International Workshop on Machine Learning for Space Weather: Fundamentals, Tools and Future Prospects

7-11 November 2022
This is a hybrid meeting
Buenos Aires, Argentina

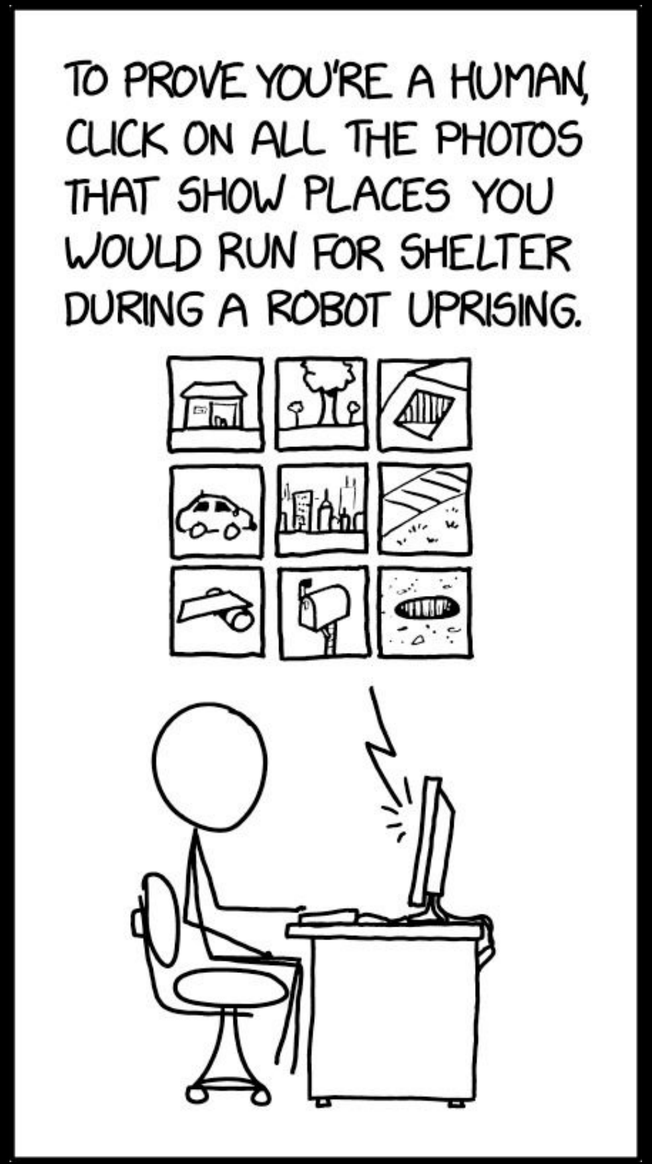
Further information:
https://indico.ictp.it/event/9840/
smr3750@ictp.it
+39-040-2240284
Elizabeth Brancaccio

Dra María Graciela Molina
FACET-UNT / CONICET
Tucumán Space Weather Center - TSWC

https://spaceweather.facet.unt.edu.ar/
IG -> @spaceweatherargentina

gmolina@herrera.unt.edu.ar

Machine Learning



Data-driven model

A DATA SCIENCE PIPELINE

Scientific Question

DATA

Data Exploration

Data Cleanning

Data wrangling/ munging

Modelling

Validation

Pre - Processing

Monitoring / Deployment

Knowledge / Decision

git DVC

General TSWC pipeline

Data acquisition

Local Sources

External Sources

RAW

FTP

debian

TSWC server

Gathers data such as: Rinex files, ionex files, time series (e.g. solar wind parameters), images (e.g. ionograms, spectrograms), structured files such as JSON from other DBs.

Server-external interaction

Data preprocessing

Cleanning and callibration

JSON

Handling raw data:

- Parser and cleaners (heterogeneous data).
- Control and fill of missing rows
- Compression methods module (e.g. ZIP, Hatanaka).
- Callibration data (e.g. TEC).
- Filtering and generating basic statistics
- Data is re-formated to JSON

Flask

API

Analysis

- Statistical and ML analysis, e.g.
  - Global TEC forecasting: NNARX, LSTM, GRU.
  - Single stationf0F2 forecasting: LSTM.

TensorFlow + Keras

Modelling and Processing

Persistent storage

Cleaned and calibrated data

mongoDB

- Data structure may change in the DB
- Plane text files (collections).
- Allows no structured or semi-structured data

Visualization

HIGHCHARTS

WebApp

django

wagtail

- Main visualization tools using highcharts

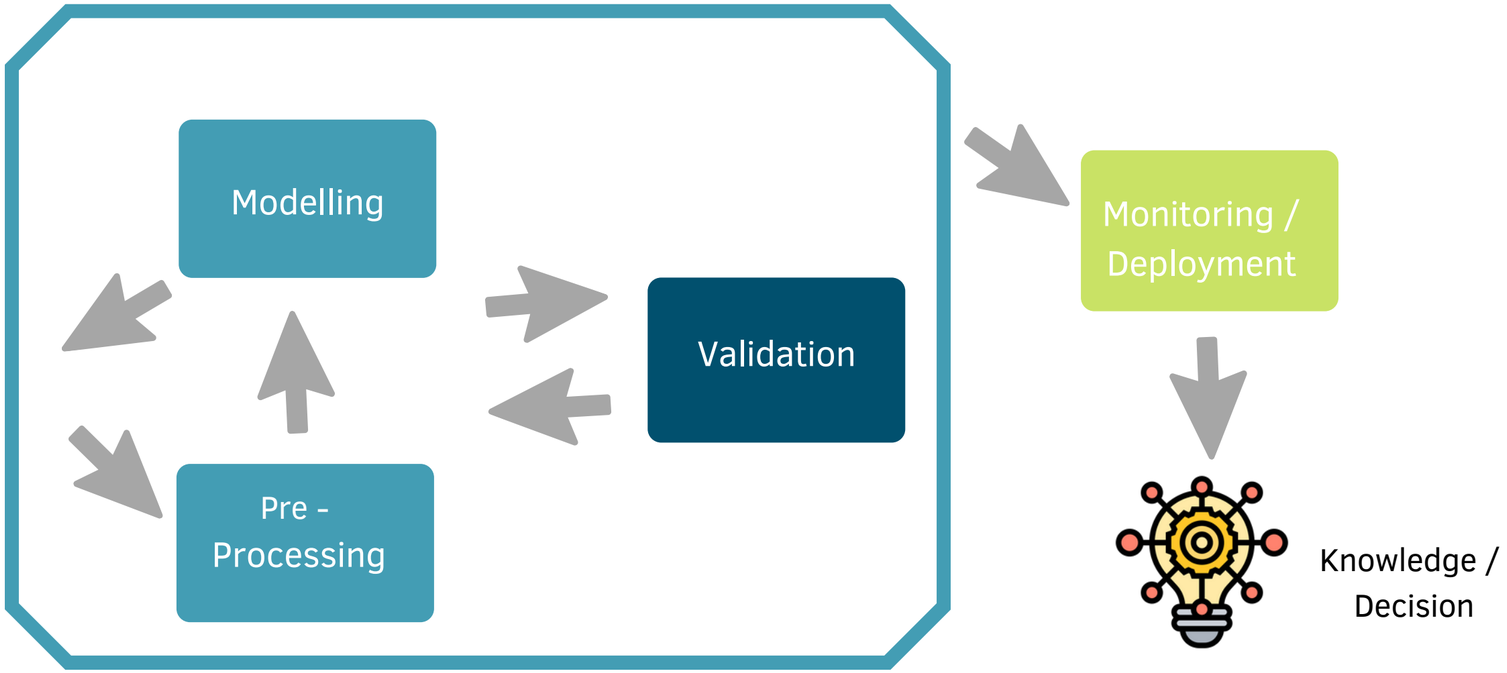
https://spaceweather.facet.unt.edu.ar/

Server-internal process

Namour et al
URSI-GASS, 2021

TSWC, 2022

ML pipeline



Monitoring /
Deployment

Knowledge /
Decision

What /how to ask?

How is the answer?

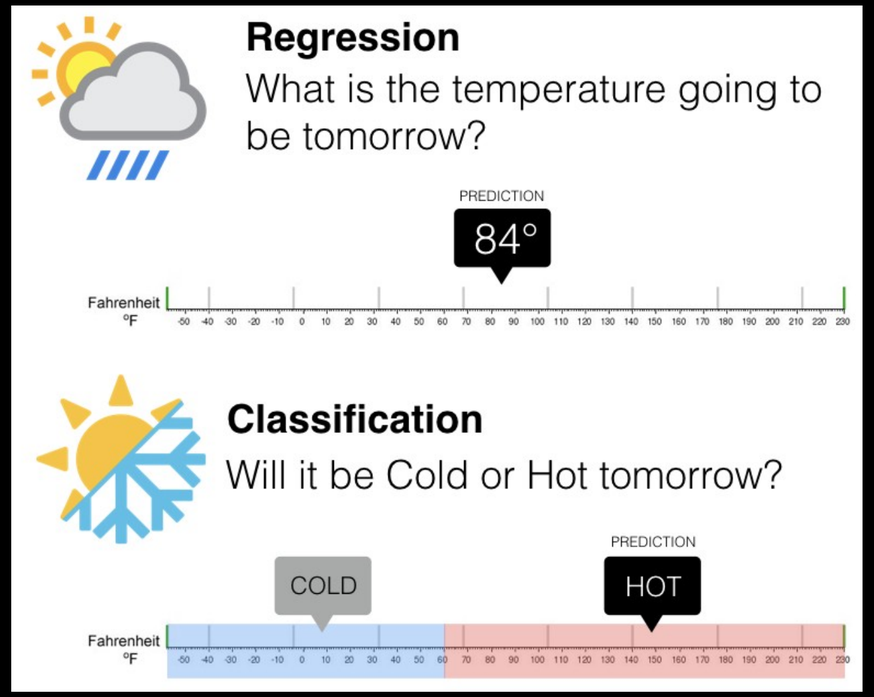
DATA (input)

Output

ML

MODEL
(program)

PROBLEMS (TYPES)

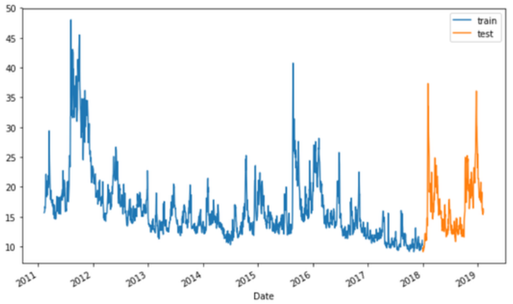


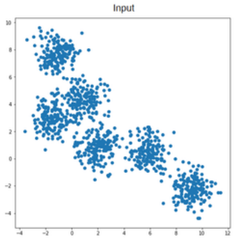
Regression: deals with predicting a continuous value

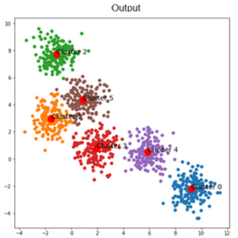
Classification: to predict output from a set of finite categorical values

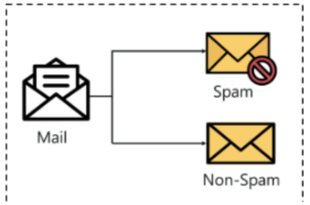
LEARNING (TYPES)

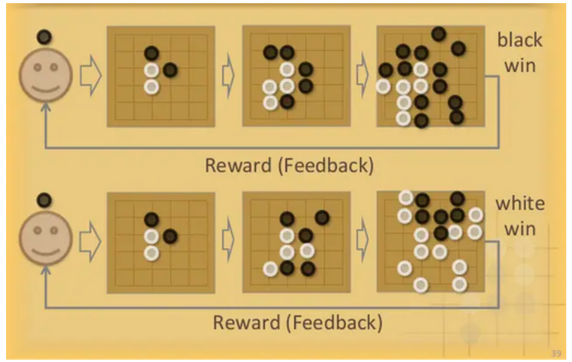
- Supervised: The target function is known. We have a labelled dataset
- Non Supervised: The dataset is not labelled
- Semi-supervised: The dataset is partially labelled.
- Reinforcement learning: ML system learns from the environment and it corrects itself by penalty or reward.

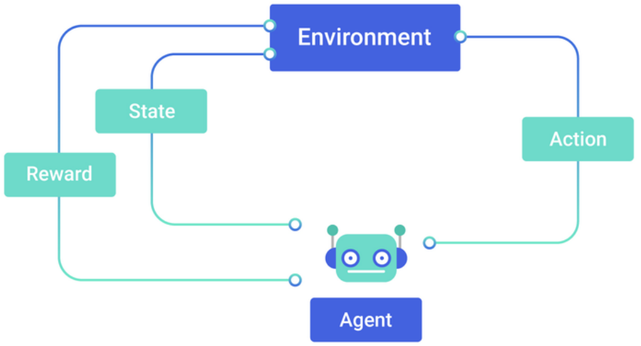




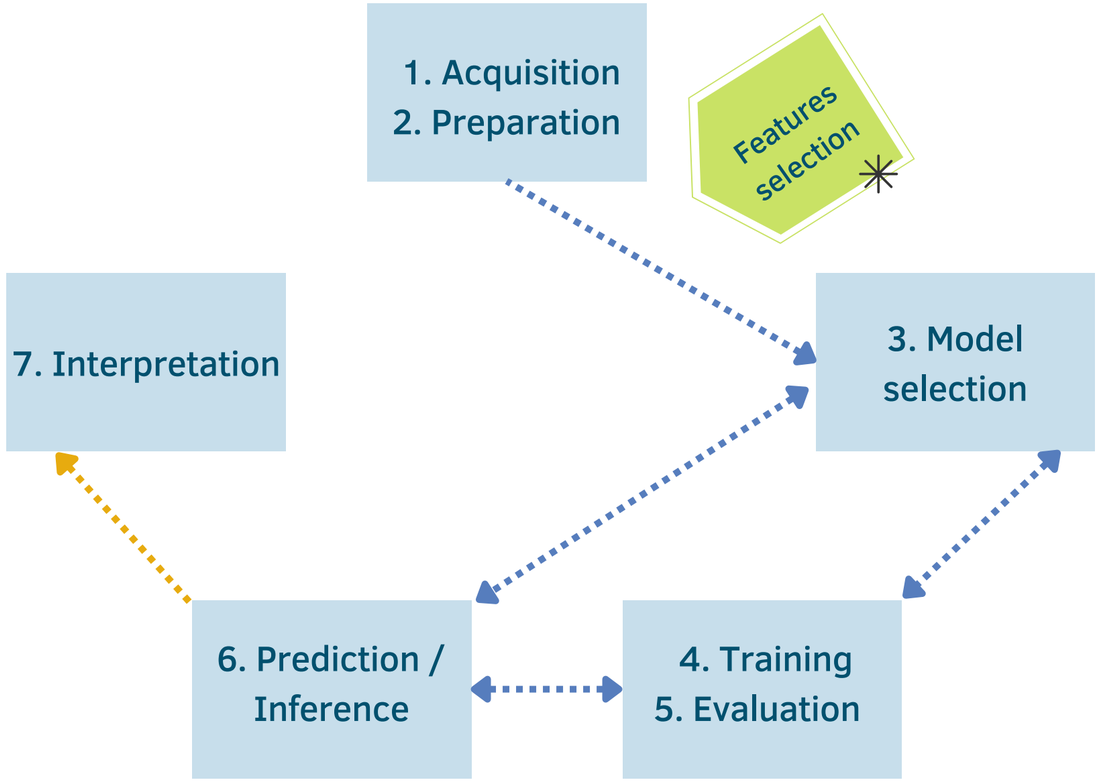








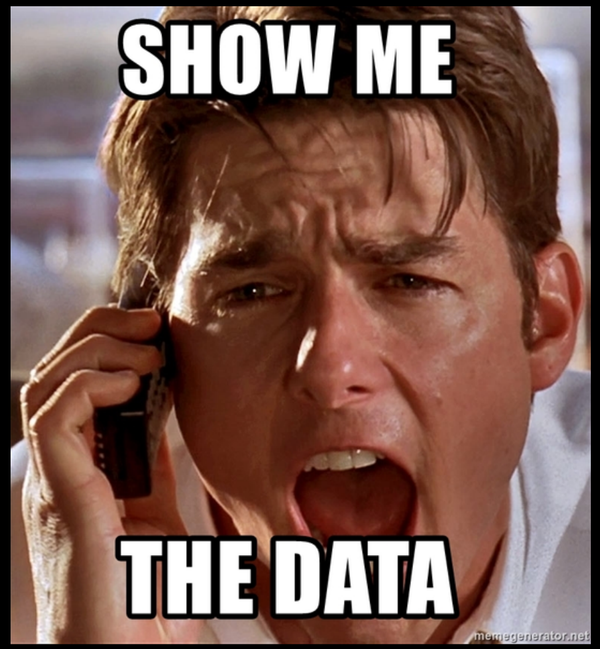
ML-based modeling



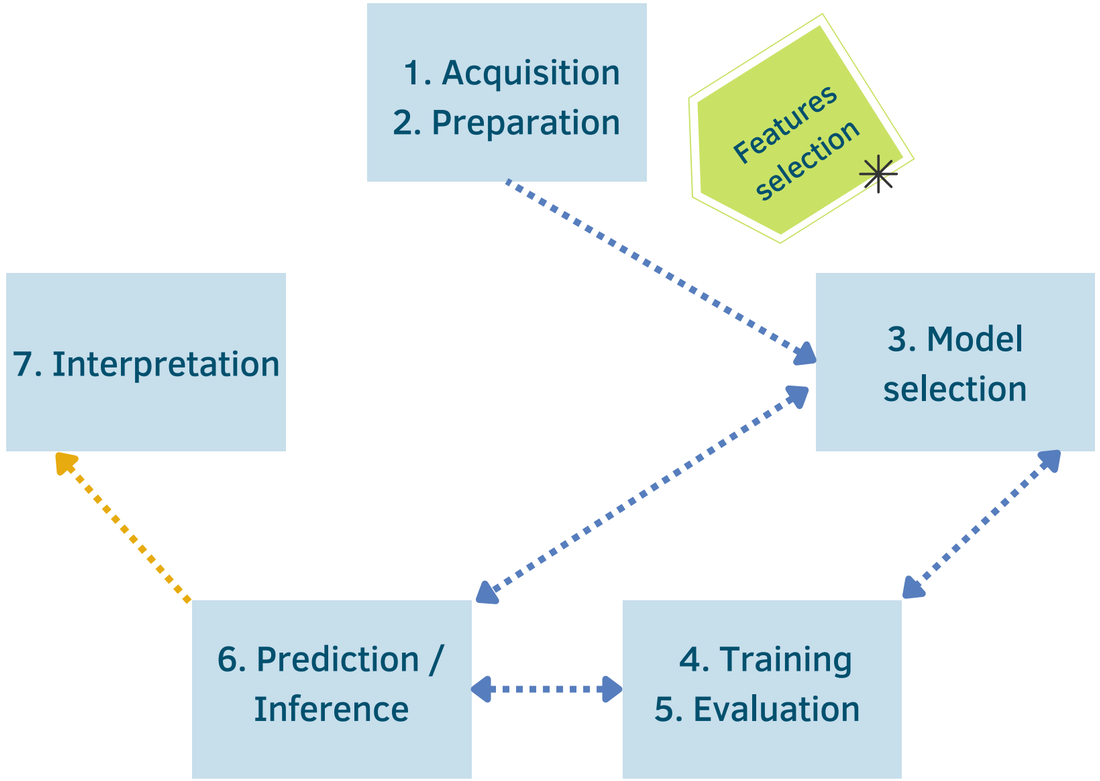
✳
- By hand = ML - automatically = DL
- Take care of dimensionality

- Acquisition + preparation
  - From ML public datasets:

https://www.kaggle.com/datasets
https://archive.ics.uci.edu/ml/index.php
https://github.com/awesomedata/awesome-public-datasets



ML-based modeling

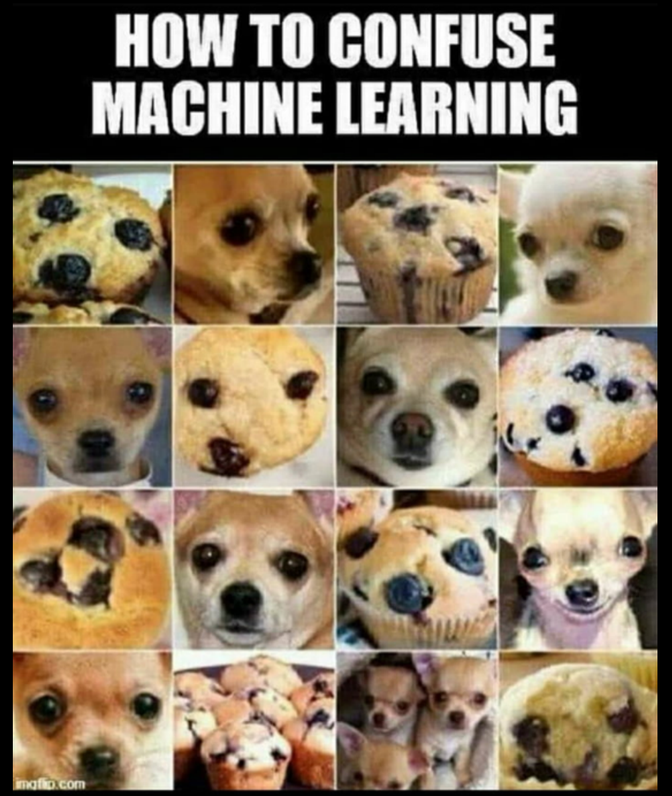


✳
- By hand = ML - automatically = DL
- Take care of dimensionality

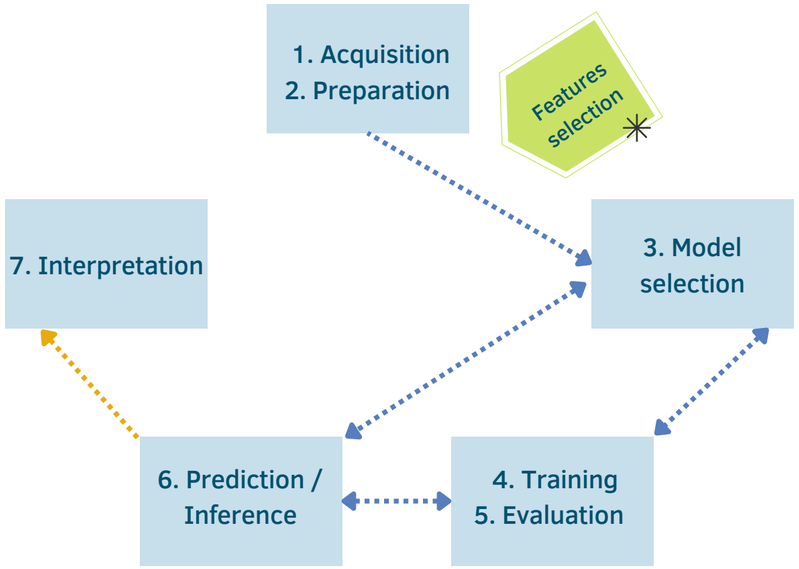
- Acquisition + preparation
  - Make our ML datasets:

- Acquisition from experiments or instruments or - simulations, etc
- Synthetic data: create datasets

Prepare the dataset: sampling, formatting, metadata, resolution, balance, dataset size, curation, standarization/normalization, integration, storing, etc



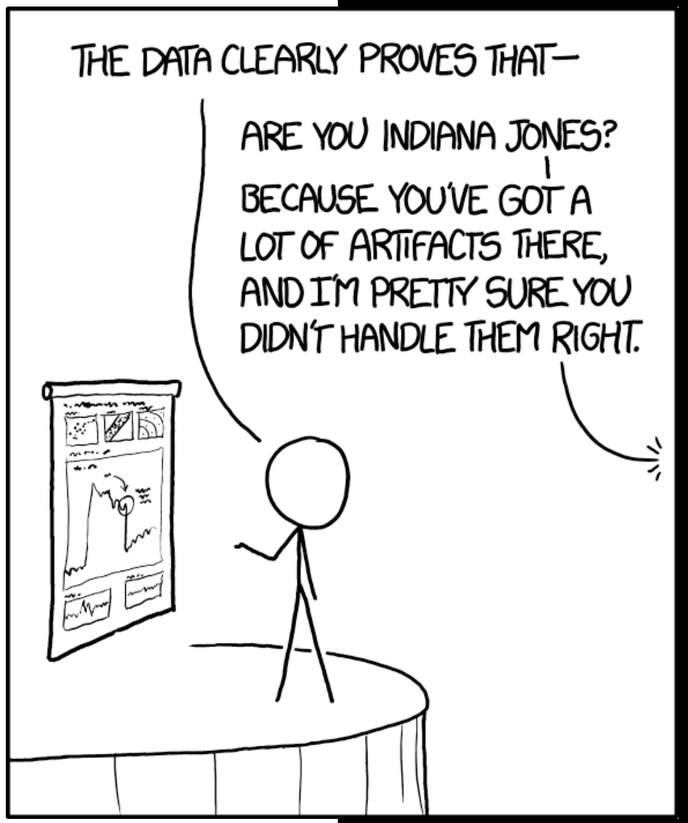
ML-based modeling



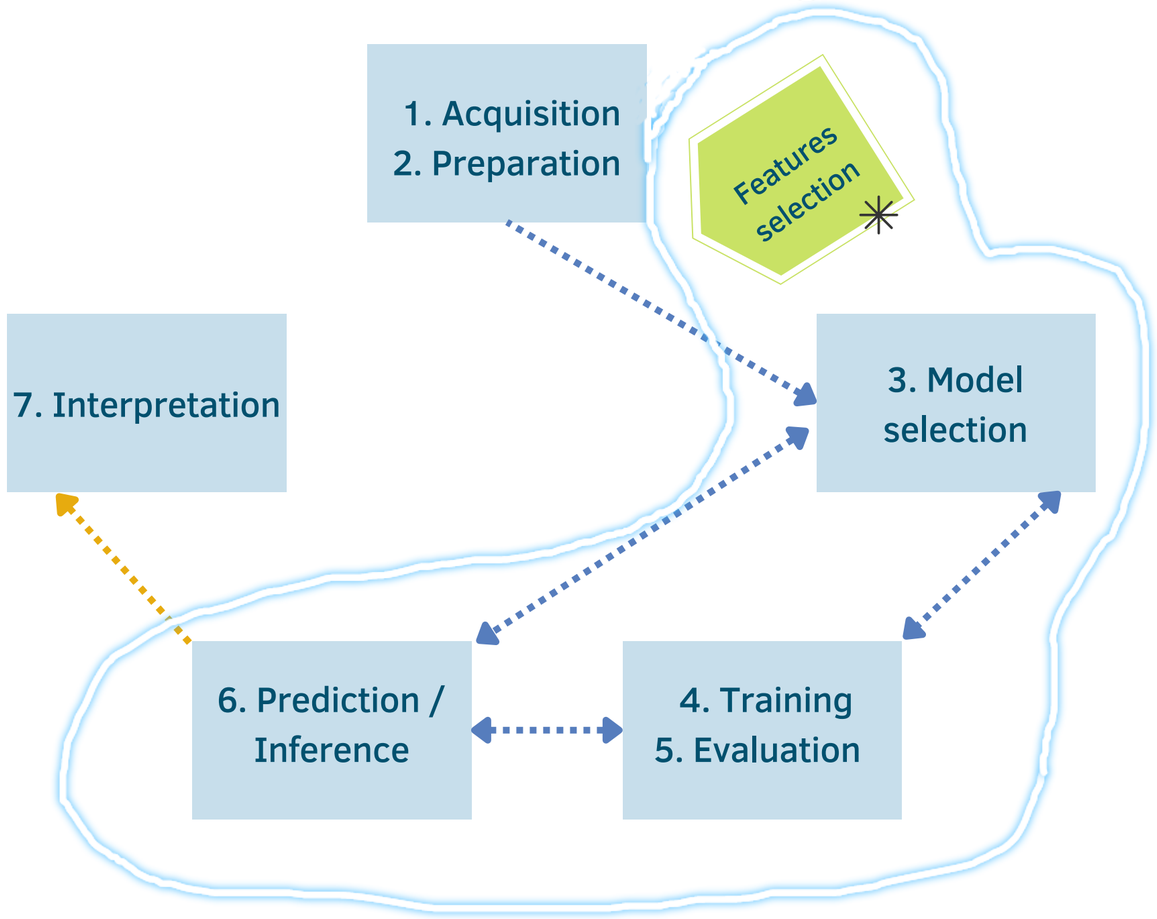
- Modeling

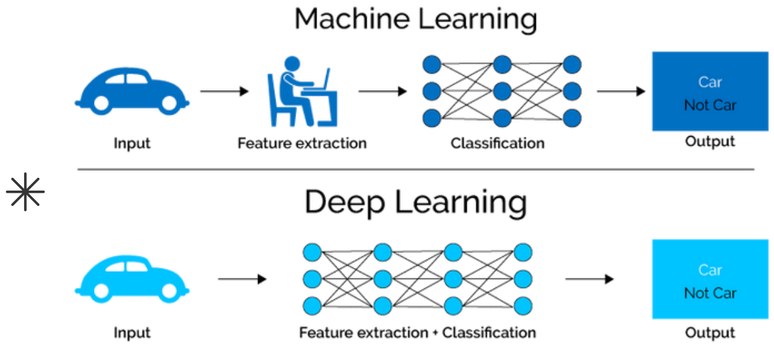
DATA PREPARATION FOR MODELING

- Basic exploration: basic statistics about the dataset: # samples, # nulls, pdfs, plot, etc
- Balance / unbalanced datasets (the problems, how to solve them)
- Missing values: to input or not to input data!
- Outliers!
- Data transformation = binning, log scales; normalization/standarization.
- Dimensionality reduction if necessary
- Data traceability!



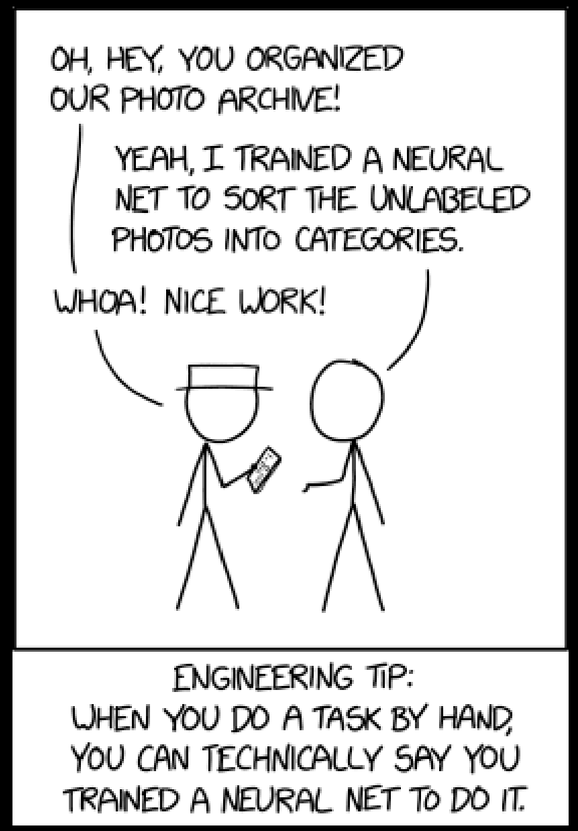
ML-based modeling





- Modeling

FEATURE ENGENEERING



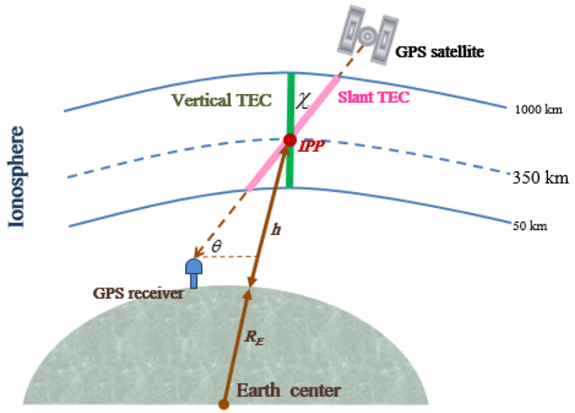
- Extract/choose relevant features from the dataset
- May include the creation of new features
- Requires experience and domain knowledge
- Feature engineering by hand: hard, slow, not robust, not scalable
- Explicit or/and implicit. (lagged features, statistics, data transformation)
- DL = automatic feature engineering

ML-based modeling

Example

- Let's forecast the ionospheric conditions! how hard could it be? Let's select the features!

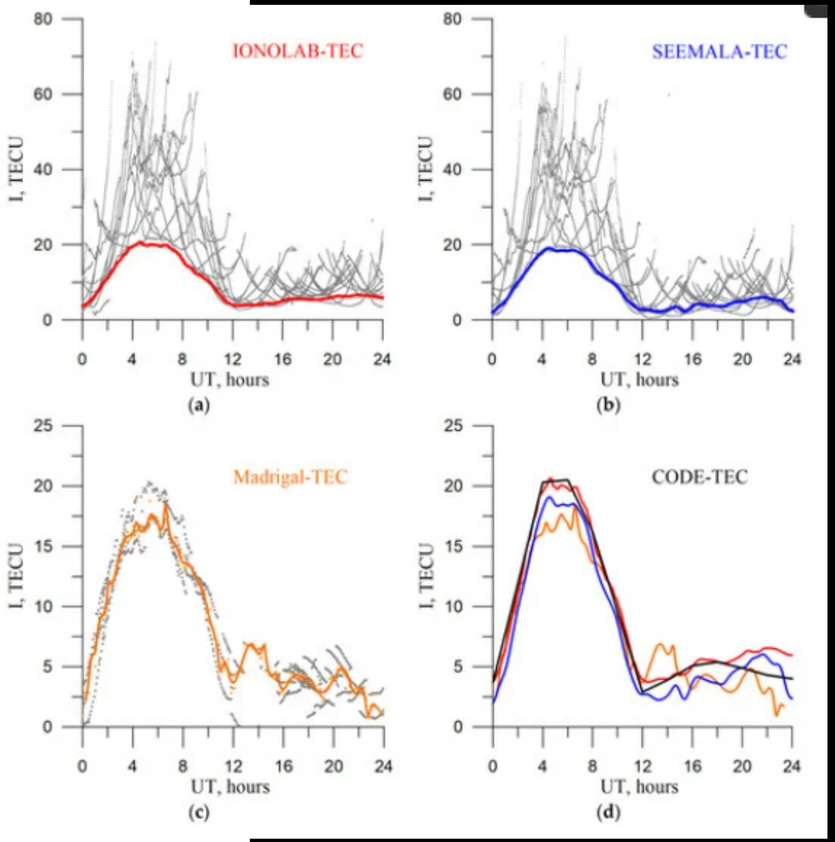
  Assumptions:
  - single station modeling
  - TEC derived from GNSS (using "some" calibration technique and "some" constellation)
  - It is a time series



- Which is/are the feature/s?

- Modeling

FEATURE ENGENEERING

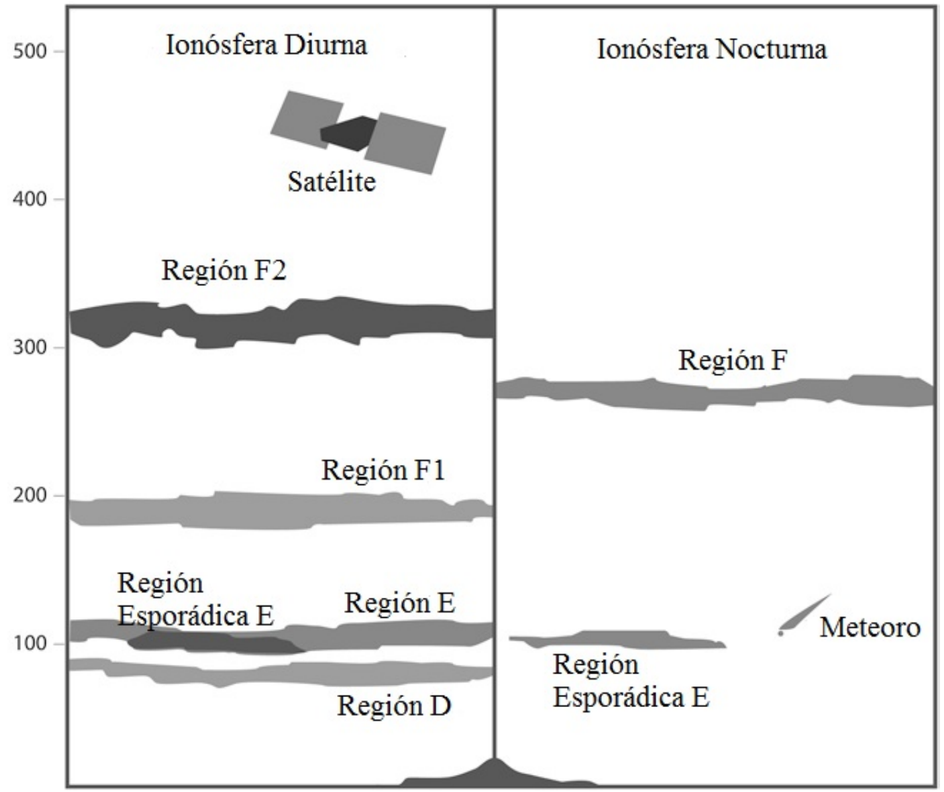


- Extract/choose relevant features from the dataset
- May include the creation of new features
- Requires experience and domain knowledge
- Feature engineering by hand: hard, slow, not robust, not scalable
- Explicit or/and implicit. (lagged features, statistics, data transformation)
- DL = automatic feature engineering

Parentesis (el medio bajo estudio)

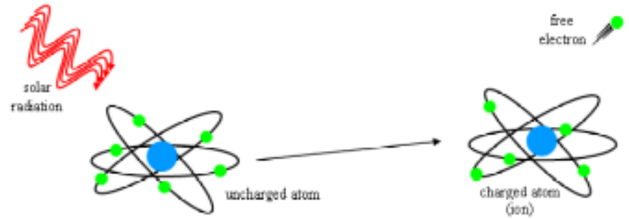
Ionosfera: porción de la alta atmósfera terrestre donde tanto iones como electrones se encuentran presentes en cantidades suficientes como para afectar las ondas de radio (IEEE Std 211-1997).

Se caracteriza por la alta desidad de electrones e iones libres producidos principalmente por foto-ionización de rayos UV y X que arrivan desde el sol (y minoritriamente en altas latitudes por ionización corpuscular)



- Molina, 2018

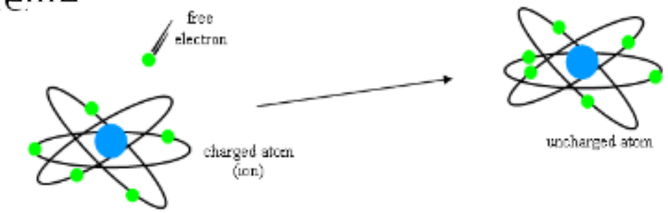
- Foto-ionización (producción de iones y electrones libres)



$$a + \frac{hc}{\lambda} \rightarrow a^{+} + e^{-}$$

h =Plank: const 6.62 × 10−34 Js, c = vel. Luz, λ = long de onda incidente

- Recombinación: fenómeno inverso. Los electrones libres se combinan con iones positivos para producir átomos neutros nuevamente



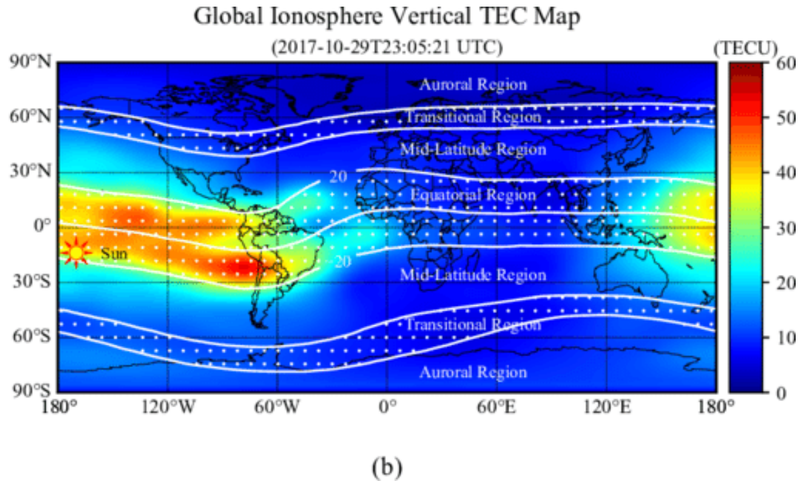
dN/dt = varición densidad electronica en fn del tpo (unidad de vol.), q = producción de electrones, l= pérdida de electrones, d= factor (pérdida por difusión, vientos neutros, drift electromag. vertical)

$$\frac{dN}{dt} = q - l + d$$

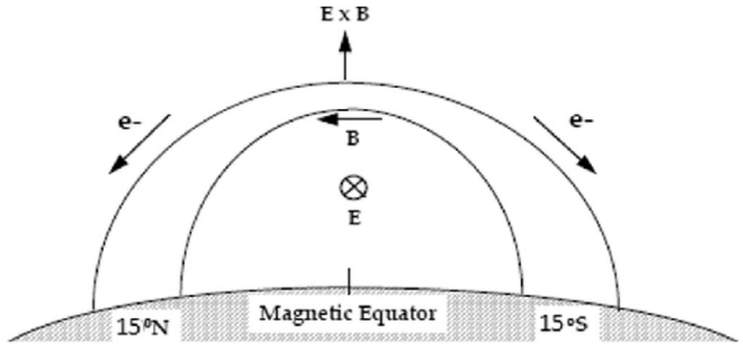
Parentesis (el medio bajo estudio)

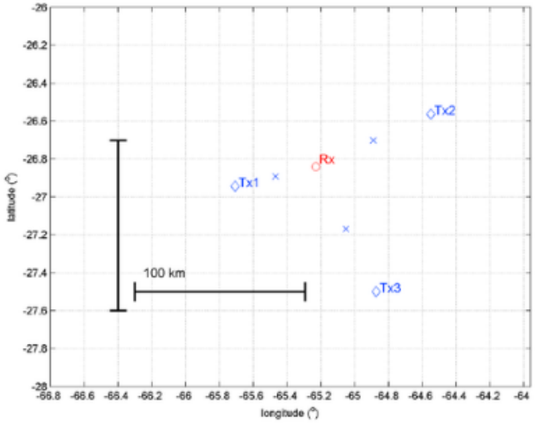
Variabilidad regular

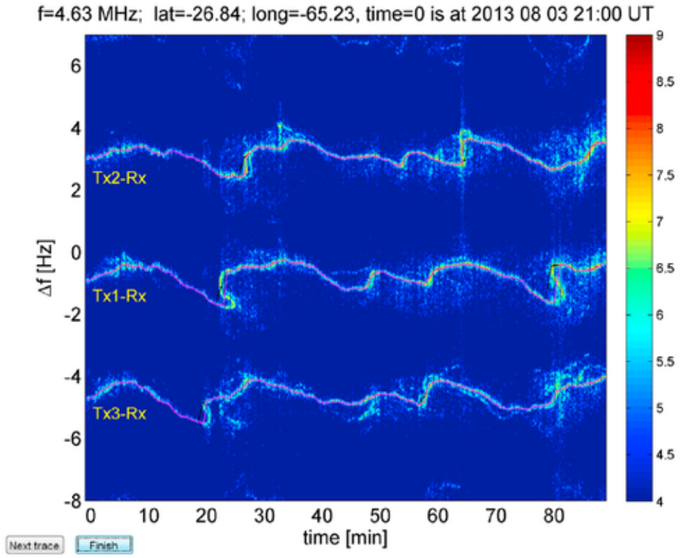
- diaria
- estacional
- geográfica/geomagnética
- ciclo solar



(b)







Irregularidades

Travelling ionospheric disturbances (TIDs)
Irregularidades de la región F expresadas como oscilaciones similares a ondas del contorno de la denidad electrónica que va descendiendo lentamente con el tiempo

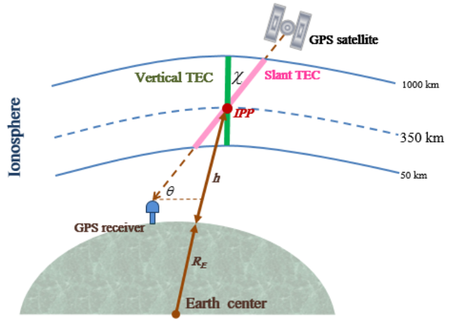
Se clasifican en : large-scale TIDs (LS-TIDs) y medium-scale TIDs (MS-TIDs).

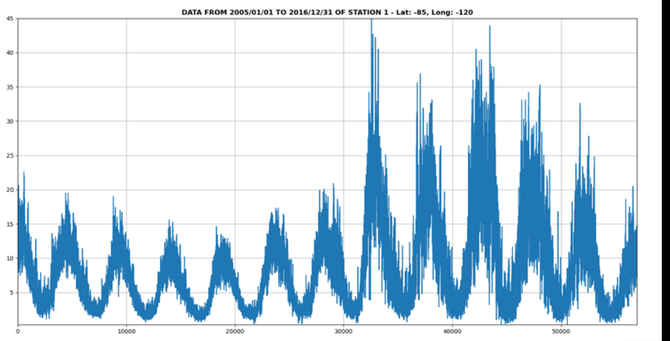
- LS-TIDs relacionadas con AGWs causadas por acoplamiento VS-M-I en zonas polares que se transportan a otras regiones (Space Weather)
- MS-TIDs: periodos de tiempo más cortos, se mueven más lento y generalmente estan relacionadas a fenómenos como vientos neutros y al terminador solar que generan AGWs a alturas ionosféricas

- TSWC-CAS

ML-based modeling

- Which is/are the feature/s?





- TEC -> time (sequence -order)
- what about shifted TEC series? e.g. 24 his shift? seasonal shift? other lags?
- Latitudinal characteristics (how to add?) - how important is this? How about irregularities? (scales!!!)
- TEC is a model! how much impact will have if we approximate TEC with different calibration techniques or the number of PRN/constellations? Which we should choose? is it relevant to add different time series (for different techniques or constellations?)

$$STEC = \int_s N_e ds \ \ (\text{TECU})$$

- and this is for a single station!

- Modeling

FEATURE ENGENEERING

- Extract/choose relevant features from the dataset
- May include the creation of new features
- Requires experience and domain knowledge
- Feature engineering by hand: hard, slow, not robust, not scalable
- Explicit or/and implicit. (lagged features, statistics, data transformation)
- DL = automatic feature engineering

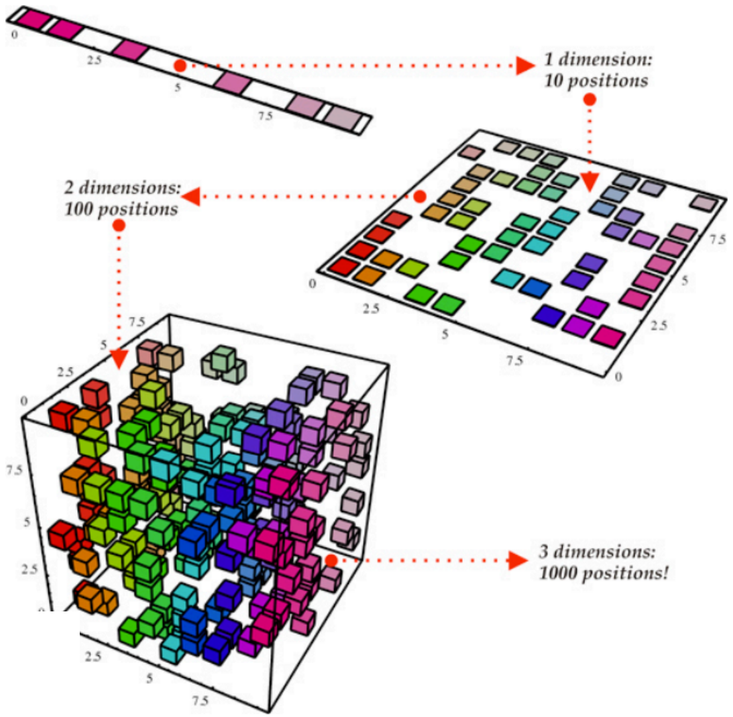
How may is too many?

ML-based modeling

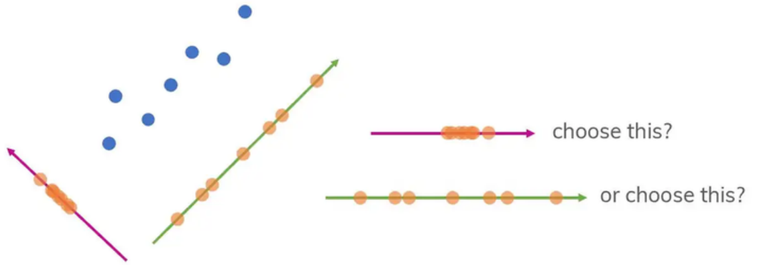
☑ FEATURES (Selection)

+ features + data + computing effort + sparce data

- What if one of the features is strongly related to another feature? how to know they are related?
- How many features (dimensions) do the ML needs to get a satisfactory model?



Distance concentration: many ML algorithms are based on the calculation of distances. The distance between 2 points tends to grow as the dimension grows.



nature › nature methods › this month › article

This Month | Published: 31 May 2018

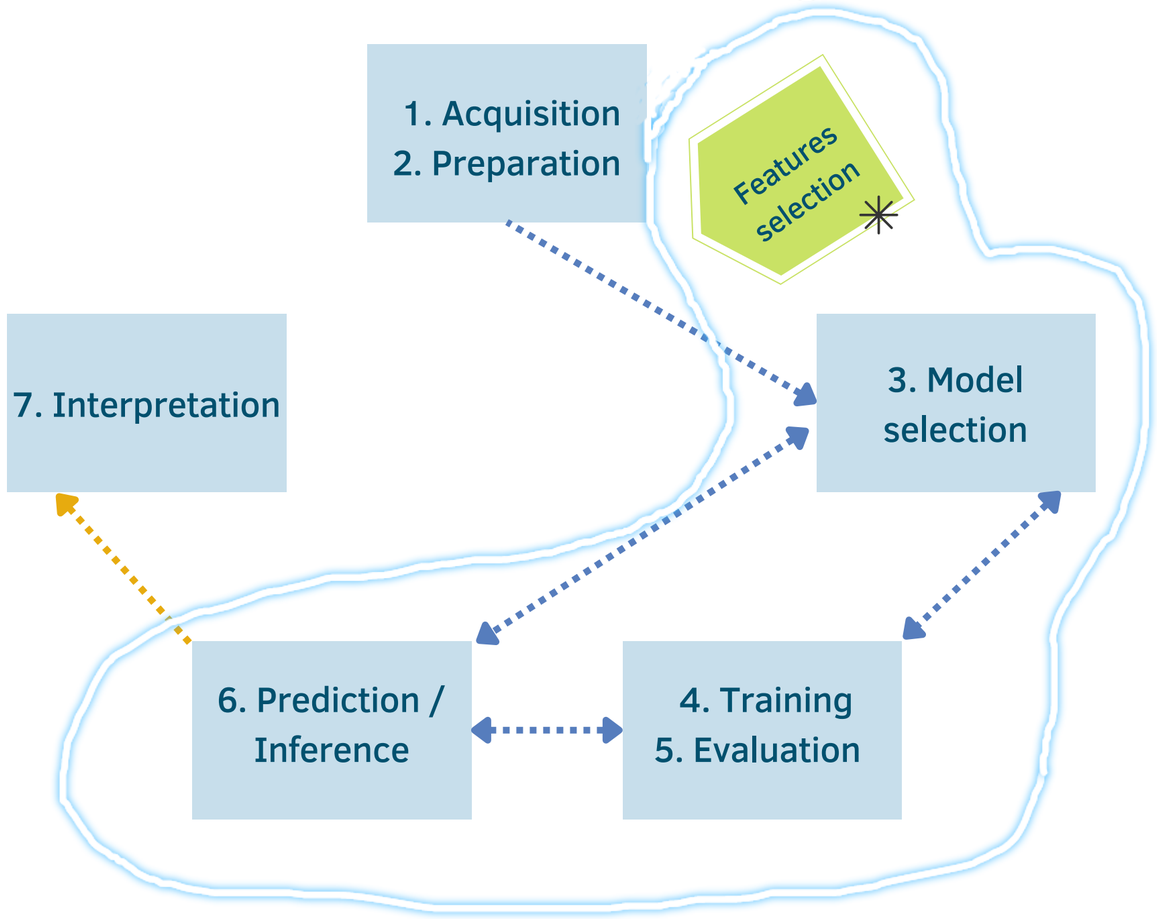
POINTS OF SIGNIFICANCE

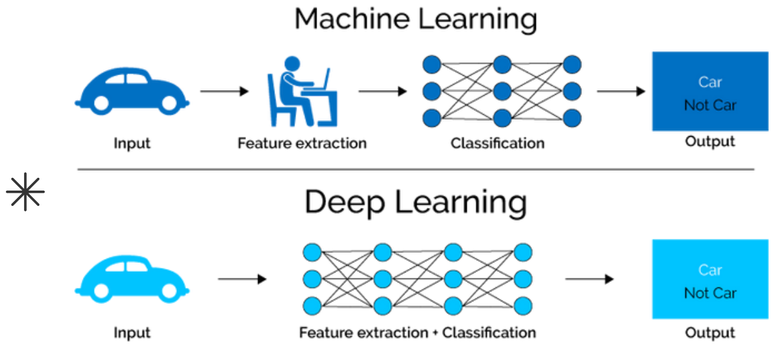
The curse(s) of dimensionality

Naomi Altman & Martin Krzywinski

https://www.nature.com/articles/s41592-018-0019-x

ML-based modeling

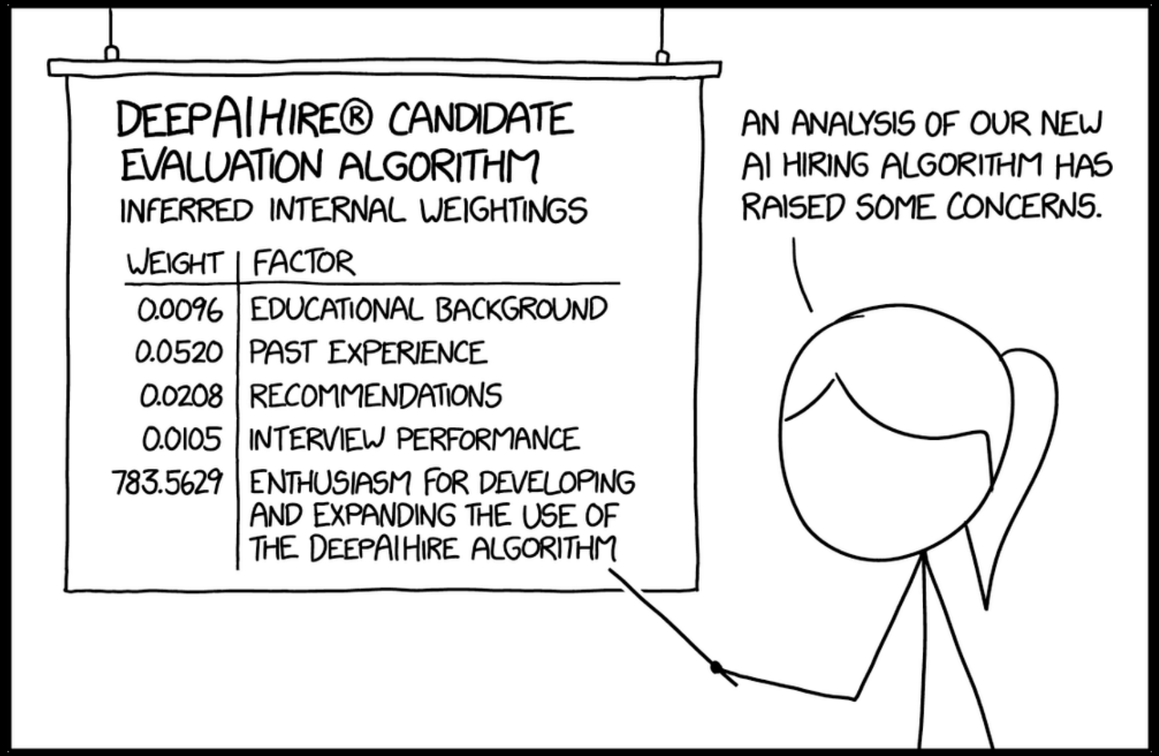




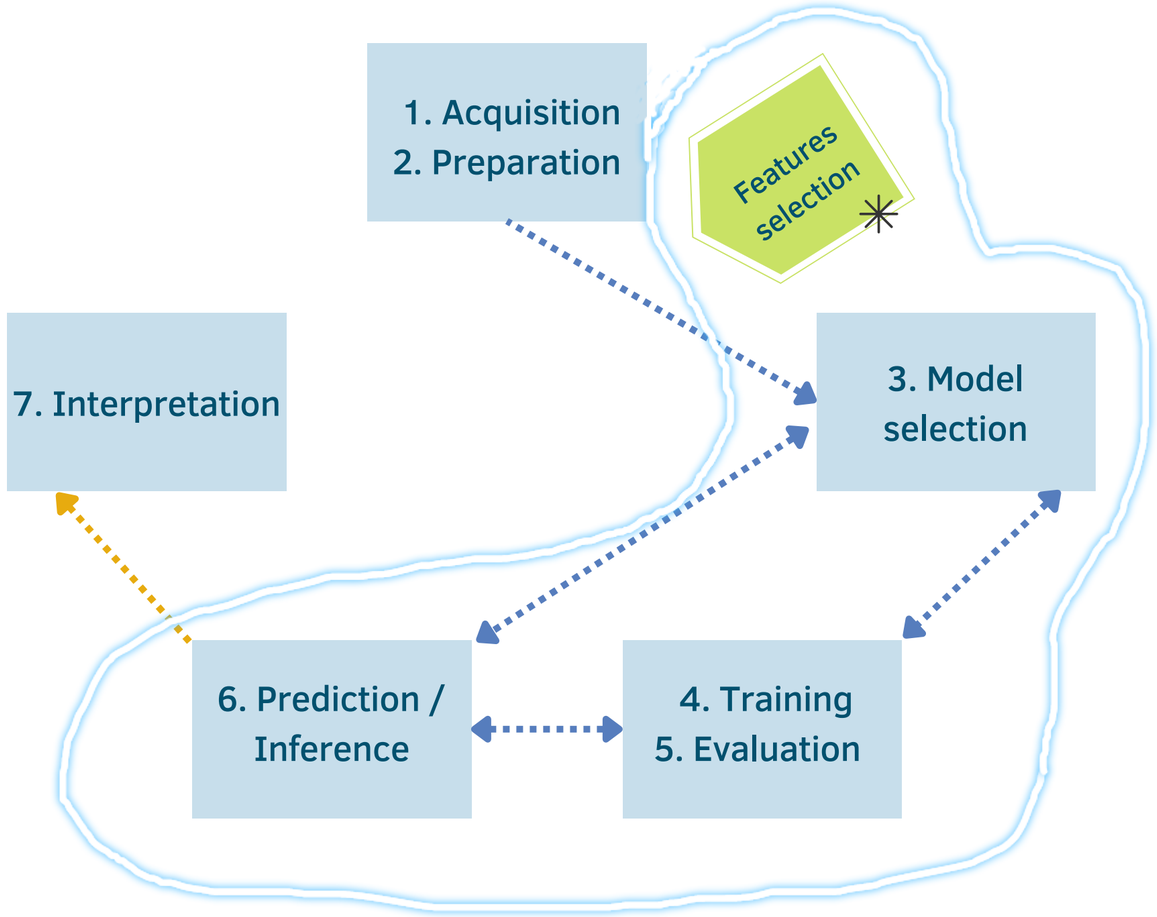
- Modeling

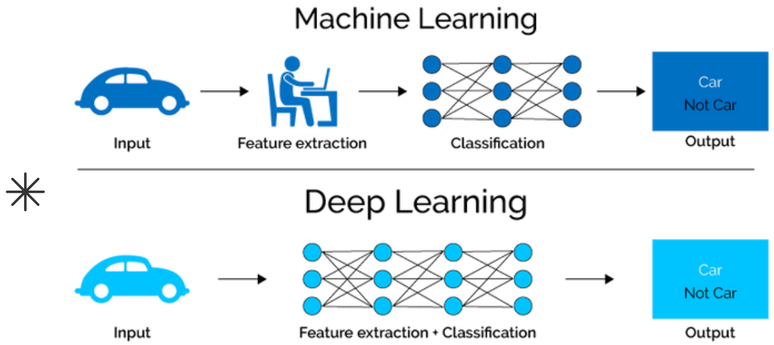
in detail in this course!

Iterative steps
- MODEL SELECTION
- TRAINING
- EVALUATION
- PREDICTION



ML-based modeling





- Modeling

INTERPRETATION

XAI Goal
Trustworthiness Causality
Transferability Informativeness
Confidence Fairness Accessibility
Interactivity Privacy awareness

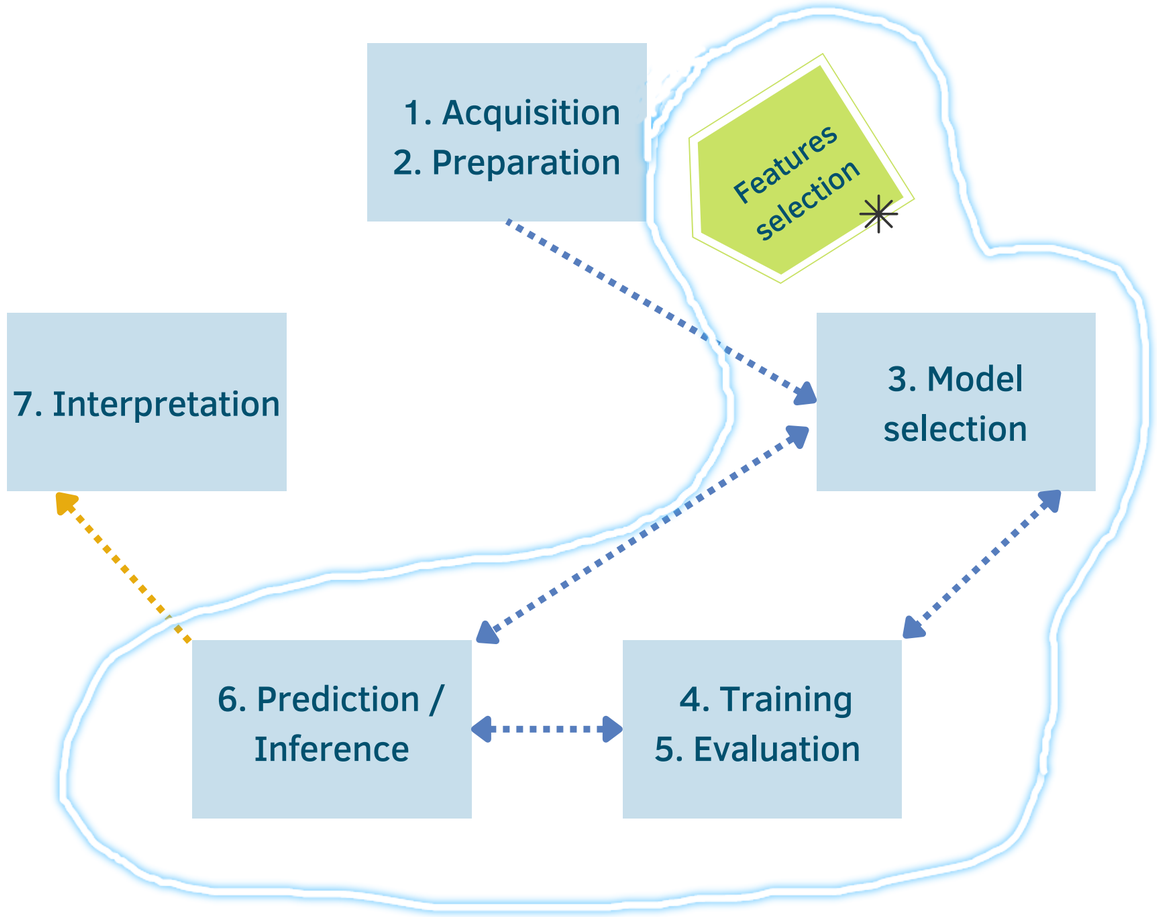
- interpretability: transparency.
- explainability: action or procedure taken by a model with the intent of clarifying or detailing its internal functions.

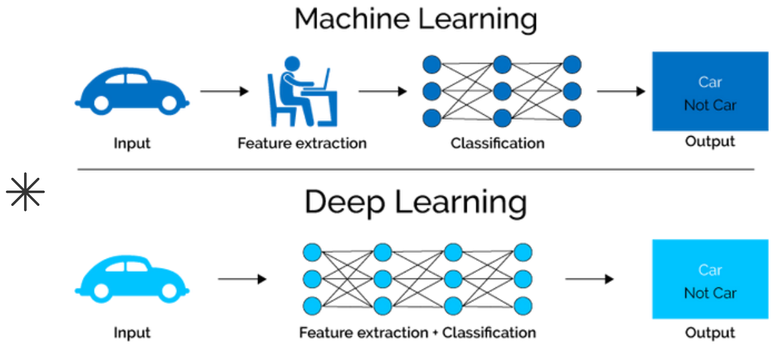
eXplainable AI (XAI): ML techniques

- Explainable models while maintaining a high level of learning performance (e.g., prediction accuracy)
- Understand, trust, manage AI

Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI
Barredo Arrieta et.al. (2019)

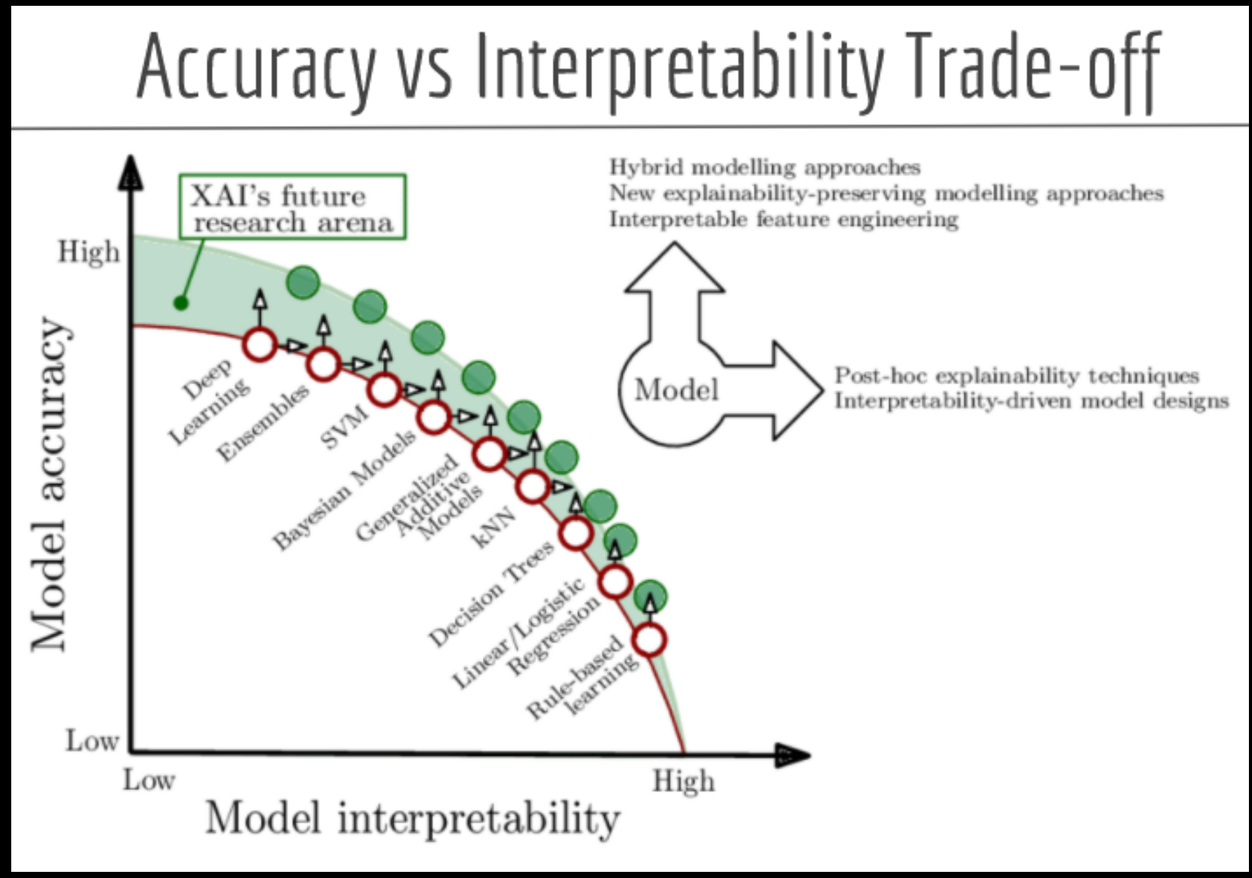
ML-based modeling



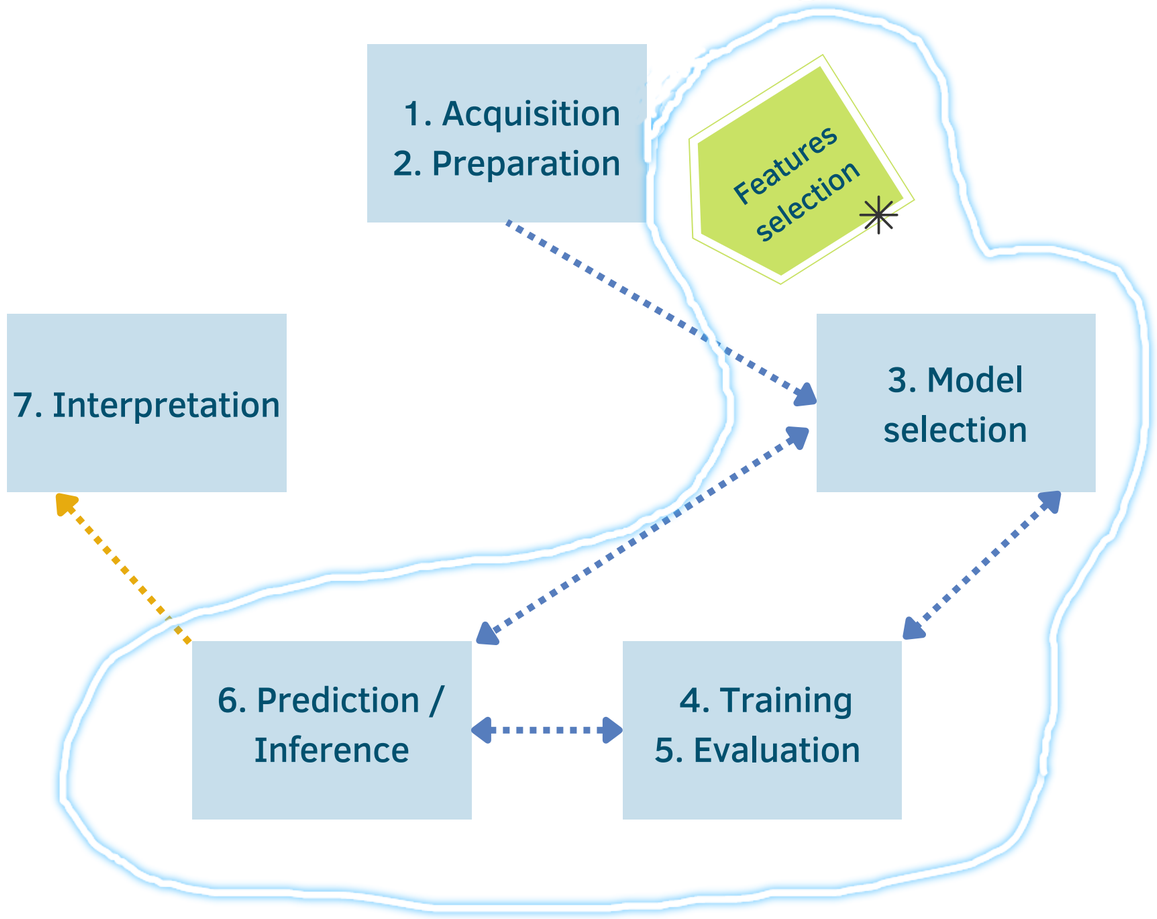


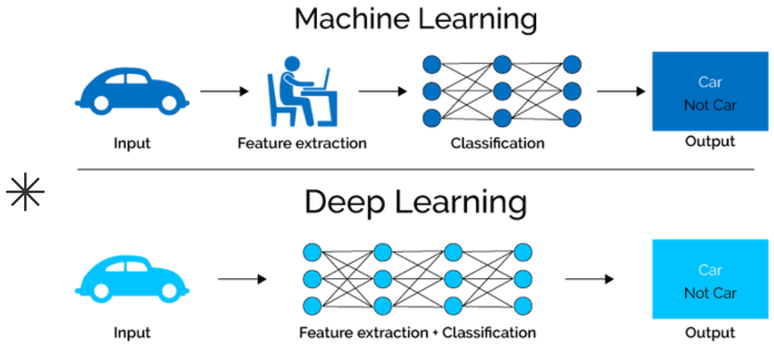
- Modeling

INTERPRETATION



ML-based modeling



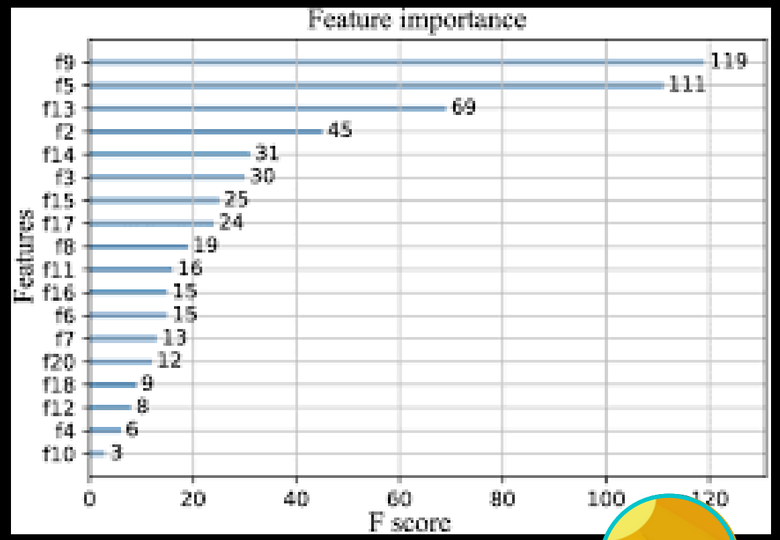


- Modeling

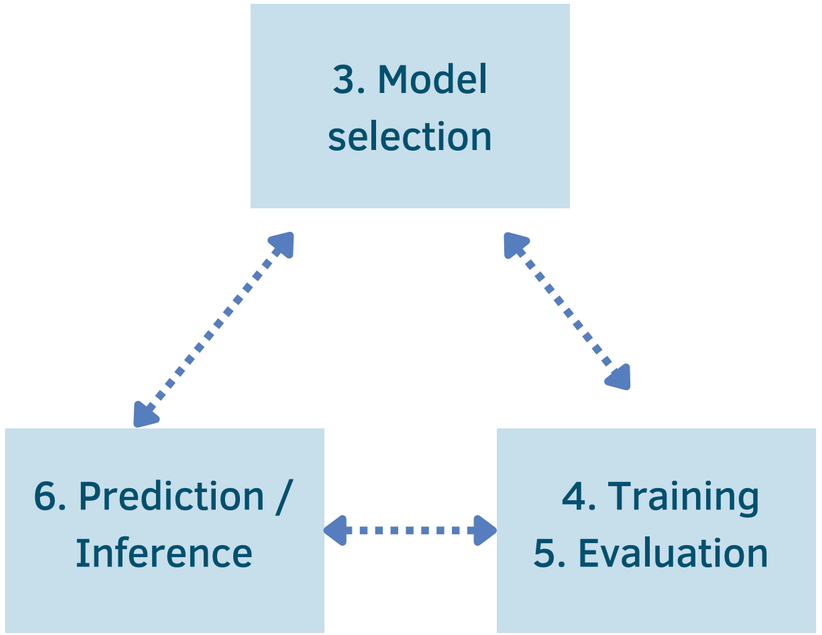
INTERPRETATION

Post-hoc explainability techniques (explaining the black box problem)

- Text explanations (semantic mapping from model to symbols)
- Visualizations (of model's behaviour, e.g. dimensionality reduction),
- Local explanations,
- Explanations by example (centred in extracting representative examples that grasp the inner relationships and correlations found by the model being analyzed),
- Explanations by simplification
- Feature relevance (scoring of features).



ML-based modeling



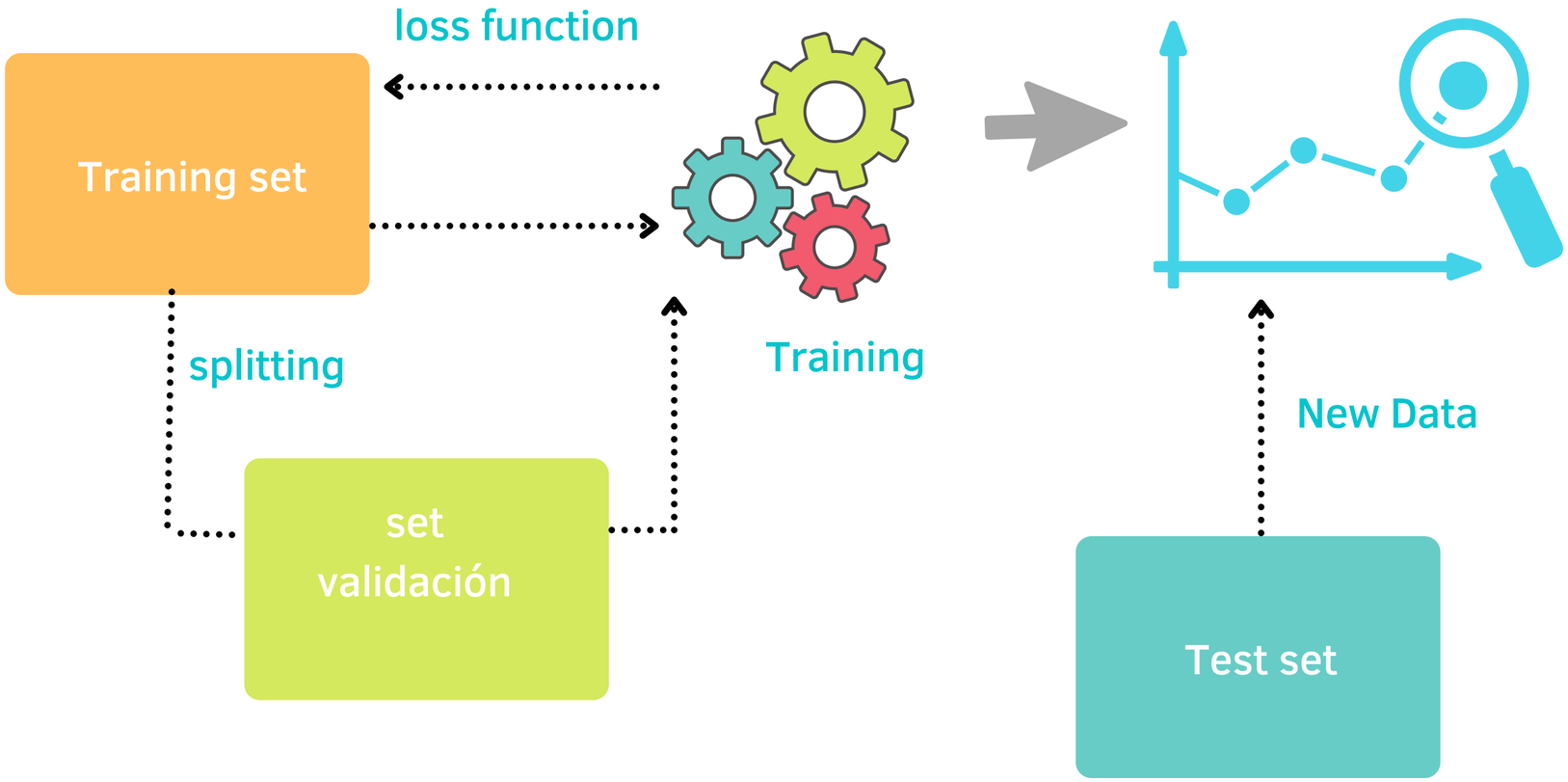
- Modeling

MODEL SELECTION

- The amount of data is important
- Do we have numeric or categorical data? is it labeled? or can it be labeled?
- How "transparent" do we need the model to be?
- Ask the right questions? (that an ML model can answer)
- Explore the bibliography related to your topic
- Compare more than one technique

MODEL IS CHOSEN!

ML-based modeling



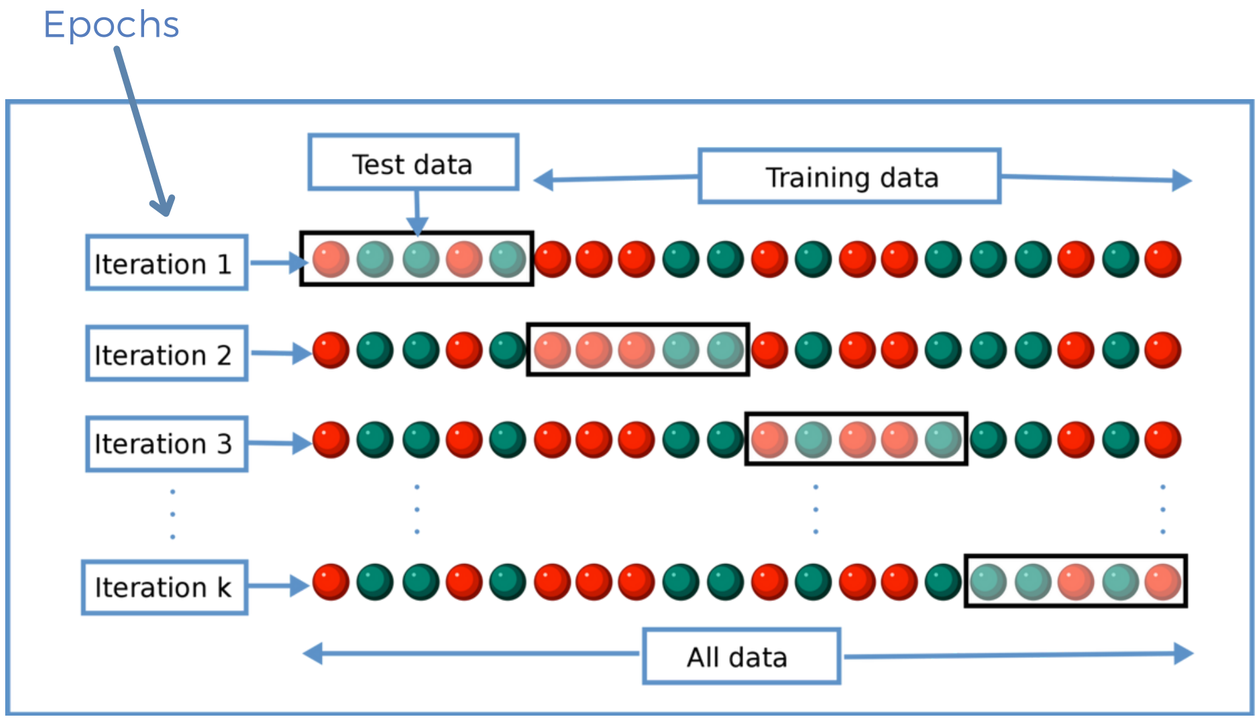
TRAINING

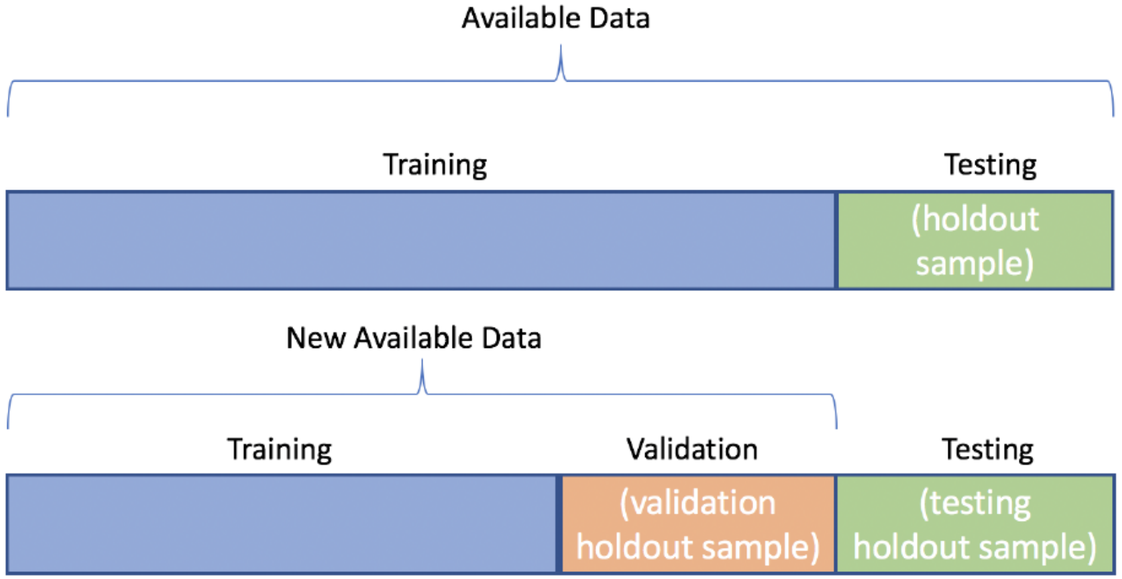
- Learning from the data

SPLITTING

split randomly (*):

- Training set
- Validation set (to choose hyperparameters)
- Test set





Hyperparameters

- Set before training
- E.g. #capas, dropout, learning rate, #neurons, loss function, etc

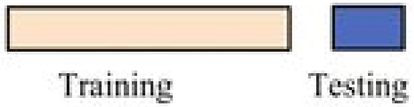
Hyperparameter tuning (choose optimal parameters)

- Methods (e.g. grid search, evolutionary optimization)
- Previous domain knowledge (e.g. constraints)

SPLITTING

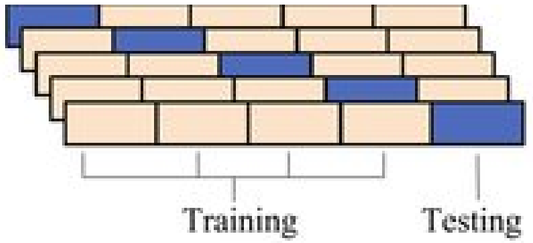
- **data sampling strategies!**

Validation strategies for target prediction methods
(Mathai et al, 2019)



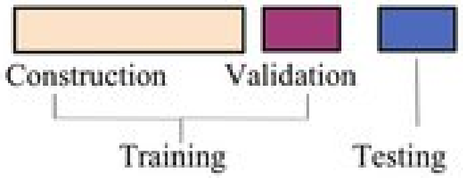
Single train-test split

Single train-test time split



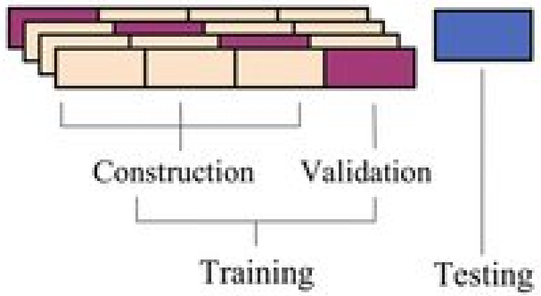
Cross-validation (5-fold)

D



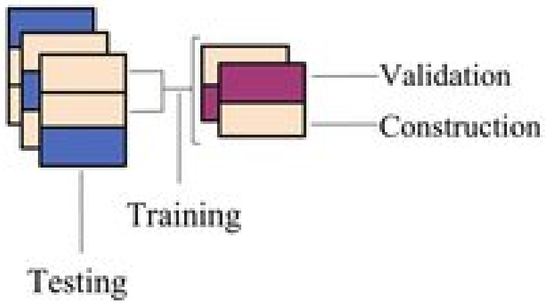
Single train-test split with an external testing set

E



Cross-validation (4-fold) used for internal validation and an external testing set used for

F

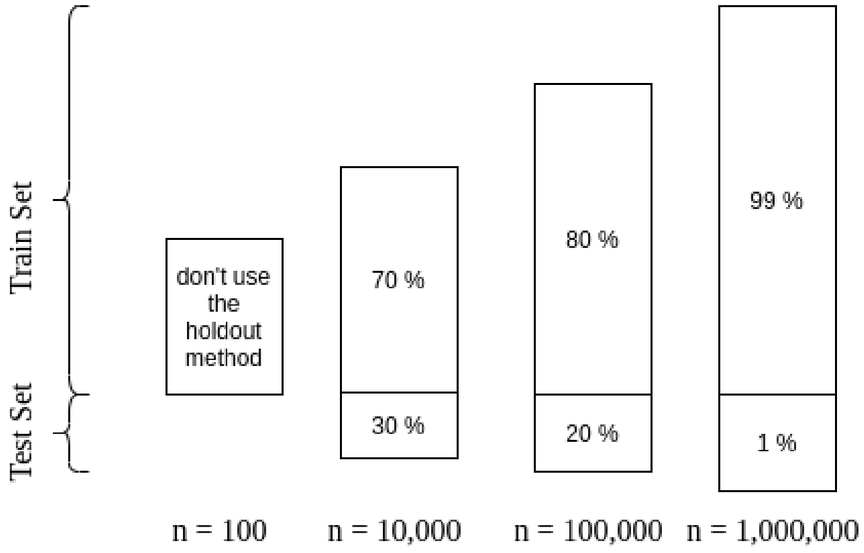


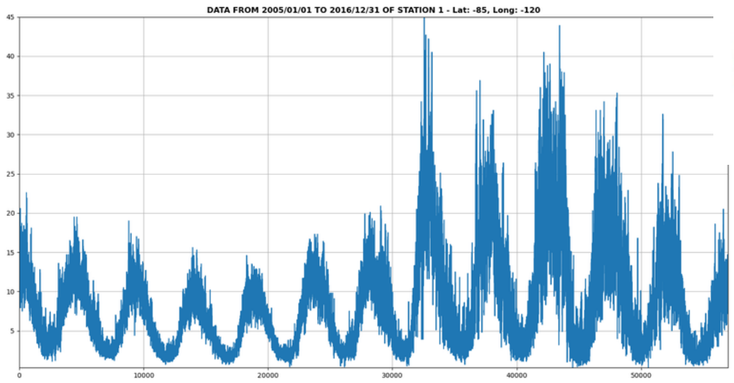
Nested cross-validation with a 2-fold internal validation loop and a 3-fold external validation

SPLITTING

- **data sampling strategies!**

Validation strategies for target prediction methods
(Mathai et al, 2019)

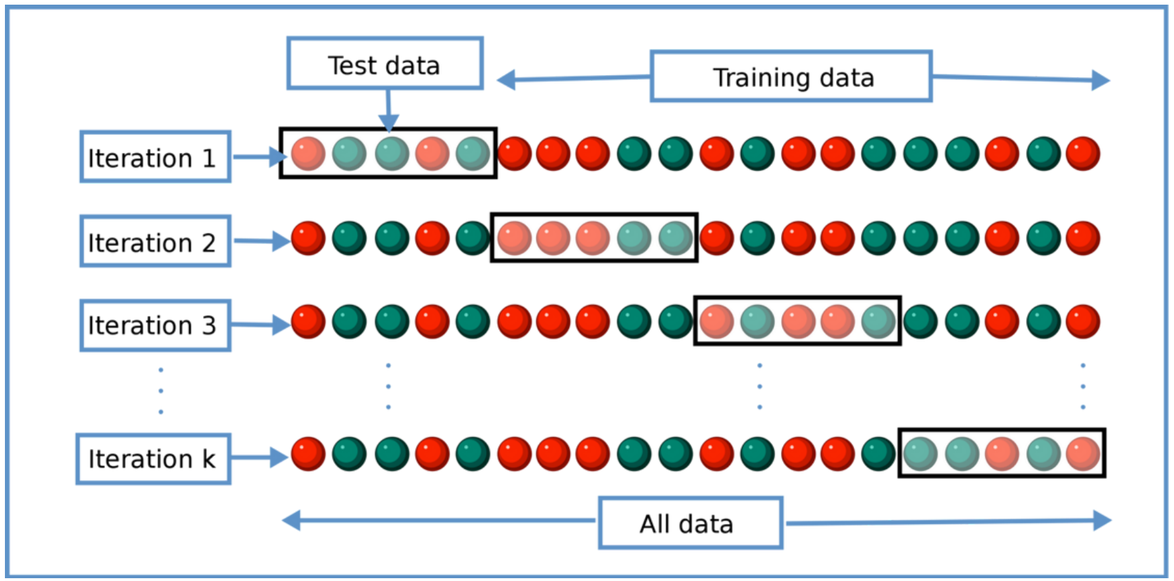


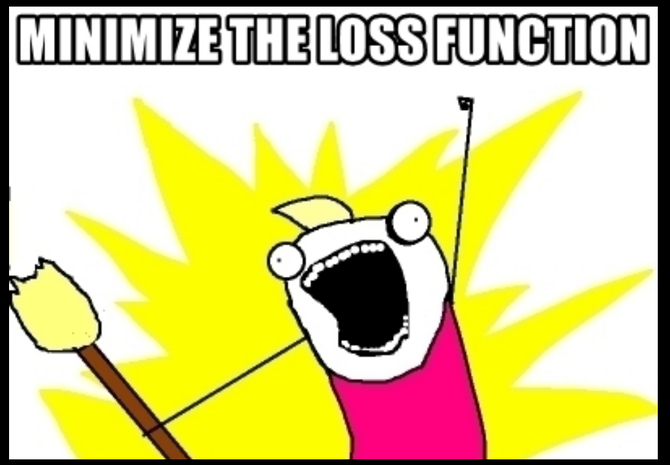


- **Balanced/imbalanced datasets: another story!**

LOSS FUNCTION

- or "cost function"or "target function"
- Measure the cost of inaccurate predictions
- measure how far an estimated value is from its true value
- is a method of evaluating how well specific algorithm models the given data.
- Goal: Minimize the loss function during the training (optimization problem)





WHILE NOT CONVERGE
TRAIN

MIN (loss function)

LOSS FUNCTION

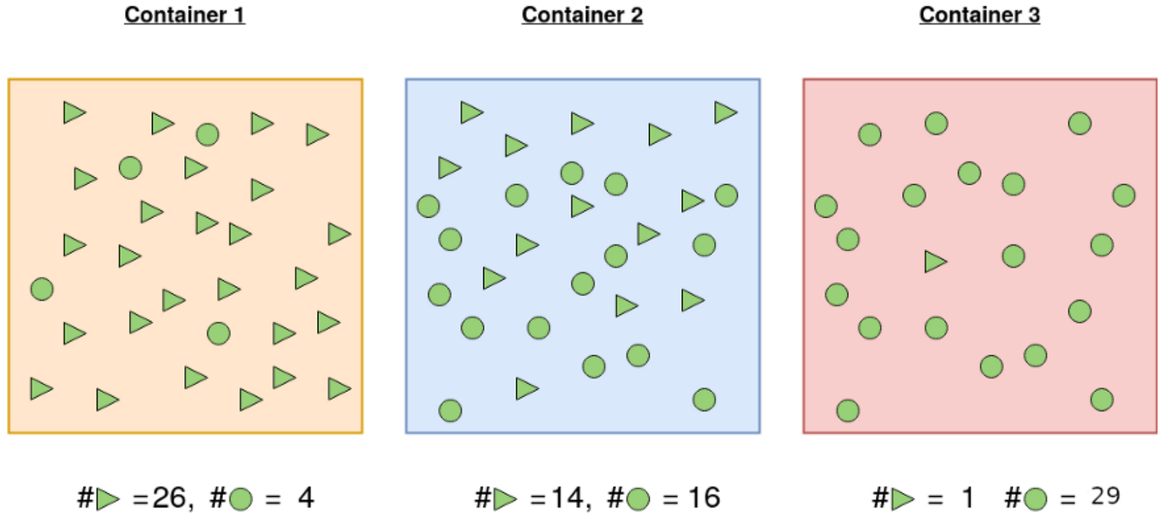
- or "cost function"or "target function"
- Measure the cost of inaccurate predictions
- measure how far an estimated value is from its true value
- is a method of evaluating how well specific algorithm models the given data.
- Goal: Minimize the loss function during the training (optimization problem)

Tipo de problema	**Loss function**	😊
Regresión	Error cuadrático medio (Mean Square Error)	Cercano a 0
Clasificación	Cross entropy Binary cross entropy (mide la distancia entre las probabilidades de salida y la de los valores verdaderos)	Cercano a 0

LOSS FUNCTION CLASSIFICATION

- Entropy of a random variable X is the level of uncertainty inherent in the variable's possible outcome.
- If the entropy is higher, that means we need more information to represent an event (info theory)

$$Entropy, H(X) = -\sum_{i=1}^{n} P(x_i) log_2(P(x_i))$$



The entropy for the first and third container is smaller than the second one. This is because probability of picking a given shape is more certain in container 1 and 3 than in 2.

$$\begin{aligned} H(X) &= -\sum_x p(x)\log(p(x)) \\ &= -[p(x_1)\log_2(p(x_1)) + p(x_2)\log_2(p(x_2))] \\ &= -\left[\frac{26}{30}\log_2\left(\frac{26}{30}\right) + \frac{4}{30}\log_2\left(\frac{4}{30}\right)\right] \\ &= 0.5665 \end{aligned}$$

$$\begin{aligned} H(X) &= -\sum_x p(x)\log(p(x)) \\ &= -[p(x_1)\log_2(p(x_1)) + p(x_2)\log_2(p(x_2))] \\ &= -\left[\frac{14}{30}\log_2\left(\frac{14}{30}\right) + \frac{16}{30}\log_2\left(\frac{16}{30}\right)\right] \\ &= 0.9968 \end{aligned}$$

$$\begin{aligned} H(X) &= -\sum_x p(x)\log(p(x)) \\ &= -[p(x_1)\log_2(p(x_1)) + p(x_2)\log_2(p(x_2))] \\ &= -\left[\frac{1}{30}\log_2\left(\frac{1}{30}\right) + \frac{29}{30}\log_2\left(\frac{29}{30}\right)\right] \\ &= 0.2108 \end{aligned}$$

LOSS FUNCTION CLASSIFICATION

- Cross-entropy (information theory) - the difference between two probability distributions
- Cross-entropy loss increases as the predicted probability diverge from the actual label.

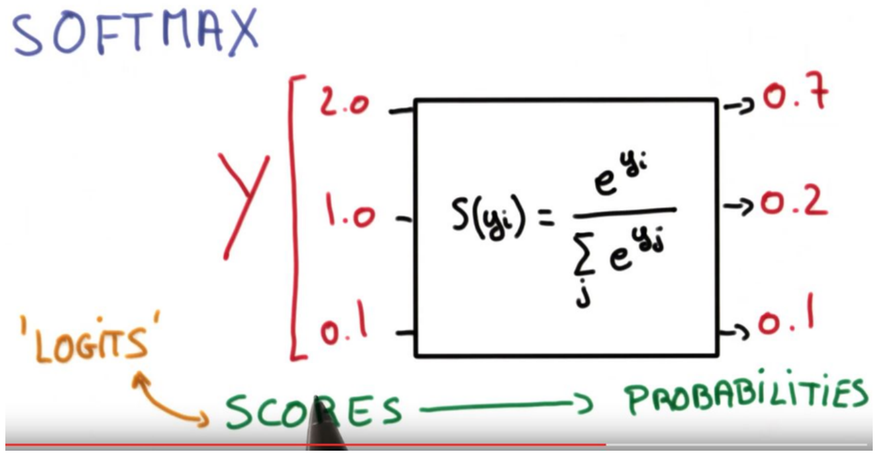
- Cross Entropy

$$H(p,q) = -\sum_{i=1}^{n} p(x_i) log_2(q(x_i))$$

- Binary Cross Entropy

$$\begin{aligned} L &= -\sum_{i=1}^{2} t_i \log(p_i) \\ &= -\left[t \log(p) + (1-t) \log(1-p)\right] \end{aligned}$$

where t_i is the truth value taking a value 0 or 1 and p_i is the Softmax probability for the i^{th} class.



logits = unnormalised (or not-yet normalised) predictions (or outputs) of a model

- is often calculated as the average cross-entropy across all data examples

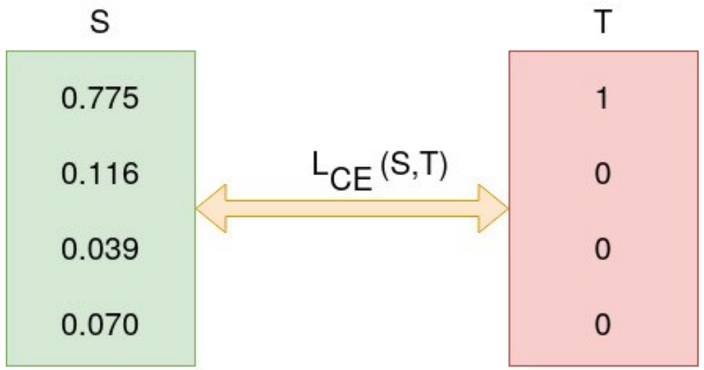
$$L = -\frac{1}{N}\left[\sum_{j=1}^{N} \left[t_j \log(p_j) + (1-t_j) \log(1-p_j)\right]\right]$$

for N data points where t_i is the truth value taking a value 0 or 1 and p_i is the Softmax probability for the i^{th} data point.

LOSS FUNCTION CLASSIFICATION

$$L = -\frac{1}{N}\left[\sum_{j=1}^{N}\left[t_j \log(p_j) + (1-t_j)\log(1-p_j)\right]\right]$$

for N data points where t_i is the truth value taking a value 0 or 1 and p_i is the Softmax probability for the i^{th} data point.



$$\begin{aligned} L_{CE} &= -\sum_{i=1} T_i \log(S_i) \\ &= -\left[1\log_2(0.775) + 0\log_2(0.126) + 0\log_2(0.039) + 0\log_2(0.070)\right] \\ &= -\log_2(0.775) \\ &= 0.3677 \end{aligned}$$

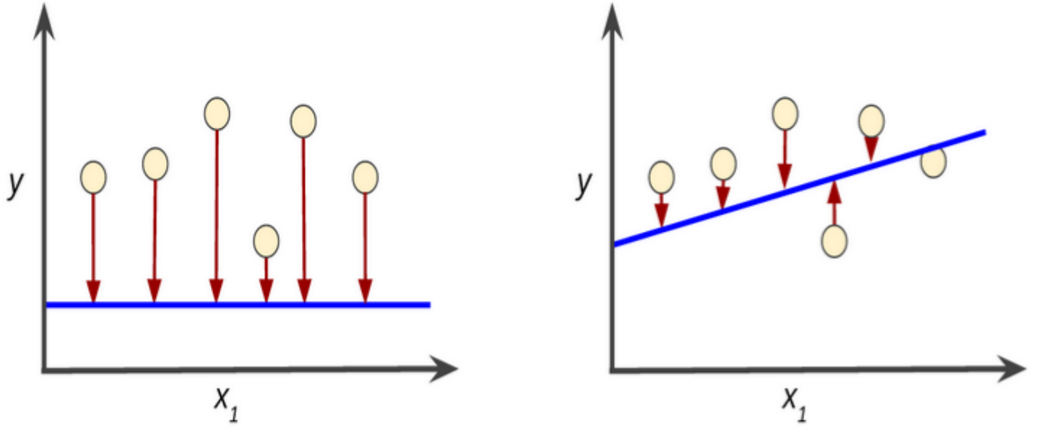
- Notice that when actual label is 1 (ti = 1), second half of function =0 whereas in case actual label is 0 (ti = 0) first half is dropped off. In short, we are just multiplying the log of the actual predicted probability for the ground truth class.

- Each predicted class probability is compared to the actual class desired output 0 or 1 and a score/loss is calculated that penalizes the probability based on how far it is from the actual expected value. The penalty is logarithmic in nature yielding a large score for large differences close to 1 and small score for small differences tending to 0During the training (different iterations using the dataset) the Loss function is optimized
- In binary classification it means reducing the cross-entropy

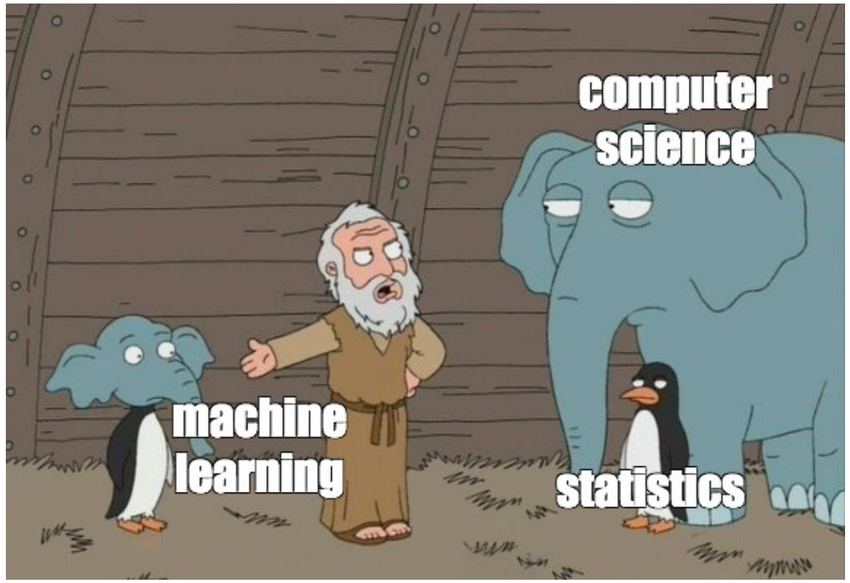
LOSS FUNCTION

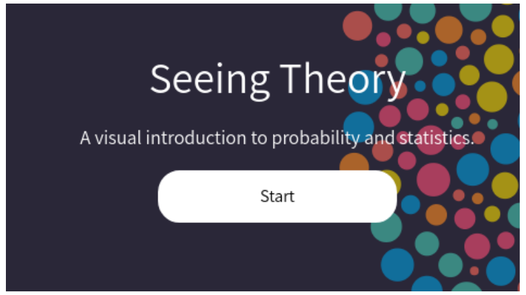
REGRESSION

- Mean square error is measured as the average of the squared difference between predictions and actual observations. It's only concerned with the average magnitude of error irrespective of their direction. However, due to squaring, predictions that are far away from actual values are penalized heavily in comparison to less deviated predictions.

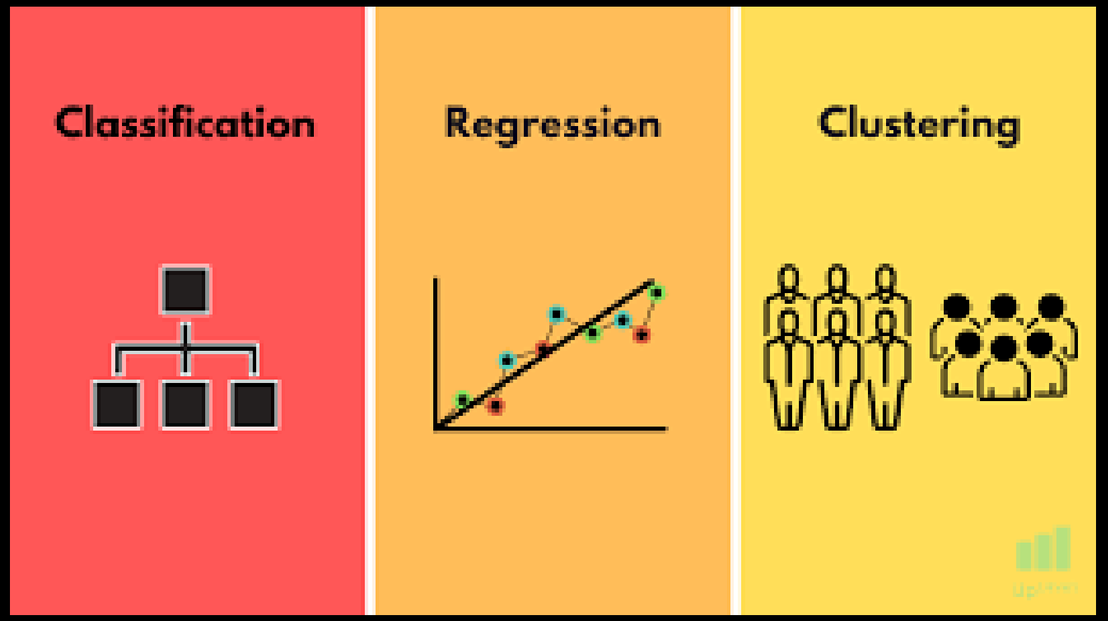


$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}$$





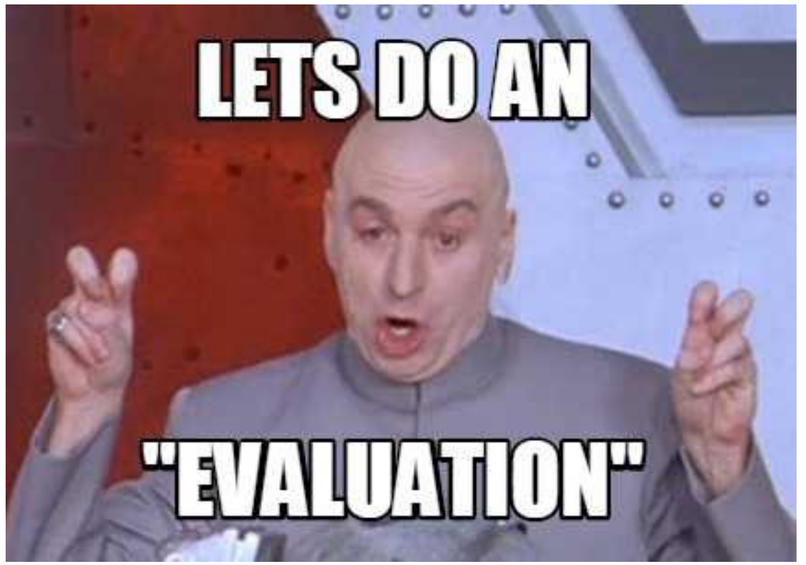
https://seeing-theory.brown.edu



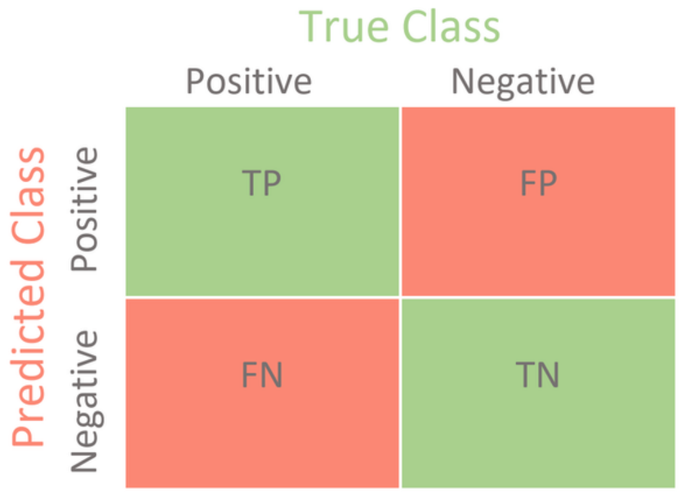
- Classification: confusion matrix (binary), AUC & ROC (multiclass)
- Regresion: MAE,MSE,RMSE (on the test set)
- Clustering: ? ? ? ?

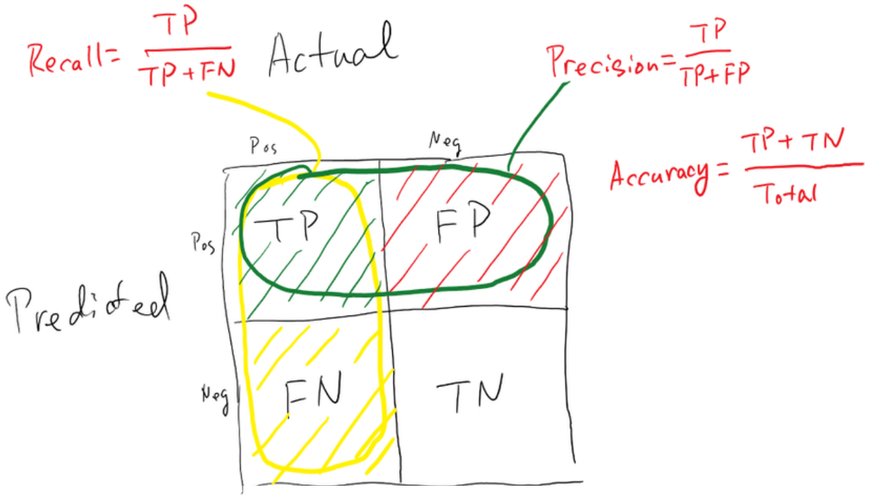
PERFORMANCE

- Sometimes also called model validation (!)
- Uses the **testset** to compare against the prediction with ML
- Metrics



CLASSIFICATION





F - measure = $\frac{2*\text{Recall}*\text{Precision}}{\text{Recall} + \text{Precision}}$

- Accuracy: It is defined as the closeness of the predicted value to the actual value.

$$\text{Accuracy} = (TP + TN)/(TP + TN + FP + FN)$$

- Precision: Precision is defined based on true positive values only out of all positive values.

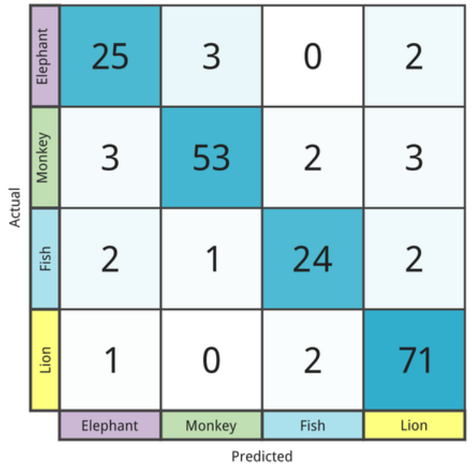
$$\text{Precision} = TP/(TP + FP)$$

- Recall: It is also known as sensitivity or hit rate or true positive rate. It tells how good our estimator or model is to predict the positive values.

$$\text{Recall} = TP/(TP + FN)$$

- https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

CLASSIFICATION



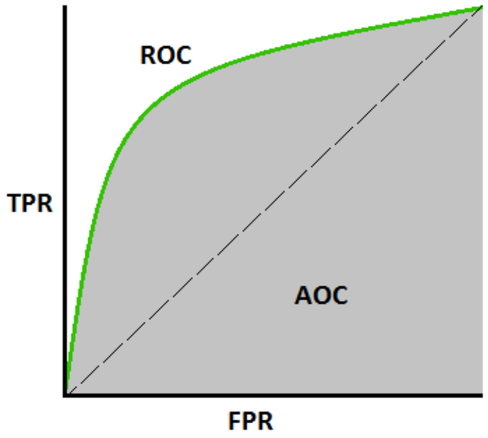
Multiclass

- Confusion matrix
- AUC =Area Under The Curve
- ROC = Receiver Operating Characteristics curve.

better model, >> AUC

AUC = tells how much the model is capable of distinguishing between classes

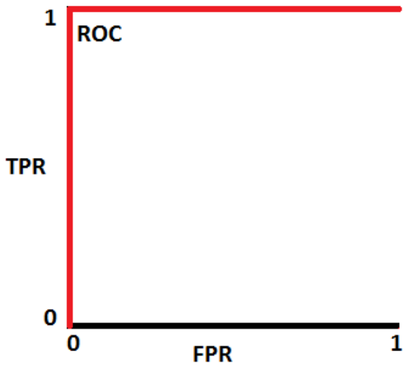
ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes

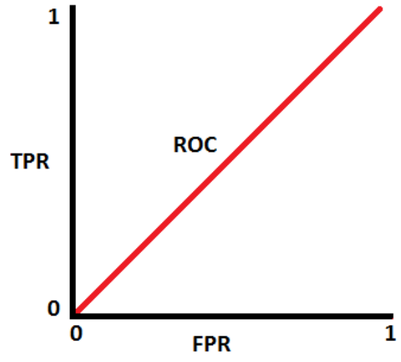


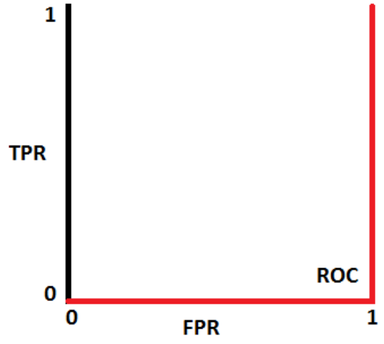
$$\text{TPR /Recall / Sensitivity} = \frac{TP}{TP + FN}$$

$$\text{Specificity} = \frac{TN}{TN + FP}$$

$$\text{FPR} = 1 - \text{Specificity} = \frac{FP}{TN + FP}$$

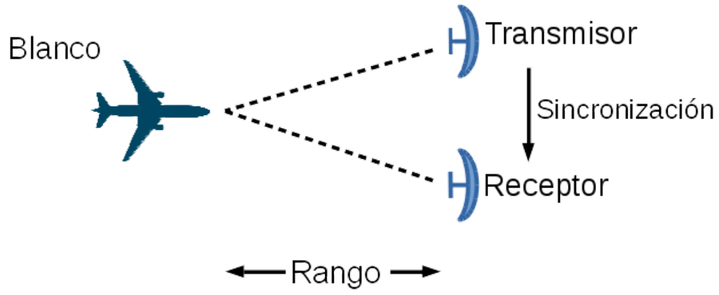






- https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

CLASSIFICATION (Application: Echo detection)



H0:
noise/interference/jamming only

H1:
noise/interference/jamming + echo from a target

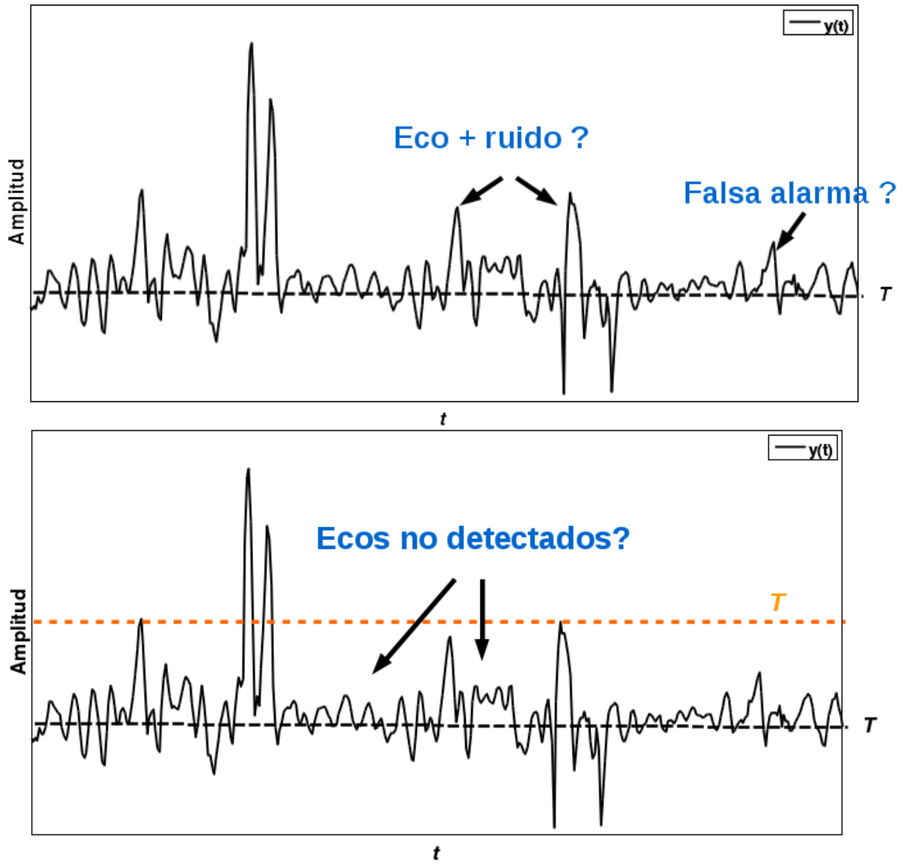
	H1	H0
Target	Positive Detection ✔	Miss ✘
No Target	False Alarm ✘	No Detection ✔

ELSEVIER
CrossMark
Available online at www.sciencedirect.com
ScienceDirect
Advances in Space Research 57 (2016) 1360–1372
ADVANCES IN SPACE RESEARCH (a COSPAR publication)
www.elsevier.com/locate/asr

Automatic ionospheric layers detection: Algorithms analysis

María G. Molina [a,b,*], Enrico Zuccheretti [c], Miguel A. Cabrera [b], Cesidio Bianchi [c], Umberto Sciacca [c], James Baskaradas [d]

Cada yi>T es una detección



- **Pd** (H_1)
- **Pfa** (H_1)
- **Pm** (H_0)

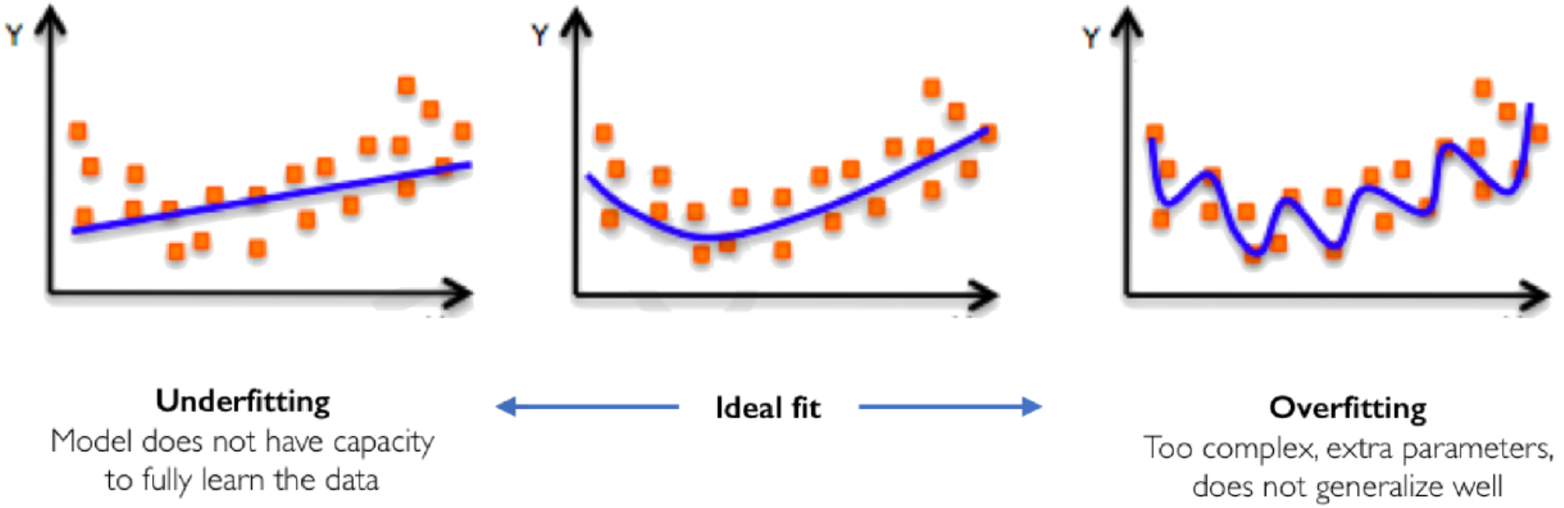
Si Pd ↑, entonces Pfa también ↑

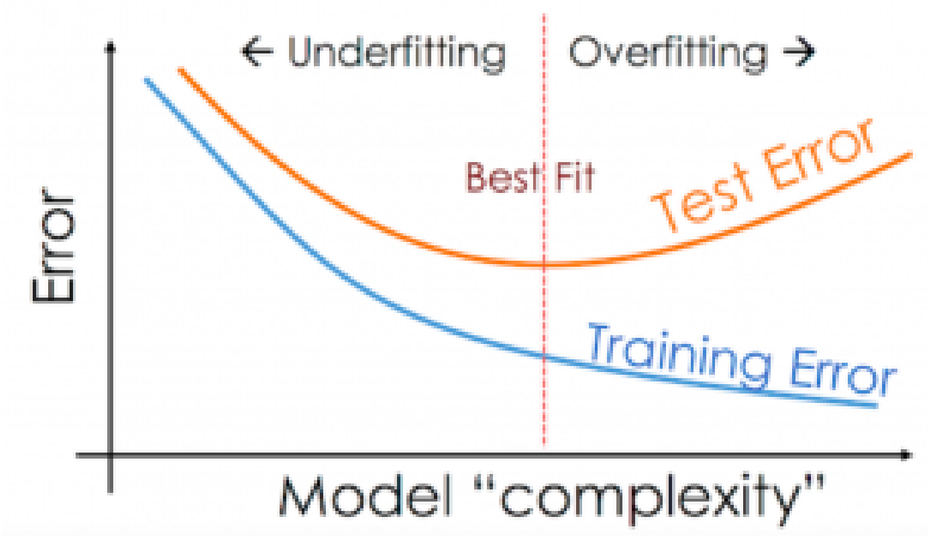
Maximizar Pd sujeta a Pfa <= β

$$T = \alpha\ \hat{y}$$

- How to think the problem from ML point of view?

PERFORMANCE





- There are techniques to overcome overfitting

International Workshop on Machine Learning for Space Weather: Fundamentals, Tools and Future Prospects

7-11 November 2022
This is a hybrid meeting
Buenos Aires, Argentina

Further information:
https://indico.ictp.it/event/9840/
smr3750@ictp.it
+39-040-2240284
Elizabeth Brancaccio

Dra María Graciela Molina
FACET-UNT / CONICET
Tucumán Space Weather Center - TSWC

https://spaceweather.facet.unt.edu.ar/
IG -> @spaceweatherargentina

gmolina@herrera.unt.edu.ar