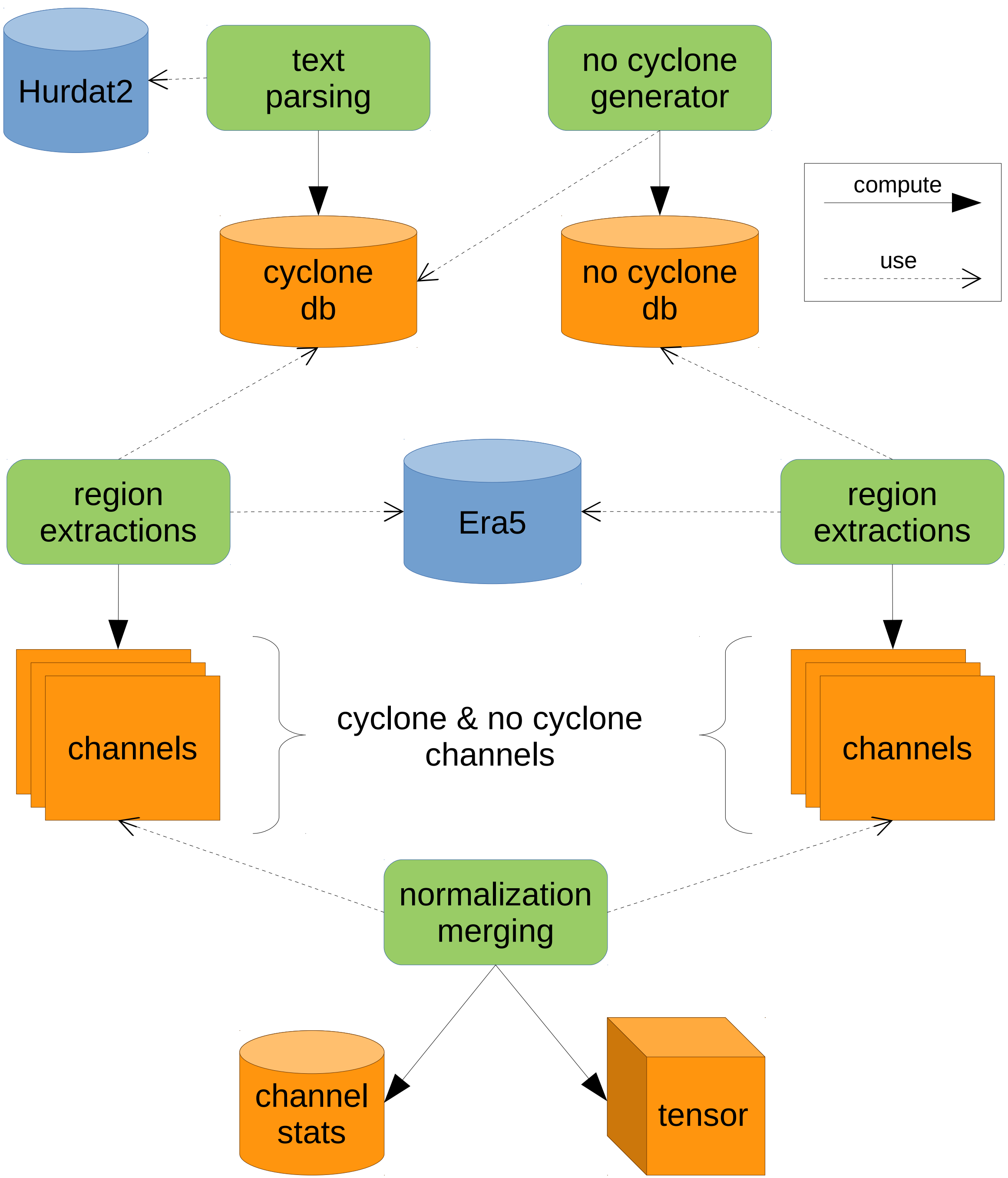
# CYCLONE DETECTION



Sébastien Gardoll

19/02/2019

# Data pre-processing




- Hurdat2: reanalysis of Atlantic basin cyclone observations from 1851 to 2017

- Using Era5 netcdf files (resolution: 0.25)

- Variables (8):
  - msl: mean surface pressure
  - ta200, ta500: temperature
  - u10, u850: wind W=>E
  - v10, v850 : wind N=>S
  - tcwv : total column water vapour

- Dataset 2000-2017
  - except 08/2000 (99 cyclone locations)
  - cyclone db: 4853 cyclone locations
  - no cyclone db: 9636 locations

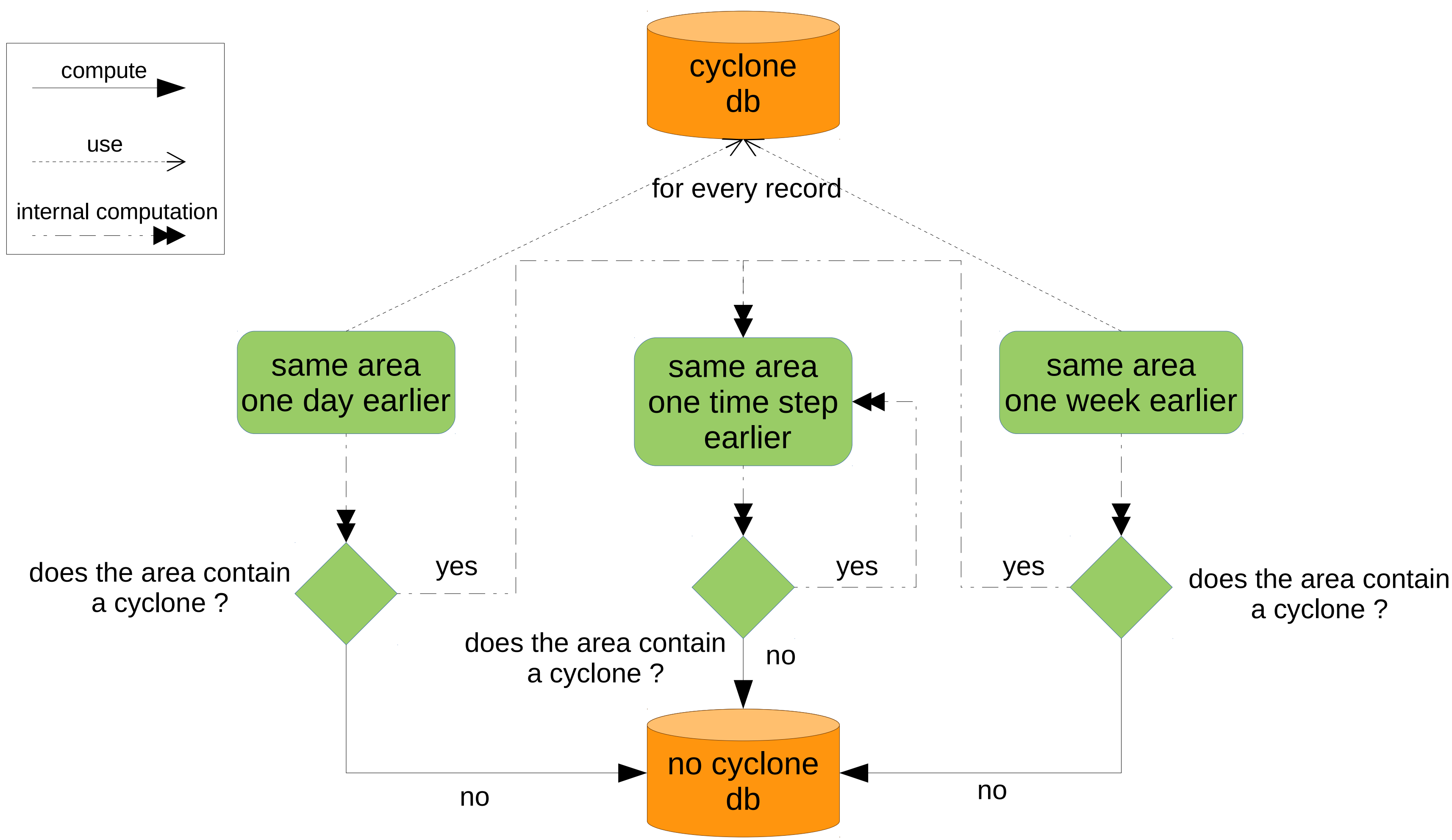
- Channel shape : 14489, 32, 32

- Tensor shape : 14489, 32, 32, 8 (453 Mo)

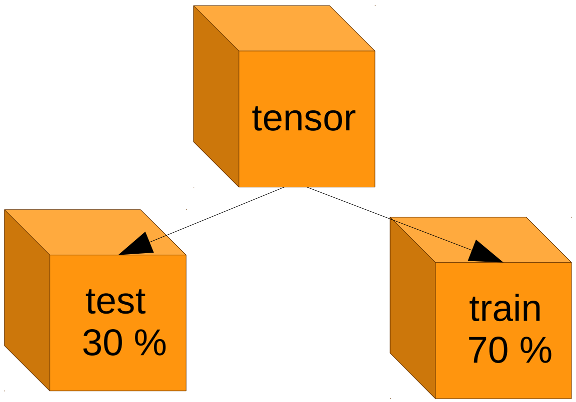
- Processing:
  - wall clock time: 15 mins (ciclad-ng)
  - ad hoc multi-processing design
    - 8 jobs
    - 4 processes/job
  - 1 Go RAM/job
  - 1728 netcdf files opened (> 150 Mo ; > 5 Go)
  - 1143 python sloc ; 123 bash sloc

- Xarray version :
  - takes around 100 Go RAM for msl
  - 15 mins for processing msl

# No Cyclone generator

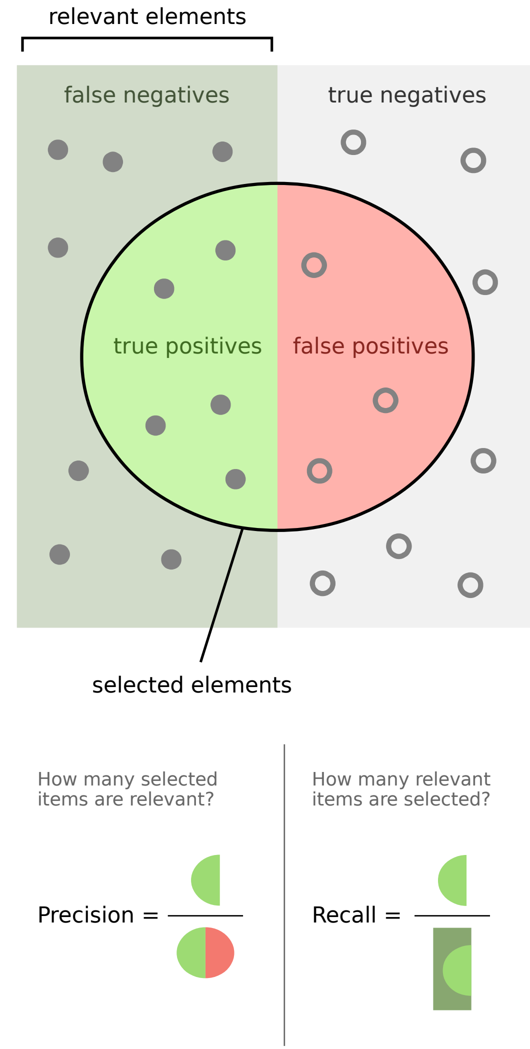


# ConvNet Training




(4347, 32, 32, 8) (10142, 32, 32, 8)

| | | |
|---|---|---|
| input | in | 32, 32, 8 |
| | out | 32, 32, 8 |
| conv1 | in | 32, 32, 8 |
| | out | 28, 28, 8 |
| max pooling1 | in | 28, 28, 8 |
| | out | 14, 14, 8 |
| conv2 | in | 14, 14, 8 |
| | out | 10, 10, 16 |
| max pooling2 | in | 10, 10, 16 |
| | out | 5, 5, 16 |
| flatten | in | 5, 5, 16 |
| | out | 400 |
| dense1 | in | 400 |
| | out | 50 |
| dense2 | in | 50 |
| | out | 2 |



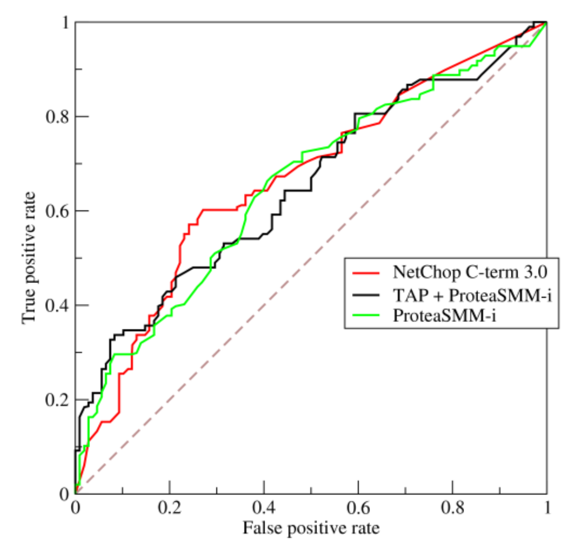

From Wikipedia, CC BY-SA 4.0

Conception from Y. Liu et al ; CoRR 2016 1605.01156

- Settings:
- Keras with TensorFlow backend
- batch_size: 5
- epochs: 75 (converges to 0.99 at 14)
- loss: binary_crossentropy
- metrics: accuracy
- optimizer: SGD

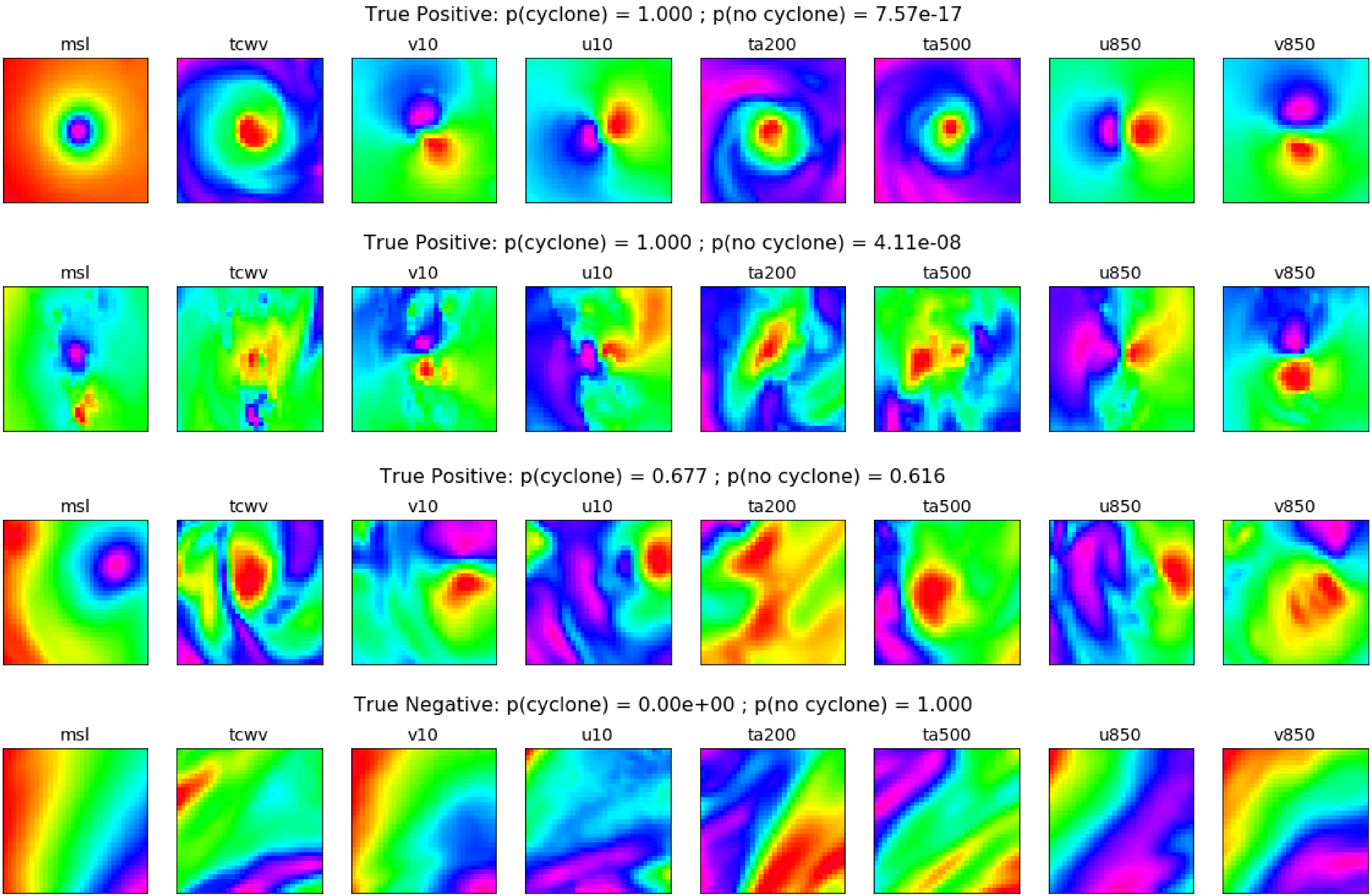
- Processing:
- wall clock time: 25 mins
- 1 node, 4 cores
- 133 python sloc ; 41 bash sloc

- Metrics:
- AUC: 0.9995
- accuracy: 0.9912
- loss: 0.0332

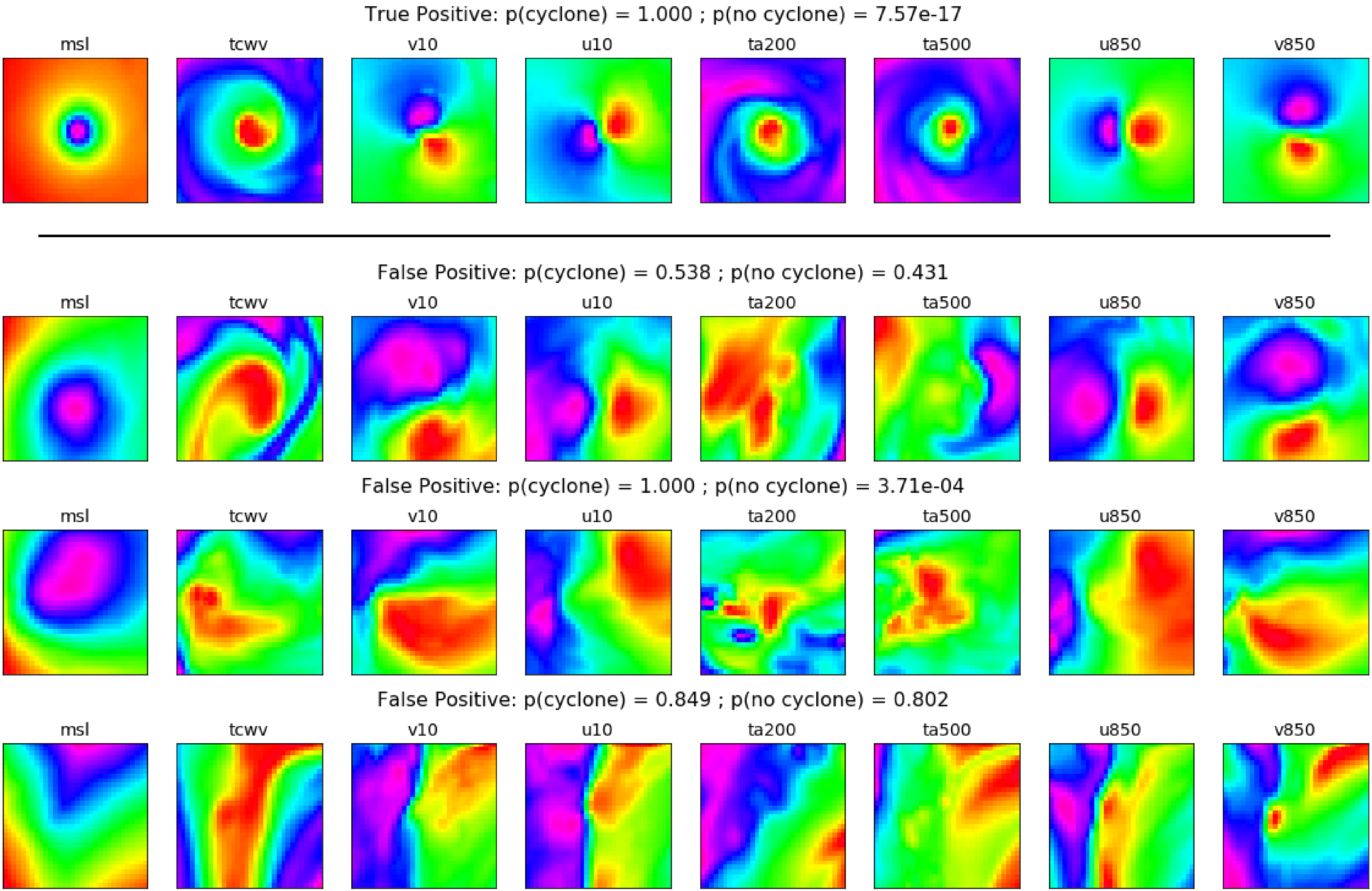



From Wikipedia CC BY-SA 3.0

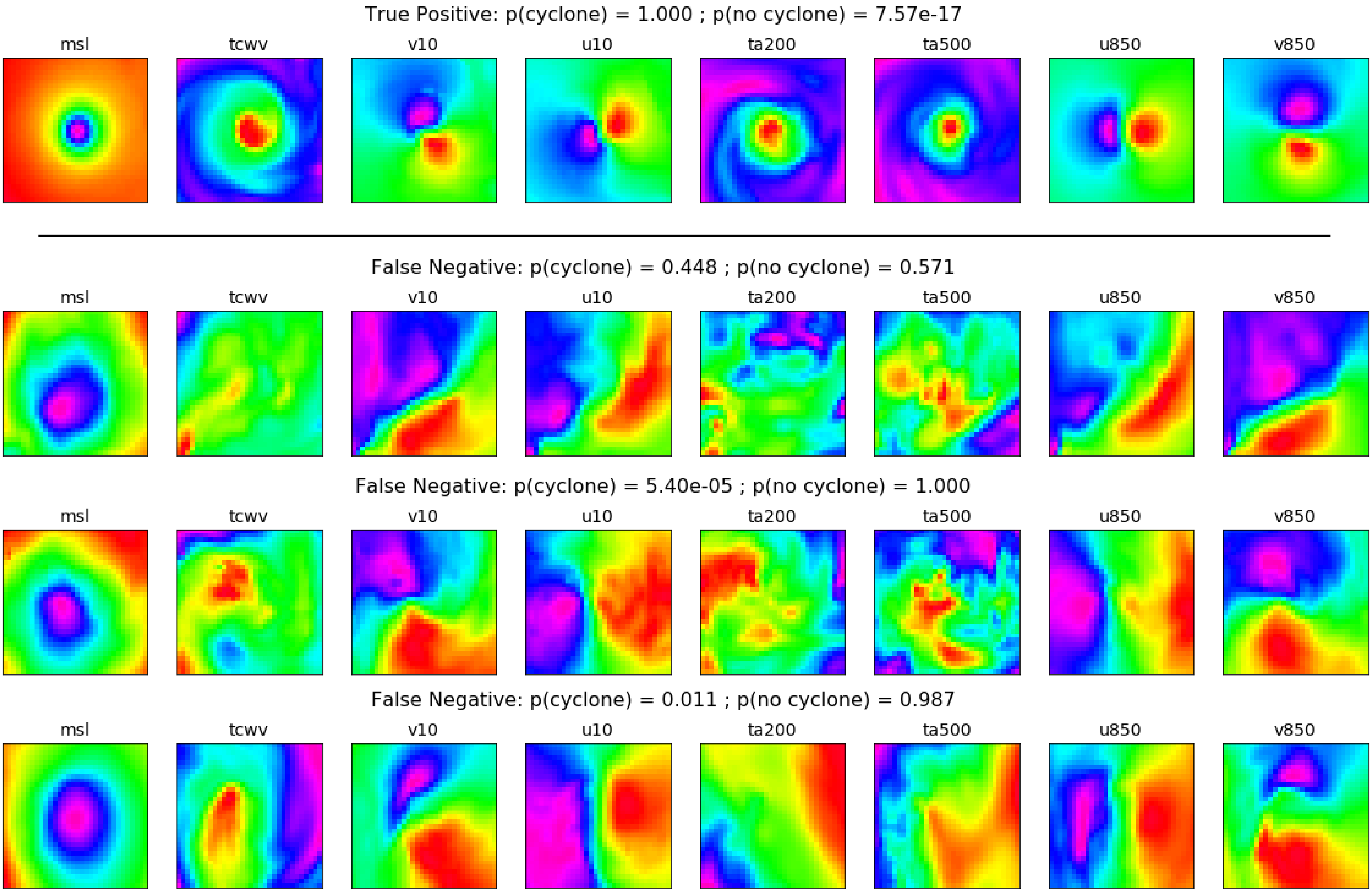
# True positives & negatives



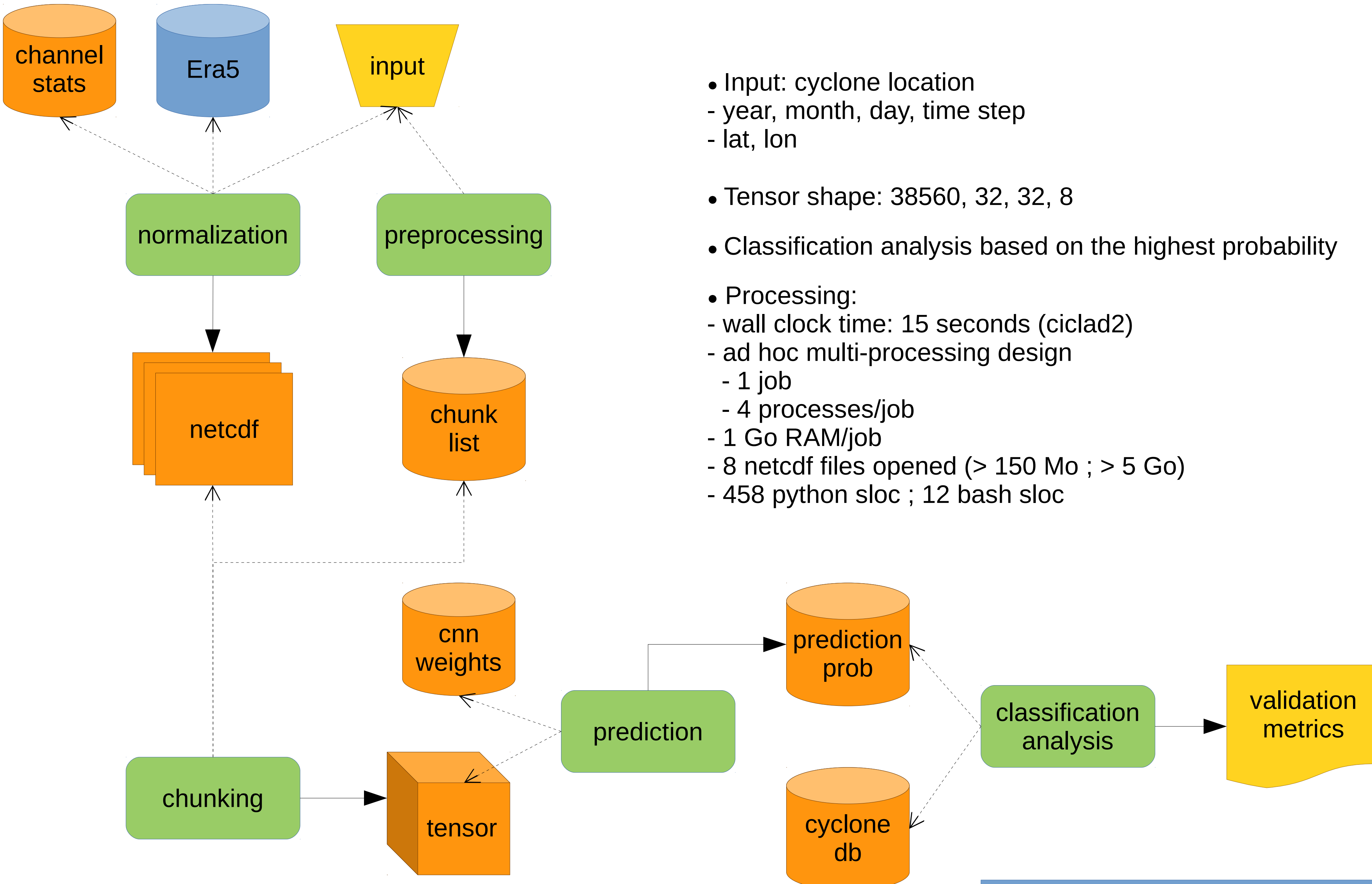
# False positives



# False negatives



# Prediction




- Input: cyclone location
- year, month, day, time step
- lat, lon

- Tensor shape: 38560, 32, 32, 8

- Classification analysis based on the highest probability

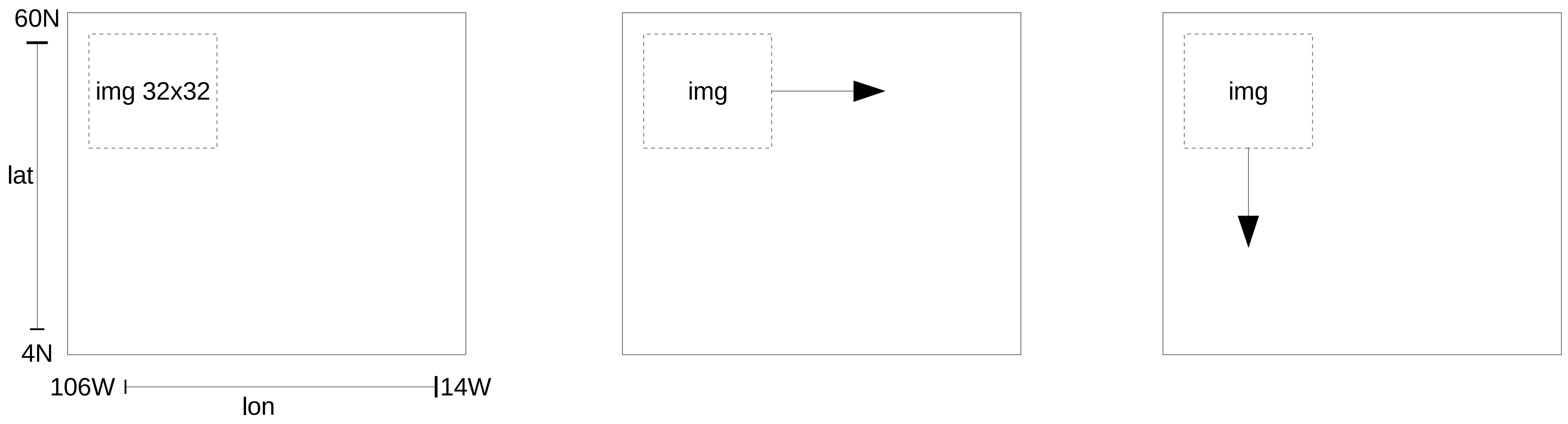
- Processing:
- wall clock time: 15 seconds (ciclad2)
- ad hoc multi-processing design
  - 1 job
  - 4 processes/job
- 1 Go RAM/job
- 8 netcdf files opened (> 150 Mo ; > 5 Go)
- 458 python sloc ; 12 bash sloc

# Chunking

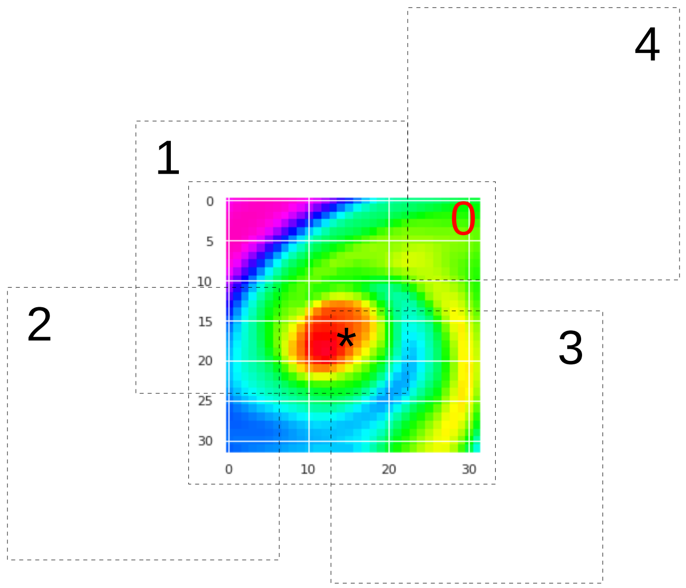
1. input area
(netcdf resolution: 0.25)

2. convolution pixel by
pixel along the two axis: x axis

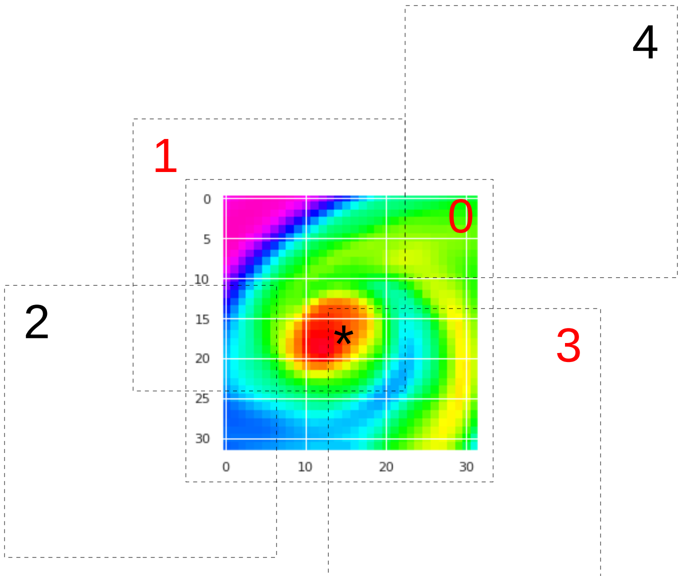
3. and the y axis

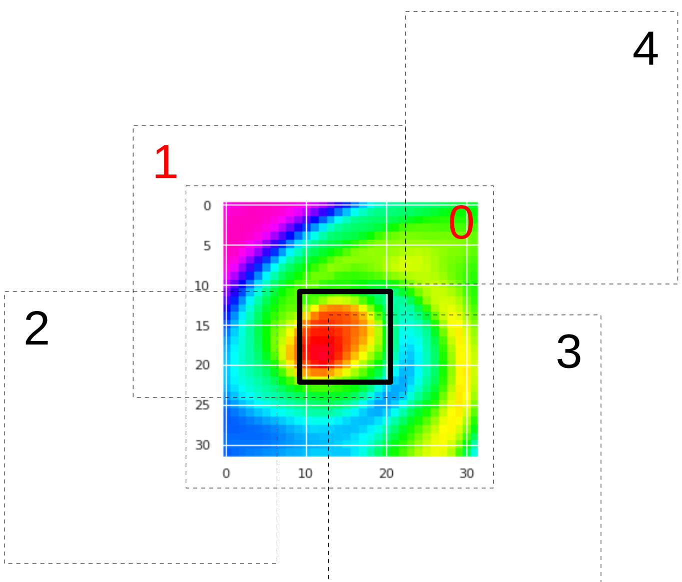



So as to generate 38 560 images that may be centered around a cyclone location



# Positives/negatives




Classification metrics imply the notion of positives/negatives.

Assumption for labelling the chunks:

Images that **contain** a specific area, are positives and the other images are negatives.

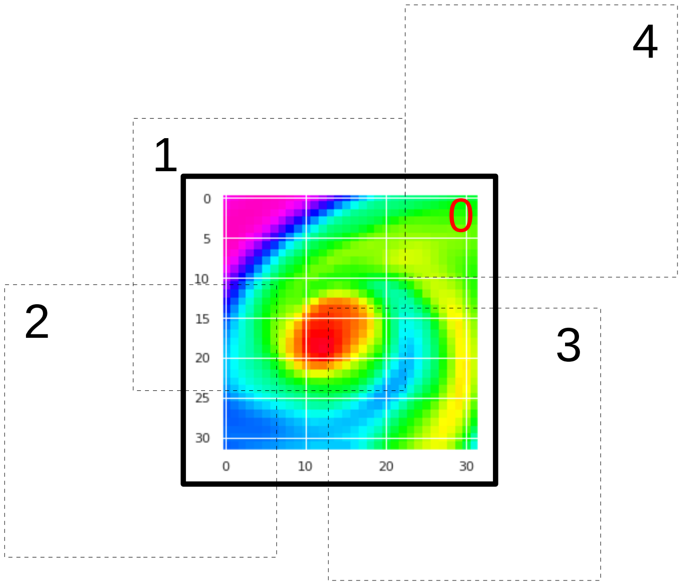
- If specific area = cyclone location (point from cyclone db)
⇒ images #0, #1 and #3 are positives
⇒ images #2 and #4 are negatives

- If specific area = zone with a cyclone location as its center
⇒ images #0 and #1 are positives
⇒ images #2, #3, #4 are negatives

# False positives

**Taking decision: specific area set as the dimension of the images.**

**⇒ only image #0 is positive**




Choosing a very restrictive way of labeling the positive images, makes **the model to produce a lot of false positives**.

Do the false positives contain a bit of the cyclone image (intersection) ?
Yes for all the 08/2000 cases (99 cyclone locations) !

**⇒ At first glance, false positives are artifacts introduced by the labeling method.**
**⇒ The precision for the label cyclone is not relevant (not yet).**

# Classification metrics

Performed on the cyclone locations of august 2000 (99 cyclone locations)

- all cases: the precision and recall of the no cyclone label are nearly equal to 1 .

- all cases: the precision of the cyclone label is less than 0.005 .

- 94/99 cases: the recall of the cyclone label is equal to 1 .

Conclusions:

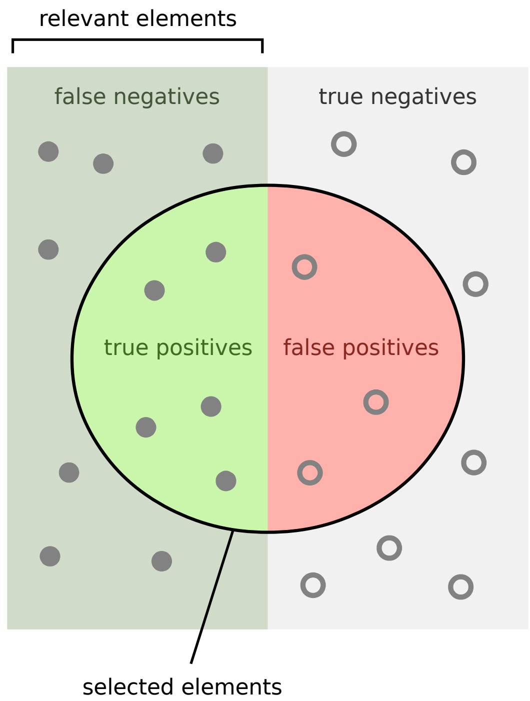
• Precision and recall of the no cyclone label are nearly perfect
**⇒ Classification of the no cyclone is nearly perfect.**
**⇒ The generator of no cyclone images is validated.**

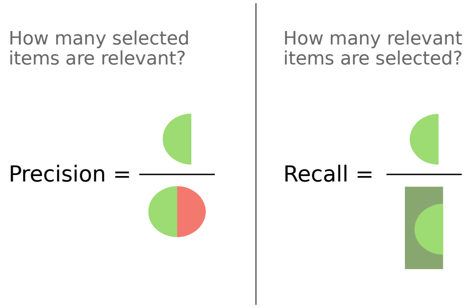
• Precision of the cyclone label is nearly null when considering the false positives. But as all the false positives contain a bit of the cyclone image, we can assume that this metric is not relevant for the moment.
**⇒ The location of the cyclones is still to be computed !**

• Recall of the cyclone label is perfect for 94/99 cases.
**⇒ Model still misses some cyclones but the exact location of the cyclone can be computed thanks to the false positives.**

• Naive attribution of label based on the highest probability is enough.

• Chunking algorithm may be improved (into something like facial recognition)



# What next ?

Thank you for your attention