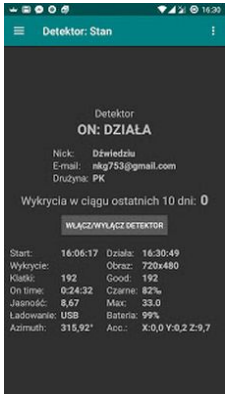
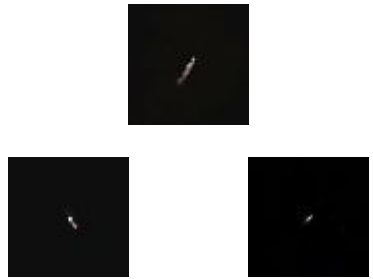
CREDO events analysis and machine learning

Michał Niedźwiecki
Cracow University of Technology

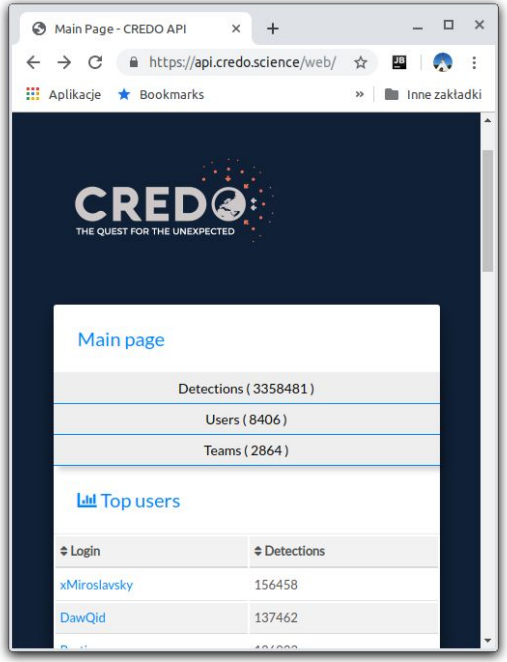
Cosmic-ray image acquisition



Android camera recording darkness
(in continuous preview mode of camera)

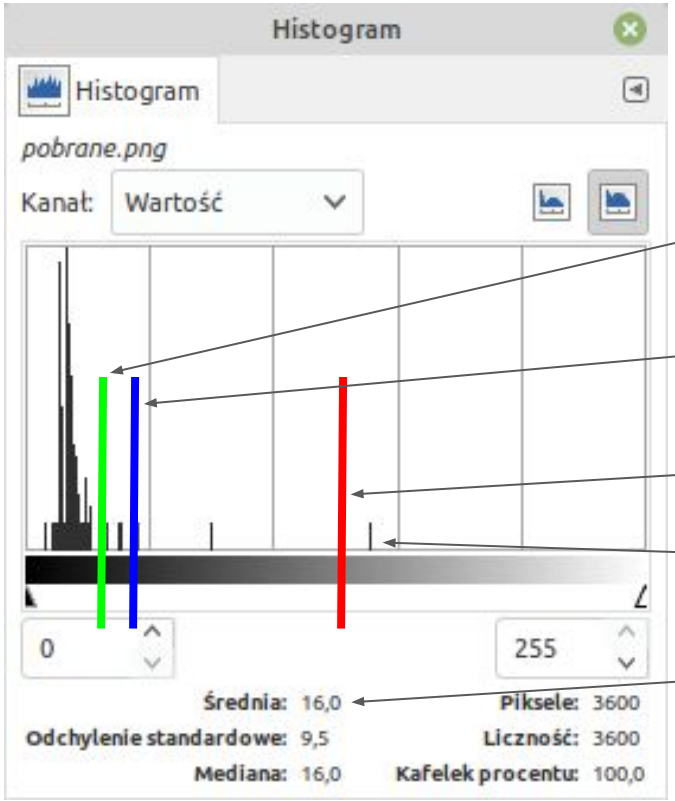


Sometimes in the darkness the bright anomaly was happen
(brightness than standard noise)



The anomalies was cropped and sent do CREDO server

Cosmic-ray cropping rules



Average threshold

Black threshold

Max threshold

Brightest pixel

Average bright of image frame

Good camera cover rules:

- ***average bright*** of frame is below the ***average*** threshold
- ***count of pixels*** brightest tan ***black*** threshold less than ***black count***

Cosmic-ray crop rules:

- ***brightest pixel*** is brighter than ***max*** threshold

The thresholds was computed in first launch by autocalibration

Technical problems with smartphones

- Hot pixels may produce false cosmic-ray detections (but it was occur on the same frame coordination)
- Bad camera coverage may produce false cosmic-ray hits (but it was exist in more than 3 next frames)
- API is public, user may produce false cosmic-ray detection (???)
- Smarpthone may be hang or turn off camera by OS.
- Time synchronization by Android is not ideal (~2s deviation), but we testing our in app time synchronization with NTP server (thanks for **Alex Ćwikła**, PK)
- More pixels, more data to parsing, more device hating, more noise and CPU throttling, app may lost image frames
- CMOS camera noise may be reduced by modern camera driver, but we develop detection method based by RAW frames (thanks for **Dariusz Burakowski**, UP)

Another device limitations

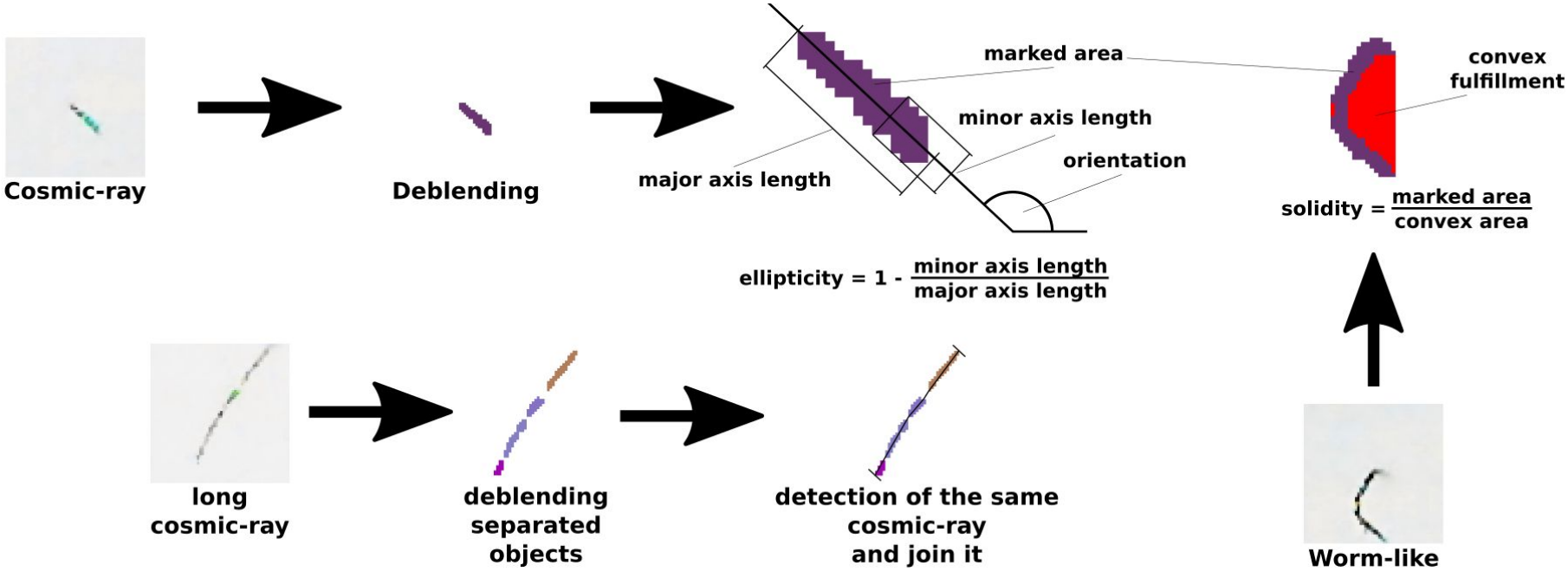
- GPS works only outside the building or near the window
- timestamp of detection have limitations despite the time synchronization:
  - start of image frame recording: ~ few seconds
  - frame rate - max 30 per second: 1/30s
  - timestamp of frame: ~20ms after end of frame reading, Android OS is not real time system,
  - RAW image mode significantly decrease time accuracy
- possible death time of detection (between frames, not always, and when CPU is to slow - lost frames)
- Android 9 and never forbid to camera working with screen off, smartphone with screen on consume more energy and more heating (like during video conference in Skype)

But we have very big database

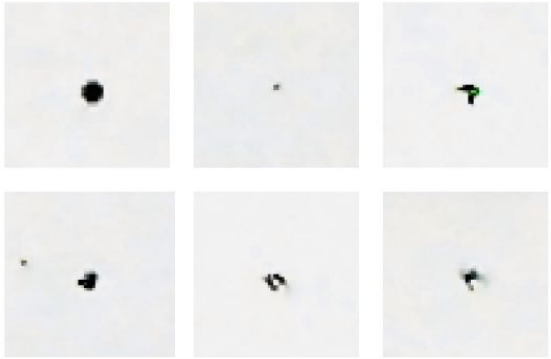
- CREDO database has ~4M cosmic-ray hits (2019-11-20)
- Each hit may be source of epoch discovery
- Each hit has many various attributes besides cosmic-ray image like GPS localisation, timestamp, smartphone model, XYZ orientation by accelerometer

- It is big data, we can do data mining:
  - Analysis each hit separately (features extraction),
  - Analysis big data

Image analysis (extract image features)



Simple classification by image features

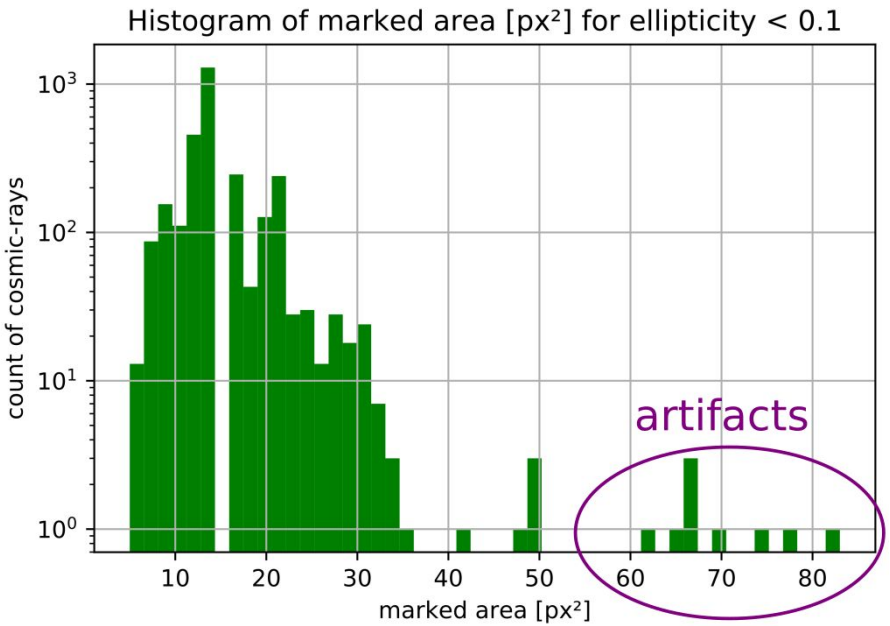


Spot-like events
ellipticity < 0.1

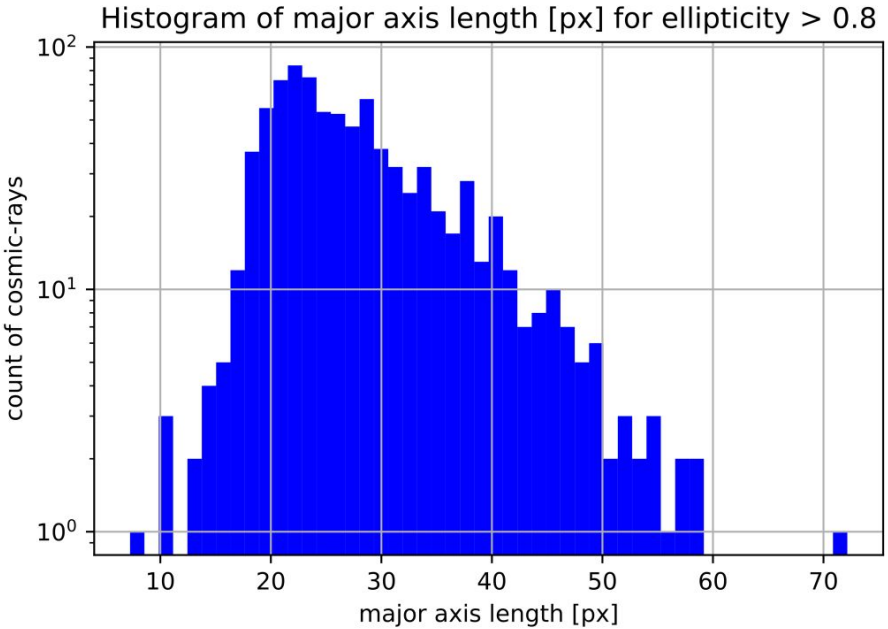
Track-like events
solidity > 0.7 & ellipticity > 0.6

Worm-like events
solidity < 0.7

Statistical analysis by extracted features and classes



Marked area of spot-like cosmic-rays



Length in pixels of track-like cosmic-rays

Can we extract more?

Yes, we can use machine learning to extract more informations:

- machine learning may better classify hits including source device differences by simplest algorithm and self autocallibration,
- machine learning may found anomalies (including artefacts and users cheatings),
- machine learning may cooperate with users (citizen science),
- machine learning may help to make new discoveries

What is machine learning?

Traditional programming

Machine learning

Data = hits images and attributes and extracted features

Machine learning more like gardening

- Seeds = Algorithms
- Nutrients = Data
- Gardener = You
- Plants = Programs

Types of Machine Learning

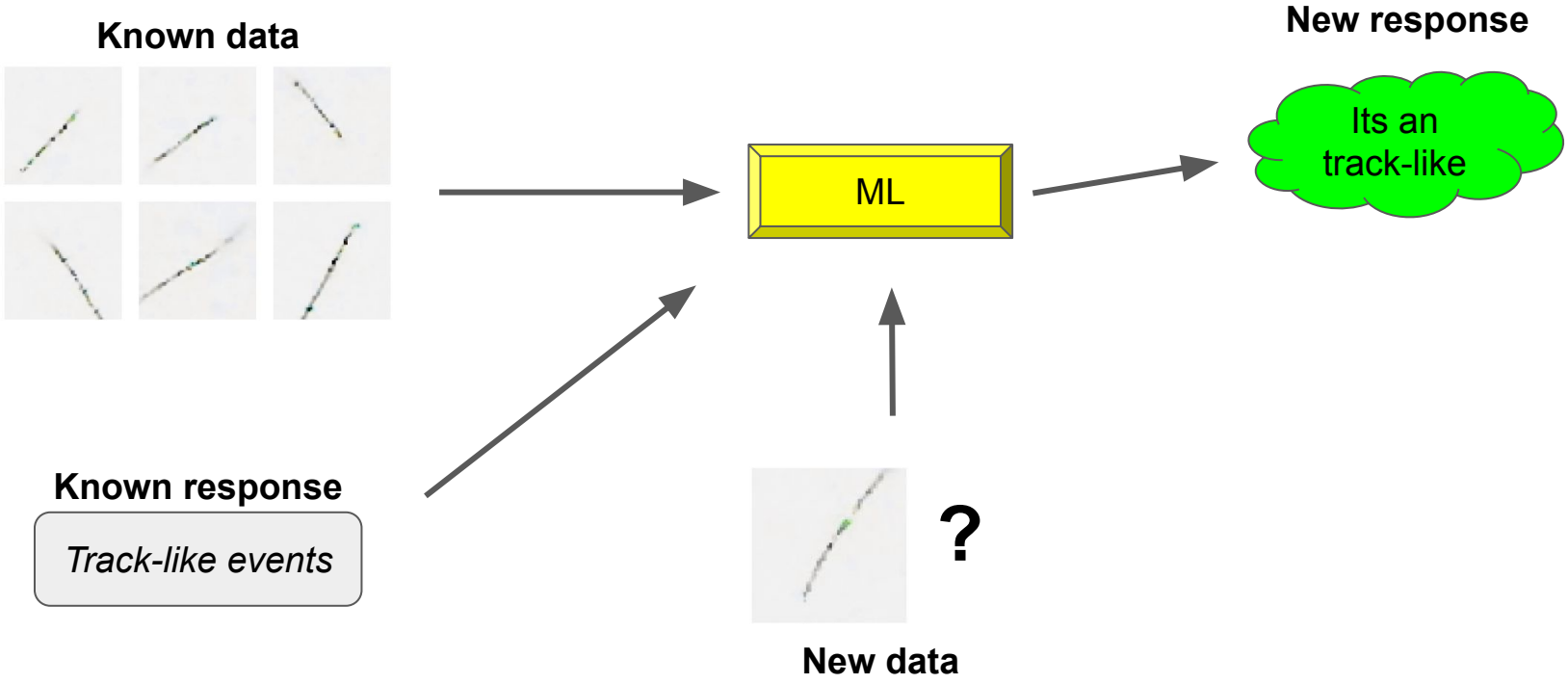
Basic machine learning types:

- supervised (the machine learns from the training data that is labelled)
- unsupervised (non-labelled training data)
- reinforcement (the machine learns on its own)

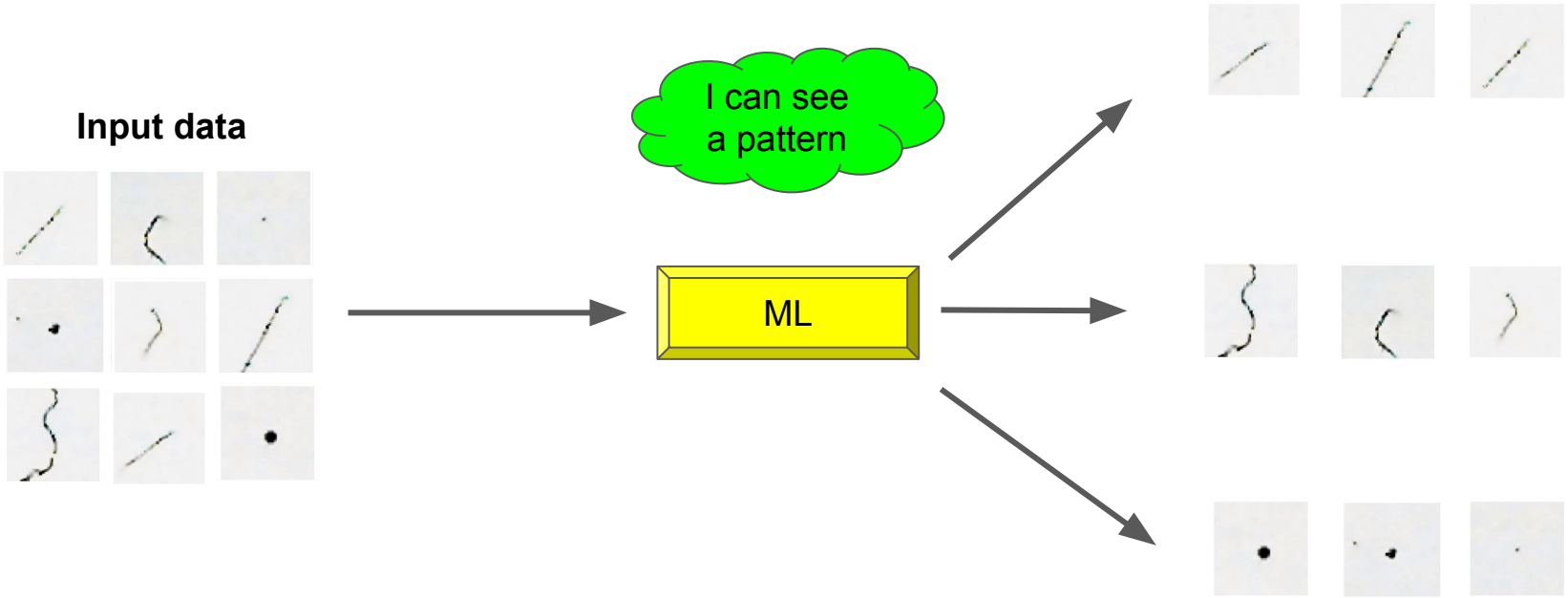
Right machine learning solutions:

- classification (used when the output is categorical like “spot”, “track”, “worm”)
- clustering (used when the data needs to be organized to find patterns)
- regression (used when a value needs to be predicted)

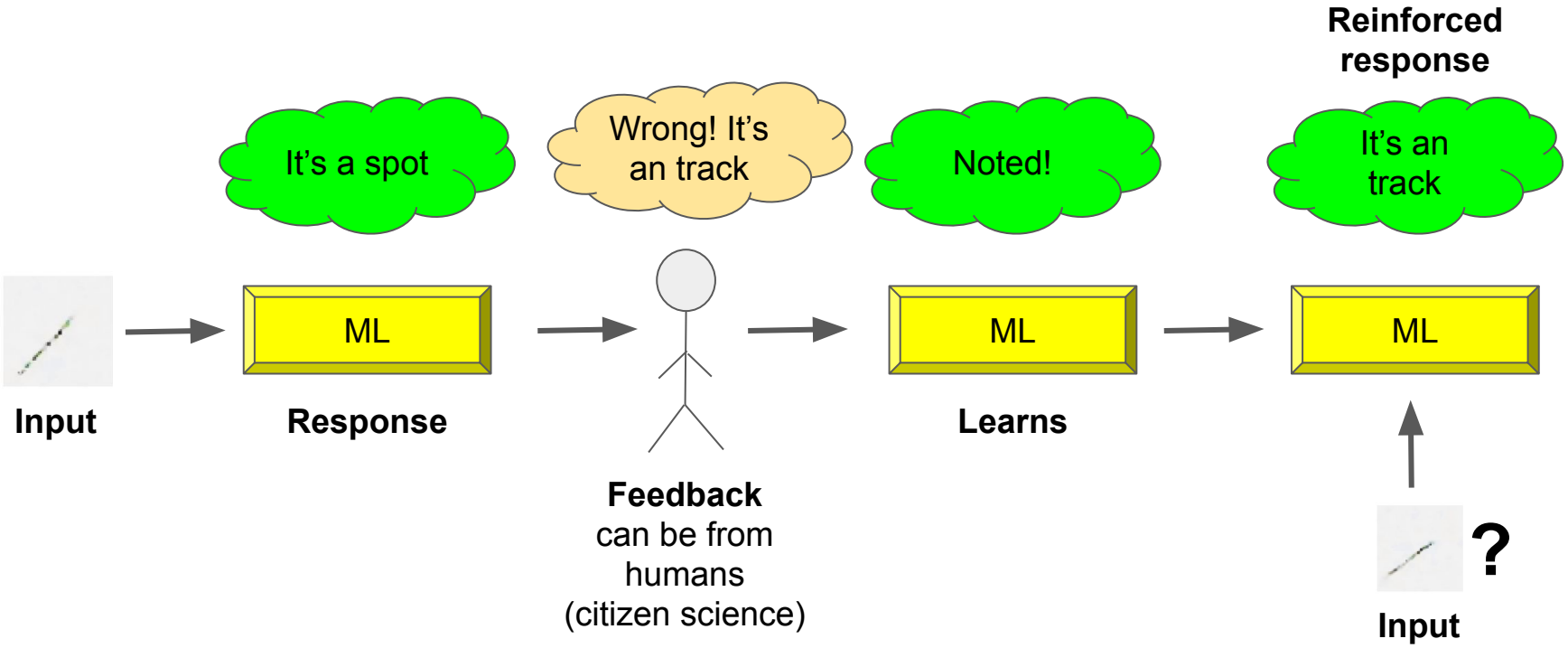
Supervised machine learning



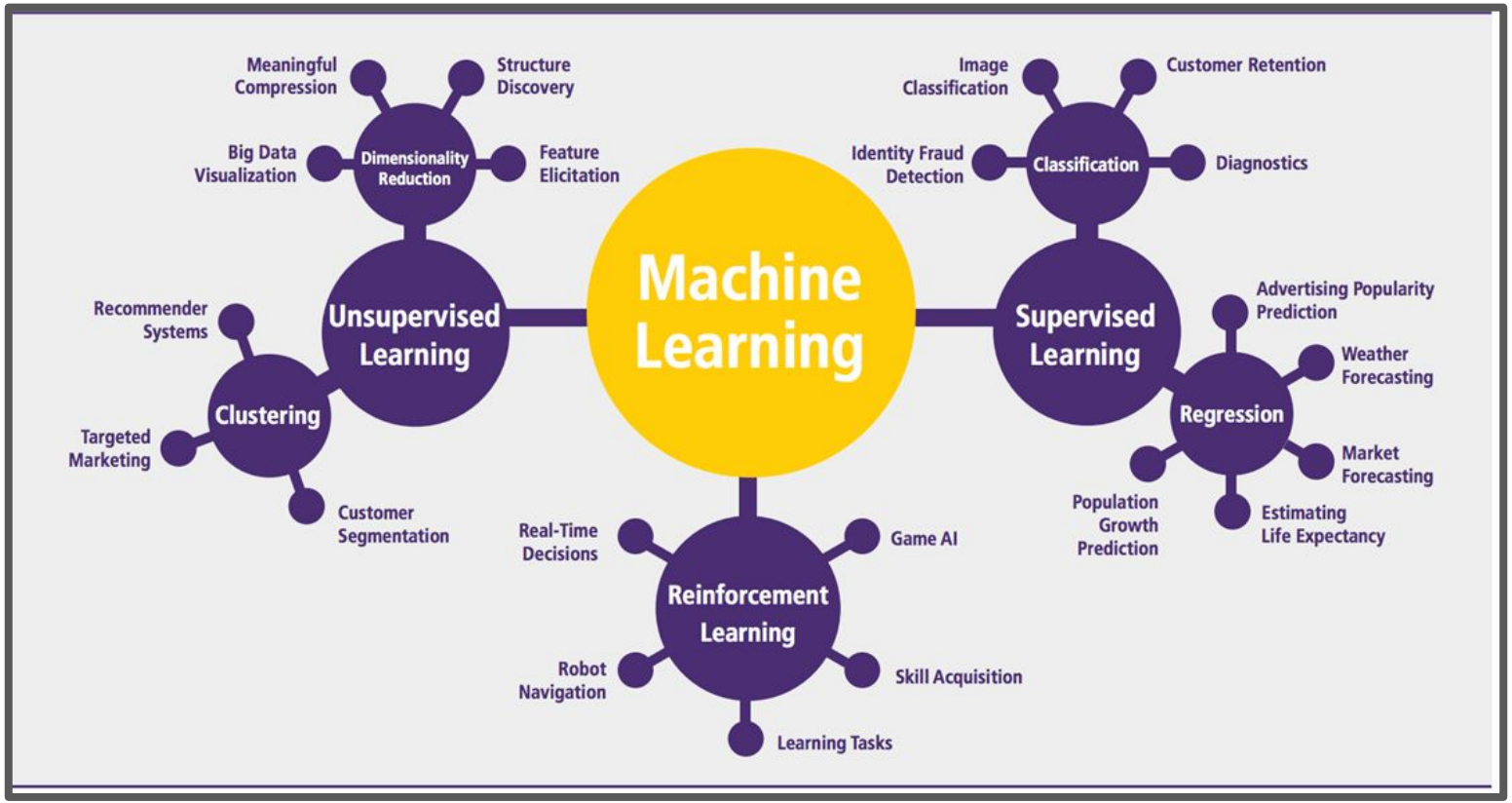
Unsupervised machine learning



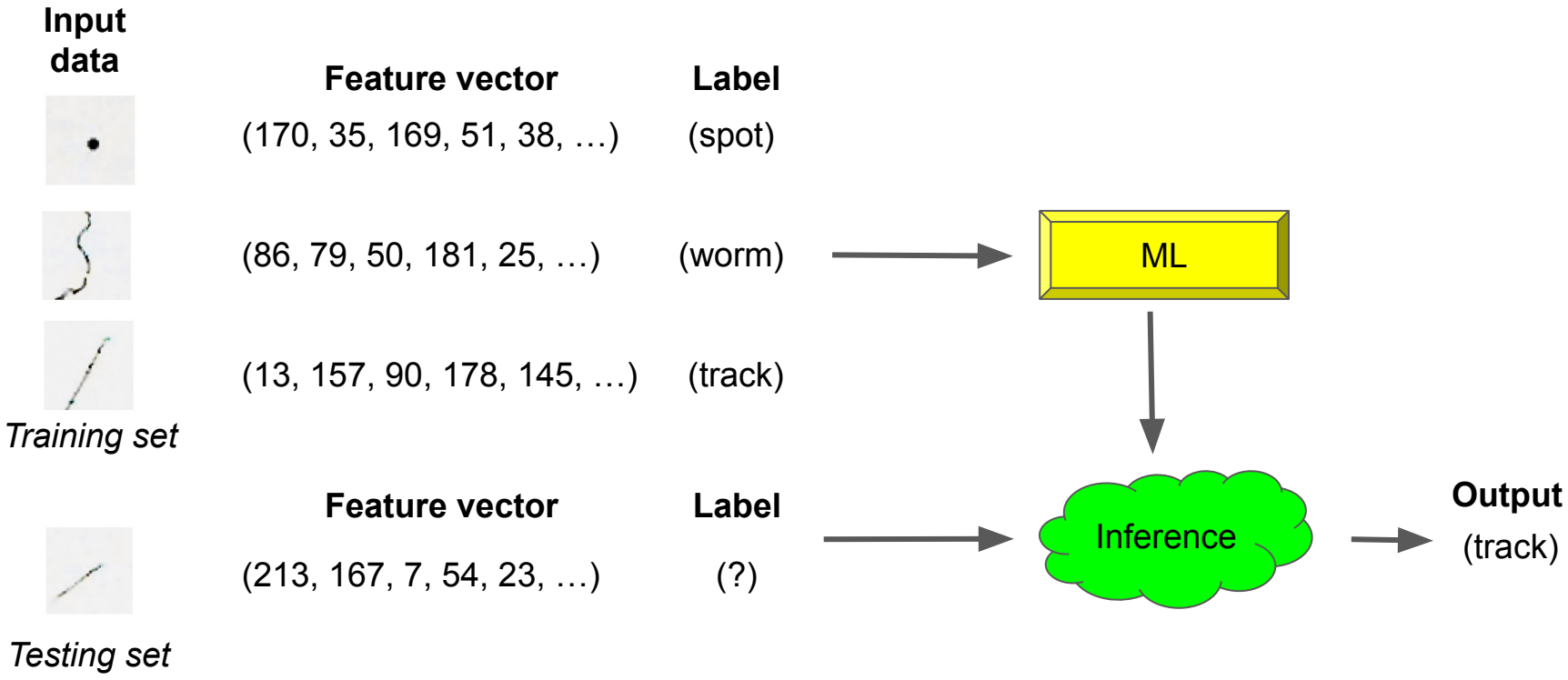
Reinforcement learning



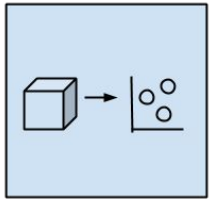
Where are machine learning used?



How it works

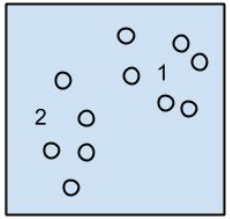

Input data
Feature vector
Label
(170, 35, 169, 51, 38, …)
(spot)
(86, 79, 50, 181, 25, …)
(worm)
(13, 157, 90, 178, 145, …)
(track)
Training set
ML
Feature vector
Label
(213, 167, 7, 54, 23, …)
(?)
Inference
Output
(track)
Testing set

Machine learning algorithms



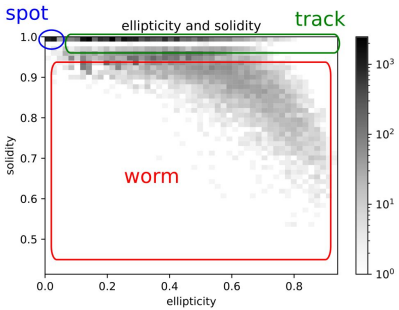
Dimensional Reduction Algorithms

- Principal Component Analysis (PCA)
- Principal Component Regression (PCR)
- Partial Least Squares Regression (PLSR)
- Sammon Mapping
- Multidimensional Scaling (MDS)
- Projection Pursuit
- Linear Discriminant Analysis (LDA)
- Mixture Discriminant Analysis (MDA)
- Quadratic Discriminant Analysis (QDA)
- Flexible Discriminant Analysis (FDA)

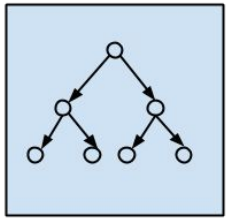


Clustering Algorithms

- k-Means
- k-Medians
- Expectation Maximisation (EM)
- Hierarchical Clustering

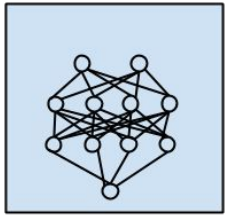


Simple classification by ellipticity and solidity to: spot-like, worm-like and track-like cosmic-rays



Decision Tree Algorithms

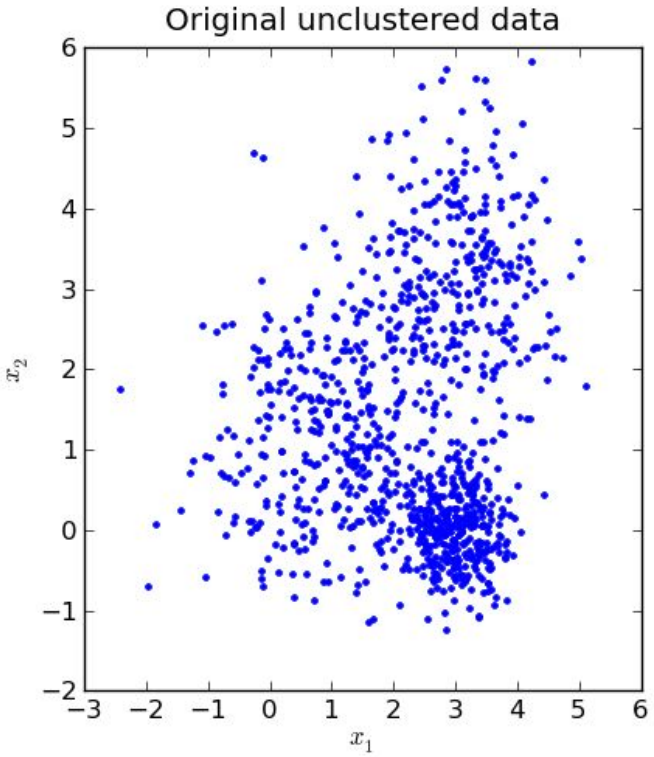
- Classification and Regression Tree (CART)
- Iterative Dichotomiser 3 (ID3)
- C4.5 and C5.0 (different versions of a powerful approach)
- Chi-squared Automatic Interaction Detection (CHAID)
- Decision Stump
- M5
- Conditional Decision Trees

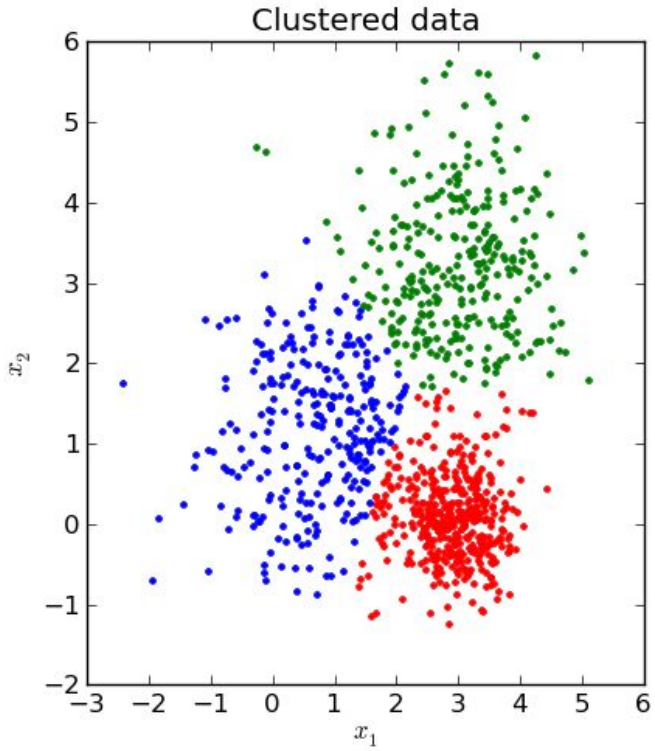


Deep Learning Algorithms

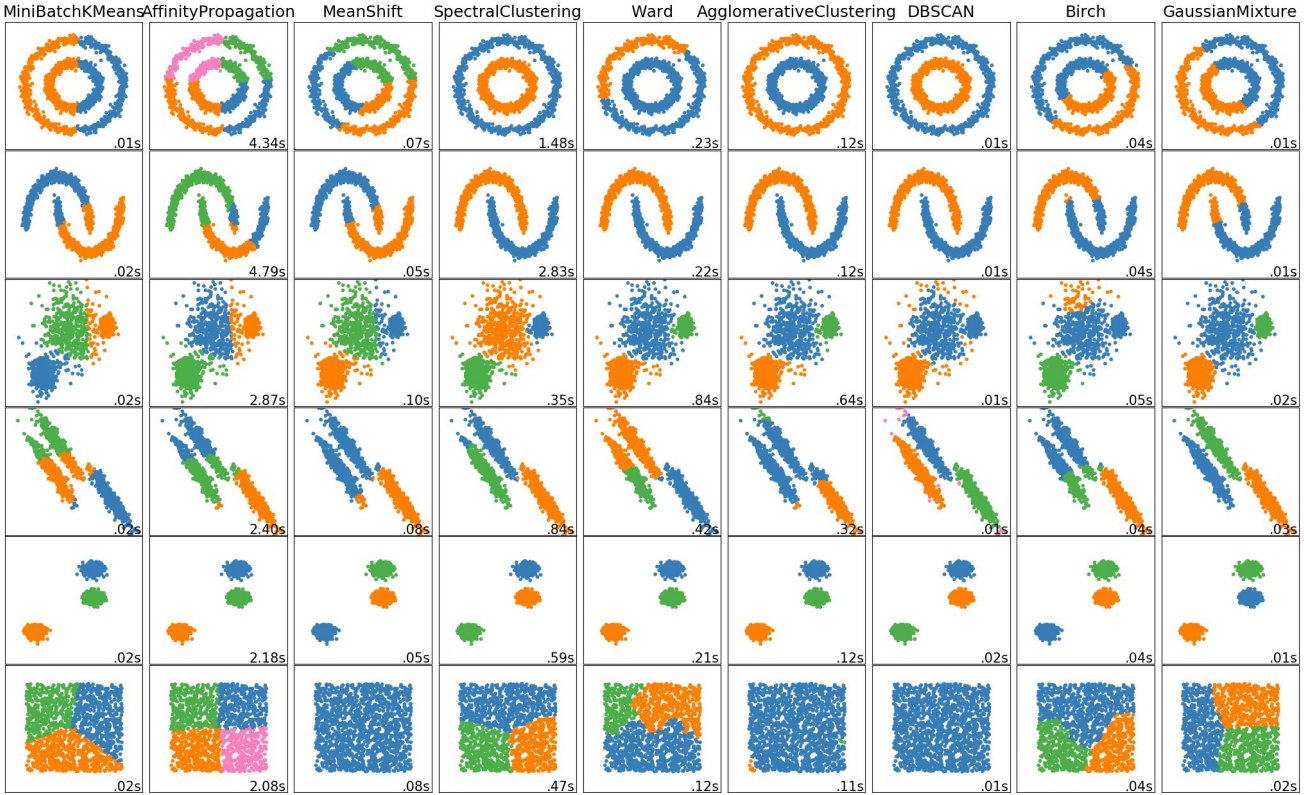
- Convolutional Neural Network (CNN)
- Recurrent Neural Networks (RNNs)
- Long Short-Term Memory Networks (LSTMs)
- Stacked Auto-Encoders
- Deep Boltzmann Machine (DBM)
- Deep Belief Networks (DBN)

K-means clustering

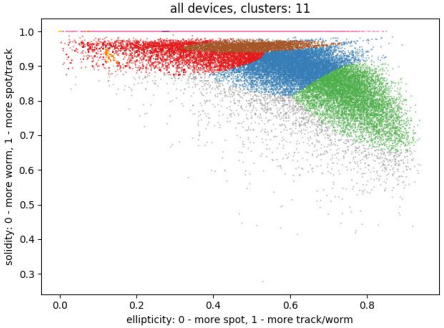




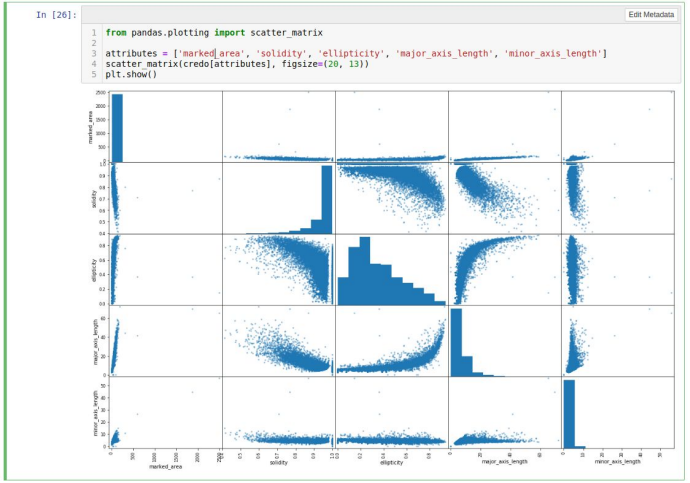
Various clustering algorithms



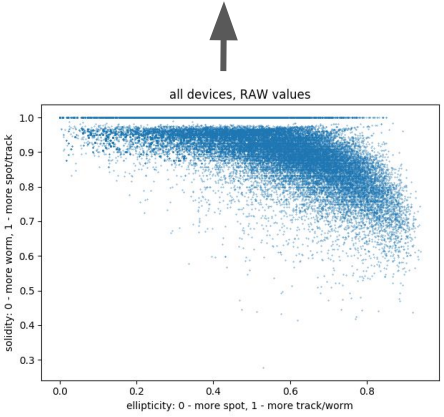
K-means in action

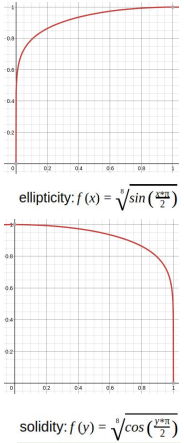


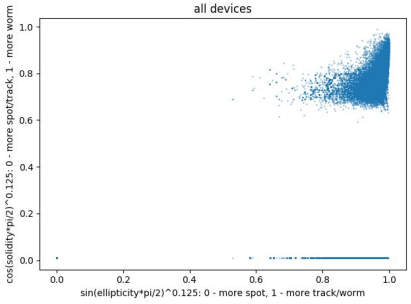
1. Select only significant features from vector

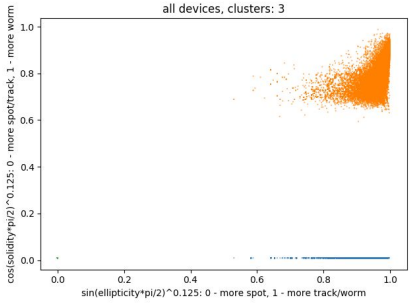


2. Reinforcement significant features by preprocessing

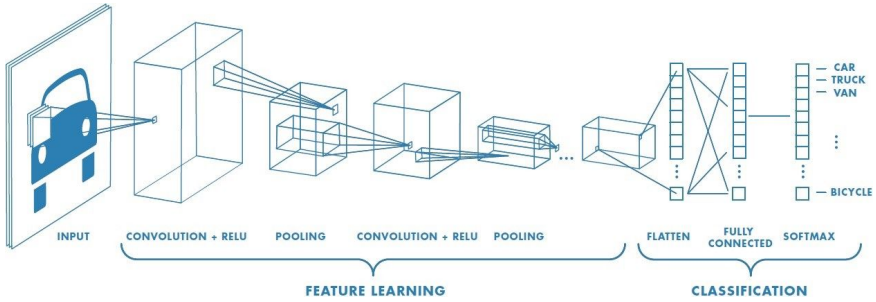


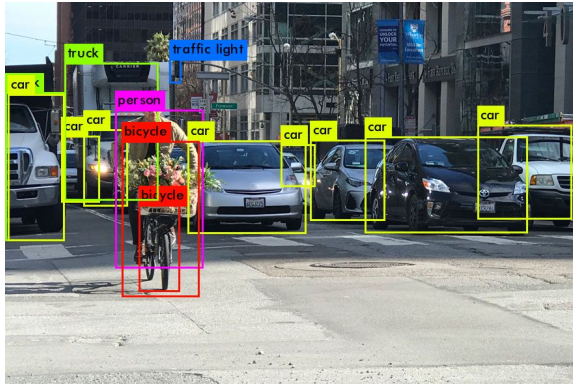




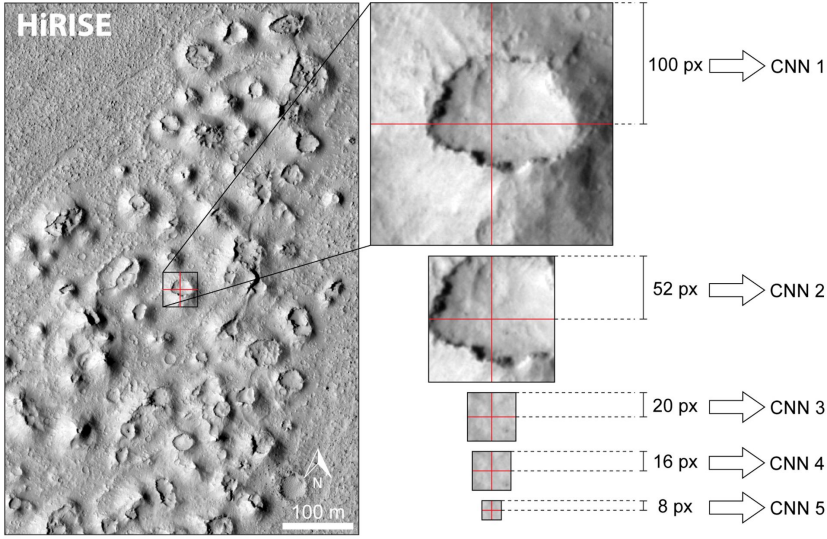


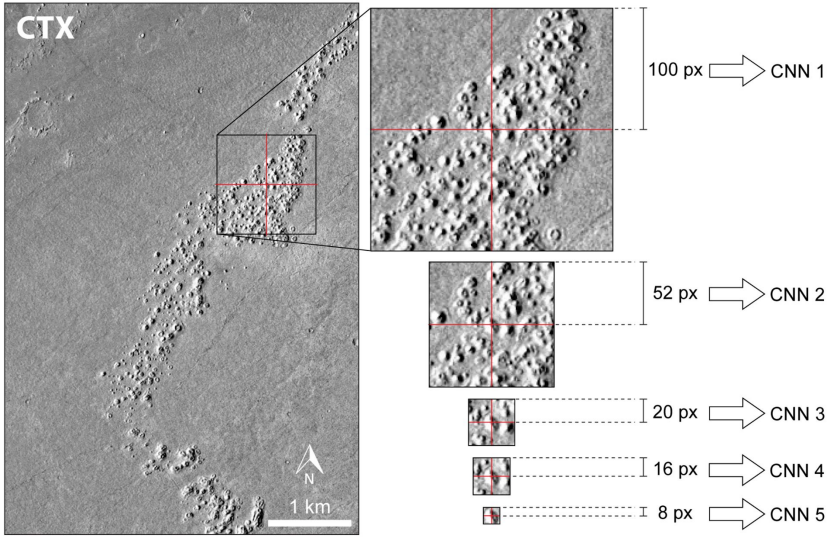
Convolution Neural Networks





DECO project uses CNN method to classification their hits:
https://www.sciencedirect.com/science/article/pii/S0927650518300859
Currently, i'm working on adopt it for CREDO database





Supervised methods needs big training set (~50K)

Training set must be prepared by manual. Training set have input data and correct output data. There are various method to get it:

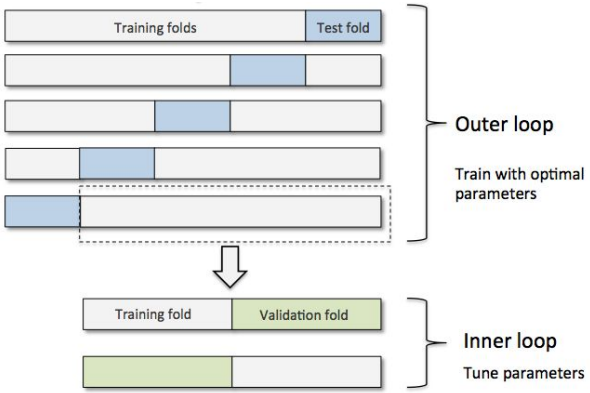
- people form citizen science (like Zoo Universe or Google recaptcha),
- generate from smaller set, in DECO some modification on images was used:
  - translation (random shift left/right and up/down from 0 to 8 px),
  - rotation (random rotation from 0° to 360°)
  - reflection (random horizontal and vertical reflections with probability 50%)
  - rescale (random rescale from 90% to 110% of original image size)

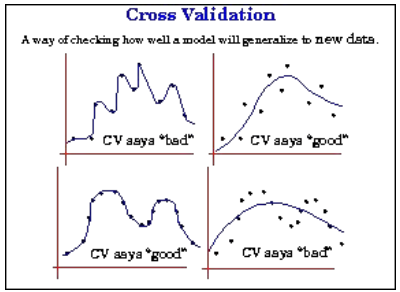
The output contains discrete set of labels with fuzzy 0-1 value

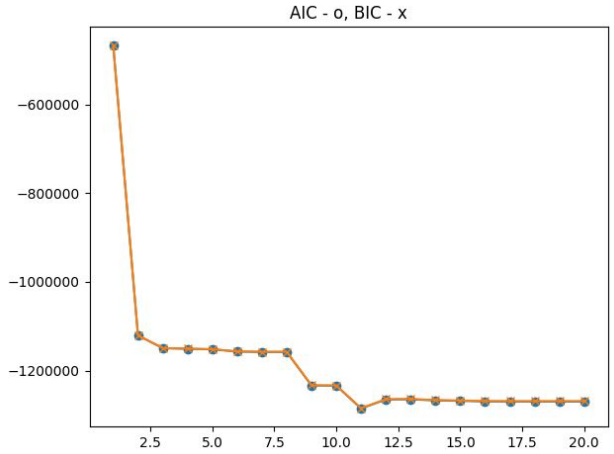
Measurement quality of ML method

What good is our ML method?

- AIC and BIC for clustering
- cross validation for training-based methods







Not only image analysis

Data for ML is the feature vector. Feature vector may be contains:

- everything from metadata (device model, timestamp, GPS etc.)
- additional data from another sources, for example: solar activity, meteo

ML may be:

- working with low quality data,
- found new patterns,

Disadvantages and threats

- bad teacher → bad method result,
  - ML method needs good data preparing, good pre processing (reinforcement significant features)
- completely random input data → random output

You can participate in data analysis

1. You must have CREDO Detector account
2. You must contact with us to give download privileges
3. You can download whole database by CREDO API tools in Python: https://github.com/credo-science/credo-api-tools
4. Server limited download data per user per day, please patient
5. CREDO API tools save data in JSON
6. By our tools you can extract hits to PNG files and other features to CSV file https://github.com/dzwiedziu-nkg/credo-analysis
7. These tools including some features extraction from basic image analysis

Our developer environment for ML

- Python programming language
- Jupyter notebooks
- scikit-learn library (everything free)
- tensorflow library

Recommended hardware:

- 1 TB HDD for data
- 16 GB RAM
- mid-end CPU
- tensorflow can use CUDA (GeForce gaming video card)

Summary

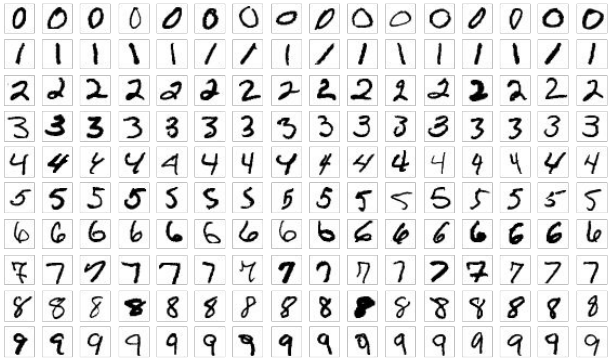
Machine learning help to analisis CREDO data

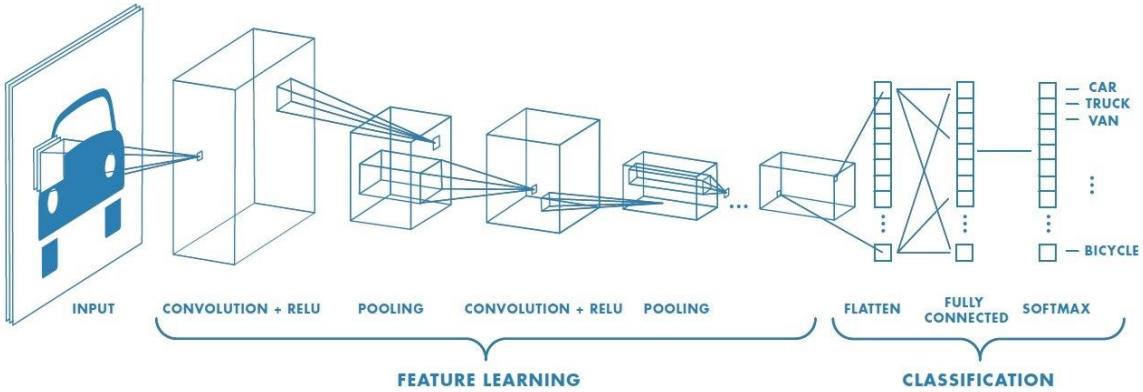
Supervised methods can help to found patterns what we know

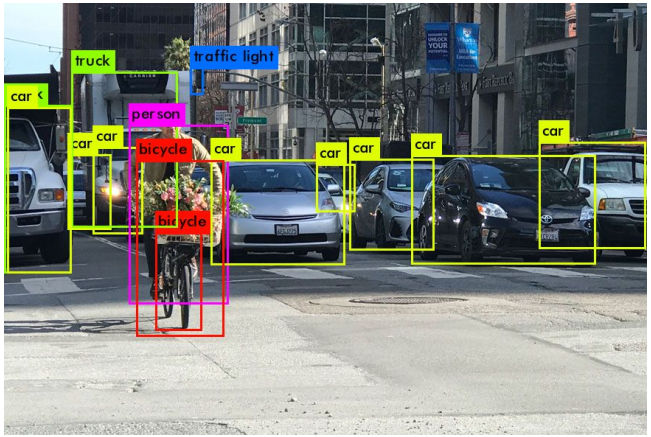
Unsupervised methods can help to found new patterns

We need the citizen science for prepare training data set and as feedback for reinforcement methods

Convolution Neural Networks







DECO project uses CNN method to classification their hits:
https://www.sciencedirect.com/science/article/pii/S0927650518300859