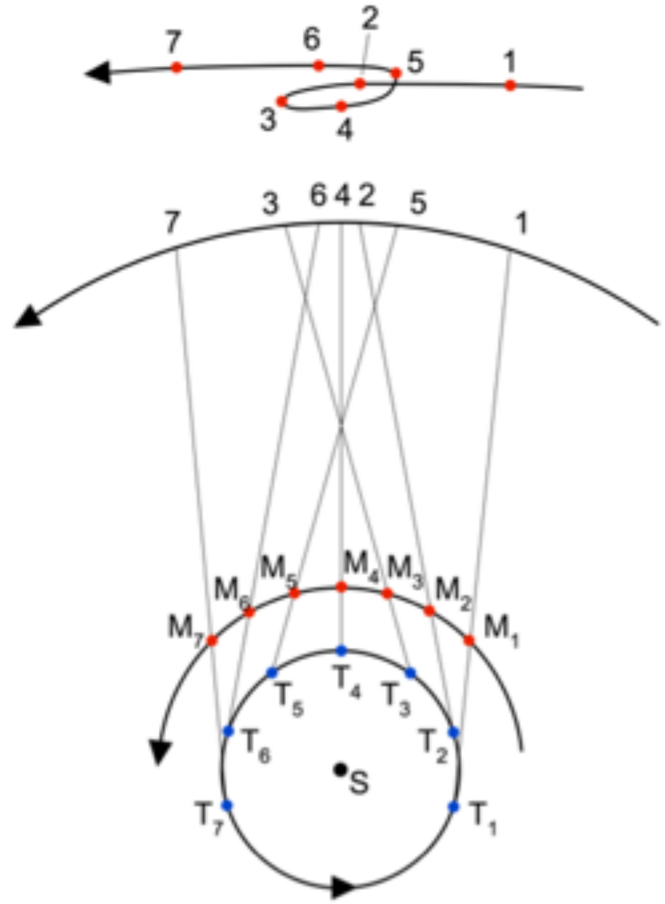
(a fixed length binary indicator storm/not storm at each time step)

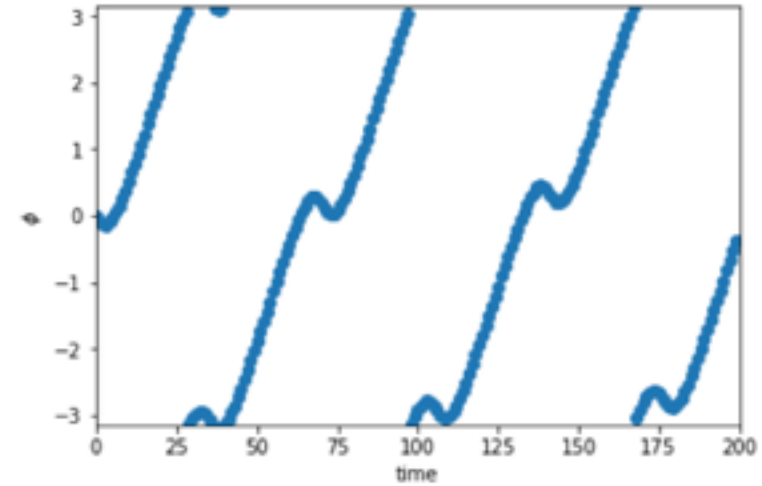
GRADUATE STUDENT DESCENT

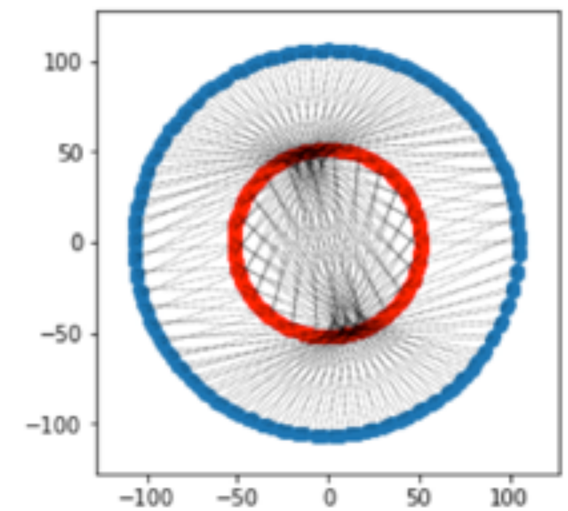
Saclay MSc students competing and learning from each other submitting 700 predictive models in three weeks

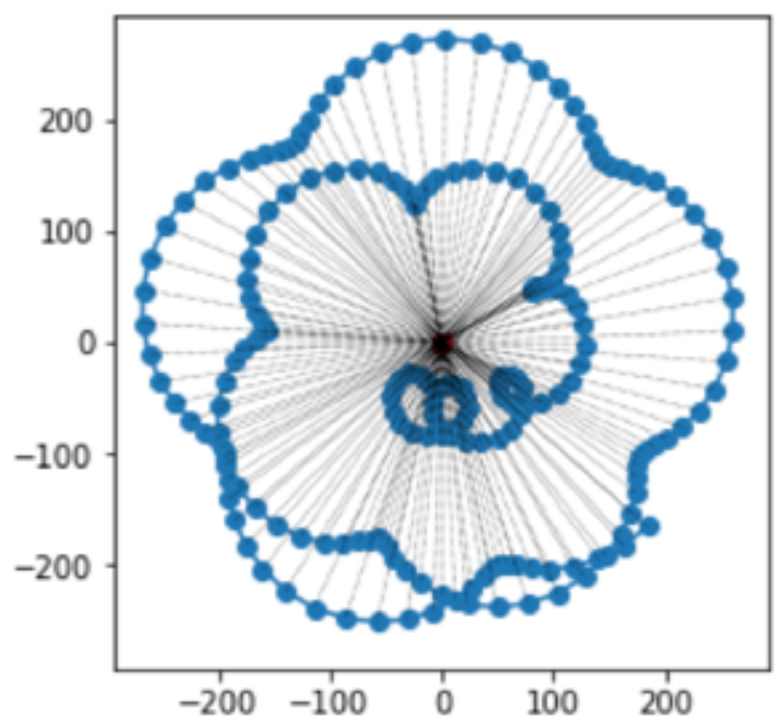


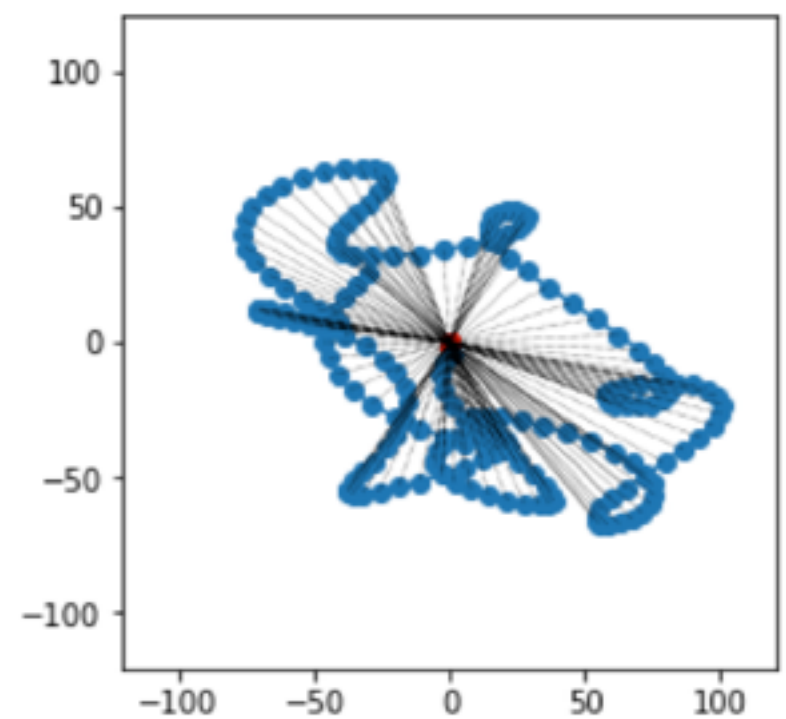
Discovering physics (mechanics)



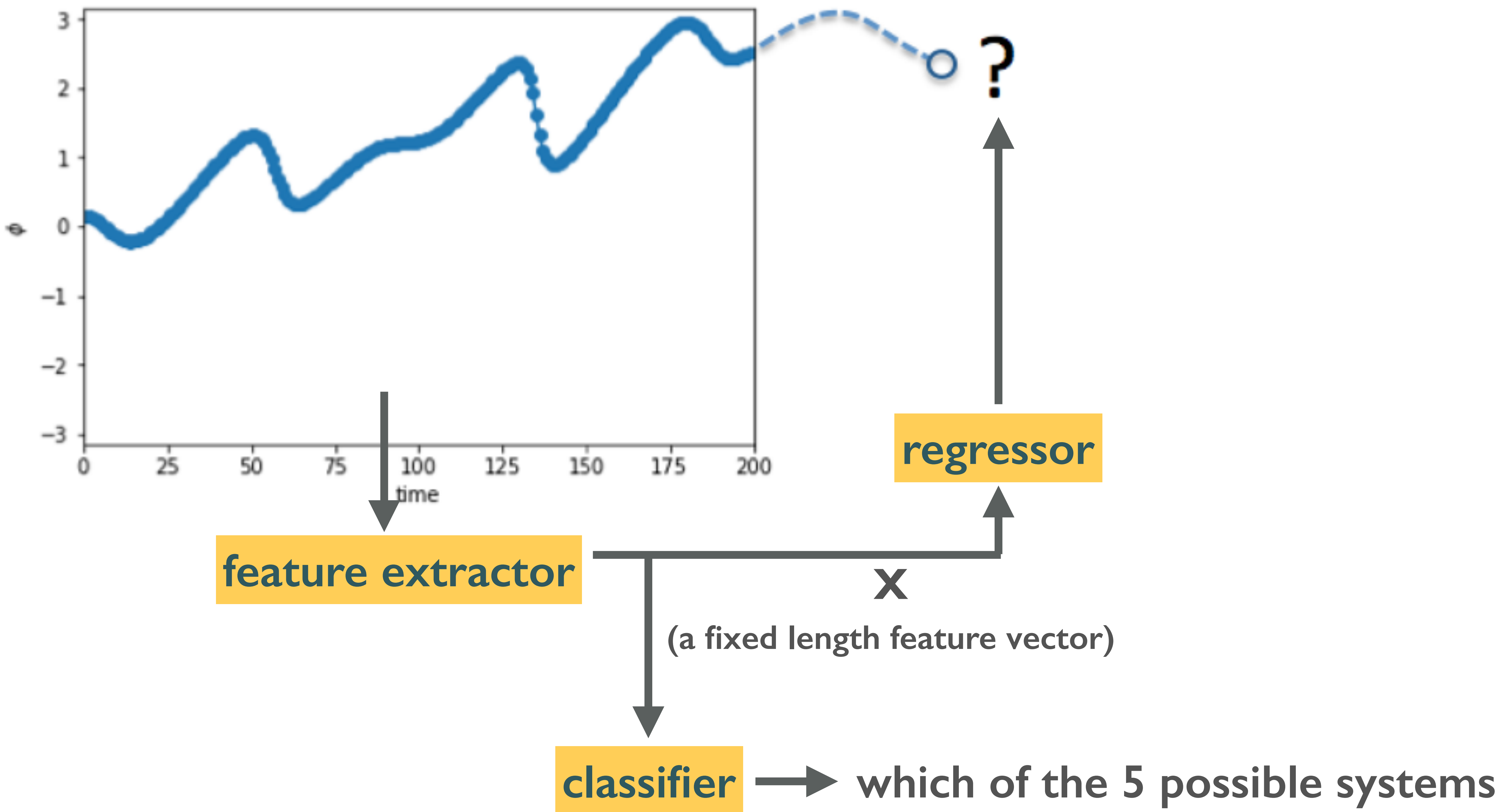








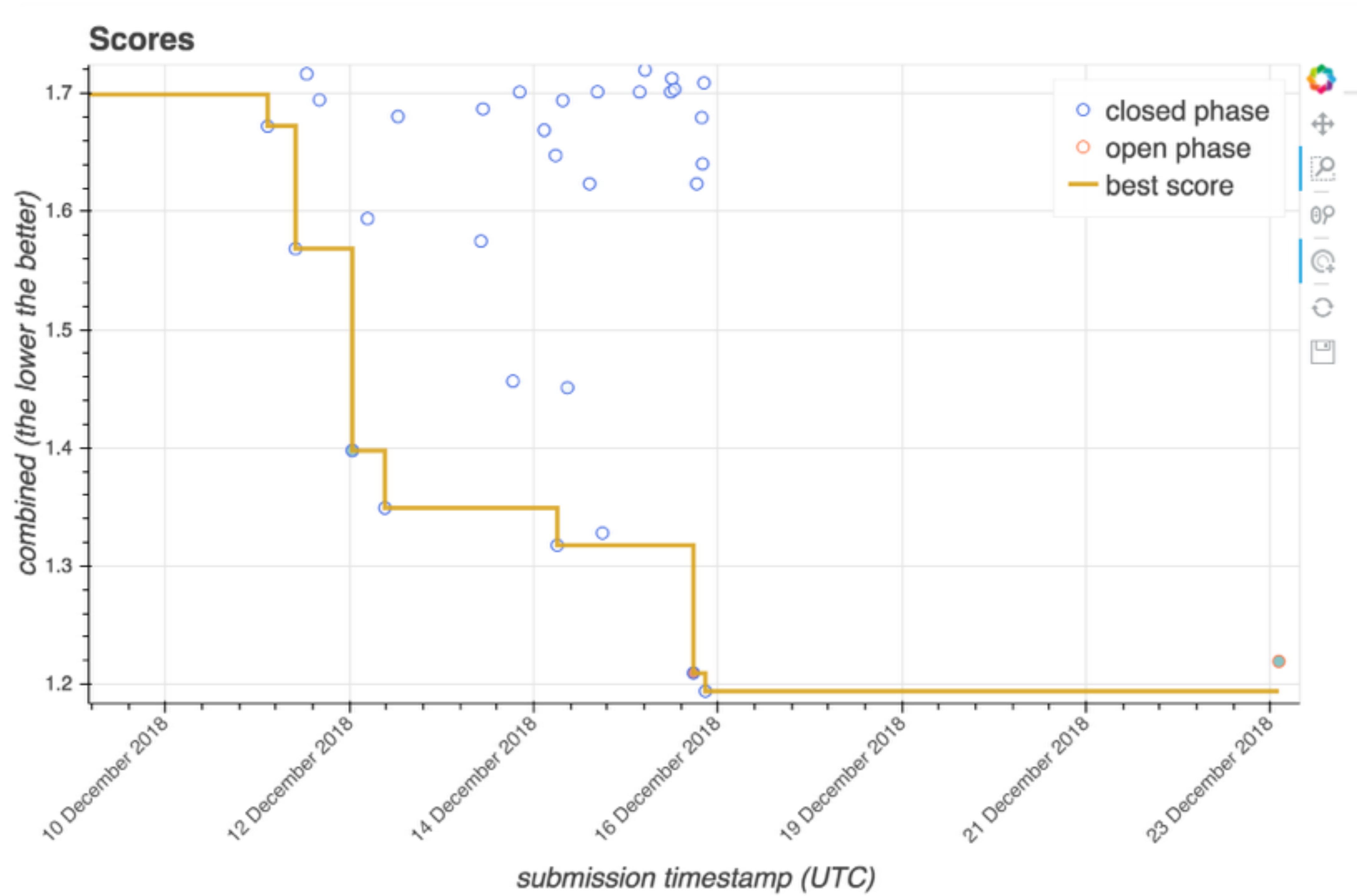
DISCOVERING PHYSICS (MECHANICS)



GRADUATE STUDENT DESCENT

Saclay MSc students competing and learning from each other submitting 40 predictive models in three weeks

Mechanics classification, Saclay M2 Data Camp 2018/19



We built and optimized
~20 predictive workflows
for three years

Funded by Université Paris-Saclay and CNRS

Team


Balázs Kégl


Alex Gramfort


Akin Kazakçi


Mehdi Cherti


Yohann Sitruk


Guillaume Lemaître


Alexandre Boucaud


Joris Van den Bossche

Alumni


Djalel Benbouzid

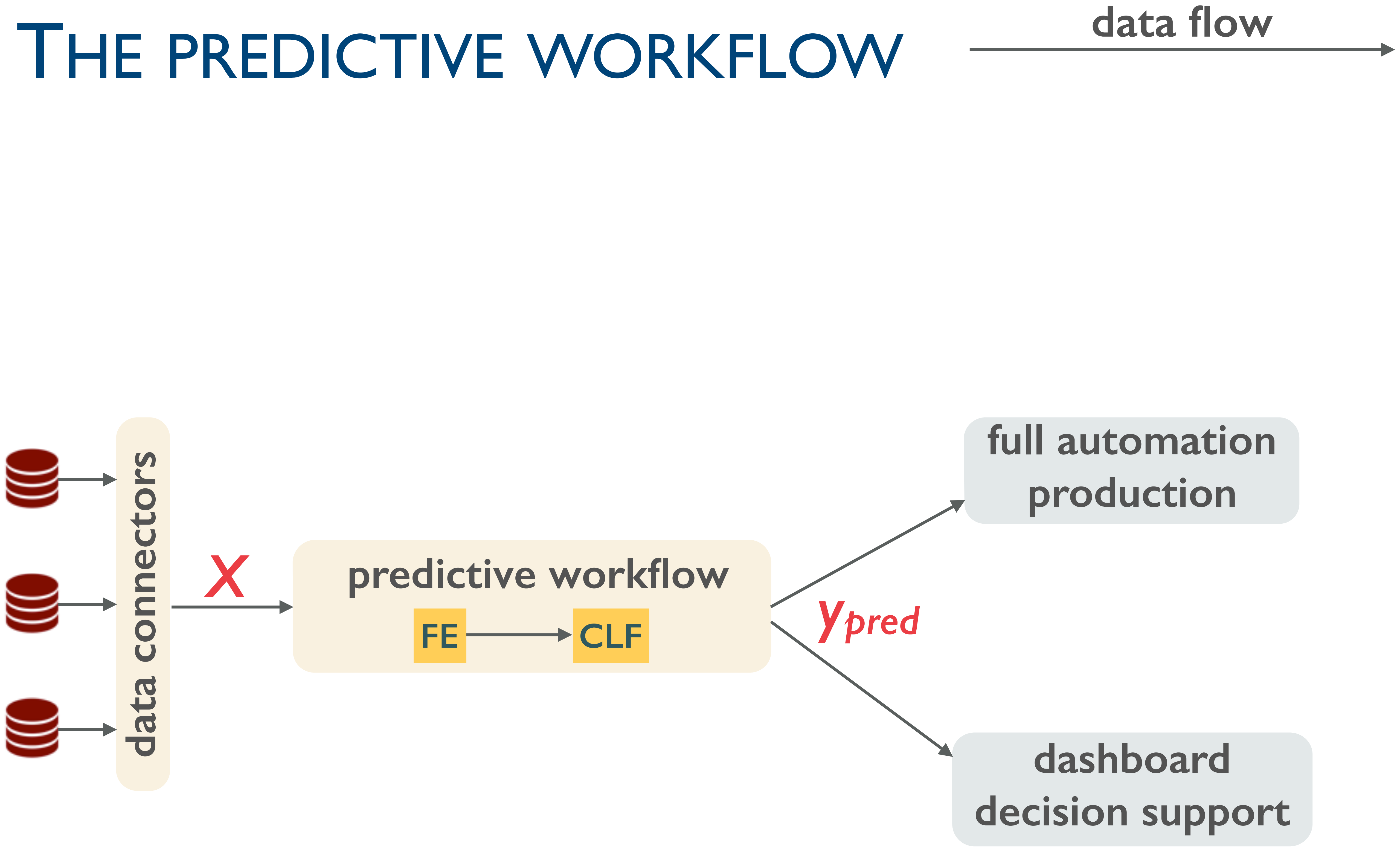

Camille Marini

What have we learned?

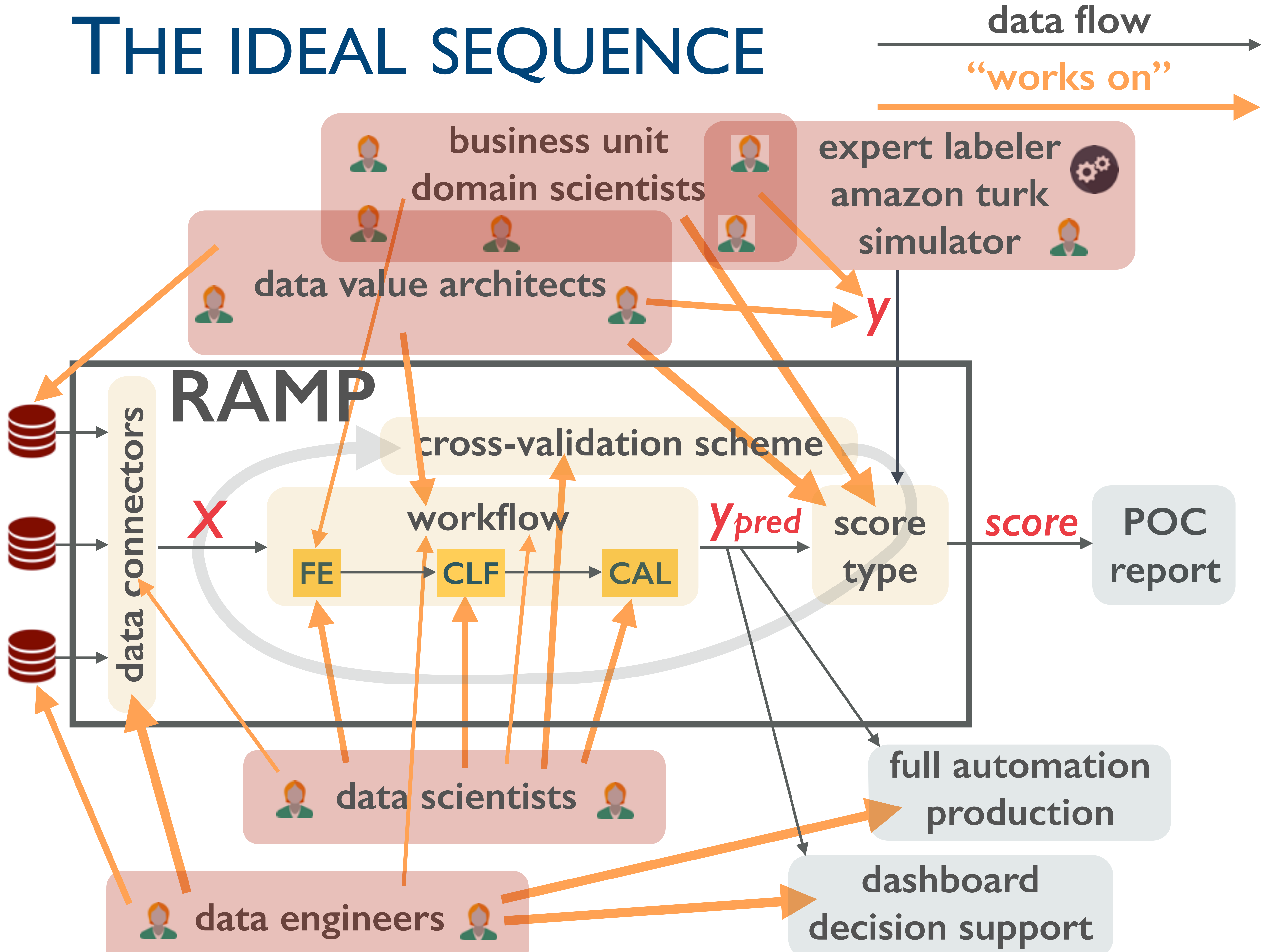
Building predictive workflows
what have we learned?

Building the workflow:
what are the tasks and who does what

THE PREDICTIVE WORKFLOW



The ideal sequence



Building predictive workflows
what have we learned?

What is a predictive workflow?

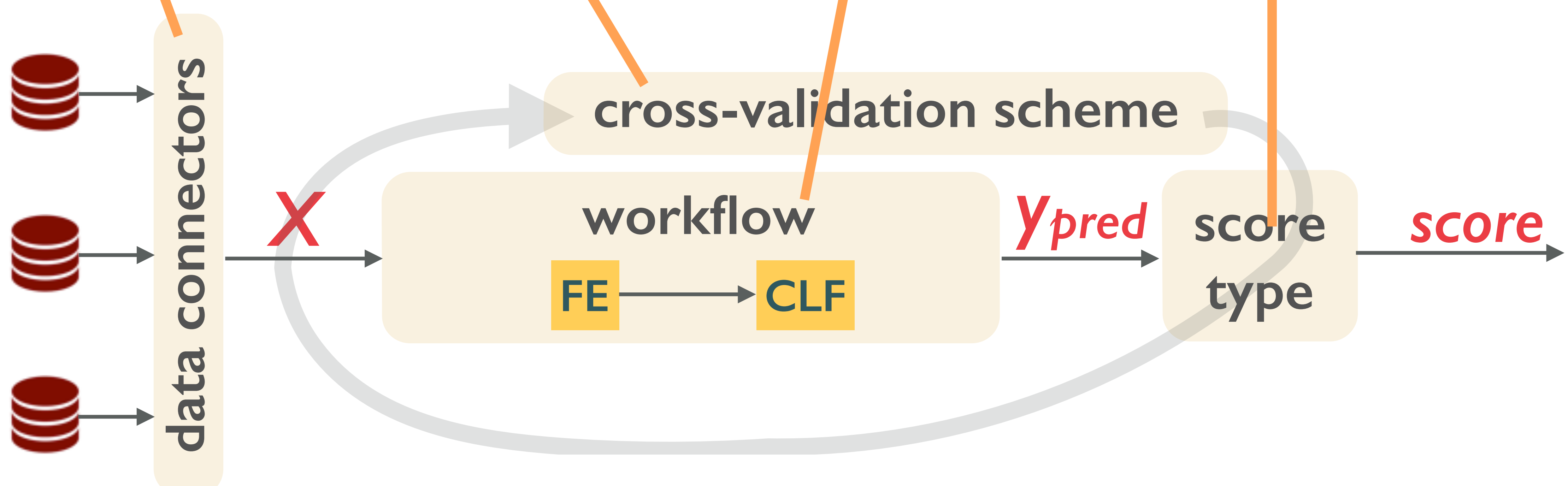
What are the parametrizable components?

What can be put into a unique training/scoring script?

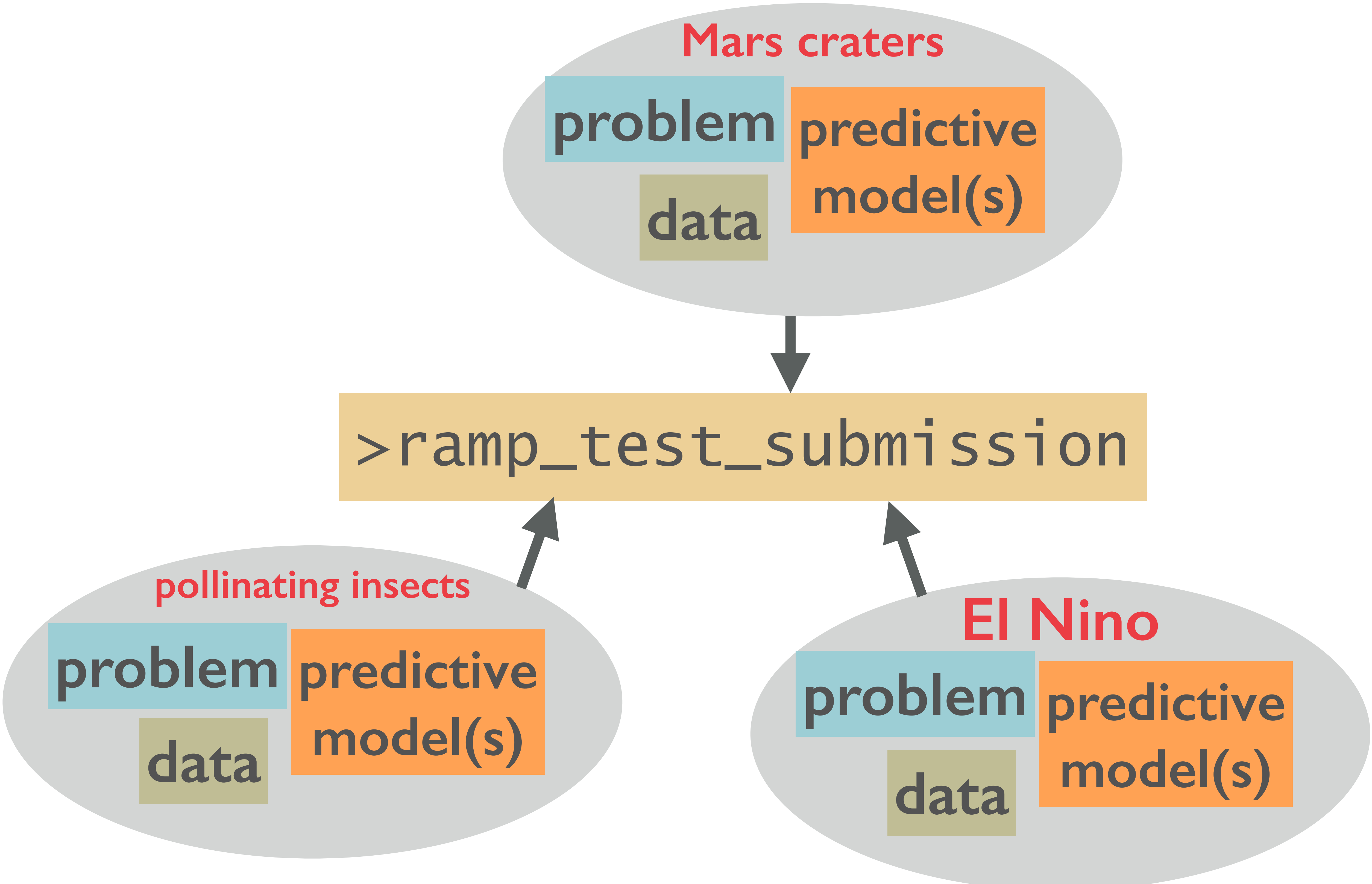
A SINGLE SCRIPT TO DEFINE THE BUNDLE

```python


def get_cv(X, y):
    unique_replicates = np.unique(X['replicate'])
    r = np.arange(len(X))
    for replicate in unique_replicates:
        train_is = r[(X['replicate'] != replicate).values]
        test_is = r[(X['replicate'] == replicate).values]
        yield train_is, test_is


def _read_data(path, f_name):
    data = pd.read_csv(os.path.join(path, 'data', f_name))
    y_array = data[_target_column_name]
    X_df = data.drop([_target_column_name], axis=1)
    return X_df, y_array


def get_train_data(path='.'):
    f_name = 'train.csv.gz'
    return _read_data(path, f_name)


def get_test_data(path='.'):
    f_name = 'test.csv.gz'
    return _read_data(path, f_name)
```

```python
import os
import numpy as np
import pandas as pd
import rampwf as rw

problem_title = \
    'Cell population identification from single-cell mass cytometry data'
_target_column_name = 'cell type'
_prediction_label_names = [
    'B-cell Frac A-C (pro-B cells)', 'Basophils', 'CD4 T cells', 'CD8 T cells',
    'CLP', 'CMP', 'Classical Monocytes', 'Eosinophils', 'GMP', 'HSC',
    'IgD- IgMpos B cells', 'IgDpos IgMpos B cells', 'IgM- IgD- B-cells',
    'Intermediate Monocytes', 'MEP', 'MPP', 'Macrophages', 'NK cells',
    'NKT cells', 'Non-Classical Monocytes', 'Plasma Cells', 'gd T cells',
    'mDCs', 'pDCs']
# A type (class) which will be used to create wrapper objects for y_pred
Predictions = rw.prediction_types.make_multiclass(
    label_names=_prediction_label_names)
# An object implementing the workflow
workflow = rw.workflows.FeatureExtractorClassifier()

score_types = [
    rw.score_types.BalancedAccuracy(name='bac', precision=3),
    rw.score_types.Accuracy(name='acc', precision=3),
    rw.score_types.NegativeLogLikelihood(name='nll', precision=3),
]
```



A UNIQUE SCRIPT TO RUN THE BUNDLES



A UNIQUE SCRIPT TO RUN THE BUNDLES

```
1 read training and test data
2 read submission
3 create train and valid folds
  on training data
4 for all train and valid folds:
5       train submission on train
6       score submission on train,
        valid, and test
7 summarize scores
```

```
silver6:autism kegl$ ramp_test_submission
Testing Autism Spectrum Disorder classification
Reading train and test files from ./data ...
Reading cv ...
Training ./submissions/starting_kit ...
CV fold 0
Couldn't re-order the score matrix..
        score    acc    auc
        test   0.696  0.765
        train  0.767  0.847
        valid  0.611  0.647
CV fold 1
Couldn't re-order the score matrix..
        score    acc    auc
        test   0.478  0.659
        train  0.766  0.842
        valid  0.628  0.662
CV fold 2
Couldn't re-order the score matrix..
        score    acc    auc
        test   0.609  0.720
        train  0.786  0.854
        valid  0.615  0.645
CV fold 3
Couldn't re-order the score matrix..
        score    acc    auc
        test   0.565  0.758
        train  0.769  0.849
        valid  0.619  0.645
----------------------------
Mean CV scores
----------------------------
Couldn't re-order the score matrix..
        score             acc               auc
        test   0.587 ± 0.0784    0.725 ± 0.042
        train  0.772 ± 0.0081   0.848 ± 0.0042
        valid  0.618 ± 0.0065    0.65 ± 0.0072
----------------------------
Bagged scores
----------------------------
Couldn't re-order the score matrix..
        score    auc
        test   0.735
        valid  0.647
```

RAMP-WORKFLOW & RAMP-KITS

- toolkit: https://github.com/paris-saclay-cds/ramp-workflow
  - for **designing workflows**
  - set of ready-made **metrics, workflows, CV schemes,** data readers
  - `ramp_test_submission`: unique command-line **test script**
- examples: https://github.com/ramp-kits
  - a zoo of **problems**, **experiments**, **workflows**
  - (at least) one **initial solution**

Building predictive workflows what have we learned?

How to make
(novice) data scientists efficient

How to make data scientists efficient

- Principles
  - **incite them** to work on the problem
  - give them a working (but unoptimized) **model to start with**
  - make **incremental contributions** easy
  - **gamification**
  - help them to **collaborate** and to **learn from each other**
  - **"hide" heavy engineering** and computational obstacles

The Jupyter notebook

- Concise description of the
  - scientific / business goal
  - the data
  - what are the steps, what do I have to do
  - initial solution
- Make the data scientist operational in a couple of hours

[
](http://www.datascience-paris-saclay.fr)

RAMP on Mars craters detection

Alexandre Boucaud (CDS), Joris van den Bossche (CDS), Balazs Kegl (CDS), Frédéric Schmidt (GEOPS), Anthony Lagain (GEOPS)

1. Introduction
2. Preprocessing
3. Workflow
4. Evaluation
5. Local testing/exploration
6. Submission

Introduction

Impact craters in planetary science are used to date planetary surfaces, to characterize surface processes and to study the upper crust of terrestrial bodies in our Solar System (Melosh, 1989). Thanks to the Martian crater morphology, a wide amount of information could be deduced on the geological history of Mars, as for example the evolution of the surface erosion rate, the presence of liquid water in the past, the volcanic episodes or the volatiles layer in the subsurface (Carr & Head, 2010). These studies are widely facilitated by the availability of reference crater databases.

Surveying impact craters is therefore an important task which traditionally has been achieved by means of visual inspection of images. The enormous number of craters smaller than one kilometer in diameter, present on high resolution images, makes visual counting of such craters impractical. In order to overcome this problem, several algorithms have been developed to automatically detect impact structures on planetary images (Bandeira et al., 2007 ; Martins et al., 2009). Nevertheless, these method allow to detect only 70-80 % of craters (Urbach & Stepinski, 2009).

The prediction task

This challenge proposes to design the best algorithm to detect crater position and size starting from the most complete Martian crater database containing 384 584 verified impact structures larger than one kilometer of diameter (Lagain et al. 2017). We propose to give to the users a subset of this large dataset in order to test and calibrate their algorithm.

THE INITIAL SOLUTION

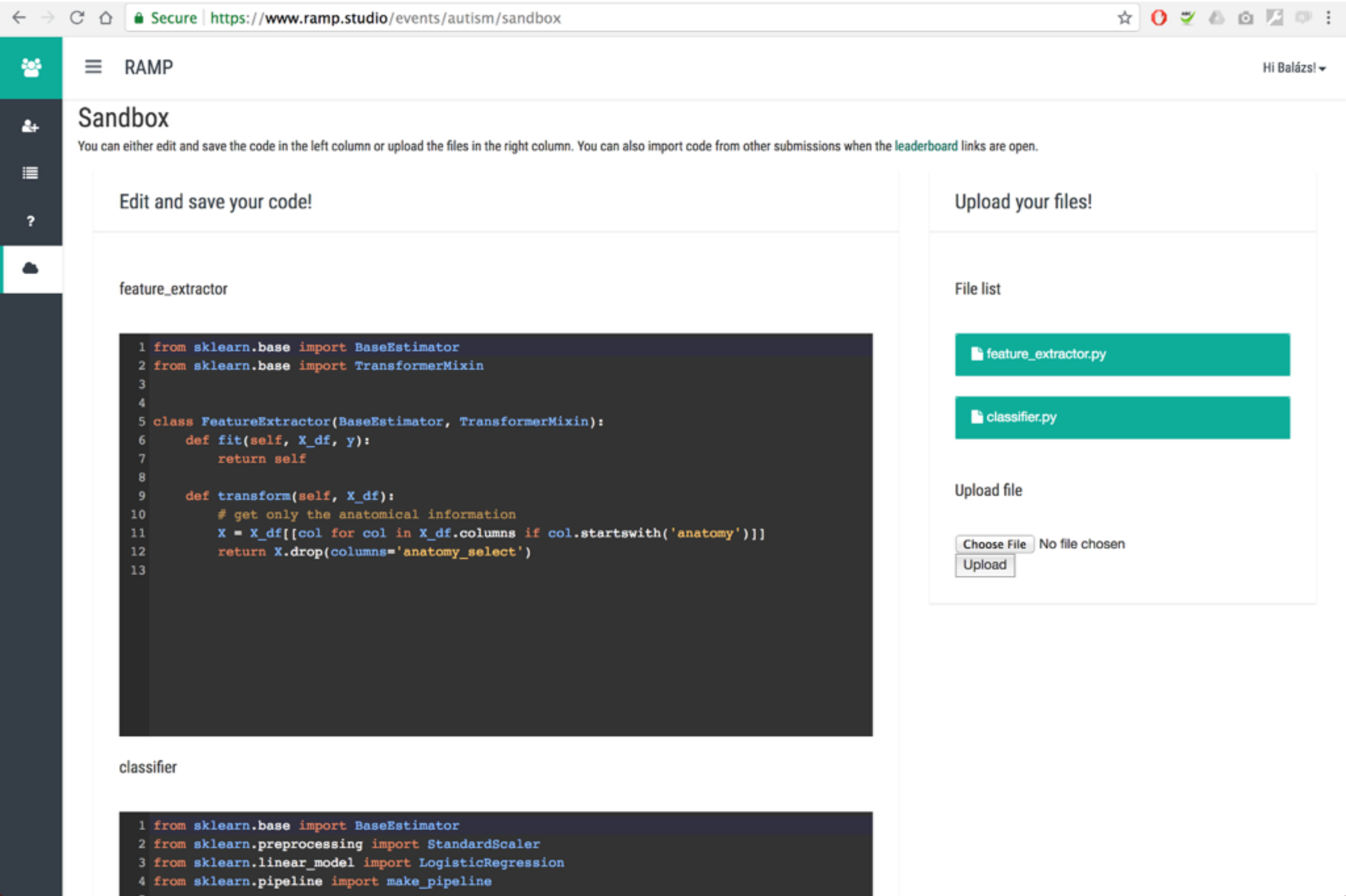
feature_extractor.py

```python
from sklearn.base import BaseEstimator
from sklearn.base import TransformerMixin


class FeatureExtractor(BaseEstimator, TransformerMixin):
    def fit(self, X_df, y):
        return self

    def transform(self, X_df):
        # get only the anatomical information
        X = X_df[[
            col for col in X_df.columns
            if col.startswith('anatomy')]]
        return X.drop(columns='anatomy_select')

```

classifier.py

```python
from sklearn.base import BaseEstimator
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import make_pipeline


class Classifier(BaseEstimator):
    def __init__(self):
        self.clf = make_pipeline(
            StandardScaler(), LogisticRegression(C=1.))

    def fit(self, X, y):
        self.clf.fit(X, y)
        return self

    def predict(self, X, y):
        return self.clf.predict(X)

    def predict_proba(self, X):
        return self.clf.predict_proba(X)

```

THE FRONTEND



THE LEADERBOARD

Secure https://www.ramp.studio/events/mars_craters_saclay_datacamp_17/competition_leaderboard

RAMP

Hi Balázs!

mars_craters_saclay_datacamp_17

Leaderboard

Combined score: 0.269

Show 10 entries

Search:

rank	team	submission	ospa	train time [s]	test time [s]	submitted at (UTC)
1	qixiang.peng	resnet	0.451	6520	455	2018-01-30 16:06:56 Tue
2	mogolola	submit_ssd_vgg_base	0.452	1649	58	2017-12-19 08:24:18 Tue
3	bruckert.alexandre	ssd_vgg	0.465	19795	308	2018-01-16 01:24:51 Tue
4	imbert.arthur	dorante_2	0.469	1039	568	2017-12-24 09:03:25 Sun
5	glemaitre	haar_like_all_feat	0.473	1601	536	2018-01-18 13:16:17 Thu
6	schehtman	basic_keras_ssd	0.523	1519	705	2017-12-14 00:02:33 Thu
7	boyao.zhou	test	0.525	1456	708	2018-01-31 17:28:35 Wed
8	shuopeng.wang	test	0.528	1613	709	2017-12-11 16:14:32 Mon
9	jorisvandenbossche	keras_ssd7_basic	0.533	1574	733	2017-11-06 14:38:52 Mon
10	kexin.tang	keras_ssd7_test	0.538	69241	2147	2017-12-14 10:28:24 Thu

THE BACKEND ON AMAZON WEB SERVICES

Launch Instance | Connect | Actions

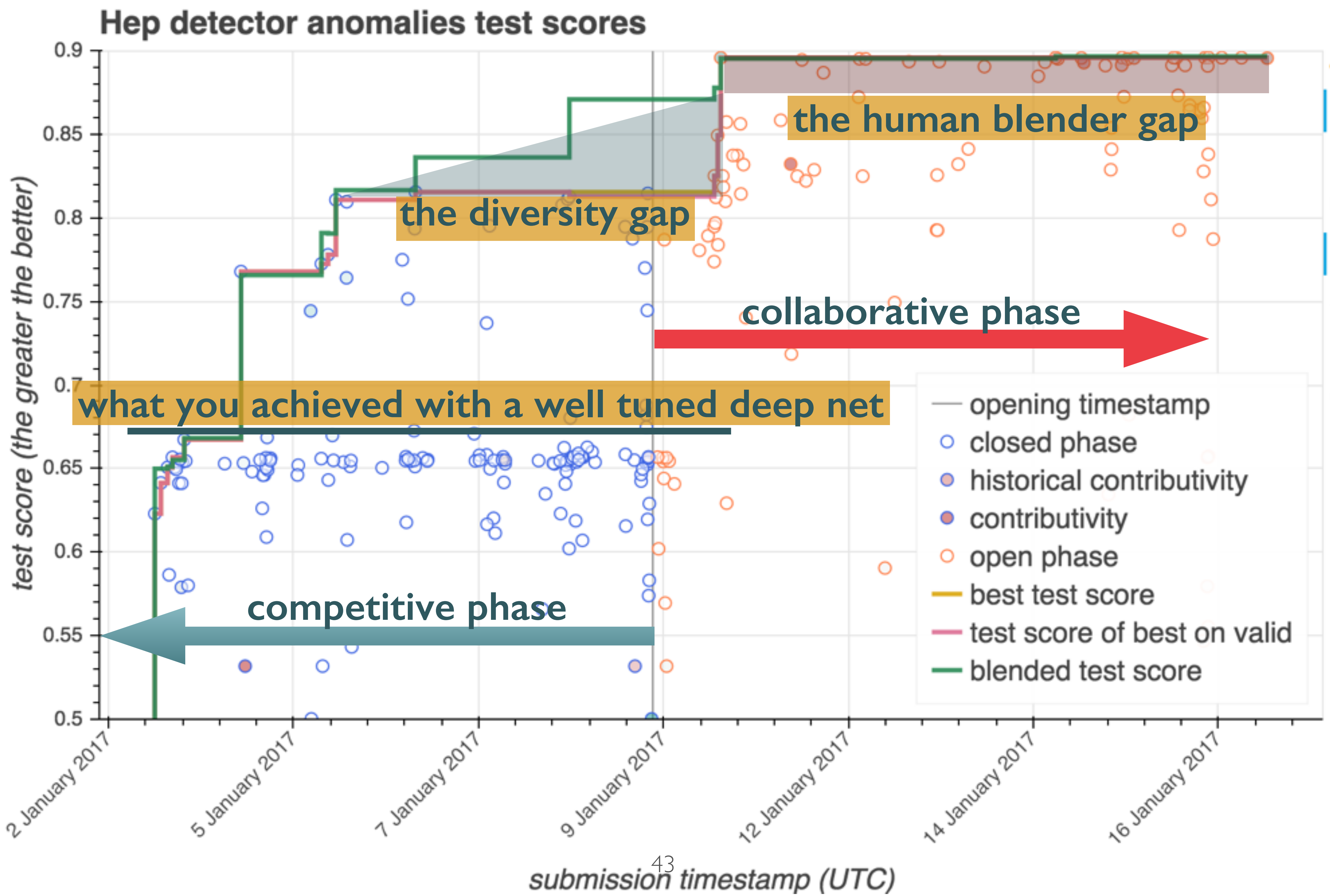
Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance Stat
8475_submission_id0	i-0f9411820a116930e	m5.xlarge	us-west-2a	terminatec
8474_test2	i-0a8afc1be7d824cb1	m5.xlarge	us-west-2a	terminatec
8473_test	i-0322ebb27a7e4e5...	m5.xlarge	us-west-2a	terminatec
8473_test	i-09f50b72b6a3a0155	m5.xlarge	us-west-2a	terminatec
8472_TunedClassif+svc	i-04dca6a8fd8738b5e	t2.small	us-west-2c	terminatec
8472_TunedClassif+svc	i-0f0529a6ab890f17c	t2.small	us-west-2b	terminatec
8469_test2	i-049ba0f2c8075fdd6	m5.xlarge	us-west-2a	terminatec
8467_TunedClassif+svm	i-08018a0ee2b90b2ec	t2.small	us-west-2b	running

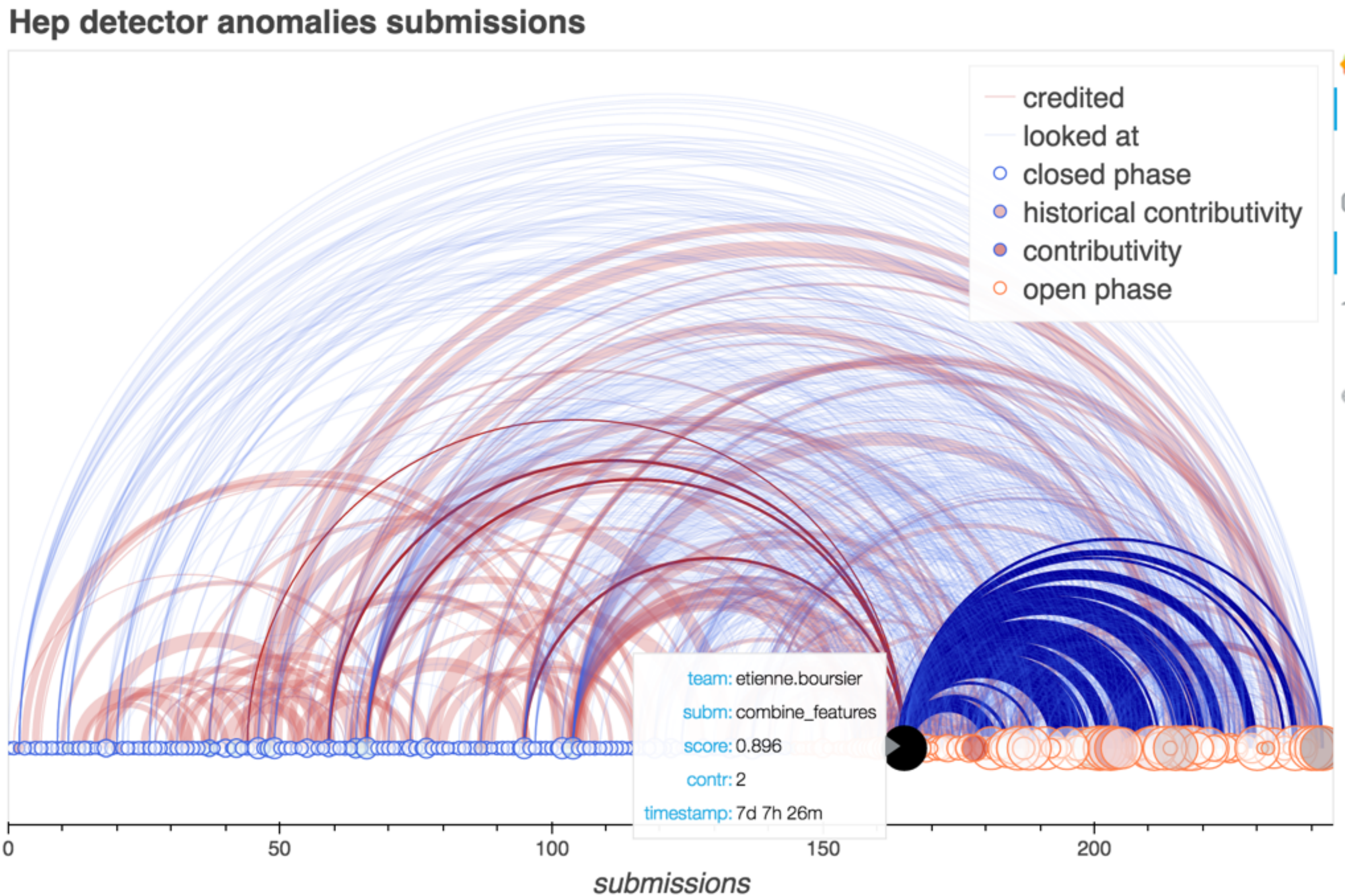
Why code submission

1. lets us deliver a working prototype
2. lets the participants collaborate

The power of the (collaborating) crowd
optimizing graduate student descent



COMMUNICATION AND REUSE



You can

1. Use RAMP in teaching or training
2. Use the toolkit for your own workflows

Links

frontend:

www.ramp.studio

toolkit:

github.com/paris-saclay-cds/ramp-workflow

server:

github.com/paris-saclay-cds/ramp-board

examples:

github.com/ramp-kits

slack:

ramp-studio.slack.com

LINKS

- **medium.com/@balazskegl**
  - The data science ecosystem (industrial edition)
  - Teaching the data science process
  - How to build a data science pipeline
- **RAMP paper**
  - https://openreview.net/forum?id=Syg4NHz4eQ