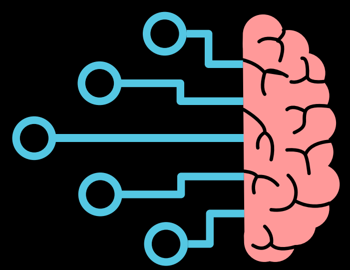
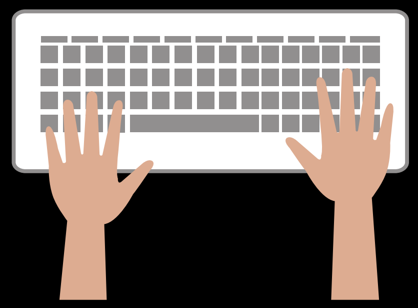
Info



- AI - ML - DL
- Concepts & recommendations

- SWx concepts
- How to see SWx with data scientist glasses
- Open discussions

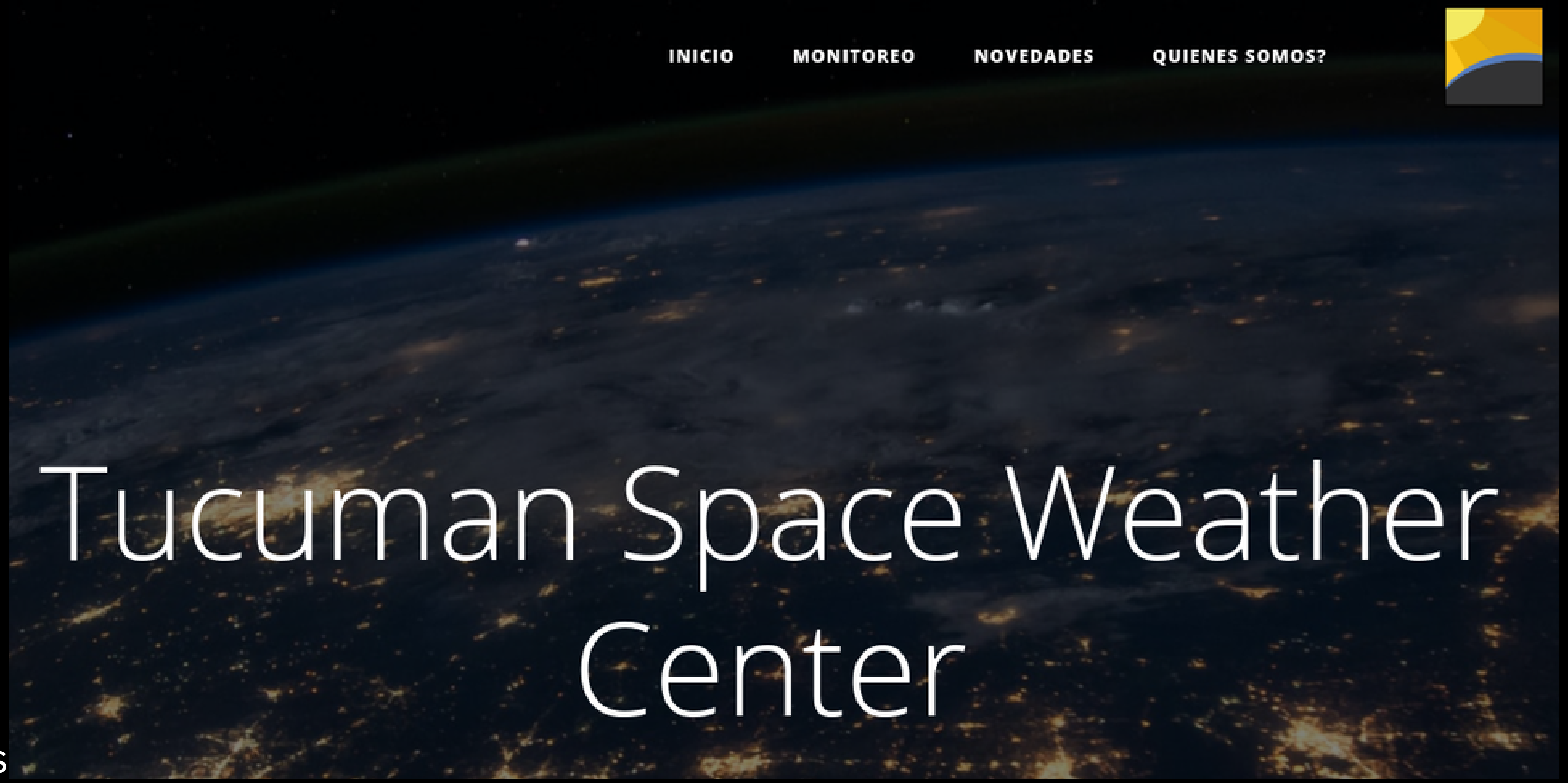
- Tools & best practices

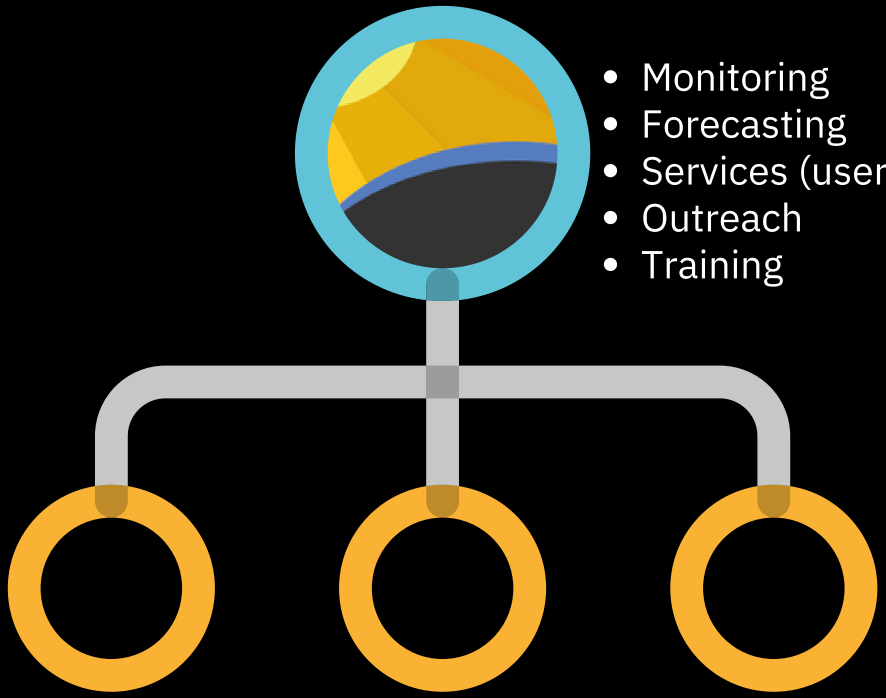


- Hands-on

To pass the course: present TPs

About us ...





- Monitoring
- Forecasting
- Services (user tailored)
- Outreach
- Training

Instrumentation

- Deployment
- Design and Development
- Networks
- Radar systems development

Data Managment and Operative tools

- Data infraestructure
- Tailored software development
- DS approach: AI modeling/ forecasting, data bases. Operative AI
- ML assisted forecasters tools

Scientific goals

- Atmospheric physics
- Space Weather chain of events understanding. Sun-Earth connections
- Geomagnetism
- Radiopropagation
- ML and stats methods for SWx
- XAI

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

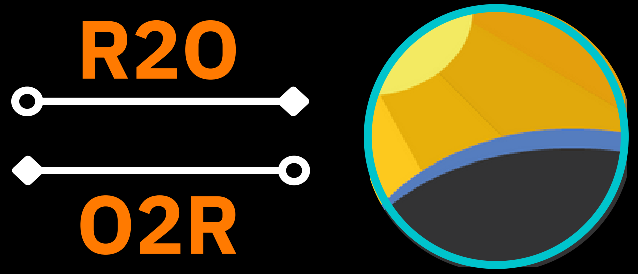
Instagram: /spaceweatherargentina

- Instrumentation development
- Data-driven modeling
- Physics
- Tailored software development
- Operative for ICAO
- Training & Capacity building

Tucuman Space Weather Center FACET-UNT

Canal oficial Tucumán Space Weather Center (TSWC) El TSWC es el centro de monitoreo de meteorología del espacio de la...

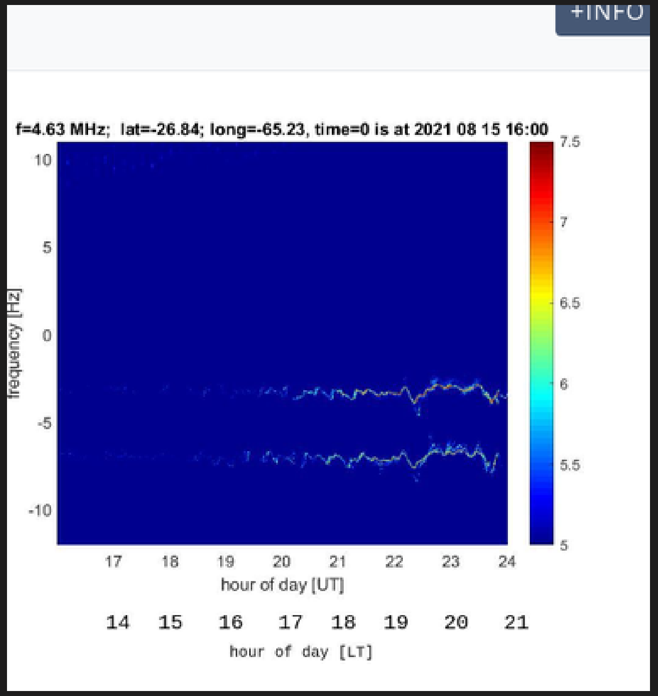
YouTube



About us …

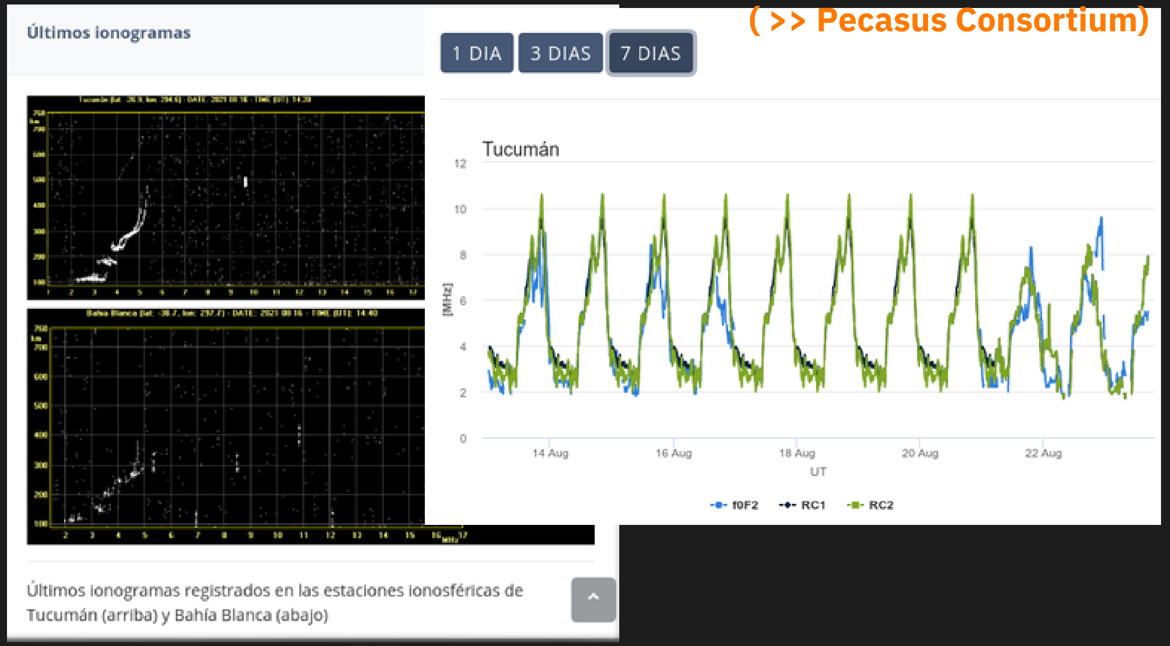
- Ionogram scaling correction by experts (tool for ionospheric interpreters+ DB)
- Cosmic Ray detector: LAGO collaboration
- Ionospheric conditions forcasting (f0F2 and TEC, single station, regional and global) - AI modelling (status: concept testing)
- NeQuick ingested -> f0F2
- Automatic Flare Analysis Tool (UCR)
- TIDs automatic detection
- TEC calibration service
- New GNSS receiver - New 3D Doppler radar
- …

AGWs automatic detection

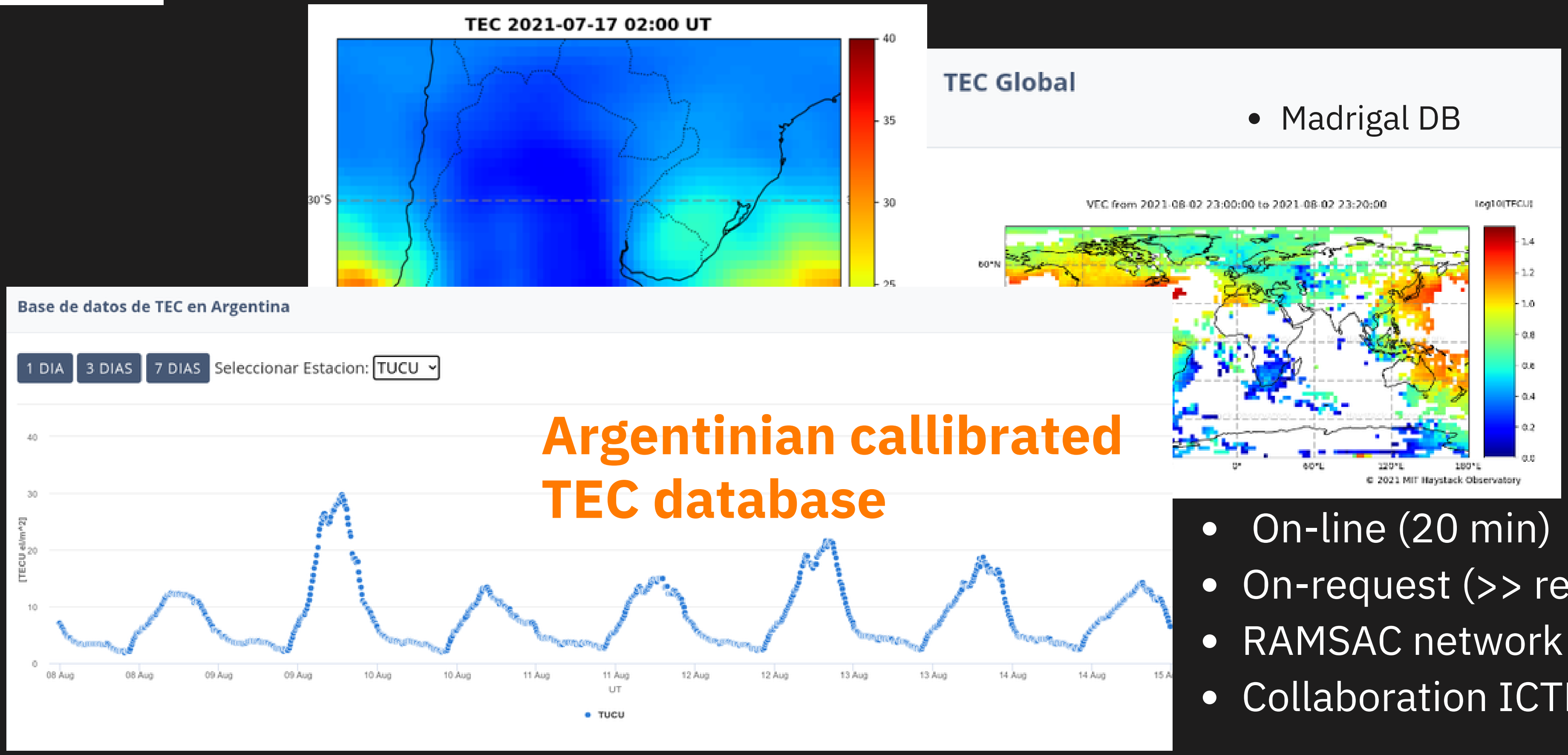


Solar wind - magnetosphere coupling monitoring

Ionosonde Argentine Net
(>> Pecasus Consortium)

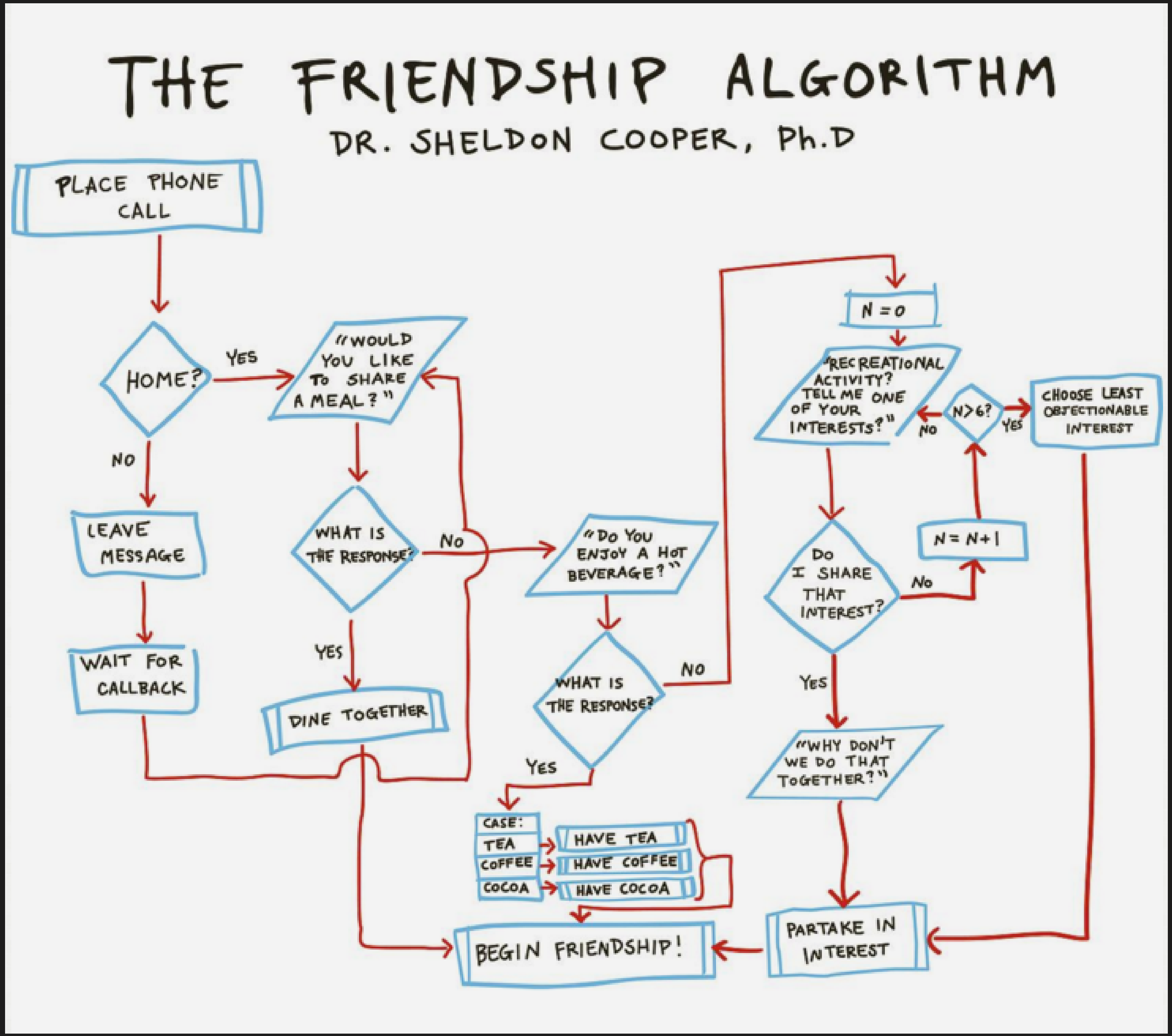


Global and Regional TEC maps

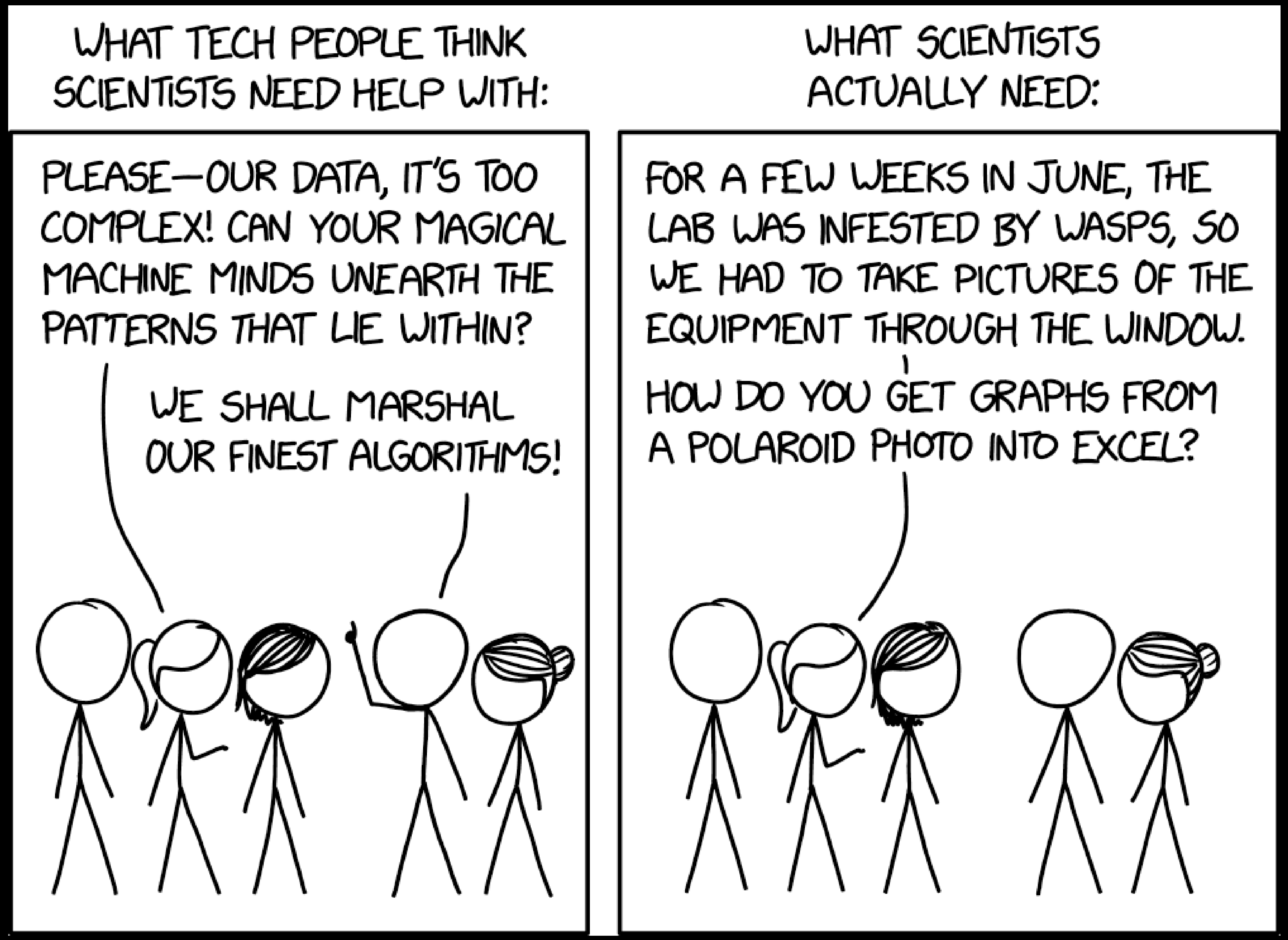


- On-line (20 min)
- On-request (>> resol)
- RAMSAC network
- Collaboration ICTP researchers

What about you?



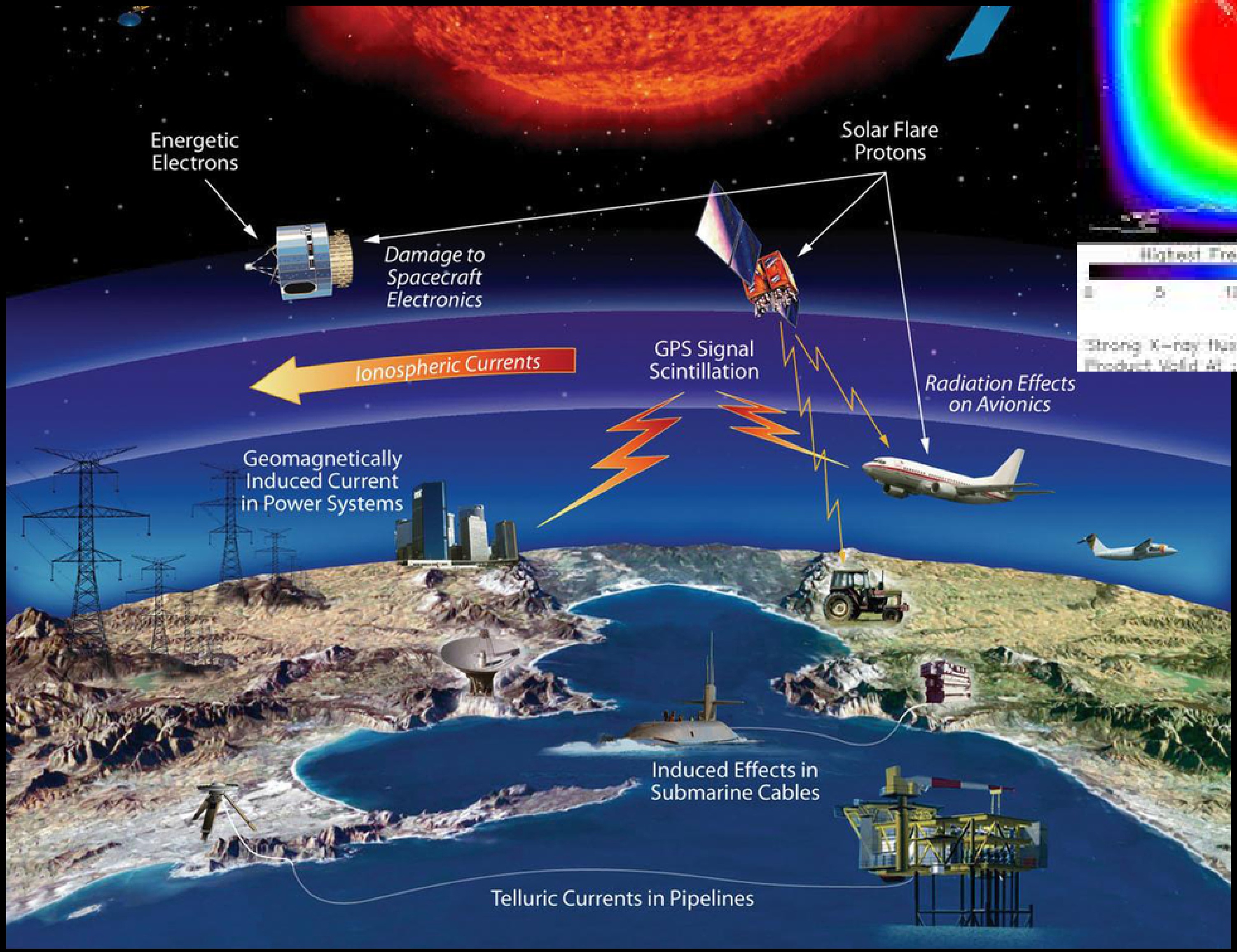
Intro

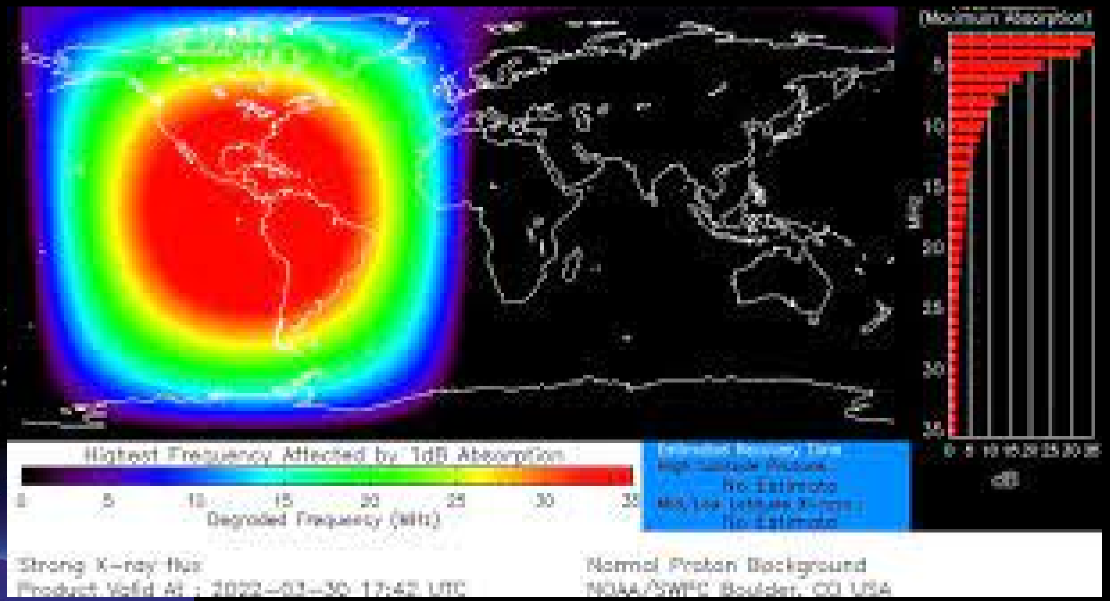

WHAT TECH PEOPLE THINK SCIENTISTS NEED HELP WITH:
PLEASE—OUR DATA, IT'S TOO COMPLEX! CAN YOUR MAGICAL MACHINE MINDS UNEARTH THE PATTERNS THAT LIE WITHIN?
WE SHALL MARSHAL OUR FINEST ALGORITHMS!
WHAT SCIENTISTS ACTUALLY NEED:
FOR A FEW WEEKS IN JUNE, THE LAB WAS INFESTED BY WASPS, SO WE HAD TO TAKE PICTURES OF THE EQUIPMENT THROUGH THE WINDOW.
HOW DO YOU GET GRAPHS FROM A POLAROID PHOTO INTO EXCEL?

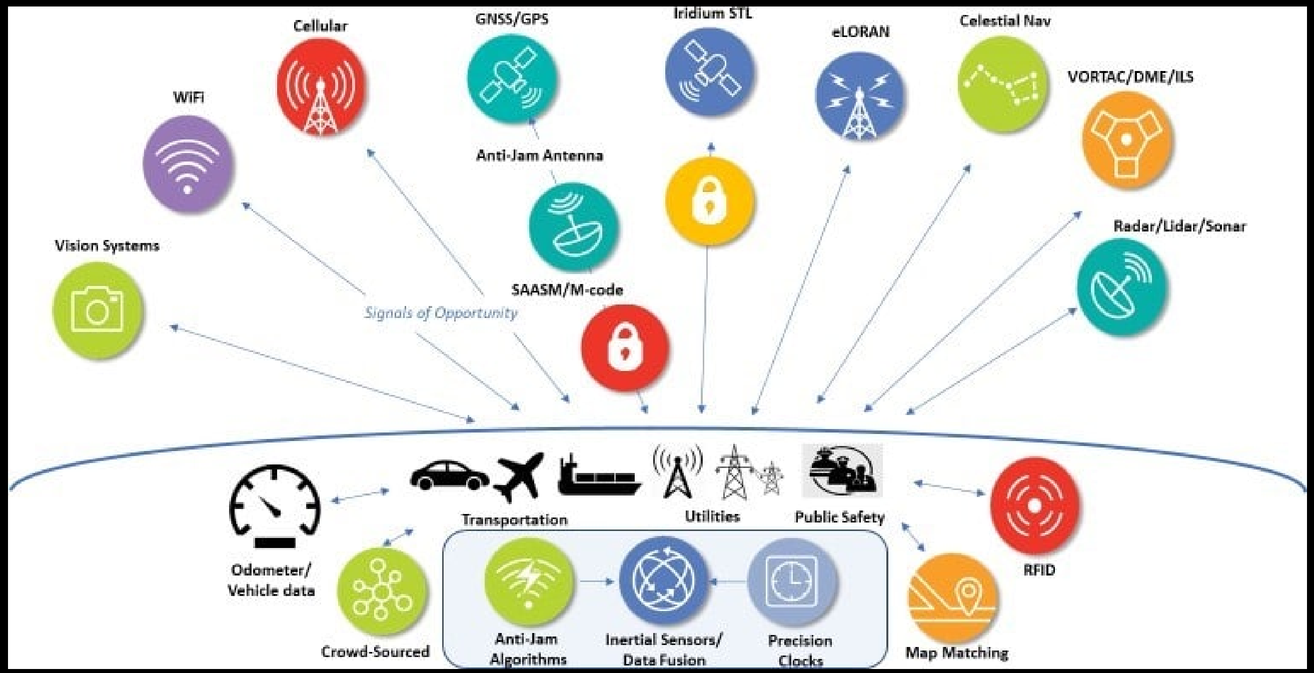
Space Weather (SWx)

Many definitions …

Space Weather is the physical and phenomenological state of natural space environments. The associated discipline aims, through observation, monitoring, analysis and modelling, at understanding and predicting the state of the Sun, the interplanetary and planetary environments, and the solar and non-solar driven perturbations that affect them, and also at forecasting and nowcasting the potential impacts on biological and technological systems. -COST Action 724 , 2009







COST (European Cooperation in Science and Technology) is a funding organisation for research and innovation networks

TSWC, 2022

Space Weather (SWx)

COST (European Cooperation in Science and Technology) is a funding organisation for research and innovation networks

Space Weather is the physical and phenomenological state of natural space environments. The associated discipline aims, through observation, monitoring, analysis and modelling, at understanding and predicting the state of the Sun, the interplanetary and planetary environments, and the solar and non-solar driven perturbations that affect them, and also at forecasting and nowcasting the potential impacts on biological and technological systems. -COST Action 724 , 2009

Many definitions ...

Sun, Interplanetary and planetary environments (coupled system)

- Science

Impacts on biological and technological systems (health and economical impacts)

- Steakholders
- R2O
- Drives the services (e.g. ICAO - interested in HF)

Instrumentation

- Coverage
- Partial observations
- Scientific + military + commercial
- Networks

Interdisciplinary field (complex problem)

- Now/Forecasting
- Modelling
- Analysis
- Monitoring
- Observations

Space Weather (SWx)

COST (European Cooperation in Science and Technology) is a funding organisation for research and innovation networks

Space Weather is the physical and phenomenological state of natural space environments. The associated discipline aims, through observation, monitoring, analysis and modelling, at understanding and predicting the state of the Sun, the interplanetary and planetary environments, and the solar and non-solar driven perturbations that affect them, and also at forecasting and nowcasting the potential impacts on biological and technological systems. -COST Action 724 , 2009

Many definitions ...

Sun, Interplanetary and planetary environments (coupled system)

- Science

Impacts on biological and technological systems (health and economical impacts)

- Steakholders
- R2O
- Drives the services (e.g. ICAO - interested in HF)

DATA & PROGRAMMING

Instrumentation

- Coverage
- Partial observations
- Scientific + military + commercial
- Networks

Interdisciplinary field (complex problem)

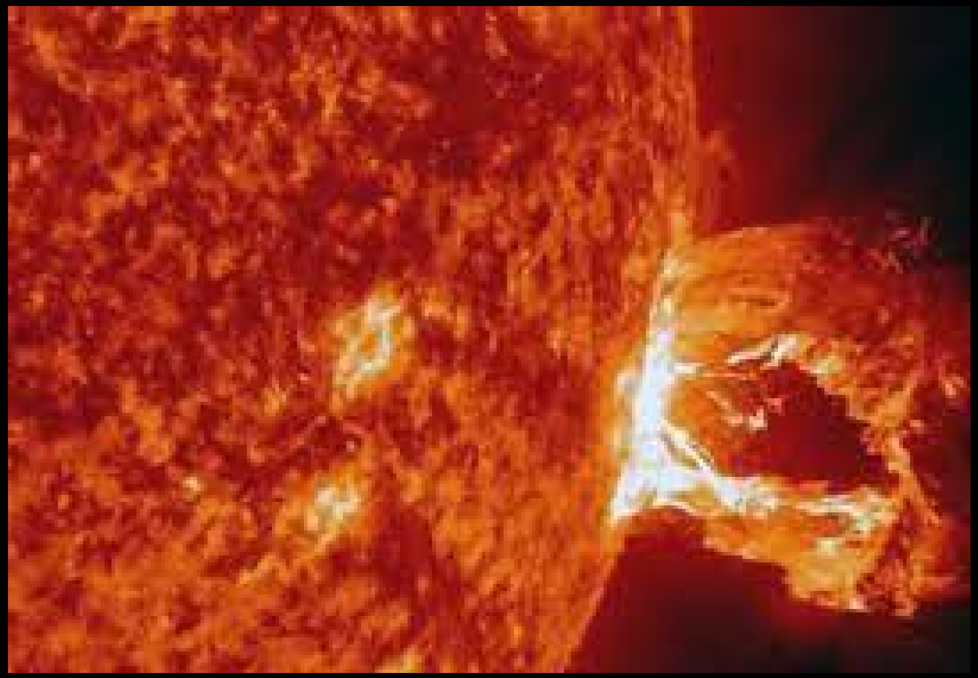
- Now/Forecasting
- Modelling
- Analysis
- Monitoring
- Observations

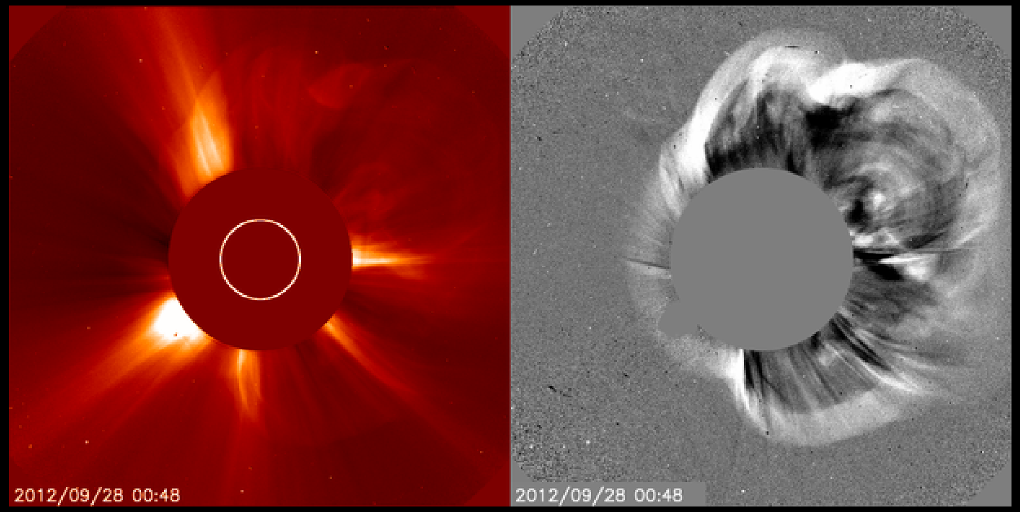
TSWC, 2022

Space Weather (SWx) Main sources & propagation - Observations - Consequences

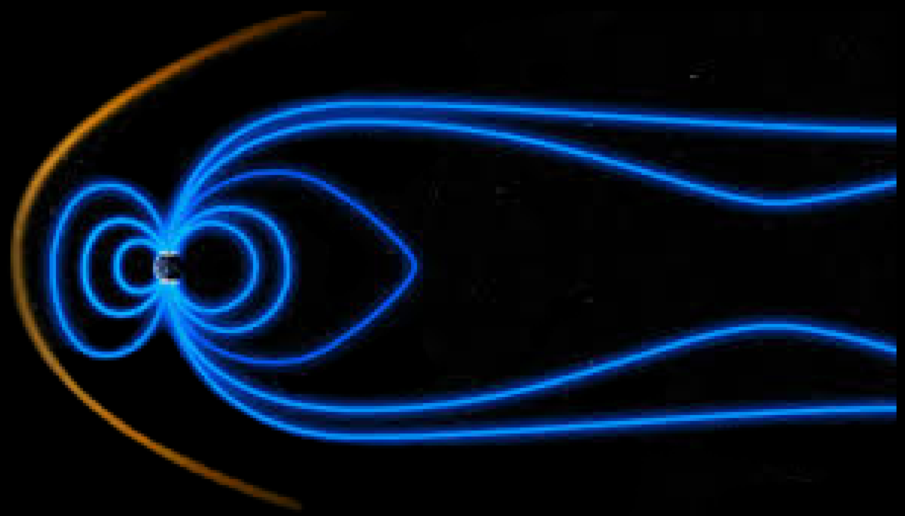
CMEs: large portions of the corona (outer atmosphere) of the Sun explosively blown (plasma) into space. Sometimes are directed toward the Earth. If CMEs reach the Earth's magnetosphere, CMEs magnetic field can interact with the Earth's magnetic fields and produce (under special conditions) the so-called geomagnetic storms

https://umbra.nascom.nasa.gov/lasco/observations/halo/2012/120928/





- Solar wind: constant outflow of electrons and protons from the Sun (interacts with Earth Mag field)



- Can reach the Earth in ~ 1.5 - 3 days (typically)
- SLOW events but difficult to predict

Geomagnetic Storms: Temporary disturbances of the Earth's magnetic field (coupling solar wind-magnetosphere through magnetic reconection)

Space Weather (SWx) Main sources & propagation - Observations - Consequences

Solar flares: Energetic bursts of radiation (radio, white light, EUV, soft X-rays, hard X-rays and Gamma-rays) and particles triggered by the release of magnetic energy on the Sun.

Others: e.g. filaments

CMEs

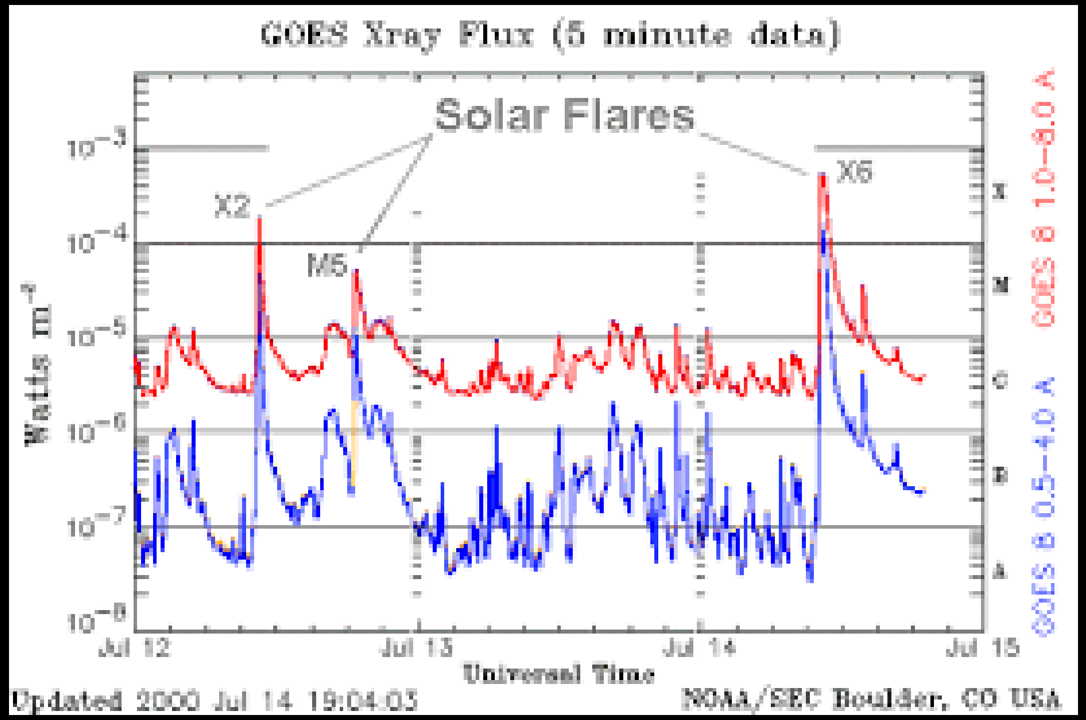
- Slow events: ~ 1.5 - 3 days (typically)
- Difficult to predict

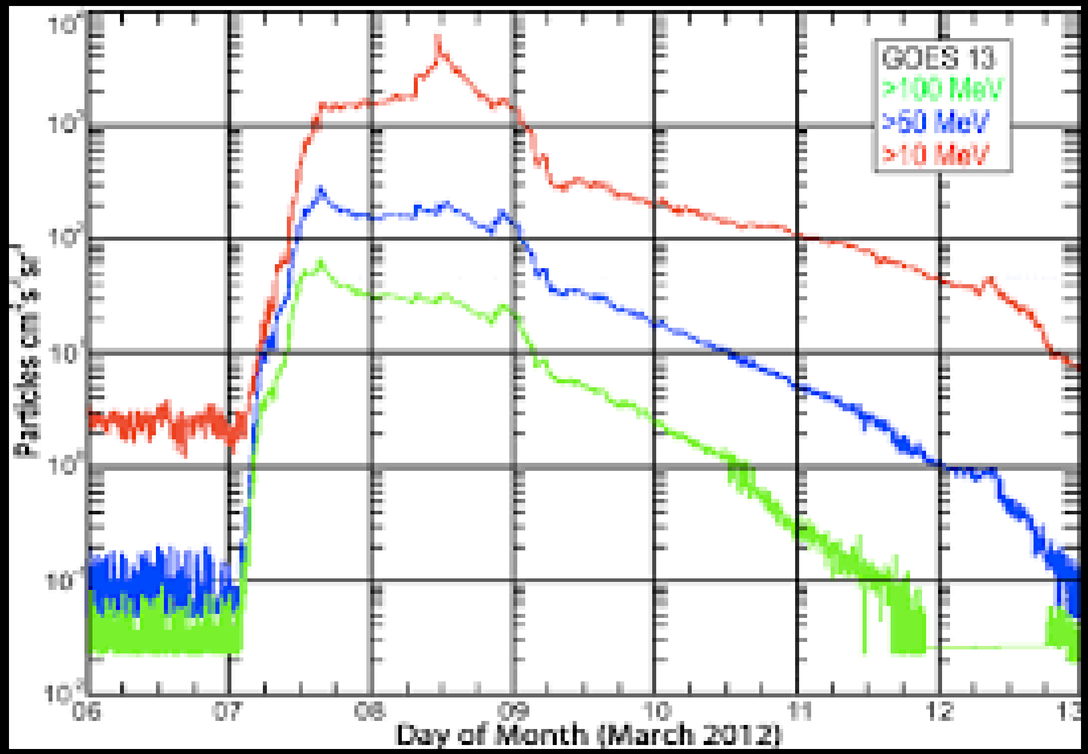
SOFT X-RAY ENHANCEMENT:

- Very Fast events: ~8 m to reach the Earth (dayside)
- Hard to predict (almost instantaneous with the observations)
- X-flares (X-ray radiation emission) classification (Near Earth): A, B, C, M or X [W/m2] of X-rays between 1- 8 Å
- Produces sudden ionization in the ionosphere

SEPs

- Can reach the Earth in ~ 1.5 - 3 days (typically)
- fast events: ~20 min



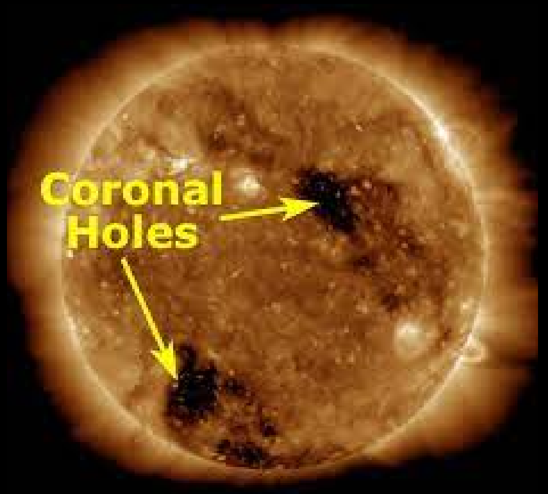


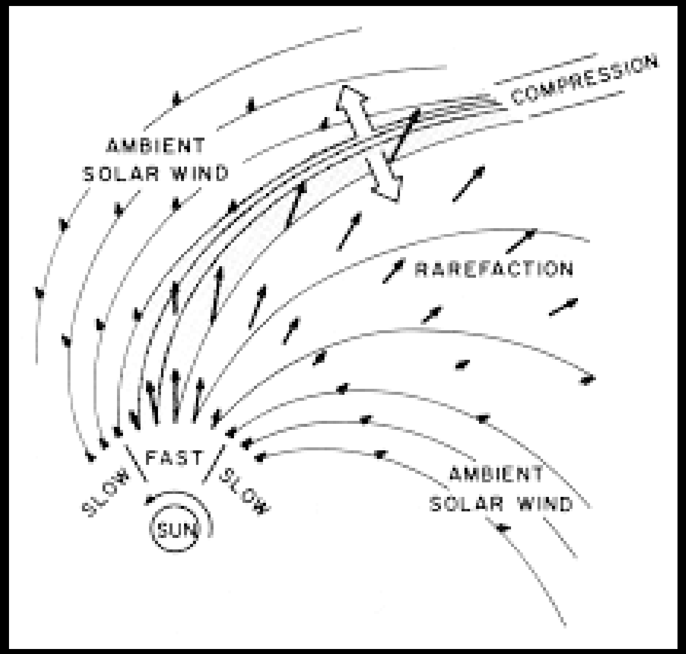
TSWC, 2022

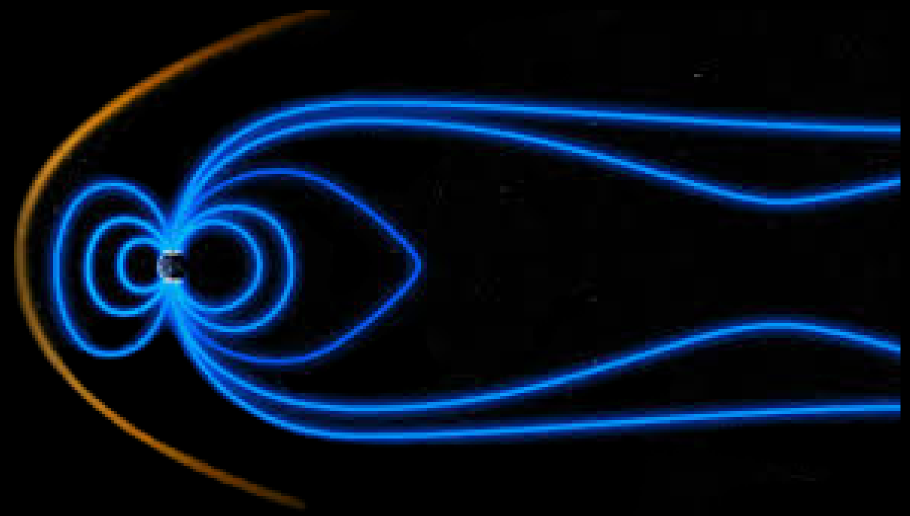
Space Weather (SWx) Main sources & propagation - Observations - Consequences

CHs: Portion of the Sun atmosphere with lower density. These are regions where the Sun magnetic field lines are connected directly with the interplanetary medium, allowing solar material to escape out in a high-speed stream of solar wing.

Geomagnetic Storms: Temporary disturbances of the Earth's magnetic field (coupling solar wind-magnetosphere through magnetic reconection)







Consequences of a Solar Wind Stream Interaction Region on the Low Latitude Ionosphere: Event of 7 October 2015

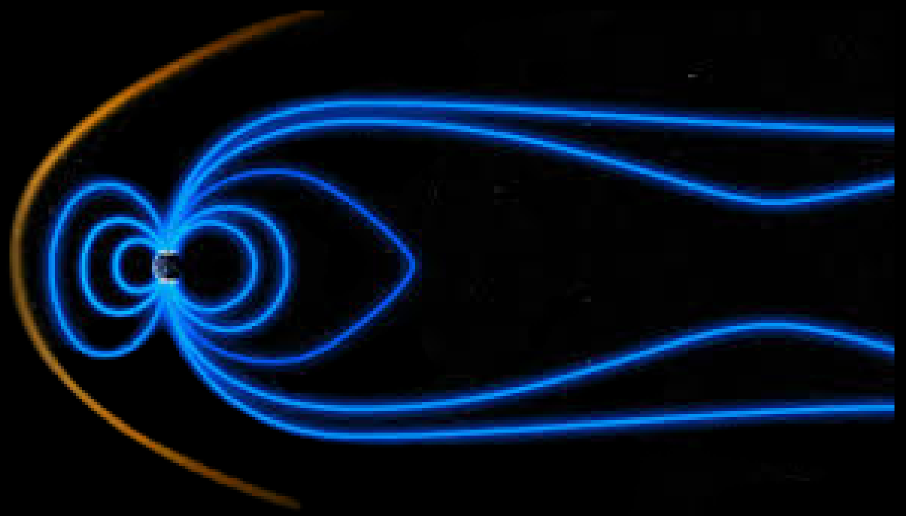
(Molina et al, 2020)

- Faster than background solar wing, but less speed than CMEs (typically)
- SLOW events - A CH can be present for more than a Sun rotation (~27 days), still difficult to predict

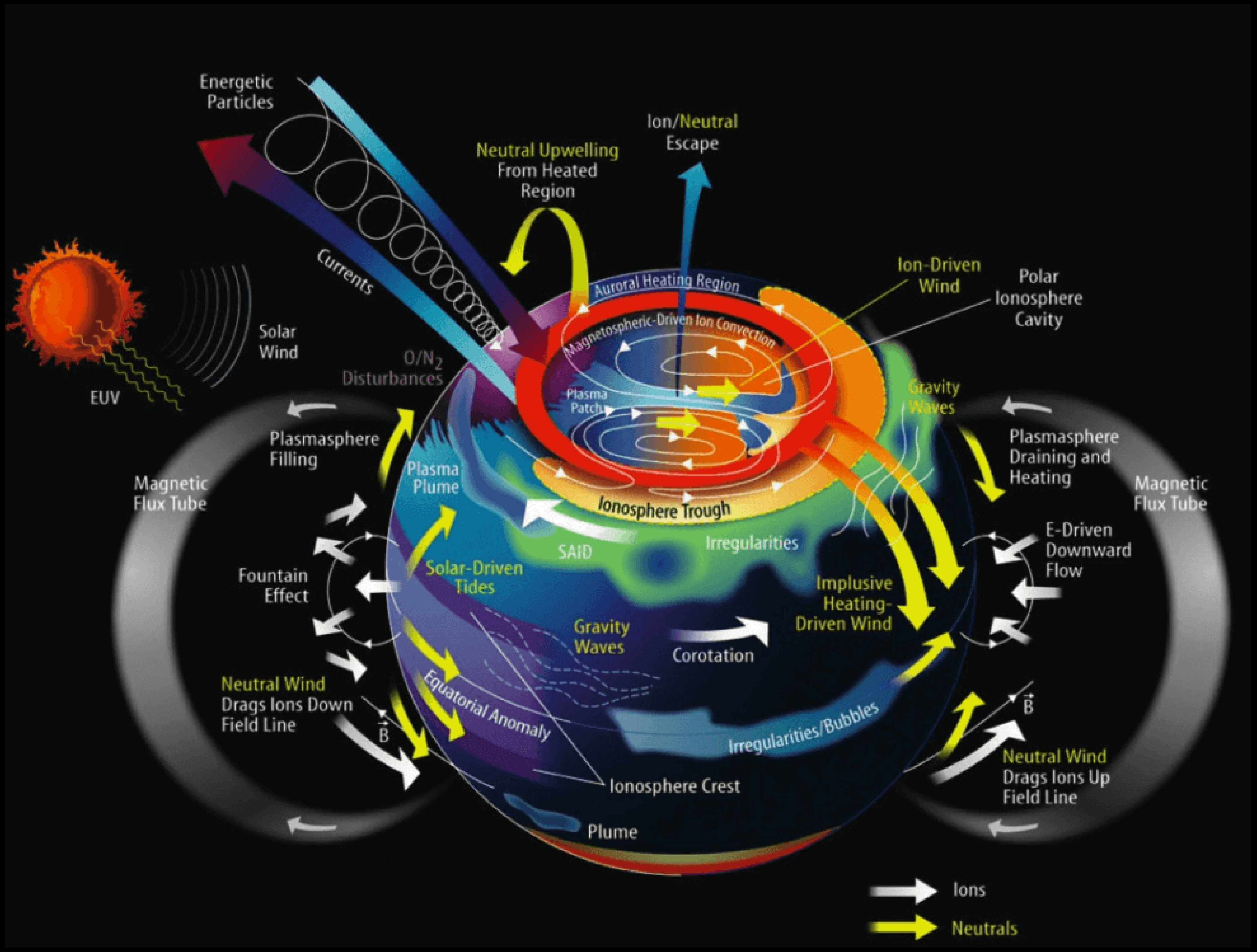
TSWC, 2022

Space Weather (SWx) Main sources & propagation - Observations - Consequences

Geomagnetic Storms:
Temporary disturbances of the Earth's magnetic field (coupling solar wind-magnetosphere through magnetic reconection)



Solar Wind - Magnetosphere - Ionosphere - Thermosphere coupling

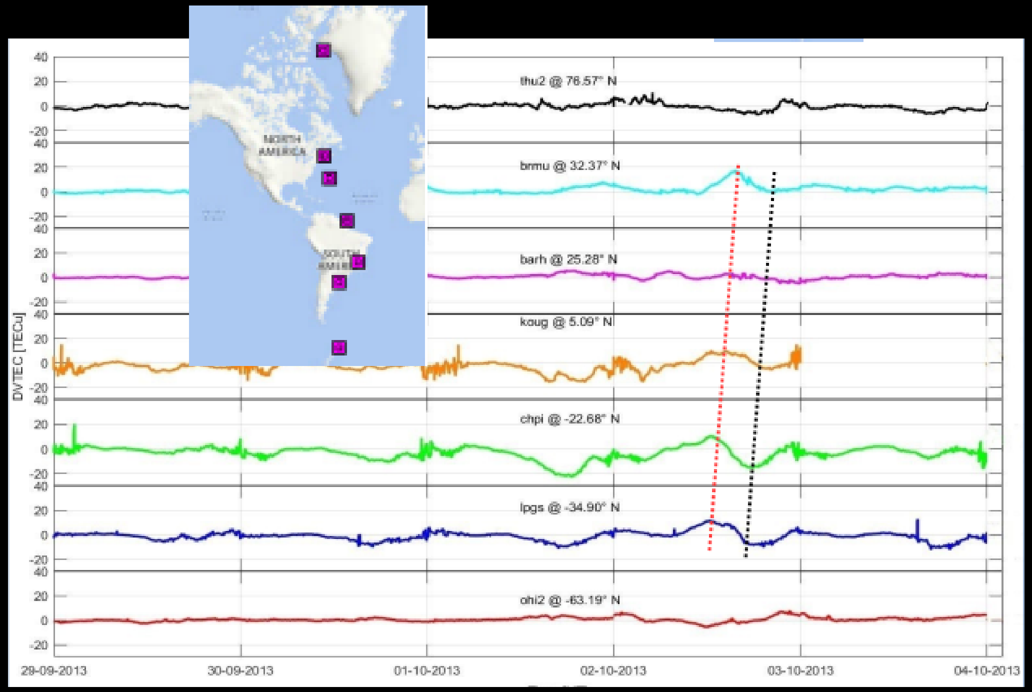


Space Weather (SWx) Main sources & propagation - Observations - Consequences

SW-M-I-T

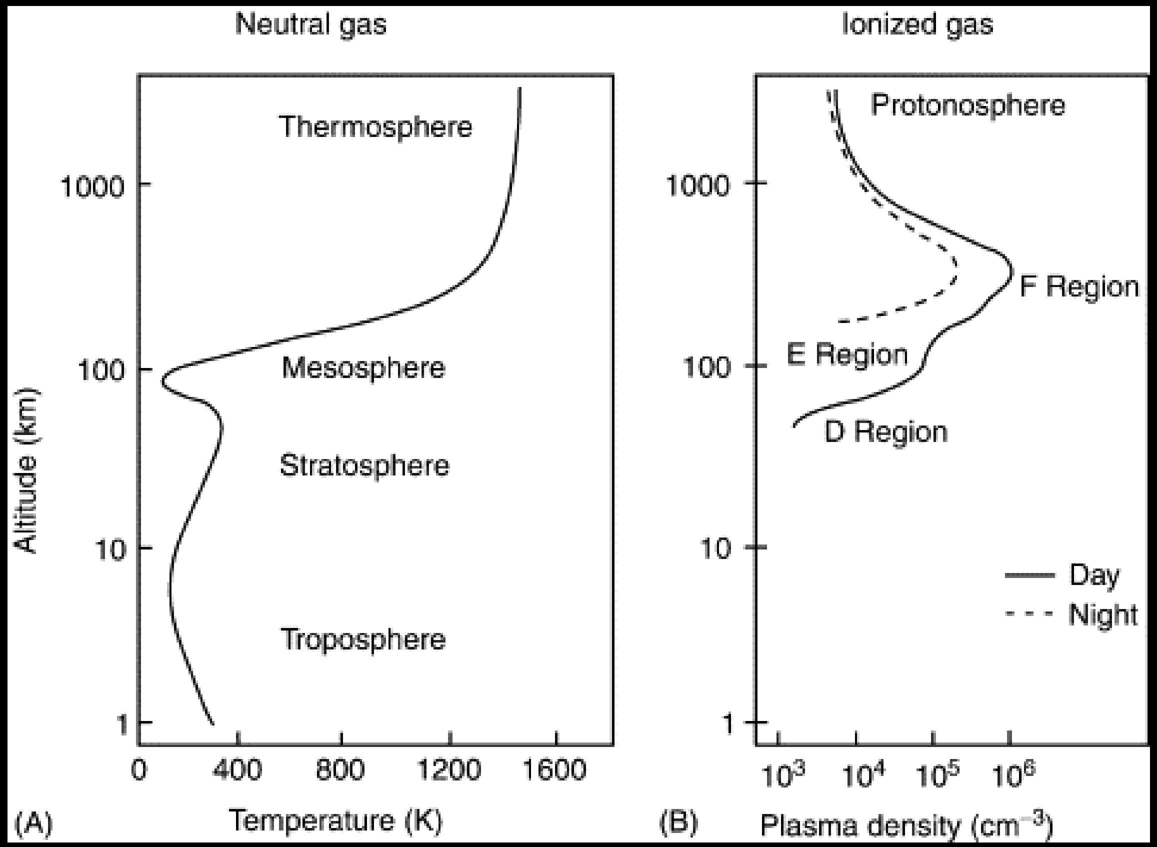
Geomagnetic field perturbations:
- Auroral lat: AU/AL/AE/AO indeces
- Sub-auroral lat: Kp
- Low lat: Dst
- Regional (South America): KSA

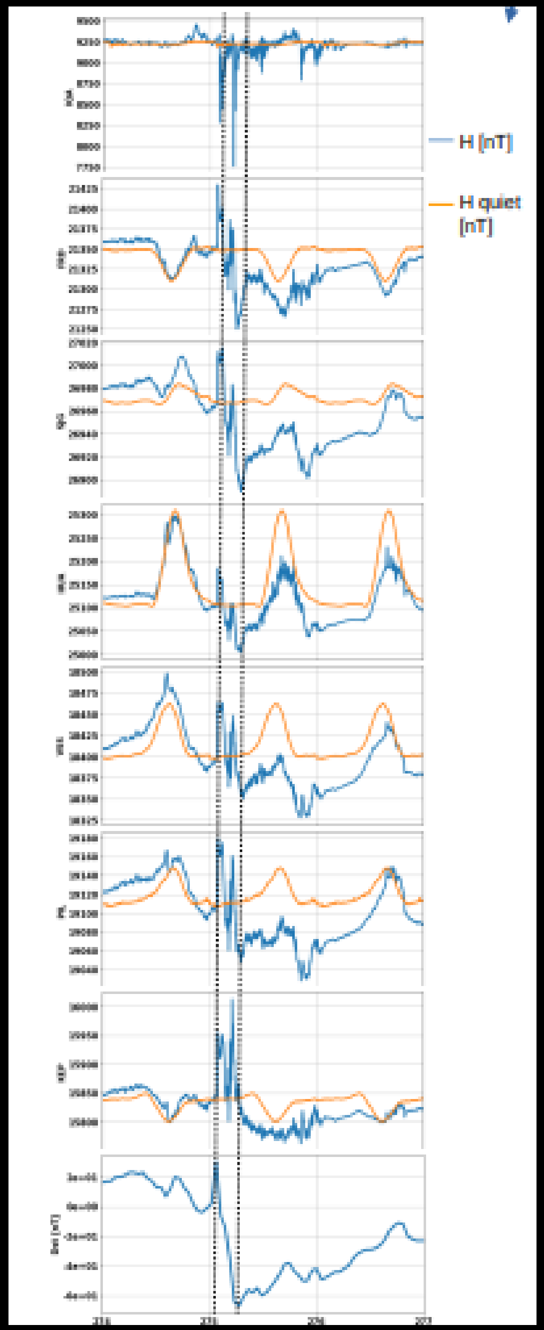
Particle precipitaion: enhancement at auroral latitudes



Ionospheric response to the geomagnetic storm on 2nd October 2013: Longitudinal chain analysis over the American sector. (Molina et al, 2018)

Ionosphere:
- Modification of the ionospheric current system ar high lat (auroral electroject) - PPEF
- Modification of the ionospheric current system at low lat (equatorial electroject)
- Ionospheric heating
- Travelling Ionospheric Disturbances
- (and more)





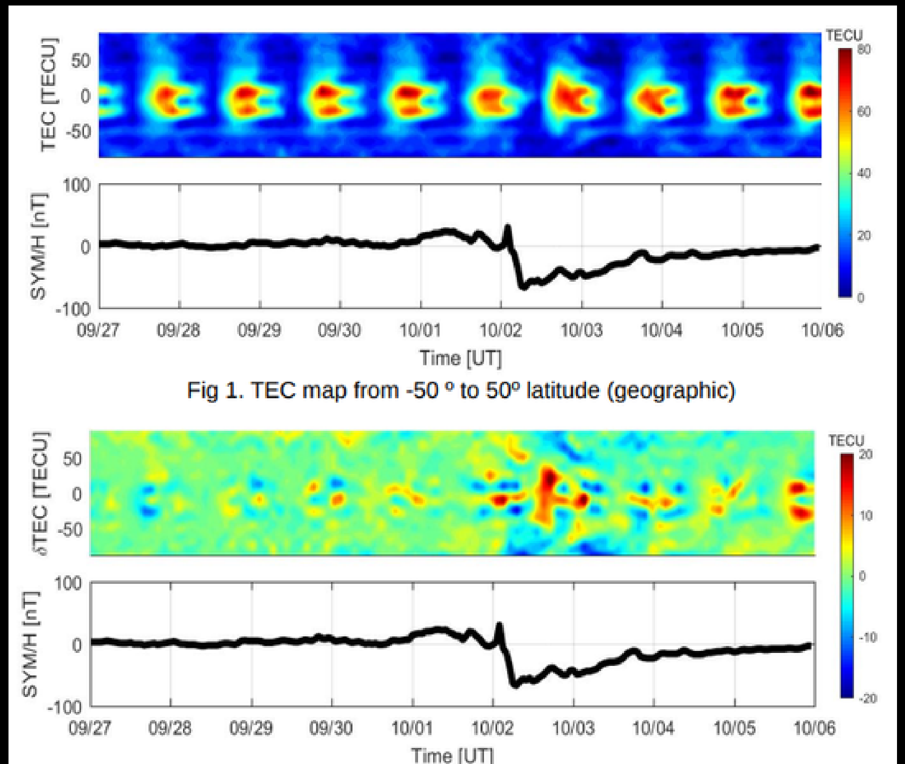


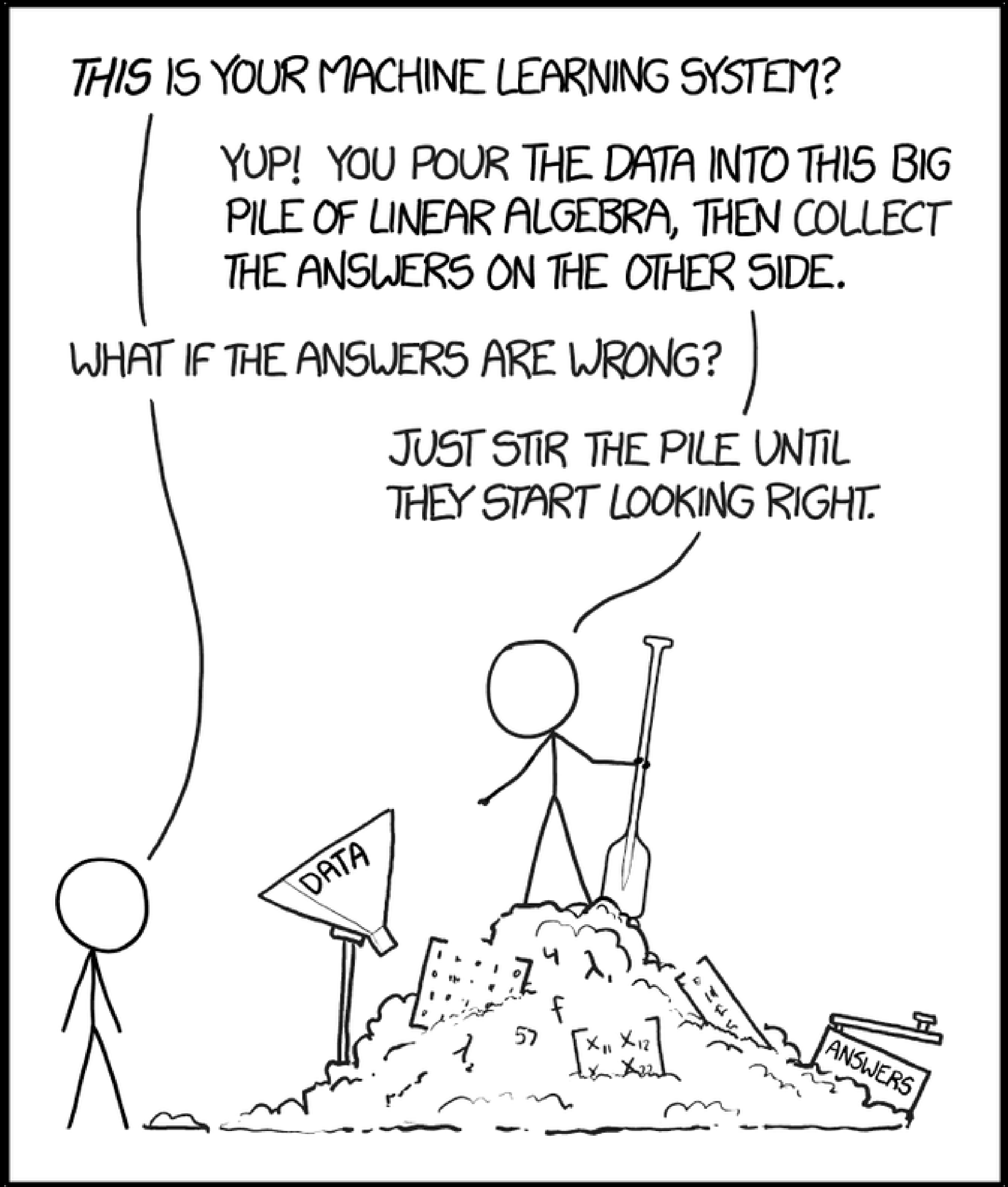
Fig 1. TEC map from -50 º to 50º latitude (geographic)

Fig 2. δTEC map from -50 º to 50º latitude (geographic)

Space Weather (SWx)

The data! just few thing to think about ...

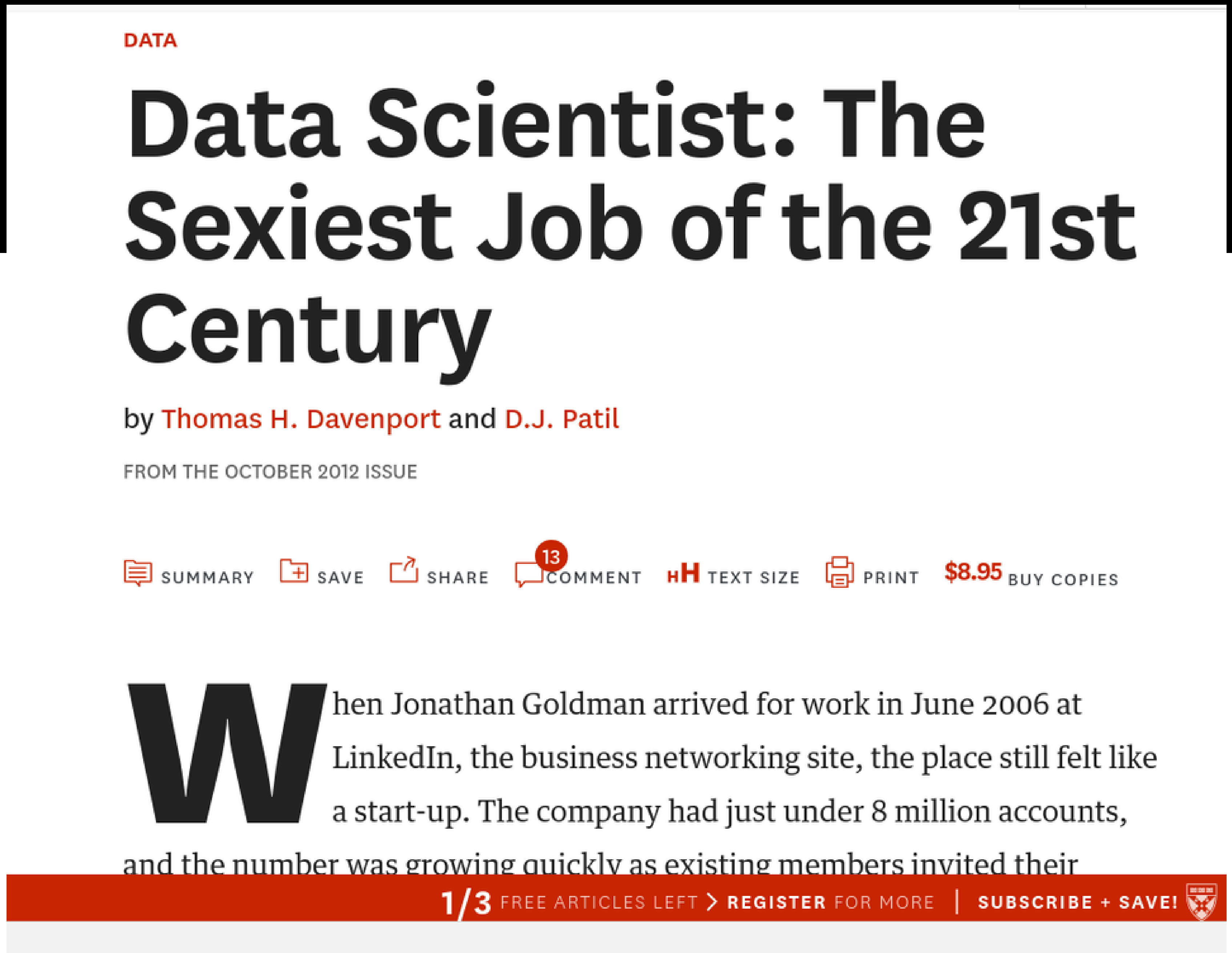
- Heterogenous Data
- Huge amount of data
- Coverage: partially observe a portion of the problem
- Data availability (remote data, military data, etc)
- Data quality: high quality = science; less quality = operations; levels of pre-processing
- Formating madness! resolution madness!
- Not straight-forward to understand
- Produced by instruments, interpreters, simulations or models, metadata
- Storage!
- Complex problems, complex data
- Data infrastructure: data model, a model for data management, hardware
- Intrinsically unbalanced data
- Scales!!!!



DATA SCIENCE

Data Science

- Interdisciplinary field
- Extract knowledge from data
- Transform data into knowledge/information
- Aids decision making



DATA

Data Scientist: The Sexiest Job of the 21st Century

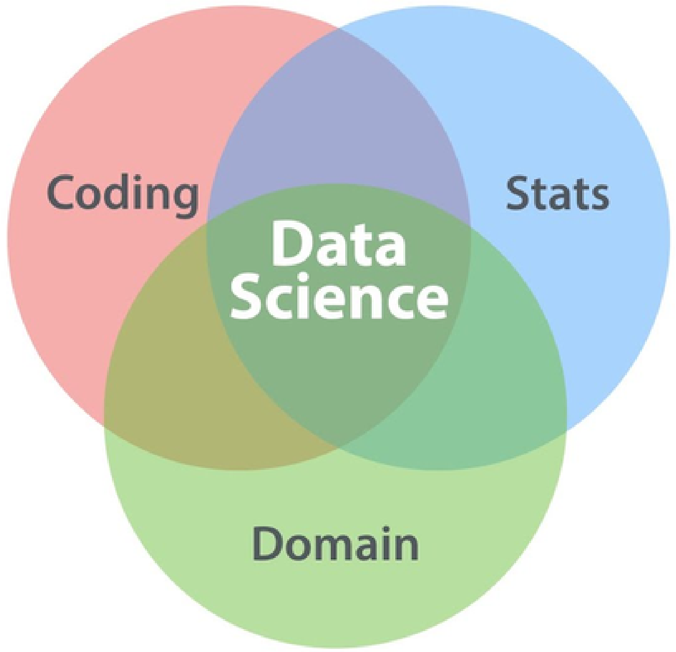
by Thomas H. Davenport and D.J. Patil

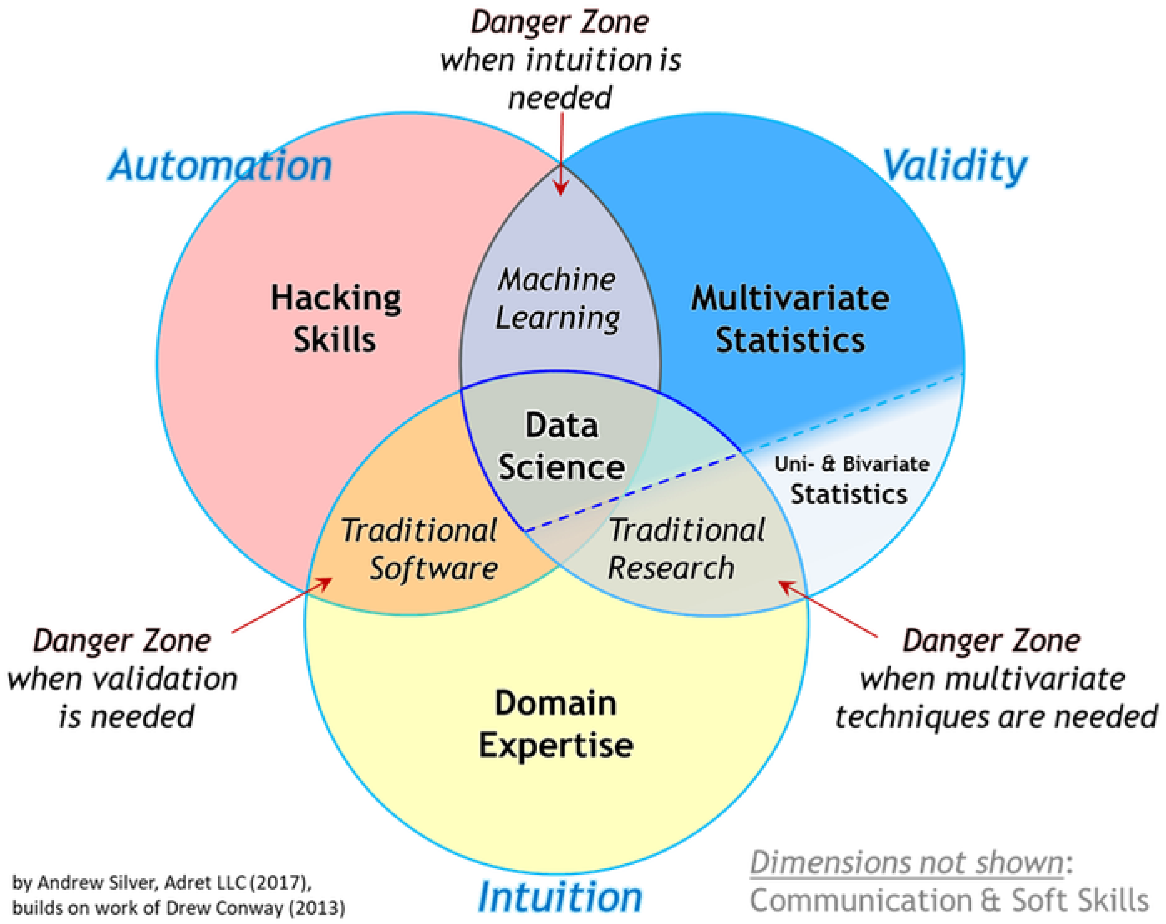
FROM THE OCTOBER 2012 ISSUE

SUMMARY SAVE SHARE 13 COMMENT TEXT SIZE PRINT $8.95 BUY COPIES

When Jonathan Goldman arrived for work in June 2006 at LinkedIn, the business networking site, the place still felt like a start-up. The company had just under 8 million accounts, and the number was growing quickly as existing members invited their

1/3 FREE ARTICLES LEFT › REGISTER FOR MORE | SUBSCRIBE + SAVE!





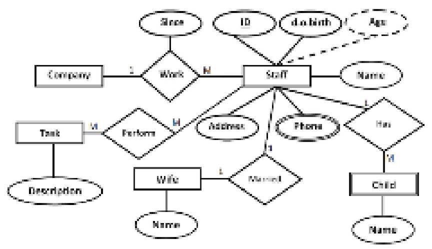
TSWC, 2022

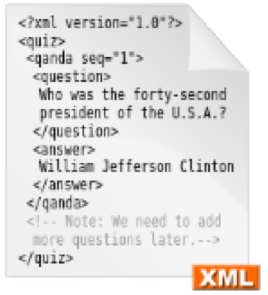
Data Science

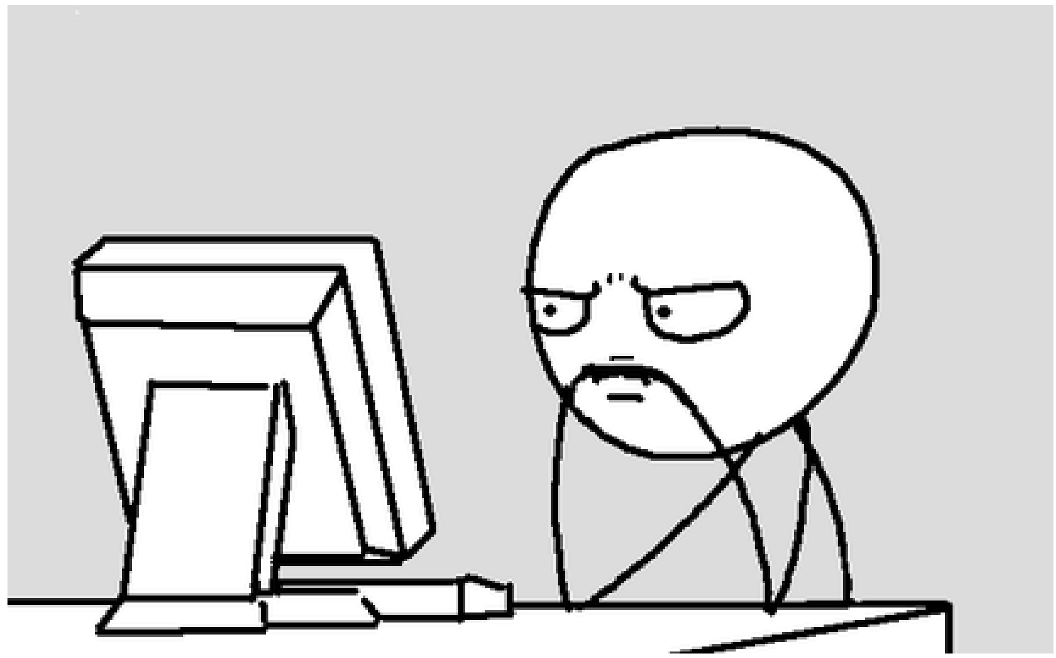
KNOWLEDGE & SKILLS

- Experience (domain)
- **Methods (e.g. ML)**
- **Programming Lenguage + tools + libraries + data formats**
- Data access / transformation
- Data storage
- Visualization tools
- Google skills!

https://spase-group.org/data/

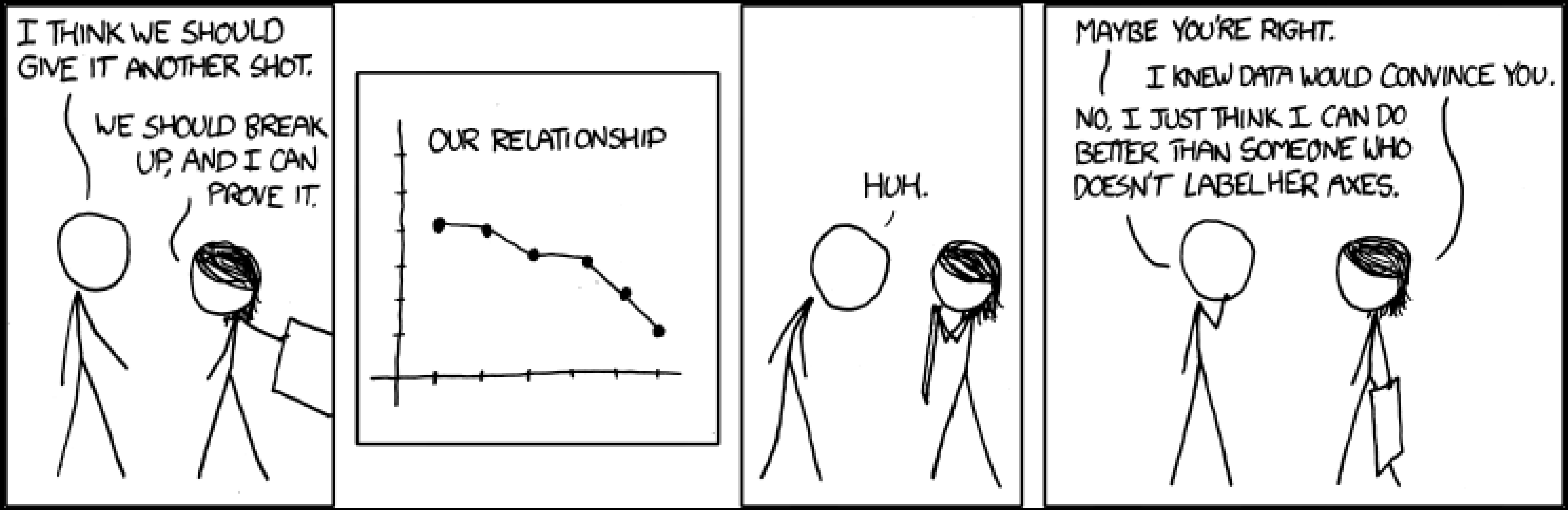






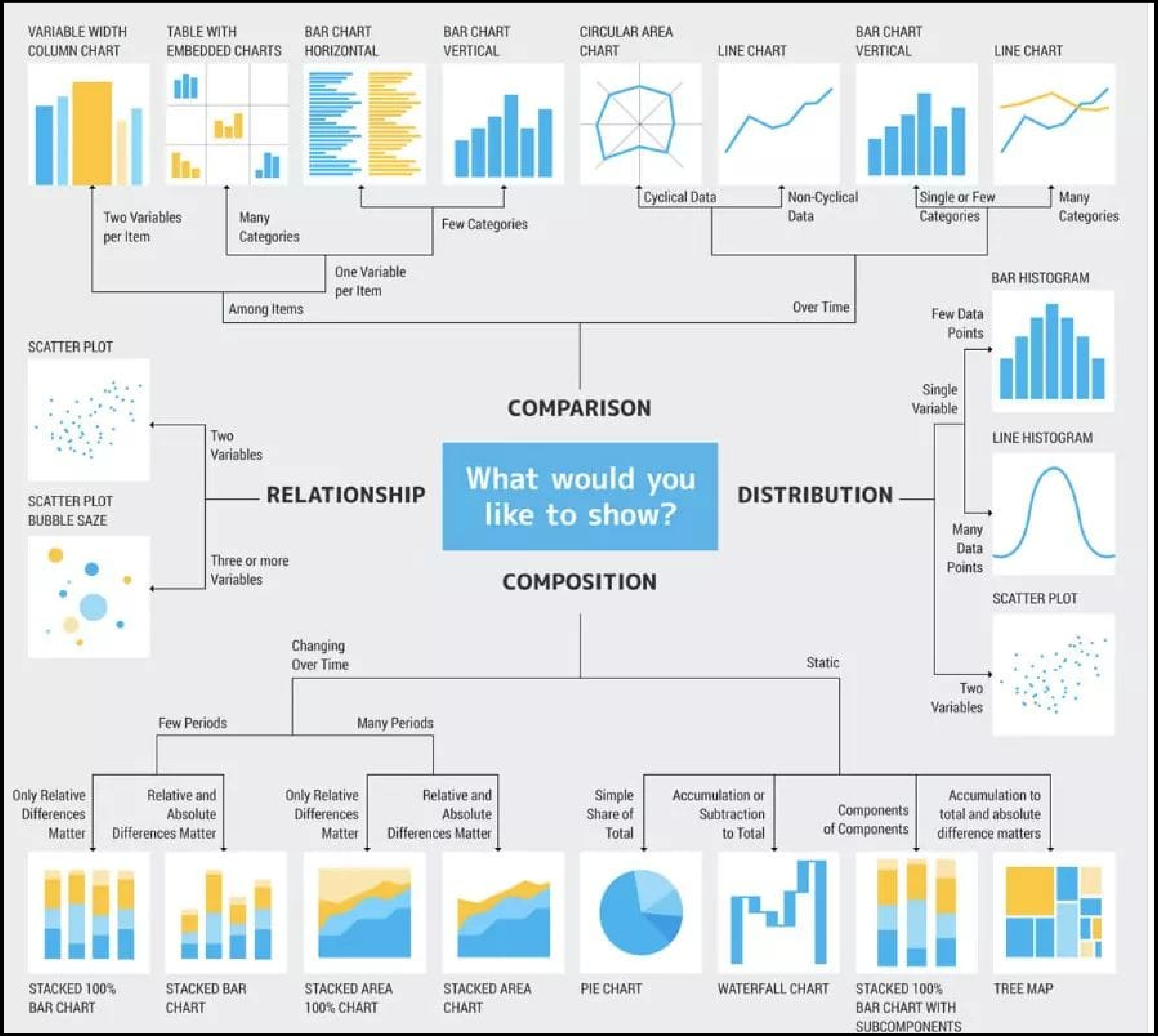
KNOWLEDGE & SKILLS

- **Visualization tools**
- **Google skills!**
    - https://www.kaggle.com/
    - https://es.stackoverflow.com/

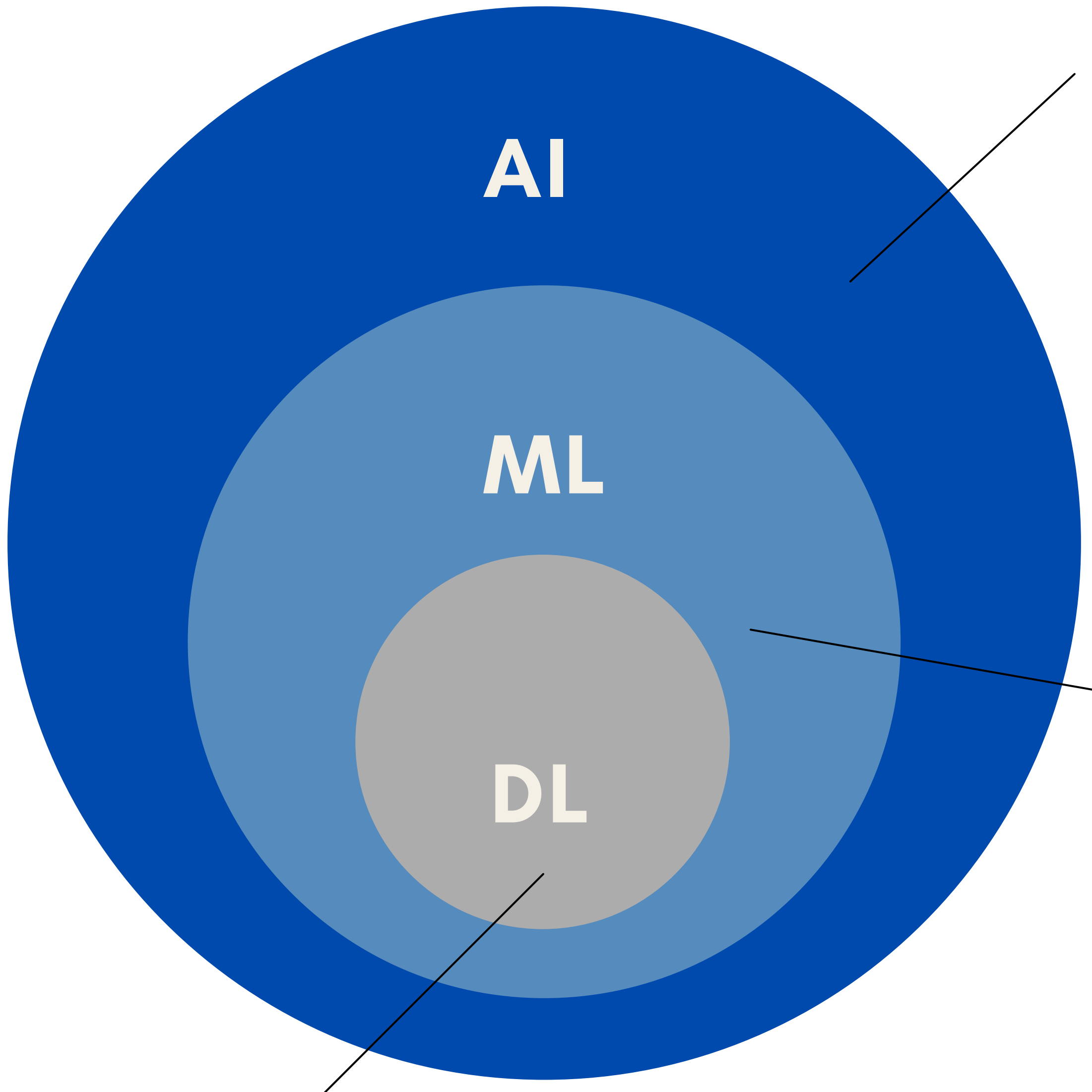


KNOWLEDGE & SKILLS

- **Visualization tools**
- **Google skills!**
  - https://www.kaggle.com/
  - https://es.stackoverflow.com/



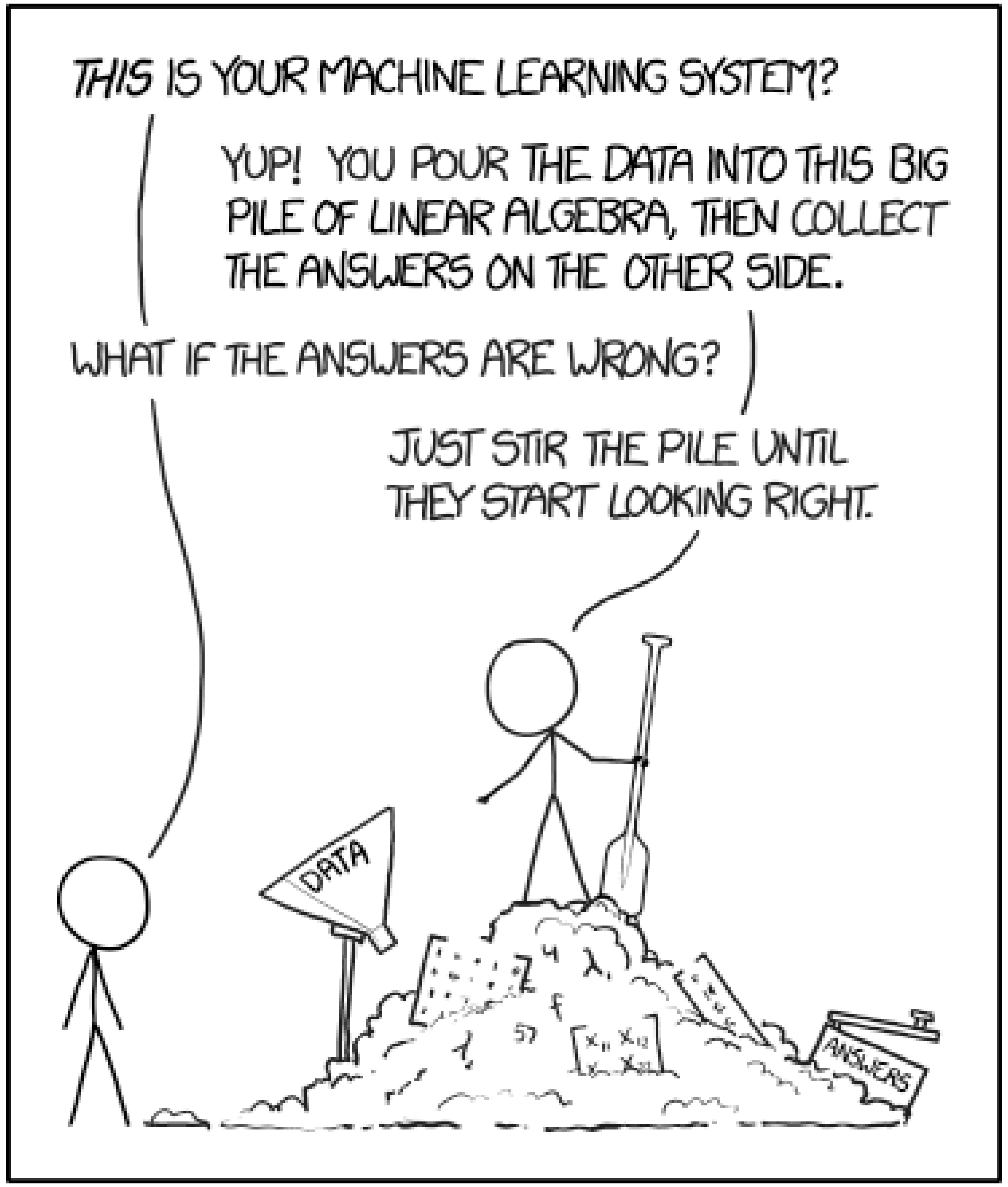
TSWC, 2022



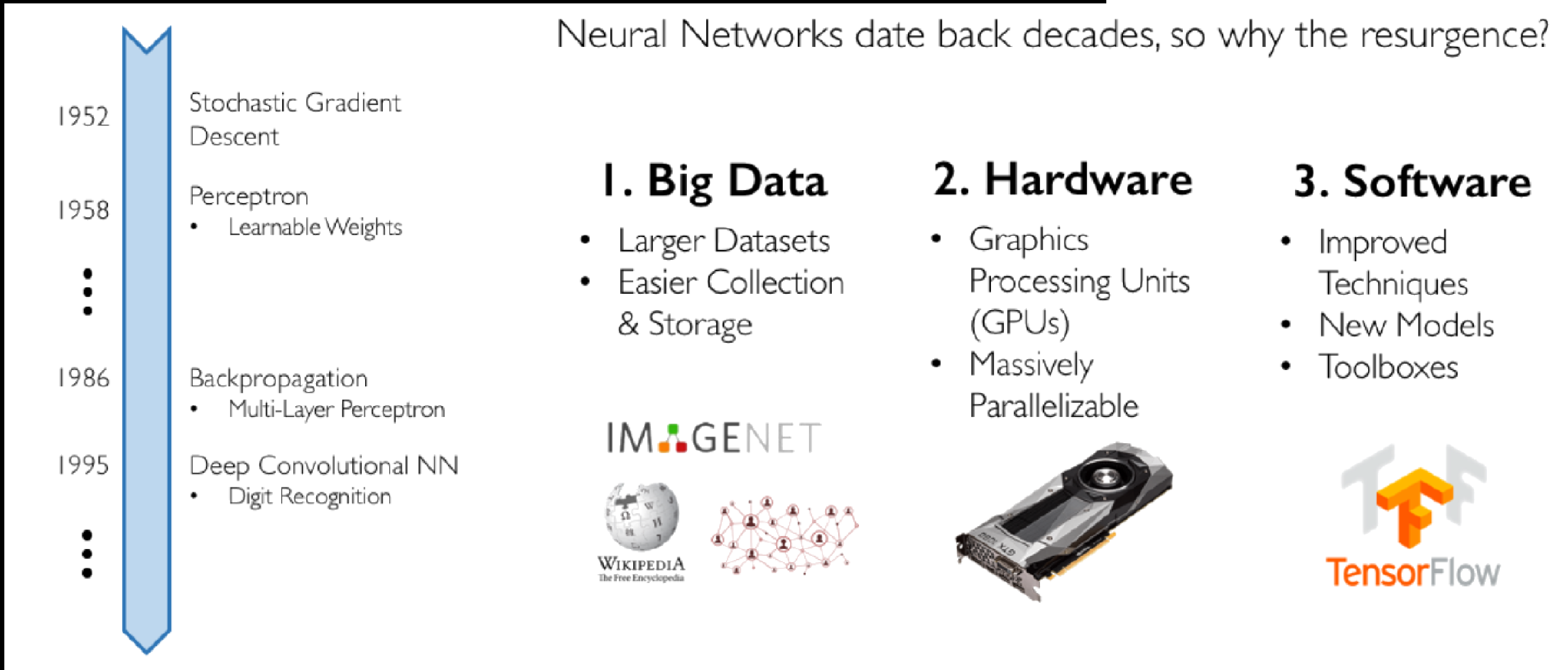
AI = area of computer science that emphasizes the creation of intelligent machines that work and react like humans.

ML = method of data analysis that automates data model building. ML uses algorithms that learn from data and can find insights without explicit programming.

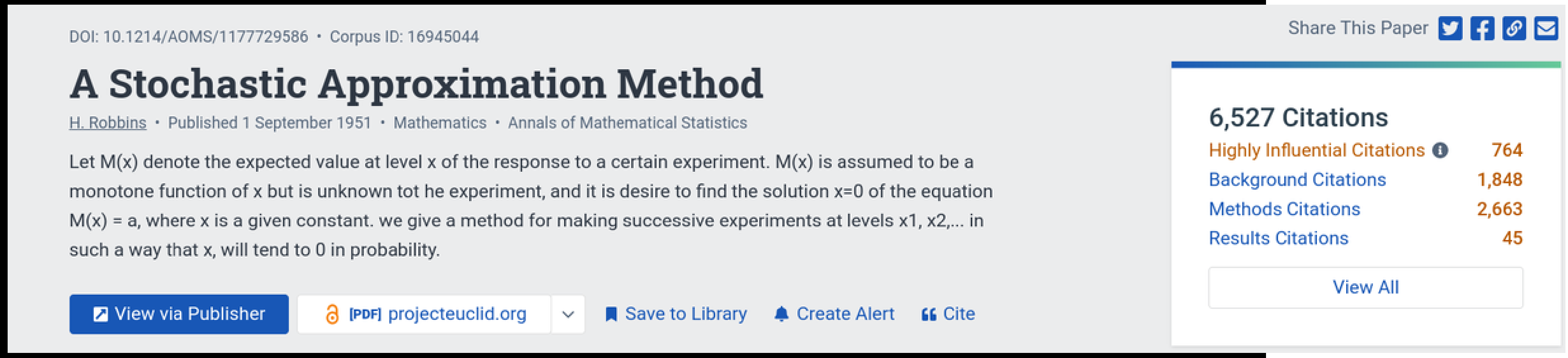
DL = a subset of ML and refers to artificial neural networks that are composed of many layers.



ML (DL) WHY NOW?



+ more mature algorithms
+ applications
+ bigger community



DOI: 10.1214/AOMS/1177729586 • Corpus ID: 16945044

A Stochastic Approximation Method

H. Robbins • Published 1 September 1951 • Mathematics • Annals of Mathematical Statistics

Let M(x) denote the expected value at level x of the response to a certain experiment. M(x) is assumed to be a monotone function of x but is unknown tot he experiment, and it is desire to find the solution x=0 of the equation M(x) = a, where x is a given constant. we give a method for making successive experiments at levels x1, x2,... in such a way that x, will tend to 0 in probability.

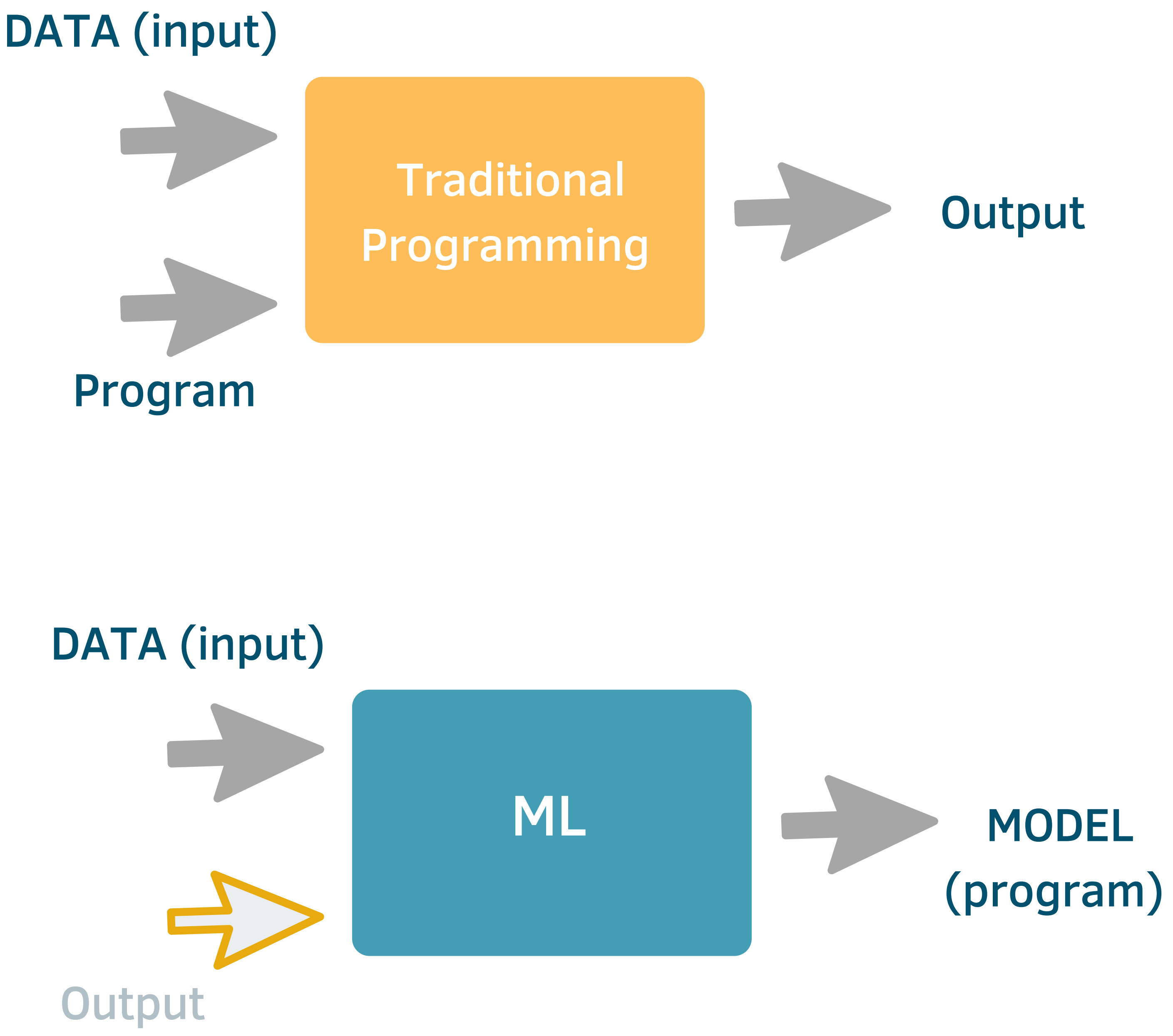
View via Publisher | [PDF] projecteuclid.org | Save to Library | Create Alert | Cite

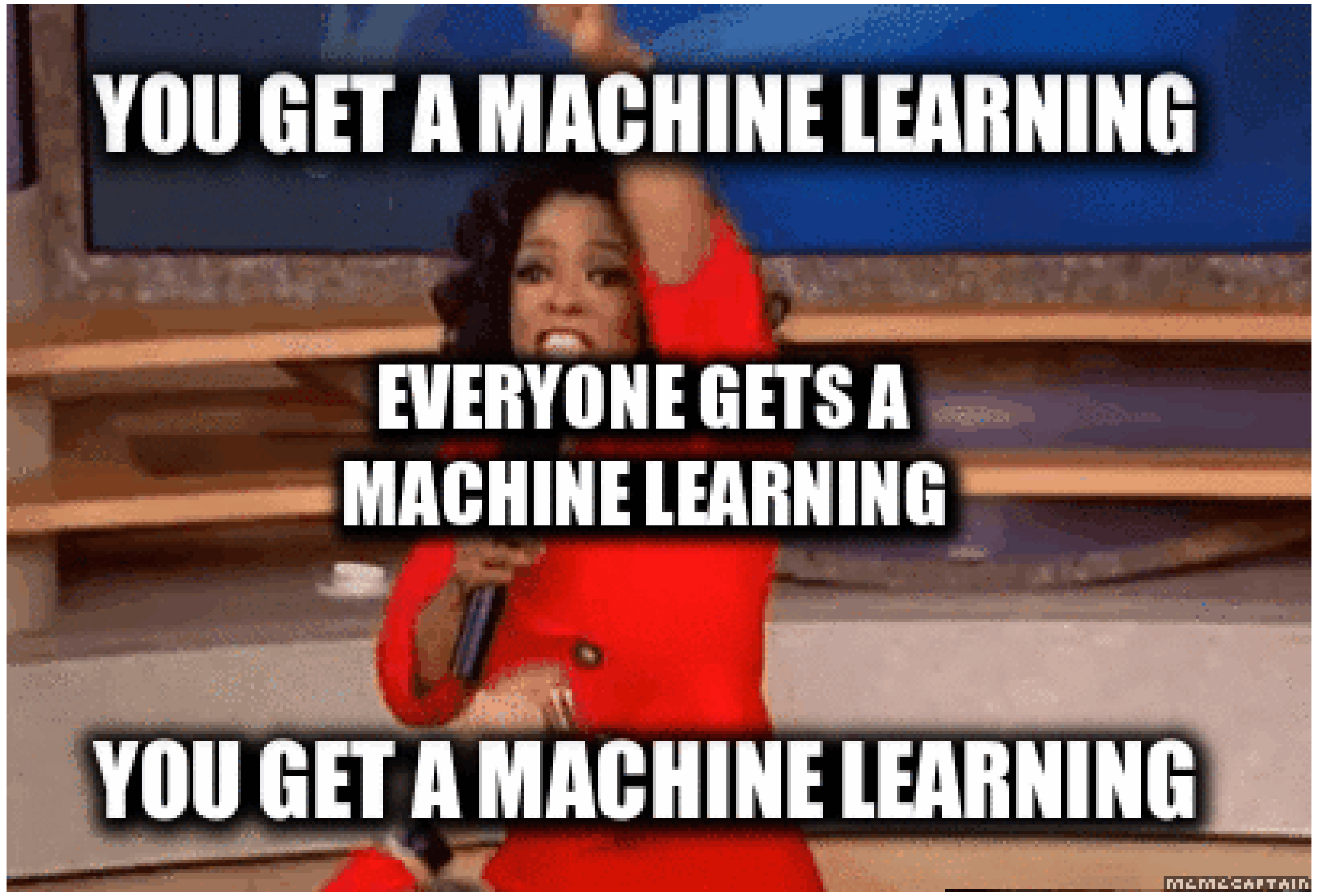
Share This Paper

6,527 Citations	
Highly Influential Citations	764
Background Citations	1,848
Methods Citations	2,663
Results Citations	45

View All

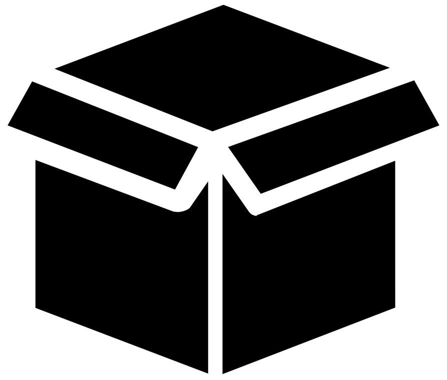
ML (DL) WHY NOW?


DATA (input)
Traditional Programming
Output
Program
DATA (input)
ML
MODEL (program)
Output


YOU GET A MACHINE LEARNING
EVERYONE GETS A MACHINE LEARNING
YOU GET A MACHINE LEARNING

Data-driven modeling

data
+
hyperparameters

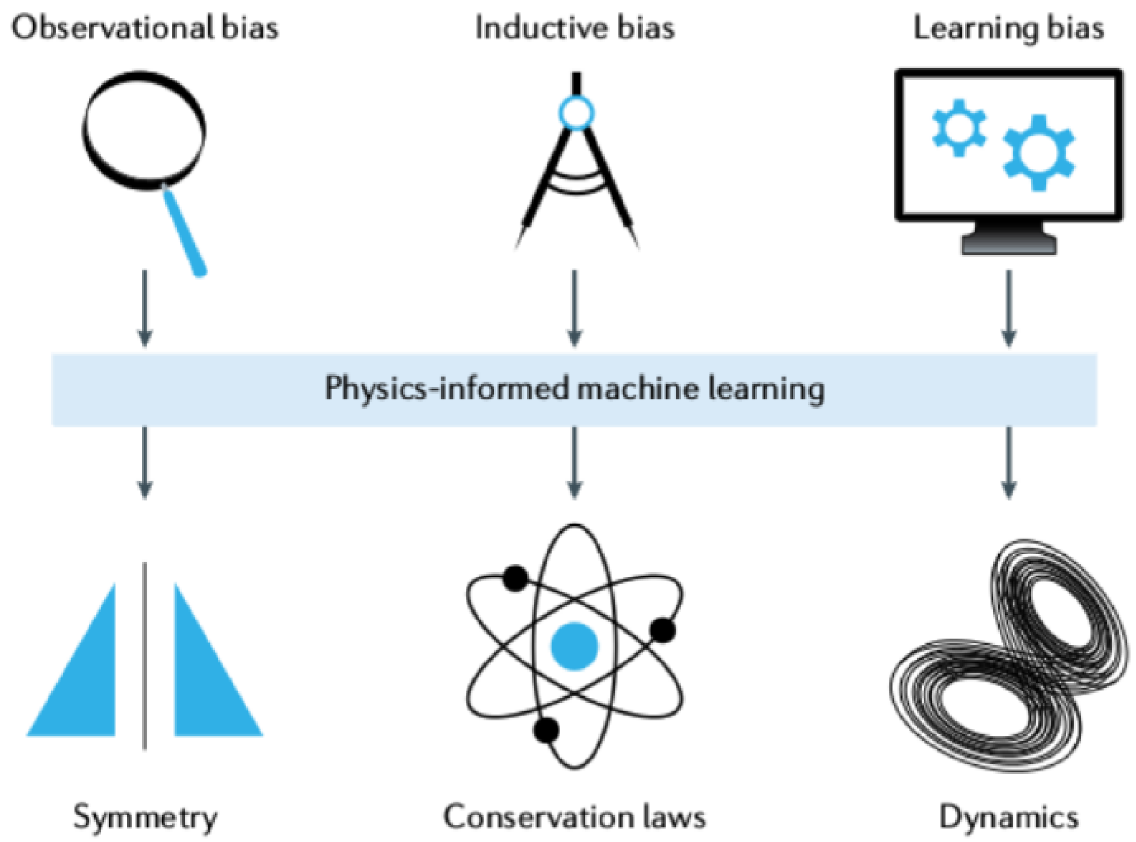


model (parameters)

data
+
hyperparameters
+
physics/constraints/domain knowledge

model
(parameters)

- Physical consistency (definitions, conservation laws…)
- Ability to generalize outside of the training set
- Interpretability
- Stability
- Data limitations



goal = enhance generalization

Physics-informed learning = process by which prior knowledge stemming from our observational, empirical, physical or mathematical understanding of the world can be leveraged to improve the performance of a learning algorithm

Karniadakiset al. Physics-informed machine learning. Nat Rev Phys (2021).

TSWC, 2022

Data-driven modeling

- Should we give a chance to the black box?

data
+
hyperparameters

model
(parameters)

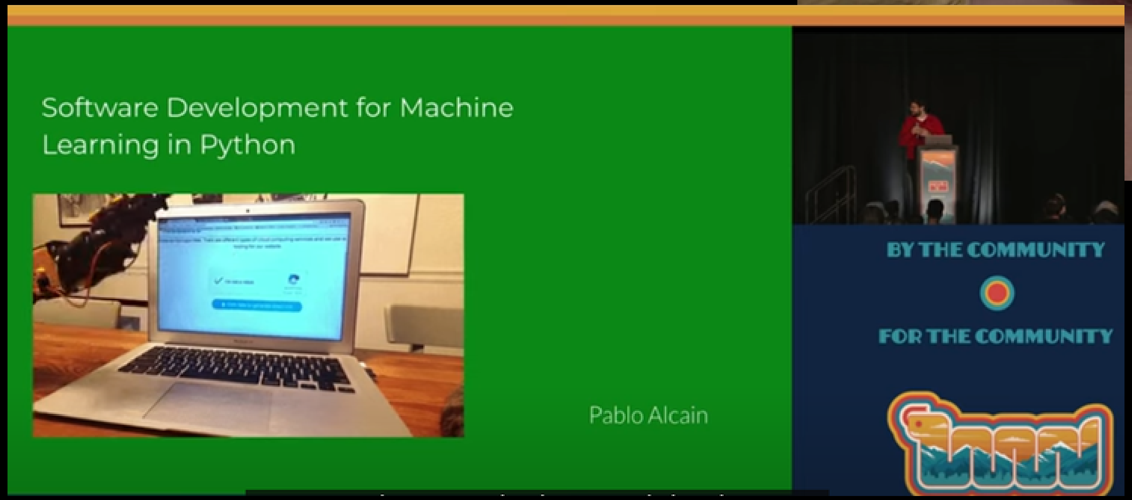
TSWC, 2022

Scientific programming

WTPC | WORKSHOP EN TÉCNICAS DE PROGRAMACIÓN CIENTÍFICA

coming soon!

https://wtpc.github.io/



Python ?)

Programming Language Hall of Fame

Year	Winner
2021	Python
2020	Python
2019	C
2018	Python
2017	C
2016	Go
2015	Java
2014	JavaScript
2013	Transact-SQL
2012	Objective-C
2011	Objective-C
2010	Python

https://www.tiobe.com/tiobe-index/
(July 2022)

2022 Developer Survey

In May 2022 over 70,000 developers told us how they learn and level up, which tools they're using, and what they want.

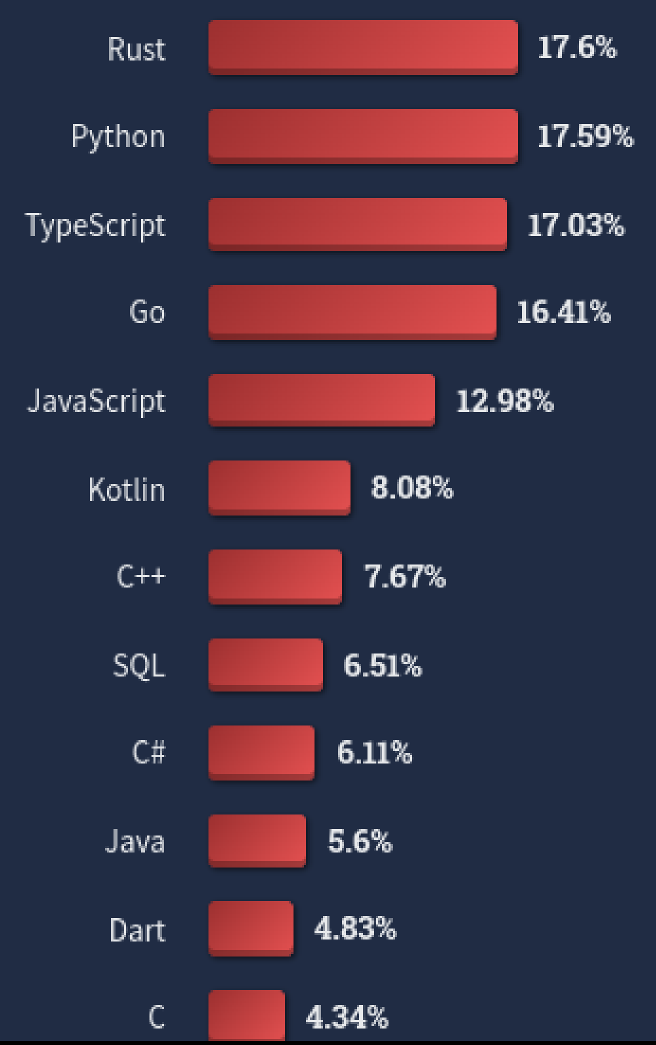
Read the overview → Methodology →

Programming, scripting, and markup languages

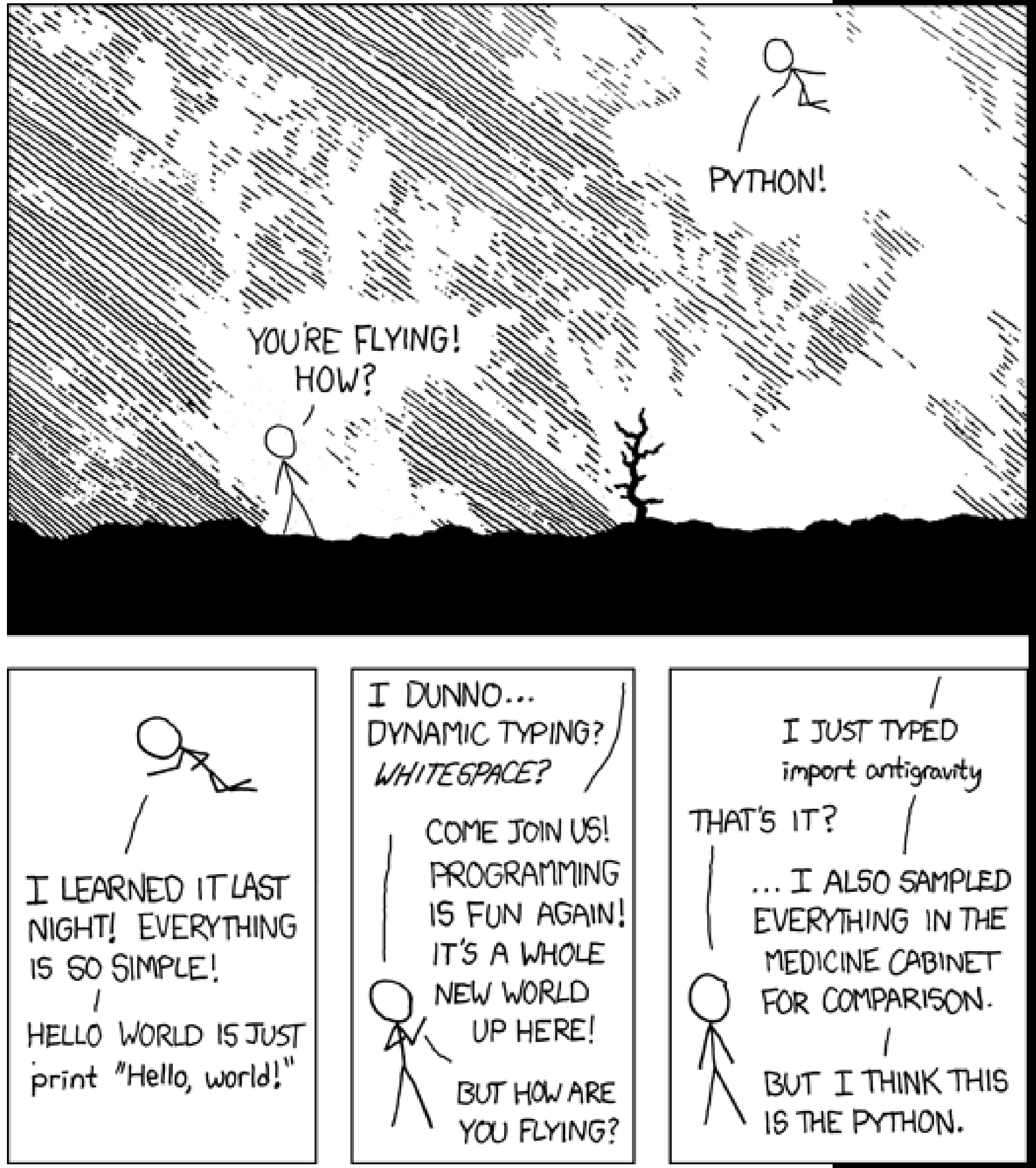
Rust is on its seventh year as the most loved language with 87% of developers saying they want to continue using it.

Rust also ties with Python as the most wanted technology with TypeScript running a close second.

Loved vs. Dreaded Want



TSWC, 2022

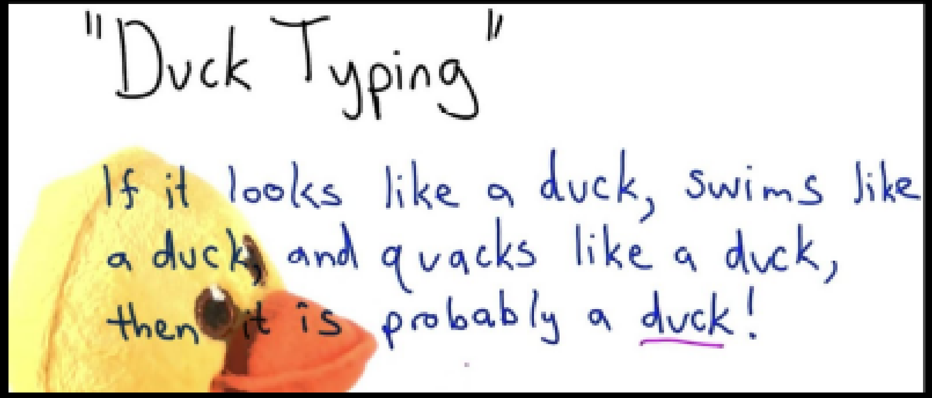


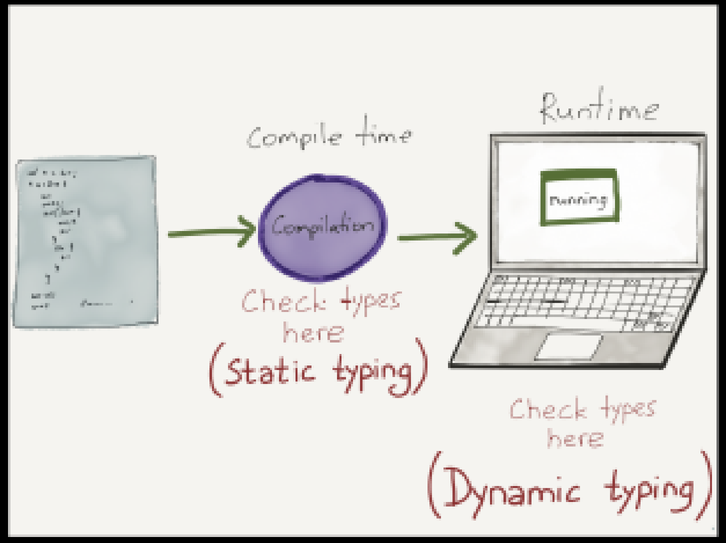
- Todo son objetos (de alguna clase)
- Fuertemente tipado
- Tipado dinámico
- Duck typing

Objeto	.	atributo
Objeto	.	método

"hola" + 5 → da error

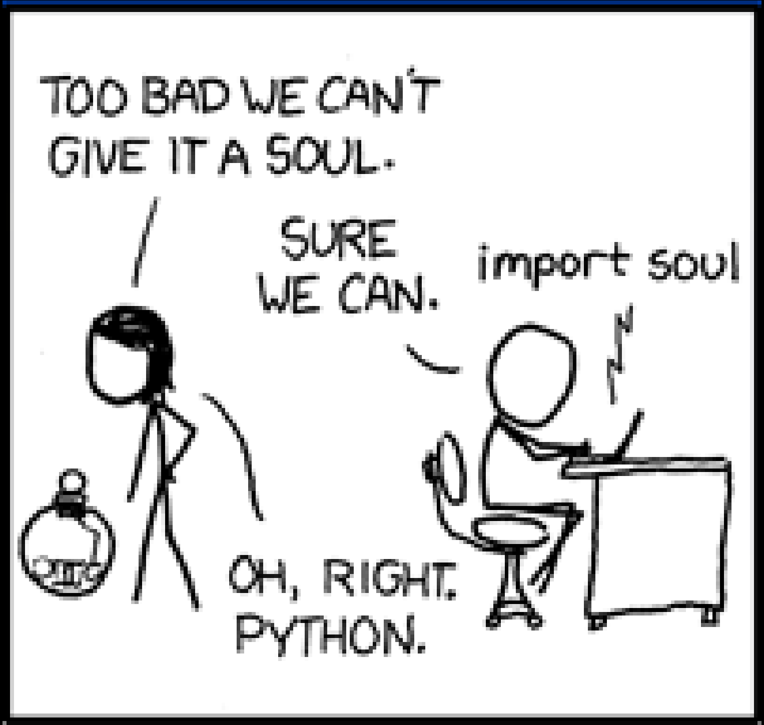
```
def suma(a,b):
    return a+b
```





a="hola"
a=5
a=a*2

Espacios de nombres (namespaces)
y Módulos



Python

NumPy pandas

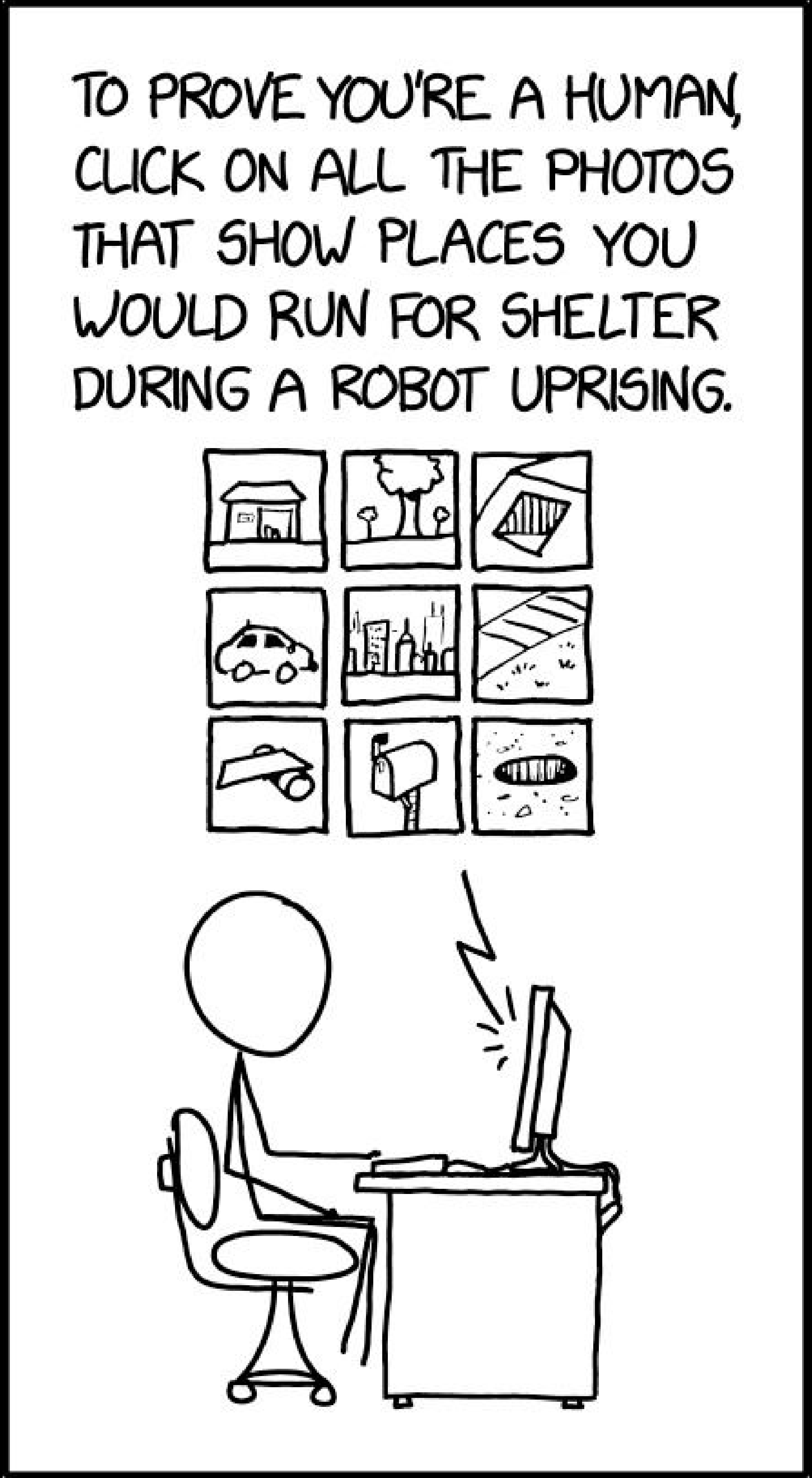
TSWC, 2022

HANDS-ON LAB

TP - 0

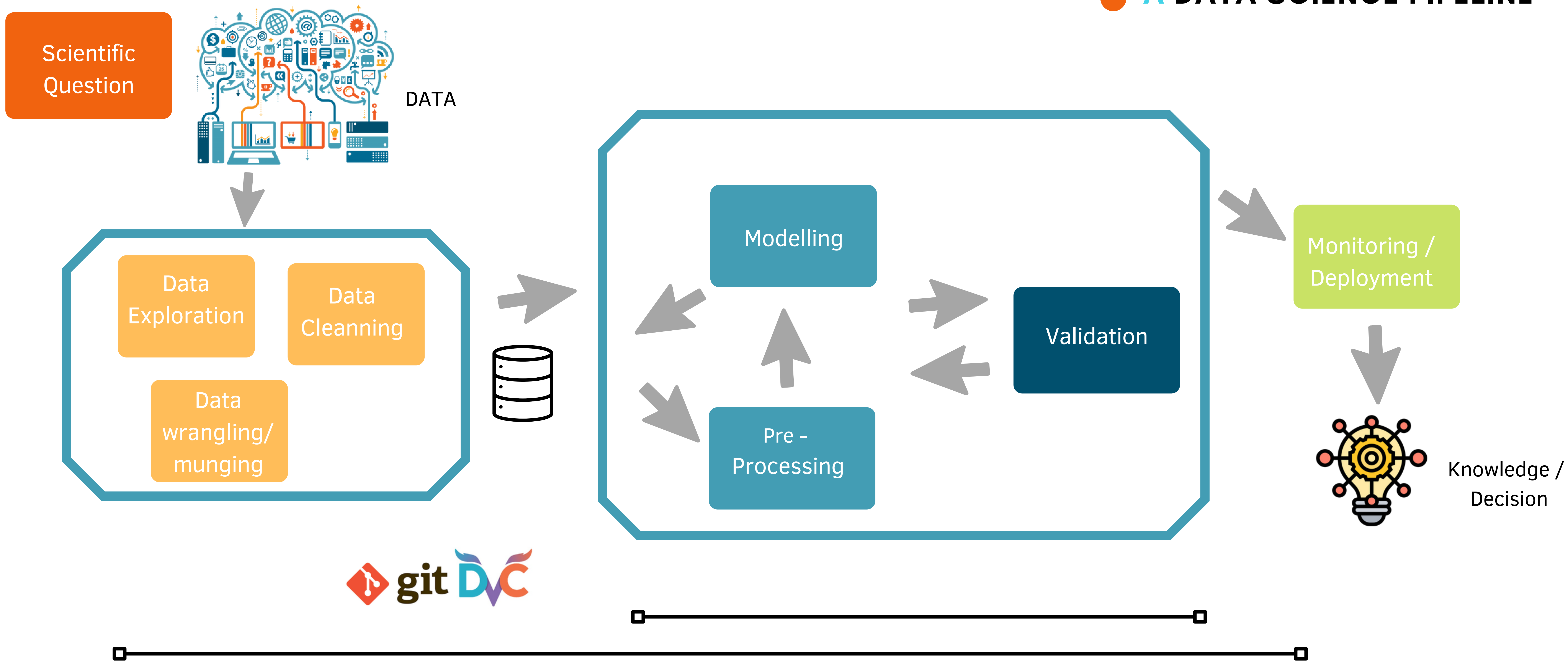
- Google Colabs - Jupyter Notebooks
- Codes & Data access
- Python refresh (numpy + Pandas)
- sklearn - Playing with a dataset (splitting and modeling)

Machine Learning



Data-driven model

A DATA SCIENCE PIPELINE



General TSWC pipeline

Data acquisition

Local Sources

External Sources

FTP

RAW

TSWC server

debian

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

TensorFlow + Keras

Modelling and Processing

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

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

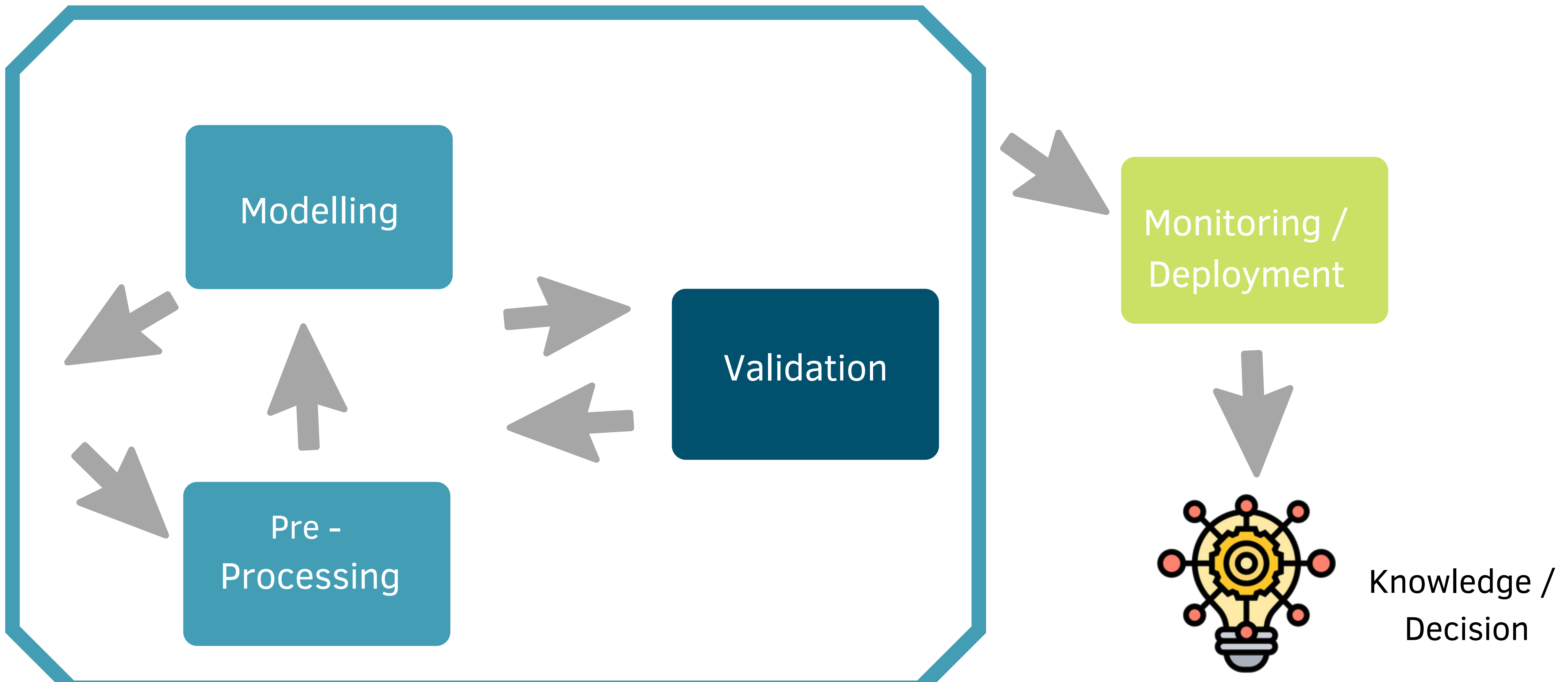
https://spaceweather.facet.unt.edu.ar/

Server-internal process

Namour et al
URSI-GASS, 2021

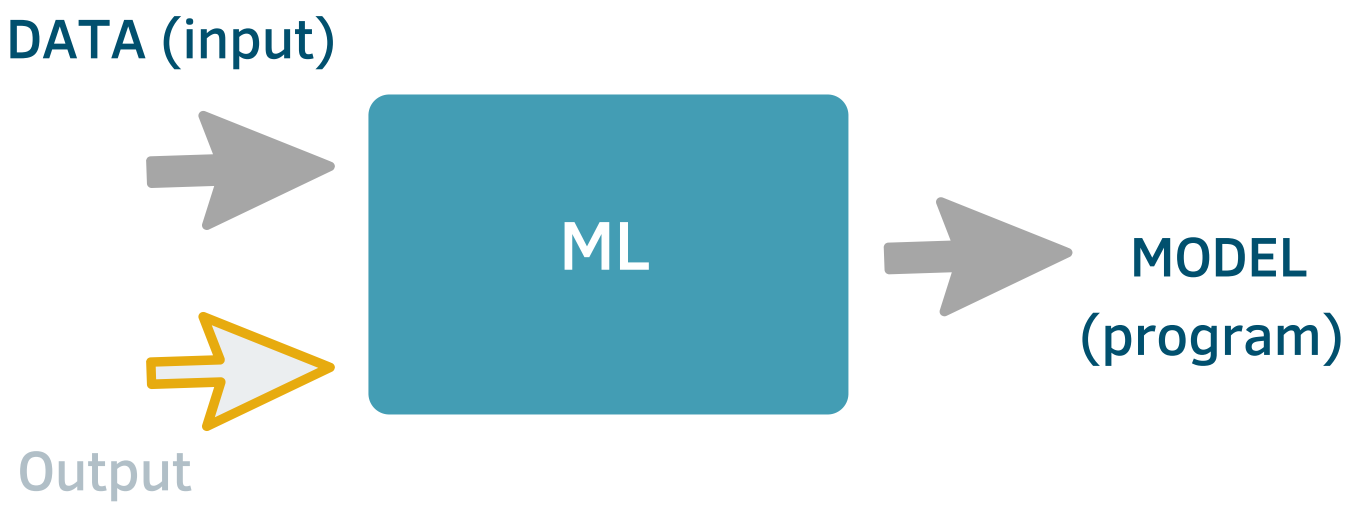
TSWC, 2022

ML pipeline

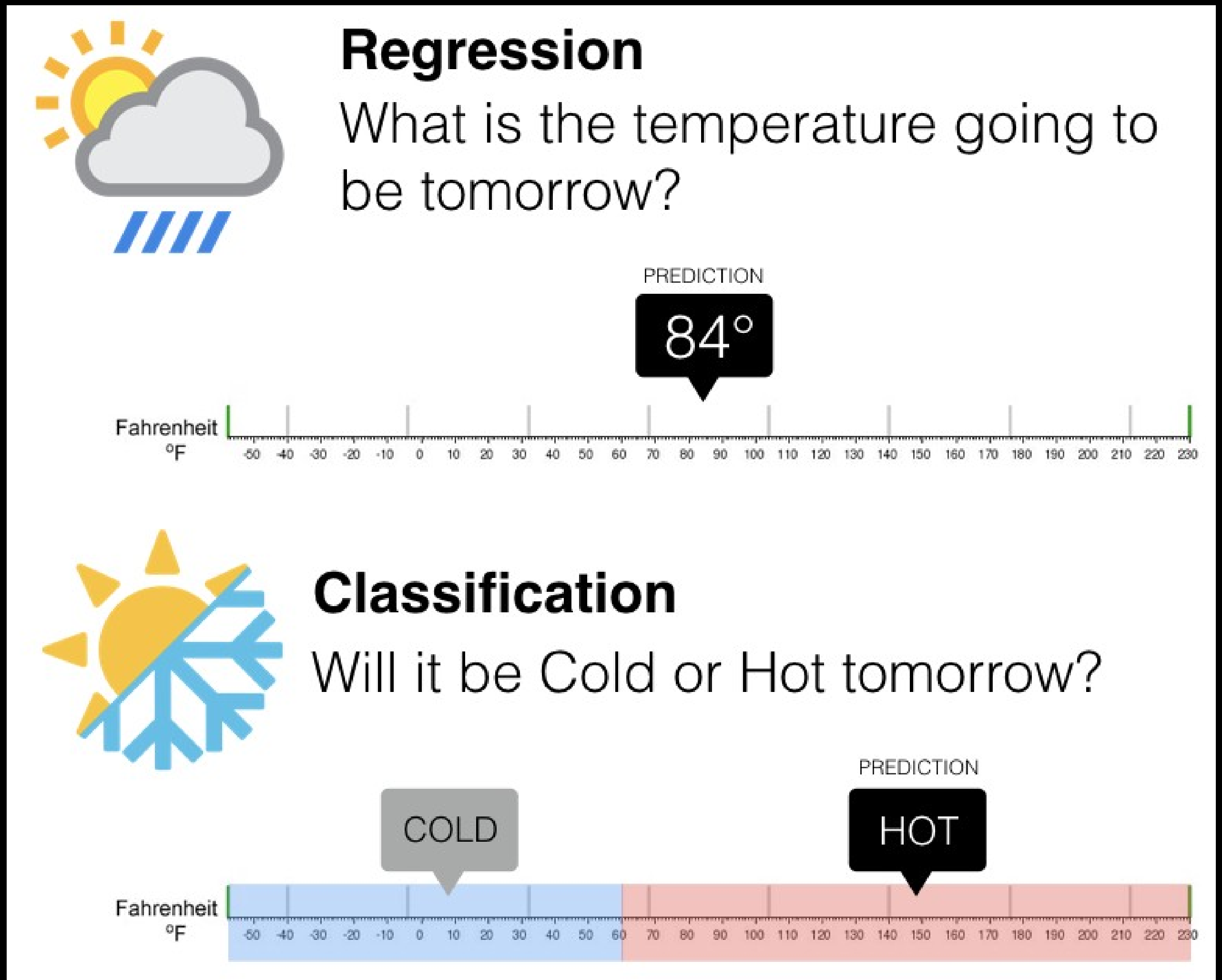


What /how to ask?

How is the answer?



PROBLEMS (TYPES)

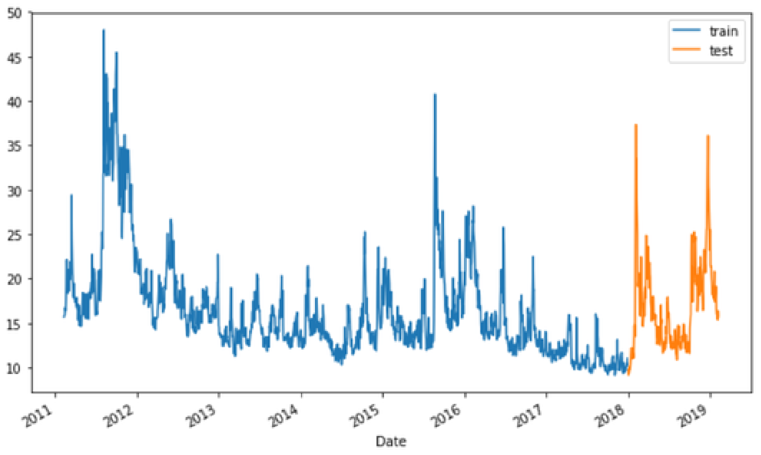


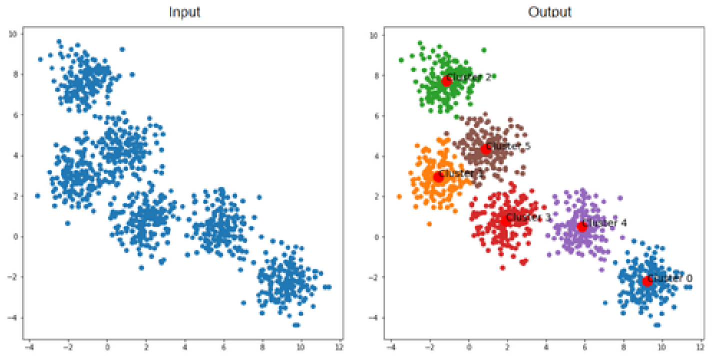
Regression: deals with predicting a continuous value

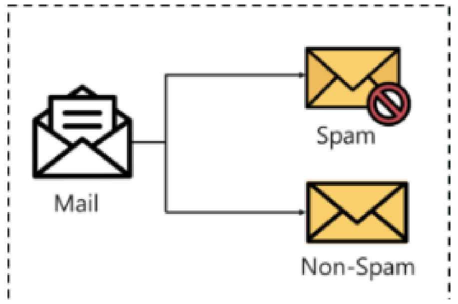
Classification: to predict output from a set of finite categorical values

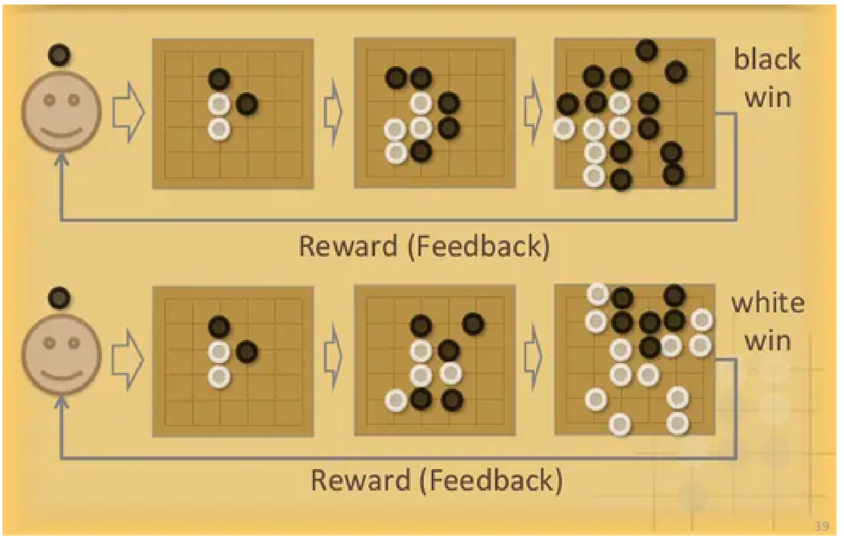
LEARNING (TYPES)

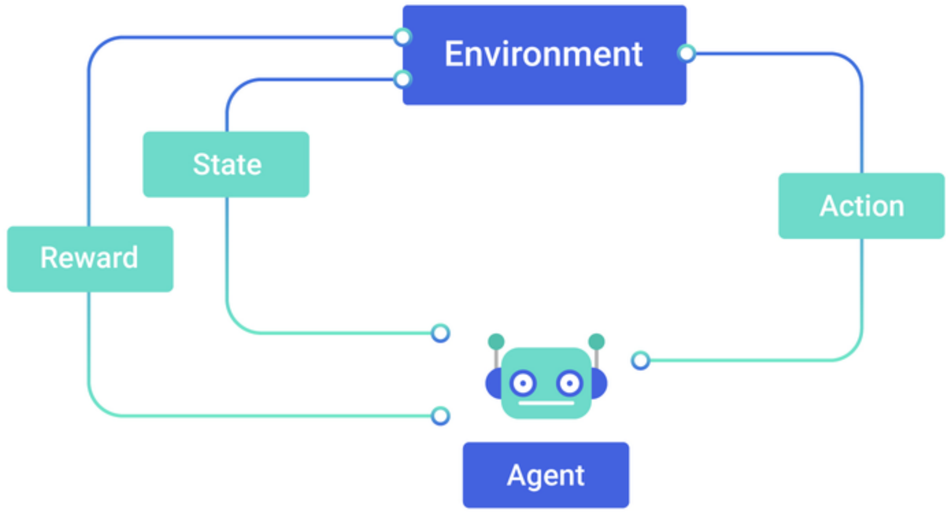
- Supervised: The target function is known. We have a labelled dataset
- Non Supervised: The dataset is not labelled
- Semi-supervised: The dataset is partially labelled.
- Reinforcement learning: ML system learns from the environment and it corrects itself by penalty or reward.



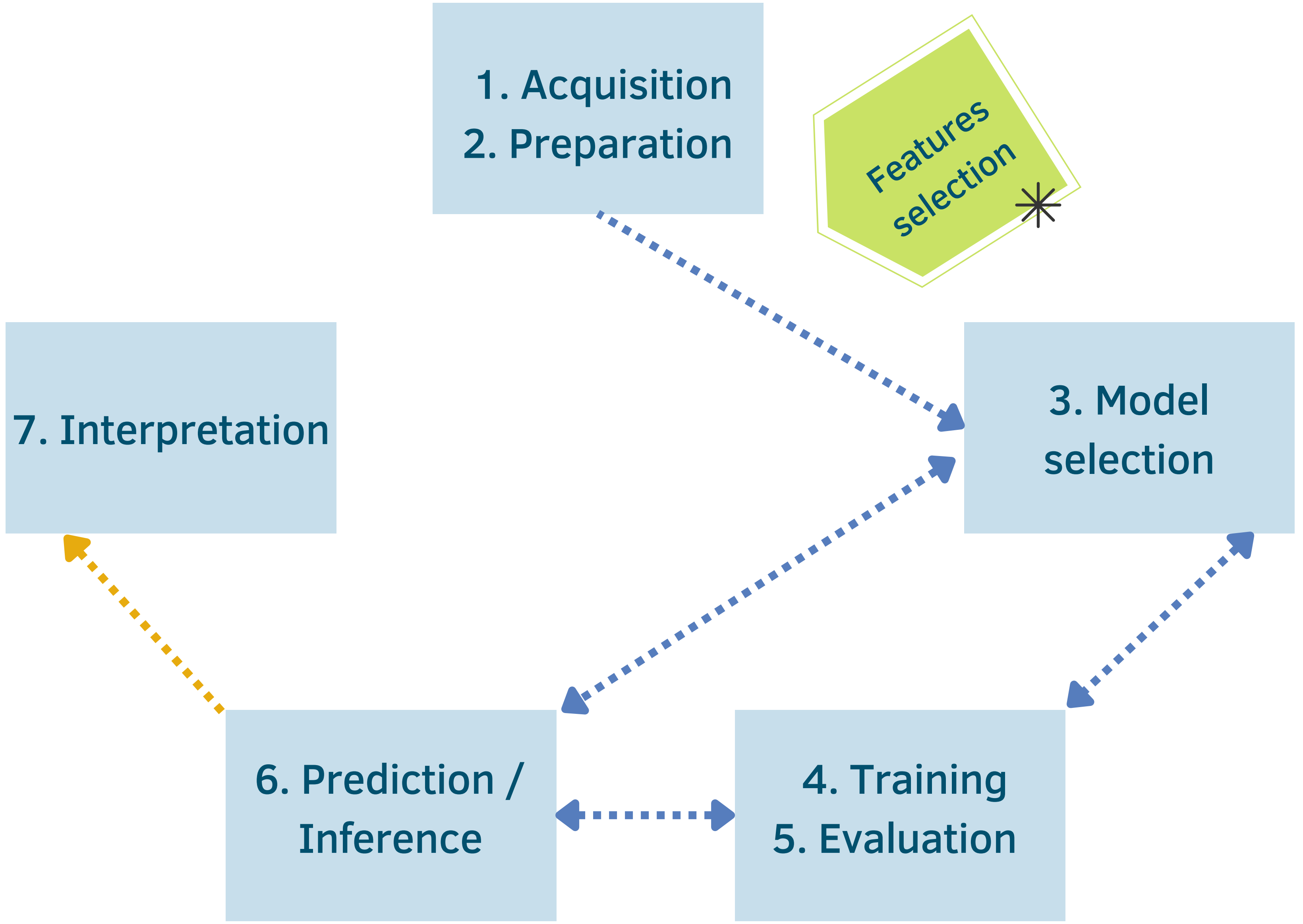








ML-based modeling

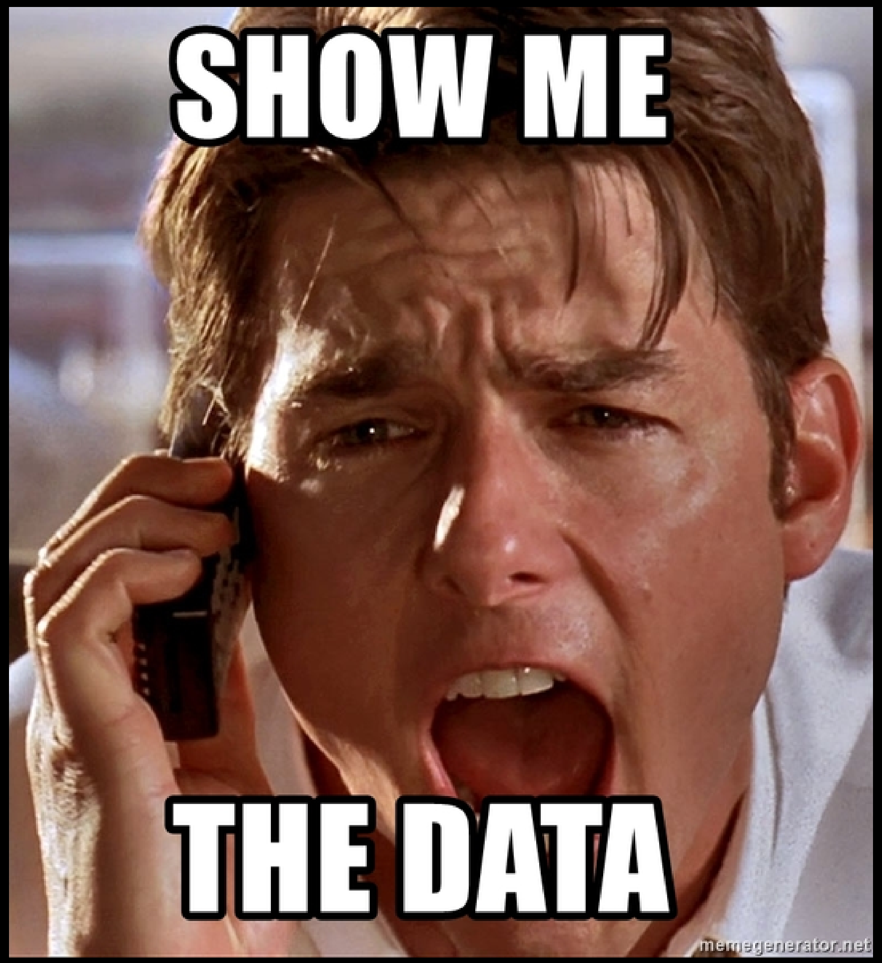


✳ • By hand = ML - automatically = DL
• Take care of dimensionality

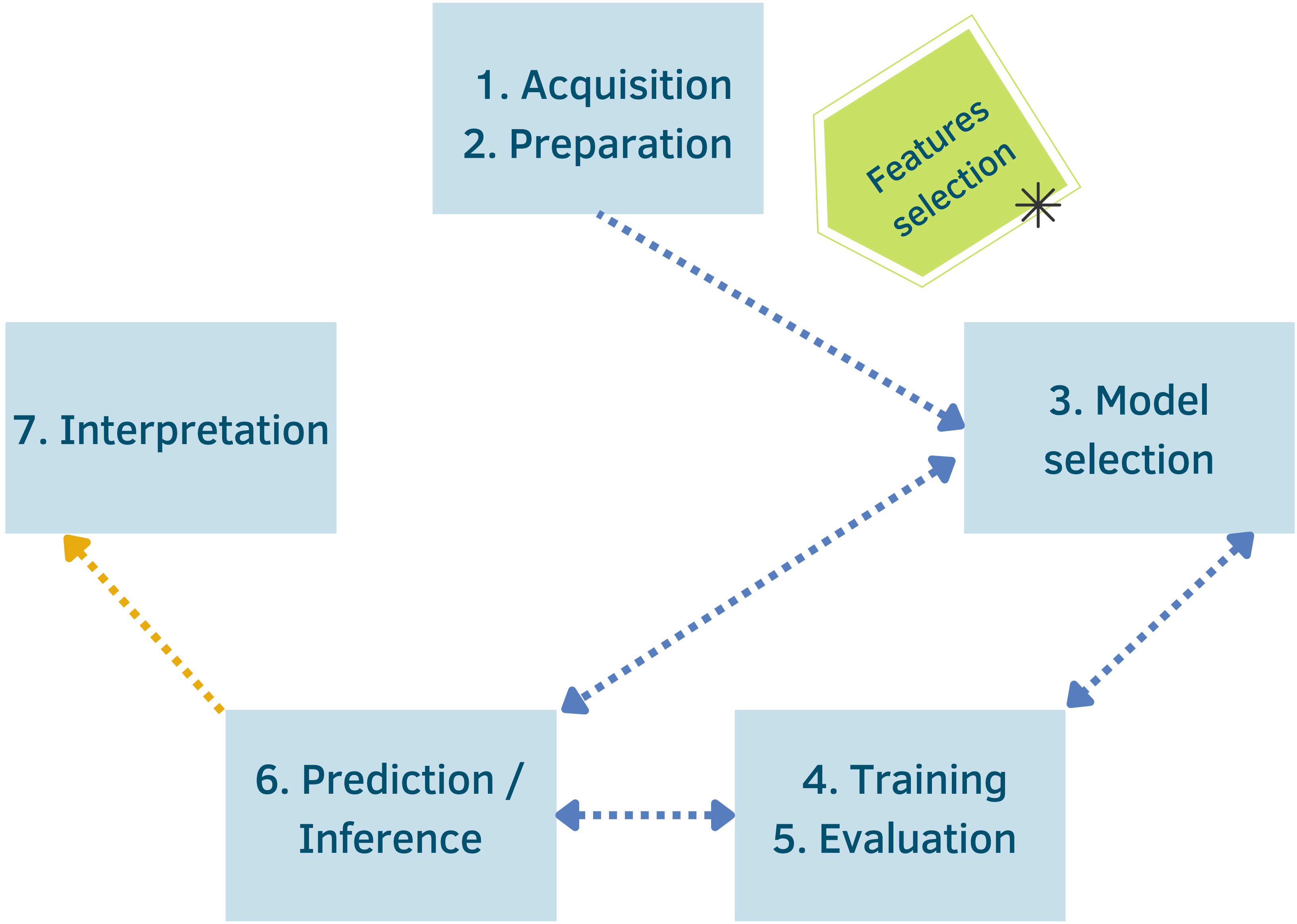
- Acquisition + preparation

  - 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

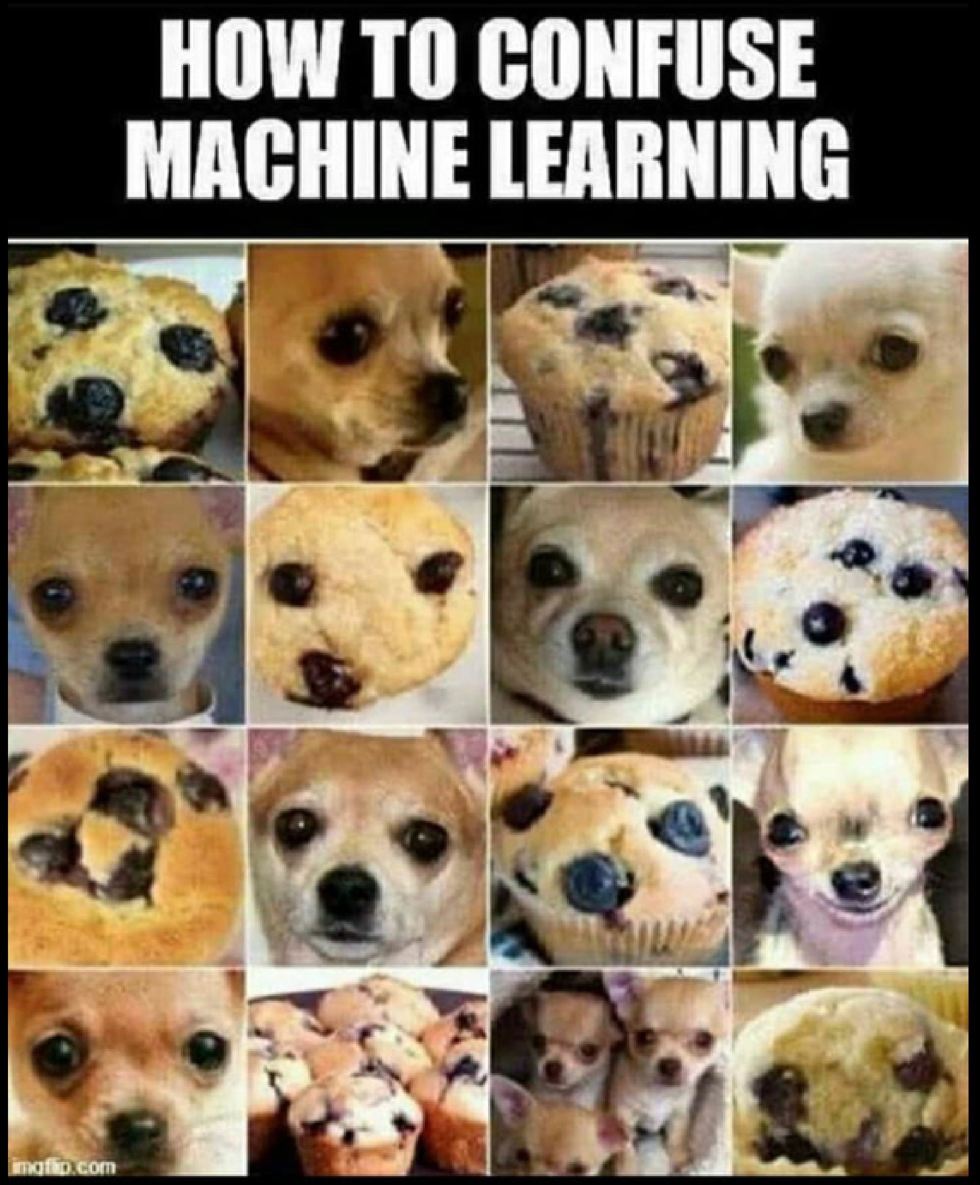


- By hand = ML - automatically = DL
- Take care of dimensionality

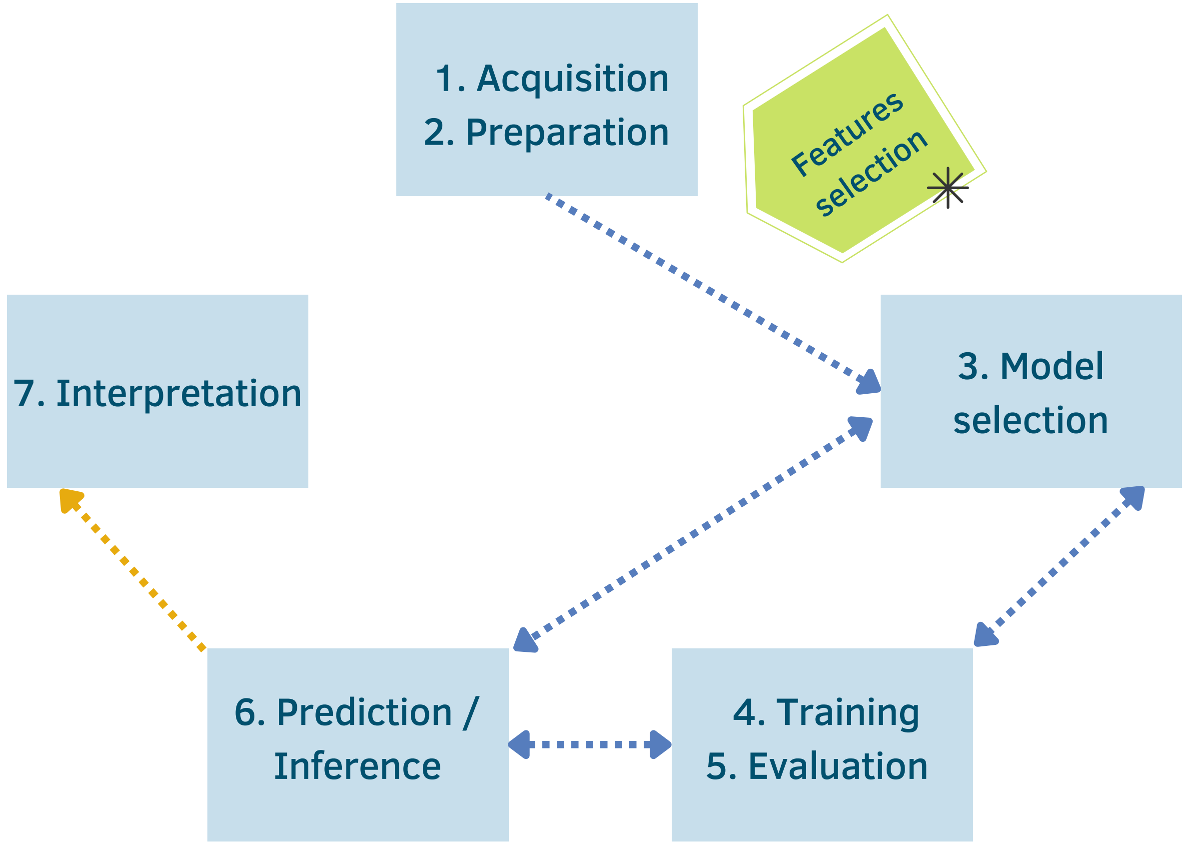
- Acquisition + preparation
  - Make our ML datasets:

- 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

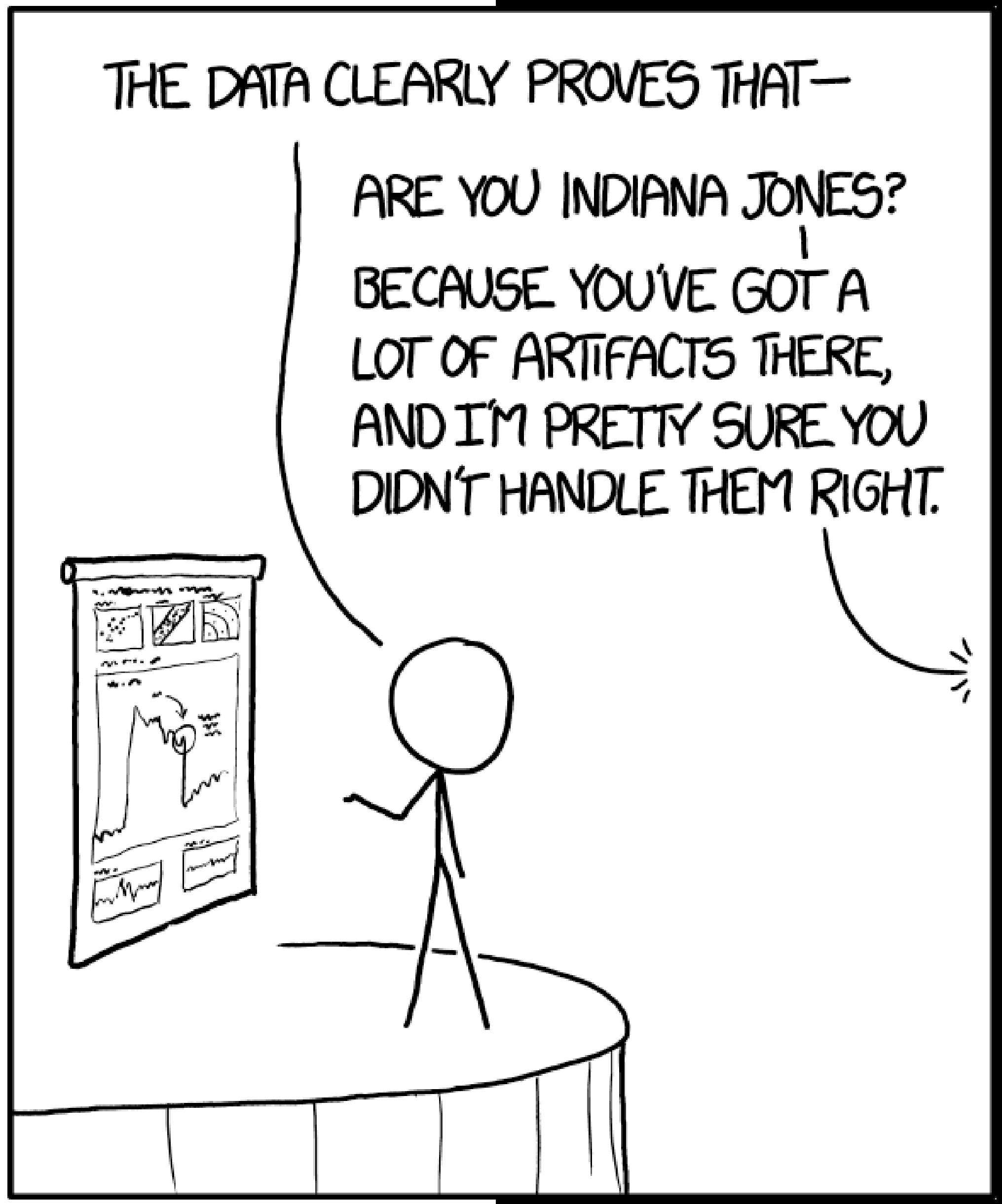


ML-based modeling



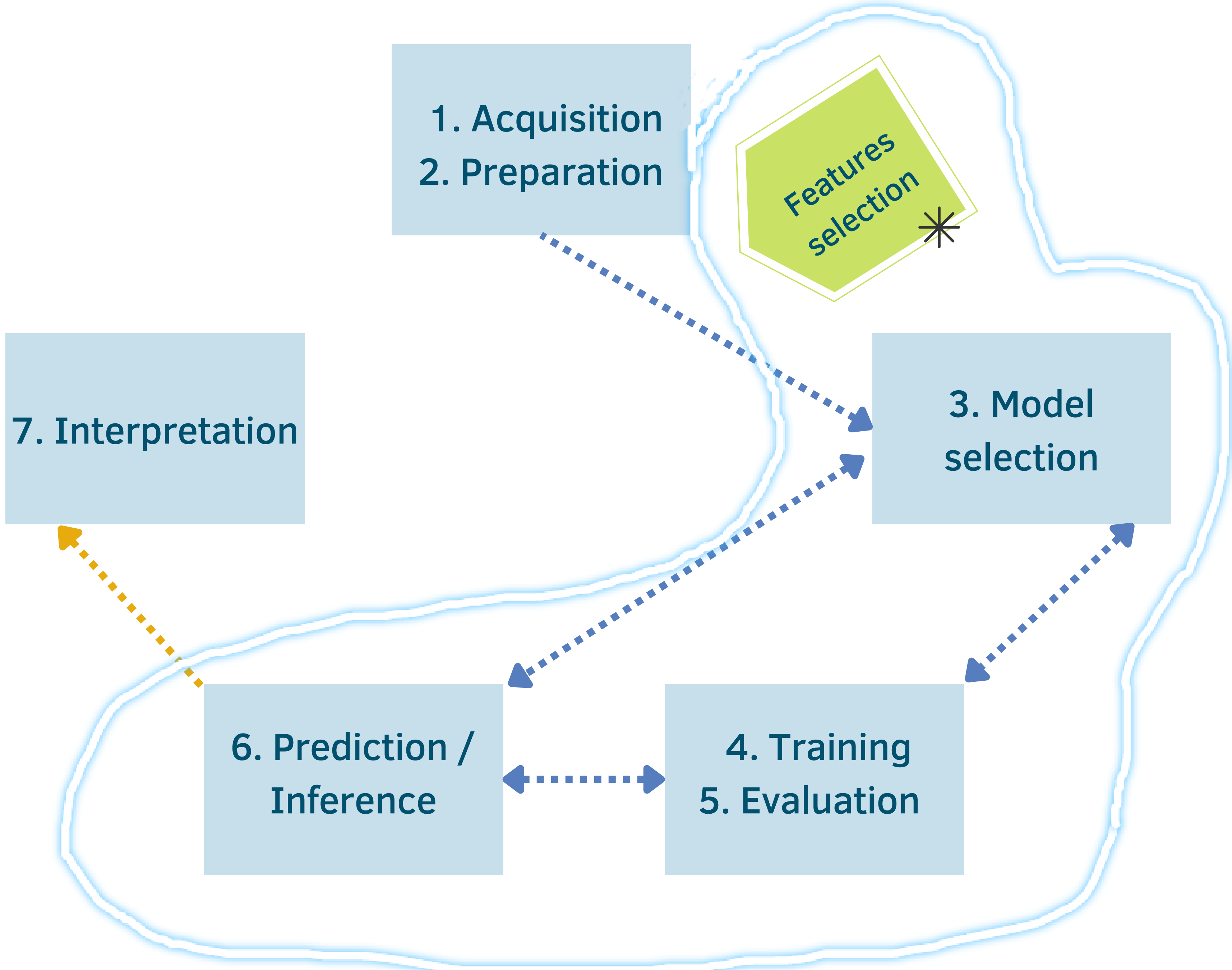
- Modeling

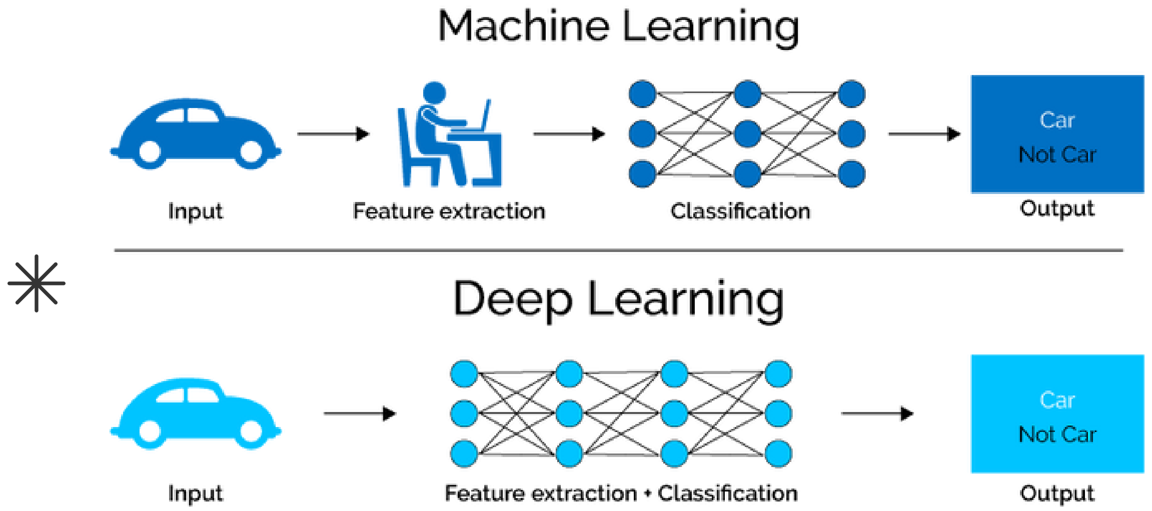
DATA PREPARATION FOR MODELING



- 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!

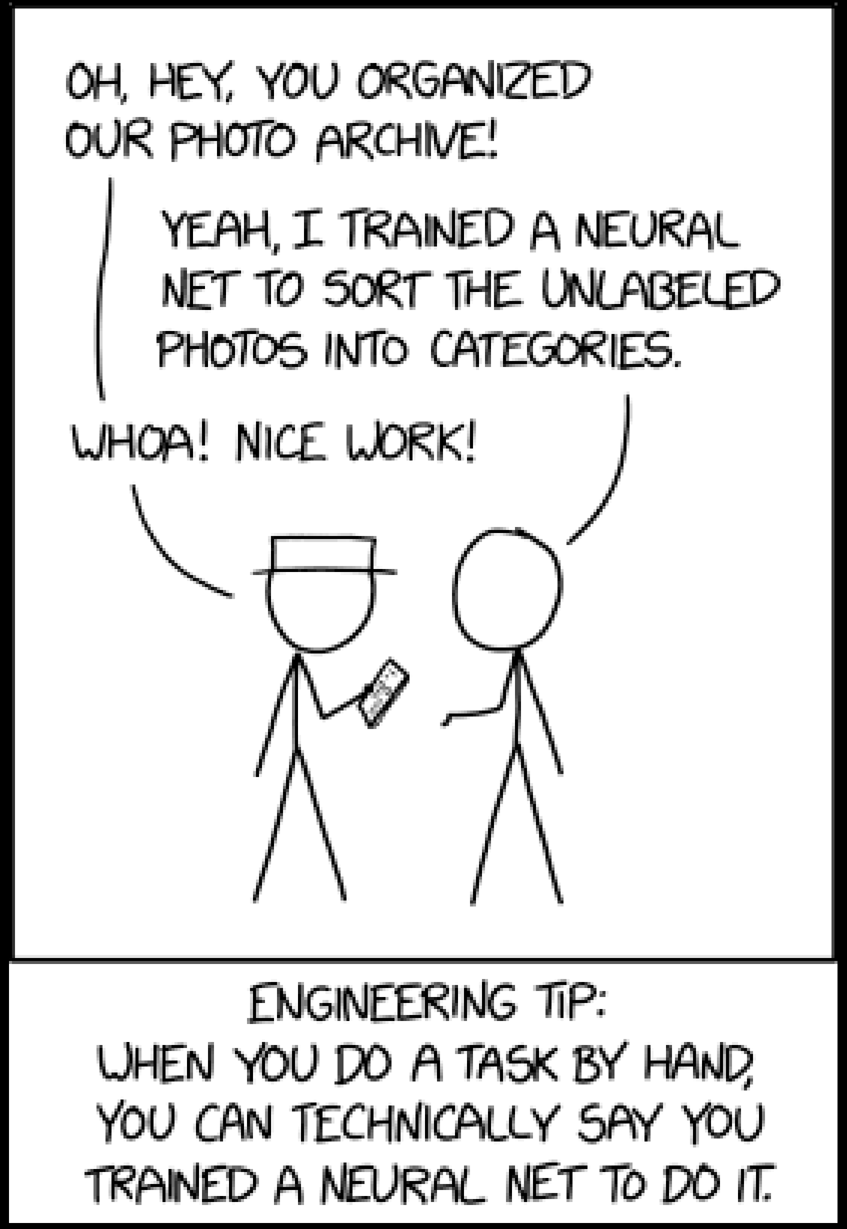
ML-based modeling





- Modeling

FEATURE ENGENEERING



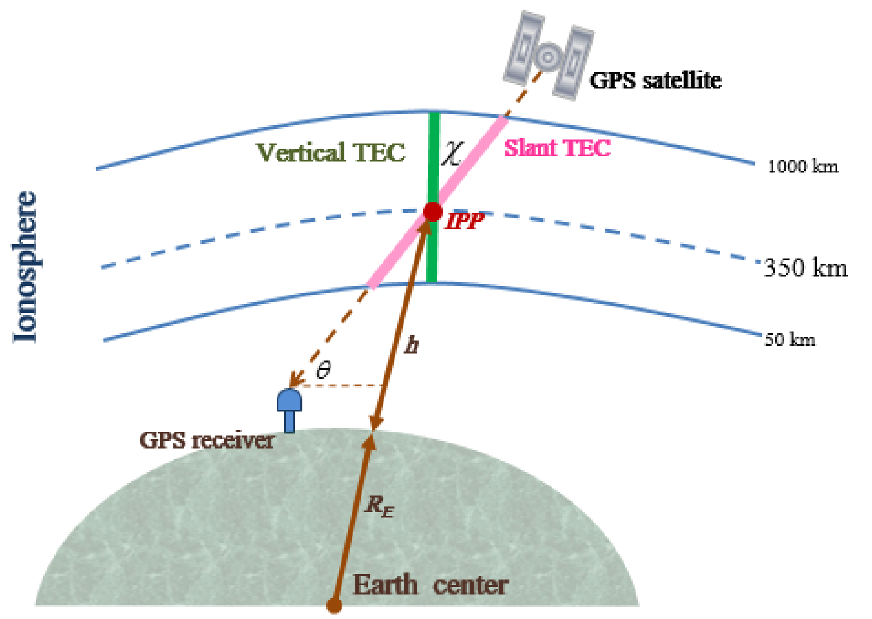
- Extract/choose relevant features from the dataset
- May include the creation of new features
- Requires experience and domain knowledge
- Feature engineering by hand: hard, slow, not robust, not scalable
- Explicit or/and implicit. (lagged features, statistics, data transformation)
- DL = automatic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

ML-based modeling

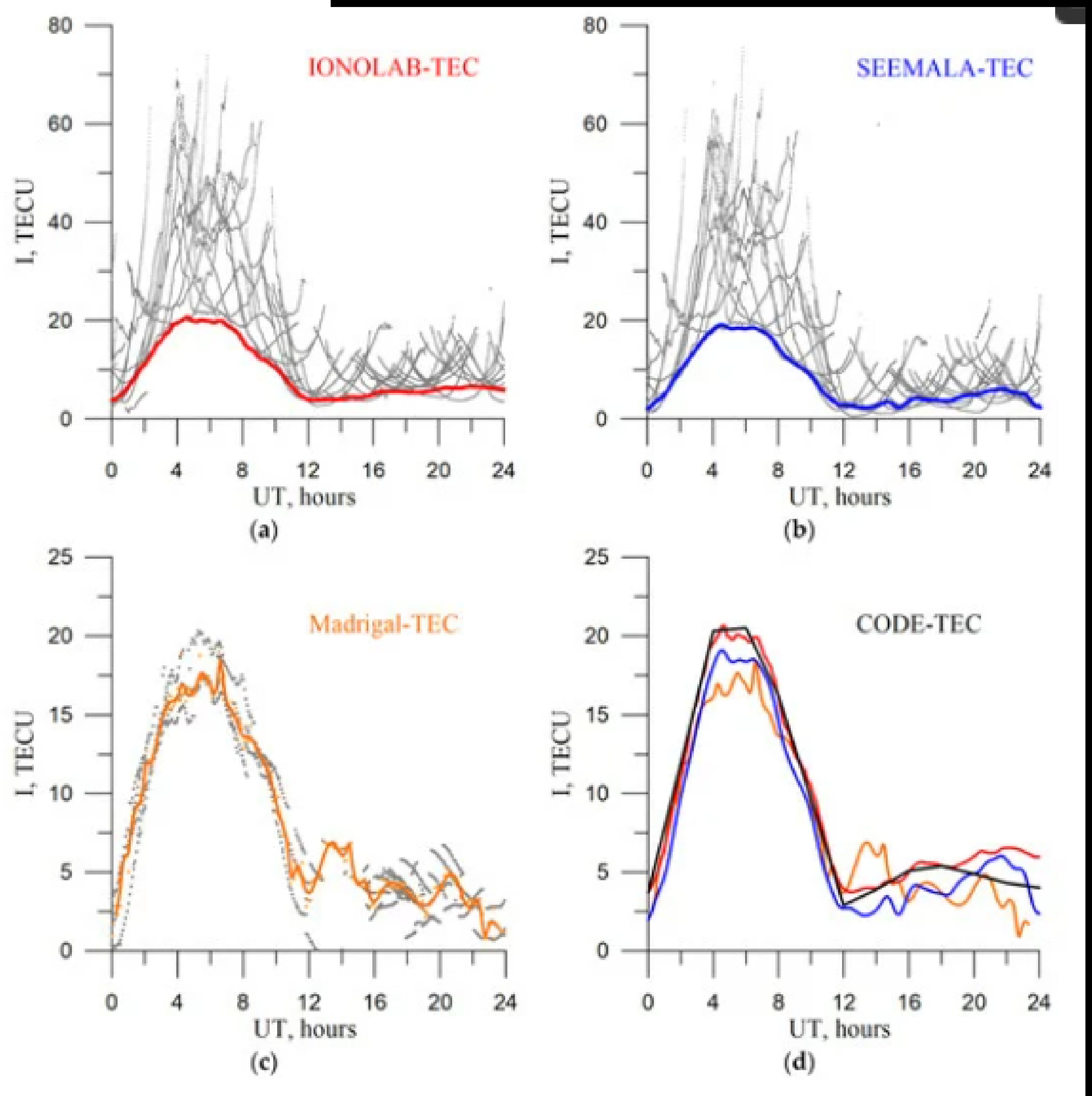
Example

- 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

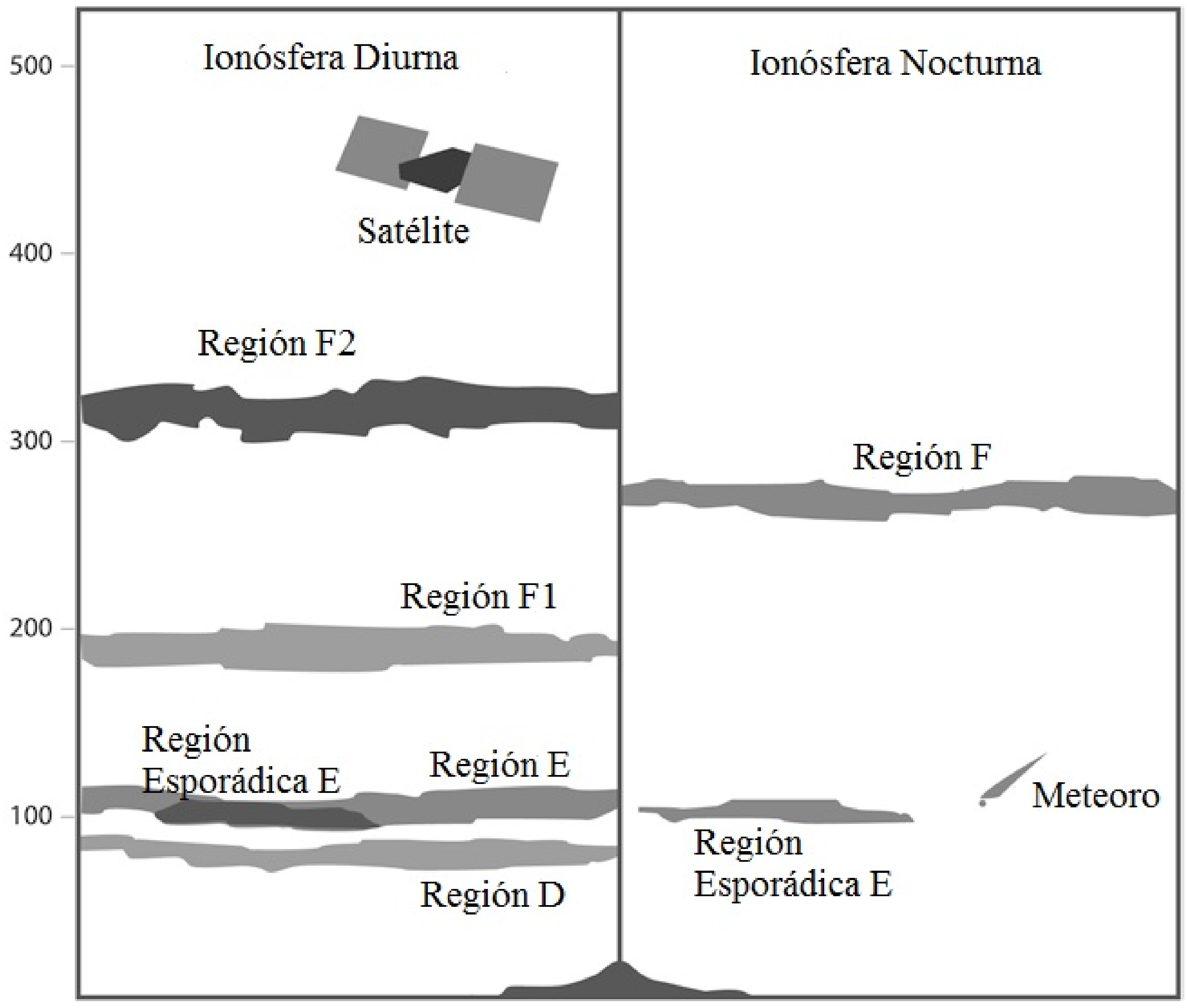
FEATURE ENGINEERING

- 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)

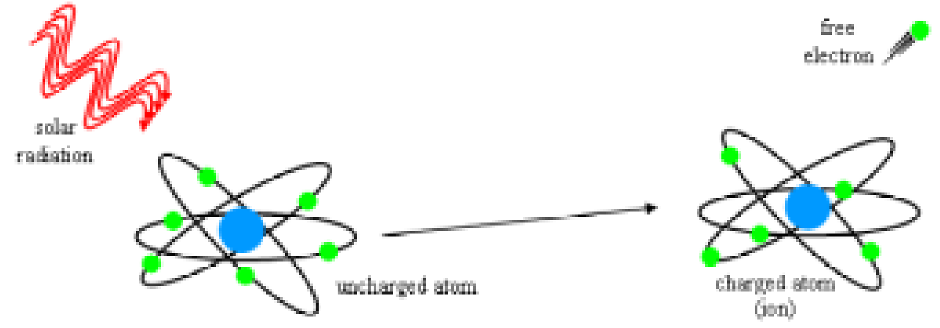
Ionosfera: porción de la alta atmósfera terrestre donde tanto iones como electrones se encuentran presentes en cantidades suficientes como para afectar las ondas de radio (IEEE Std 211-1997).

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

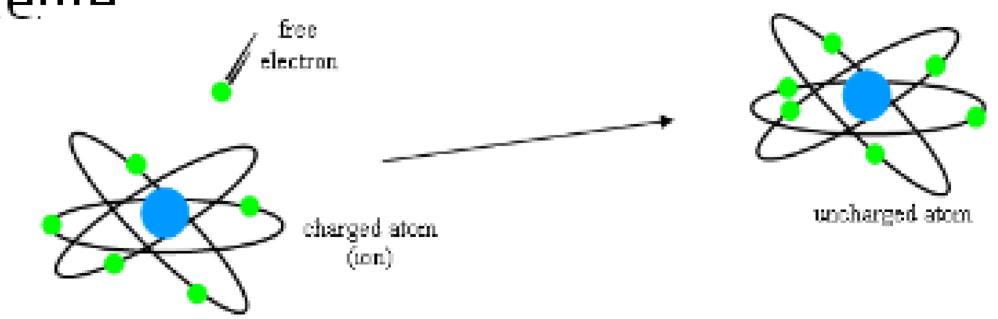
- Foto-ionización (producción de iones y electrones libres)



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

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



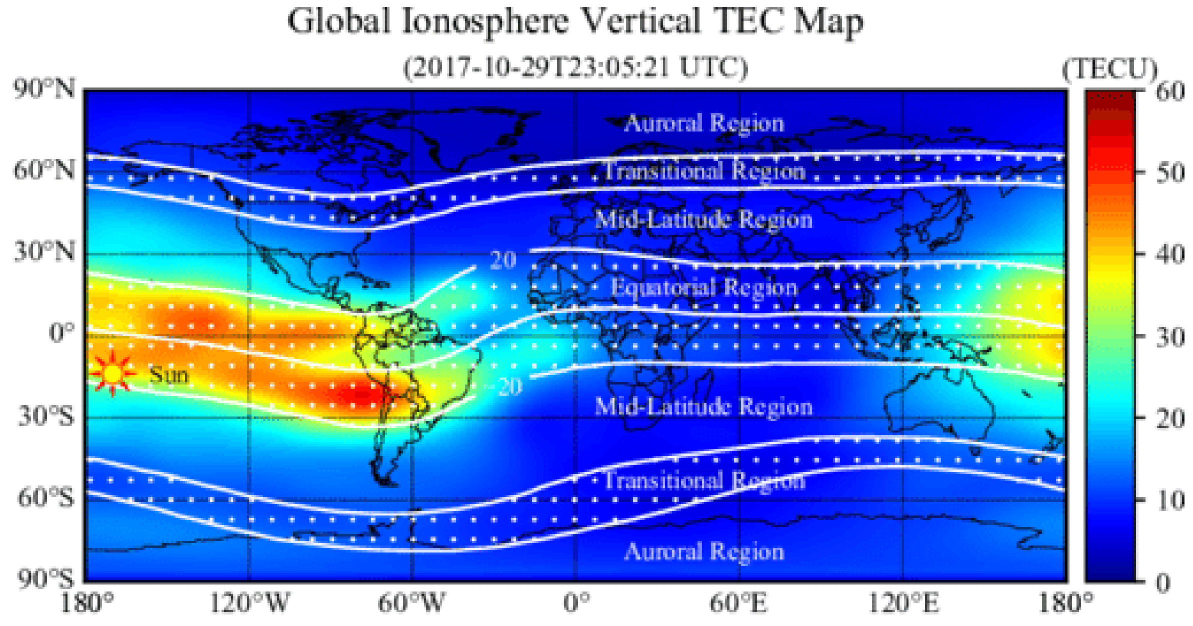
dN/dt = varición densidad electronica en fn del tpo (unidad de vol.), q = producción de electrones, l= pérdida de electrones, d= factor (pérdida por difusión, vientos neutros, drift electromag. vertical)

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

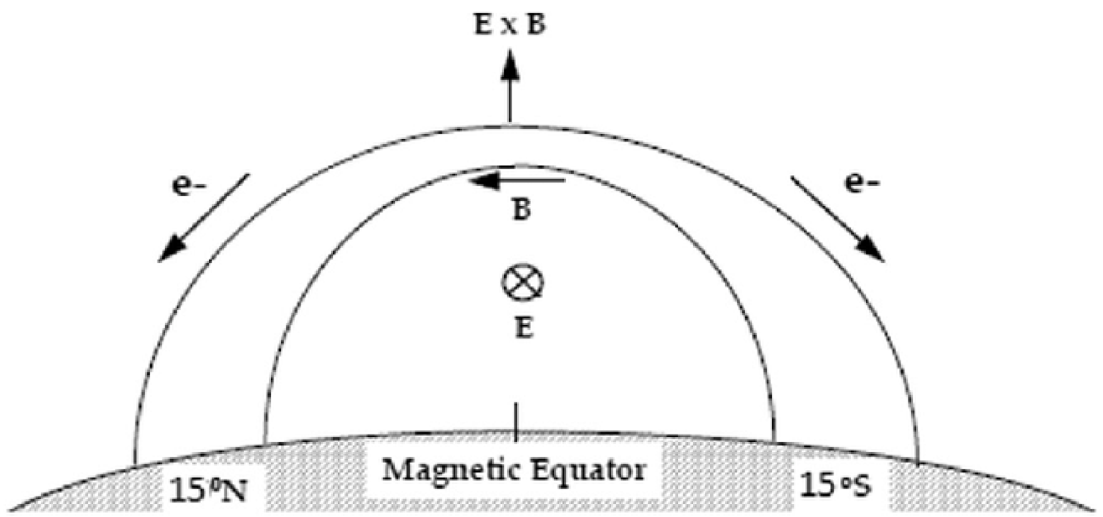
Parentesis (el medio bajo estudio)

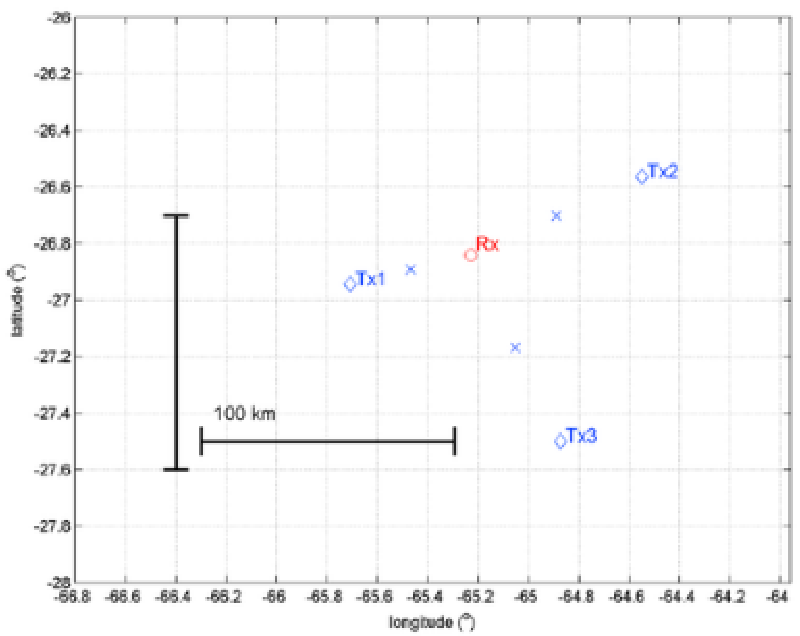
Variabilidad regular

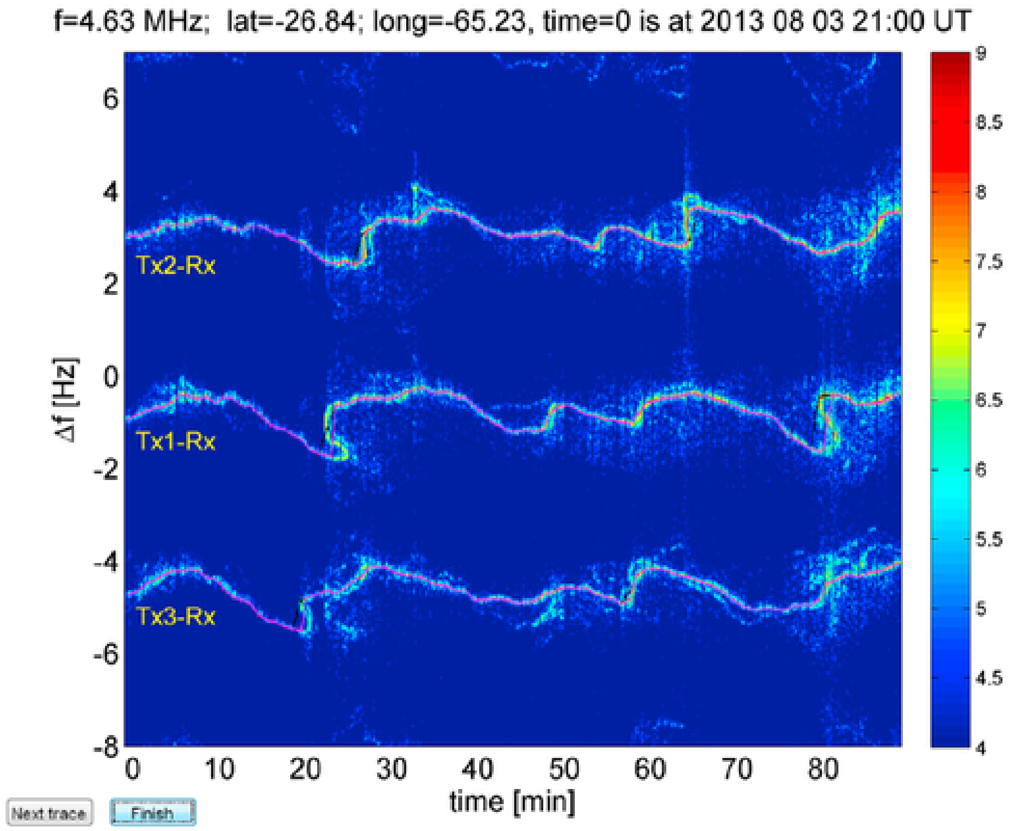
- diaria
- estacional
- geográfica/geomagnética
- ciclo solar



(b)







- TSWC-CAS

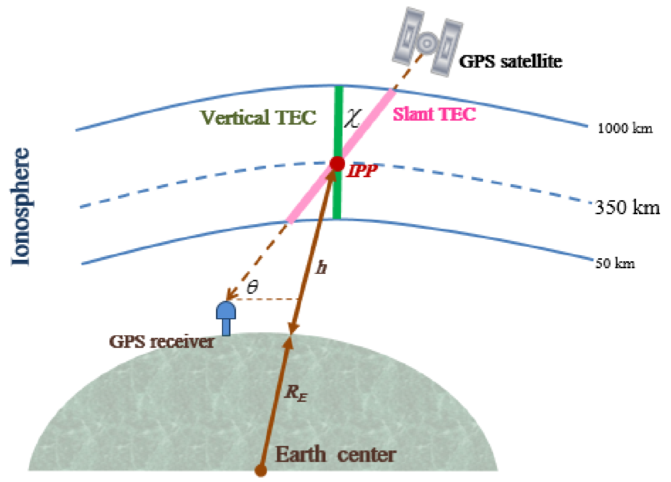
Irregularidades

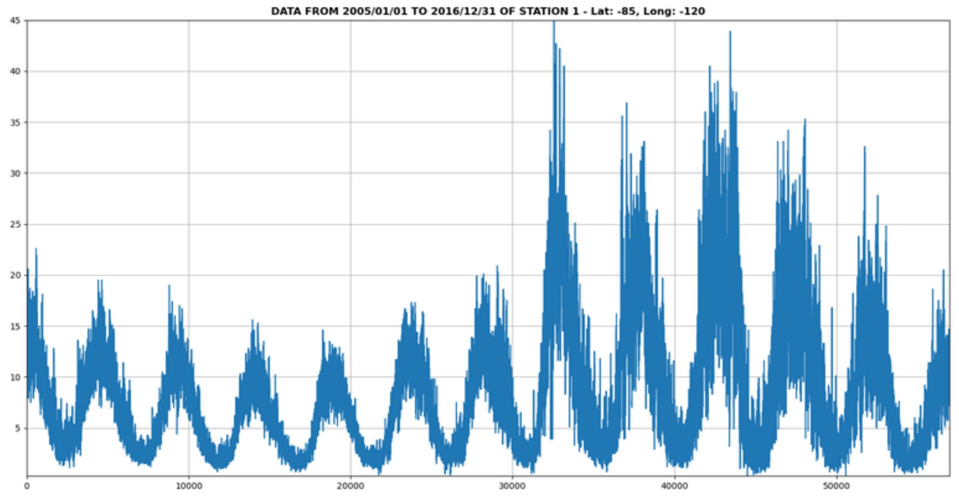
Travelling ionospheric disturbances (TIDs)
Irregularidades de la región F expresadas como oscilaciones similares a ondas del contorno de la denidad electrónica que va descendiendo lentamente con el tiempo
Se clasifican en : large-scale TIDs (LS-TIDs) y medium-scale TIDs (MS-TIDs).

- 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

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 ENGINEERING

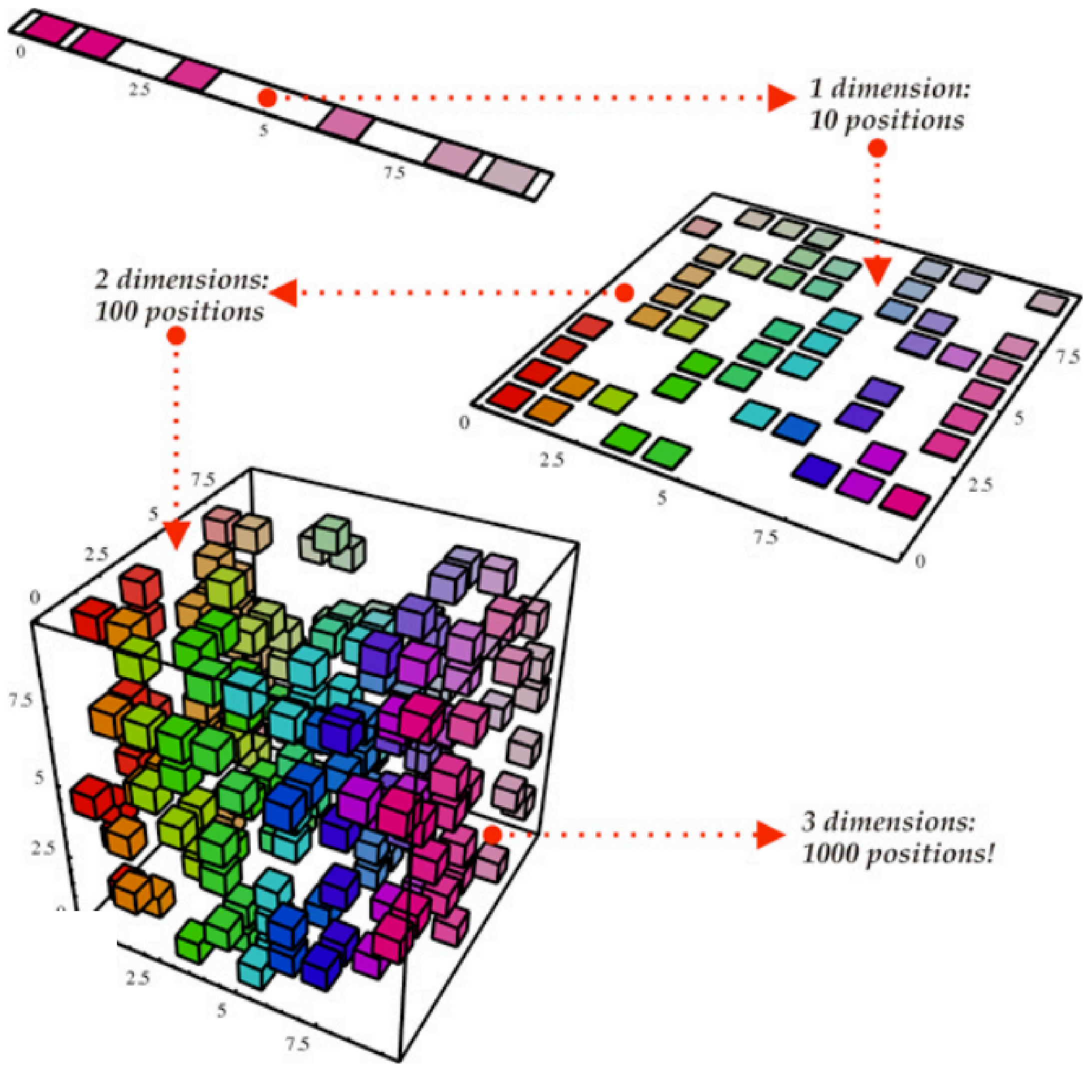
- Extract/choose relevant features from the dataset
- May include the creation of new features
- Requires experience and domain knowledge
- Feature engineering by hand: hard, slow, not robust, not scalable
- Explicit or/and implicit. (lagged features, statistics, data transformation)
- DL = automatic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

How may is too many?

ML-based modeling

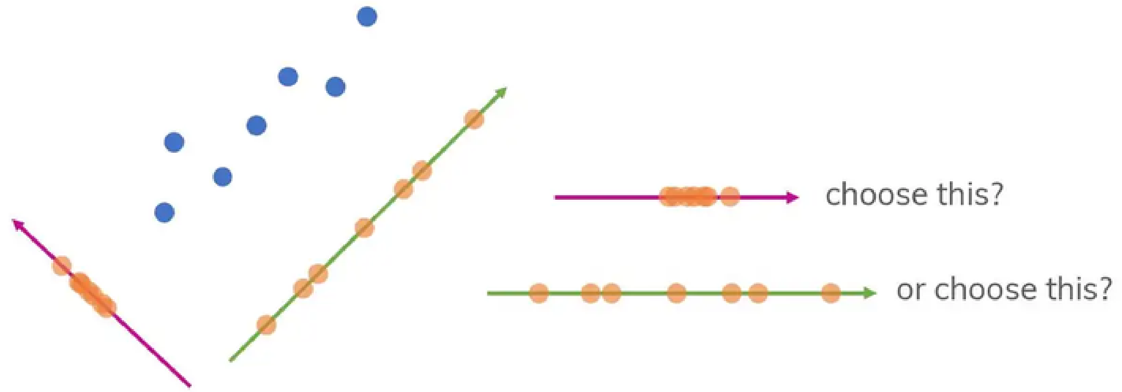
☑
☑ FEATURES (Selection)
☐

- 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?



+ features + data + computing effort + sparce data

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

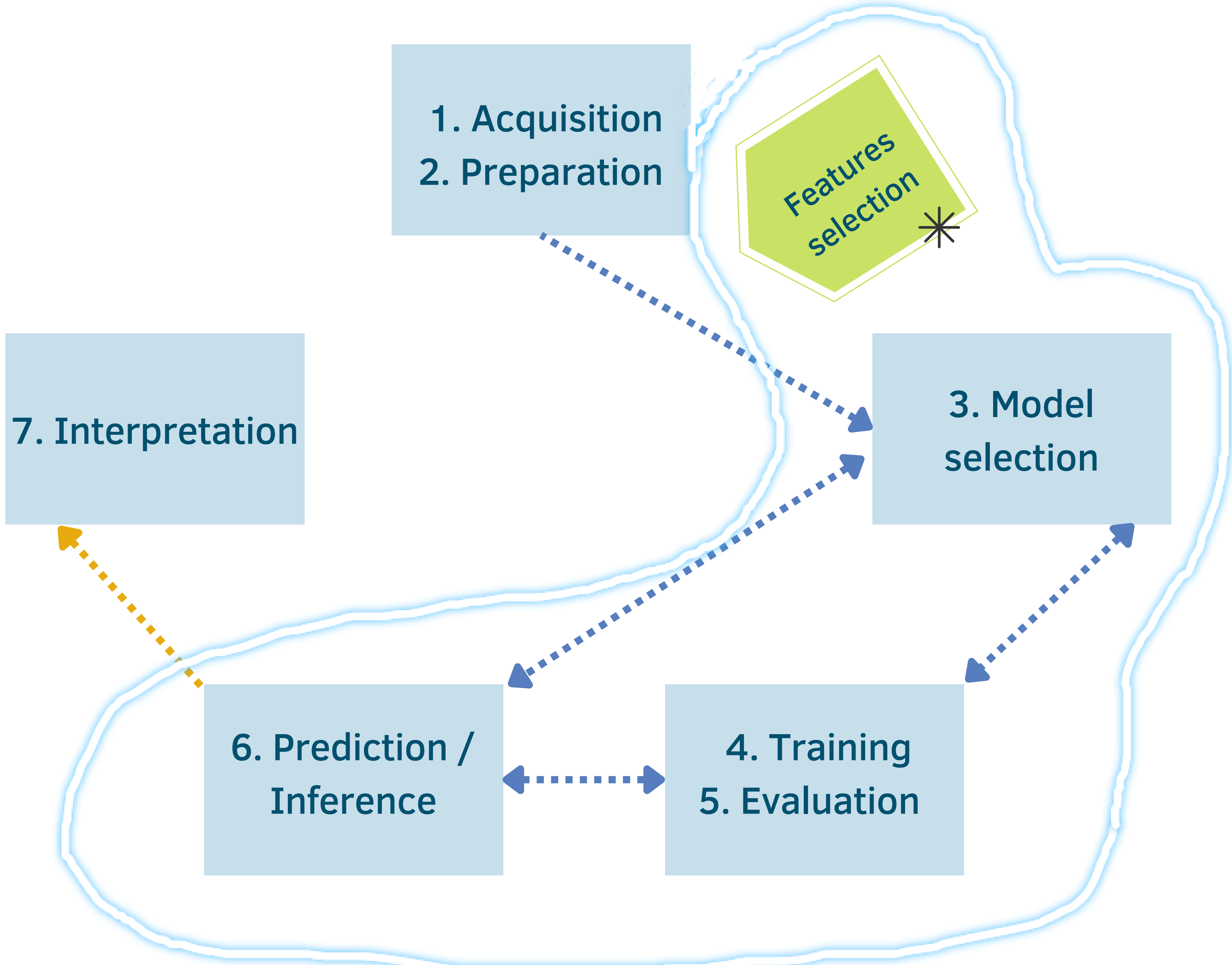
POINTS OF SIGNIFICANCE

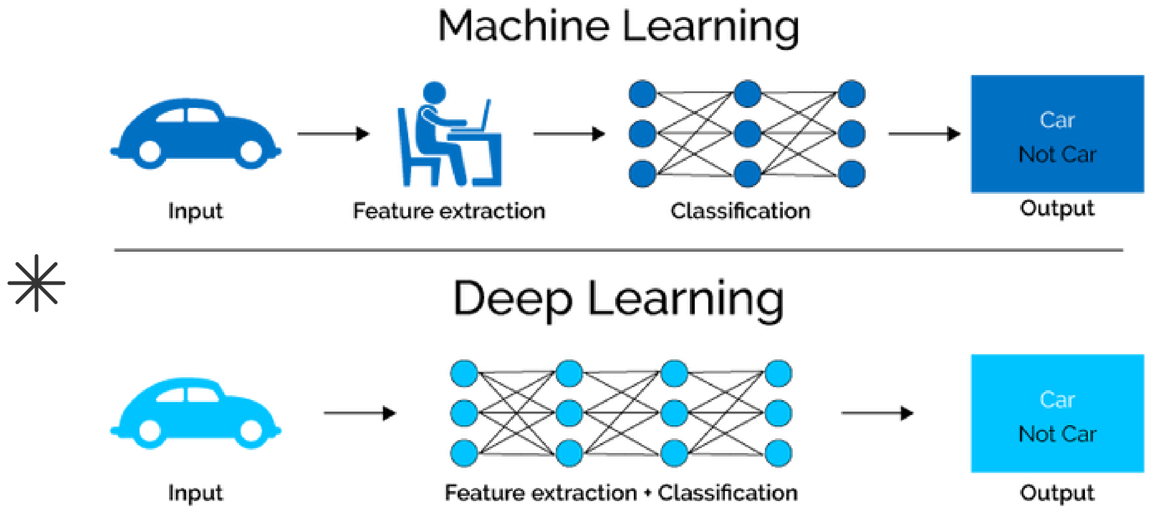
The curse(s) of dimensionality

Naomi Altman & Martin Krzywinski

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

ML-based modeling

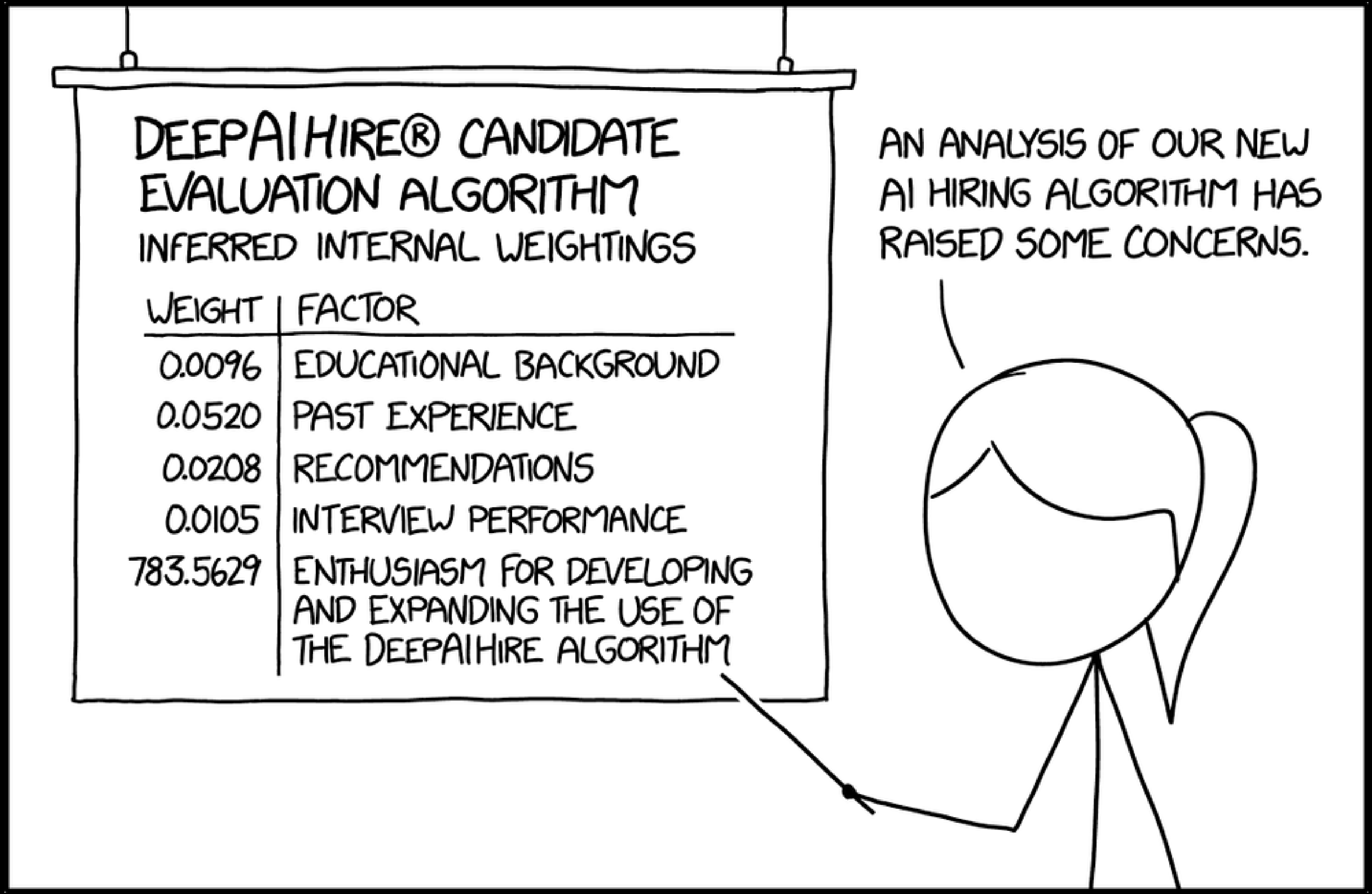




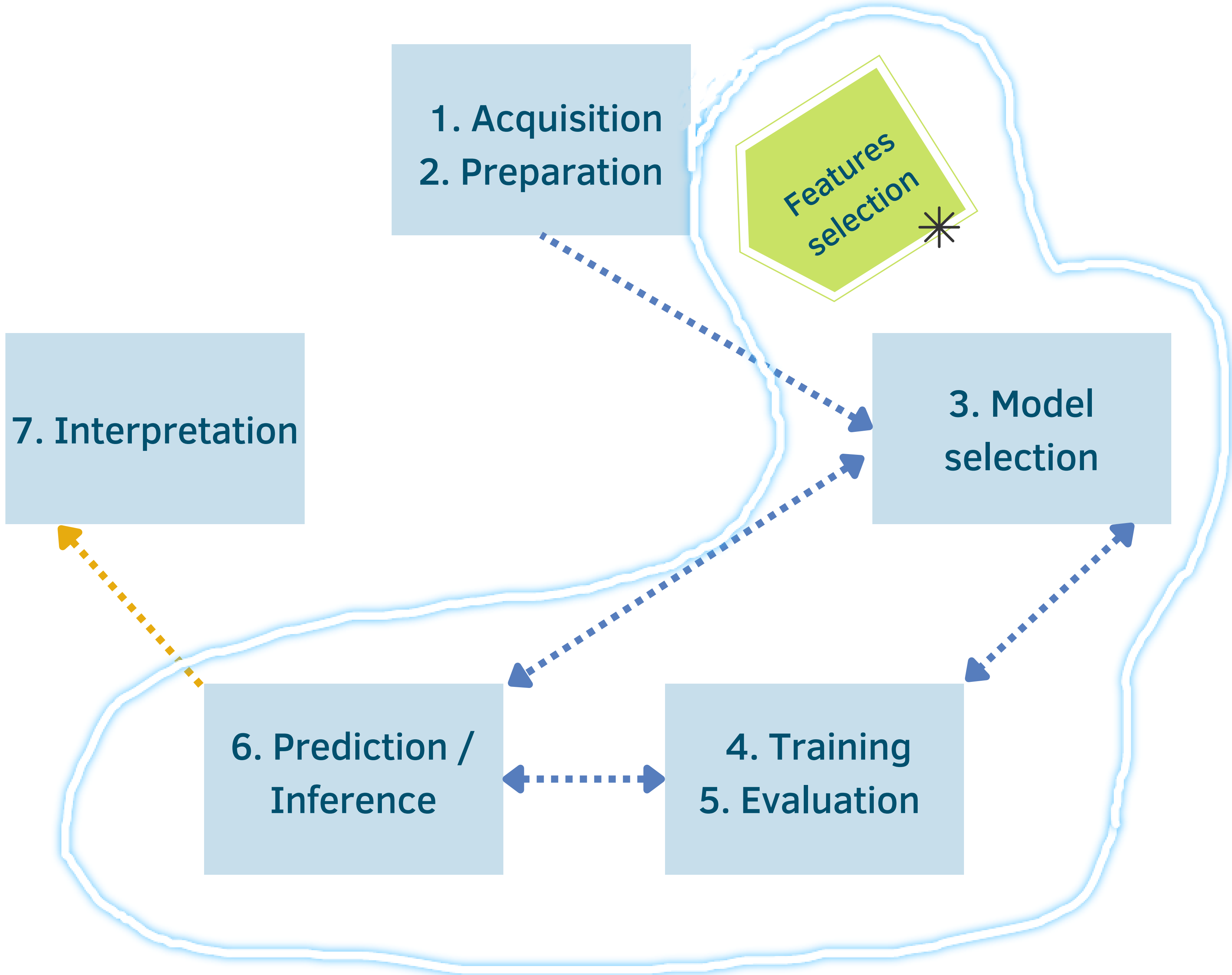
- Modeling

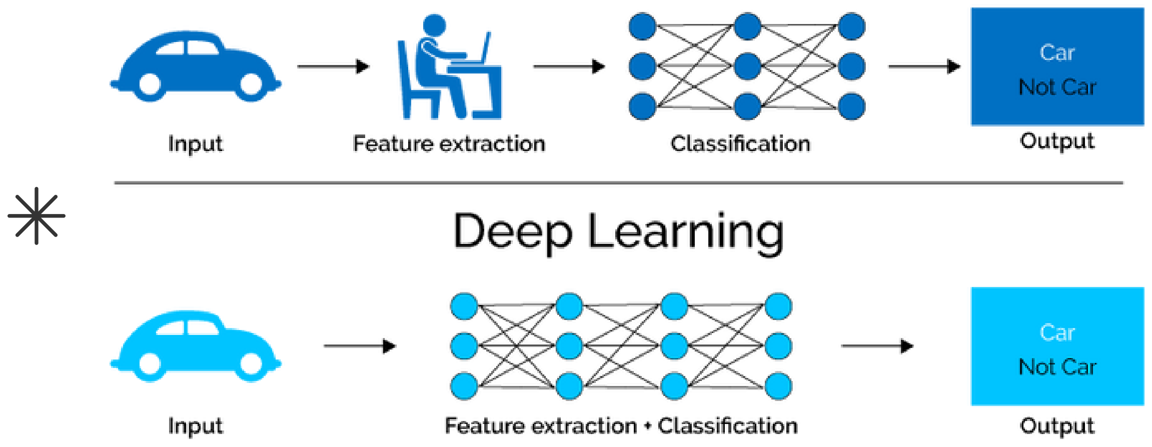
in detail in this course!

Iterative steps
- MODEL SELECTION
- TRAINING
- EVALUATION
- PREDICTION



ML-based modeling





- Modeling

INTERPRETATION

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

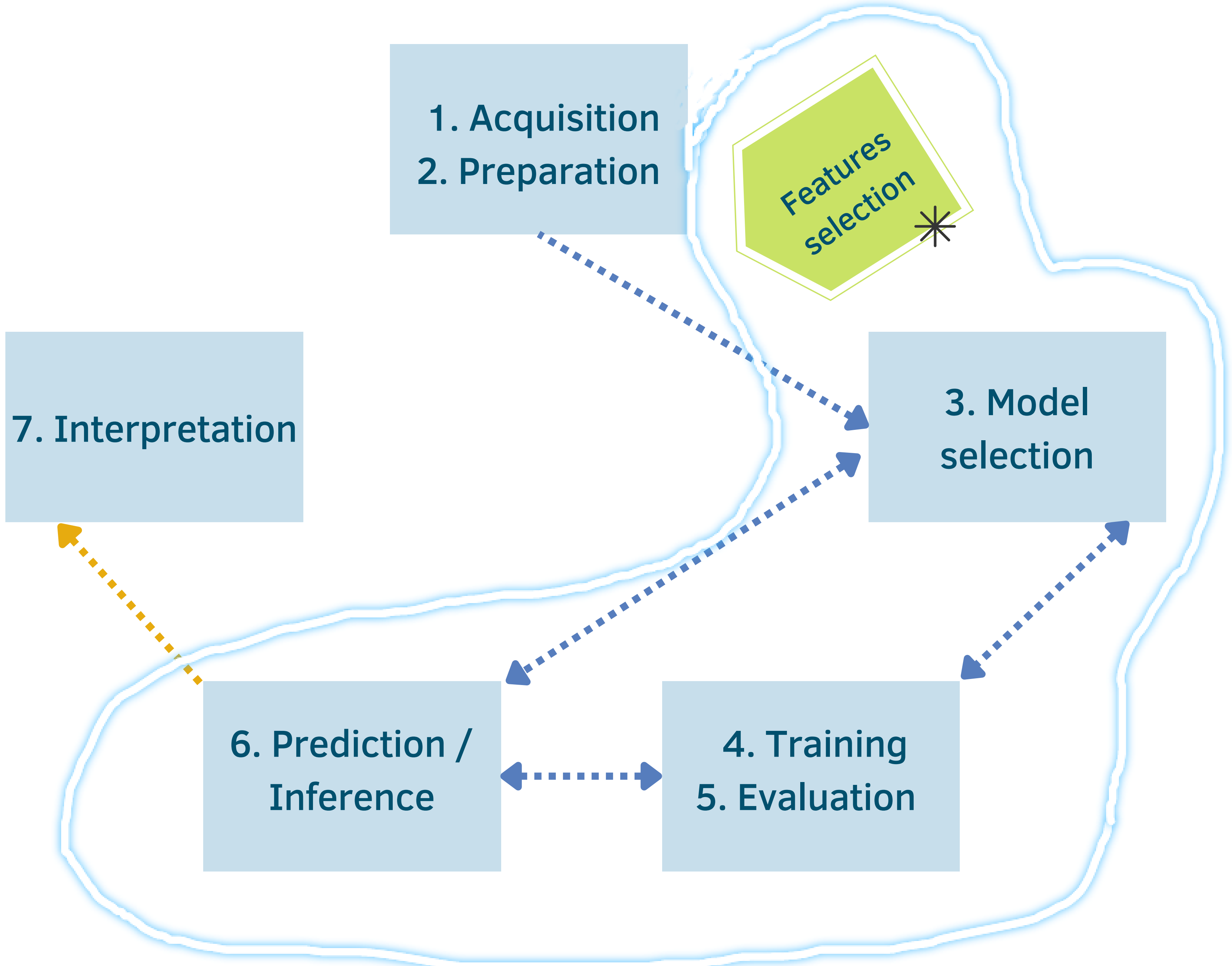
- interpretability: transparency.
- explainability: action or procedure taken by a model with the intent of clarifying or detailing its internal functions.

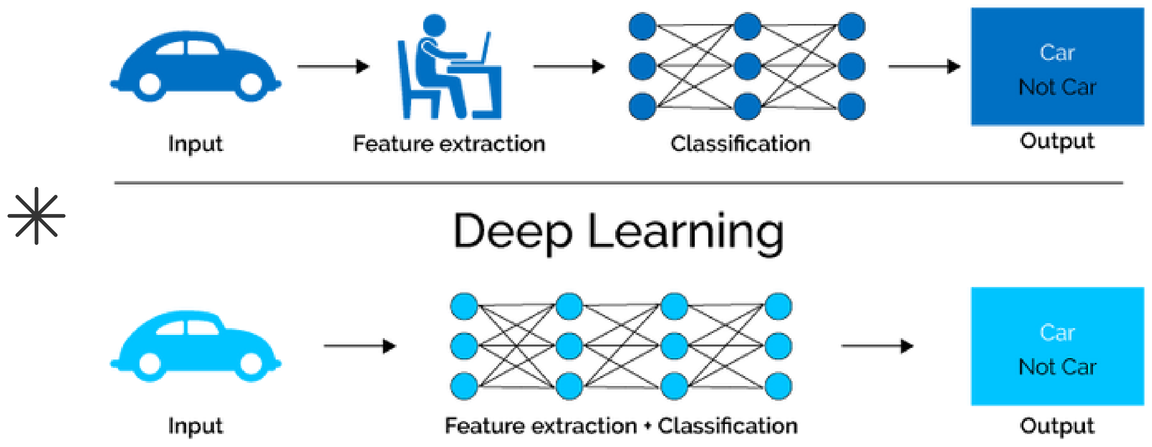
eXplainable AI (XAI): ML techniques

- 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)

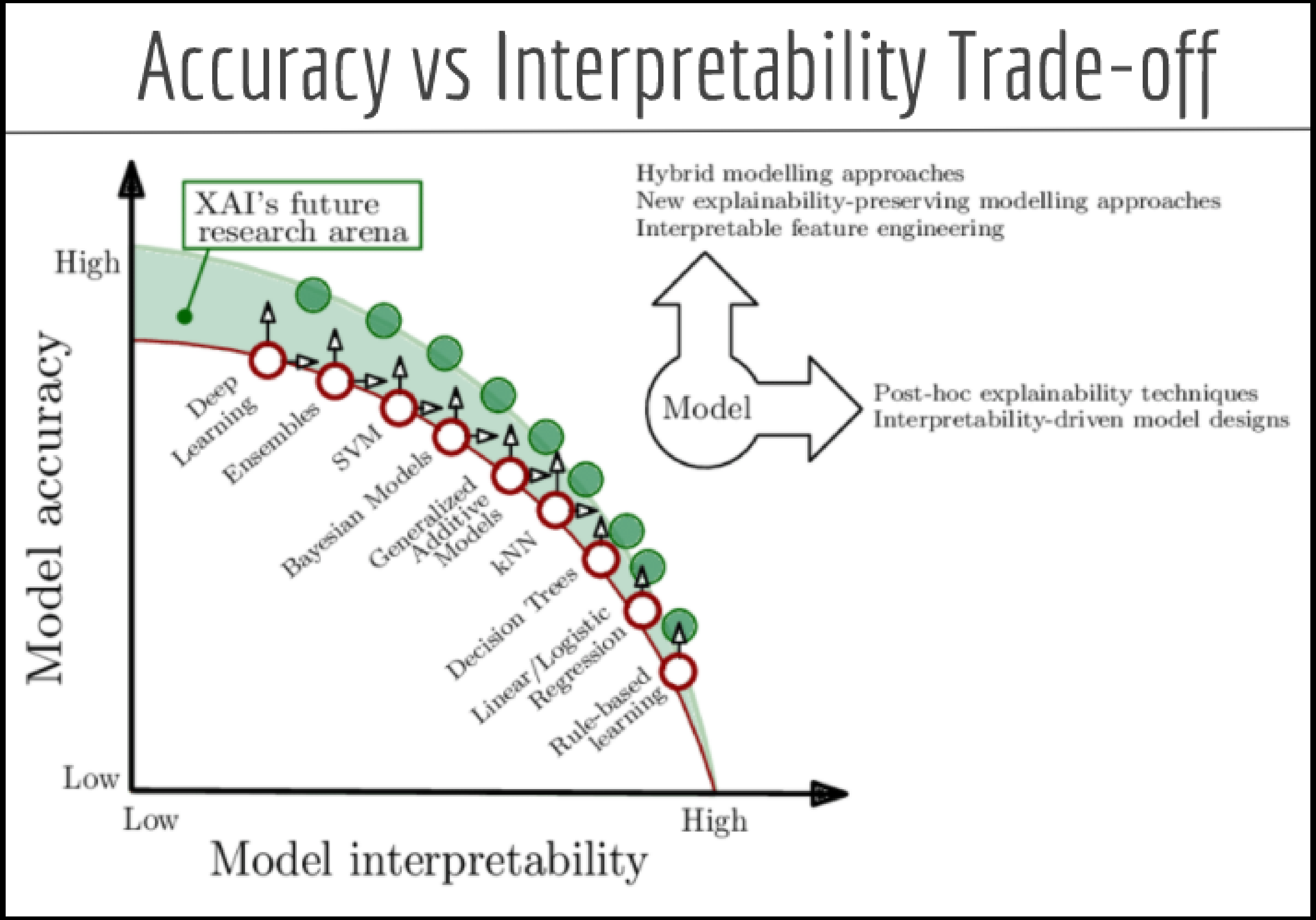
ML-based modeling



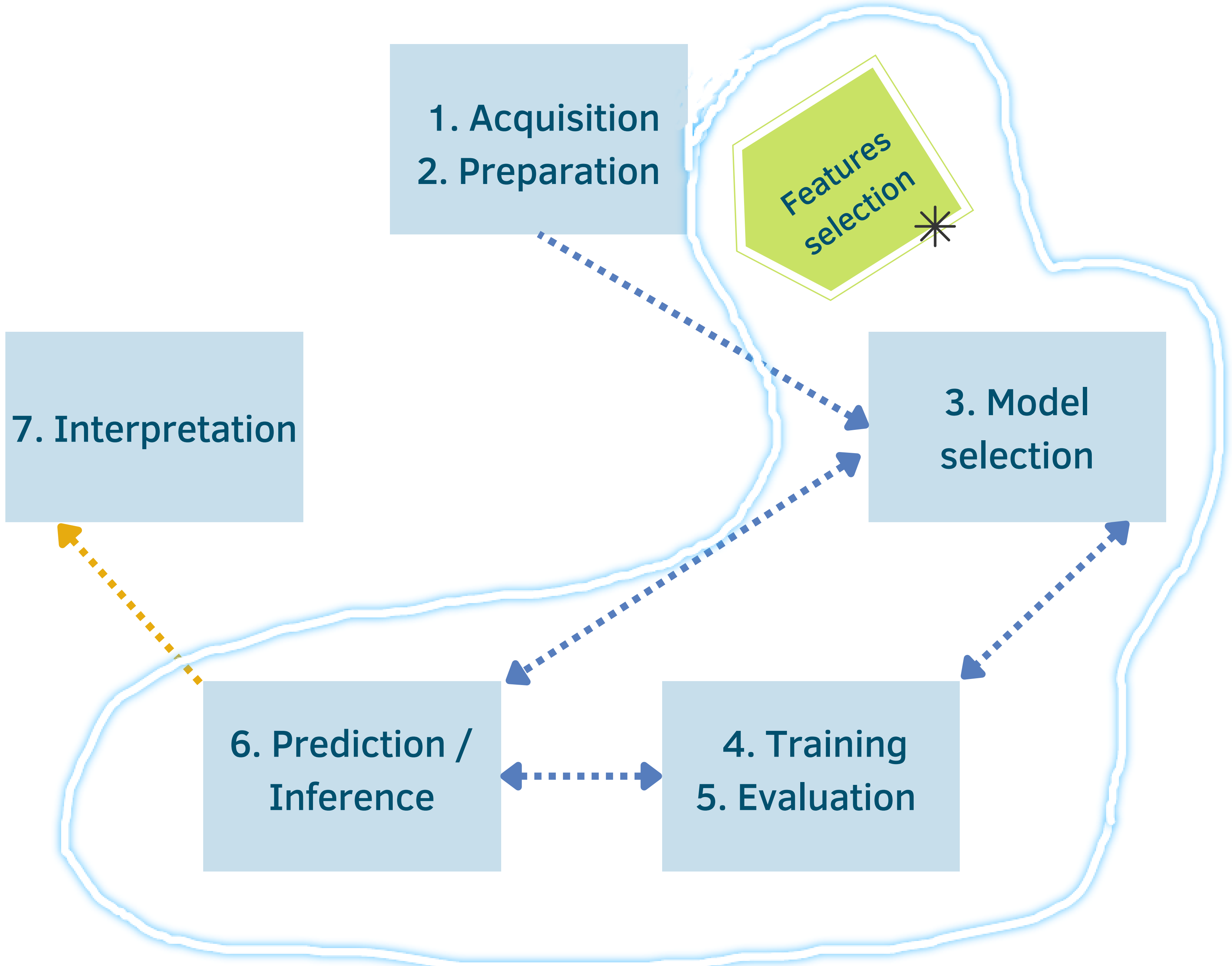


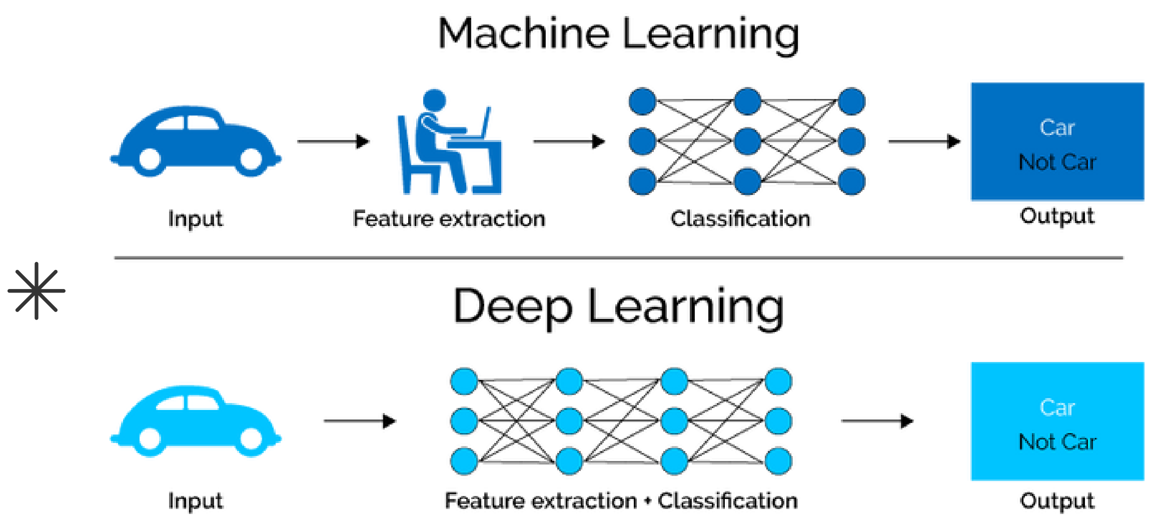
- Modeling

INTERPRETATION



ML-based modeling



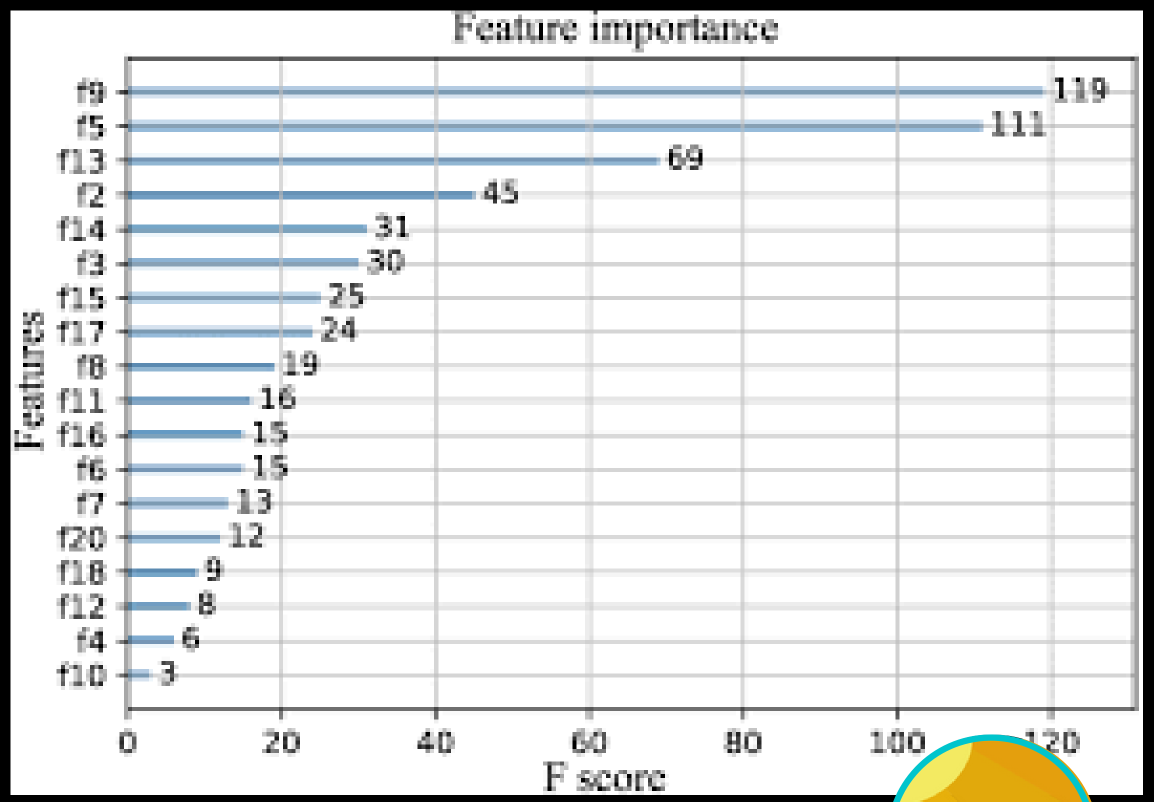


- Modeling

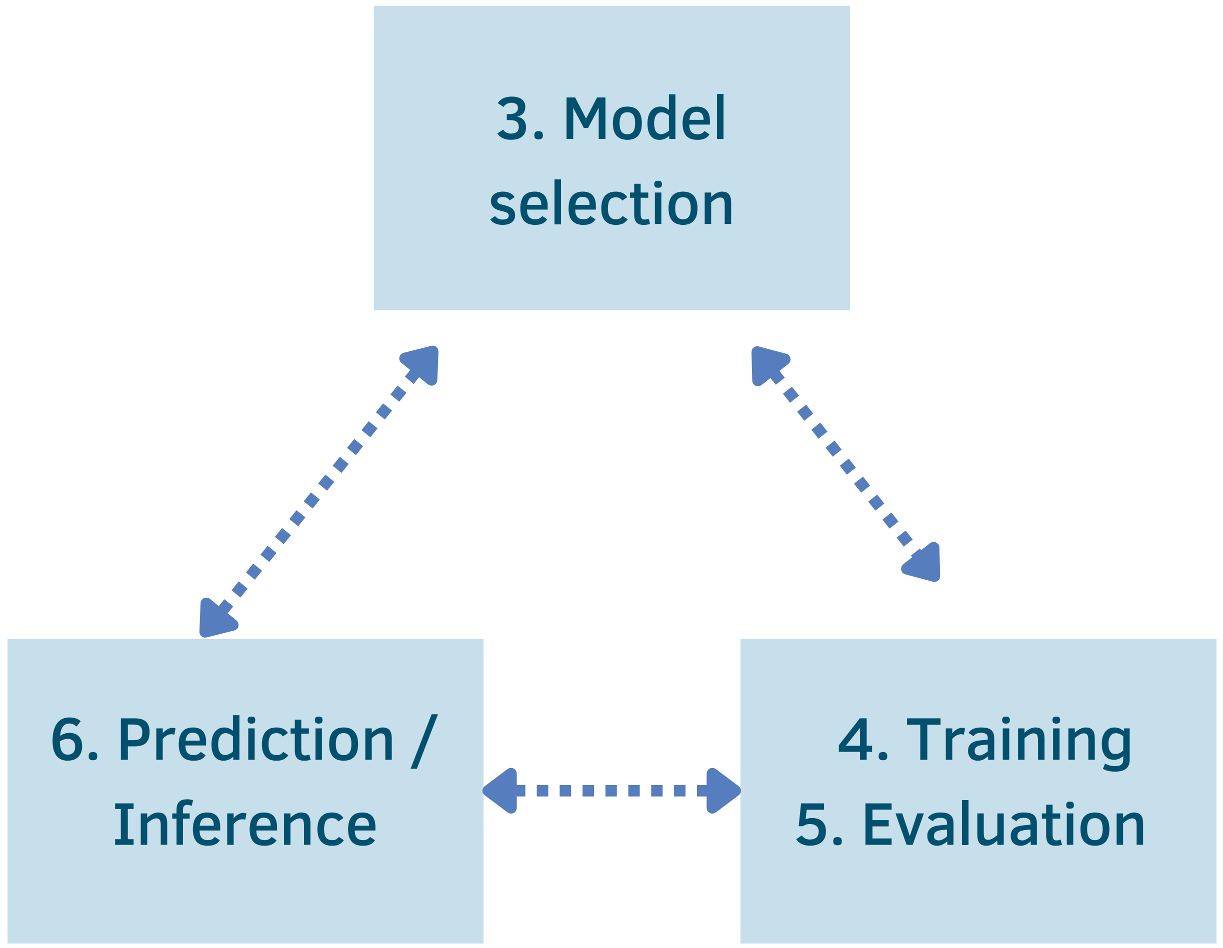
INTERPRETATION

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



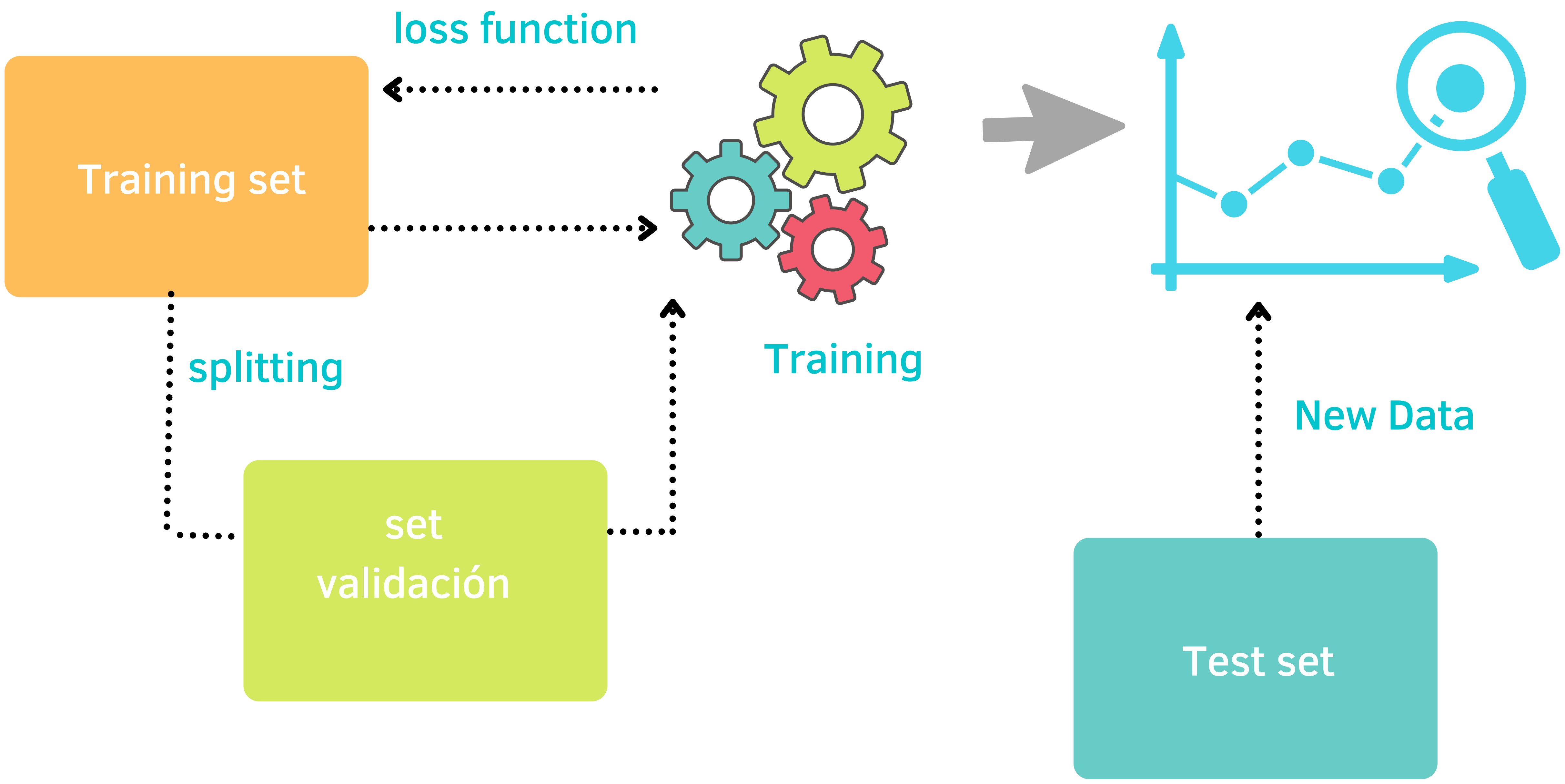
- Modeling

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



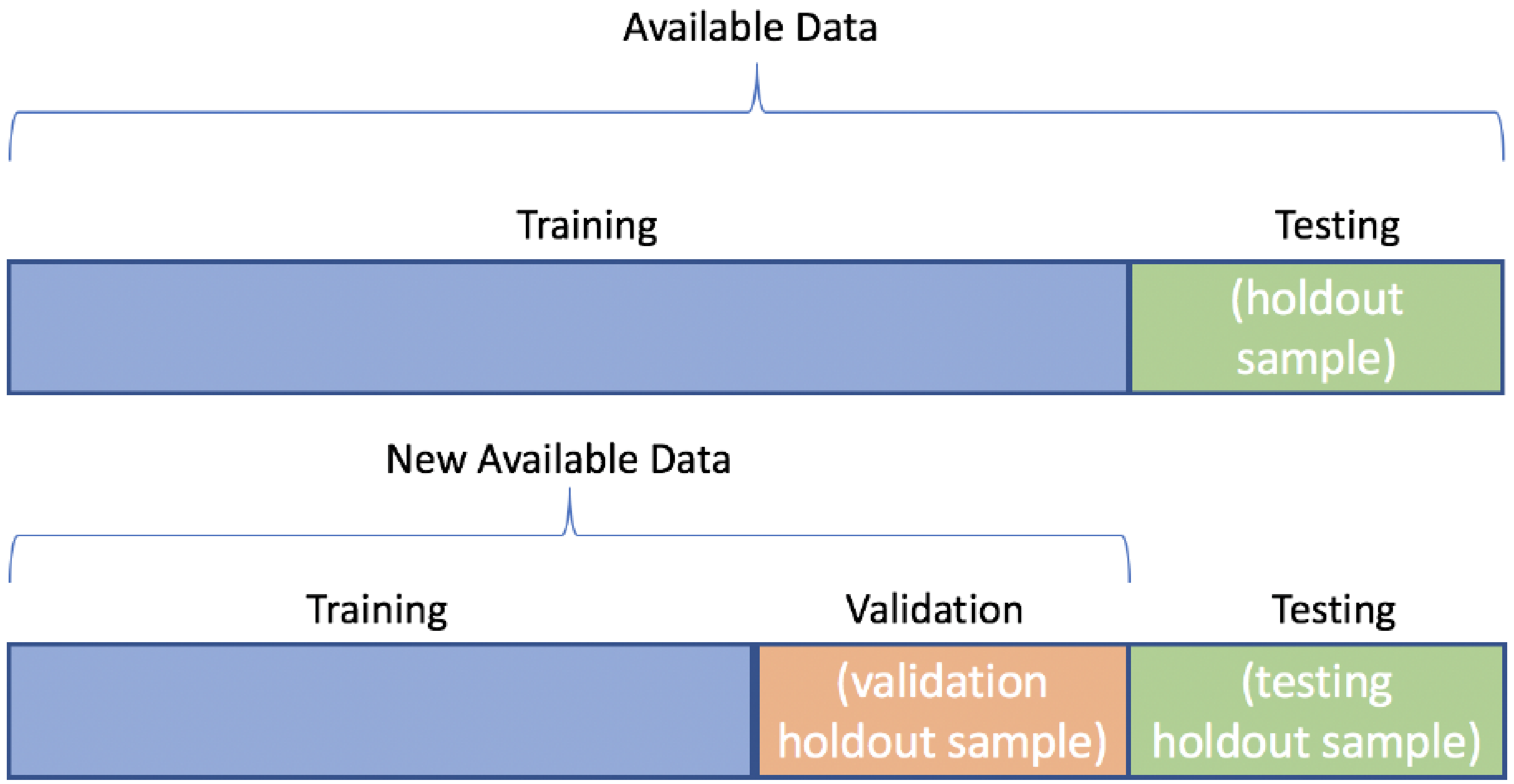
TRAINING

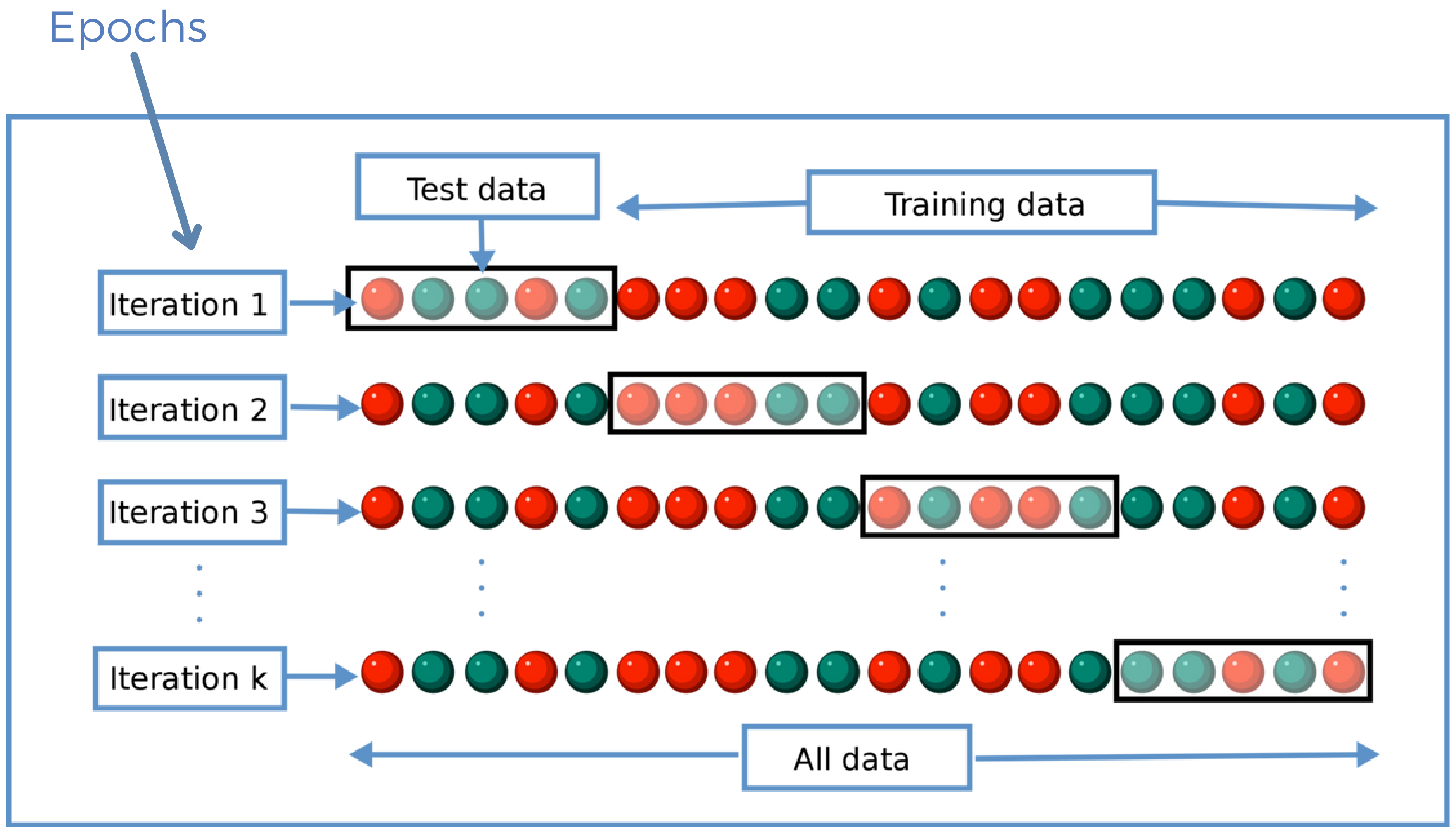
- Learning from the data

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

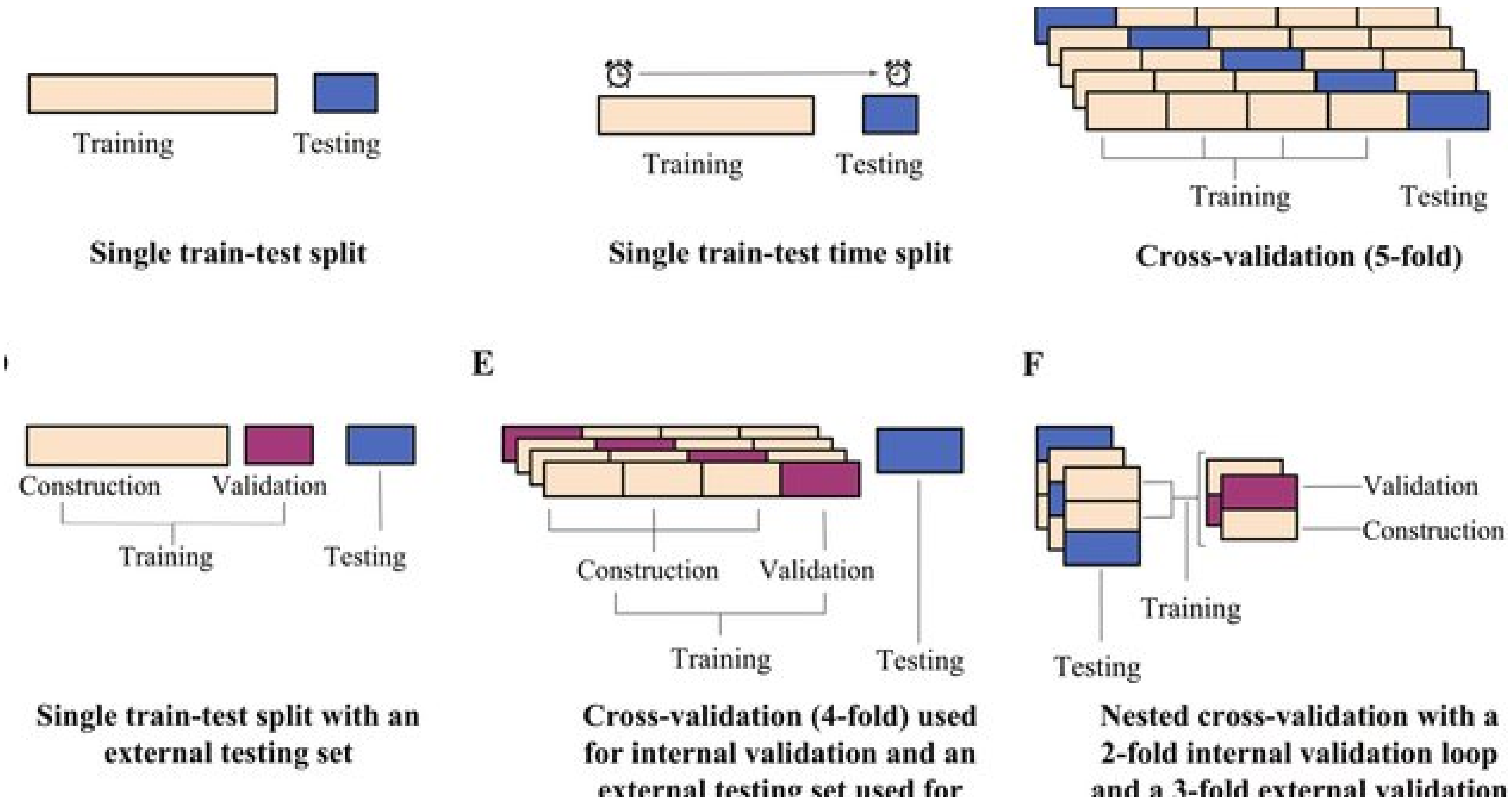
Hyperparameter tuning (choose optimal parameters)

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

SPLITTING

- **data sampling strategies!**

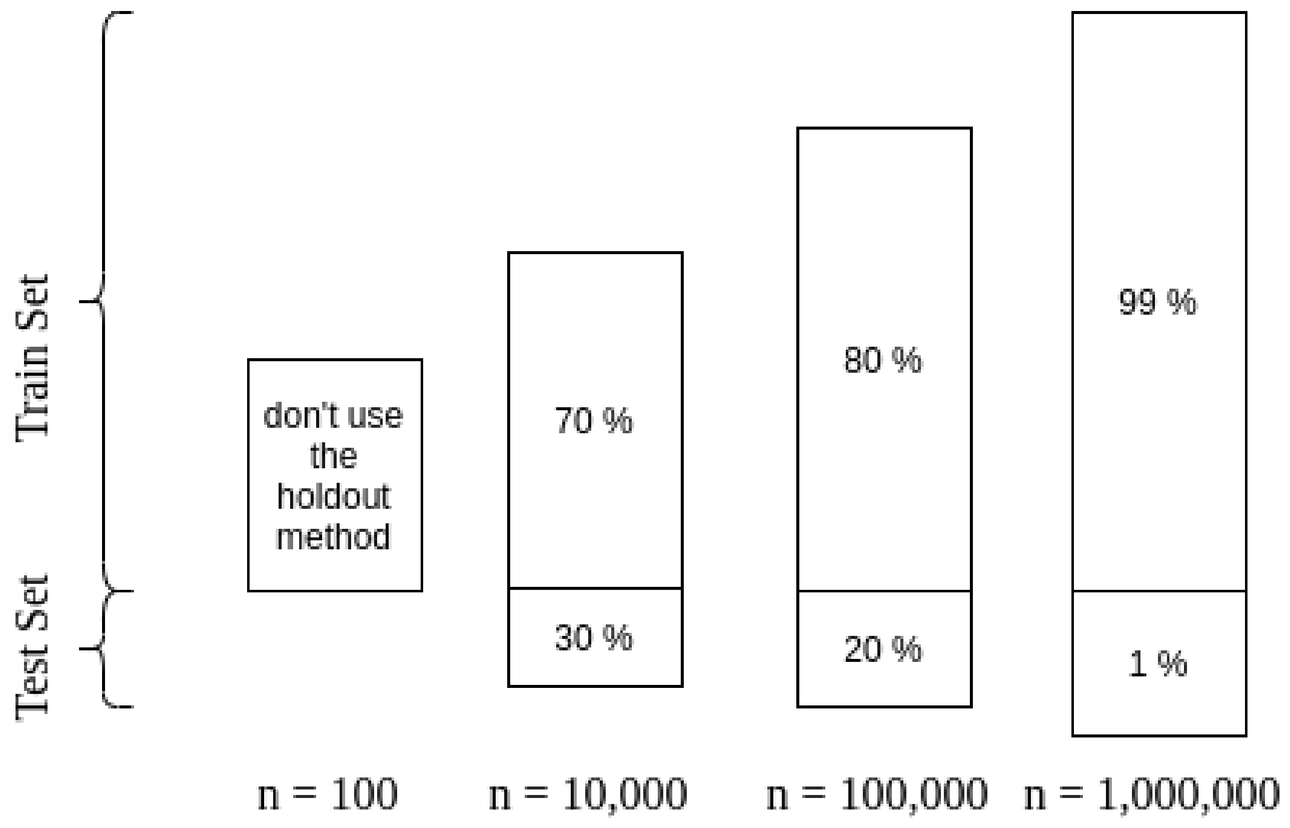
Validation strategies for target prediction methods
(Mathai et al, 2019)

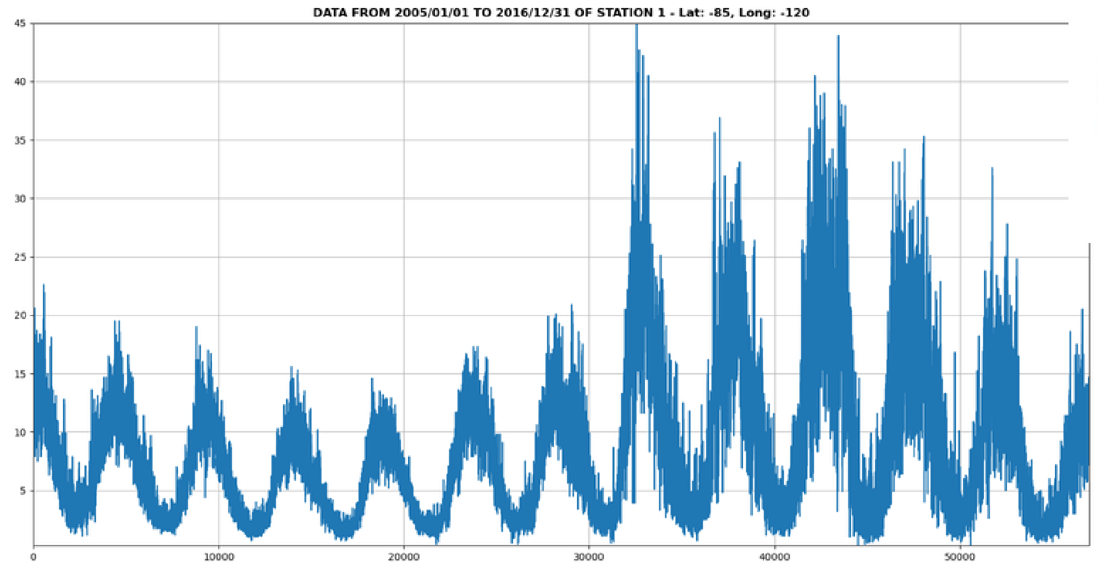


SPLITTING

- **data sampling strategies!**

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

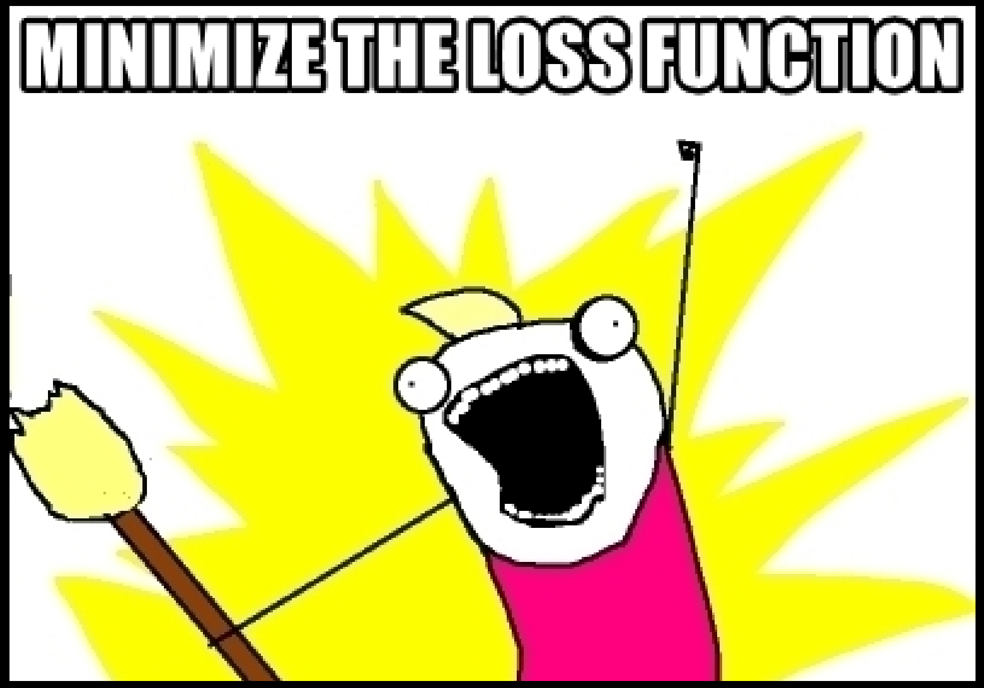


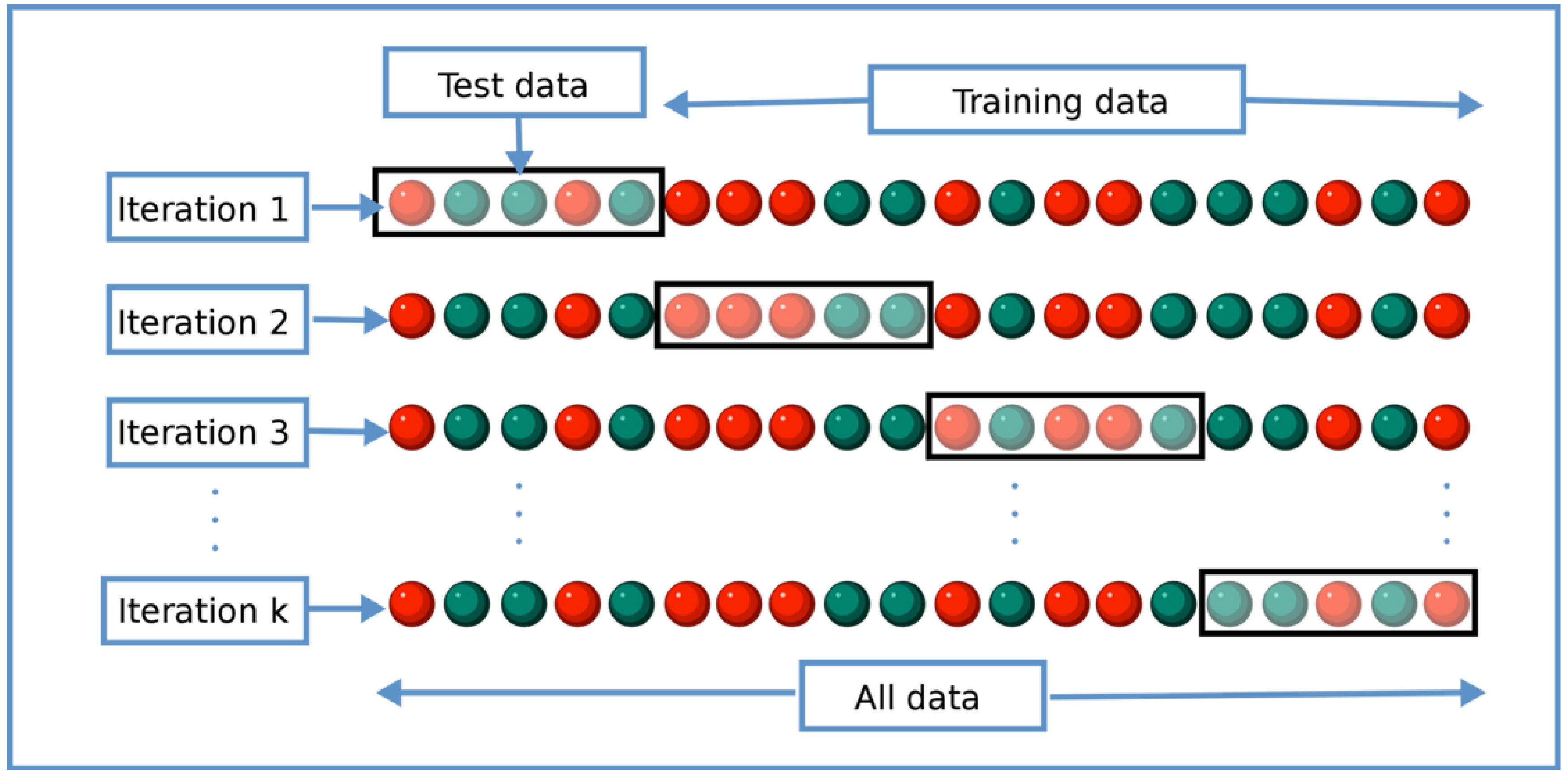


- **Balanced/imbalanced datasets: another story!**

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

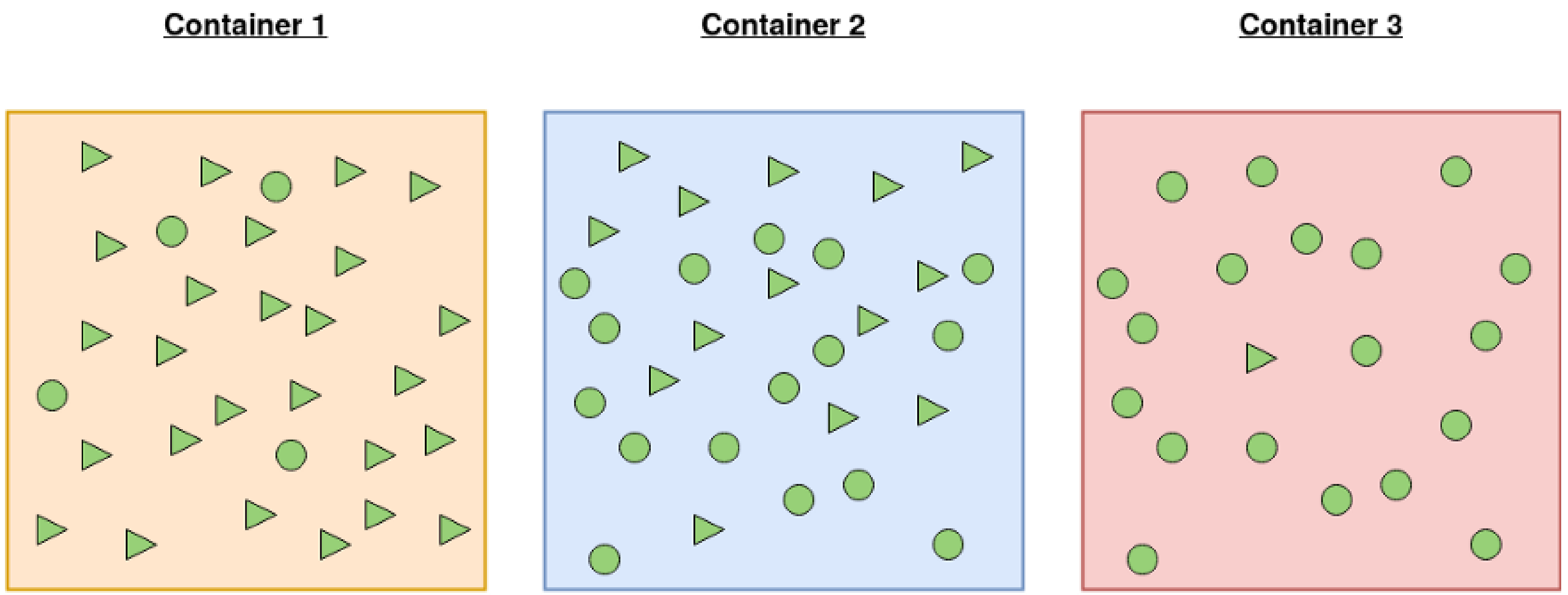
- or "cost function"or "target function"
- Measure the cost of inaccurate predictions
- measure how far an estimated value is from its true value
- is a method of evaluating how well specific algorithm models the given data.
- Goal: Minimize the loss function during the training (optimization problem)

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)

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



#▷ =26, #● = 4   #▷ =14, #● = 16   #▷ = 1 #● = 29

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.

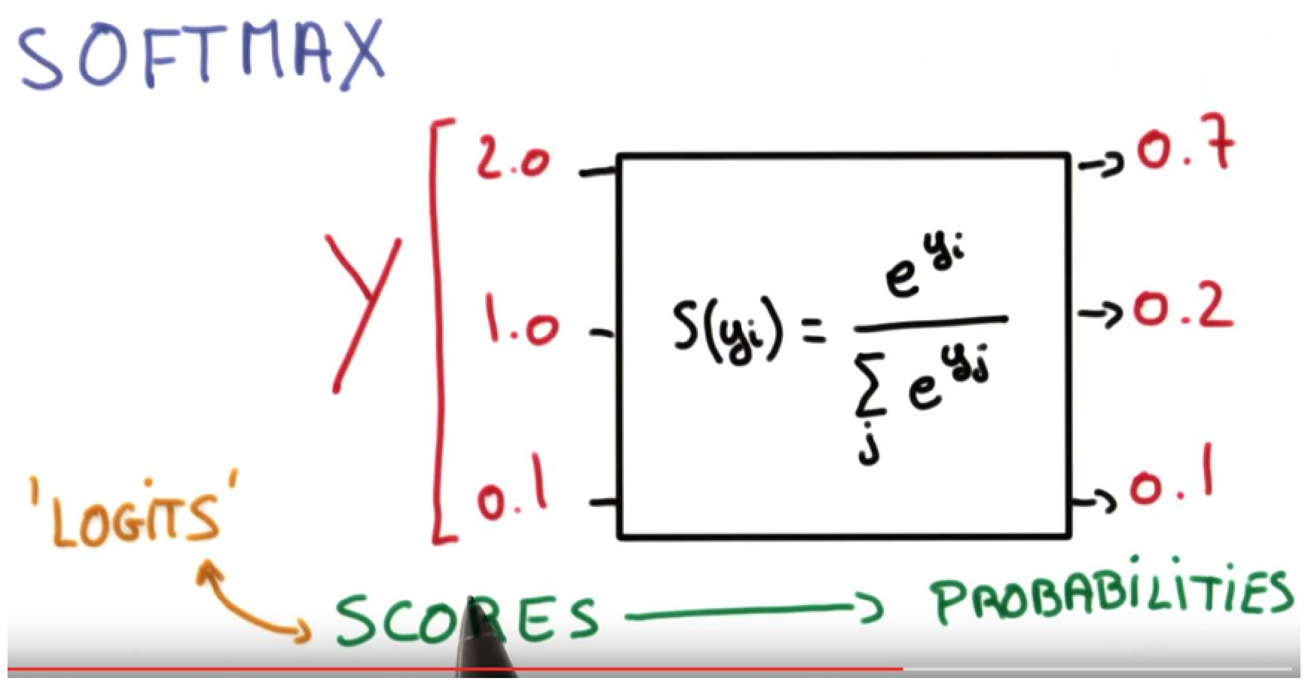
- Cross Entropy

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

- Binary Cross Entropy

$$L = -\sum_{i=1}^{2} t_i \log(p_i)$$
$$= -[t\log(p) + (1-t)\log(1-p)]$$

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

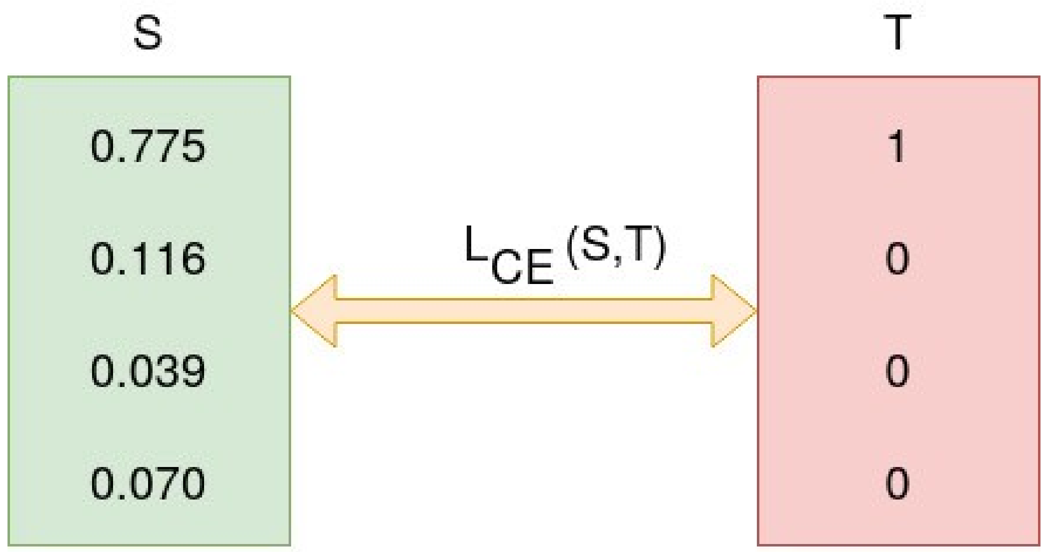
$$L = -\frac{1}{N}\left[\sum_{j=1}^{N} [t_j \log(p_j) + (1-t_j)\log(1-p_j)]\right]$$

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.



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

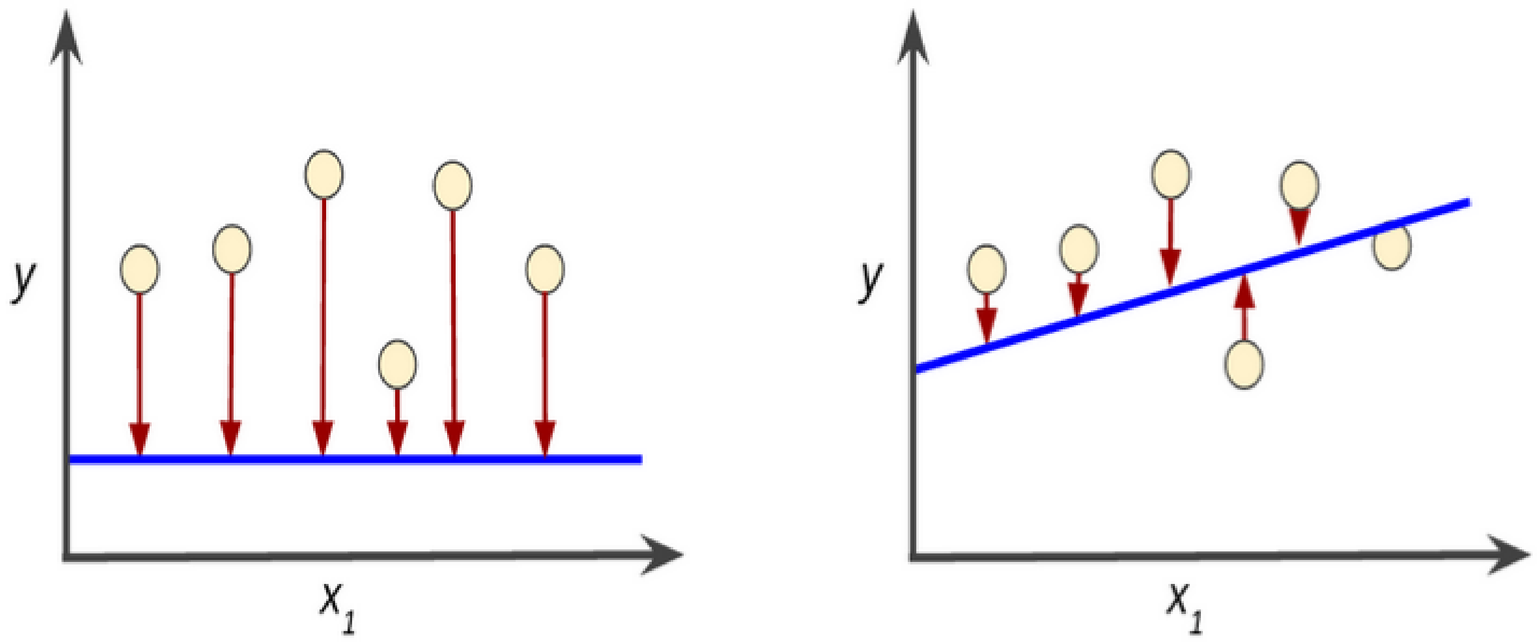
- Notice that when actual label is 1 (ti = 1), second half of function =0 whereas in case actual label is 0 (ti = 0) first half is dropped off. In short, we are just multiplying the log of the actual predicted probability for the ground truth class.

- 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

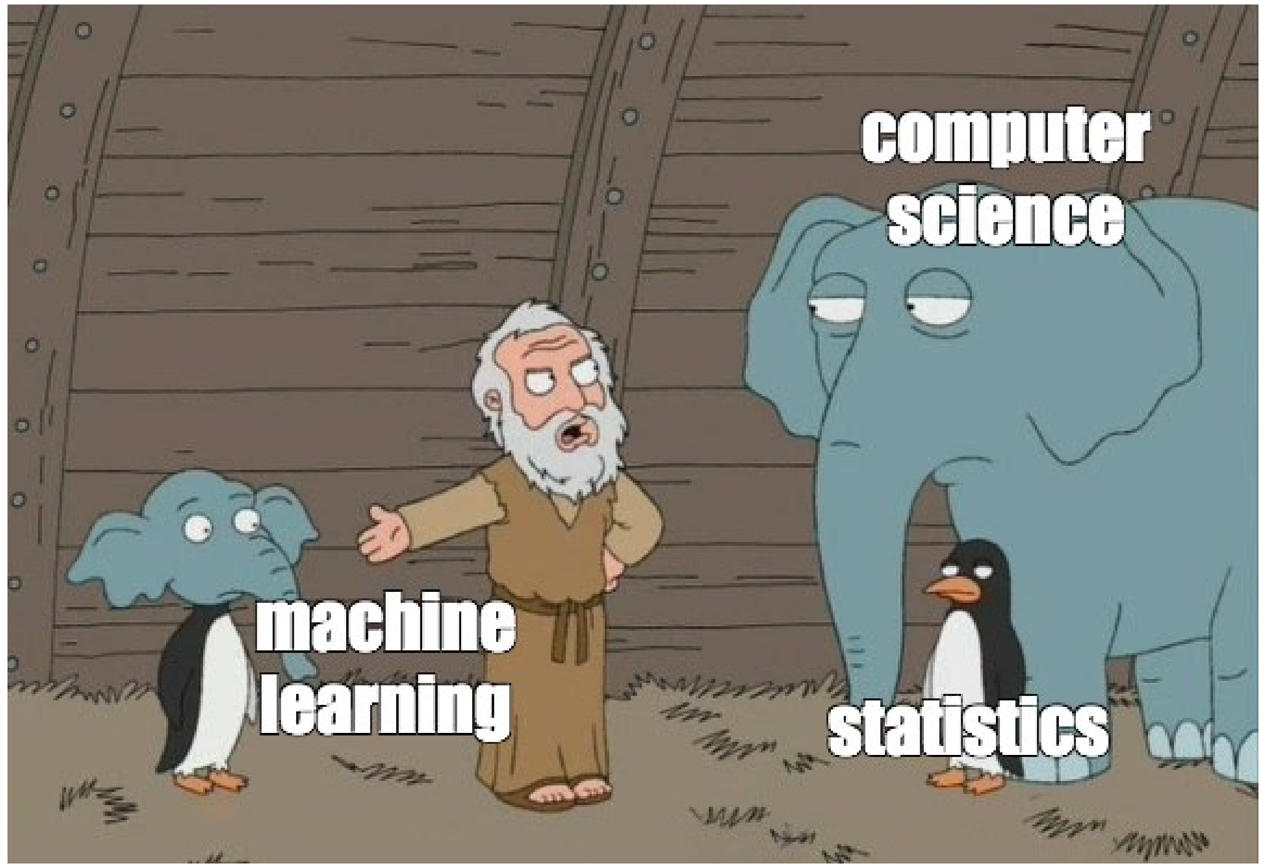
TSWC, 2022

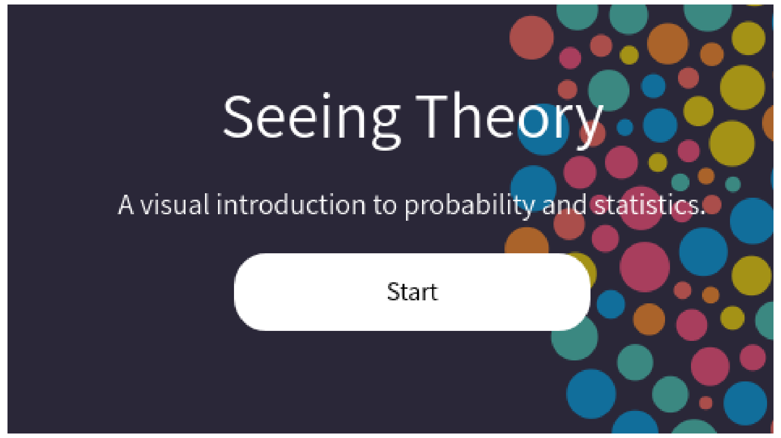
LOSS FUNCTION **REGRESSION**

- 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.

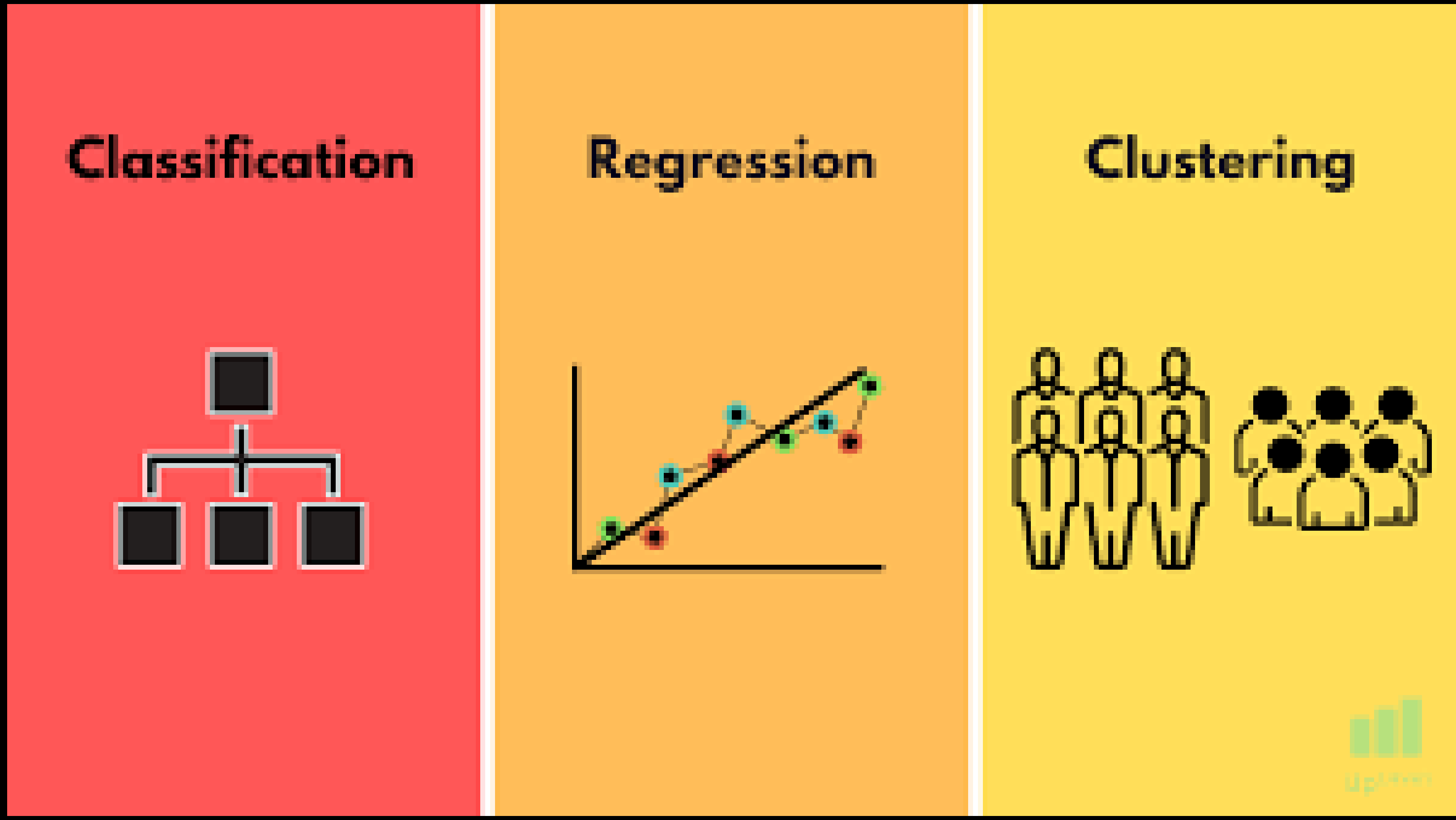


$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}$$





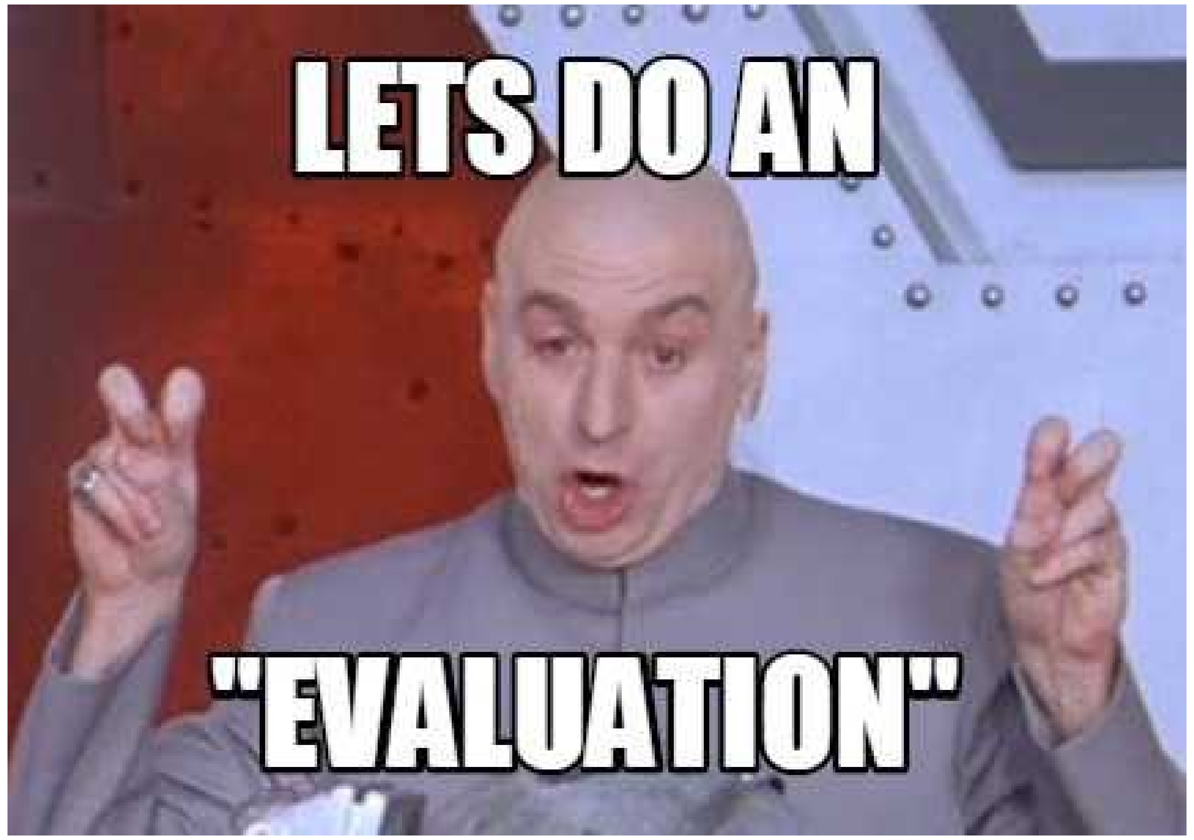
https://seeing-theory.brown.edu



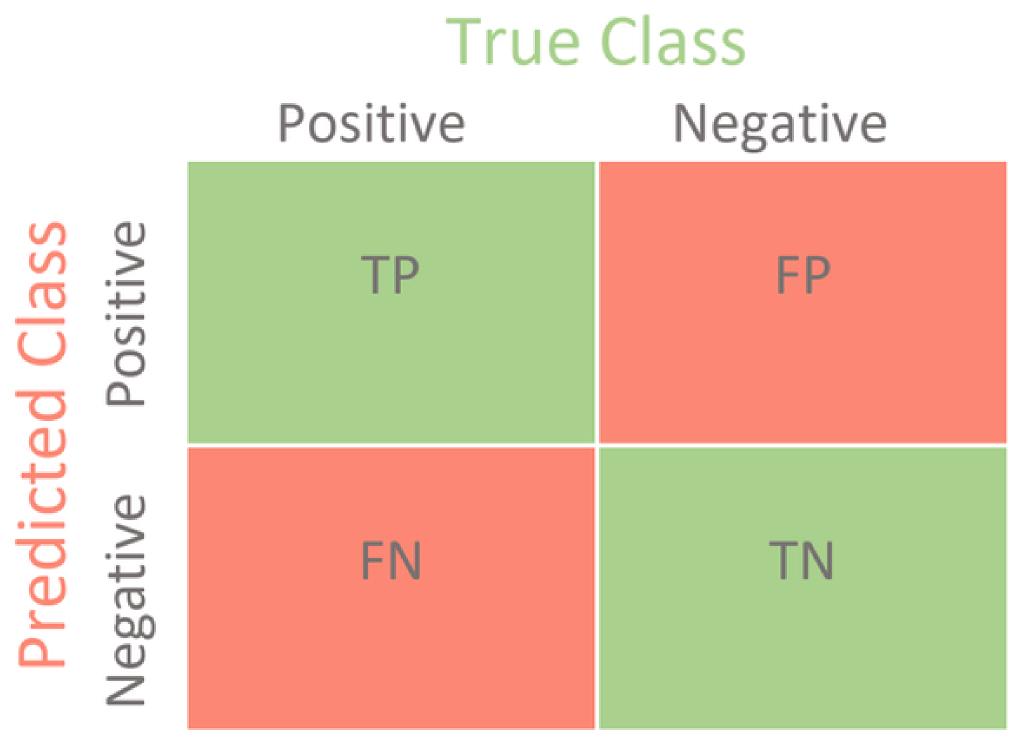
- Classification: confusion matrix (binary), AUC & ROC (multiclass)
- Regresion: MAE,MSE,RMSE (on the test set)
- Clustering: ? ? ? ?

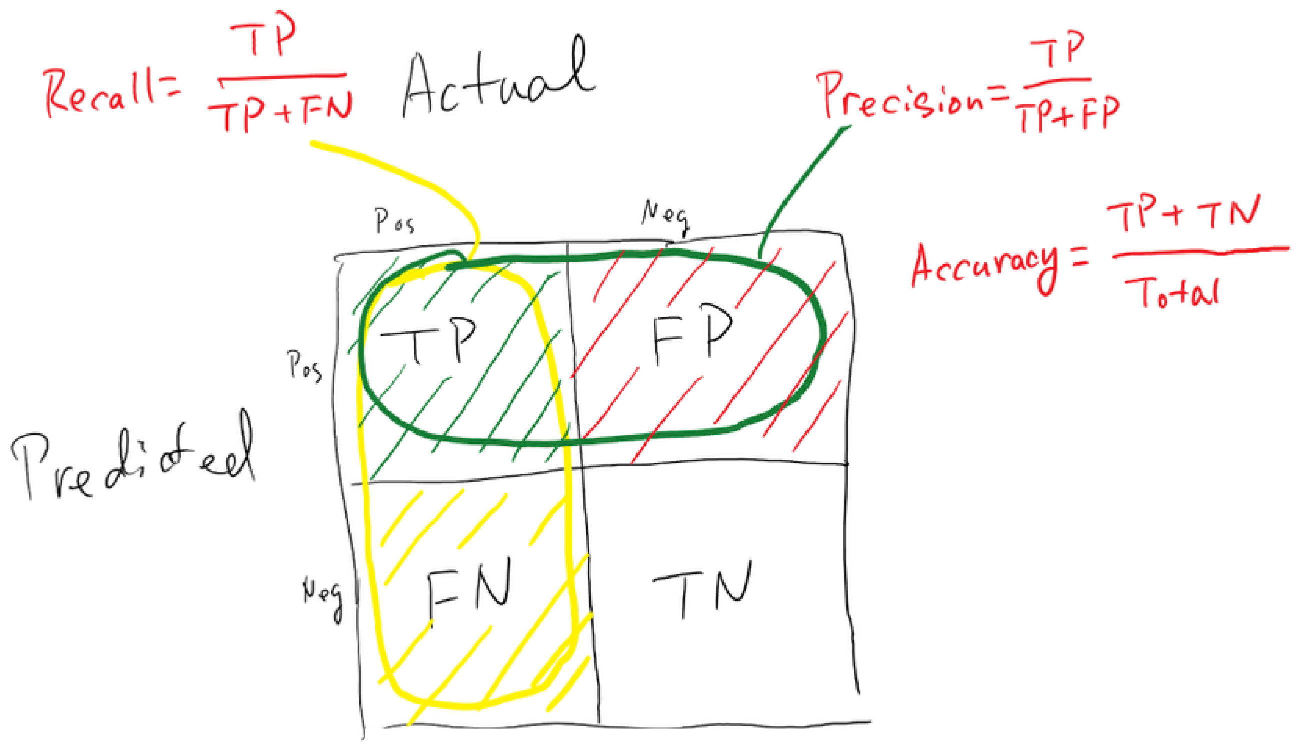
PERFORMANCE

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



CLASSIFICATION





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

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

$$Accuracy = (TP + TN)/(TP + TN + FP + FN)$$

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

$$Precision = TP/(TP + FP)$$

- 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.

$$Recall = TP/(TP + FN)$$

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

Actual \ Predicted	Elephant	Monkey	Fish	Lion
Elephant	25	3	0	2
Monkey	3	53	2	3
Fish	2	1	24	2
Lion	1	0	2	71

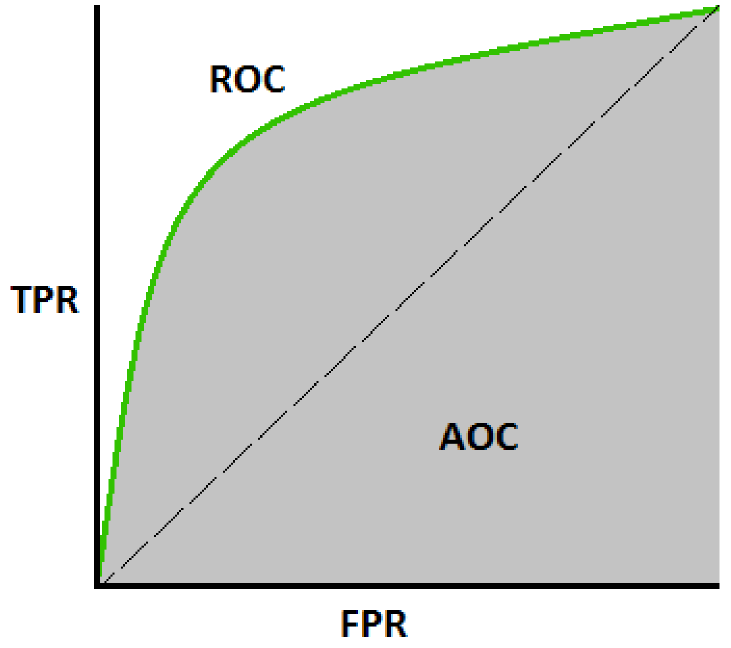
Multiclass

- 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

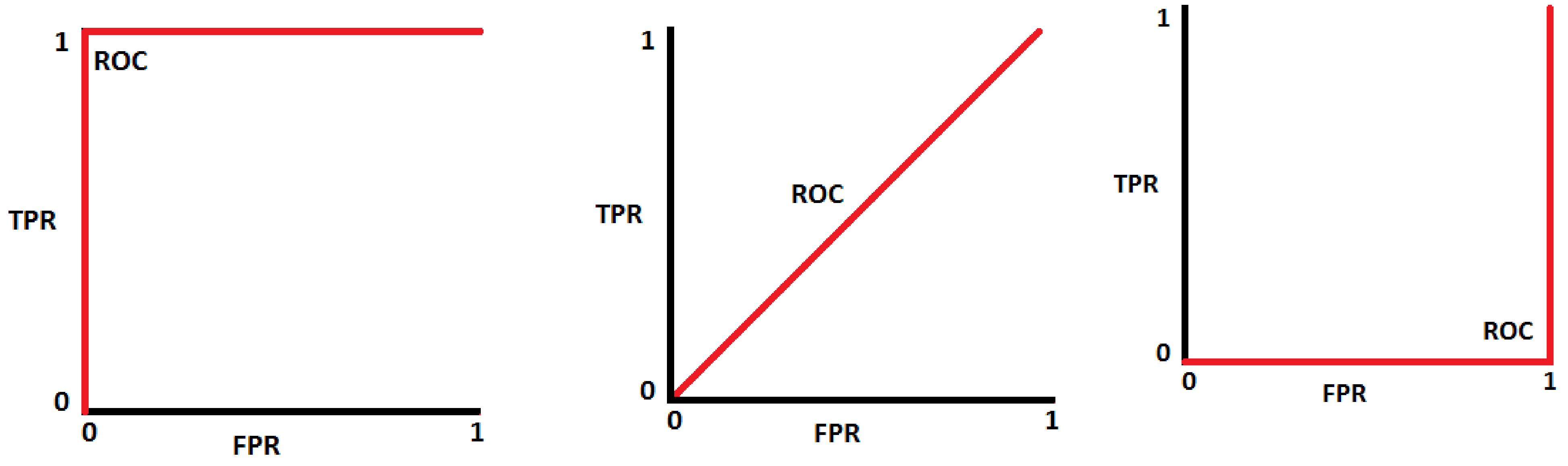


$$\text{TPR /Recall / Sensitivity} = \frac{TP}{TP + FN}$$

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

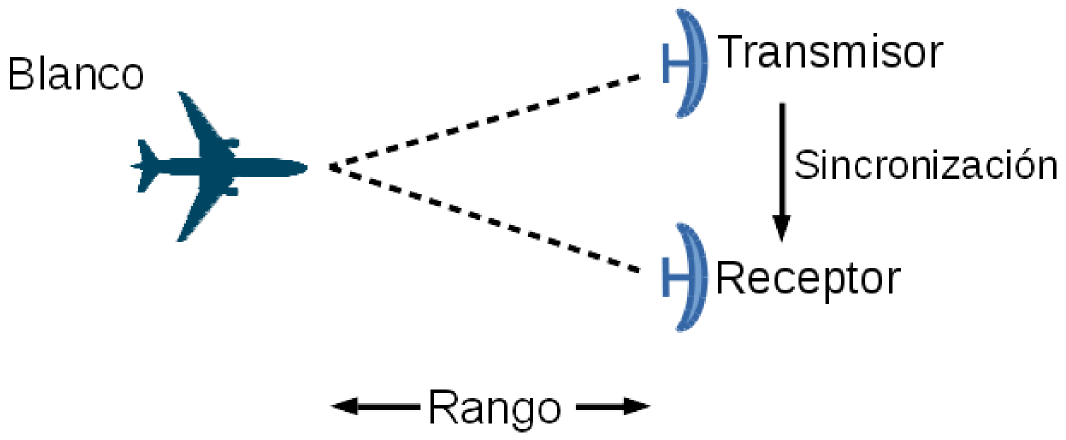
$$\text{FPR} = 1 - \text{Specificity}$$

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



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

CLASSIFICATION (Application: Echo detection)

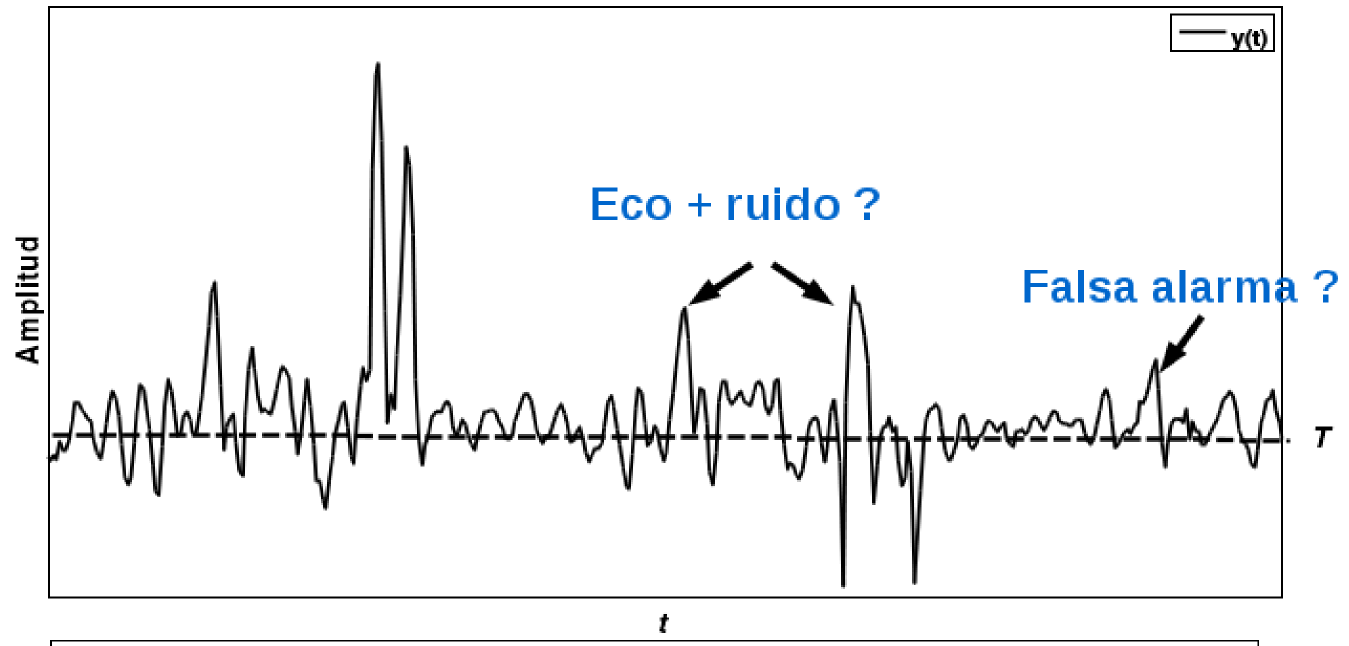


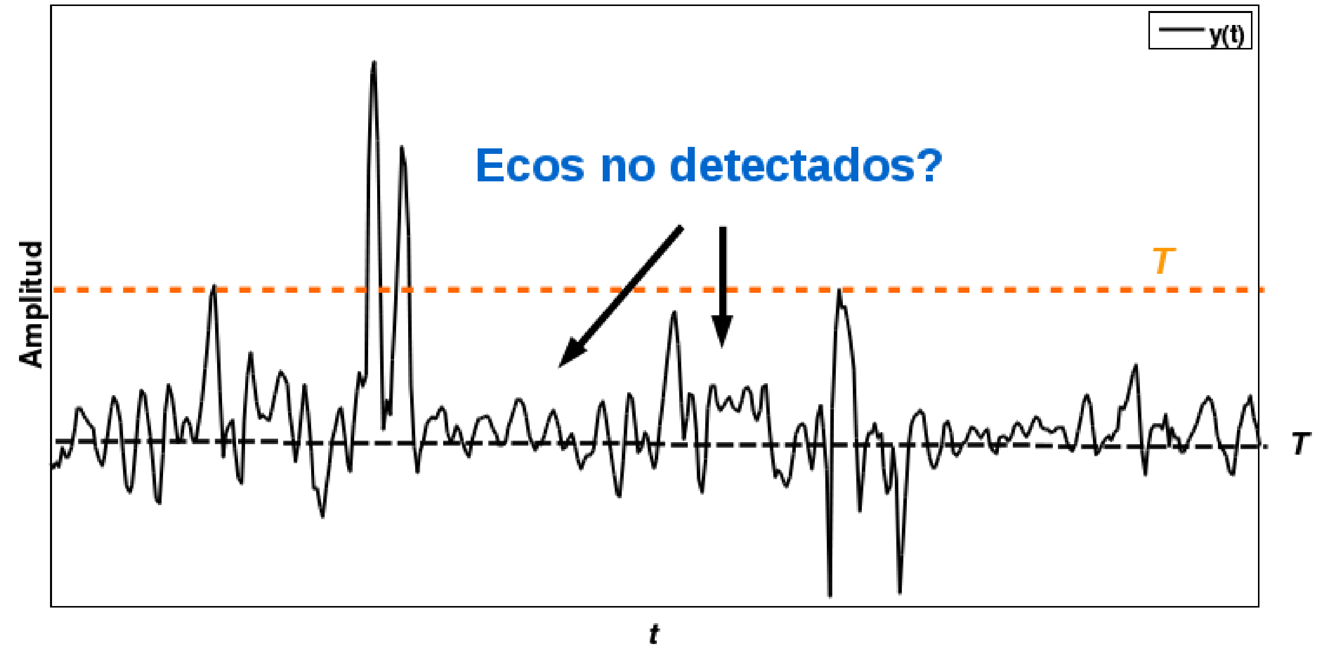
H0: noise/interference/jamming only

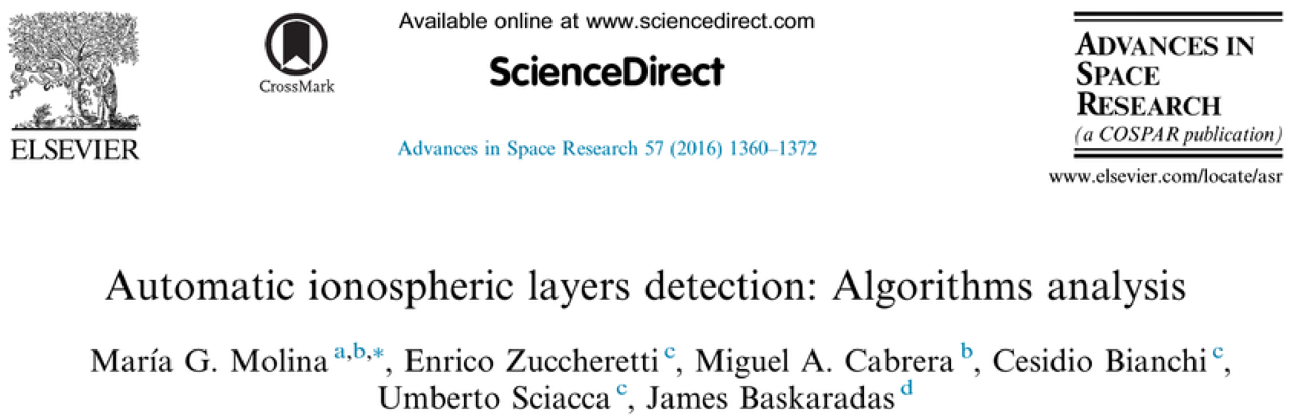
H1: noise/interference/jamming + echo from a target

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

Cada yi>T es una detección







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

Si Pd ↑, entonces Pfa también ↑

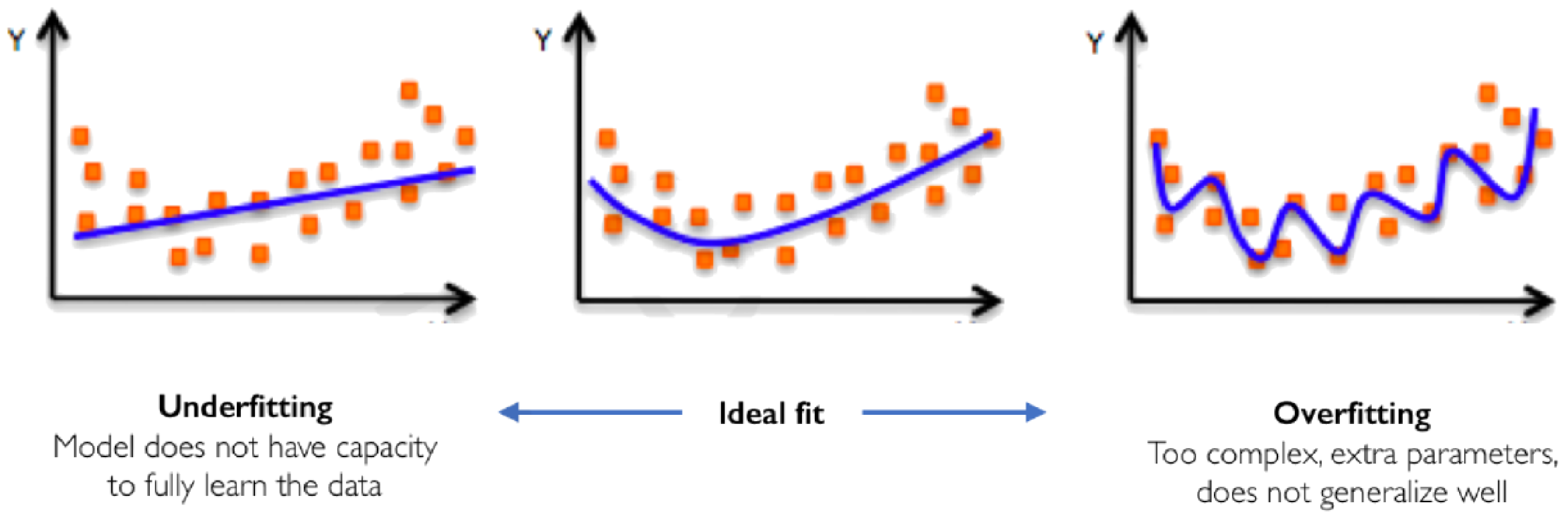
Maximizar Pd sujeta a Pfa <= β

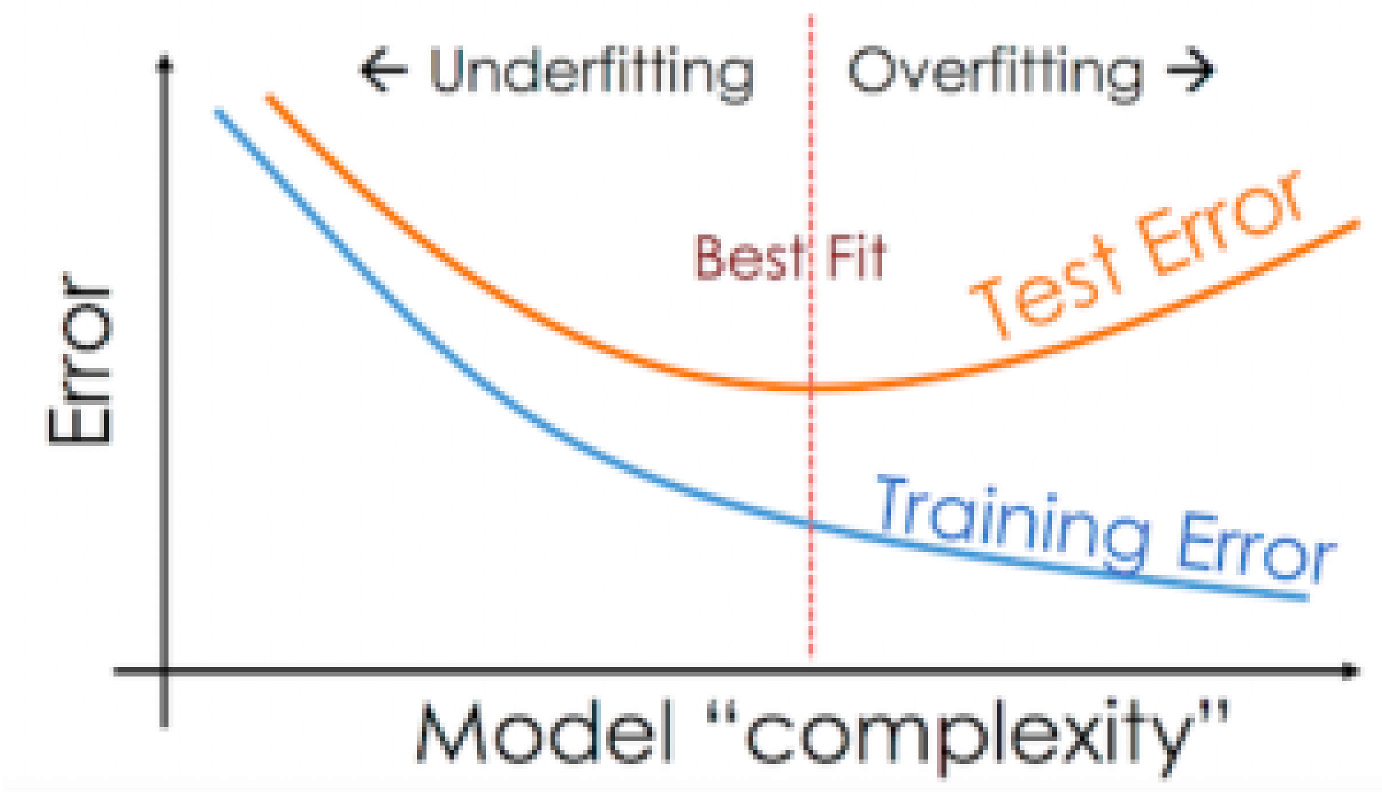
$$T = \alpha\ \hat{y}$$

TSWC, 2022

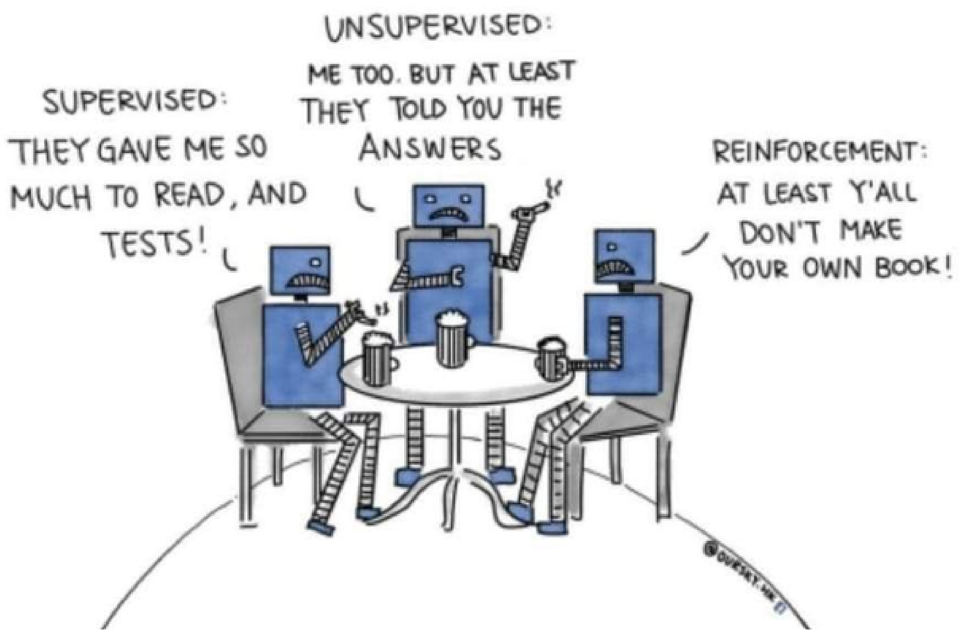
- How to think the problem from ML point of view?

PERFORMANCE





- There are techniques to overcome overfitting



MACHINE LEARNING

Algorithms

Supervised | **Unsupervised**

Regression
- Linear
- Polynomial

Clustering
- K-means
- PCA
- SVD

Continuous data

Decision Tree

Random Forest

Categorical data

Classification
- KNN
- Logistic Regression
- Naive-Bayes
- SVM

Associtіon Rule Learning

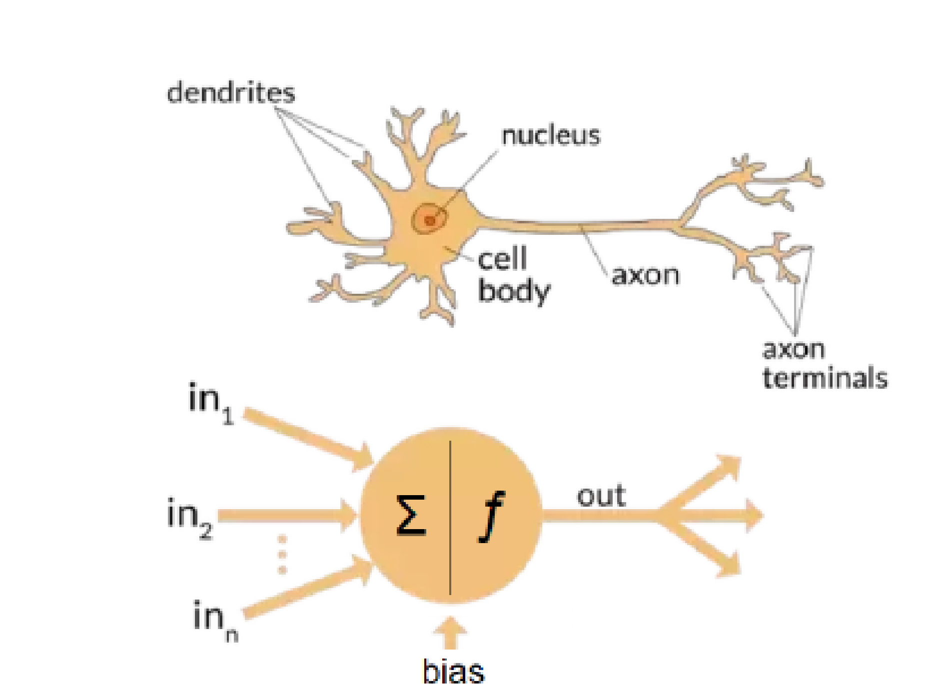
Hidden Markov Model

Artificial Neural Networks

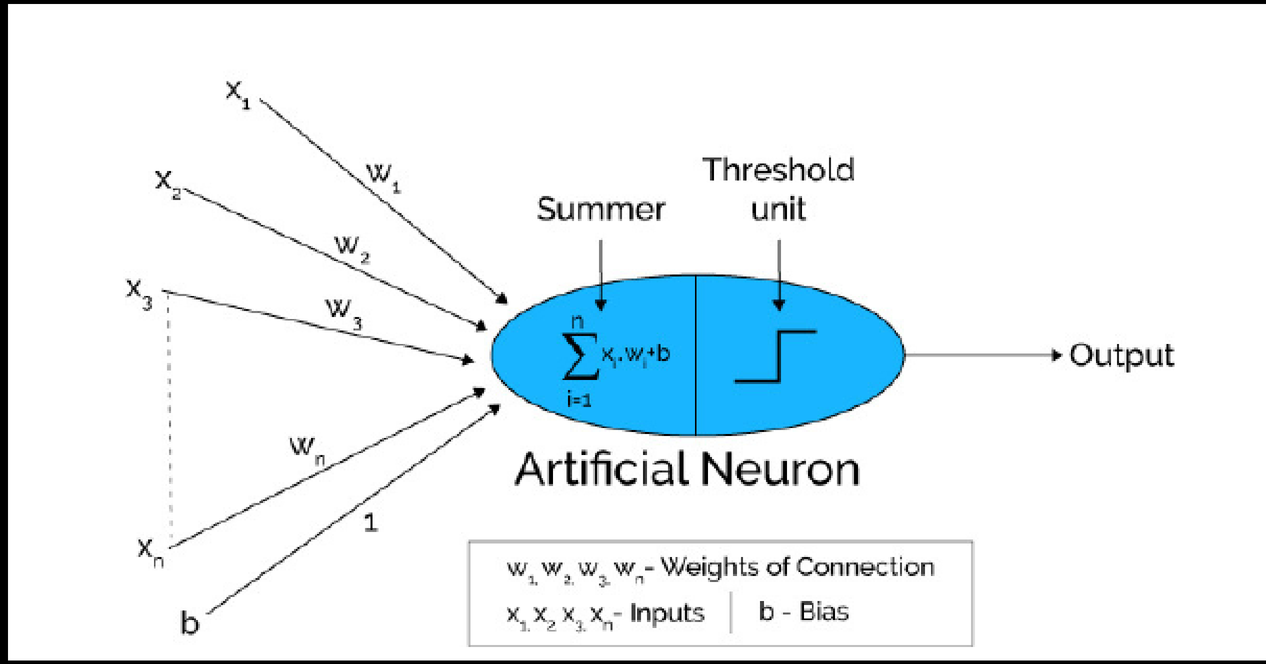
Artificial Neural Networks

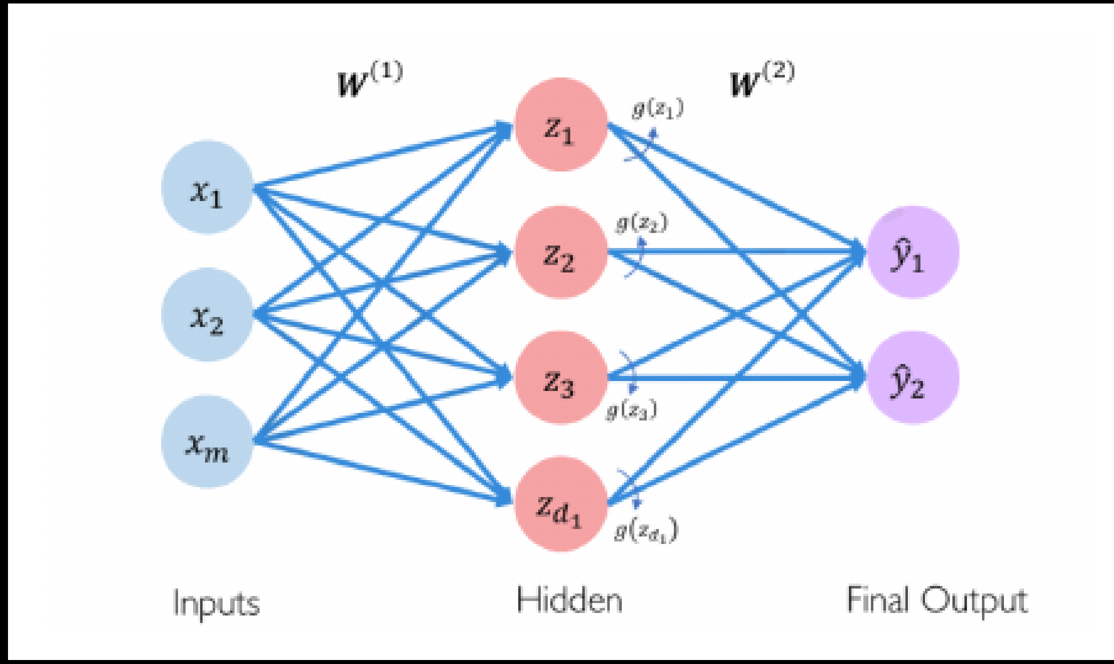
TSWC, 2022

ANN



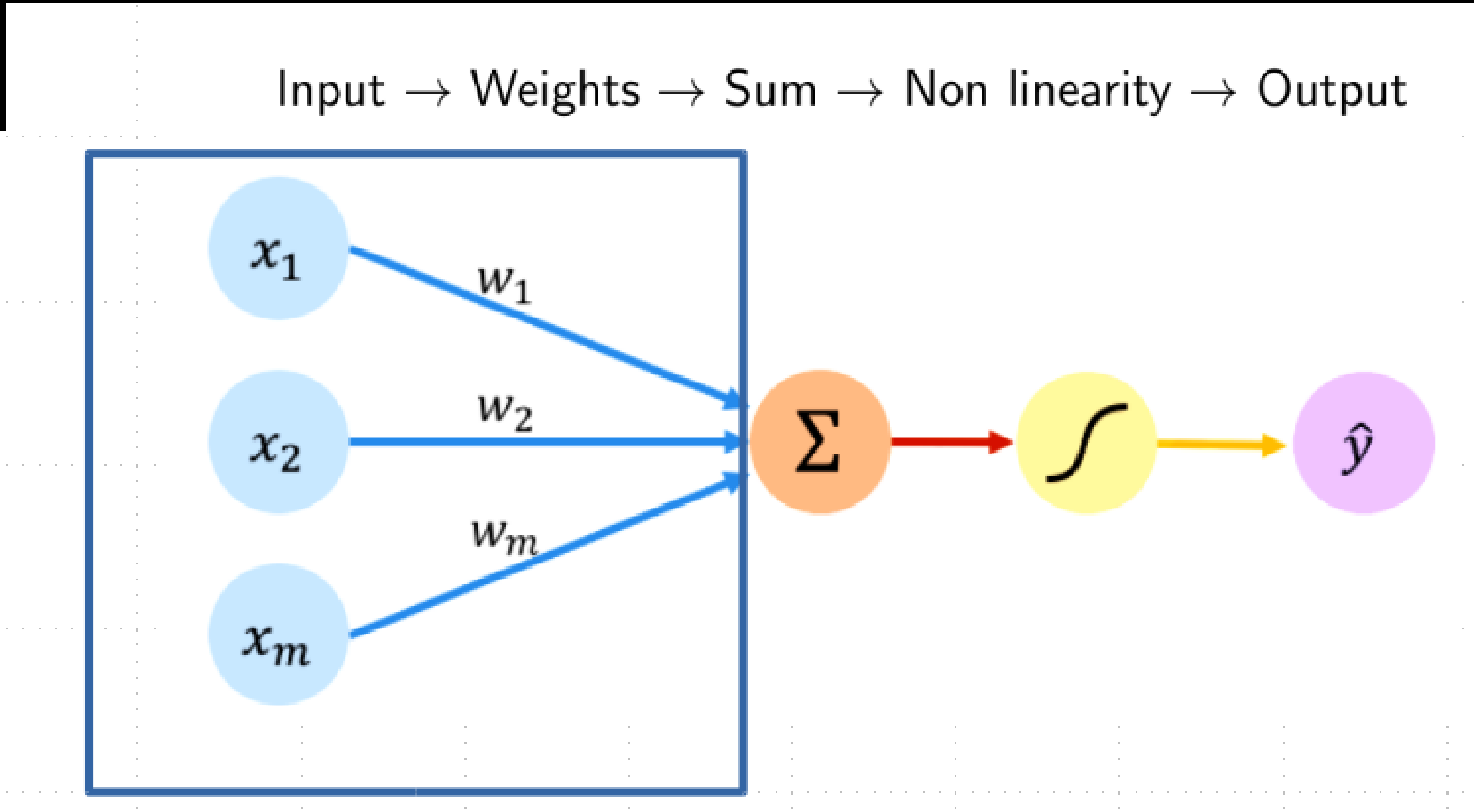
- Data-driven modeling
- Inspired in brain neural networks
- Solving wide number of complex problems: facial recognition, handwrite recognition, weather forecasting, etc.
- Black-box modelling (?) based in the composition of interconnected neurons (neural network)





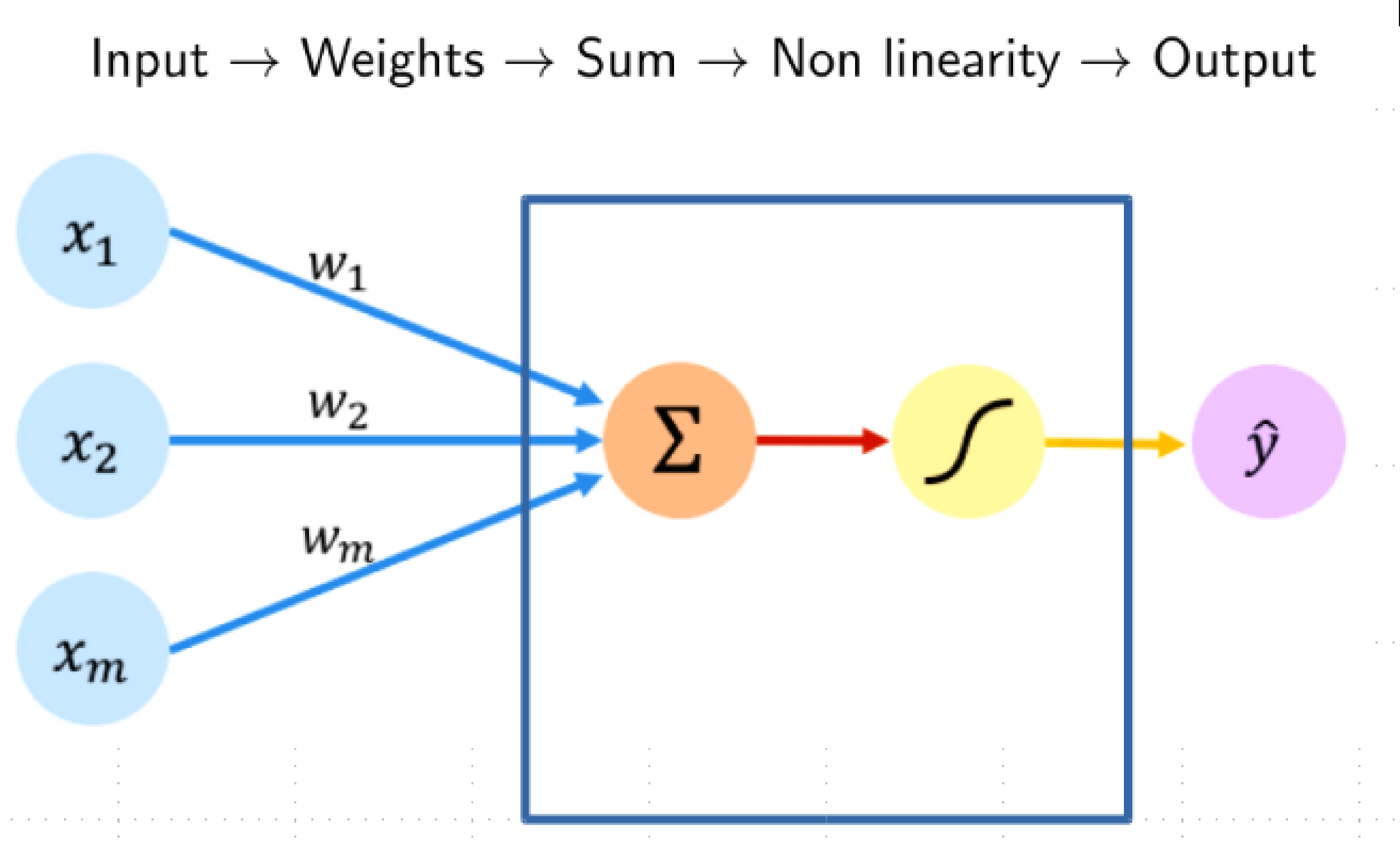
Perceptron

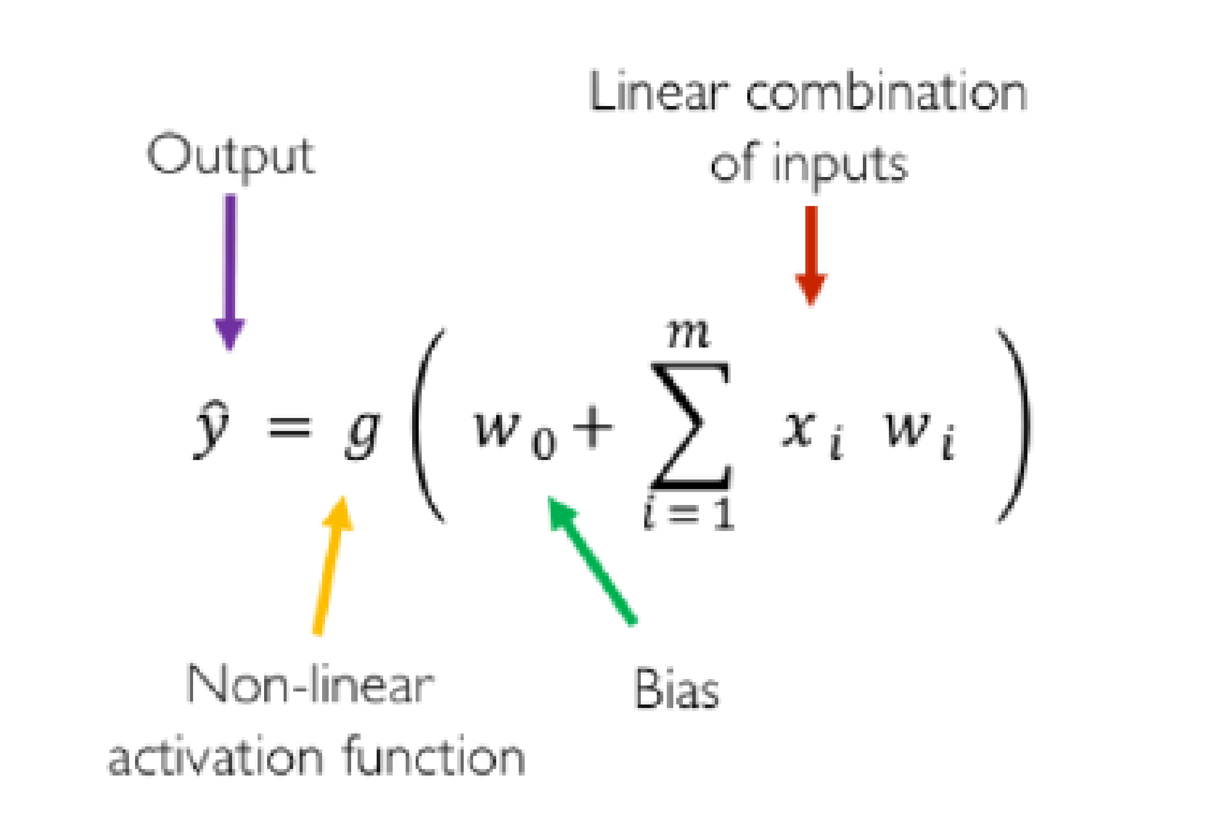
- Unidad fundamental para construir redes neuronales



- Cada una de las entradas corresponde a una característica (feature)
- A cada una de las xi se les aplica un peso (w) (ponderación de los valores de la entrada)
- Los pesos son las vbles que se ajustan durante el entrenamiento

Perceptron

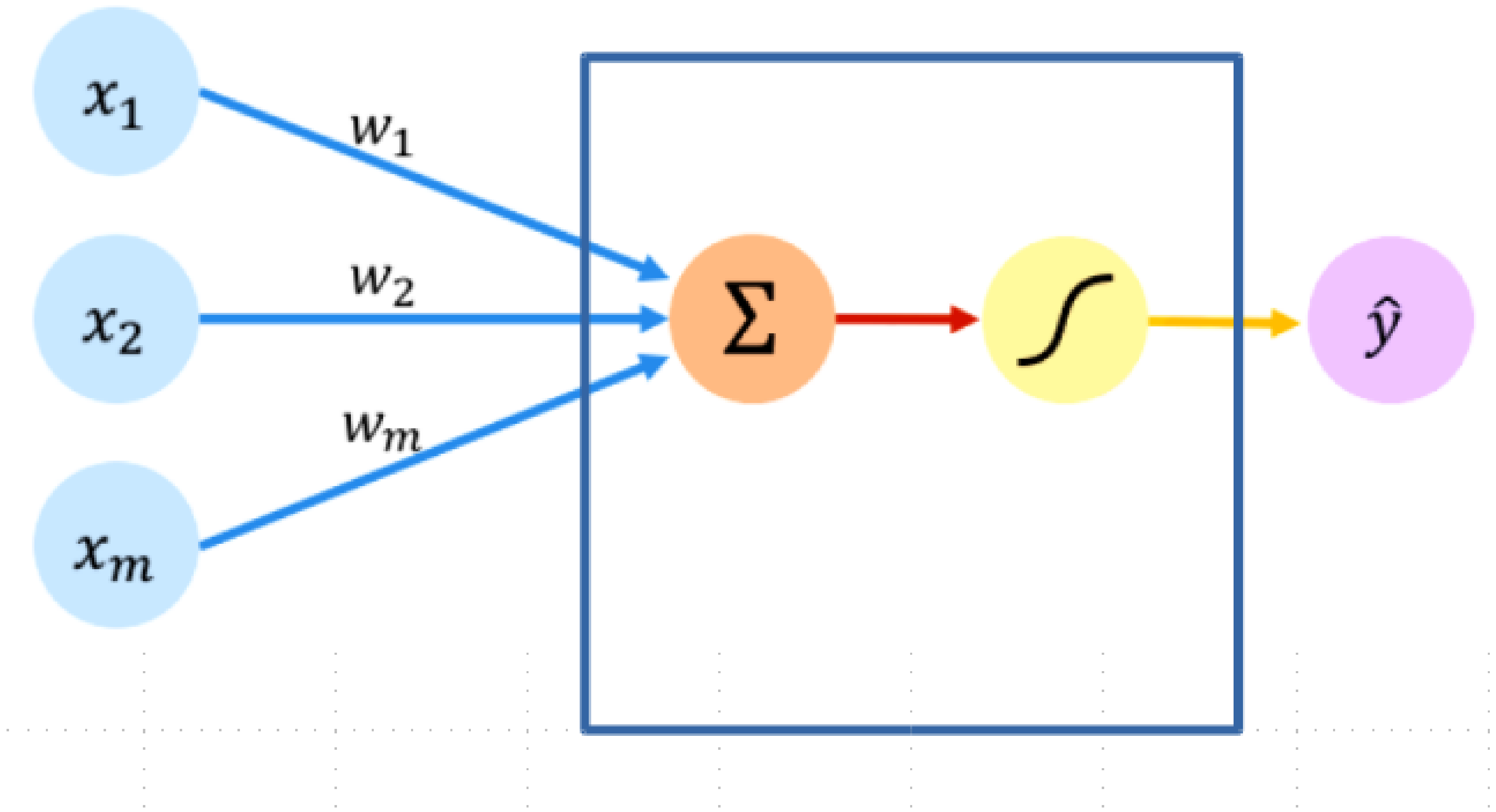




- Capa oculta (hidden layer).
- El bias es un término de ajuste
- La función de activación se aplica a la combinación lineal de las entradas ponderadas teniendo en cuenta el bias

Perceptron

Input → Weights → Sum → Non linearity → Output



$$\hat{y} = g\left(w_0 + \sum_{i=1}^{m} x_i \, w_i \right)$$

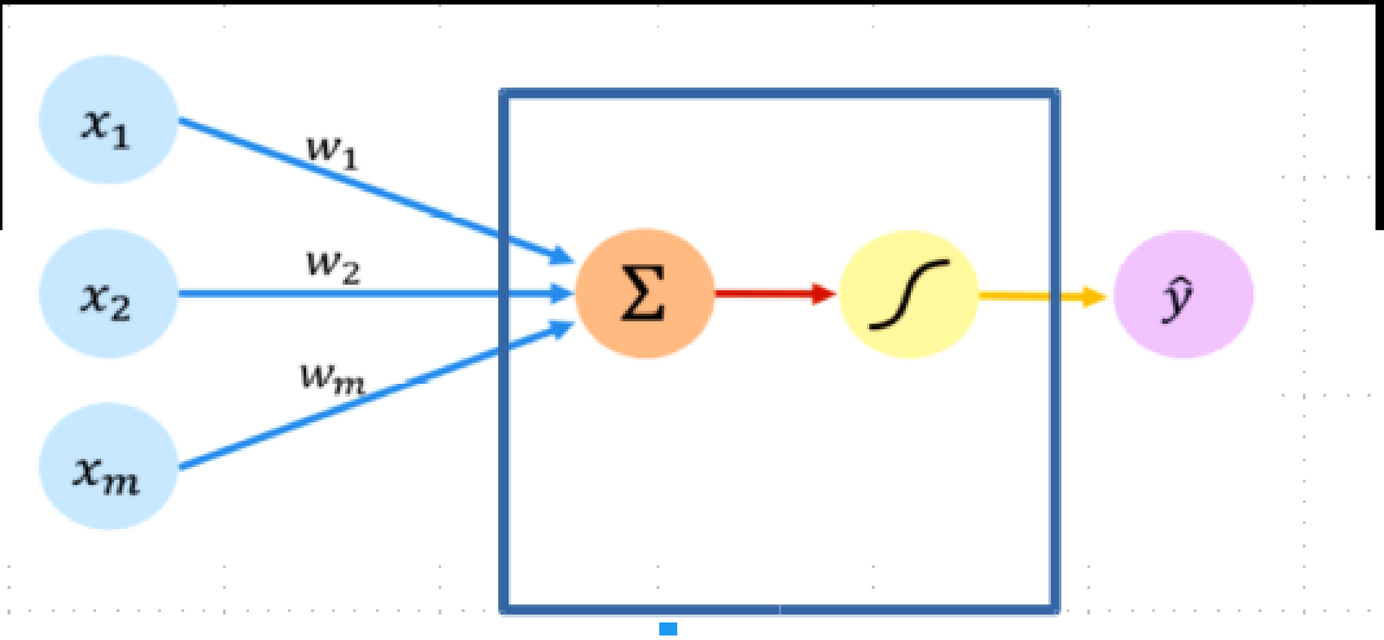
$$\hat{y} = g\left(w_0 + \boldsymbol{X}^T \boldsymbol{W} \right)$$

where: $\boldsymbol{X} = \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix}$ and $\boldsymbol{W} = \begin{bmatrix} w_1 \\ \vdots \\ w_m \end{bmatrix}$

Activation function

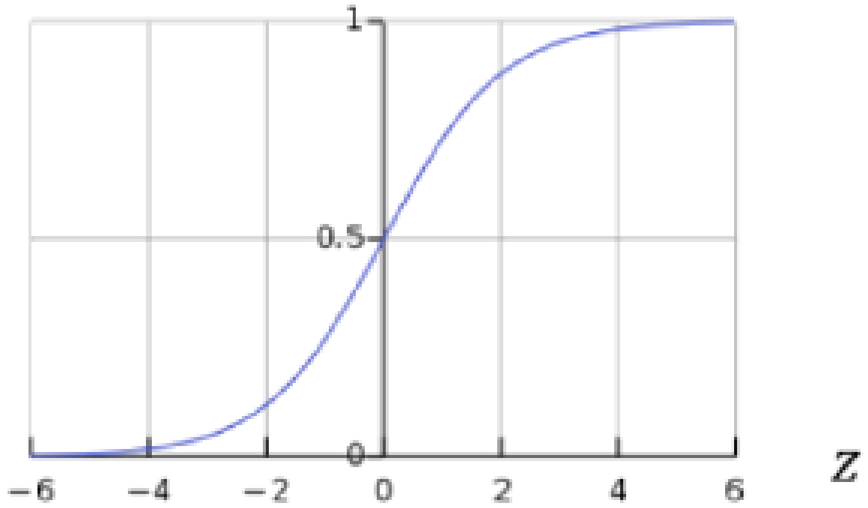
- Agrega no-linealidd al modelo

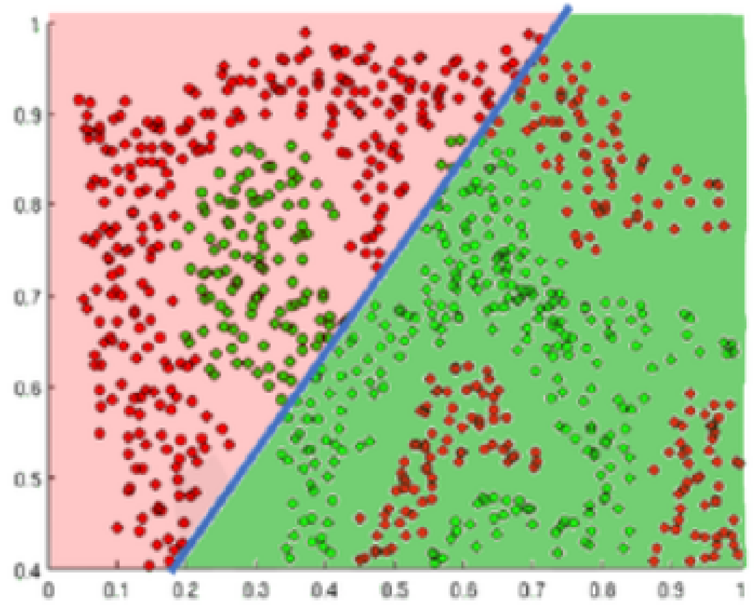
$$\hat{y} = g(w_0 + X^T W)$$



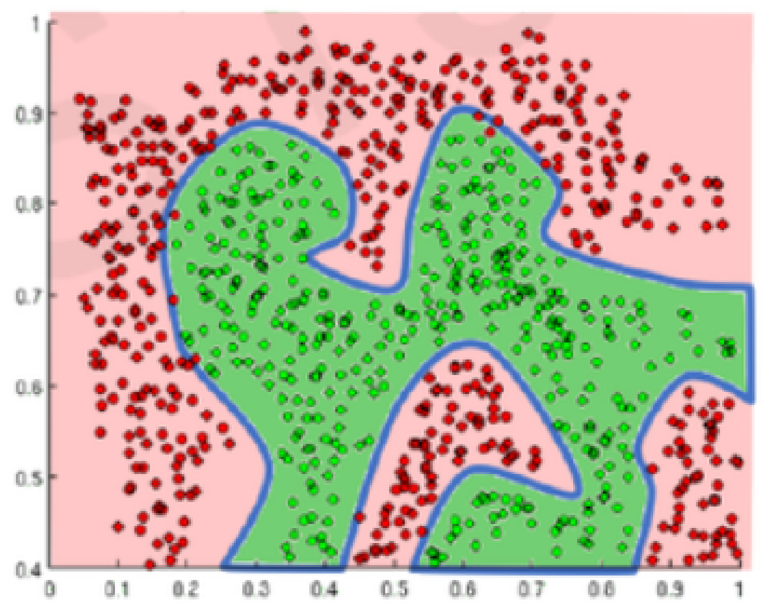
- Una de las más usadas en la función sigmoide.
- Produce resultados entre 0 y 1

$$g(z) = \sigma(z) = \frac{1}{1 + e^{-z}}$$





Linear activation functions produce linear decisions no matter the network size

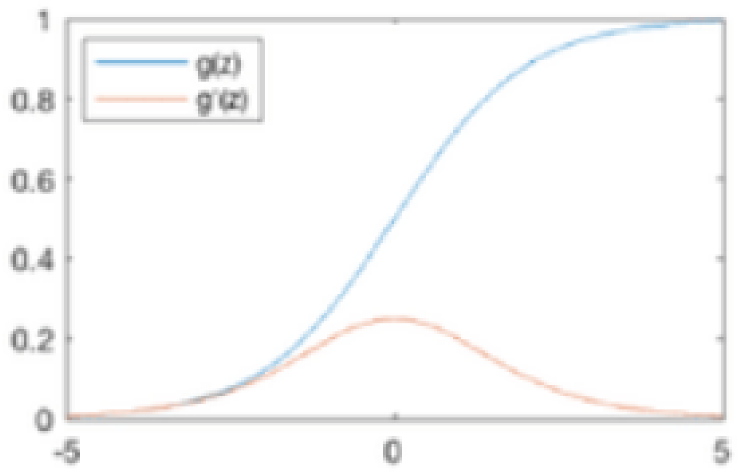


Non-linearities allow us to approximate arbitrarily complex functions

Activation function

$$\hat{y} = g(w_0 + X^T W)$$

Sigmoid Function

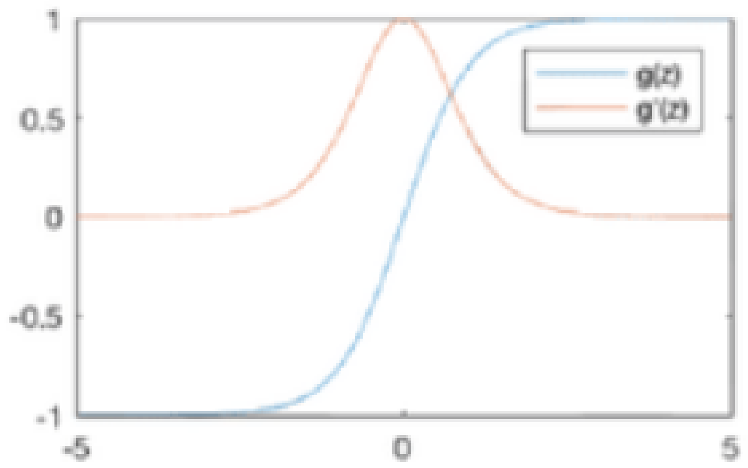


$$g(z) = \frac{1}{1 + e^{-z}}$$

$$g'(z) = g(z)(1 - g(z))$$

`tf.math.sigmoid(z)`

Hyperbolic Tangent

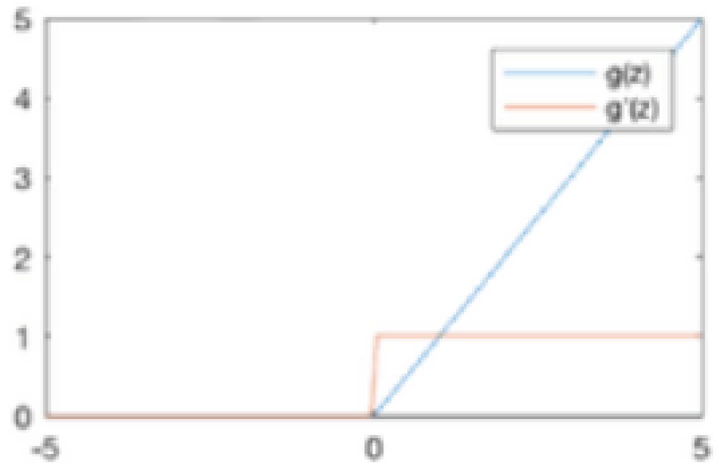


$$g(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$$

$$g'(z) = 1 - g(z)^2$$

`tf.math.tanh(z)`

Rectified Linear Unit (ReLU)



$$g(z) = \max(0, z)$$

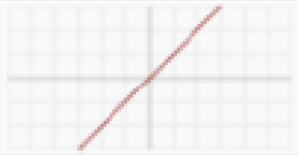
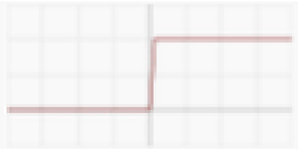
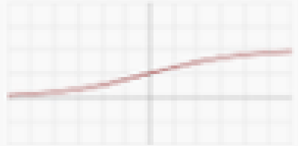
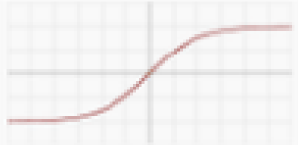
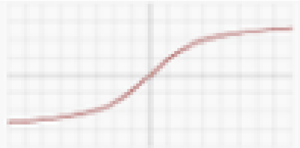
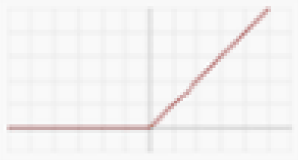
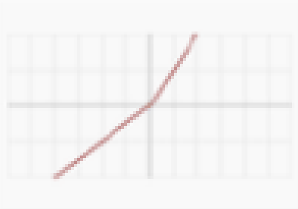
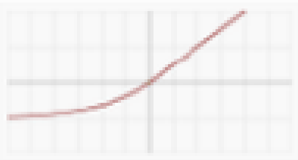
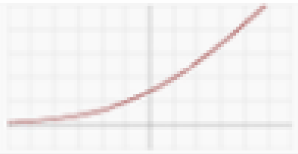
$$g'(z) = \begin{cases} 1, & z > 0 \\ 0, & \text{otherwise} \end{cases}$$

`tf.nn.relu(z)`

Activation function

- Adds no-linearity to the model

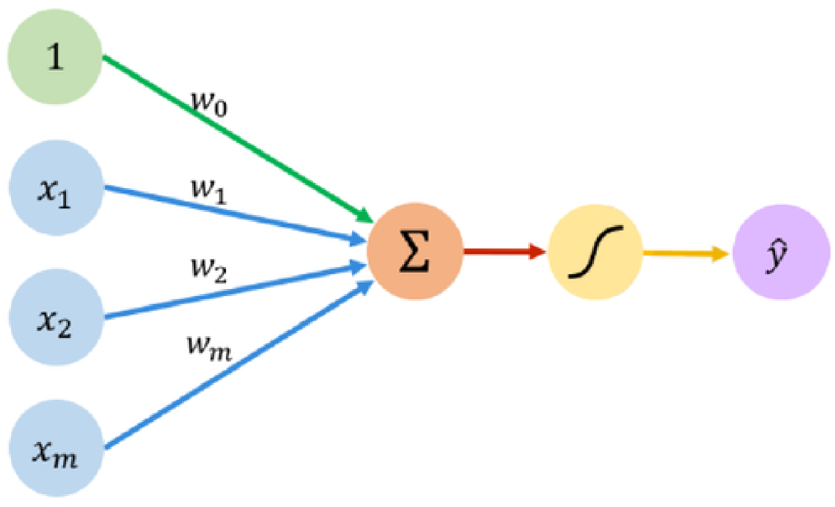
$$\hat{y} = g(\, w_0 + X^T W \,)$$

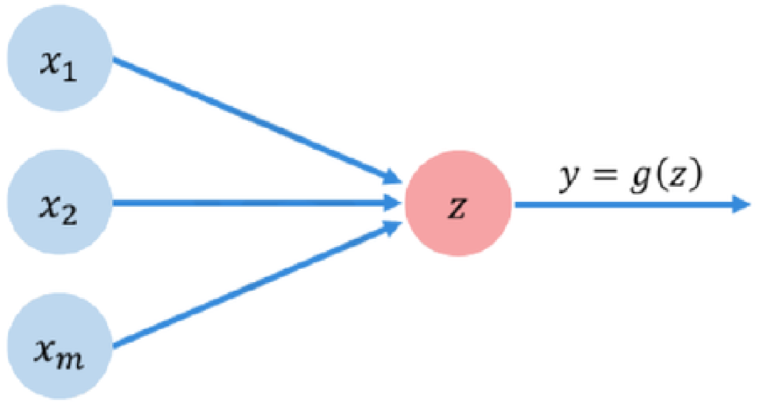
Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \ge 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \ne 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \ge 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \ge 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) [2]		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \ge 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \ge 0 \end{cases}$
Exponential Linear Unit (ELU) [3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \ge 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for } x < 0 \\ 1 & \text{for } x \ge 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1 + e^x)$	$f'(x) = \frac{1}{1+e^{-x}}$

Feed Forward (FF)

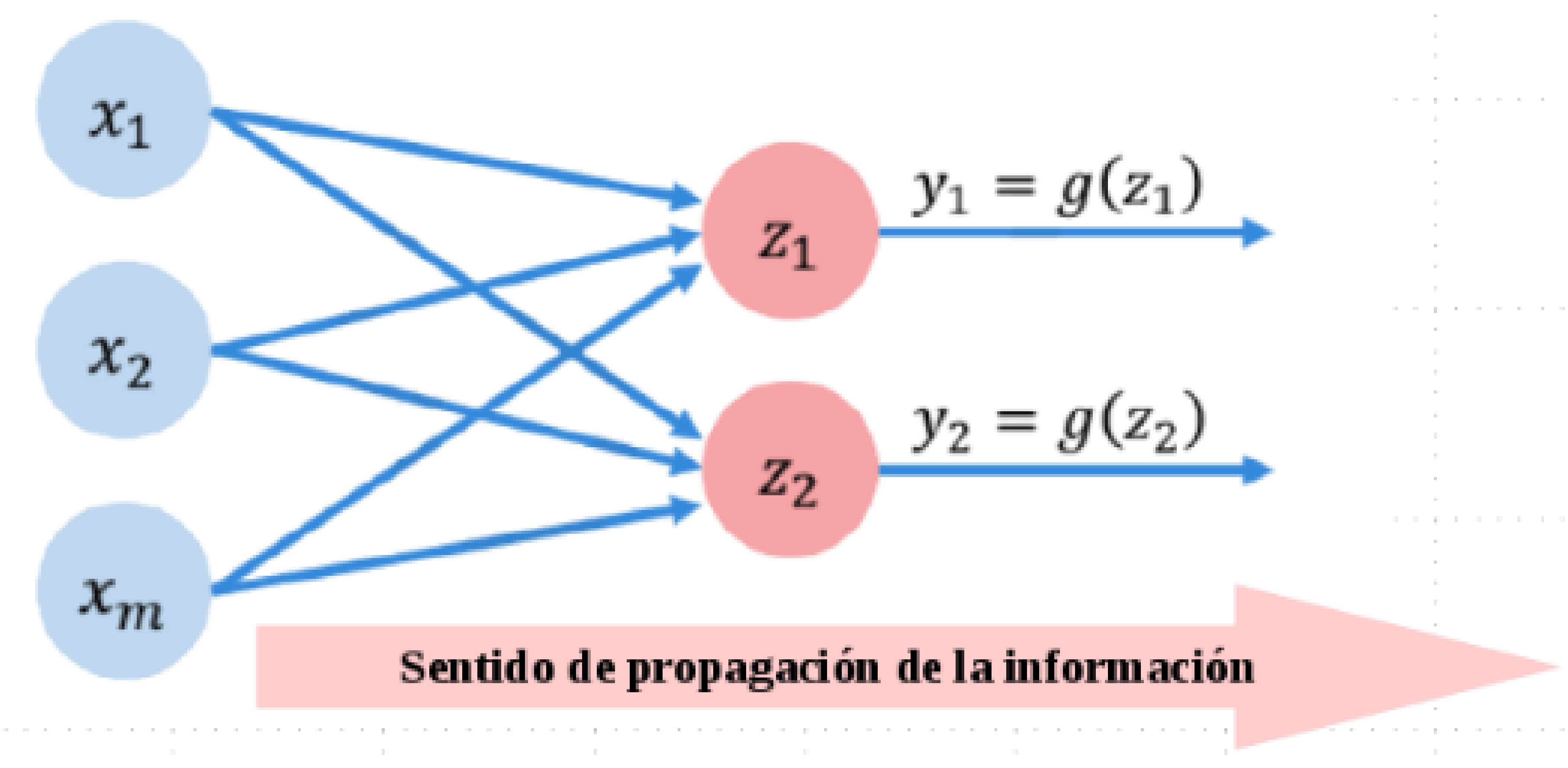
- Perceptrón simplificado (n entradas → 1 salida)

$$z = w_0 + \sum_{j=1}^{m} x_j\, w_j$$





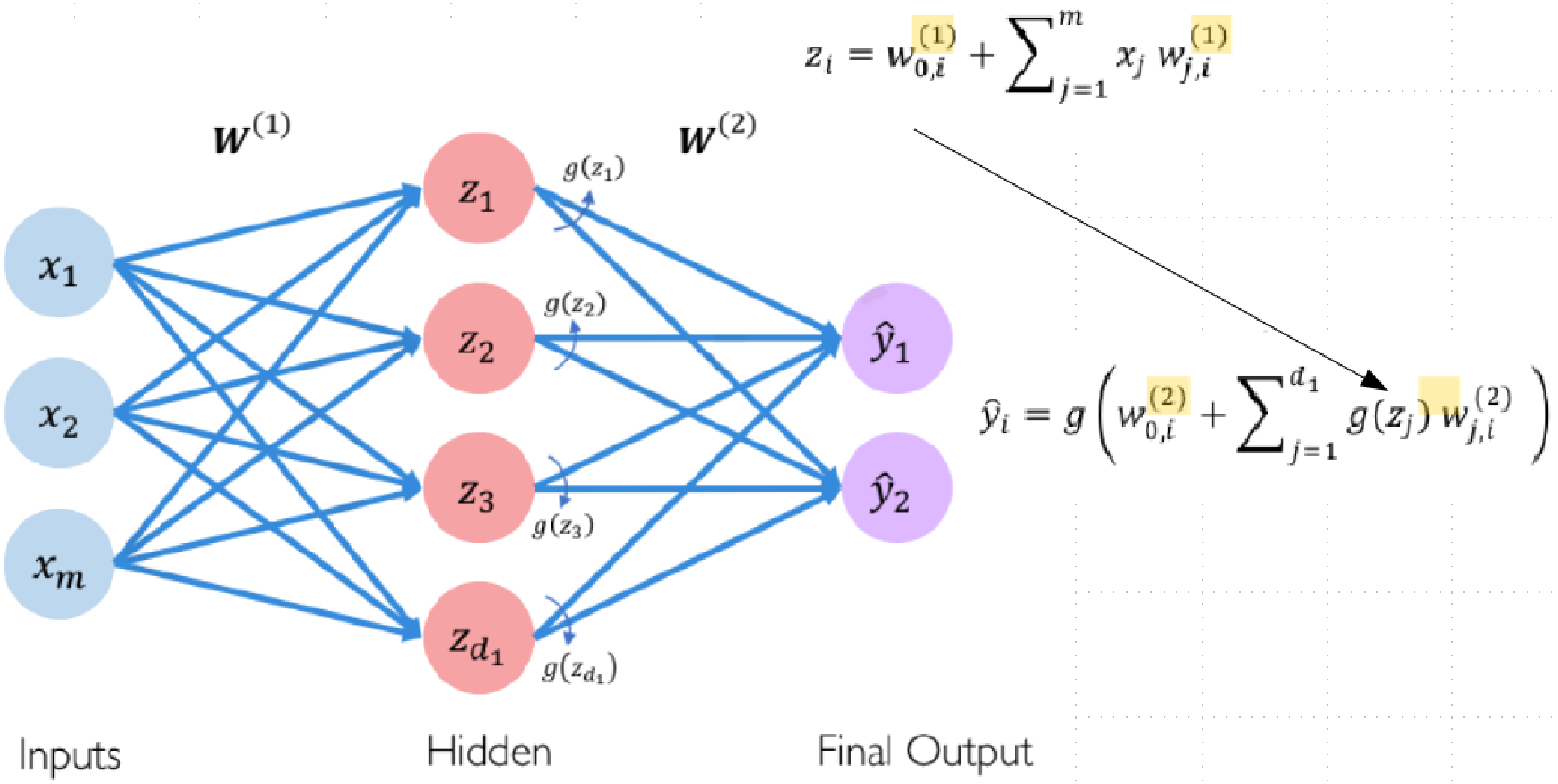
- Armemos una red neuronal compuesta por múltiples perceptrones (n entradas → m salida)



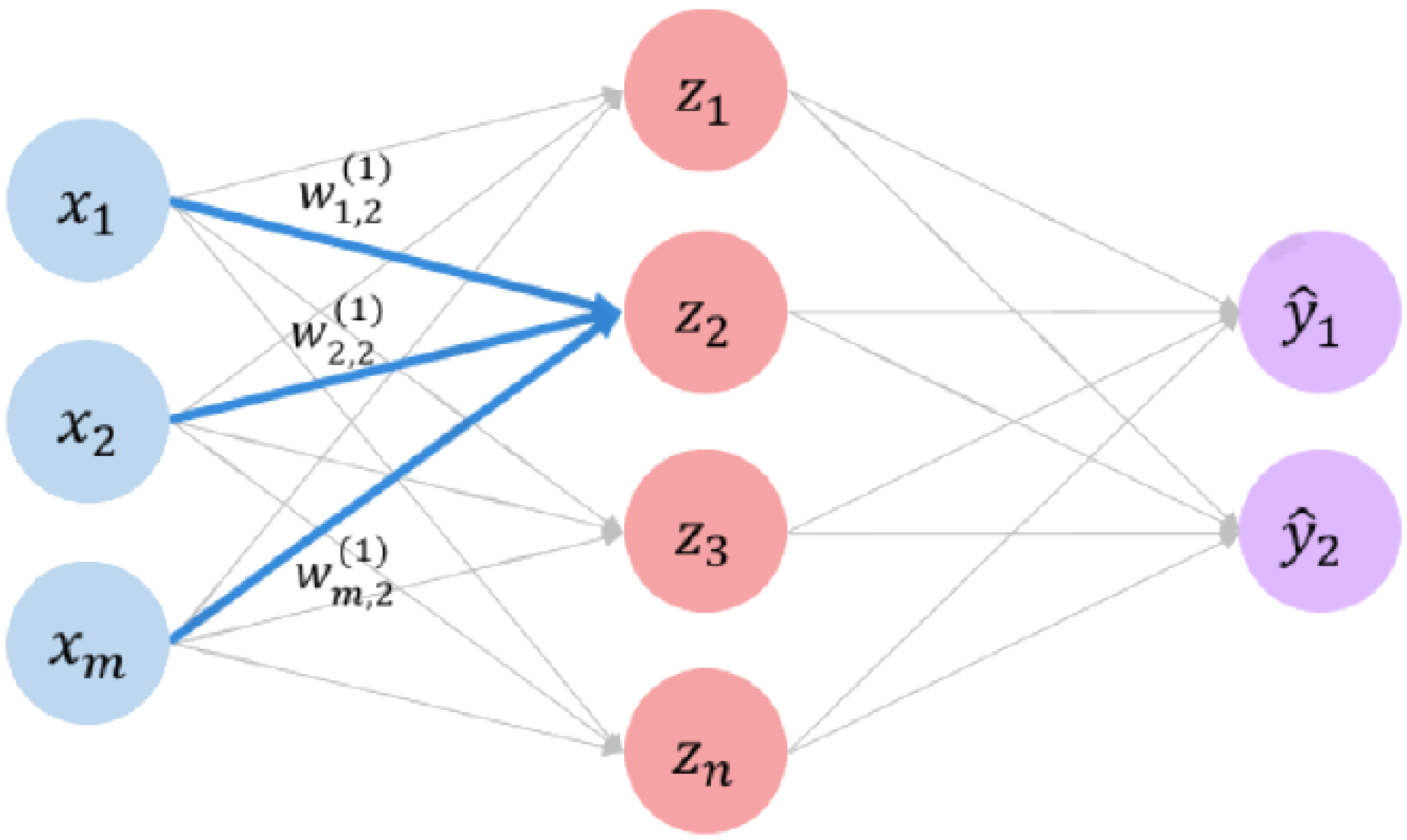
$$z_i = w_{0,i} + \sum_{j=1}^{m} x_j\, w_{j,i}$$

Observación: todas las entradas están conectadas a todas las salidas → dense layers

1-Layer Neural Network



1-Layer Neural Network

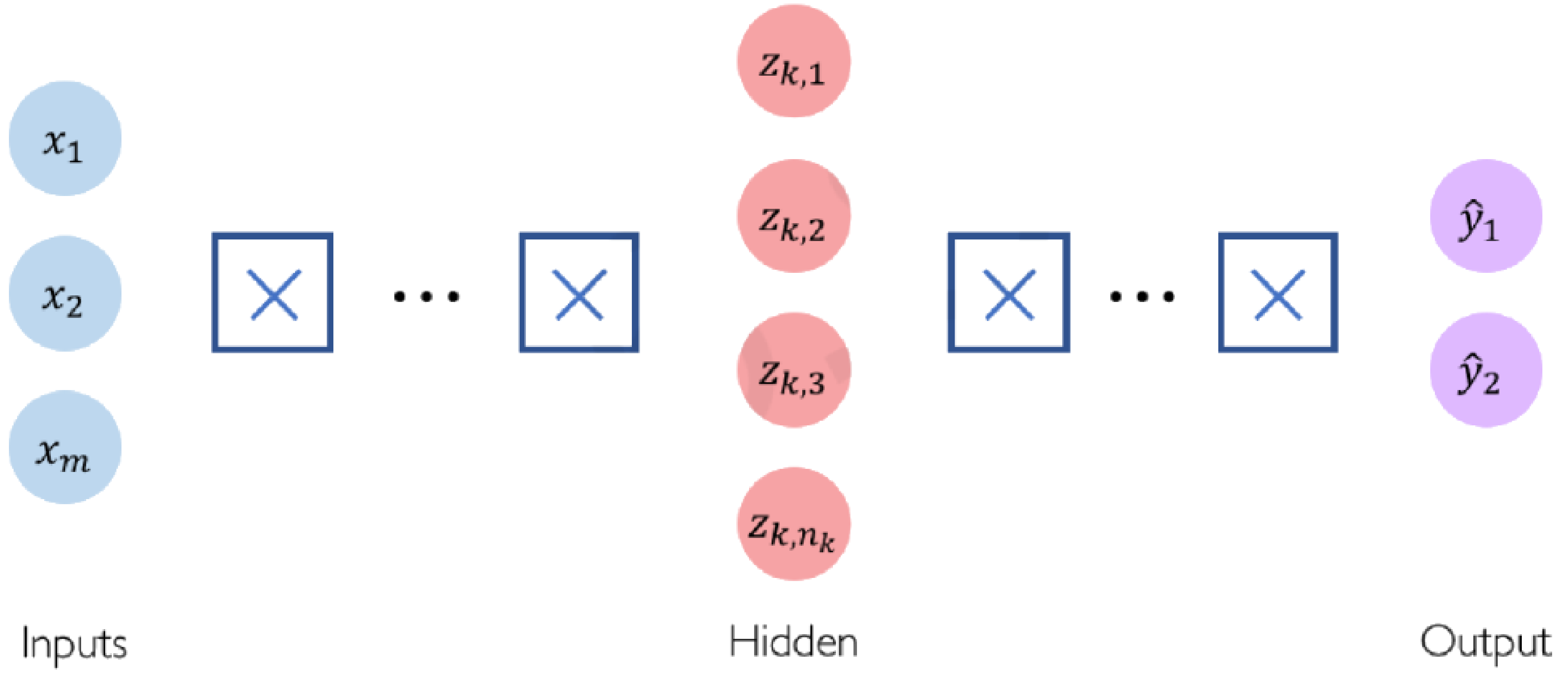


$$z_2 = w_{0,2}^{(1)} + \sum_{j=1}^{m} x_j\, w_{j,2}^{(1)}$$

$$= w_{0,2}^{(1)} + x_1\, w_{1,2}^{(1)} + x_2\, w_{2,2}^{(1)} + x_m\, w_{m,2}^{(1)}$$

M-layers Neural Network (multi-layer)

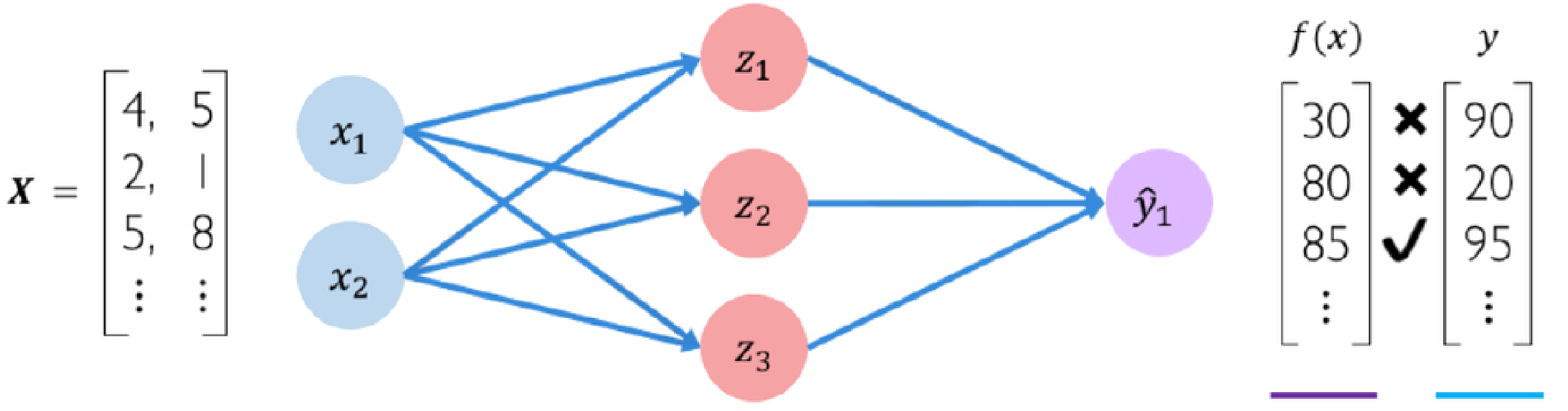
- Deep Neural Network



$$z_{k,i} = w_{0,i}^{(k)} + \sum_{j=1}^{n_{k-1}} g(z_{k-1,j})\, w_{j,i}^{(k)}$$

Loss Function

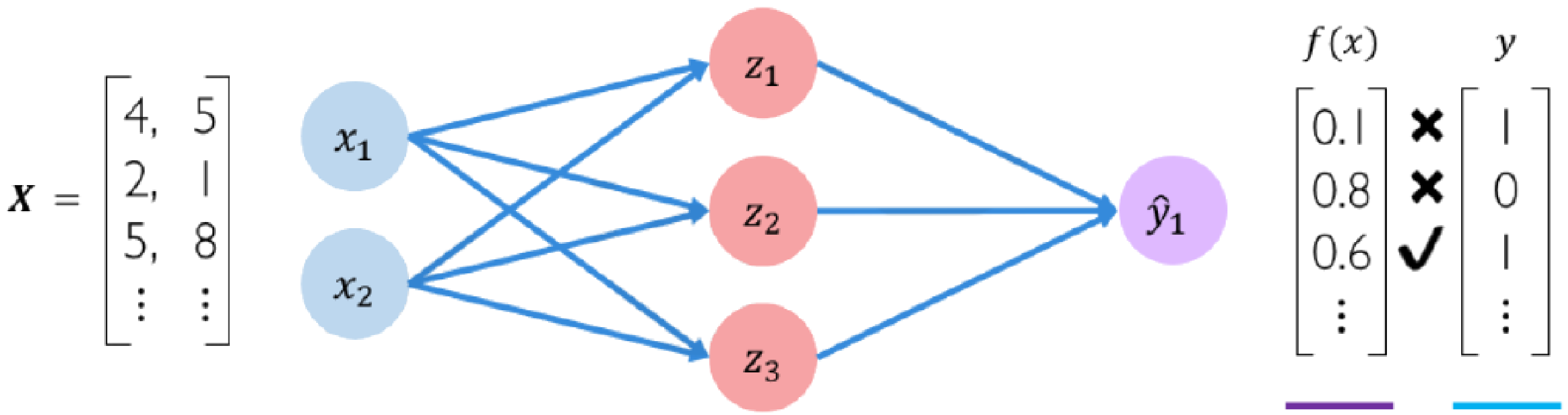
Mean squared error loss can be used with regression models that output continuous real numbers



Final Grades (percentage)

$$J(W) = \frac{1}{n}\sum_{i=1}^{n}\Big(\underbrace{y^{(i)}}_{\text{Actual}} - \underbrace{f\big(x^{(i)}; W\big)}_{\text{Predicted}}\Big)^2$$

Cross entropy loss can be used with models that output a probability between 0 and 1



$$J(W) = \frac{1}{n}\sum_{i=1}^{n}\underbrace{y^{(i)}}_{\text{Actual}}\log\Big(\underbrace{f\big(x^{(i)}; W\big)}_{\text{Predicted}}\Big) + (1 - \underbrace{y^{(i)}}_{\text{Actual}})\log\Big(1 - \underbrace{f\big(x^{(i)}; W\big)}_{\text{Predicted}}\Big)$$

symbol	name	equation
$\mathcal{L}_1$	L_1 loss	$\Vert\mathbf{y}-\mathbf{o}\Vert_1$
$\mathcal{L}_2$	L_2 loss	$\Vert\mathbf{y}-\mathbf{o}\Vert_2^2$
$\mathcal{L}_1 \circ \sigma$	expectation loss	$\Vert\mathbf{y}-\sigma(\mathbf{o})\Vert_1$
$\mathcal{L}_2 \circ \sigma$	regularised expectation loss[1]	$\Vert\mathbf{y}-\sigma(\mathbf{o})\Vert_2^2$
$\mathcal{L}_\infty \circ \sigma$	Chebyshev loss	$\max_j \vert\sigma(\mathbf{o})^{(j)} - \mathbf{y}^{(j)}\vert$
hinge	hinge [13] (margin) loss	$\sum_j \max(0, \frac{1}{2} - \hat{\mathbf{y}}^{(j)}\mathbf{o}^{(j)})$
hinge^2	squared hinge (margin) loss	$\sum_j \max(0, \frac{1}{2} - \hat{\mathbf{y}}^{(j)}\mathbf{o}^{(j)})^2$
hinge^3	cubed hinge (margin) loss	$\sum_j \max(0, \frac{1}{2} - \hat{\mathbf{y}}^{(j)}\mathbf{o}^{(j)})^3$
log	log (cross entropy) loss	$-\sum_j \mathbf{y}^{(j)}\log\sigma(\mathbf{o})^{(j)}$
log^2	squared log loss	$-\sum_j [\mathbf{y}^{(j)}\log\sigma(\mathbf{o})^{(j)}]^2$
tan	Tanimoto loss	$\frac{-\sum_j \sigma(\mathbf{o})^{(j)}\mathbf{y}^{(j)}}{\Vert\sigma(\mathbf{o})\Vert_2^2 + \Vert\mathbf{y}\Vert_2^2 - \sum_j \sigma(\mathbf{o})^{(j)}\mathbf{y}^{(j)}}$
D_{CS}	Cauchy-Schwarz Divergence [3]	$-\log\frac{\sum_j \sigma(\mathbf{o})^{(j)}\mathbf{y}^{(j)}}{\Vert\sigma(\mathbf{o})\Vert_2\Vert\mathbf{y}\Vert_2}$

- https://arxiv.org/pdf/1702.05659.pdfs

Training (loss optimization)

- Encontrar los valores de los **pesos** de la red neuronal tal que la **pérdida o costo sea mínimo**

$$\boldsymbol{W}^* = \underset{\boldsymbol{W}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\left(f\left(x^{(i)}; \boldsymbol{W}\right), y^{(i)}\right)$$

$$\boldsymbol{W}^* = \underset{\boldsymbol{W}}{\operatorname{argmin}}\, J(\boldsymbol{W})$$

- Es un problema de optimización
- Podemos usar algún método numérico de optimización

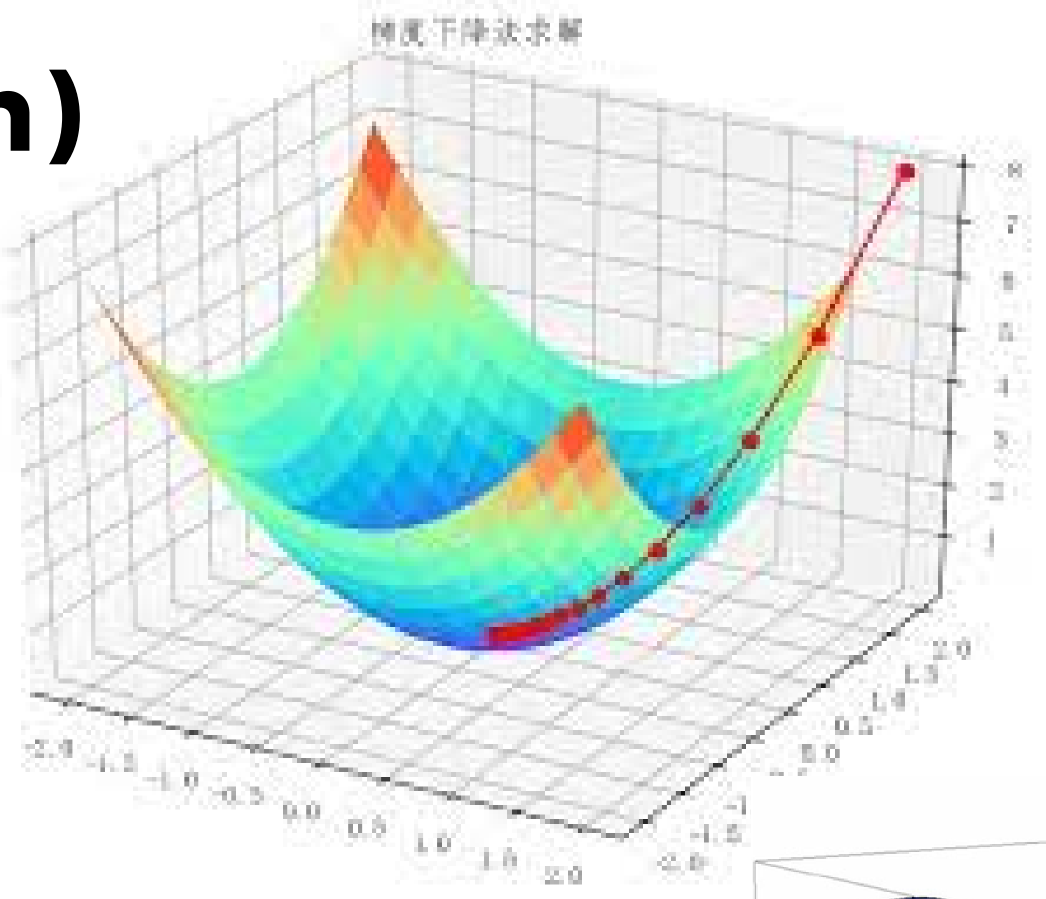
MÉTODO DEL GRADIENTE DESCENDIENTE

TSWC, 2022

Training (loss optimization)

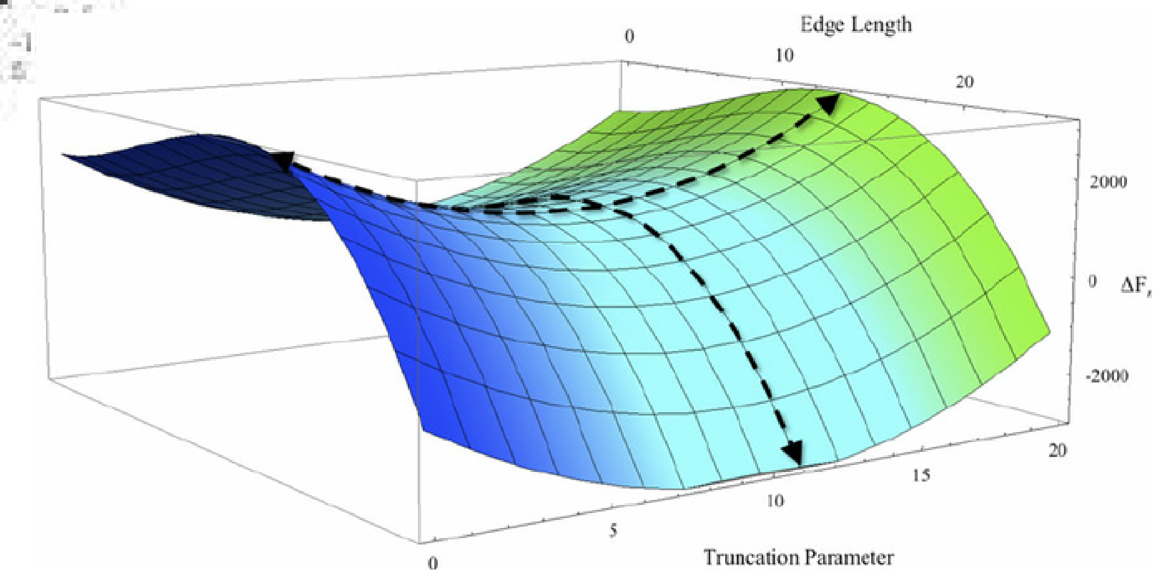
$$\boldsymbol{W}^* = \underset{\boldsymbol{W}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\left(f\left(x^{(i)}; \boldsymbol{W}\right), y^{(i)}\right)$$

$$\boldsymbol{W}^* = \underset{\boldsymbol{W}}{\operatorname{argmin}} J(\boldsymbol{W})$$



Algorithm

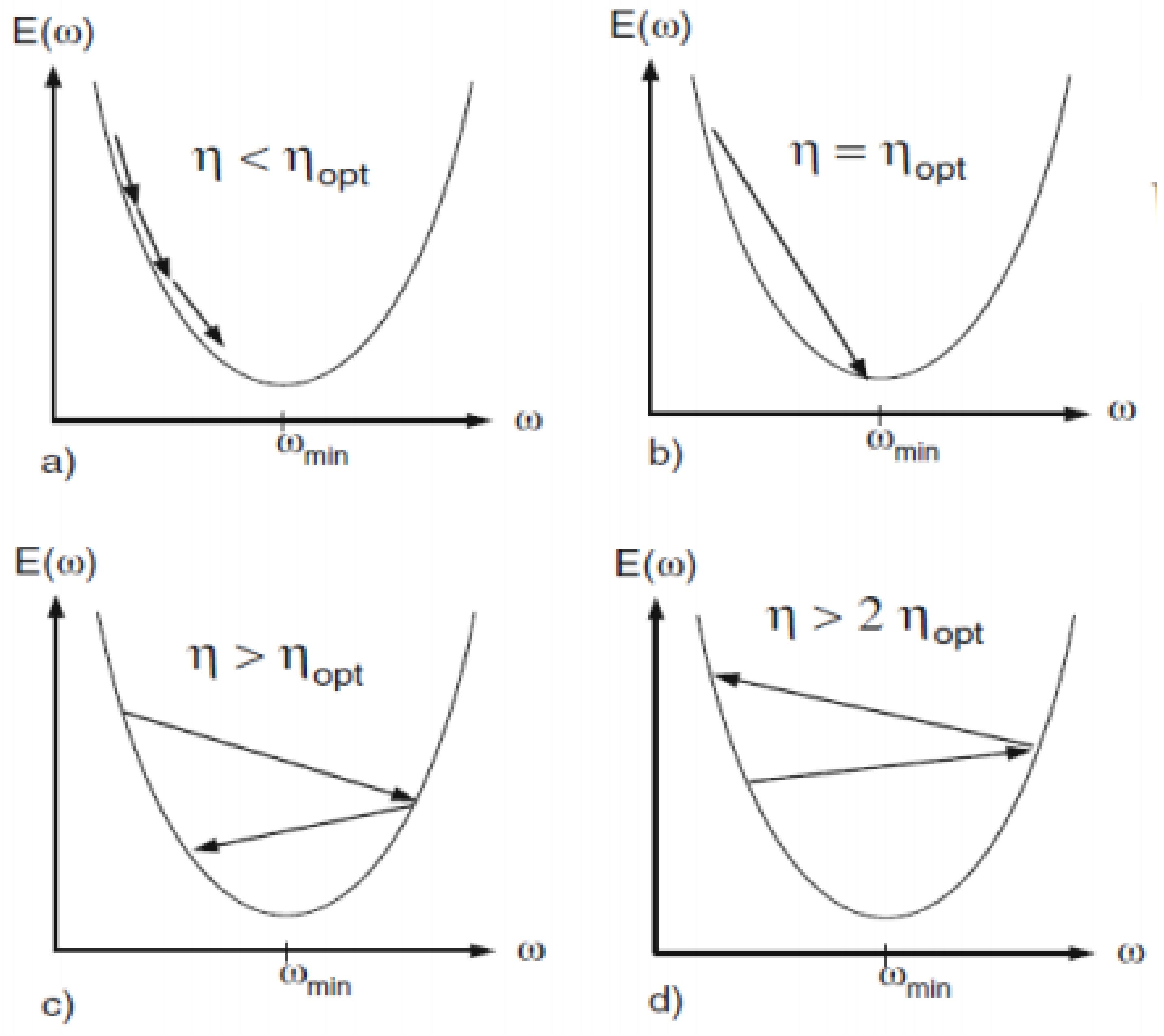
1. Initialize weights randomly $\sim\mathcal{N}(0, \sigma^2)$
2. Loop until convergence:
3. Compute gradient, $\frac{\partial J(\boldsymbol{W})}{\partial \boldsymbol{W}}$
4. Update weights, $\boldsymbol{W} \leftarrow \boldsymbol{W} - \eta \frac{\partial J(\boldsymbol{W})}{\partial \boldsymbol{W}}$
5. Return weights



Principales problemas:

- Mínimos locales
- Silla de montar: el g. d. un vez que llega a una región con gradiente cero, no puede escapar de allí independientemenete de la calidad del mínimo.

Training (loss optimization)



$$\boldsymbol{W} \leftarrow \boldsymbol{W} - \boxed{\eta}\,\frac{\partial J(\boldsymbol{W})}{\partial \boldsymbol{W}}$$

How can we set the learning rate?

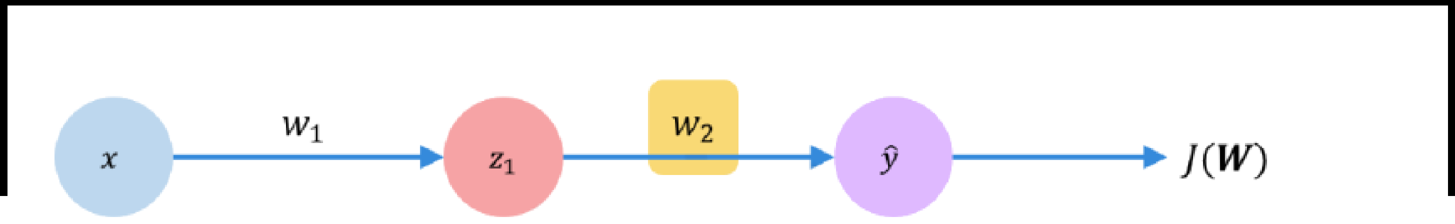
- https://ruder.io/optimizing-gradient-descent/index.html

Gradient Descent Algorithms

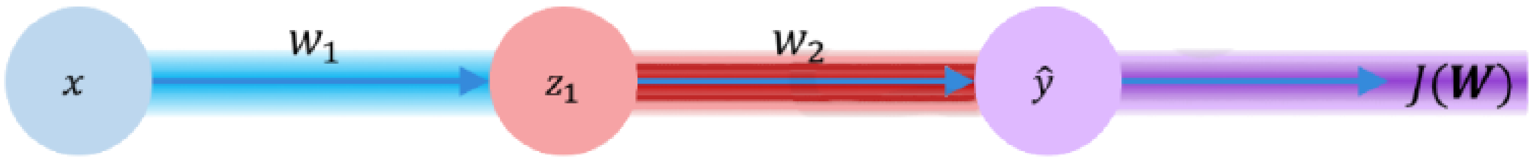
Algorithm	TF Implementation	Reference
• SGD	`tf.keras.optimizers.SGD`	Kiefer & Wolfowitz. "Stochastic Estimation of the Maximum of a Regression Function." 1952.
• Adam	`tf.keras.optimizers.Adam`	Kingma et al. "Adam: A Method for Stochastic Optimization." 2014.
• Adadelta	`tf.keras.optimizers.Adadelta`	Zeiler et al. "ADADELTA: An Adaptive Learning Rate Method." 2012.
• Adagrad	`tf.keras.optimizers.Adagrad`	Duchi et al. "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization." 2011.
• RMSProp	`tf.keras.optimizers.RMSProp`	

Backpropagation

- Qué tanto puede afectar a la función de pérdida un pequeño cambio en uno del los pesos?



- Aplicando la regla de la cadena



$$\frac{\partial J(\boldsymbol{W})}{\partial w_1} = \frac{\partial J(\boldsymbol{W})}{\partial \hat{y}} * \frac{\partial \hat{y}}{\partial z_1} * \frac{\partial z_1}{\partial w_1}$$

- Repetir para cada uno de los pesos en la red usando los gradientes de las capas siguiente!

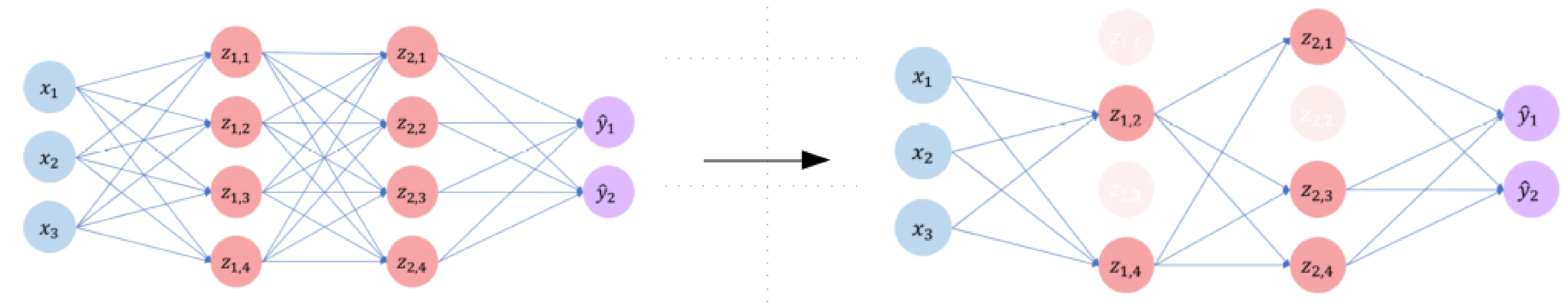
- Info se propaga desde las entradas hacia adelante mediante sus parámetros hasta que logra hacer una predicción

- Luego realiza una propagación hacia atrás a lo largo de la red para ir modificando los parámetros de manera que el error final sea el mínimo.

- El error asociado a una mala predicción es usado para ajustar los parámetros (el aprendizaje puede ser visto como una optimización). Para esto la salida y es comparada con el valor esperado (target) del conjunto de entrenamiento (z). La diferencia se entre el valor esperado y el de salida se llama error o residuo.

Regularización

- Técnica que consiste en establecer restricciones al problema de optimización para evitar llegar a modelos complejos
- La idea es poder generalizar nuestro modelo para nuevos datos

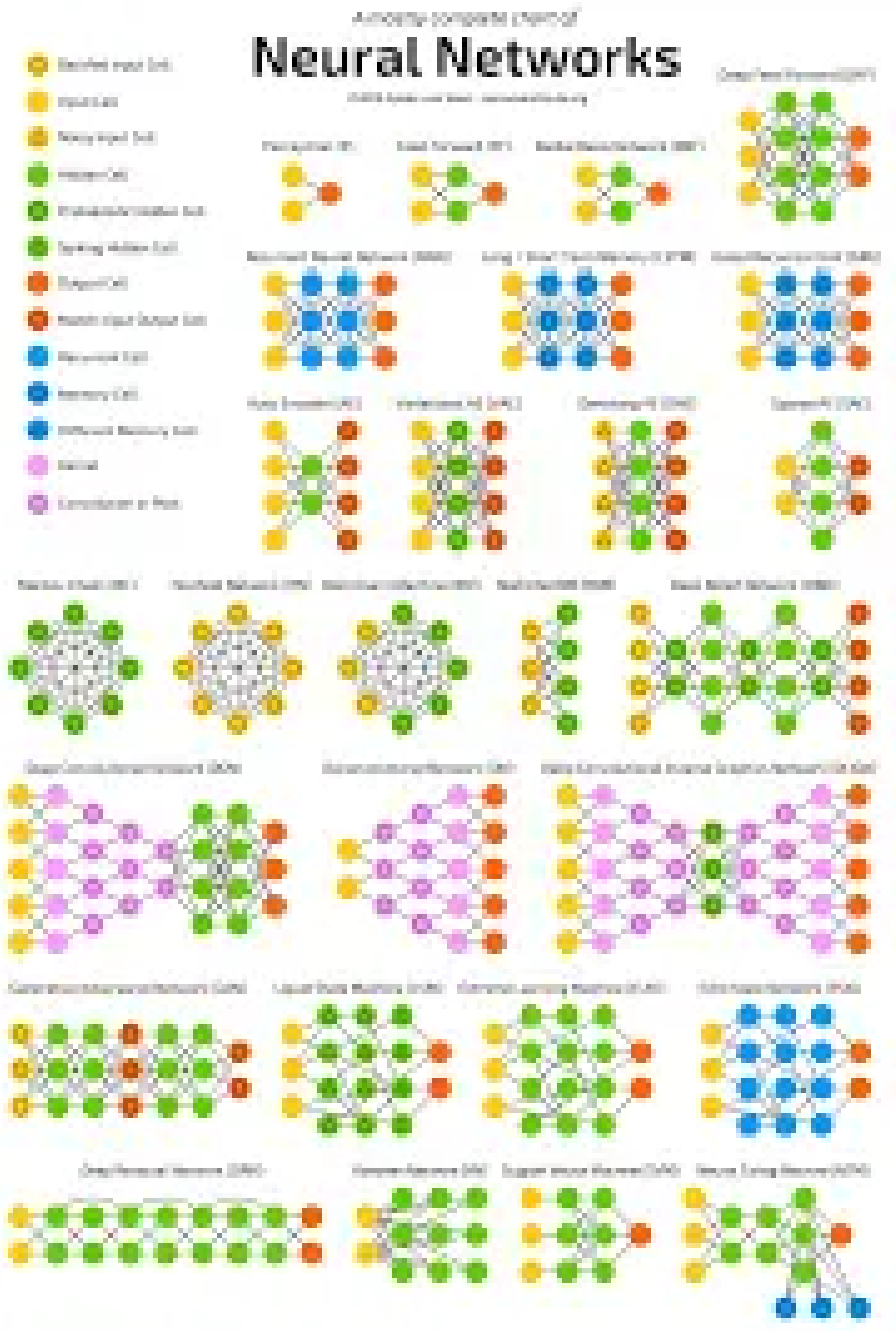


DROP OUT

- Durante el entrenamiento de manera aleatoria se setean algunas activaciones en 0.
- << overfitting

EARLY STOPING

- Consiste en parar antes de llegar al overfitting observando los datos de training y testing.


Neural Networks


ZOO

HANDS-ON LAB

TP - 1

2 problems

- Radar signal classification (mandatory)
- Flare classification (optional)

From science to data science

PROBLEM 1 -

Classify radar returns from the ionosphere as either suitable for further analysis or not ("good" or "bad").
The 17 pairs of numbers, representing 17 discrete values of the real part and the corresponding 17 values of the complex part of an ACF, are the input to the neural network.

VINCENT G. SIGILLITO, SIMON P. WING, LARRIE V. HUTTON, and KILE B. BAKER

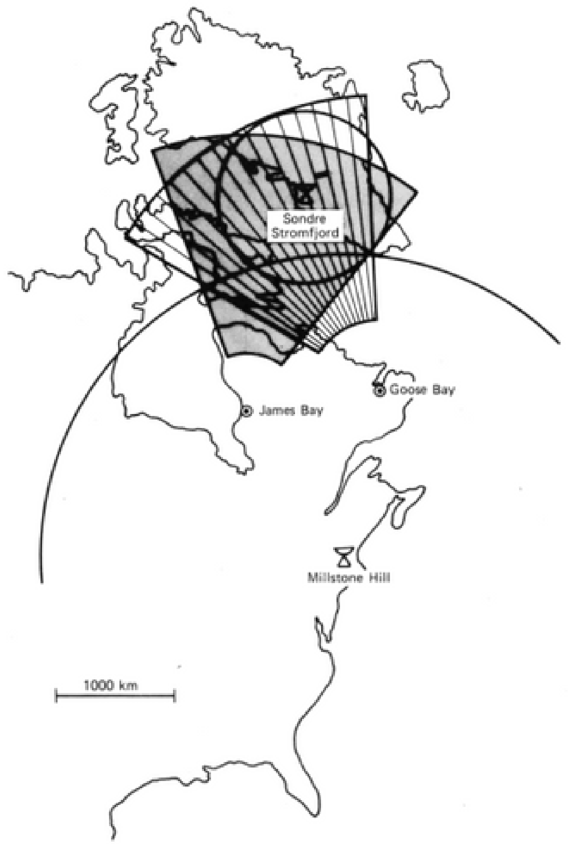
CLASSIFICATION OF RADAR RETURNS FROM THE IONOSPHERE USING NEURAL NETWORKS

In ionospheric research, we must classify radar returns from the ionosphere as either suitable for further analysis or not. This time-consuming task has typically required human intervention. We tested several different feedforward neural networks to investigate the effects of network type (single-layer versus multilayer) and number of hidden nodes upon performance. As expected, the multilayer feedforward networks (MLFN's) outperformed the single-layer networks, achieving 100% accuracy on the training set and up to 98% accuracy on the testing set. Comparable figures for the single-layer networks were 94.5% and 92%, respectively. When measures of sensitivity, specificity, and proportion of variance accounted for by the model are considered, the superiority of the MLFN's over the single-layer networks is even more striking.

Paper source:
Sigillito, V. G., Wing, S. P., Hutton, L. V., \& Baker, K. B. (1989). Classification of radar returns from the ionosphere using neural networks. Johns Hopkins APL Technical Digest, 10, 262-266.

Data source:
Space Physics Group of the Johns Hopkins University Applied Physics Laboratory.

From science to data science



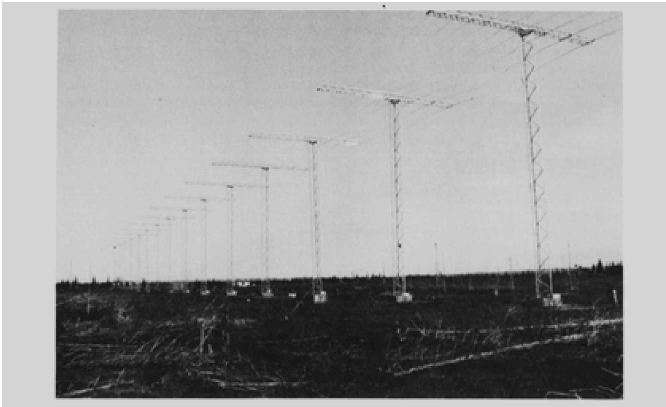
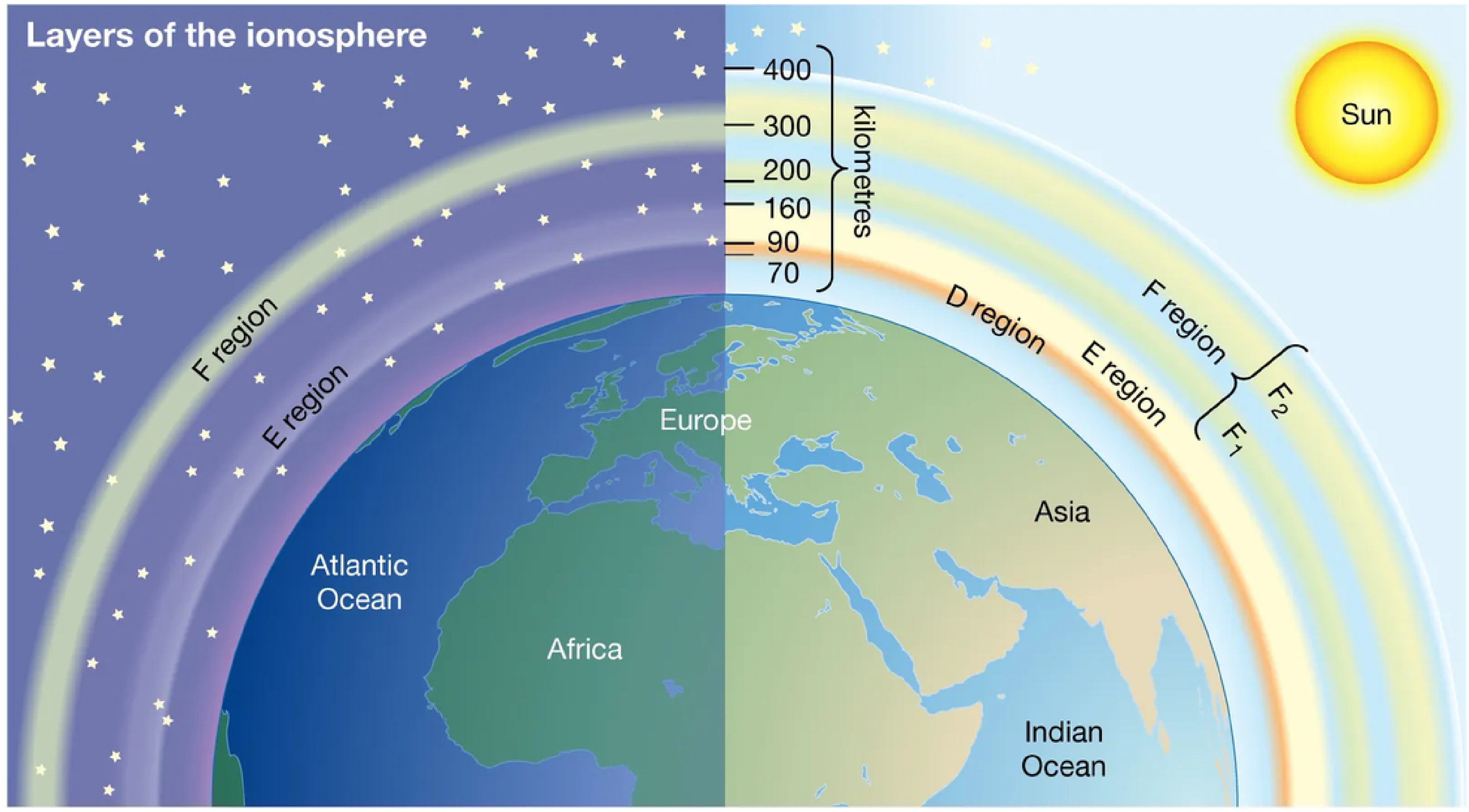
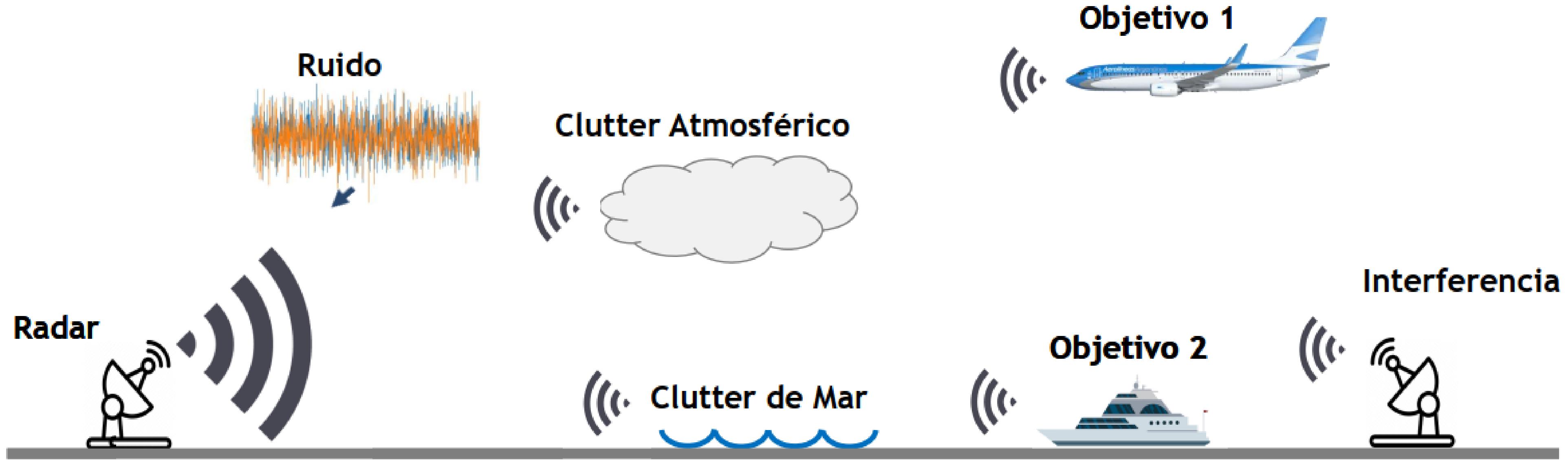


Fig. 7. View of antenna array from front, northwest corner. Shed for phasing matrix is located at the center of the array.



Dr. Z. Saavedra – FACET, UNT



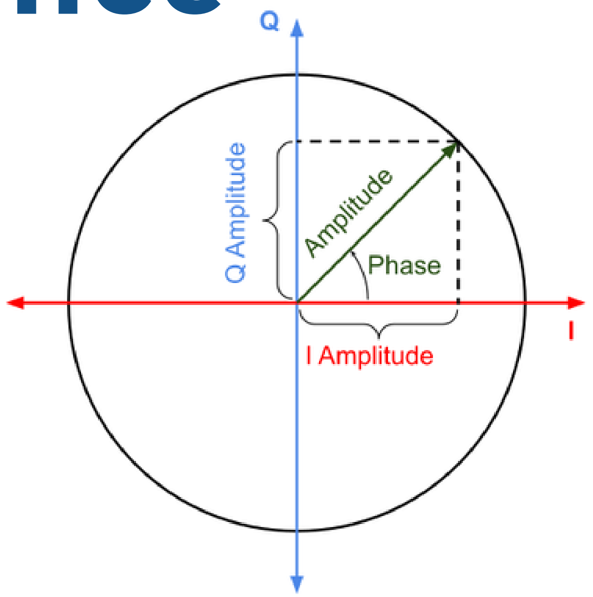
$$Eco(t) = Atte \cdot RCS \cdot u(t) \cdot m(t) \cdot e^{j\omega_c t \pm j\omega_D t + \varphi}$$

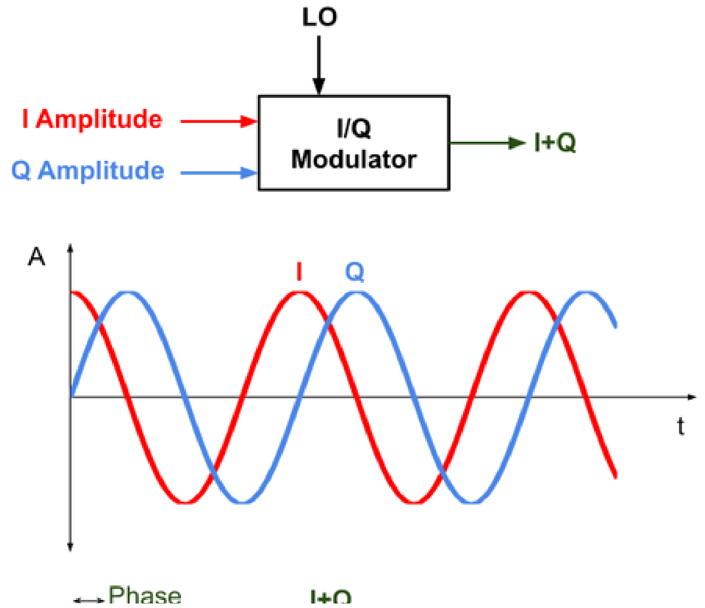
$$\boxed{S_R(t) = Eco(t) + Clutter\ (t) + Ruido(t) + Interferencia(t)}$$

*: "target" = ionosphere

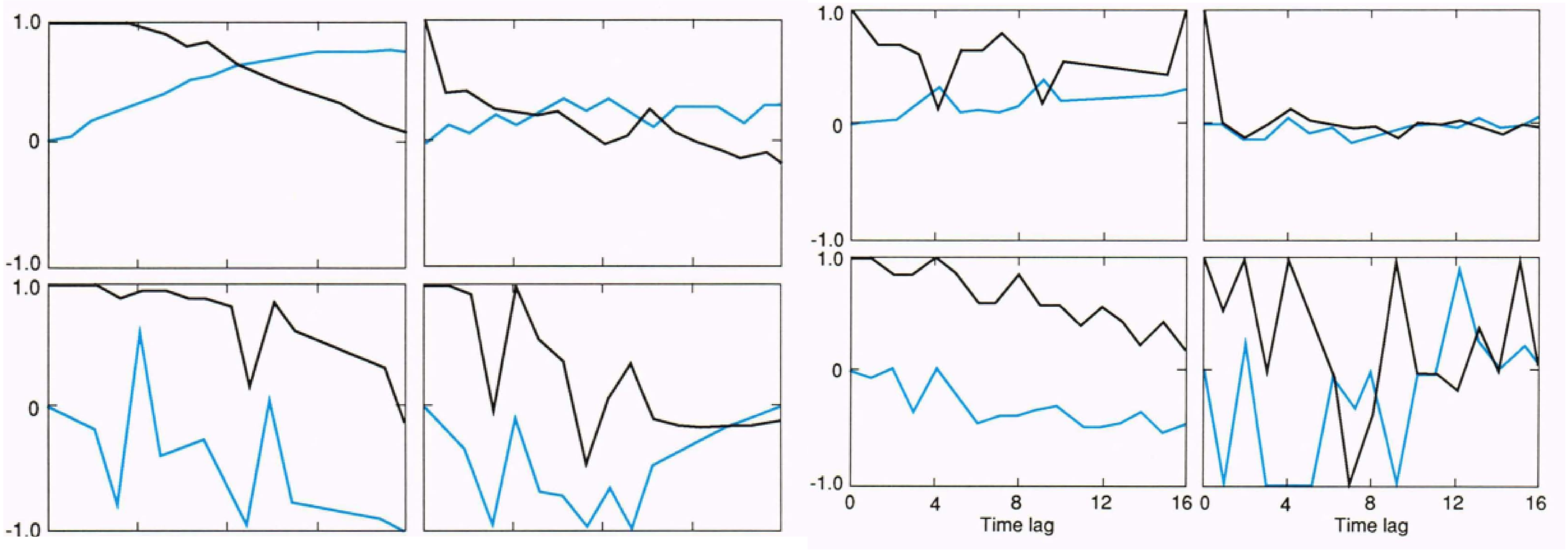
From science to data science

IQ Modulation: it is a specific Phase-Modulation type, that provides us certain information in the real component of the signal and another posible coded information in the imaginary part. We can obtain both of them using demodulation techniques such as quadrature hybrids.





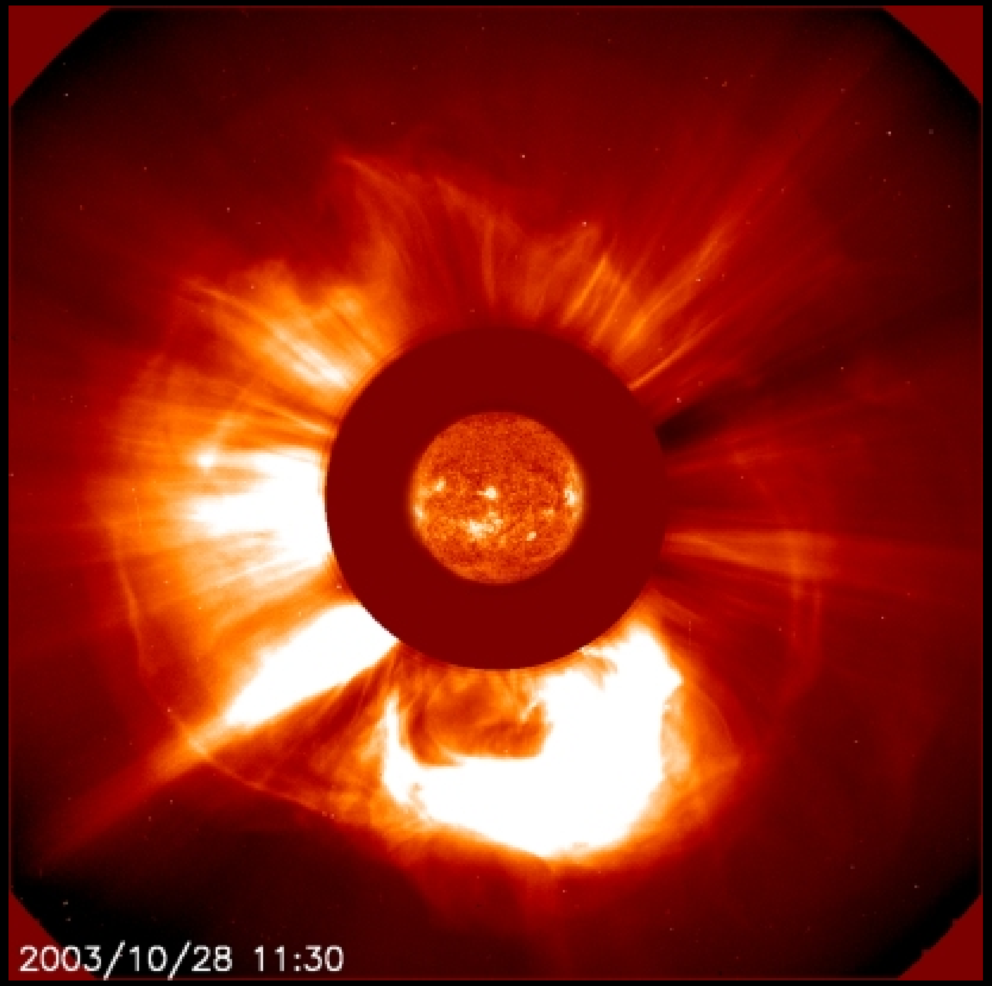
$$C(t) = A(t) + iB(t)$$



Good echo *Bad echo* *Good echo* *Bad echo*

Auto-Correlation Function Output

From science to data science



28 Oct 2003 - X 17.2 flare -> SPE + CME (geoeffective, 29 Oct)

29 Oct 2003 - X 10.0 flare (same AR) -> CME

The problem (optional)

- Automatic **classification of CMEs** according to their origin (flare/no-flare)
- From images (C2,C3 LASCO - SOHO) + expert observations
- source: https://cdaw.gsfc.nasa.gov/CME_list/index.html
- Aknowlwdgements: David Salazar de la Escuela de Física de la Universidad de Costa Rica + Salas-Matamoros & Klein, 2016.

From science to data science

TSWC, 2022

CME						Origen del CME								
						FLARE				Filamento				
Date			LASCO onset		Vprop	Coordinates		SXR onset		Observation date (Ha)			Coordinates	
DD	MM	YY	HH	MM		lat	long	HH	MM	DD	MM	YY	lat	long
7	4	97	14	27	592	-28	-11	14	8					
12	5	97	6	30	355	21	8	4	55					
21	6	98	5	35	619	18	39	5	18					
13	4	99	3	30	309	16	0	2	14					
6	6	0	15	54	1557	21	-10	15	26					
20	6	0	9	10	516	-27	38	8	27					
25	7	0	3	30	712	6	8	2	50					
21	7	6	13	54	403					20	7	6	-20	-25
29	12	6	21	30	236					29	12	6	-20	35
23	5	10	18	30	258					21	5	10	15	-20
30	1	11	19	36	521*					30	1	11	25	25
11	1	12	21	36	302					11	1	12	45	-25
16	3	12	20	36	862*					16	3	12	20	60
1	4	12	3	12	410*					31	3	12	20	-20
27	12	12	21	28	267					27	12	12	5	-20

.
.
.

Data Transformation

Columnas 1-3: Fecha de observación del CME
Columnas 4-5: Momento de la primera observación en C2 de LASCO reportado en el catálogo https://cdaw.gsfc.nasa.gov/CME_list/
Columna 6: Velocidad de propagación del CME calculado mediante la fórmula empírica en Salas-Matamoros & Klein, 2016.
En eventos asociados a erupción de filamento, la velocidad de propagación del CME es la reportada en el catálogo https://cdaw.gsfc.nasa.gov/CME_list/, donde dicha velocidad es calculada por un 'first-order polynomial fit' a partir de mediciones de altura vrs tiempo.
Columnas 7-8: Localización de la región activa que dio origen al Flare.
Columnas 9-10: Momento de inicio del busrt de SXR observado en los datos de GOES.
Columnas 11-13: Fechas de observación del filamento en imágenes en Ha (en www.solarmonitor.org).
Columnas 14-15: La posición central de cada filamento.

From science to data science

TSWC, 2022

- Features need selection & transformation

CME						Origen del CME								
						FLARE				Filamento				
Date			LASCO onset		Vprop	Coordinates		SXR onset		Observation date (Ha)			Coordinates	
DD	MM	YY	HH	MM		lat	long	HH	MM	DD	MM	YY	lat	long
7	4	97	14	27	592	-28	-11	14	8					
12	5	97	6	30	355	21	8	4	55					
21	6	98	5	35	619	18	39	5	18					
13	4	99	3	30	309	16	0	2	14					
6	6	0	15	54	1557	21	-10	15	26					
20	6	0	9	10	516	-27	38	8	27					
25	7	0	3	30	712	6	8	2	50					
21	7	6	13	54	403					20	7	6	-20	-25
29	12	6	21	30	236					29	12	6	-20	35
23	5	10	18	30	258					21	5	10	15	-20
30	1	11	19	36	521*					30	1	11	25	25
11	1	12	21	36	302					11	1	12	45	-25
16	3	12	20	36	862*					16	3	12	20	60
1	4	12	3	12	410*					31	3	12	20	-20
27	12	12	21	28	267					27	12	12	5	-20

.
.
.

- Target needs transformation

Automatic **classification of CMEs** (supervised)

From science to data science

TSWC, 2022

Data Transformation

CME						Origen del CME								
						FLARE				Filamento				
Date			LASCO onset		Vprop	Coordinates		SXR onset		Observation date (Ha)			Coordinates	
DD	MM	YY	HH	MM		lat	long	HH	MM	DD	MM	YY	lat	long
7	4	97	14	27	592	-28	-11	14	8					
12	5	97	6	30	355	21	8	4	55					
21	6	98	5	35	619	18	39	5	18					
13	4	99	3	30	309	16	0	2	14					
6	6	0	15	54	1557	21	-10	15	26					
20	6	0	9	10	516	-27	38	8	27					
25	7	0	3	30	712	6	8	2	50					
21	7	6	13	54	403					20	7	6	-20	-25
29	12	6	21	30	236					29	12	6	-20	35
23	5	10	18	30	258					21	5	10	15	-20
30	1	11	19	36	521*					30	1	11	25	25
11	1	12	21	36	302					11	1	12	45	-25
16	3	12	20	36	862*					16	3	12	20	60
1	4	12	3	12	410*					31	3	12	20	-20
27	12	12	21	28	267					27	12	12	5	-20

.
.
.

categorical

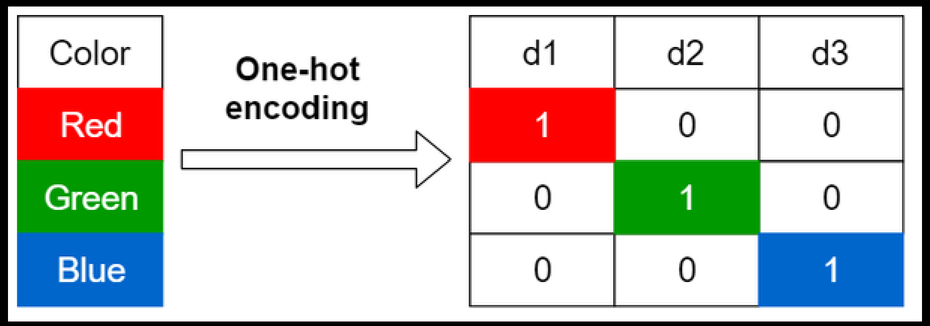
numeric data

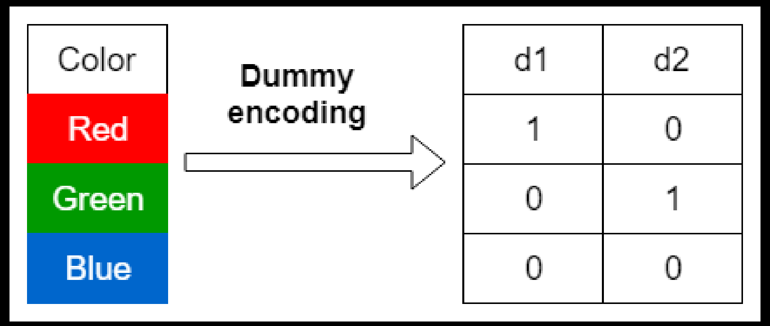
Categorical features (generally):

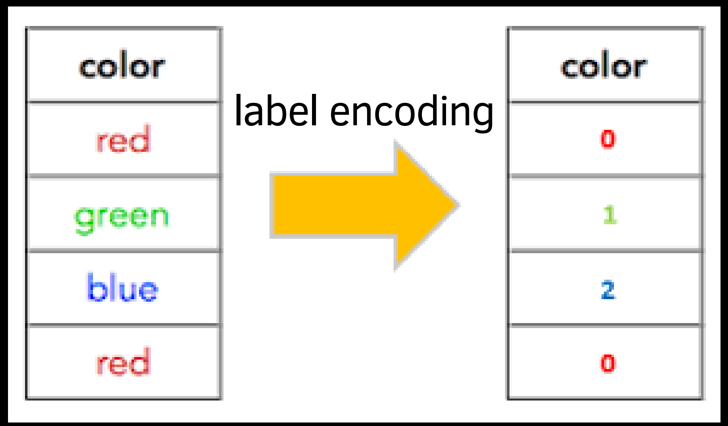
- Binary: Yes, No ; True, False
- B. Ordinal: Specific ordered Groups: low, medium, high; cold, hot, lava Hot
- Nominal: Unordered Groups: cat, dog, tiger -pizza, burger, coke

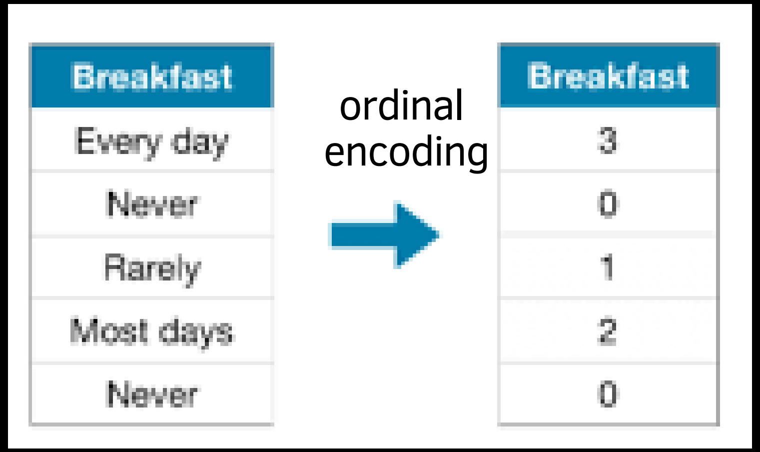
↓

ENCODING









From science to data science

- The training/test set

	A	B	C
1	SunActivity	vprop	Origin
2	low	592	flare
3	low	355	flare
4	low	619	flare
5	high	309	flare
6	high	1557	flare
7	high	516	flare
8	high	712	flare

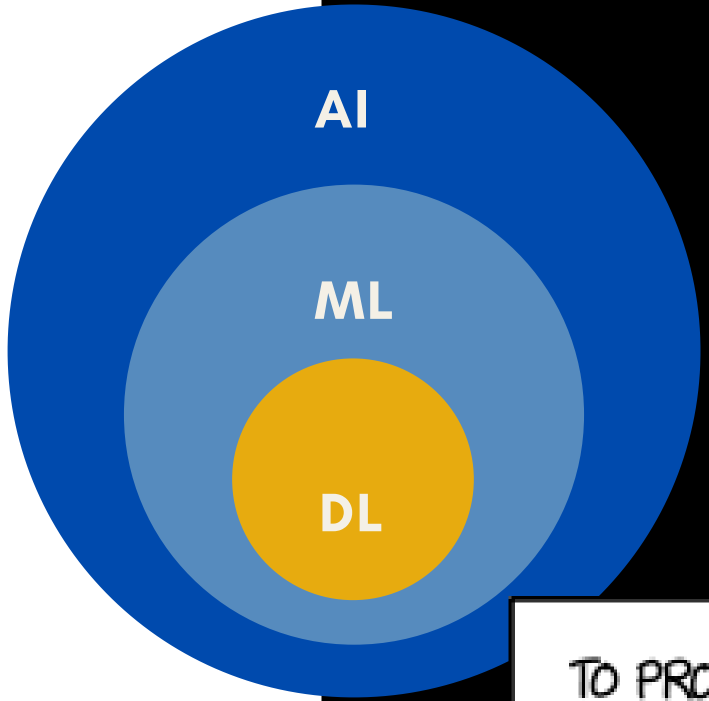
To Do:

- data exploration (outliers? nulls?, # samples, balanced? etc)
- data modeling preparation: standarization/normalization, one-hot encoding (or others), etc
- data splitting (70/30?, 80/20?, etc)
- MLP architecture! hyperparameters: #layers, Optimization algorithm, etc
- Fitting the model
- Obtain metrics
- Tunning

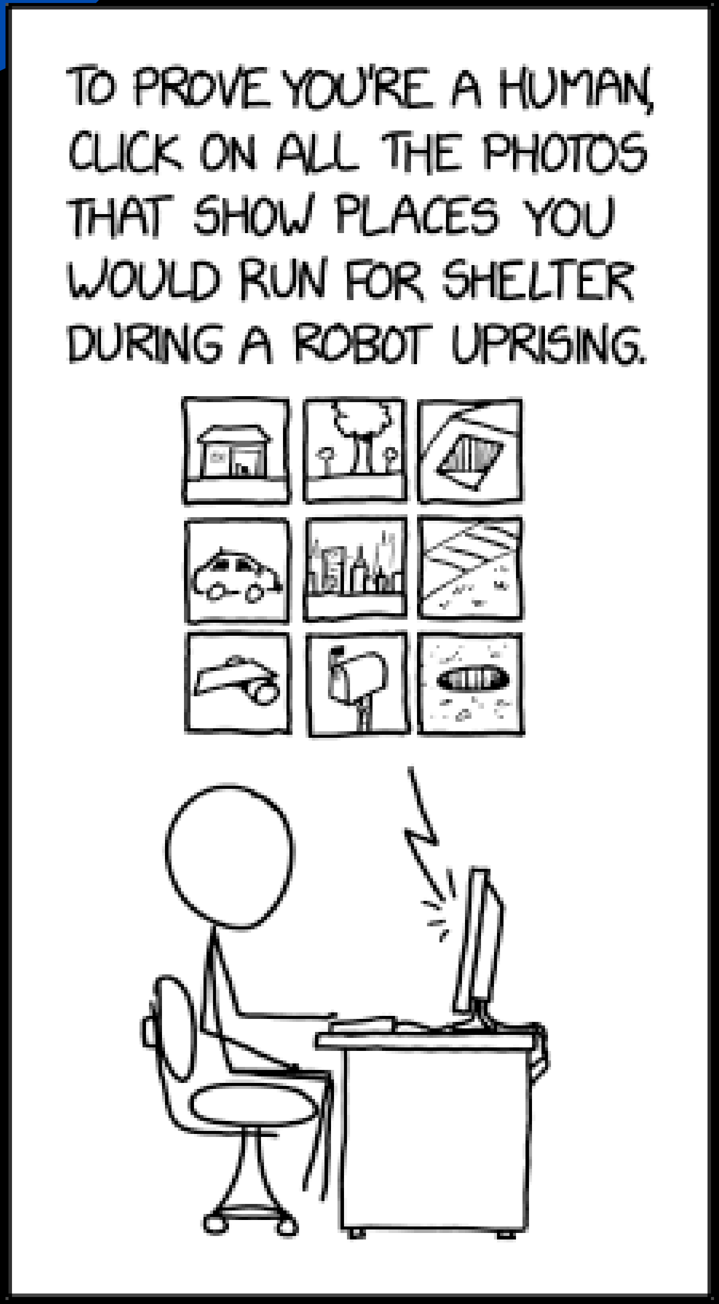
Questions

- Which is the scores for the proposed model?
- How confident are we about the model performance/validation?
- Is the dataset representative enough
- propose other questions?!

Deep Learning



Extract patterns from data using neural networks



TSWC, 2022

- https://this-person-does-not-exist.com/en

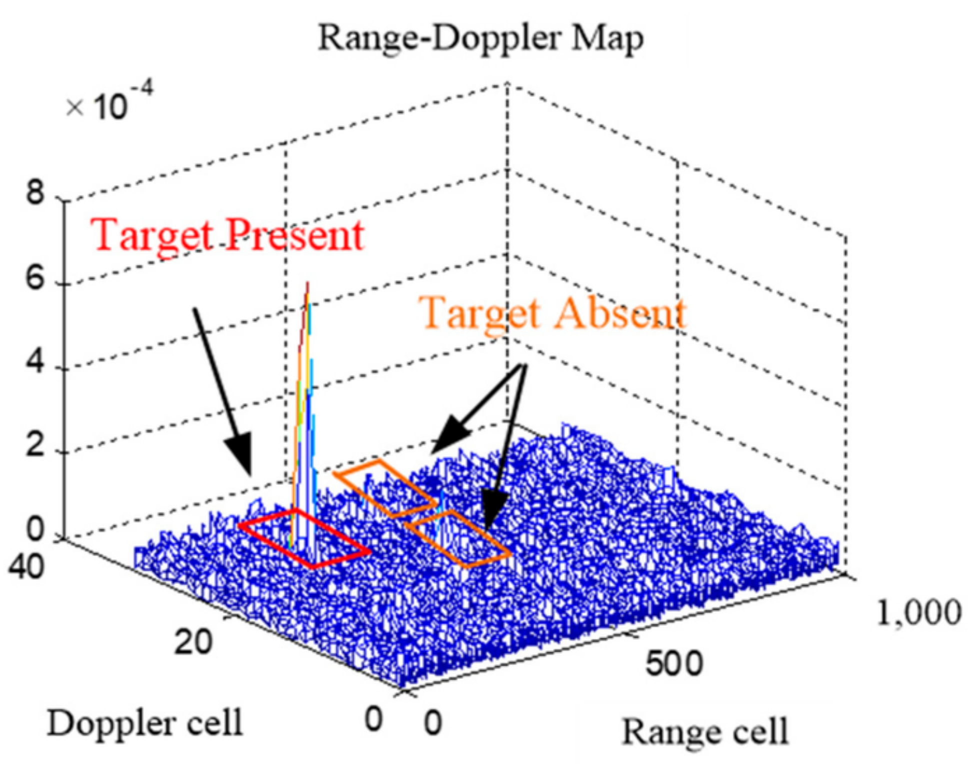
TSWC, 2022

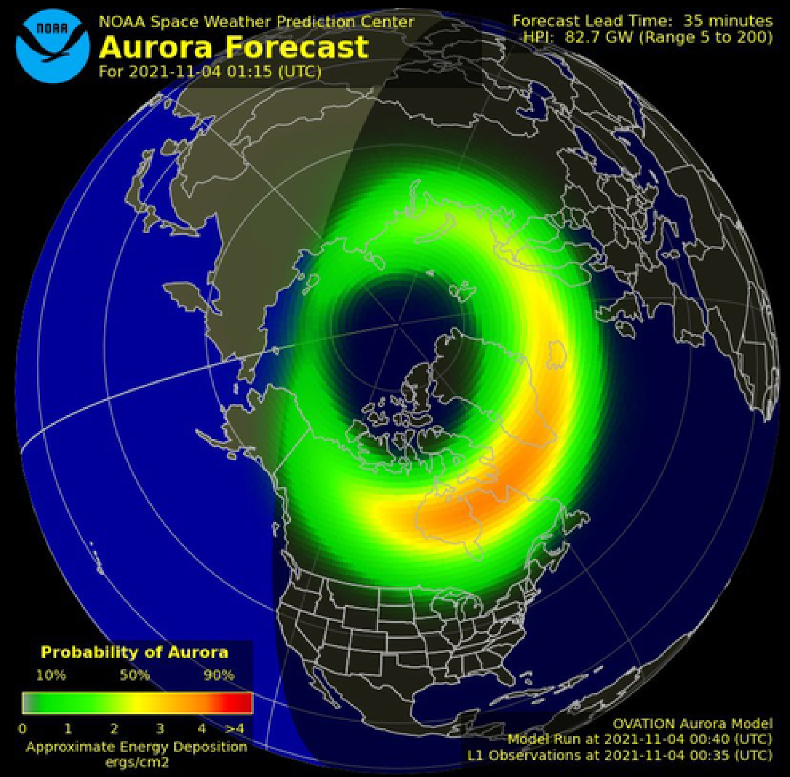
Recurrent Neural Networks

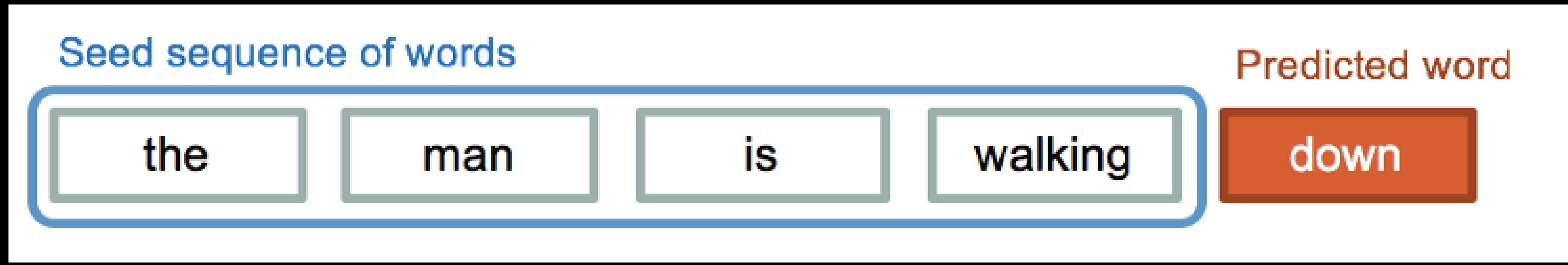
- Processing a sequence of data
  $x(t) = x(1), \dots, x(\tau)$

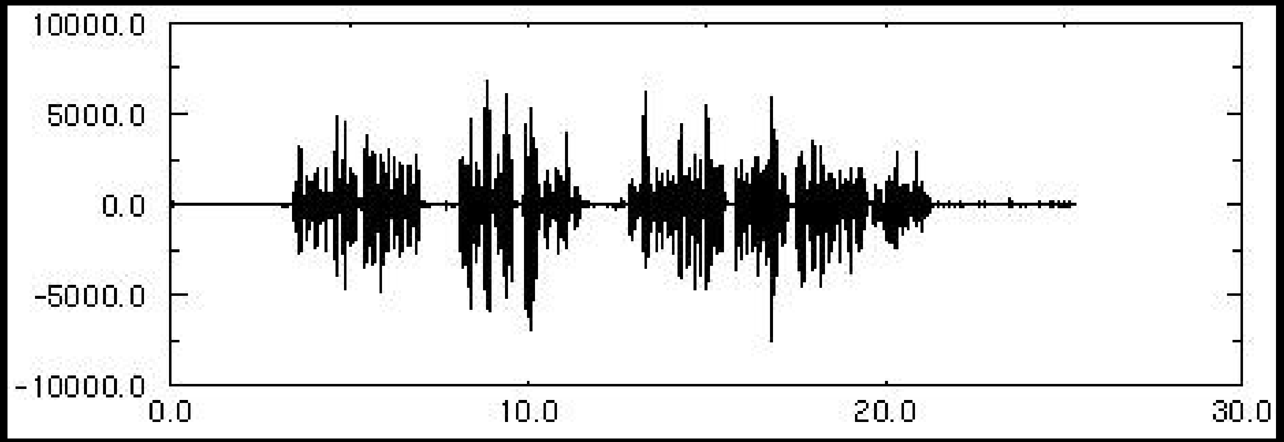
- Recurrent -> perform the same task for every element of a sequence, with the output being depended on the previous computations.

- Have a "memory"





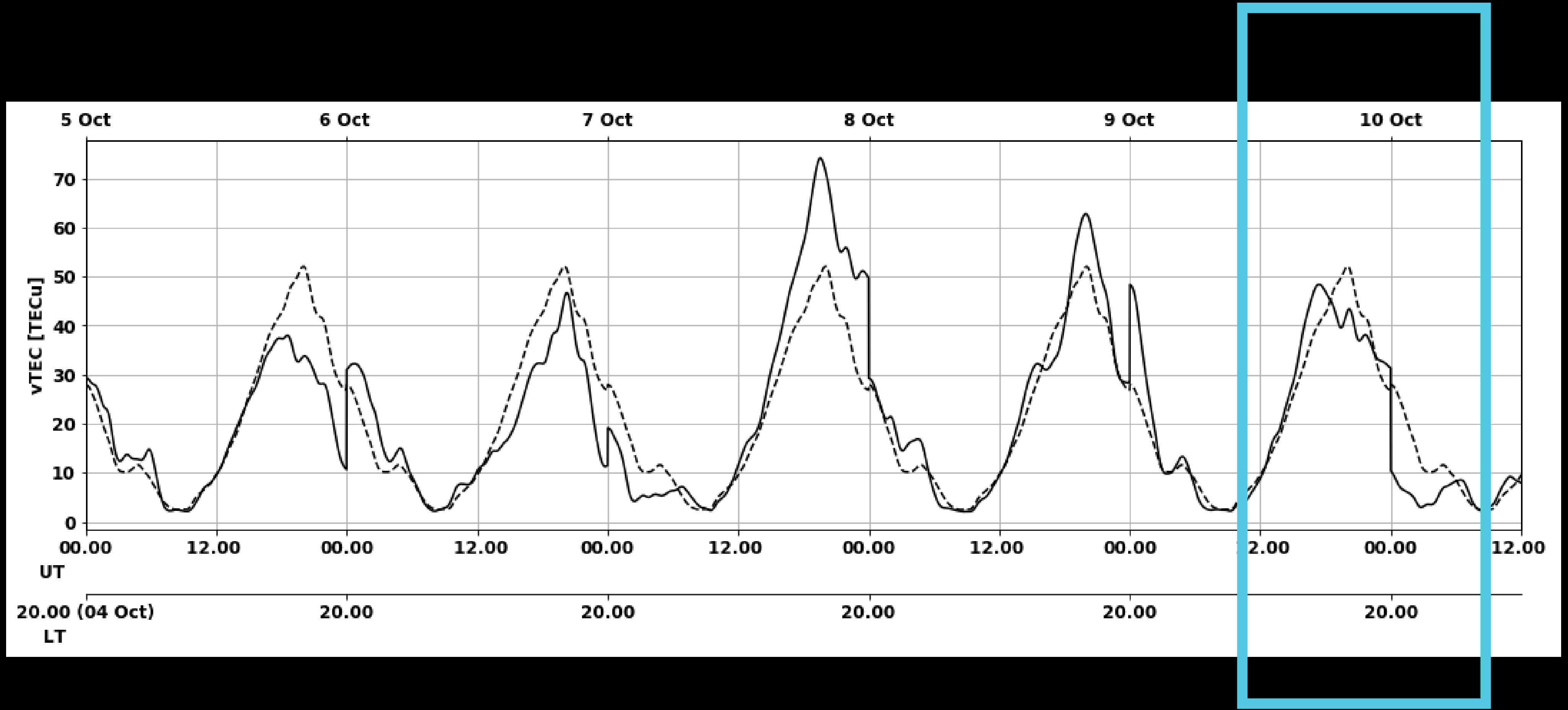




Recurrent Neural Networks

RNNs as an approach to sequence modeling problems

We want to forecast this



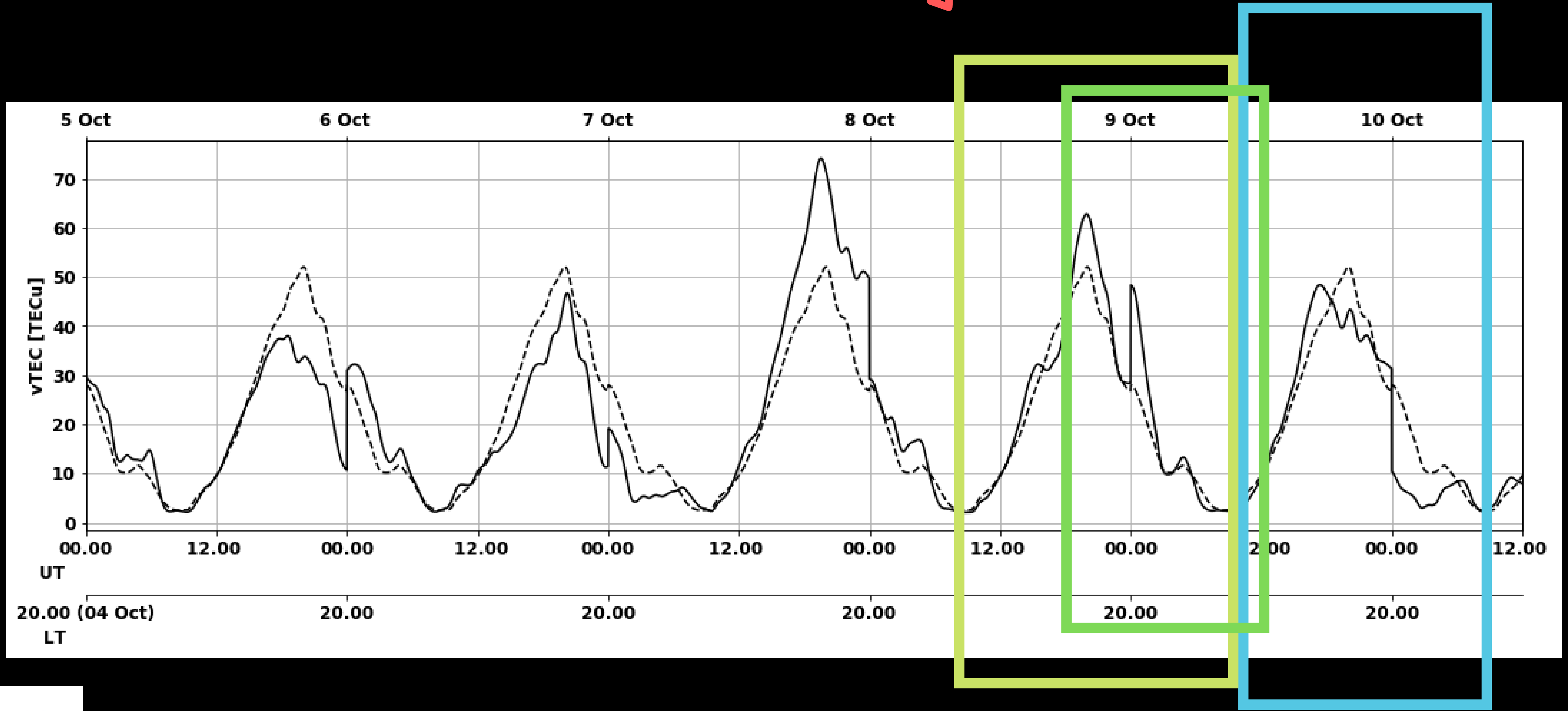
To model sequences, we need to:

- Handle variable-lenght sequences
- Track long-term dependences
- Mantain information about order
- Share parameters across the sequence

Recurrent Neural Networks

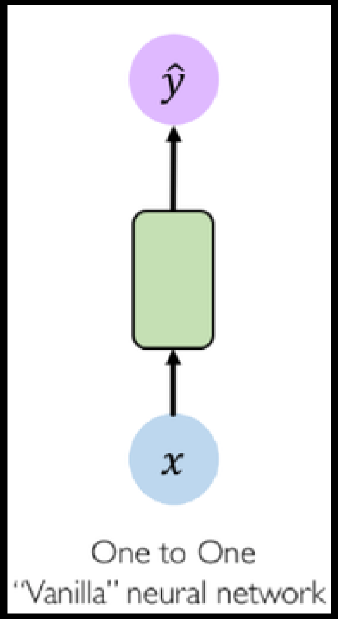
RNNs as an approach to sequence modeling problems

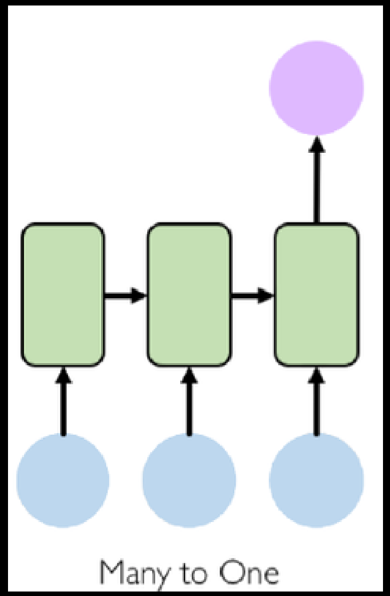
how many steps? (lenght)

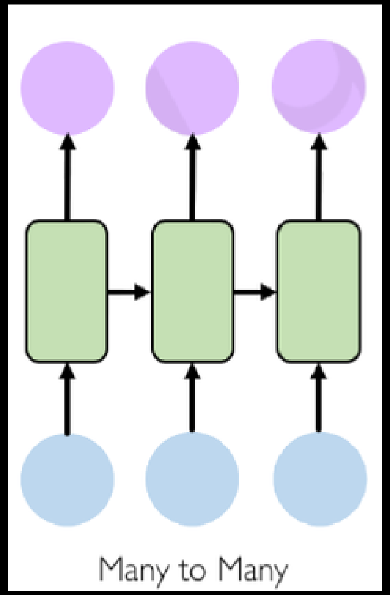


To model sequences, we need to:

- Handle variable-lenght sequences
- Track long-term dependences
- Mantain information about order
- Share parameters across the sequence







and more arquitecutes

TSWC, 2022

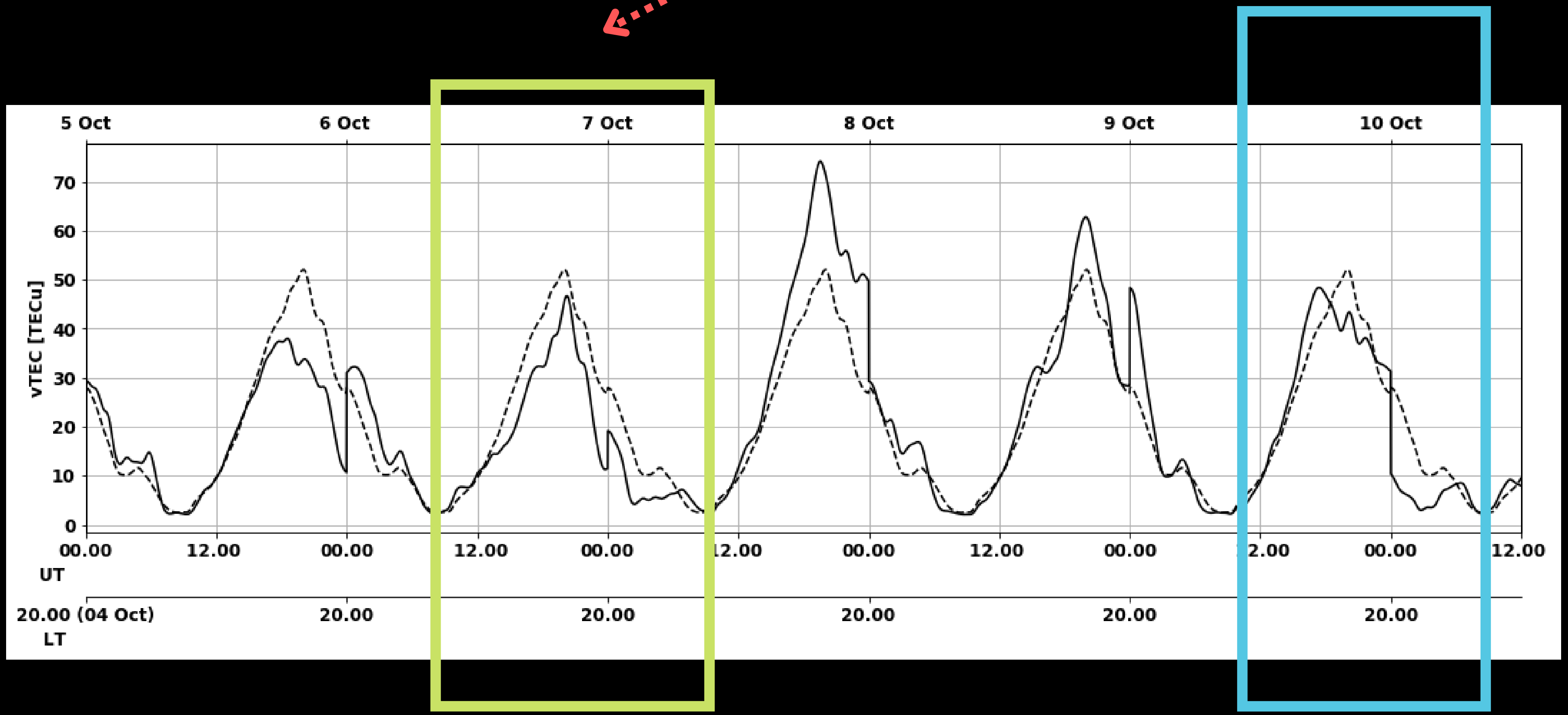
Recurrent Neural Networks

RNNs as an approach to sequence modeling problems

how important is the information on the (far) past ?

To model sequences, we need to:

- Handle variable-lenght sequences
- Track long-term dependences
- Mantain information about order
- Share parameters across the sequence



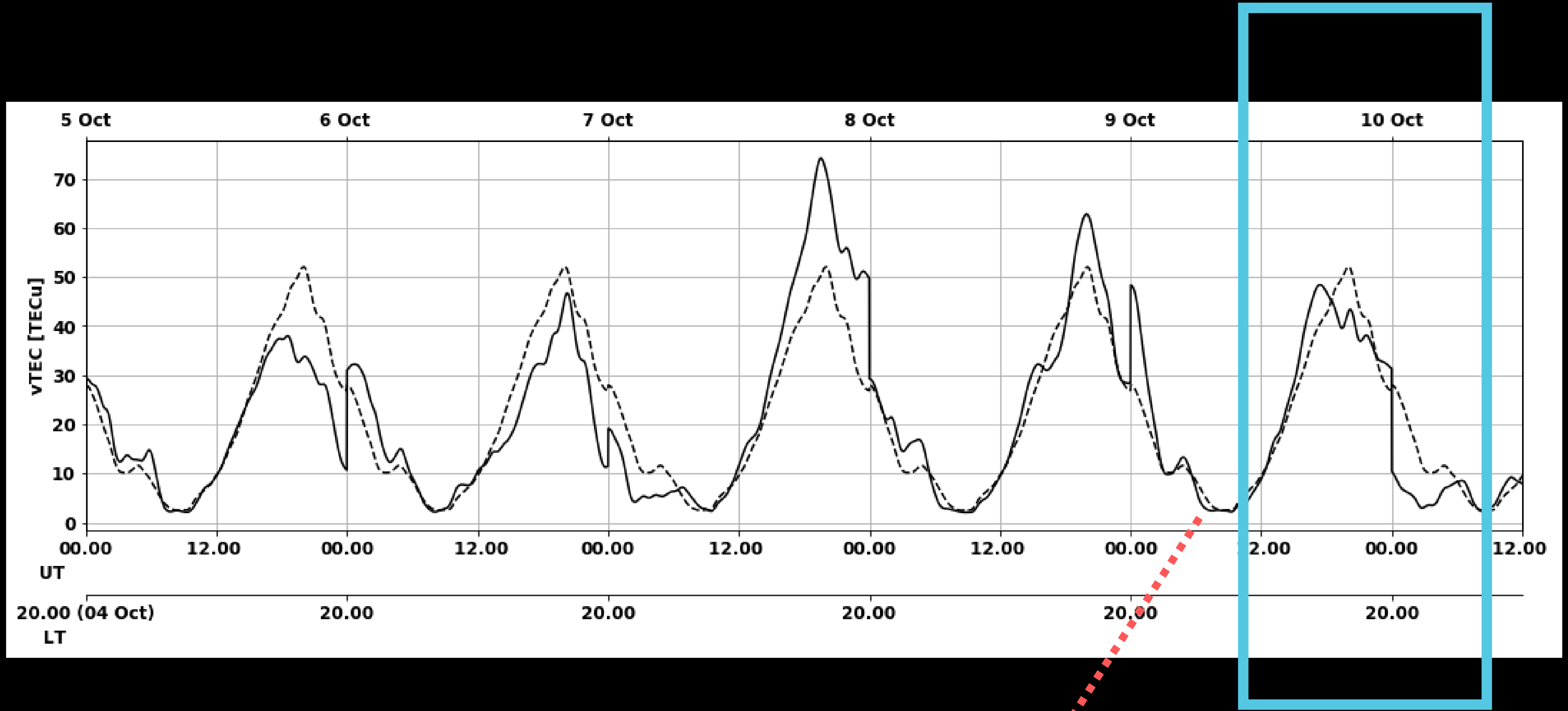
E.g. Ionosphere regular behaviour
(daily, seasonal, solar cycle, ...)

Recurrent Neural Networks

RNNs as an approach to sequence modeling problems

To model sequences, we need to:

- Handle variable-lenght sequences
- Track long-term dependences
- Mantain information about order
- Share parameters across the sequence



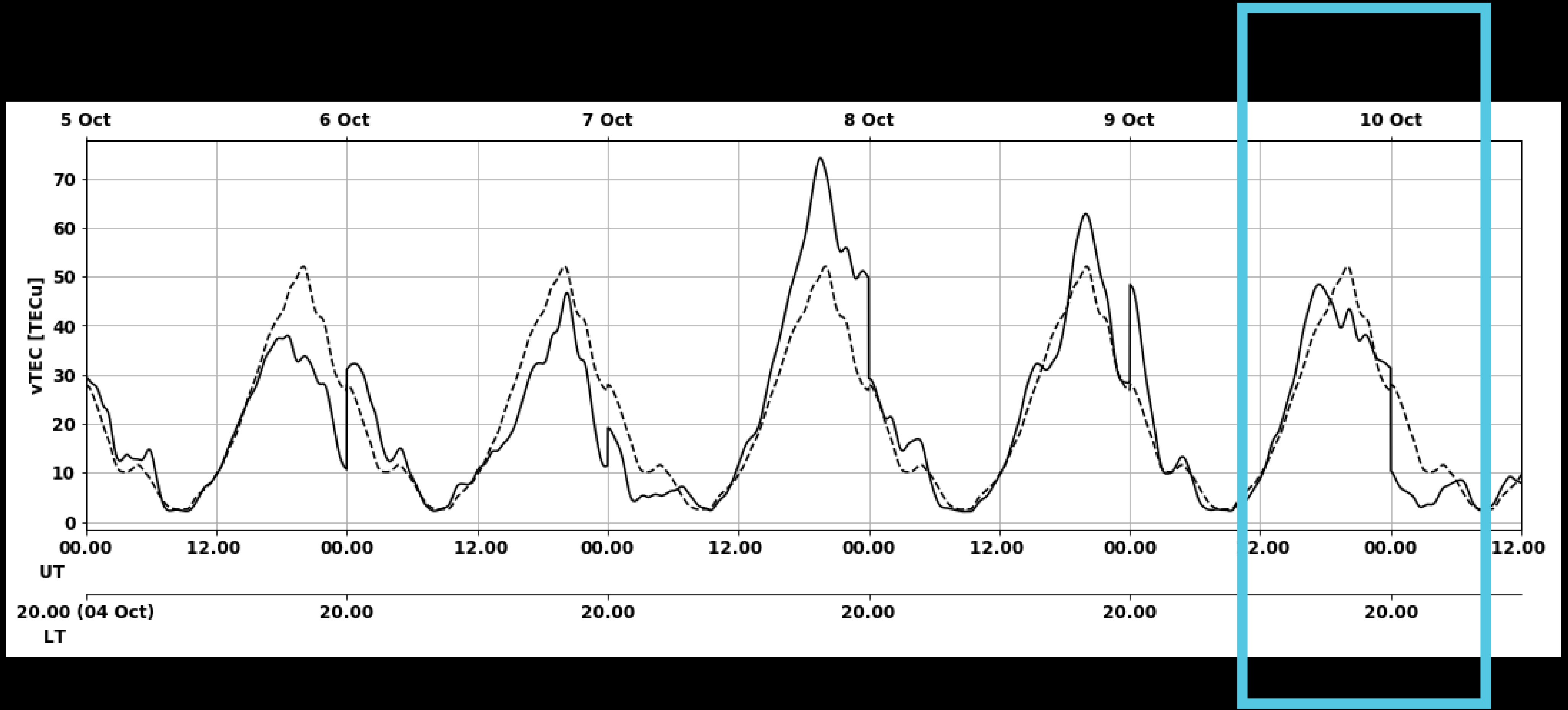
vTEC (t-2), vTEC (t-1), vTEC (t-0)   <> vTEC (t-0), vTEC (t-2), vTEC (t-1)

Recurrent Neural Networks

RNNs as an approach to sequence modeling problems

To model sequences, we need to:

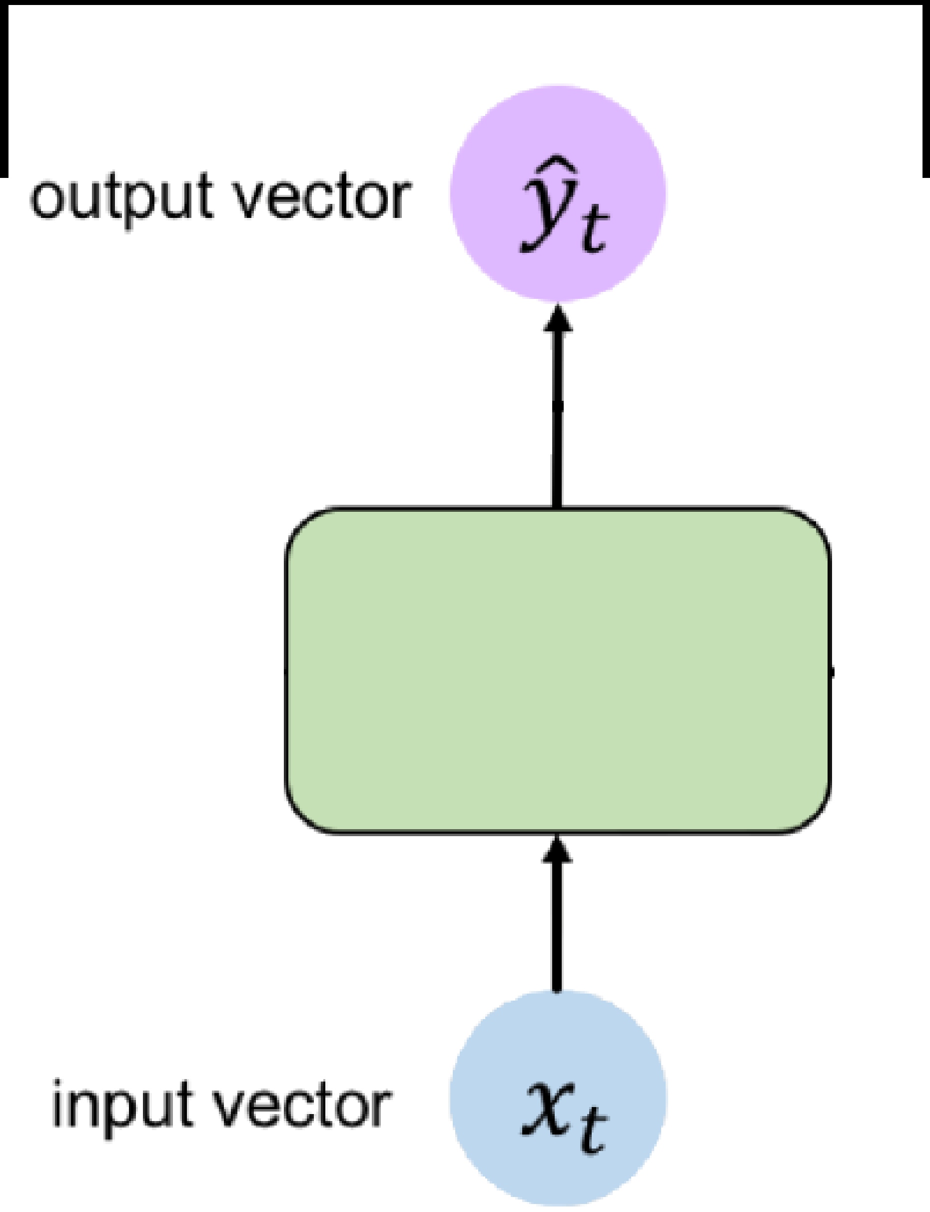
- Handle **variable-lenght** sequences
- Track **long-term** dependences
- Mantain information about **order**
- **Share parameters** across the sequence

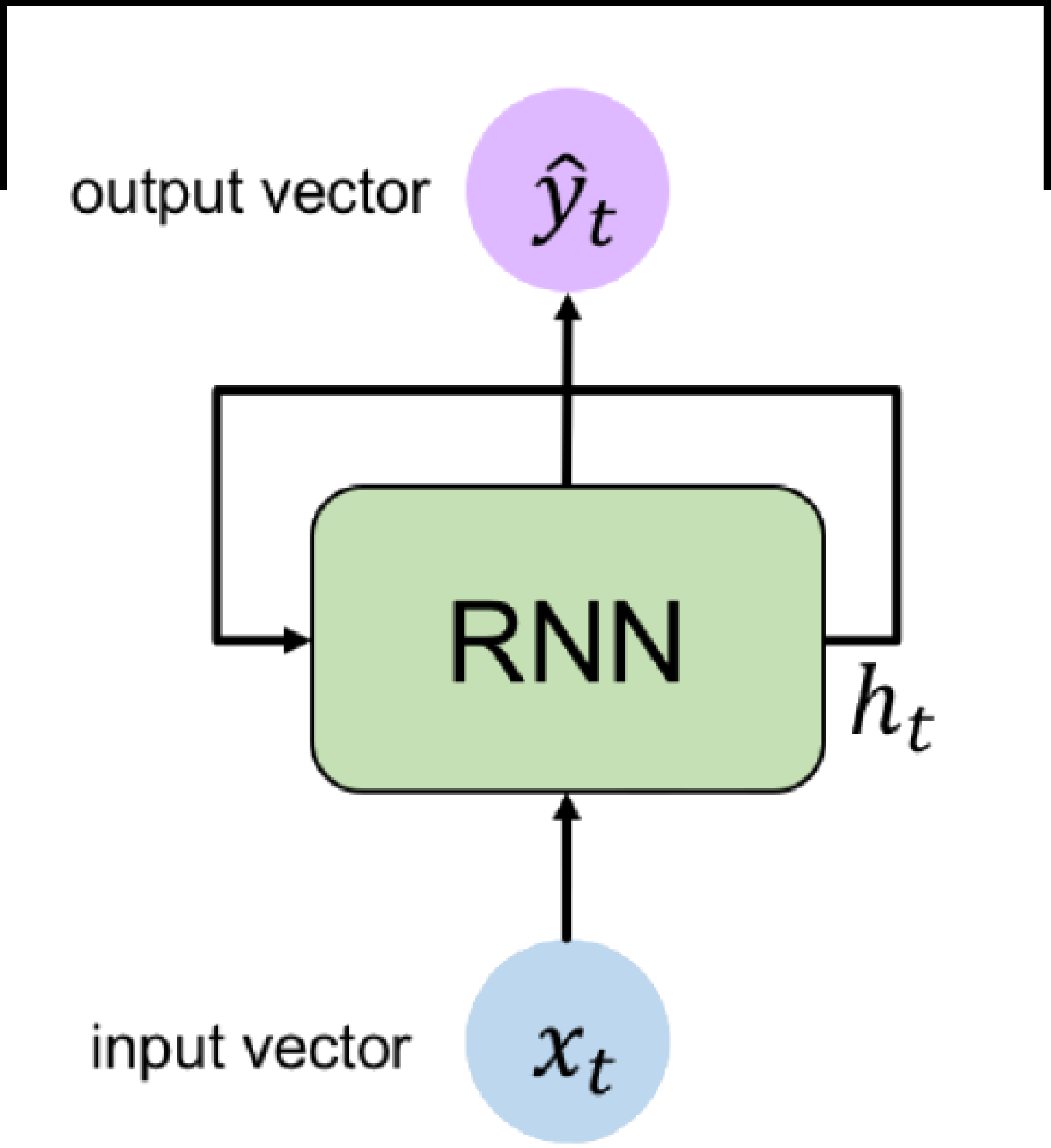


RNNs have a state (h_t), that is updated at each time as a sequence is processed using the same parameters each time step

TSWC, 2022

Recurrent Neural Networks



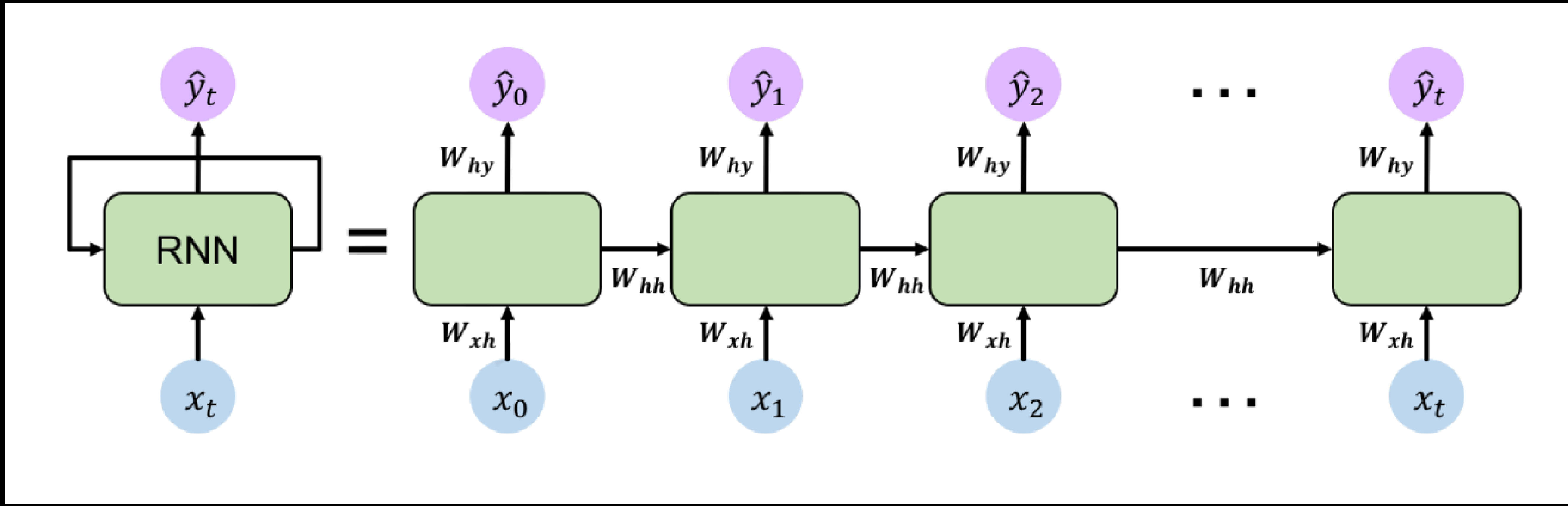


$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

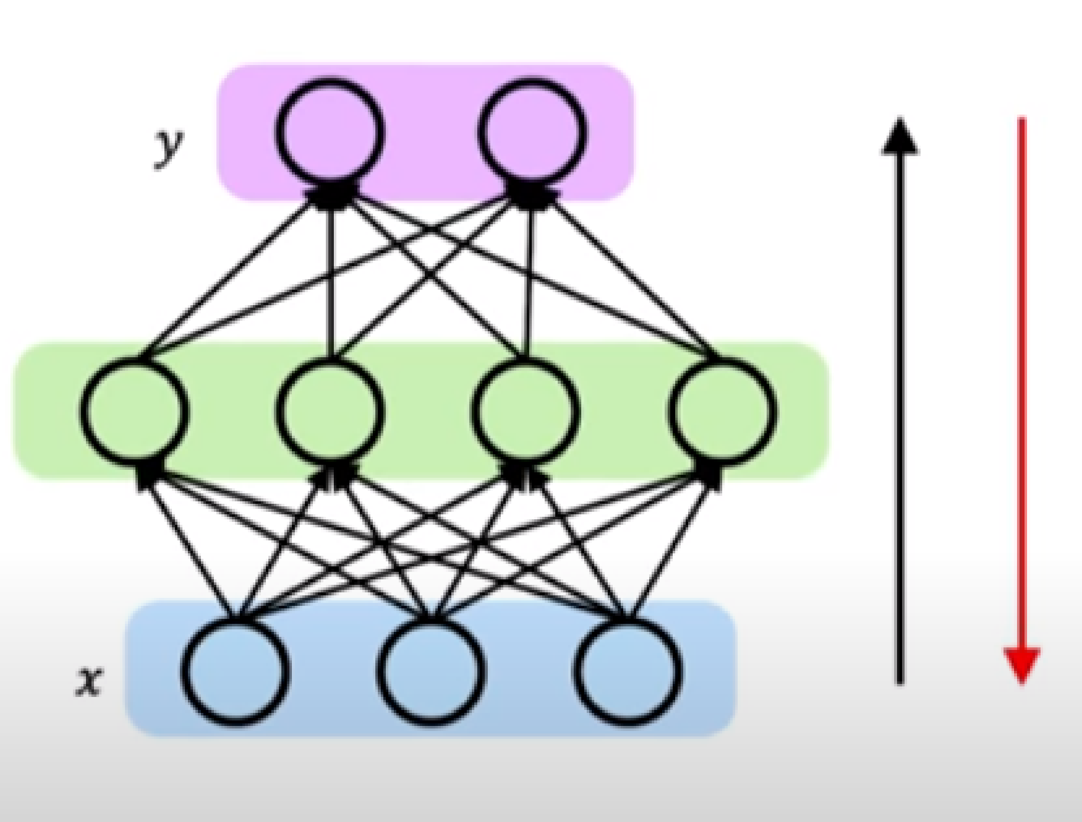
cell state — function parameterized by W — old state — input vector at time step t

Apply a recurrent relation at every time step to process a sequence

Recurrent Neural Networks



Backpropagation through time: long time dependences



$$W^* = \underset{W}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\left(f\left(x^{(i)}; W\right), y^{(i)}\right)$$

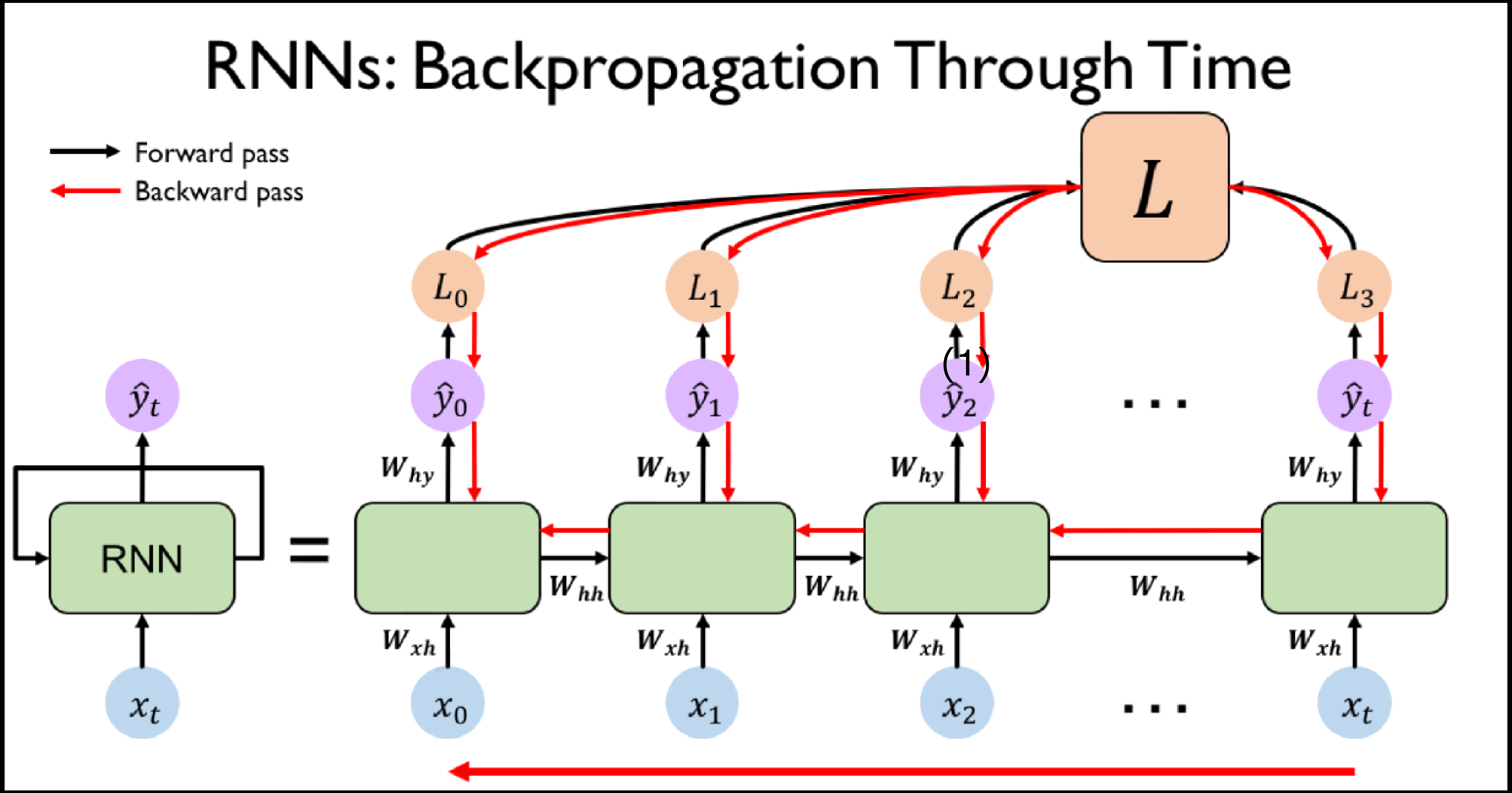
$$W^* = \underset{W}{\operatorname{argmin}}\, J(W)$$

Algorithm

1. Initialize weights randomly $\sim \mathcal{N}(0, \sigma^2)$
2. Loop until convergence:
3. Compute gradient, $\frac{\partial J(W)}{\partial W}$
4. Update weights, $W \leftarrow W - \eta \frac{\partial J(W)}{\partial W}$
5. Return weights

- take the gradient of loss with respect to each parameter
- shift parameters in order to minimize loss

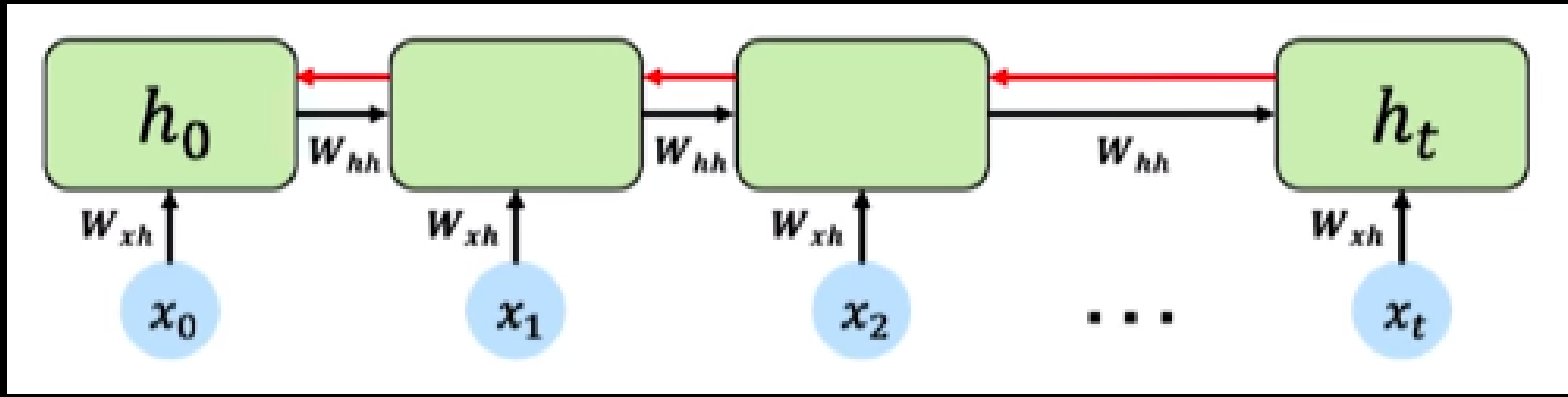
Backpropagation through time: long time dependences



- Compute individual l_i for individual time steps and sum them
- Backpropagate errors individually for each time step and then to all the time steps to the beginning of the sequence.

https : //kharshit.github.io/blog /2019/02/22/backpropagation – through – time

Backpropagation through time: long time dependences



- high computation time!

- Many values >>1 -> exploding gradient (*)
- Many values <<1 -> vanishing gradient

(*) Gradient clipping is a simple technique: If the gradient gets too large, we rescale it to keep it small.

Vanishing gradient problem

Multiply many small numbers together

Errors due to further back time steps
have smaller and smaller gradientes

Bias parameters to capture short -term
dependencies

How to tackle the problem:

- Activation function
- Weight initialization
- Network architecture

Vanishing gradient problem

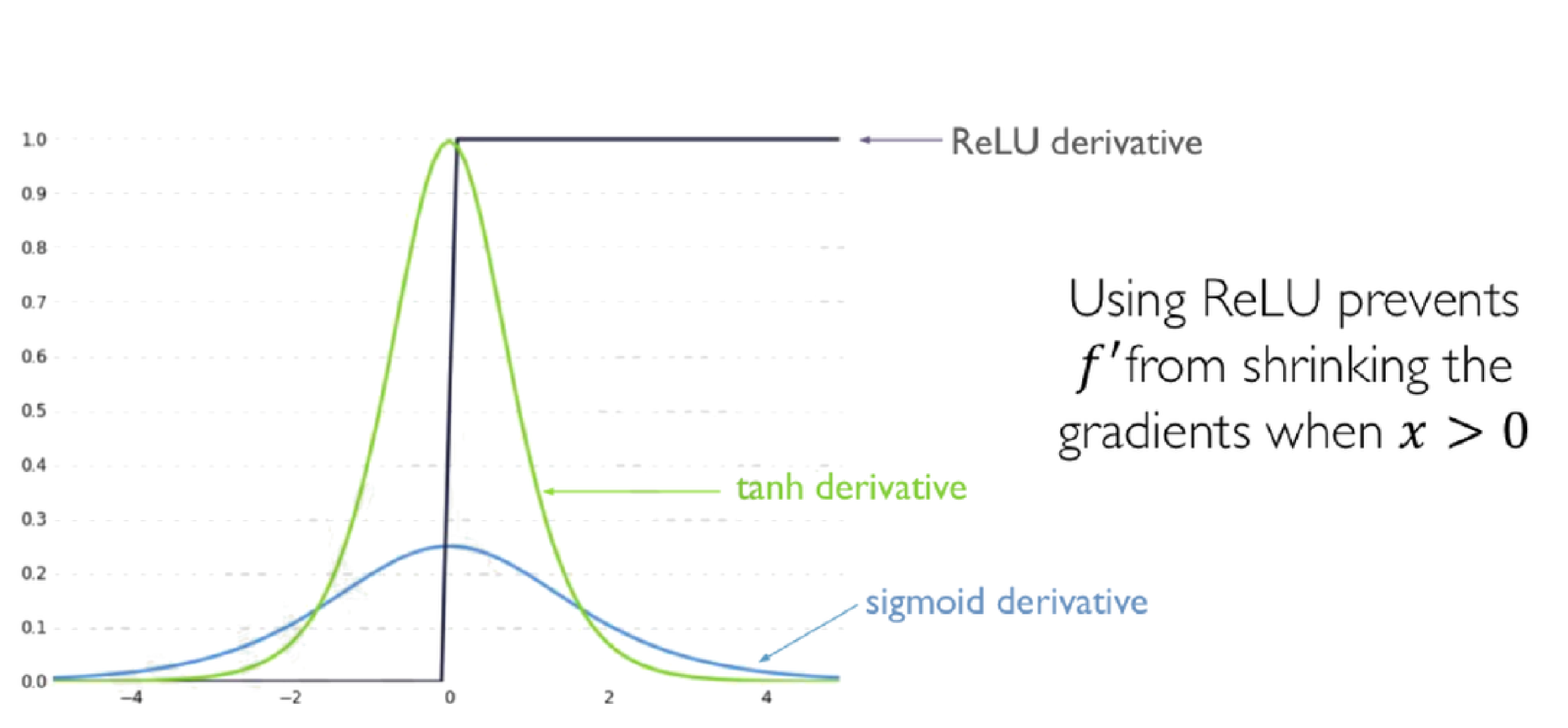
Multiply many small numbers together

Errors due to further back time steps have smaller and smaller gradientes

Bias parameters to capture short -term dependencies

How to tackle the problem:

- Activation function
- Weight initialization
- Network architecture



Vanishing gradient problem

Multiply many small numbers together

Errors due to further back time steps have smaller and smaller gradientes

Bias parameters to capture short -term dependencies (we "loose" long-term dependencies)

How to tackle the problem:

- Activation function
- Weight initialization
- Network architecture

$$I_n = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

Initialize weigths to identity matriz
Initialize biases to zero

prevent the weights to shrinking to zero

Vanishing gradient problem

Multiply many small numbers together

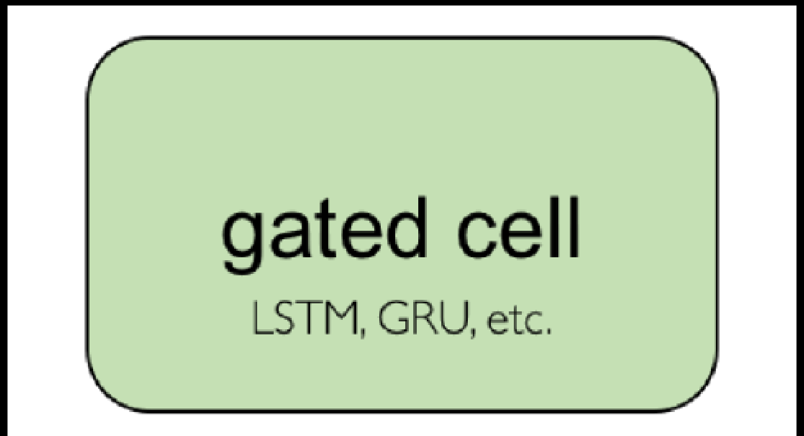
Errors due to further back time steps have smaller and smaller gradientes

Bias parameters to capture short -term dependencies (we "loose" long-term dependencies)

How to tackle the problem:

- Activation function
- Weight initialization
- Network architecture

more rubust solution

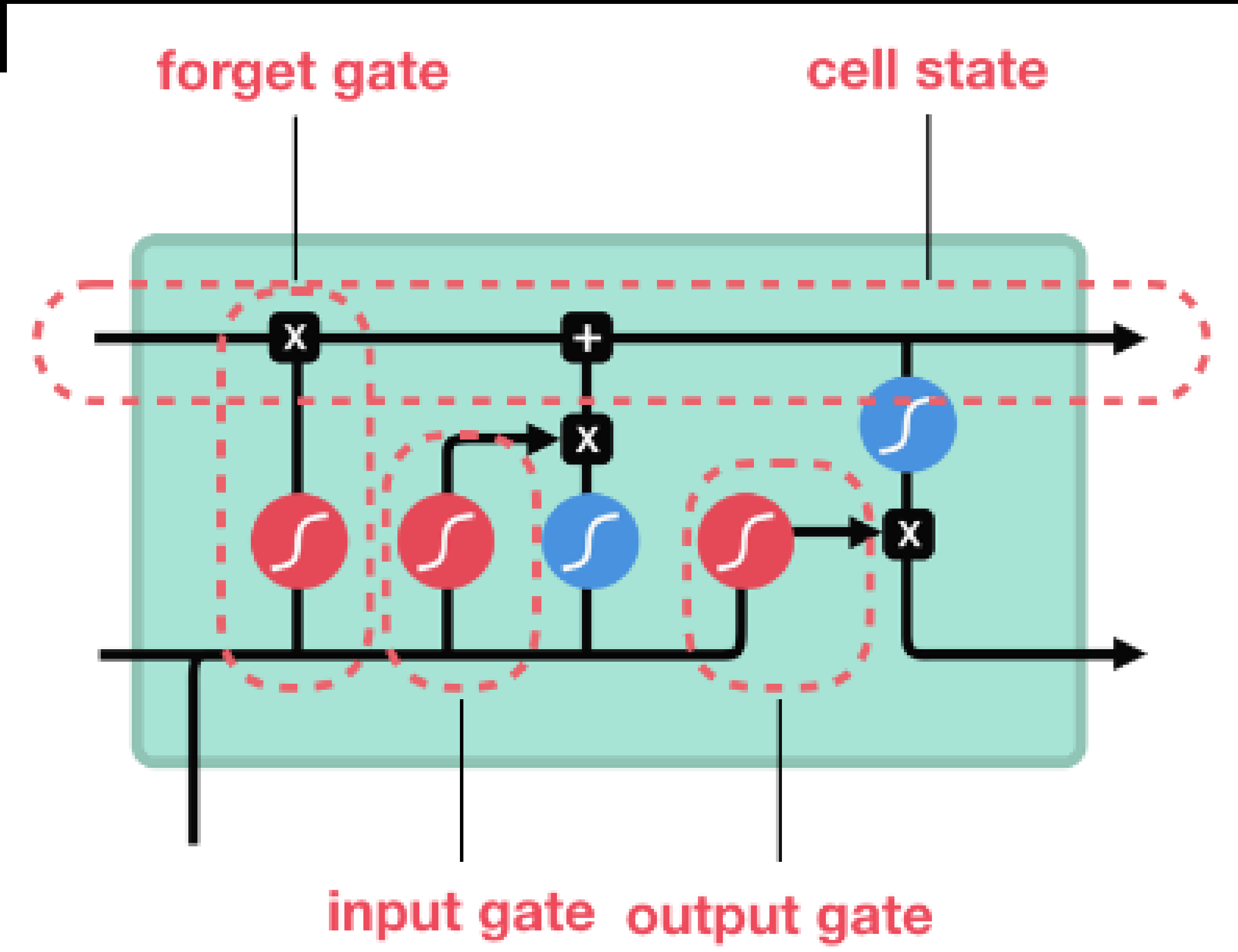


- Use a more complex recurrent unit with gates to control what information is passed through.

- gates selectively add or remove information within each recurrent uniit

Long short term memory (LSTM)

Gates: 1) Forget 2) Input (store) 3)Update 4) Output



How it works:

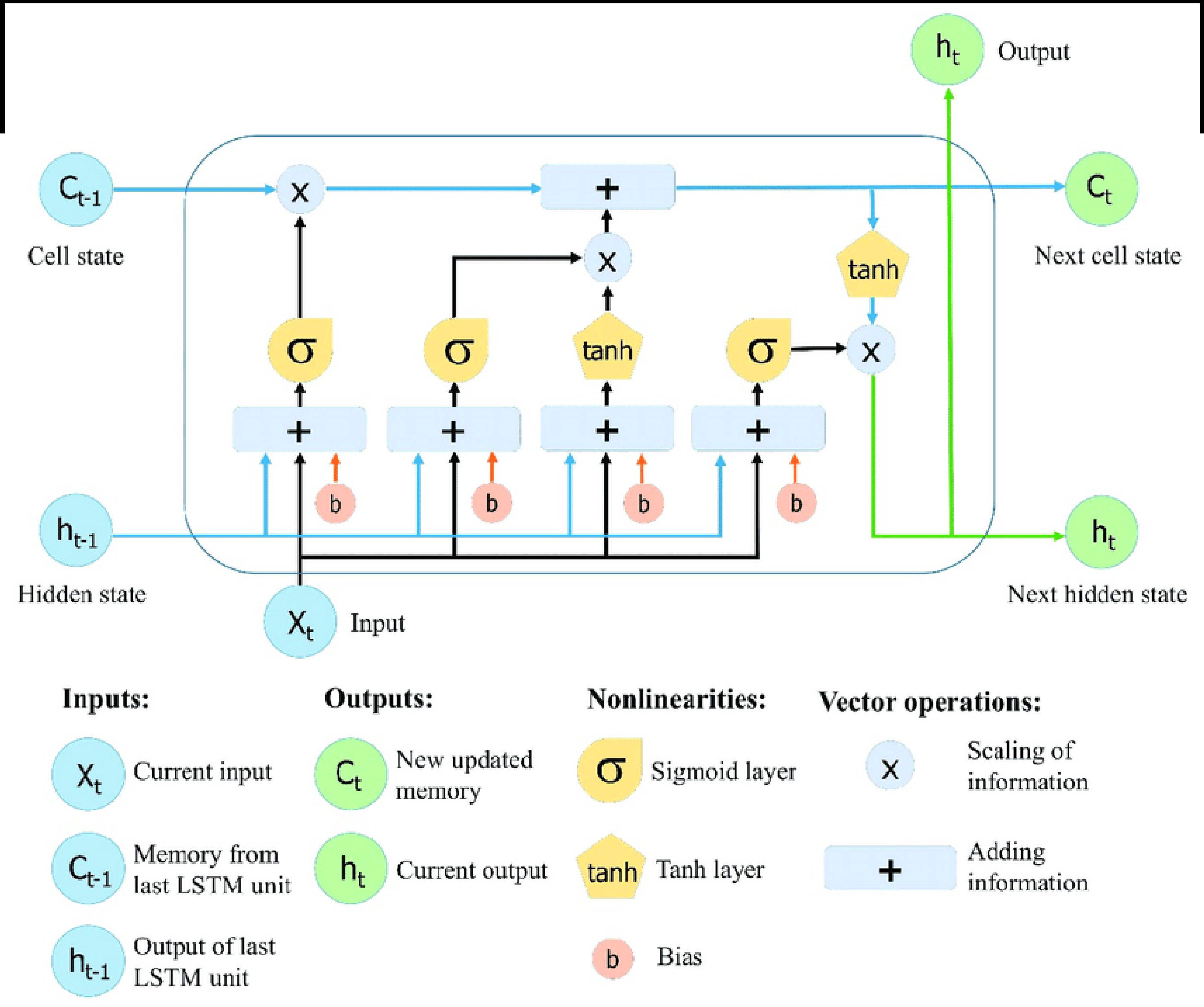
1)Maintain a cell state
2)Use gates to control the flow of information
- Forget gate gets rid of irrelevant information
- Store relevant information from the current input
- Selectively update cell state

Output gate returns a filtered version of the cell state
3) Backpropagation TT with partially uninterrupted gradient flow

Long short term memory (LSTM)

TSWC, 2022

Gates: 1) Forget 2) Input (store) 3)Update 4) Output



How it works:

1)Maintain a cell state
2)Use gates to control the flow of information

- Forget gate gets rid of irrelevant information
- Store relevant information from the current input
- Selectively update cell state

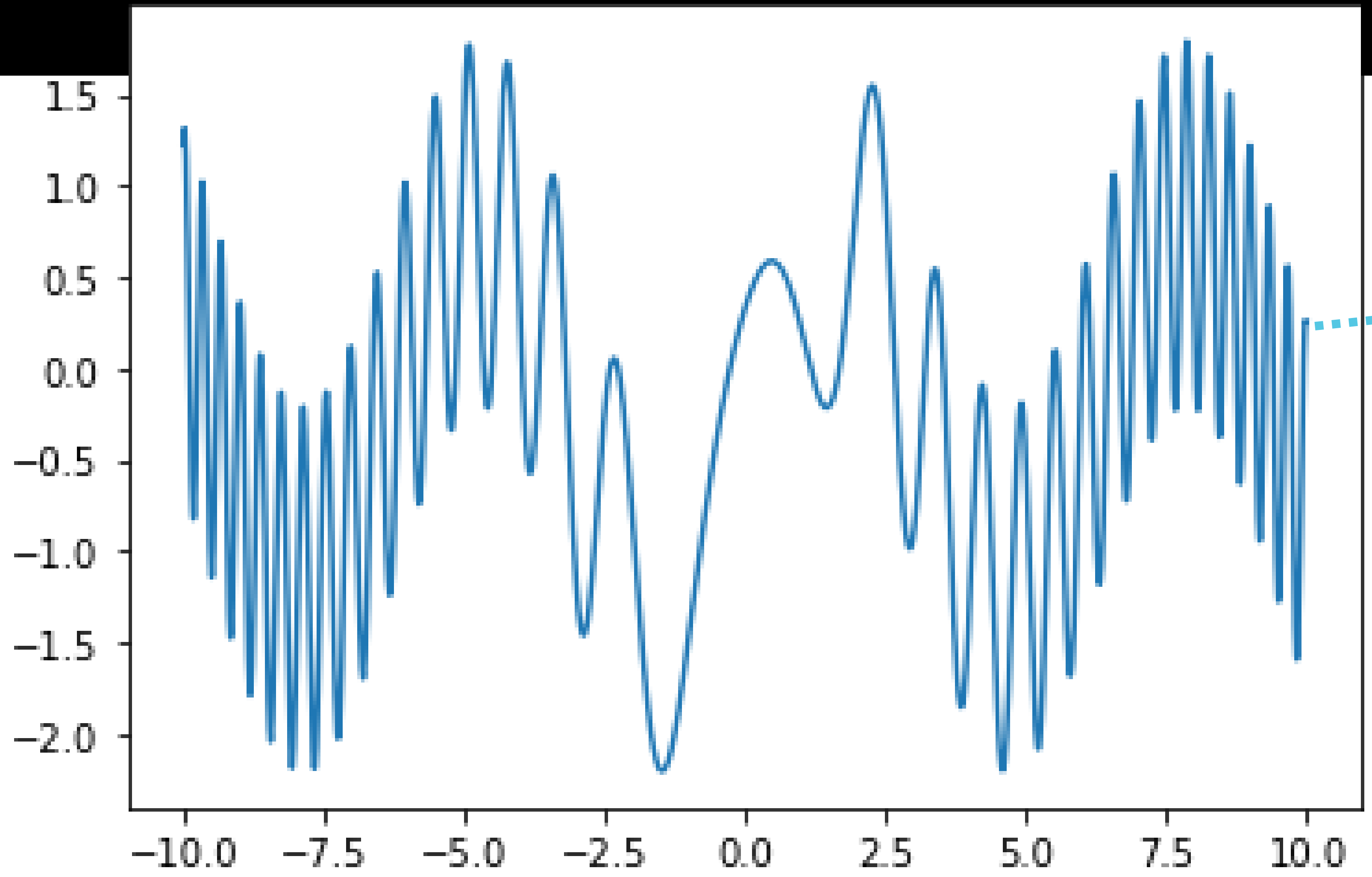
Output gate returns a filtered version of the cell state
3) Backpropagation TT with partially uninterrupted gradient flow

LSTM "not all that glitters is gold"

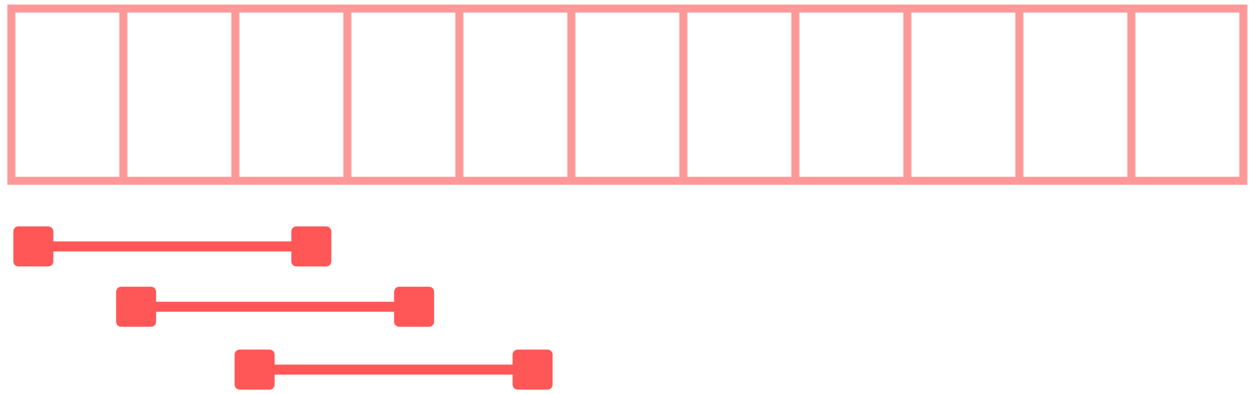
- << vanishing gradient problem .... but it doesn't completely remove it.
- >> computational resources
- affected by different random weight initialization
- Drop-out difficult to implement
- Prone to overfitting
- << performance for very long time problems

LSTM example

vanilla LSTM

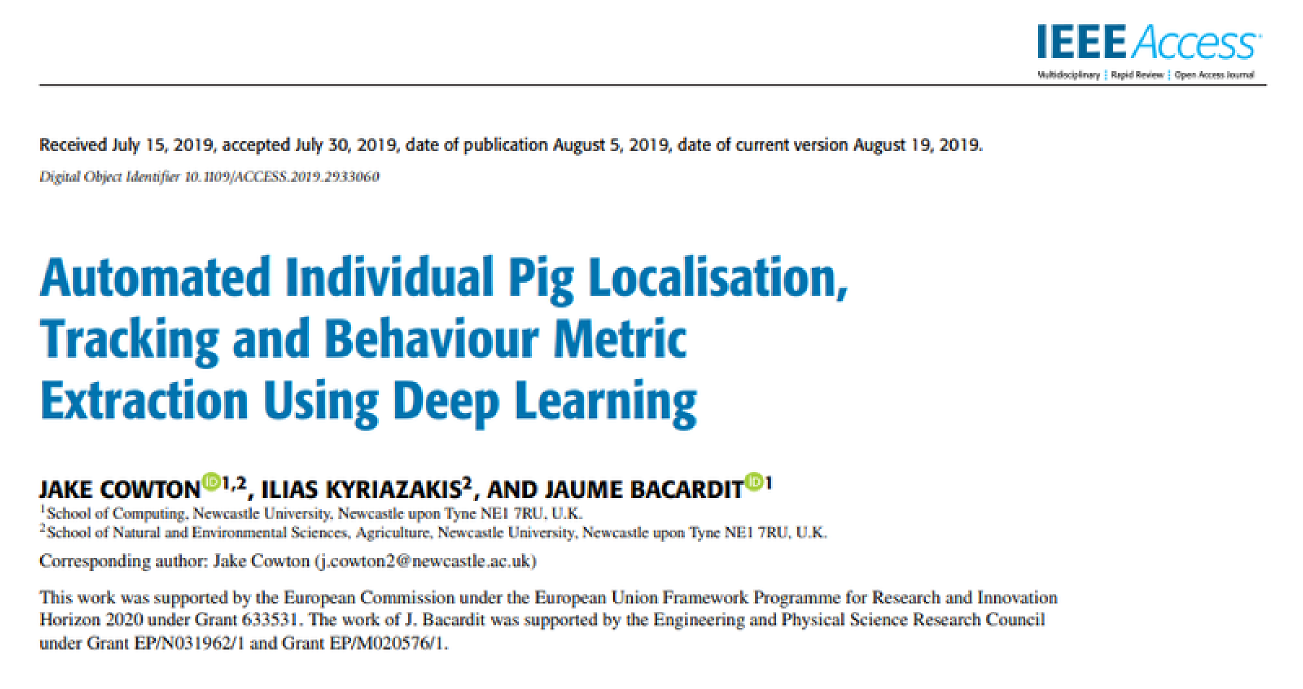


training set t in [-10, 10)
test set t=10



the training set is an array of sequences (len = 3)

Convolutional Neural Networks

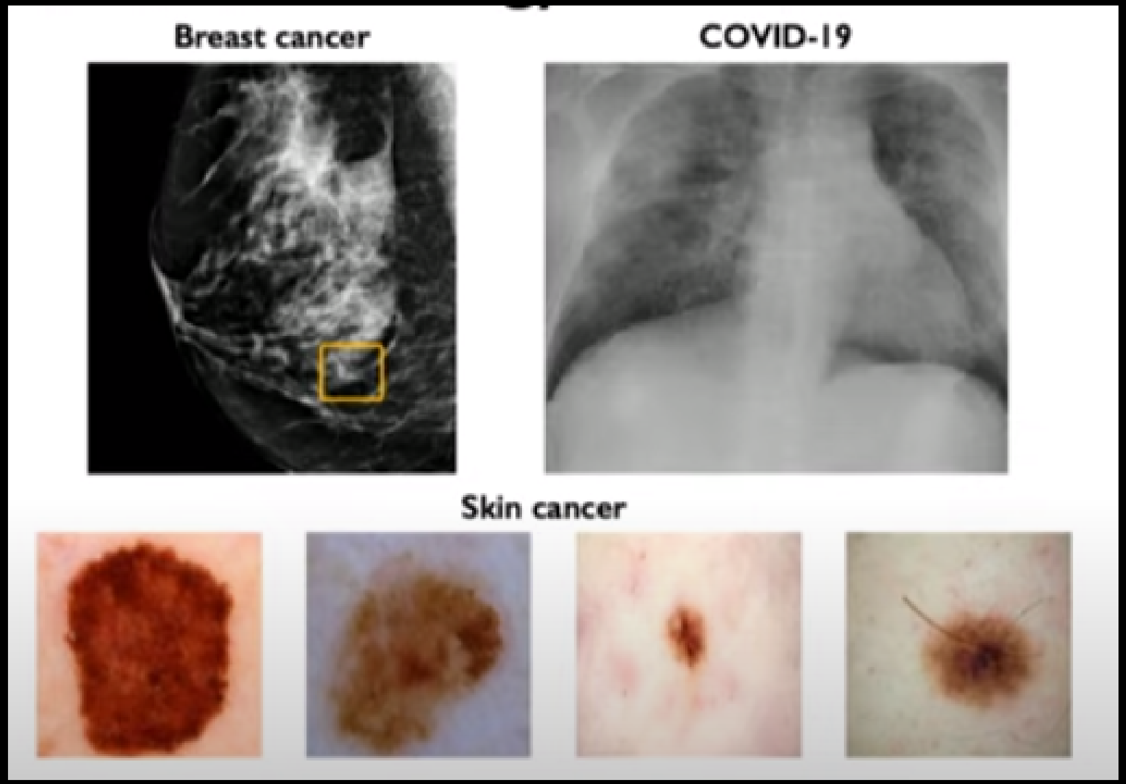

IEEE Access
Received July 15, 2019, accepted July 30, 2019, date of publication August 5, 2019, date of current version August 19, 2019.
Digital Object Identifier 10.1109/ACCESS.2019.2933060

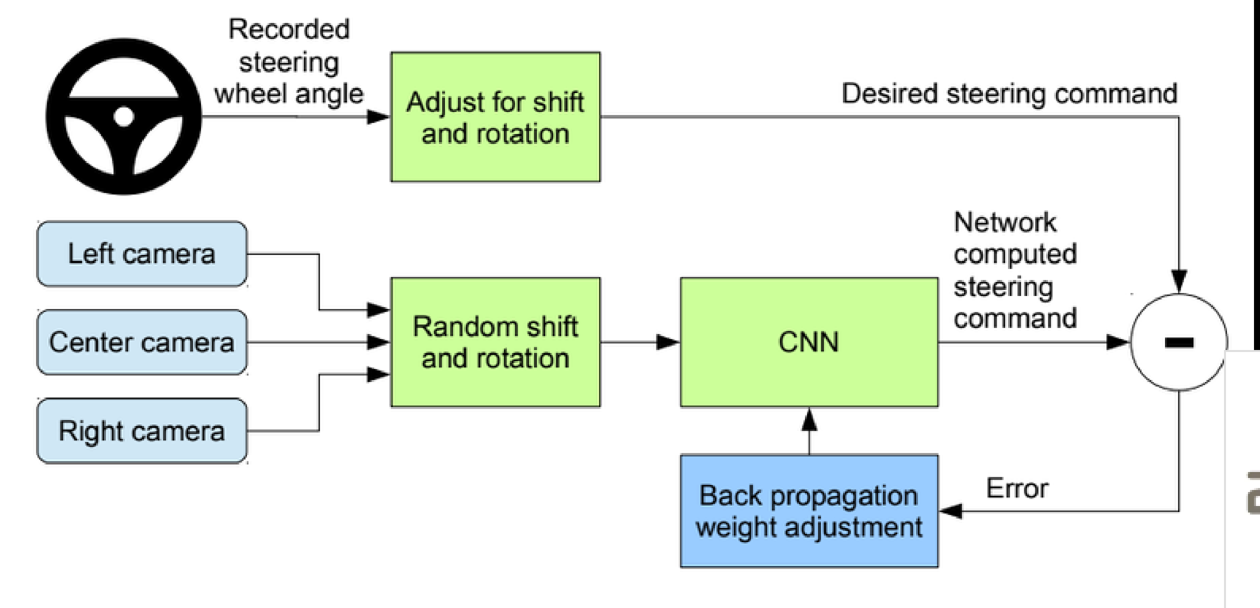
Automated Individual Pig Localisation, Tracking and Behaviour Metric Extraction Using Deep Learning

JAKE COWTON[1,2], ILIAS KYRIAZAKIS[2], AND JAUME BACARDIT[1]
[1]School of Computing, Newcastle University, Newcastle upon Tyne NE1 7RU, U.K.
[2]School of Natural and Environmental Sciences, Agriculture, Newcastle University, Newcastle upon Tyne NE1 7RU, U.K.
Corresponding author: Jake Cowton (j.cowton2@newcastle.ac.uk)

This work was supported by the European Commission under the European Union Framework Programme for Research and Innovation Horizon 2020 under Grant 633531. The work of J. Bacardit was supported by the Engineering and Physical Science Research Council under Grant EP/N031962/1 and Grant EP/M020576/1.

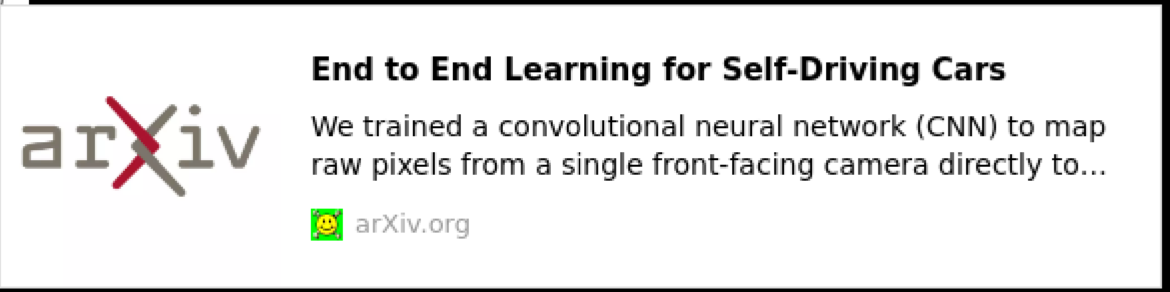
FIGURE 8. Four sample images from our pig detection test set processed by the Faster R-CNN with the feature extraction layers pre-trained on ImageNet, the rest pre-trained on Pascal Visual Object Classes Challenge 2007 and an additional fully-connected layer for the pig dataset. Detections to the left of the red wall are ignored. The top left image is from the low-light test segment. The top right image is from the densely packed test-segment. The bottom left image is from the overexposed test segment. The bottom right image is from the many pigs test segment.





Computer Vision. Some applications:

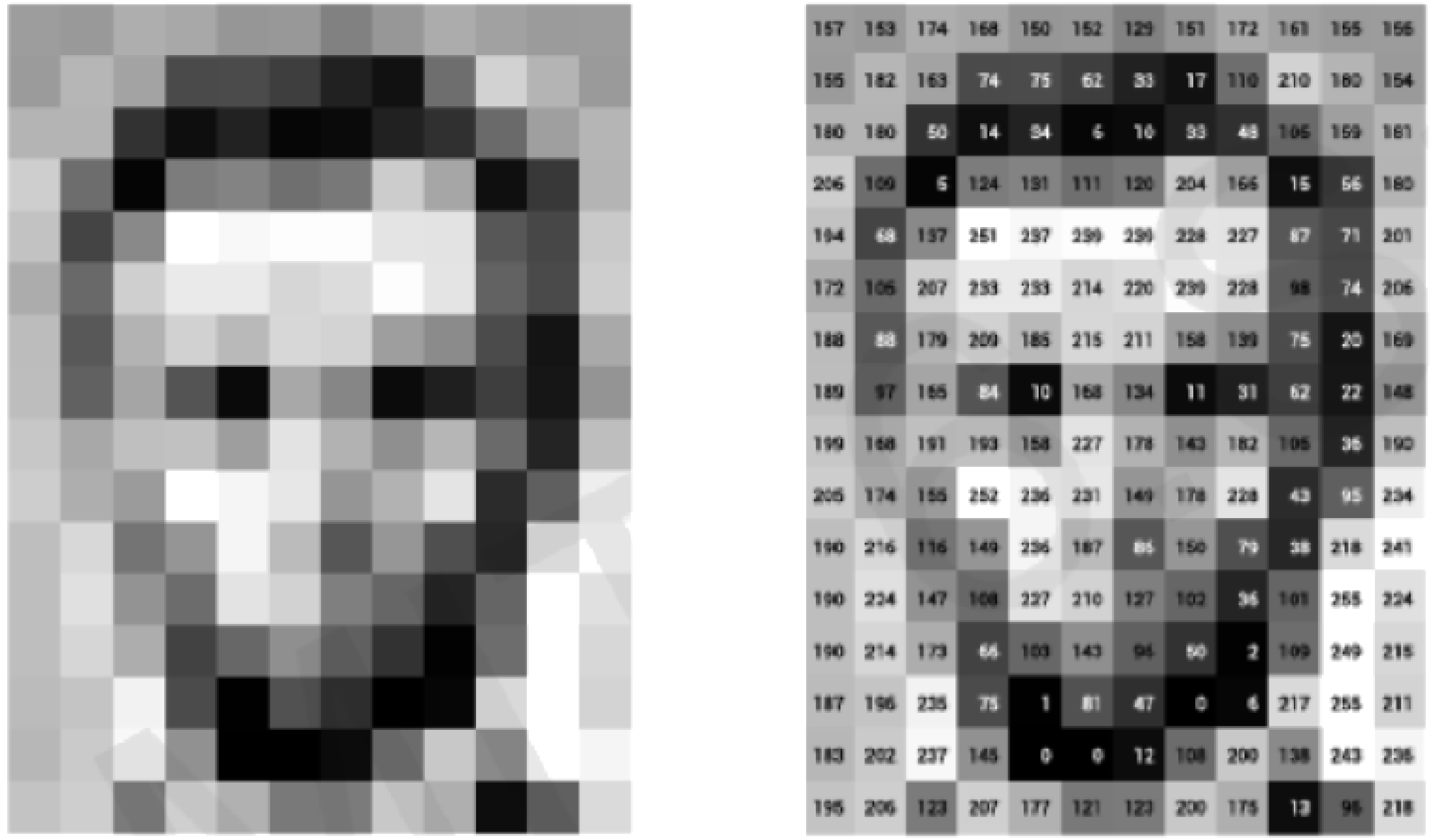
- Facial detection and recognition
- Healthcare, medicine and biology
- Self-driving vehicles



Convolutional Neural Networks

Computer Vision. Some applications:

- Facial detection and recognition
- Healthcare, medicine and biology
- Self-driving vehicles



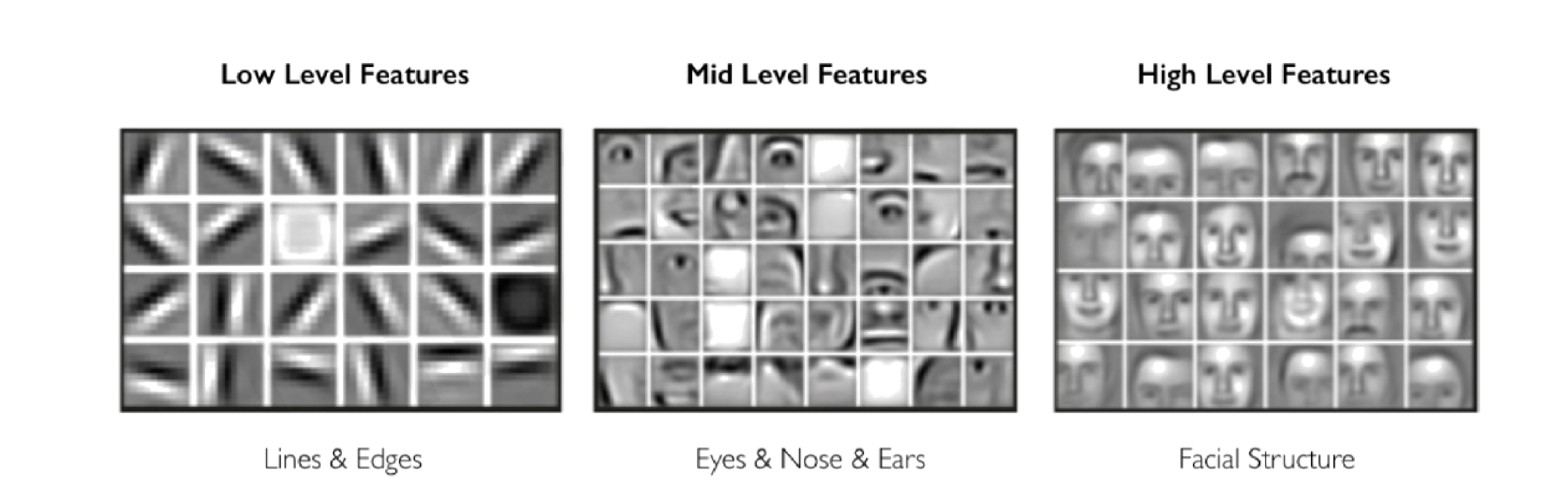
What the computer sees

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	106	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

An image is just a matrix of numbers [0,255]!
i.e., 1080x1080x3 for an RGB image

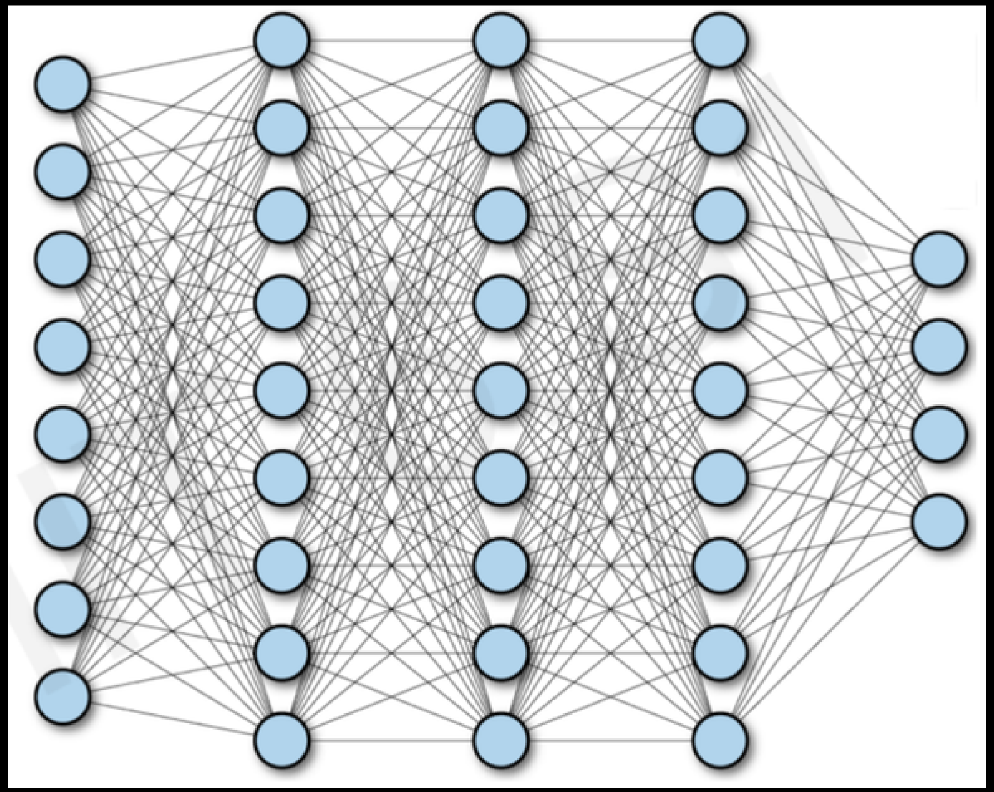
Features

(As in other NN problems)
- Regression
- Classification
- High level feature detection



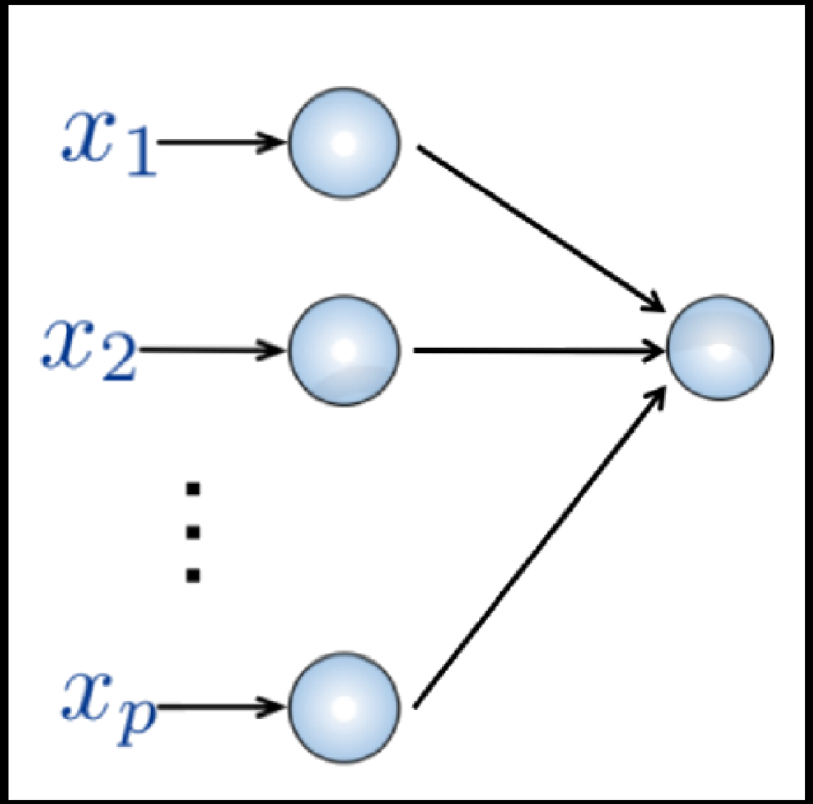
feature extraction from the data!
Learn hierarchy of features!

Fully connceted NN



Input:
- 2D image
- vector of pixel values

(flatten the image)

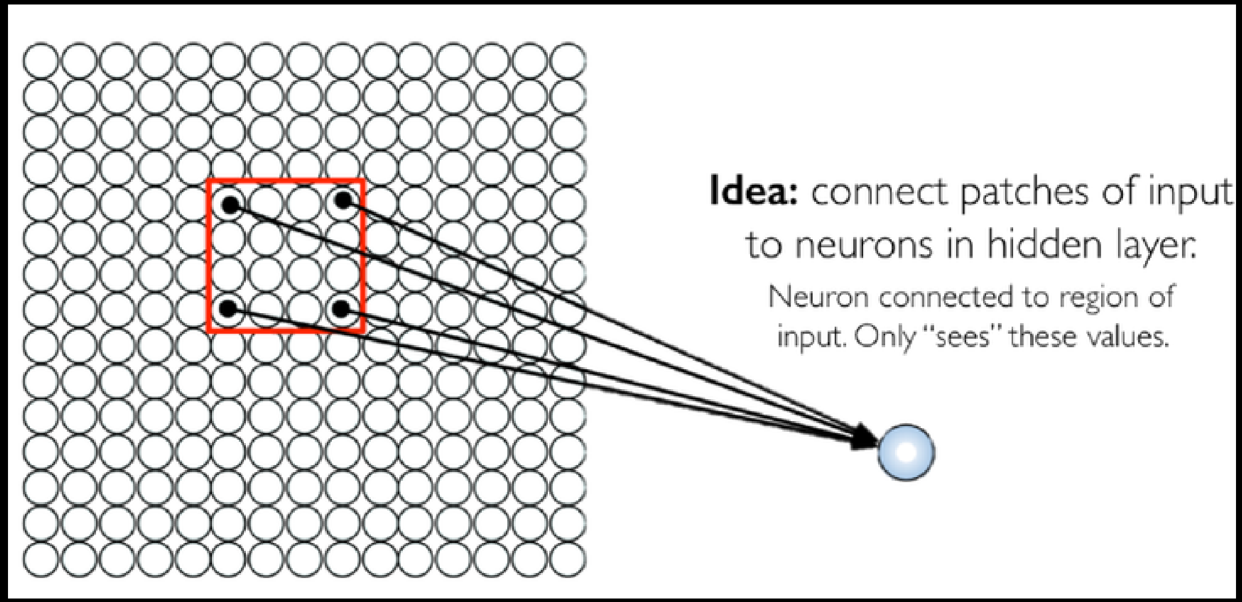


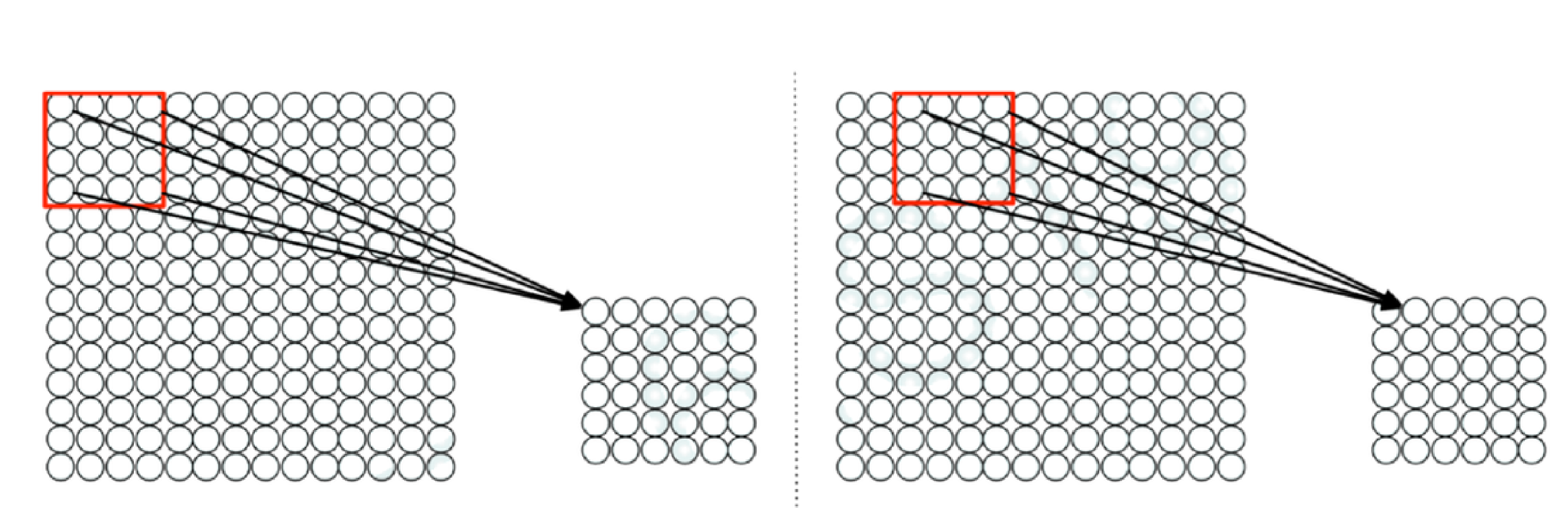
Fully connected:
- Connect neuron in hidden layer to all neurons in the input layer
- No spatial information!
- Lot of parameters

How to add spatial structure in the input?

Using spatial structure

Input:
- 2D image
- Array of pixel values

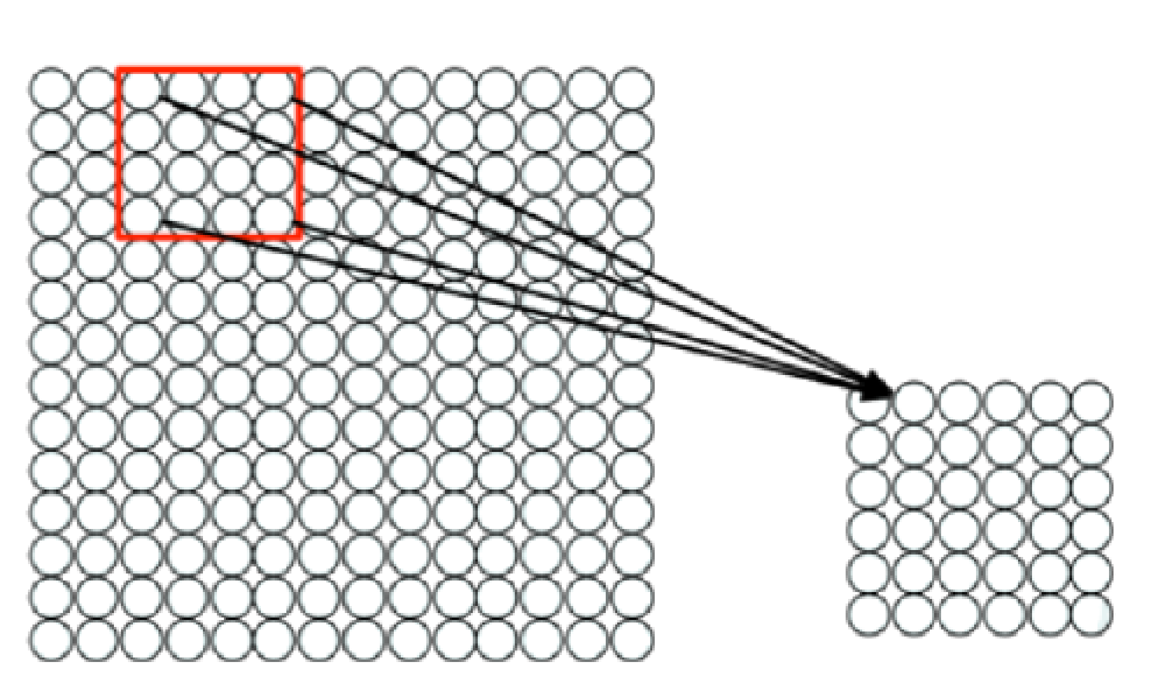




Sliding window to define conections,(connect patch in input layer to a single neuron)

The key: how can we weight the patch to detect particular features?

Using spatial structure

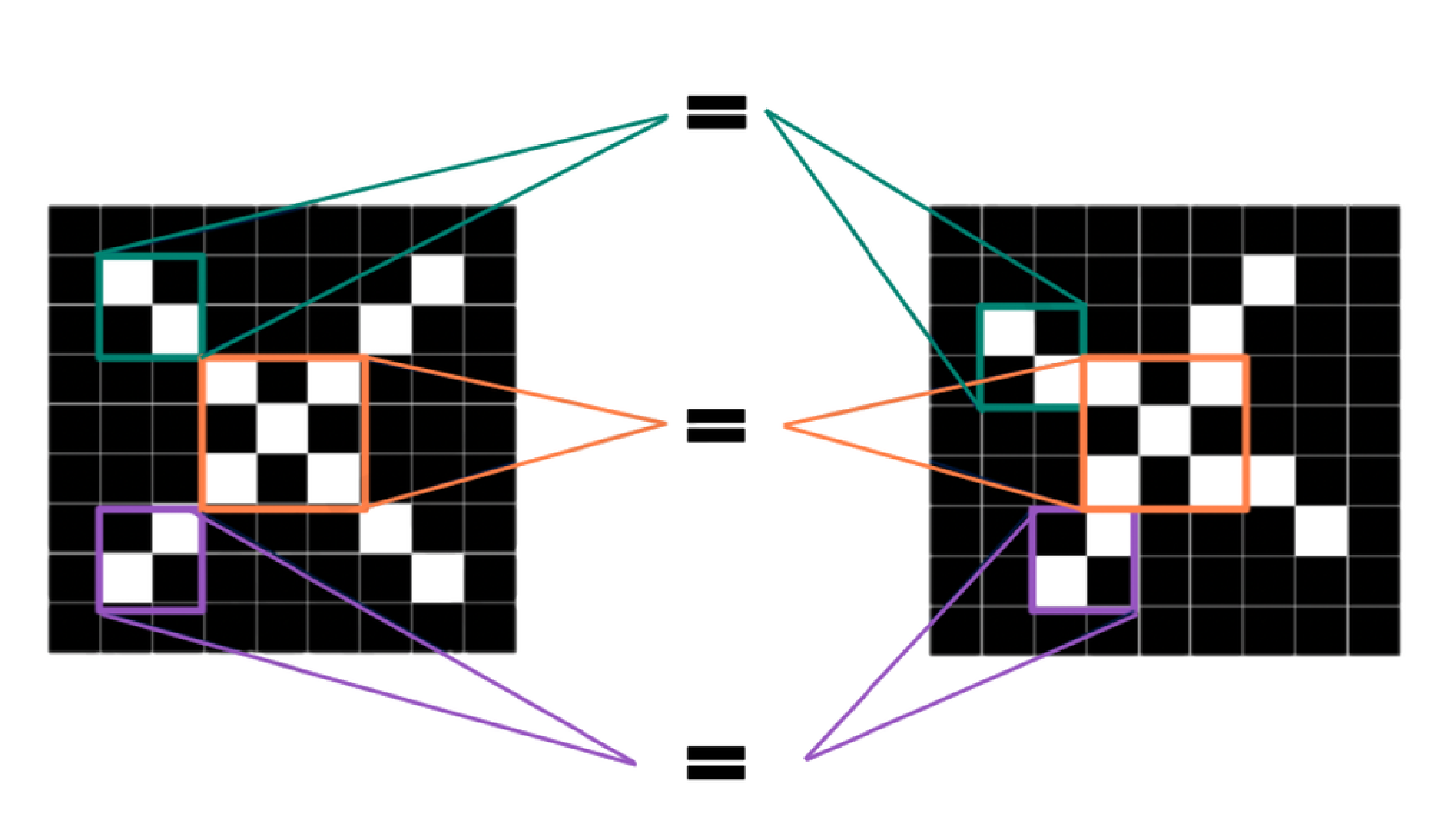


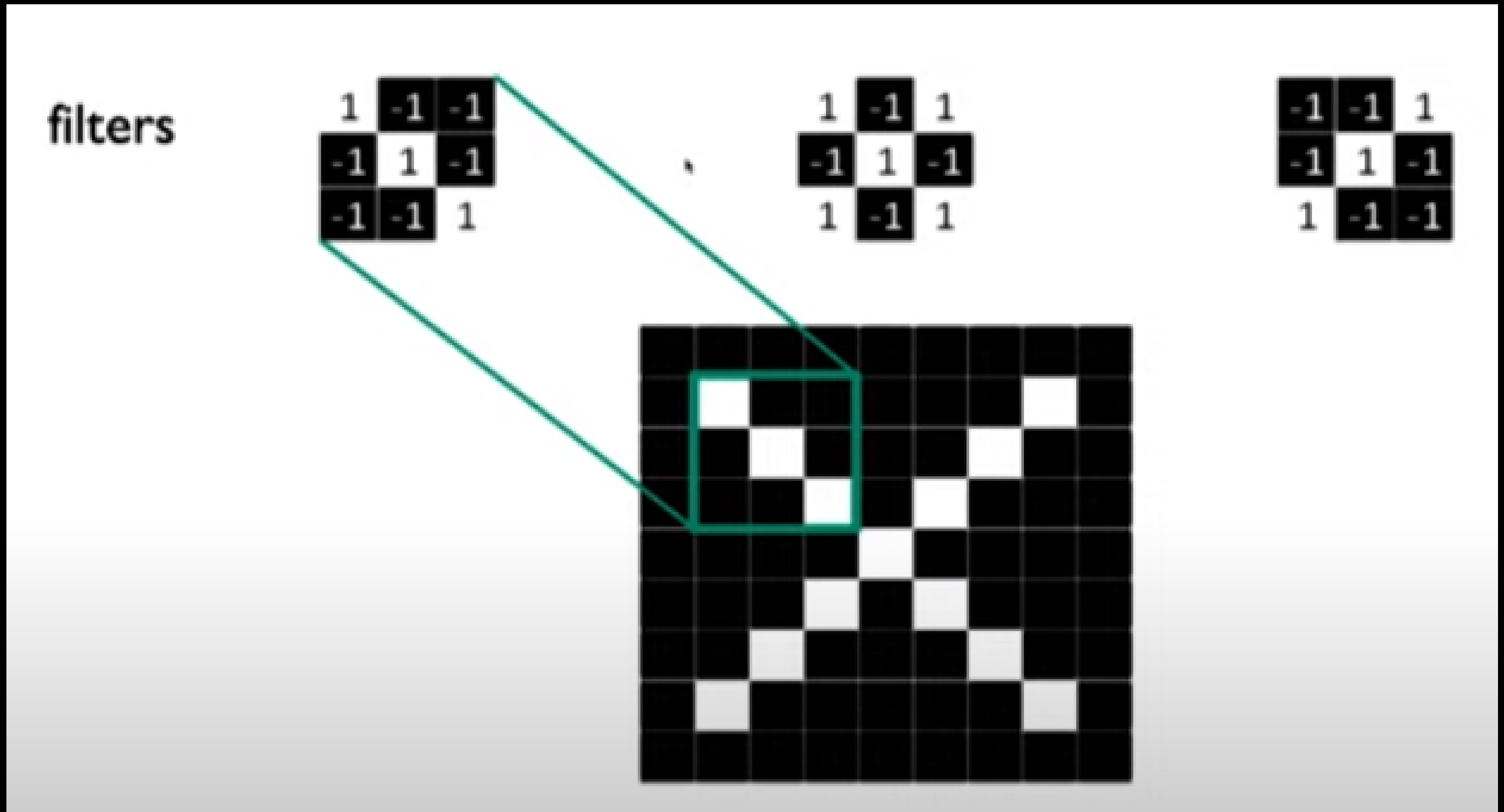
- Filter of size 4x4: 16 different weights
- Apply this same filter to a 4x4 patches in input
- Shift by 2 pixels for next patch

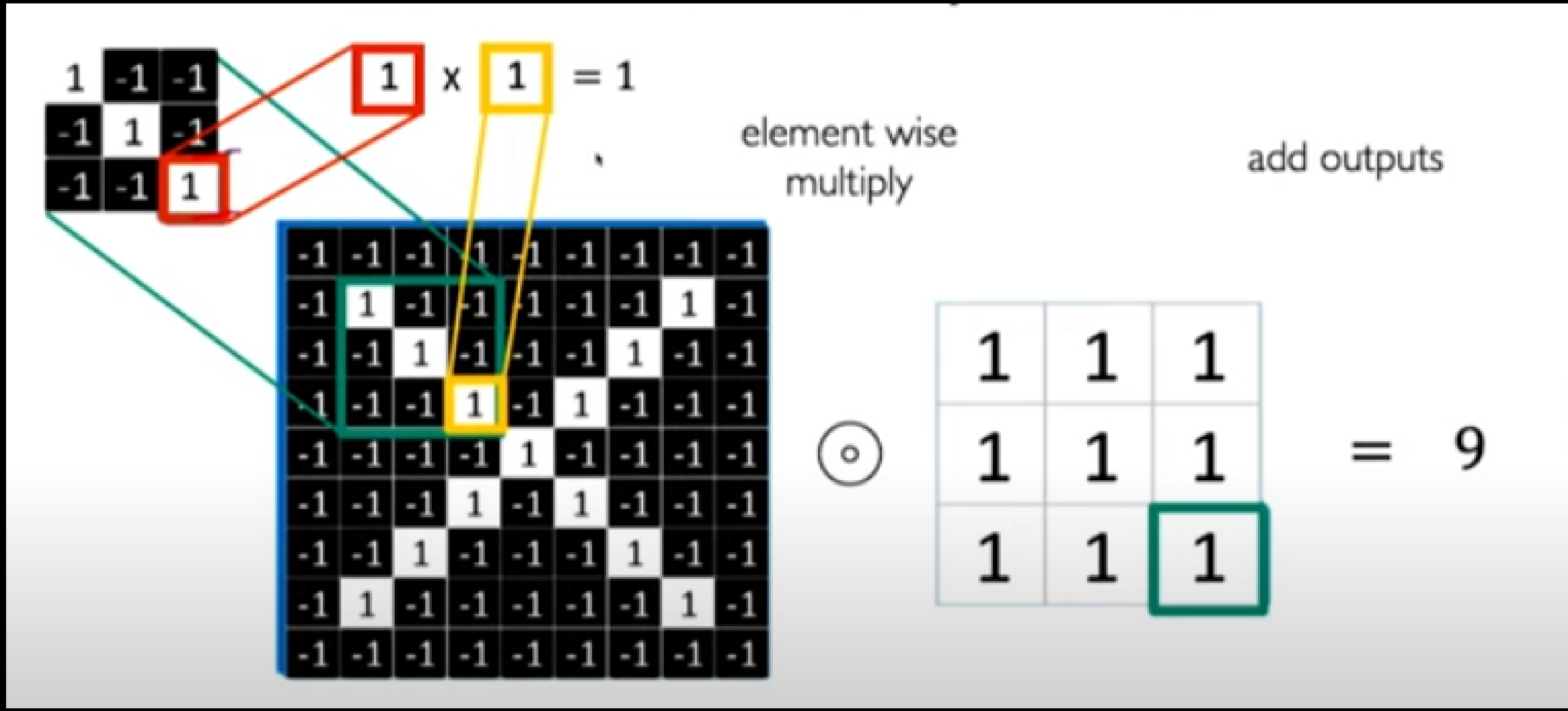
This "patchy" operation is convolution

- Apply a set of weights (a filter) to extract local features
- Use multiple filters to extract different features
- Spatially share parameters of each filter

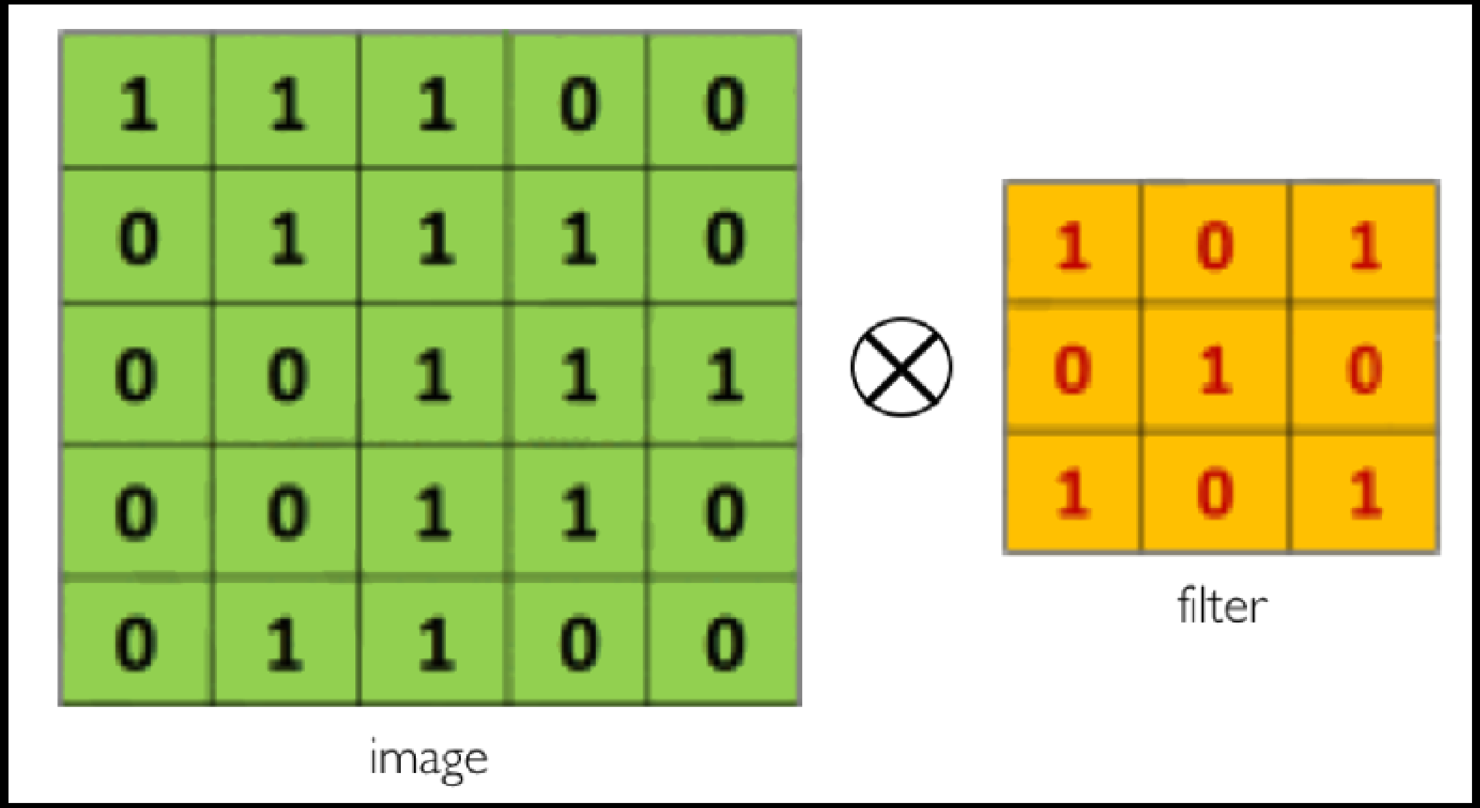
Convolutional operation

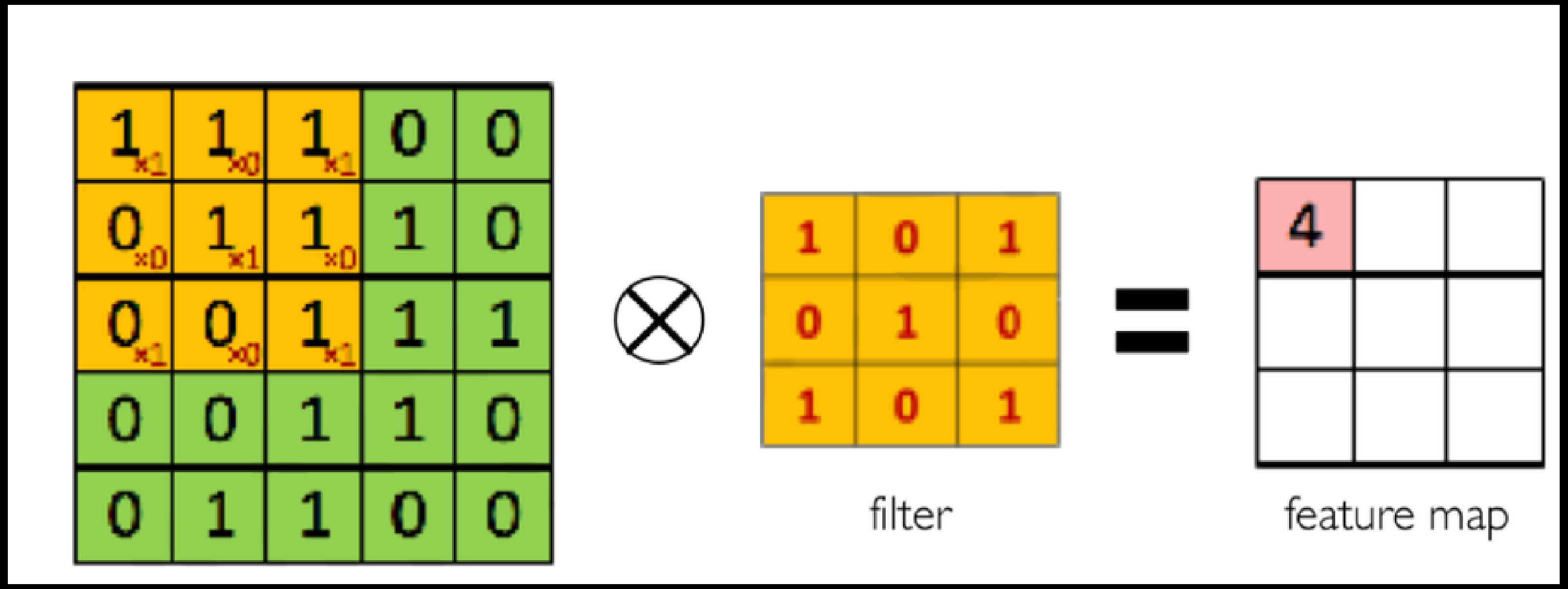


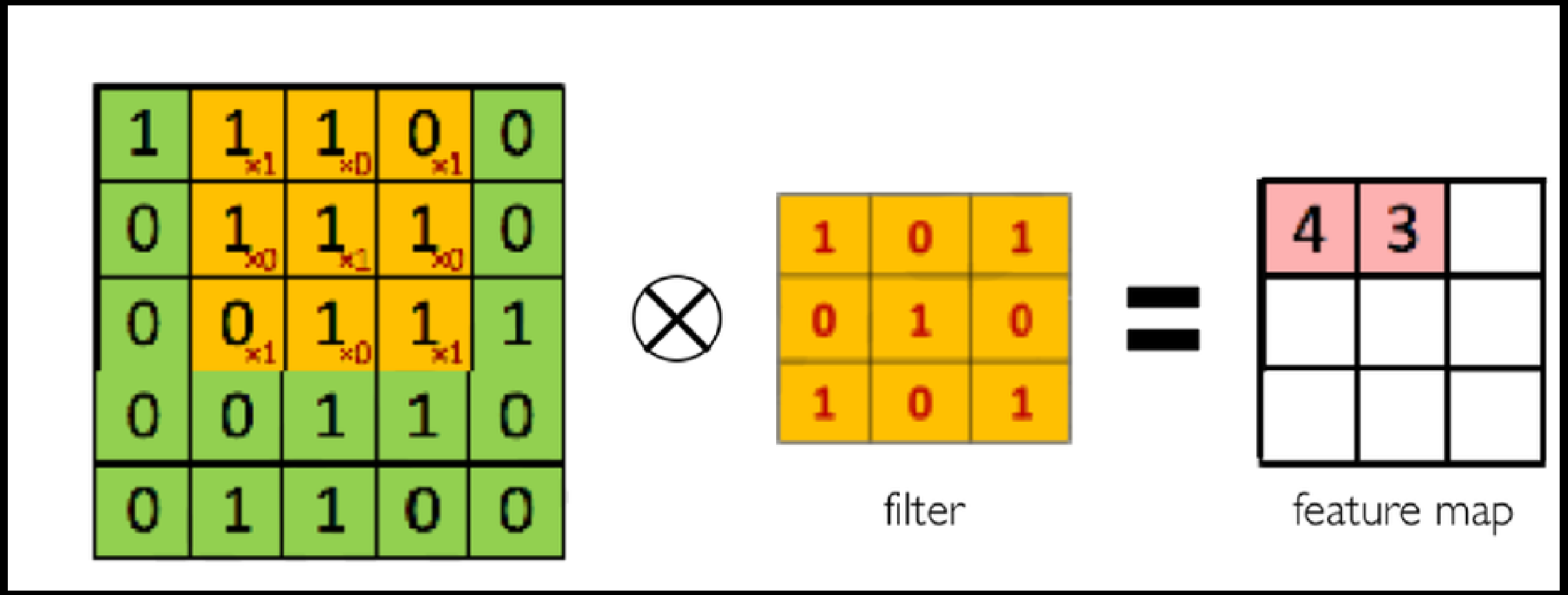




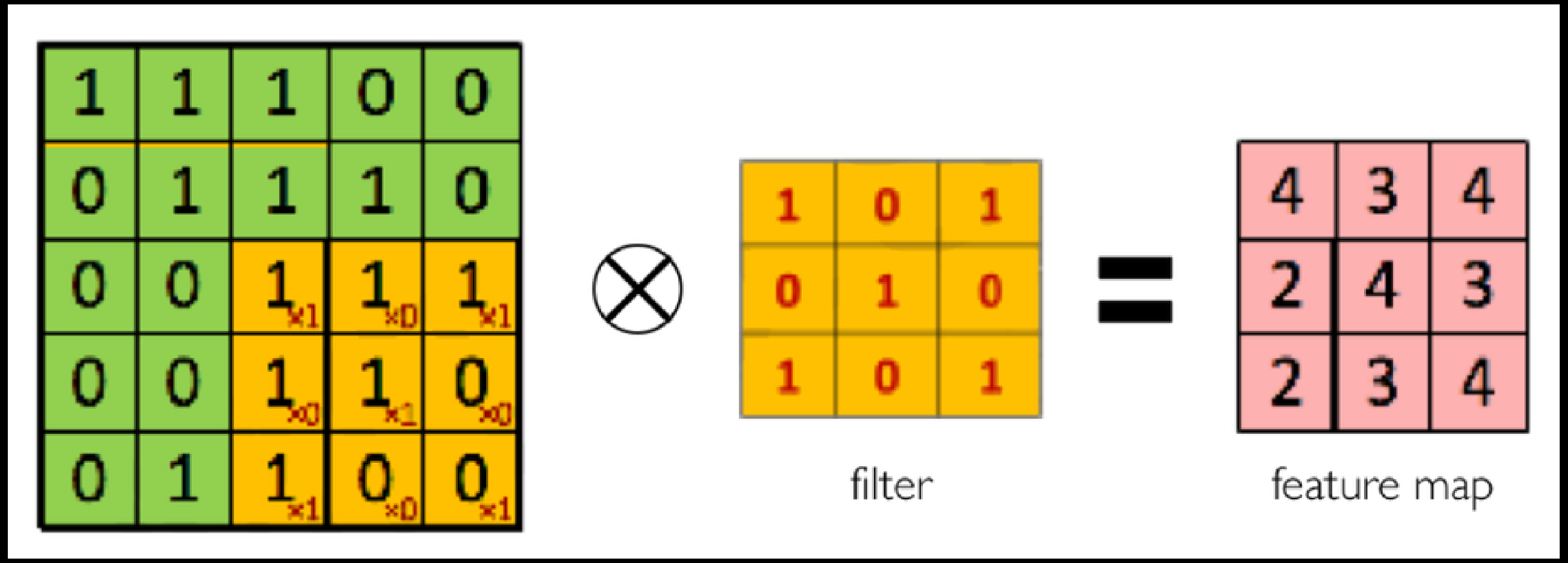
Convolutional operation







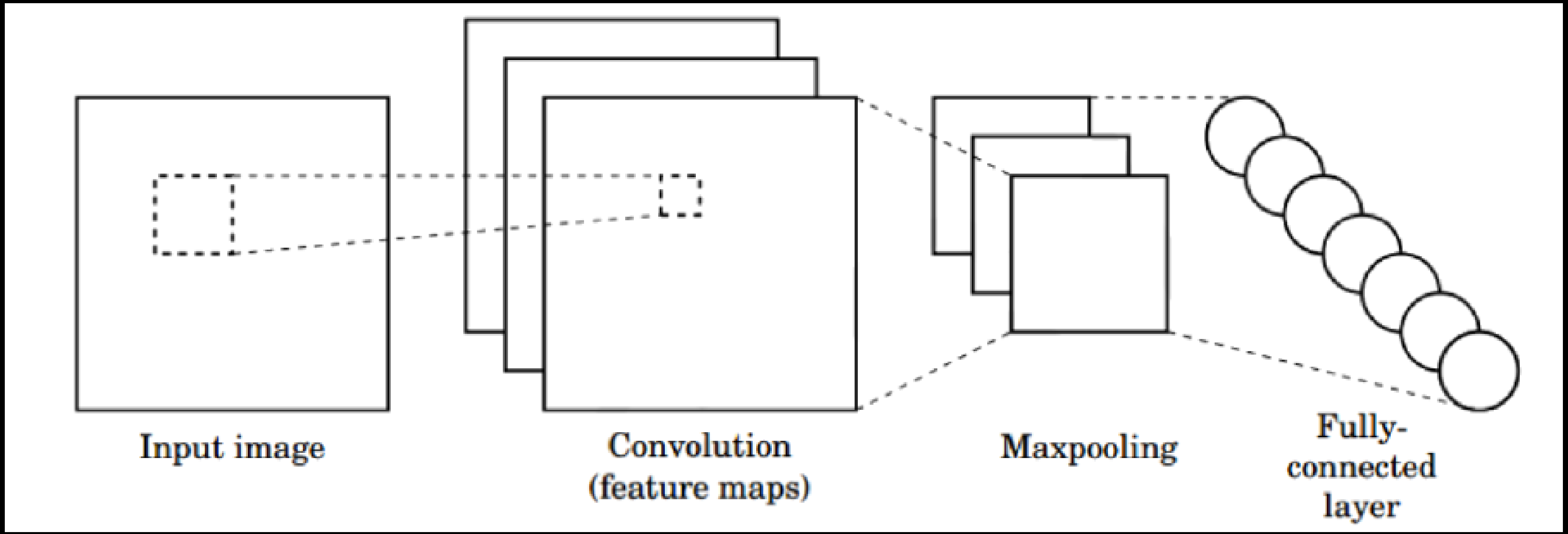
...



Original Sharpen Edge Detect "Strong" Edge Detect

Convolutional neural networks (CNN)

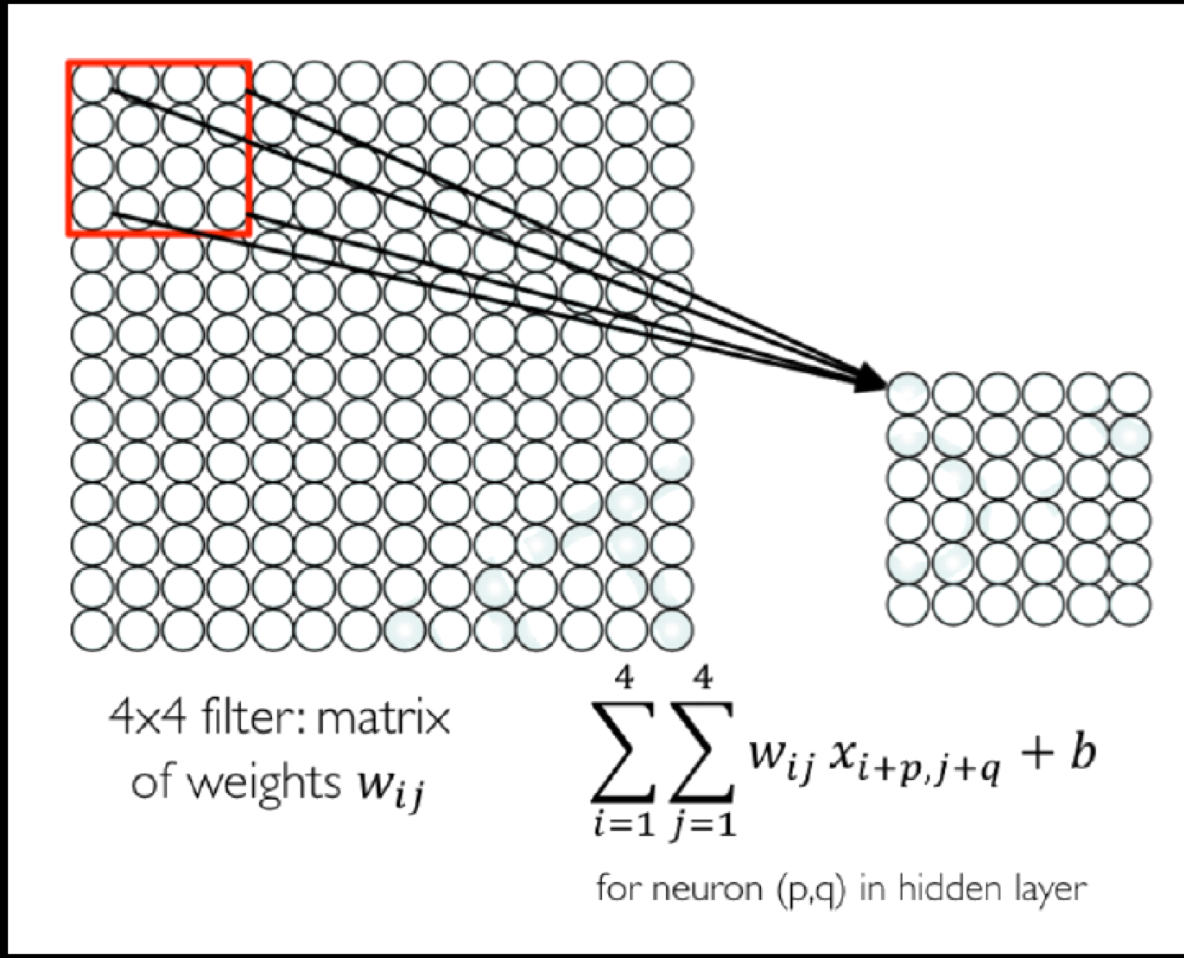
- for classification

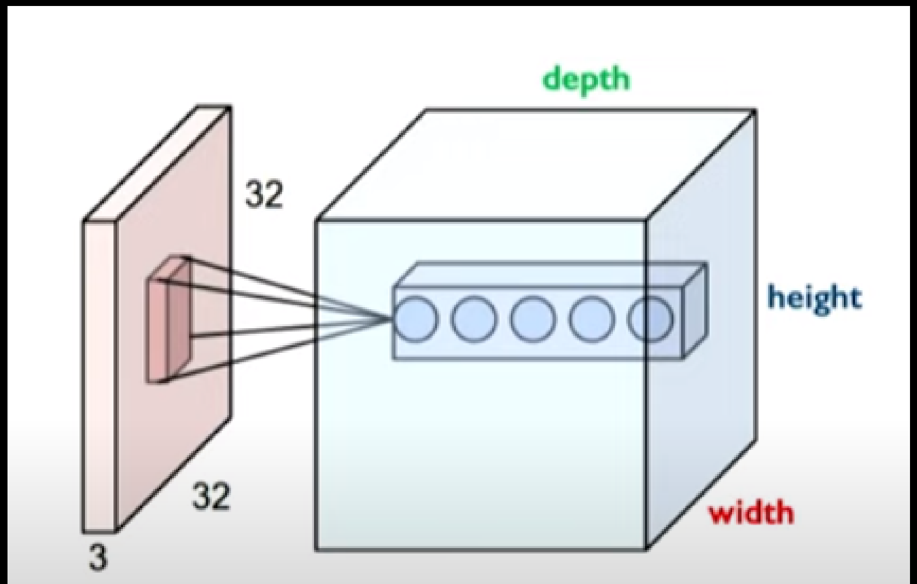


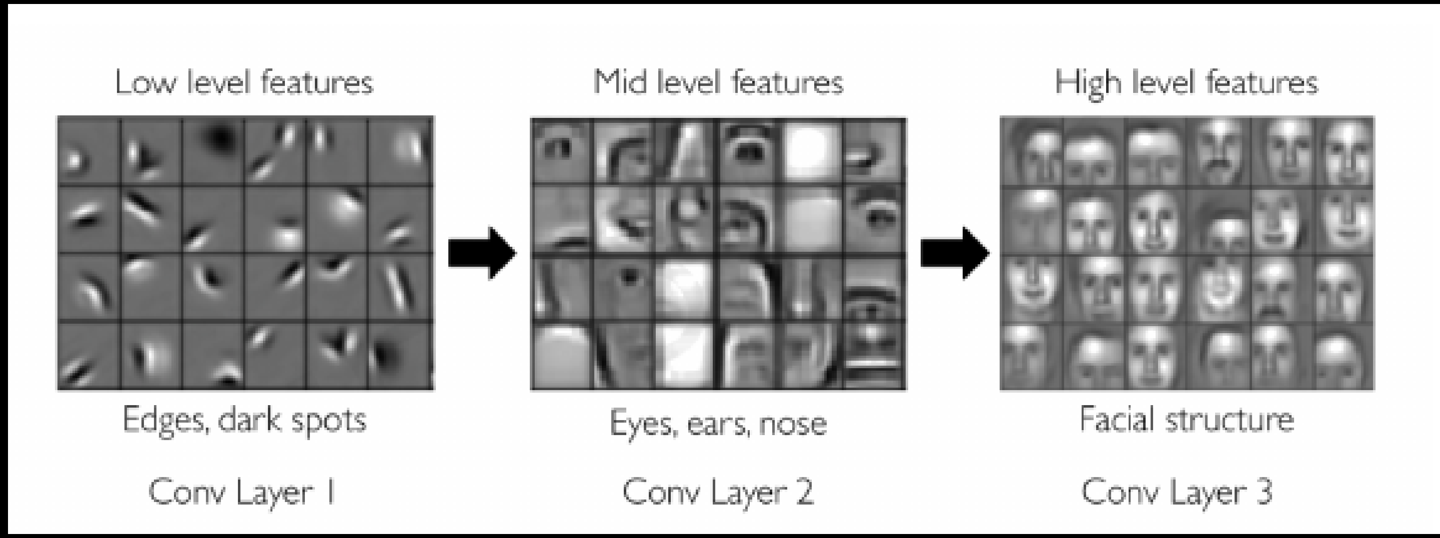
- Convolution: apply filters to generate feature maps
- Non-linearity: often ReLU
- Pooling: Downsampling operation on each feature map

Train model with image data.
Learn weights of filters in convolutional layers.

Convolutional layer



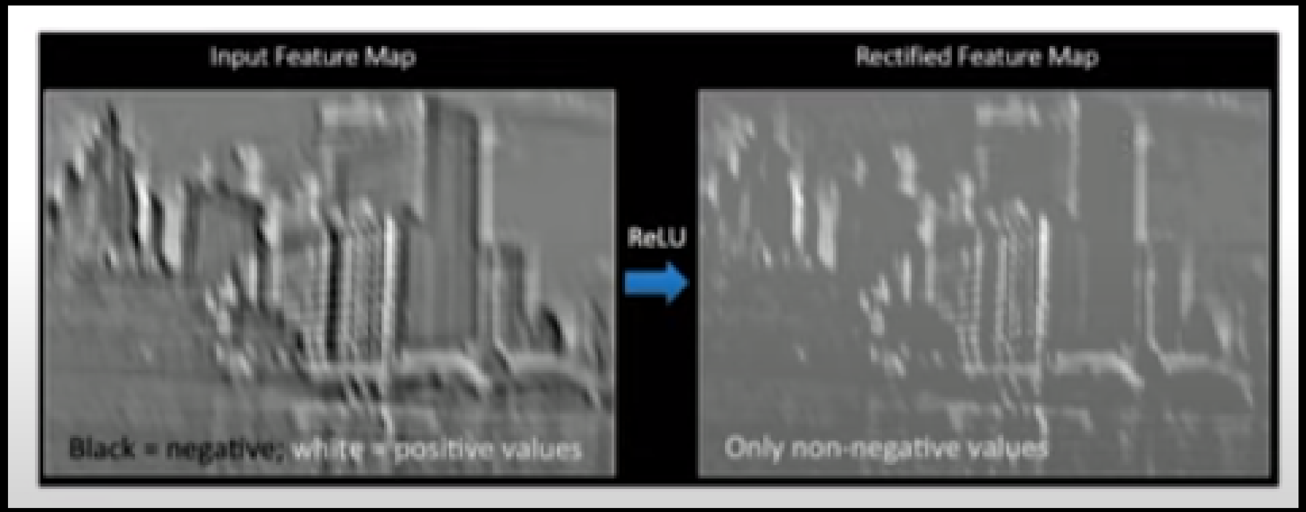


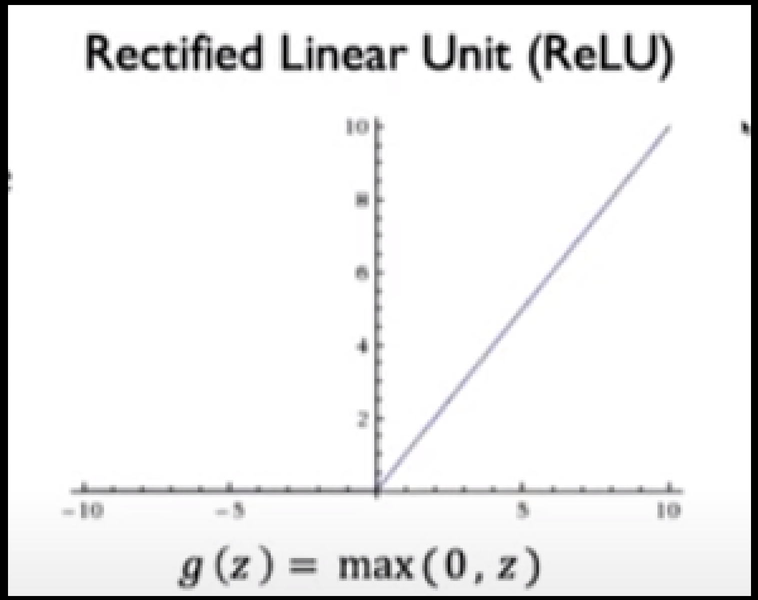


1. applying a window of weights
2. computng linear combinations

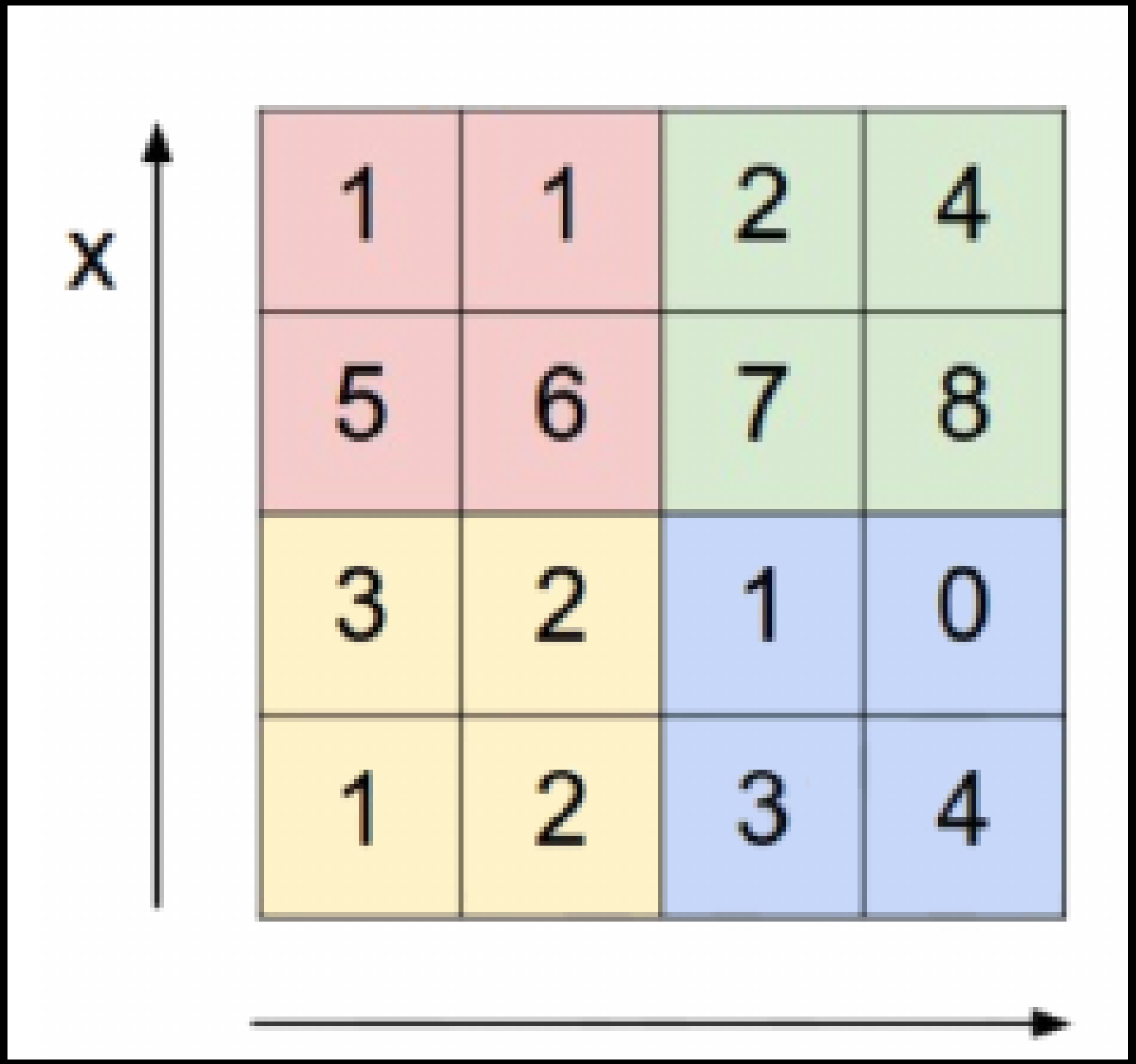
3. activating with non-linear function

For a neuron in hidden layer
- Take inputs from patch
- Compute weighted sum
- apply bias

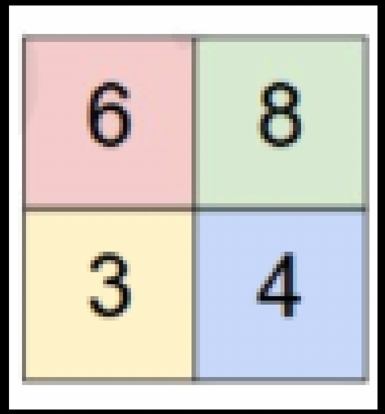




Pooling

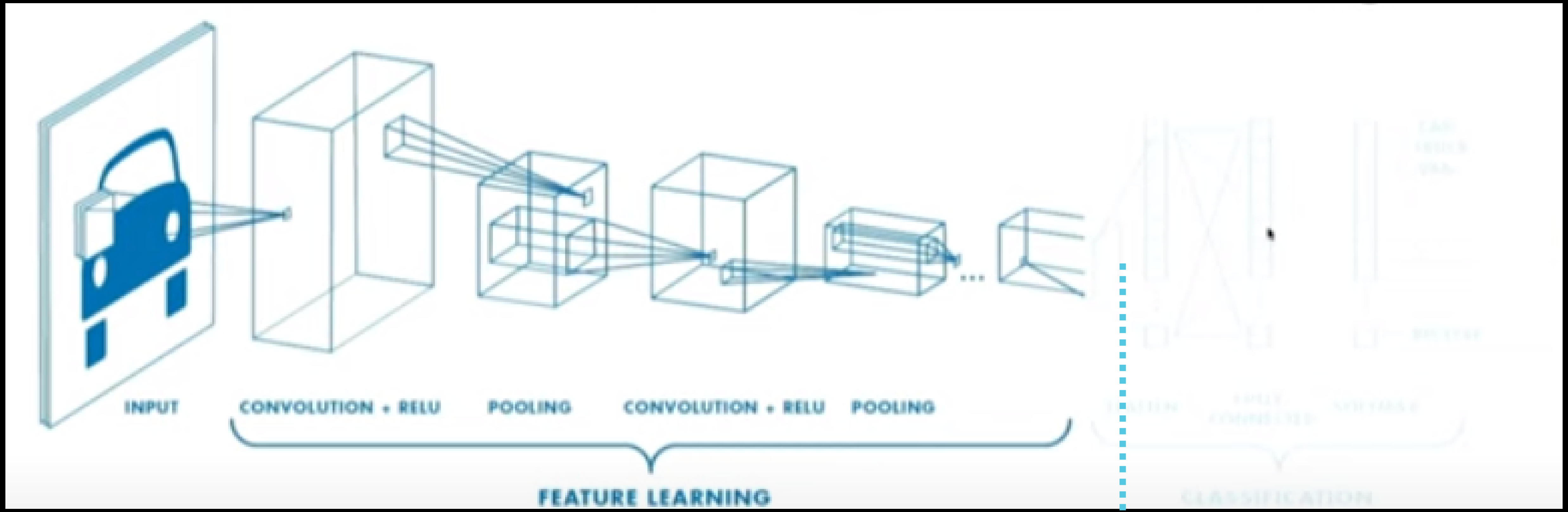


max pool with 2x2 filters
and stride 2



- Reduce dimensionality
- Spatial invariance

CNN: classification example

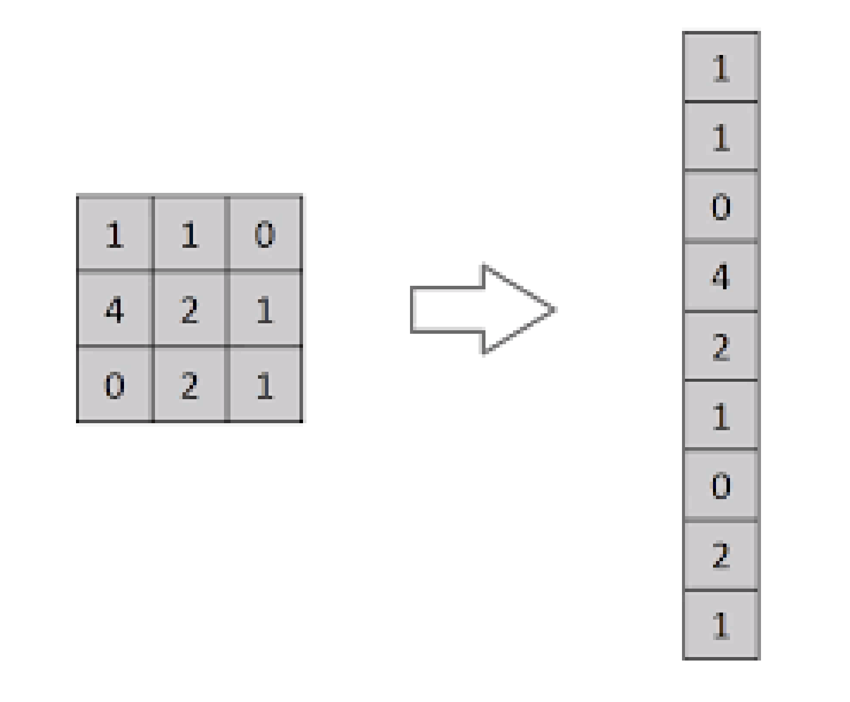


- Learns features in input image throught convolution
- Introduce non-linearity (activation func)
- Reduce dimensionality and preserve spacial invariance (pooling)

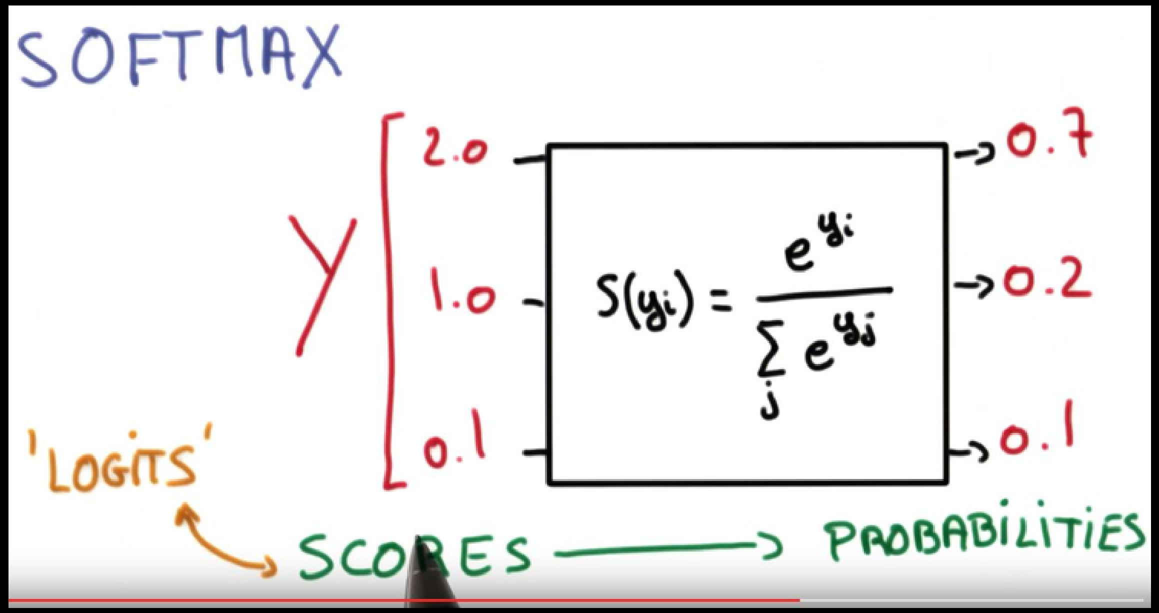
- at this point the last outpus is the i-th feature map

CNN: classification example

- CONV and POOL layers putput high-level features input
- Fully connected layer uses these features to classifying input image
- Express output as probability (image to certain class)



$$\mathrm{softmax}(y_i) = \frac{e^{y_i}}{\sum_j e^{y_j}}$$



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

HANDS-ON LAB

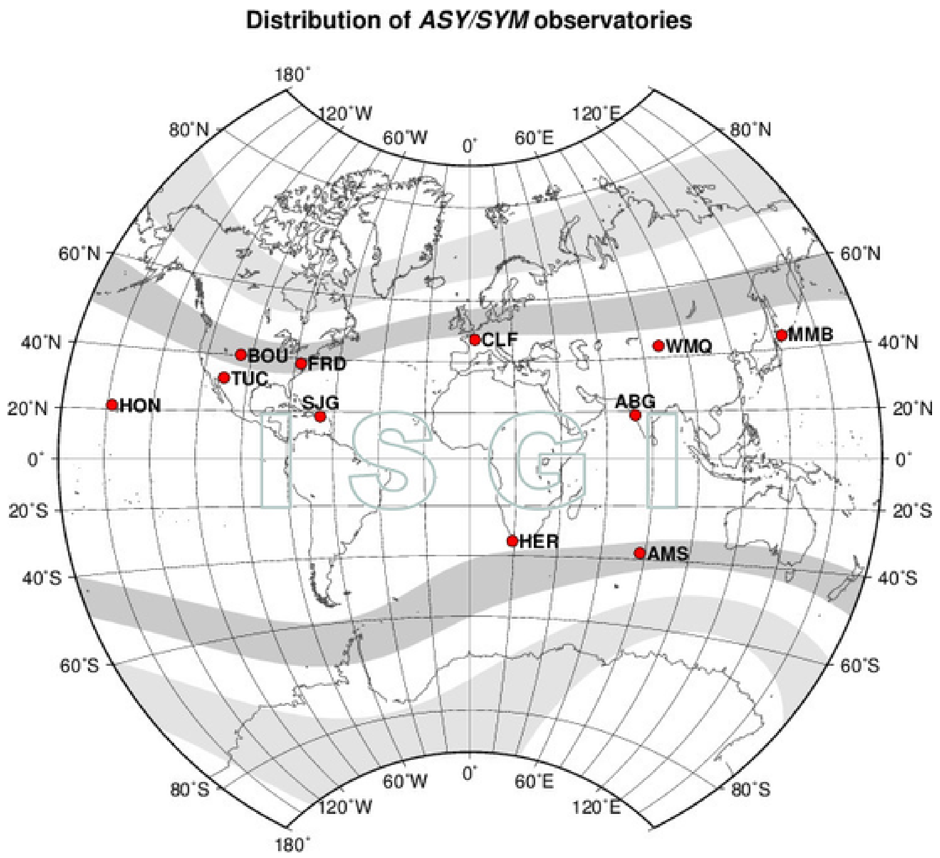
TP - 3

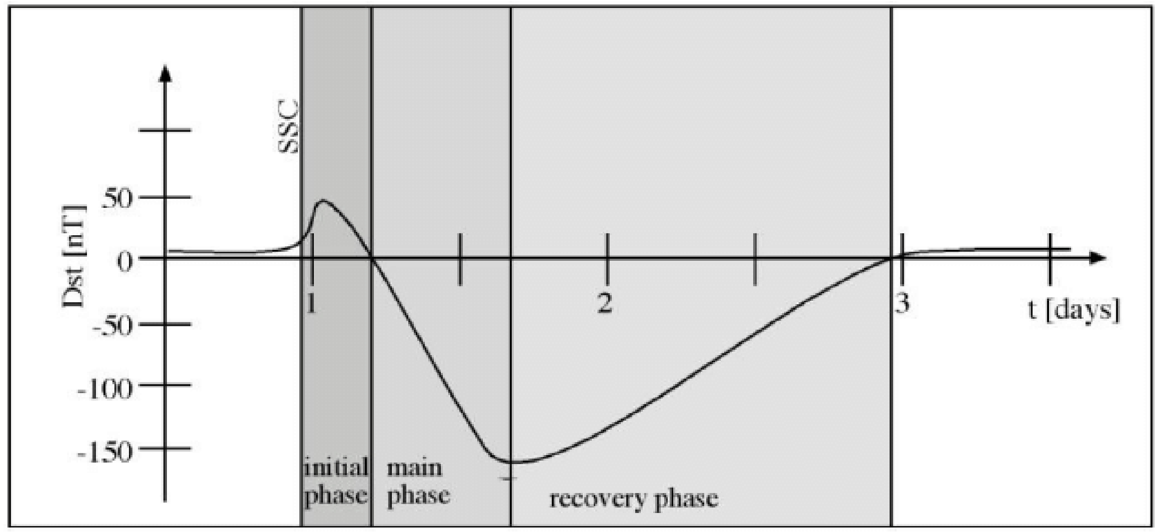
Sym-H forecasting

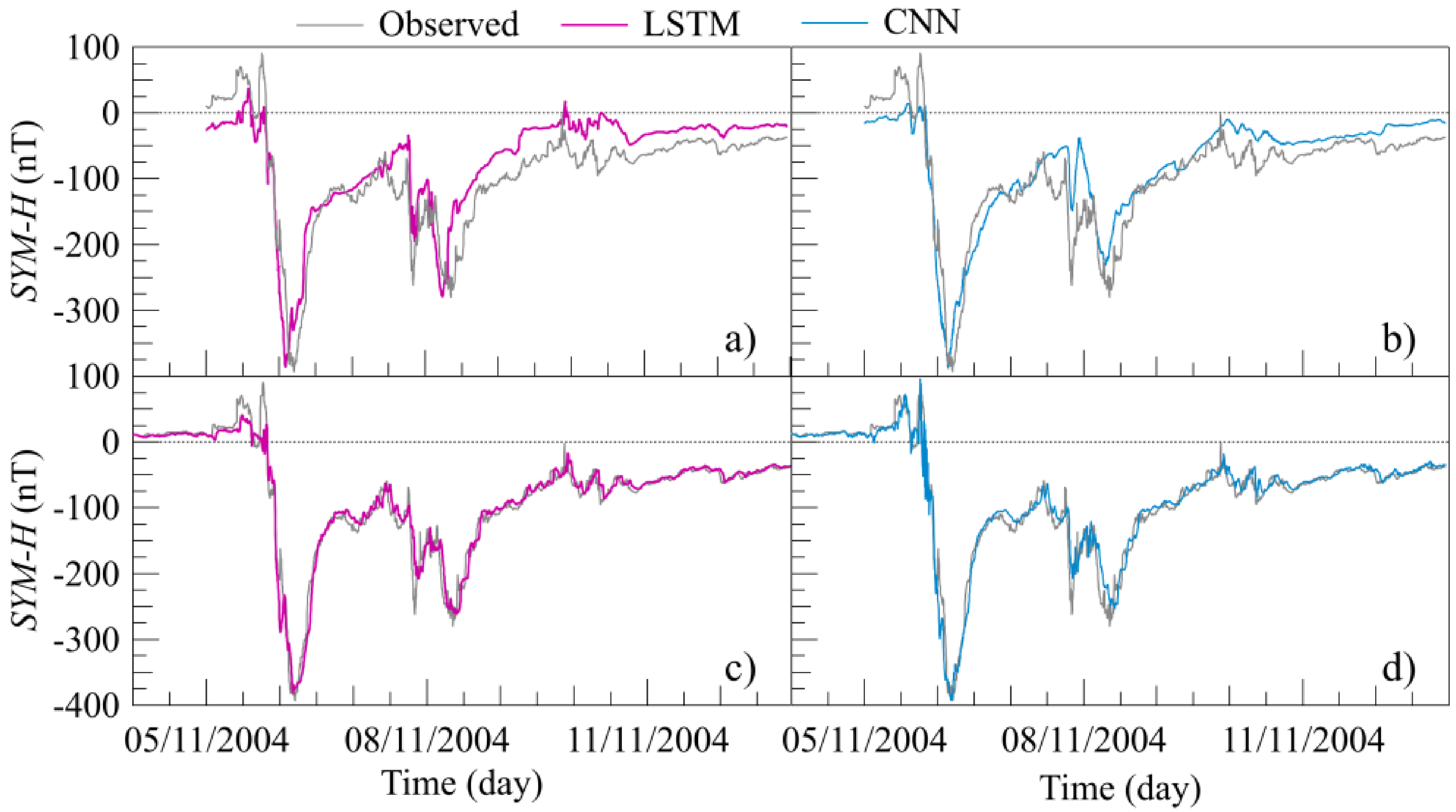
TSWC, 2022

From science to data science

http://isgi.unistra.fr/indices_asy.php







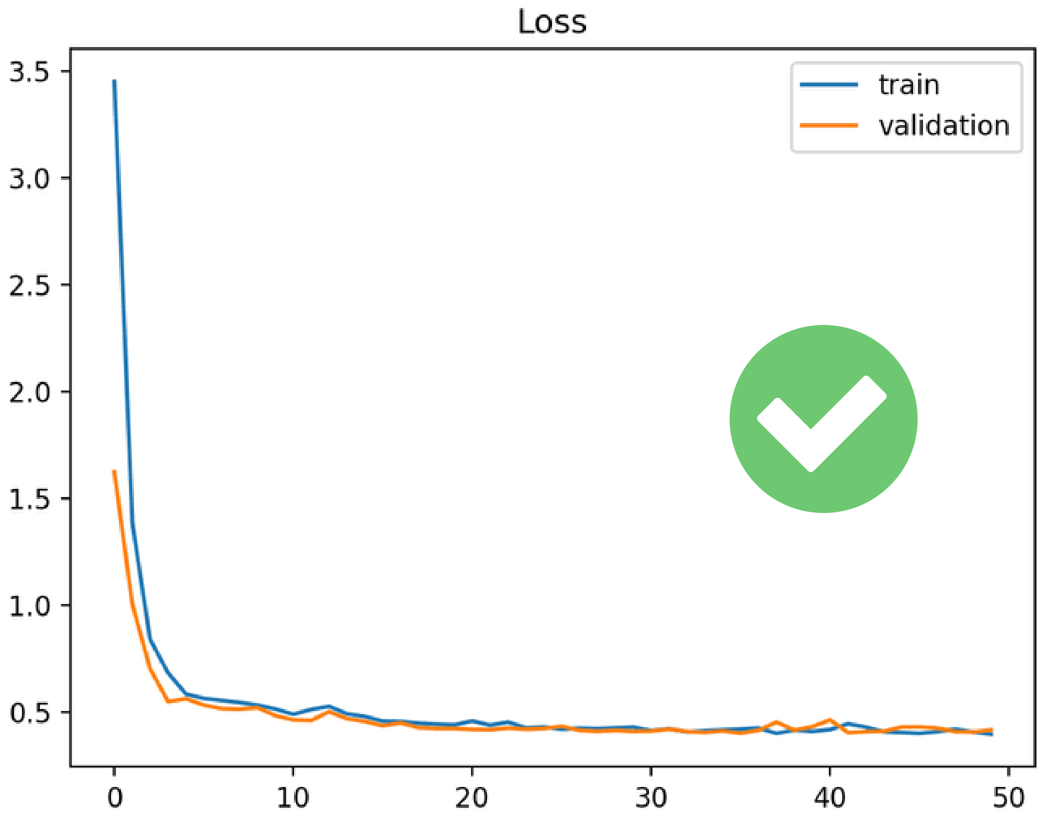
To describe the geomagnetic disturbances at mid-latitudes in terms of longitudinally asymmetric (ASY) and symmetric (SYM) disturbances for both H and D components respectively parallel and perpendicular to the dipole axis.

SYM-H is essentially the same as the Dst index with a different time resolution.

Siciliano et al, 2022

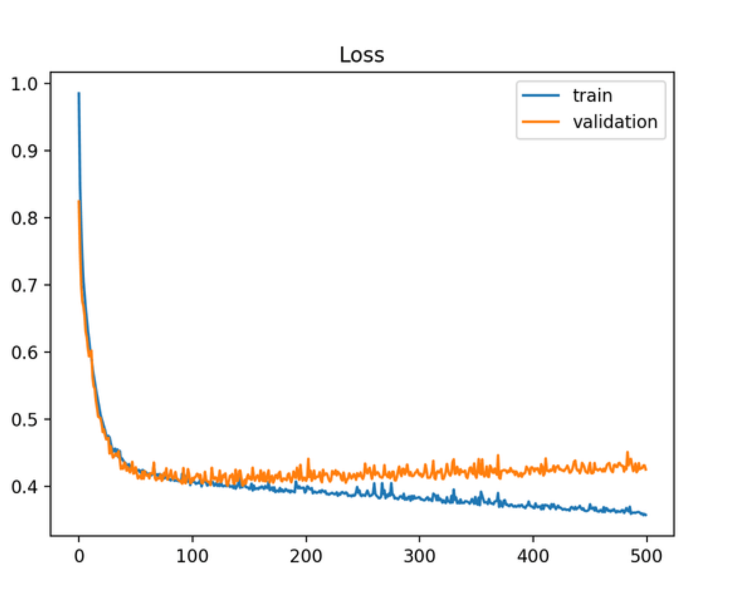
Diagnose

- The learning process

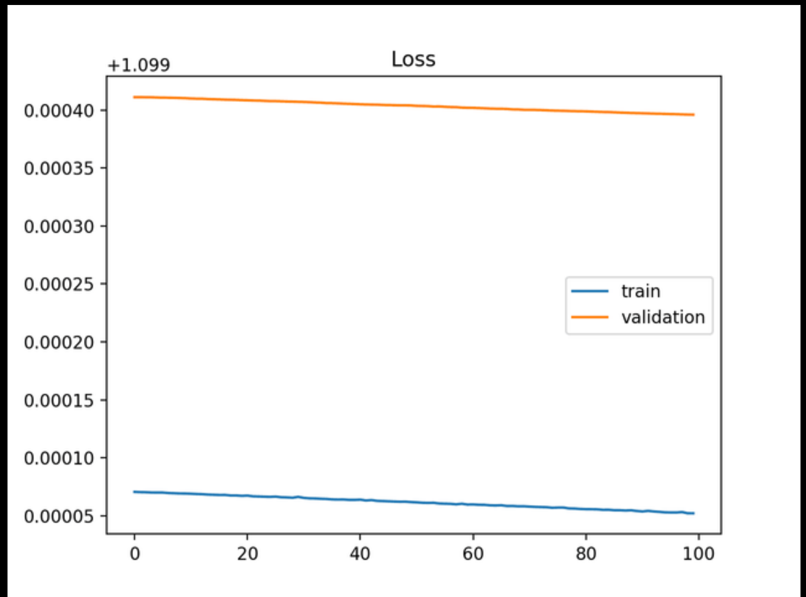


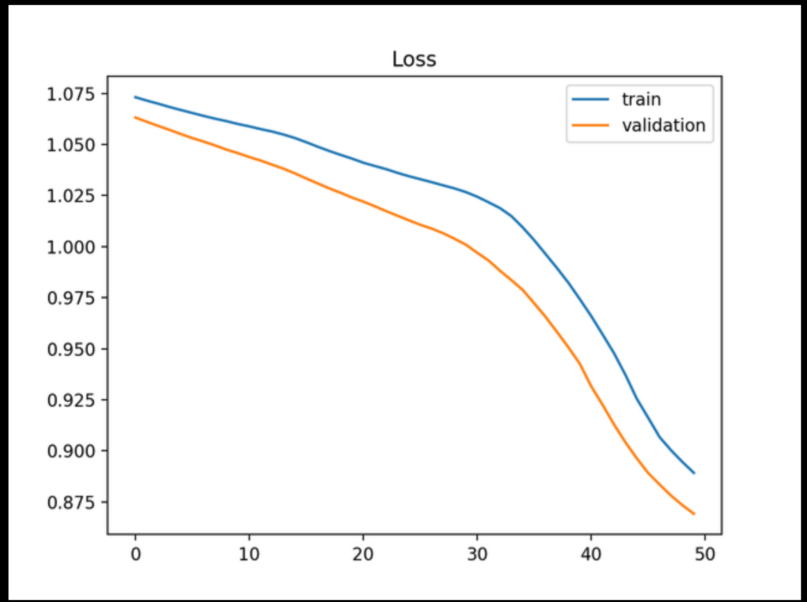
reallity

overfitting



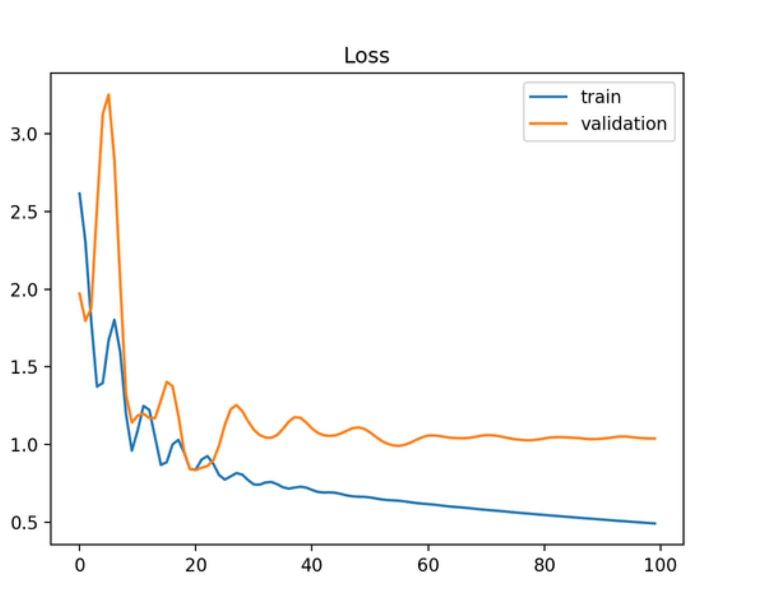
Model does not have a suitable capacity for the complexity of the dataset (underfit)





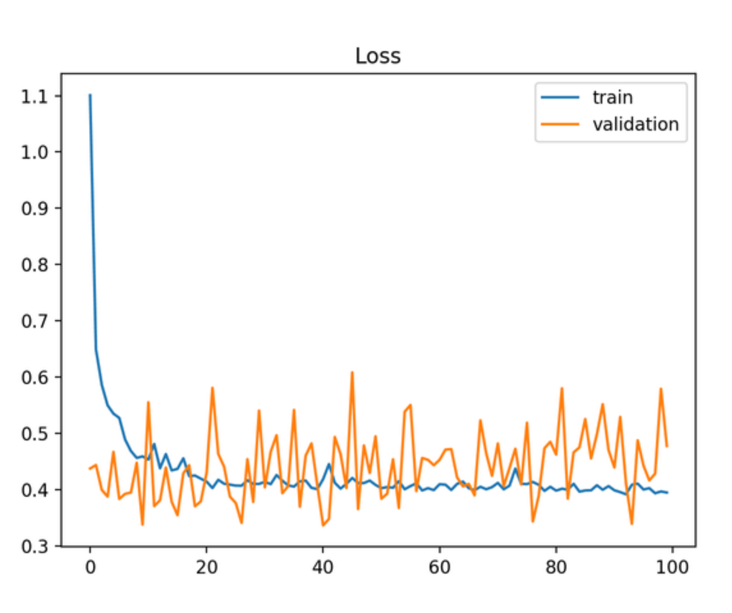
Model that requires further training (underfit)

unrepresentative training dataset

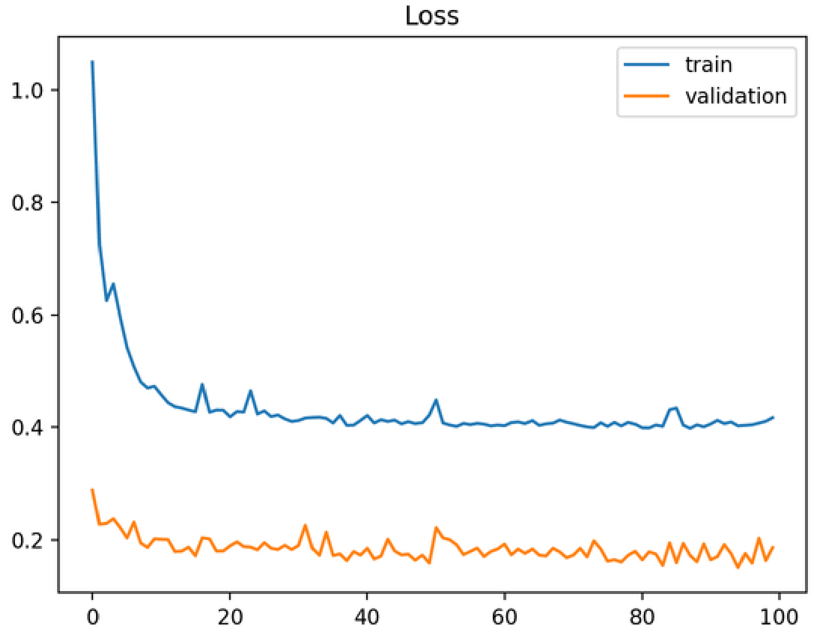


Dataset << relative to the validation dataset

unrepresentative validation dataset



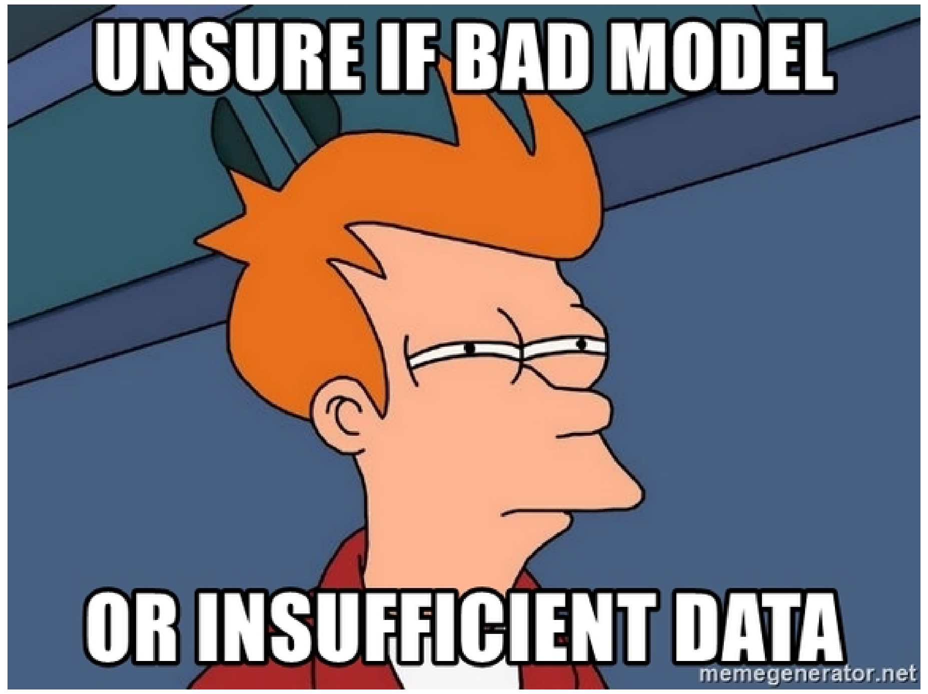
validation dataset does not provide sufficient information to evaluate the ability of the model to generalize

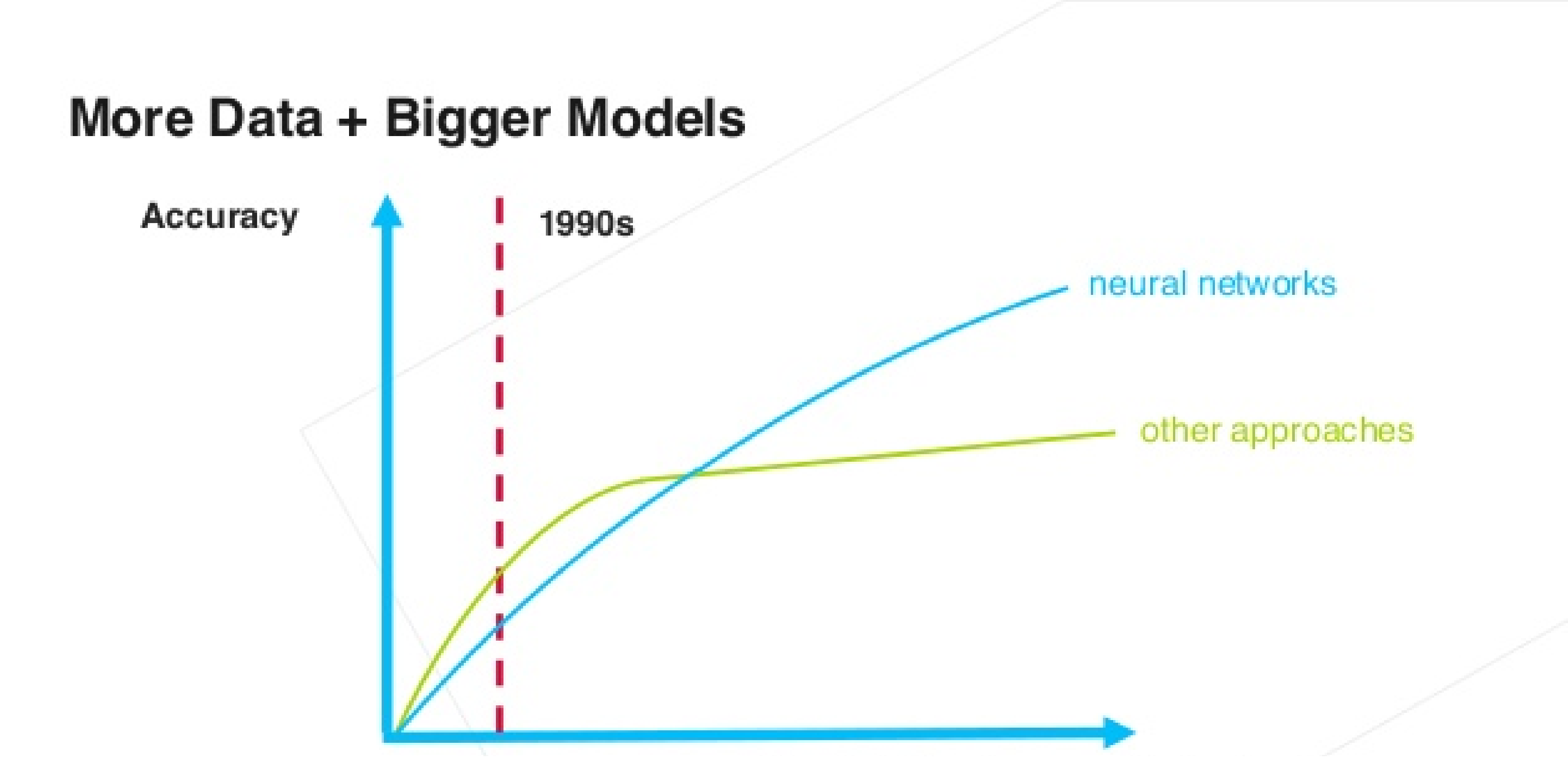


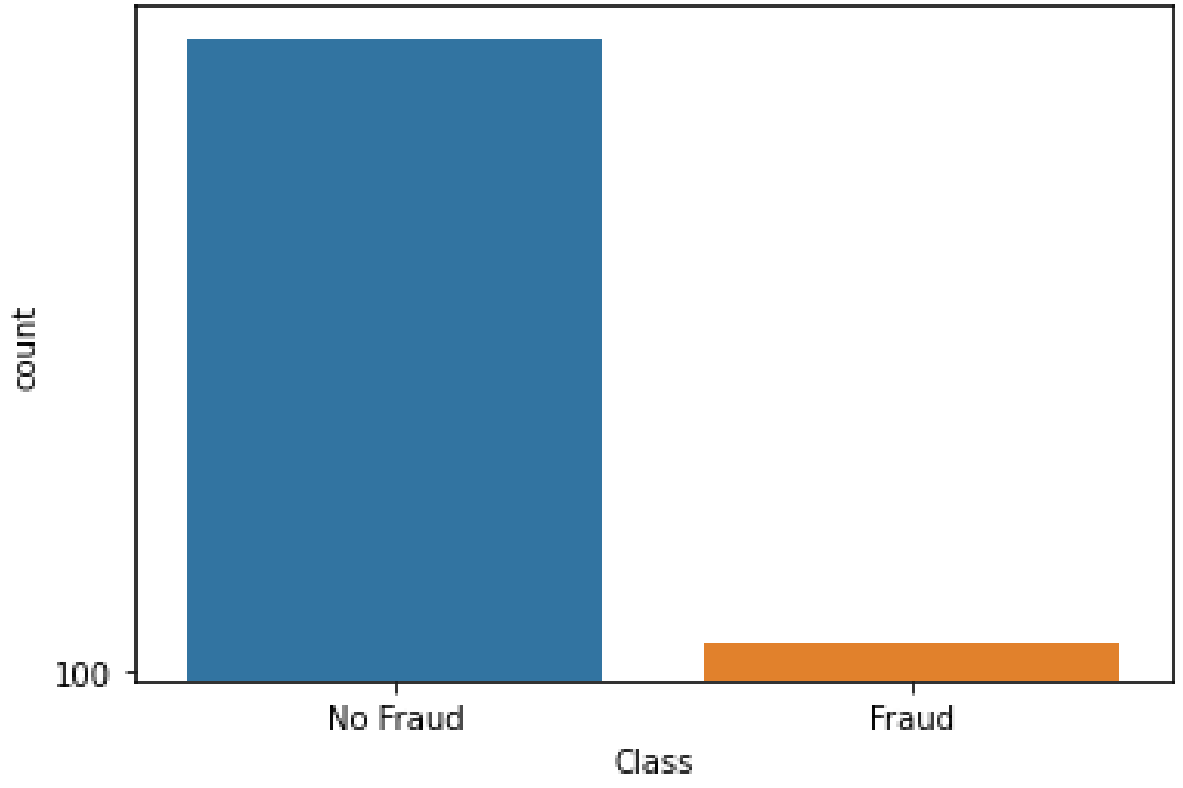
TSWC, 2022

About the dataset

- Amount of data and data balancing





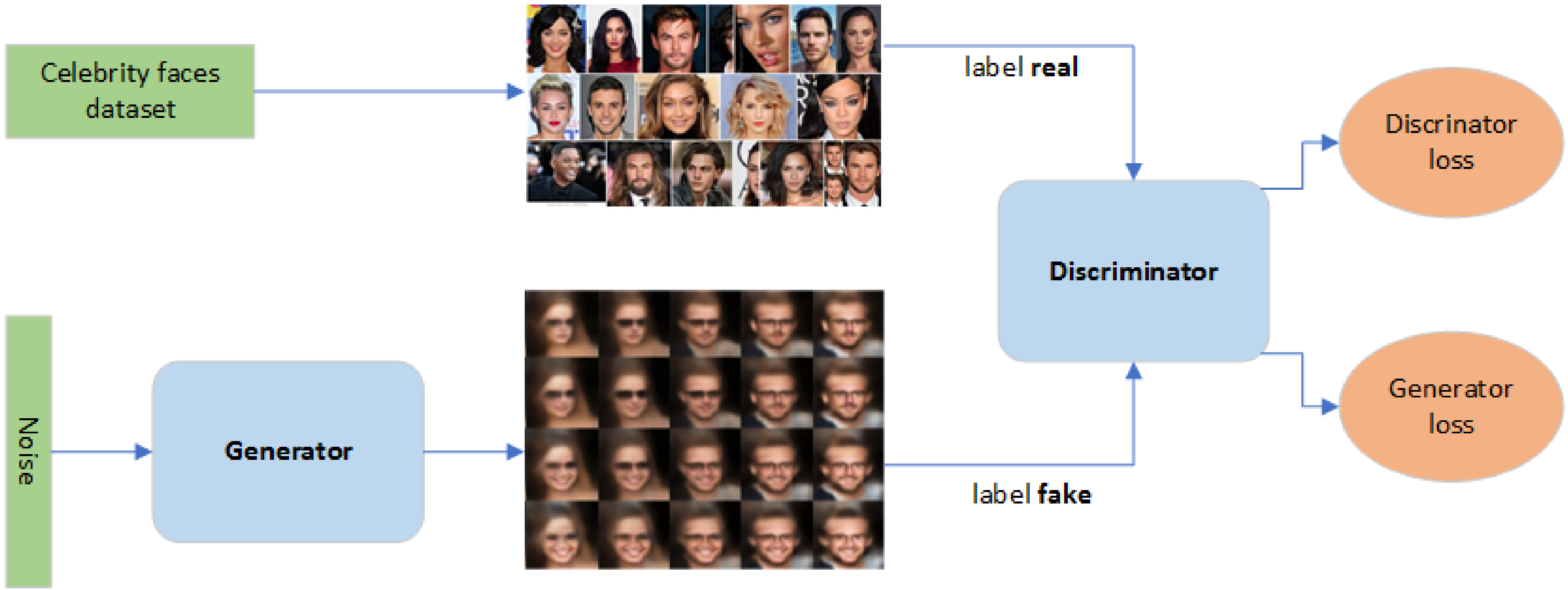


re sample (?)
generate more data (?)

About the dataset

Method Name	Category	Software	Reference
Mean impute (mean)	Mean		Little and Rubin (1987)
Expectation-Maximization (EM)	EM		Dempster et al. (1977)
EM with Mixture of Gaussians and Multinomials	EM		Ghahramani and Jordan (1994)
EM with Bootstrapping	EM	Amelia II	Honaker et al. (2011)
K-Nearest Neighbors (knn)	K-NN	impute	Troyanskaya et al. (2001)
Sequential K-Nearest Neighbors	K-NN		Kim et al. (2004)
Iterative K-Nearest Neighbors	K-NN		Caruana (2001); Brás and Menezes (2007)
Support Vector Regression	SVR		Wang et al. (2006)
Predictive-Mean Matching (pmm)	LS	MICE	Buuren and Groothuis-Oudshoorn (2011)
Least Squares	LS		Bø et al. (2004)
Sequential Regression Multivariate Imputation	LS		Raghunathan et al. (2001)
Local-Least Squares	LS		Kim et al. (2005)
Sequential Local-Least Squares	LS		Zhang et al. (2008)
Iterative Local-Least Squares	LS		Cai et al. (2006)
Sequential Regression Trees	Tree	MICE	Burgette and Reiter (2010)
Sequential Random Forest	Tree	missForest	Stekhoven and Bühlmann (2012)
Singular Value Decomposition	SVD		Troyanskaya et al. (2001)
Bayesian Principal Component Analysis	SVD	pcaMethods	Oba et al. (2003); Mohamed et al. (2009)
Factor Analysis Model for Mixed Data	FA		Khan et al. (2010)

Bertimas et.al., From Predictive Methods to Missing Data Imputation: An Optimization Approach (2018)

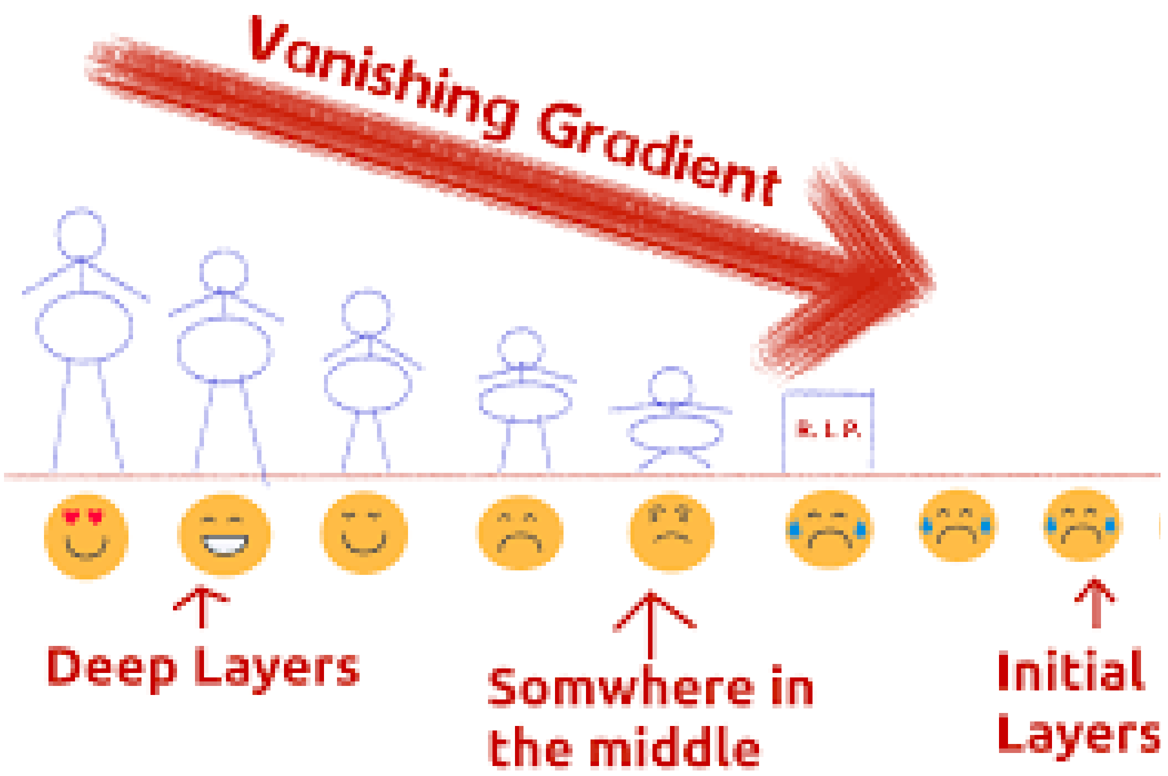


Goodfellow et.al. Generative Adversarial Networks (2014)

TSWC, 2022

Vanishing gradient problem

- activation function, architecture, weight initialization, loss optimization algorithm, learning rate, …



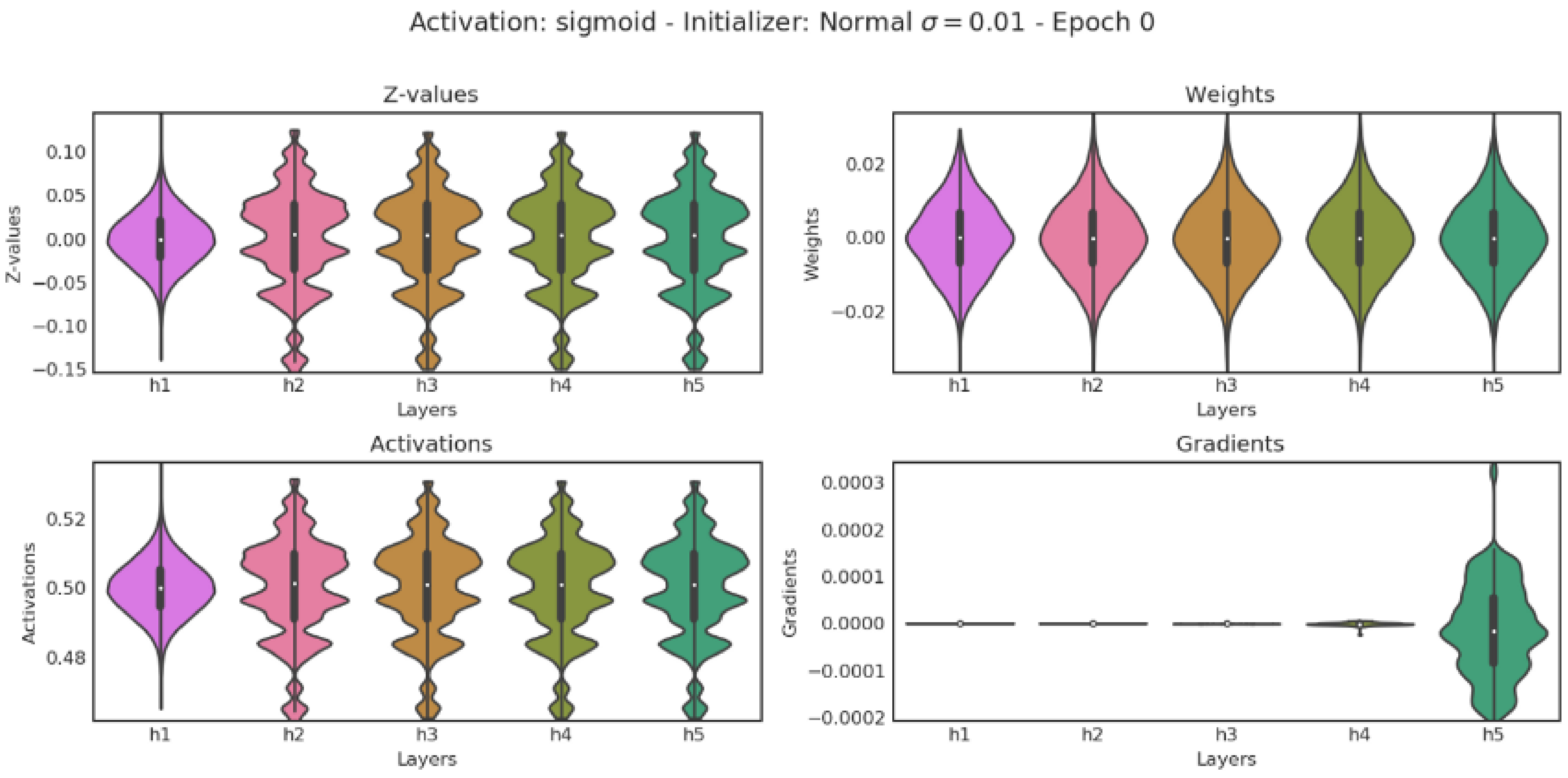


Figure 1. BLOCK model using sigmoid and naive initialization — don't try this at home!

https://towardsdatascience.com/hyper-parameters-in-action-part-ii-weight-initializers-35aee1a28404

Hyperparameters optimization

- Choose a set of optimal hyperparameters for a learning algorithm (maximizes the model performance)
- Hyperparameters are set before the learning process (#neurons, #cells, loss optimization algorithms, etc)

GRID SEARCH

1 —Identify the model's hyperparameters to optimizest.

2 — Asses error score for each combination in the hyperparameter grid.

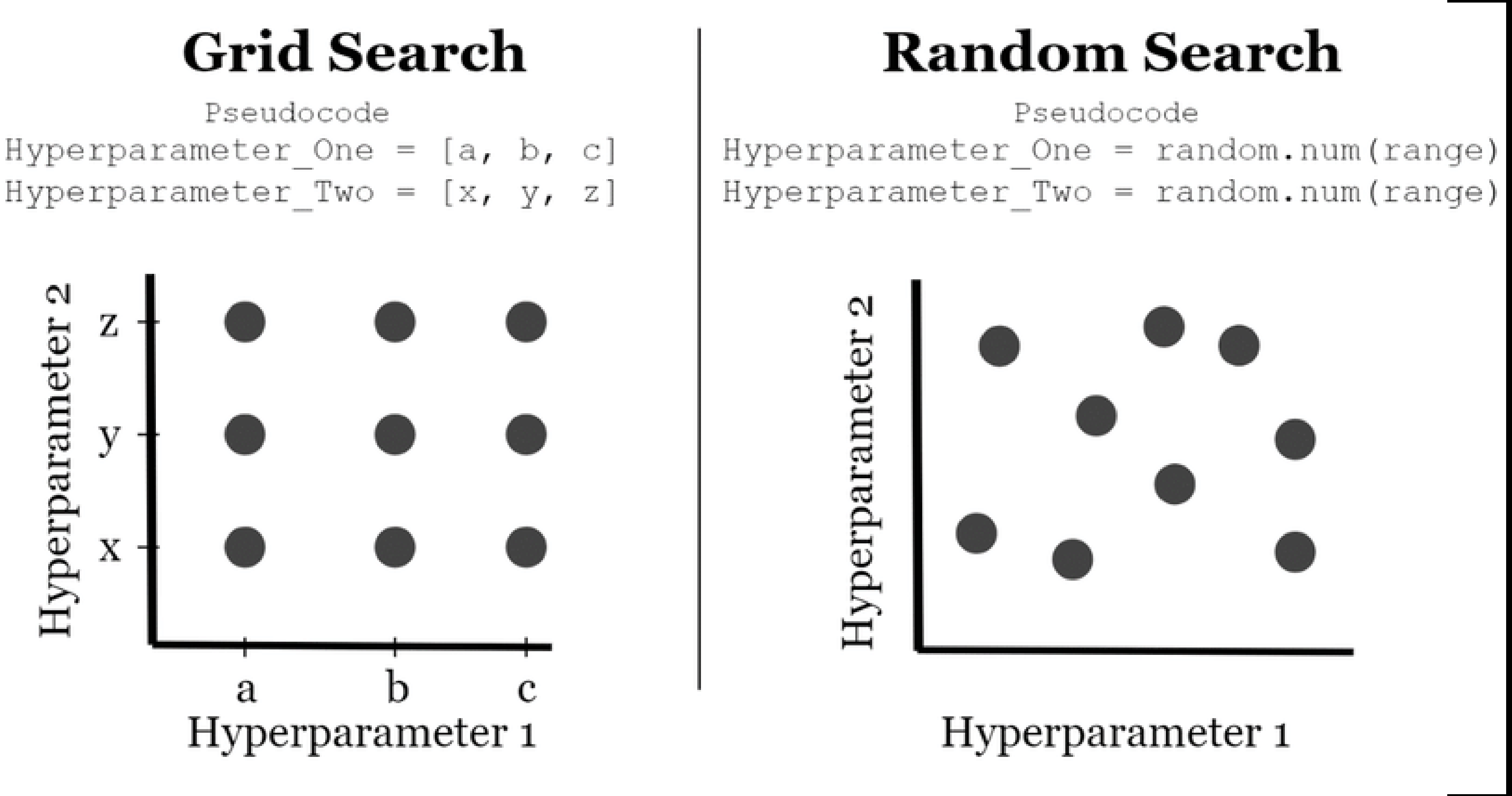
3 — Select the hyperparameter combination with the best error metric.

TSWC, 2022

Hyperparameters optimization

```
# learning_rate choices
 learning_rates = [ 0.1, 0.2, 0.3, 0.4, 0.5,
                   0.01, 0.02, 0.03, 0.04, 0.05 ]
 # iterations choices
 iterations = [ 100, 200, 300, 400, 500 ]

 parameters = []
 for i in learning_rates :
     for j in iterations :
         parameters.append( ( i, j ) )

 print("Available combinations : ",  parameters )

 # Applying linear searching in list of available combination
 # to achieved maximum accuracy on CV set

 for k in range( len( parameters ) ) :
   #  model = METHOD(..., learning_rate = parameters[k][0],iterations = parameters[k][1] )
   # ...
```

Hyperparameters optimization

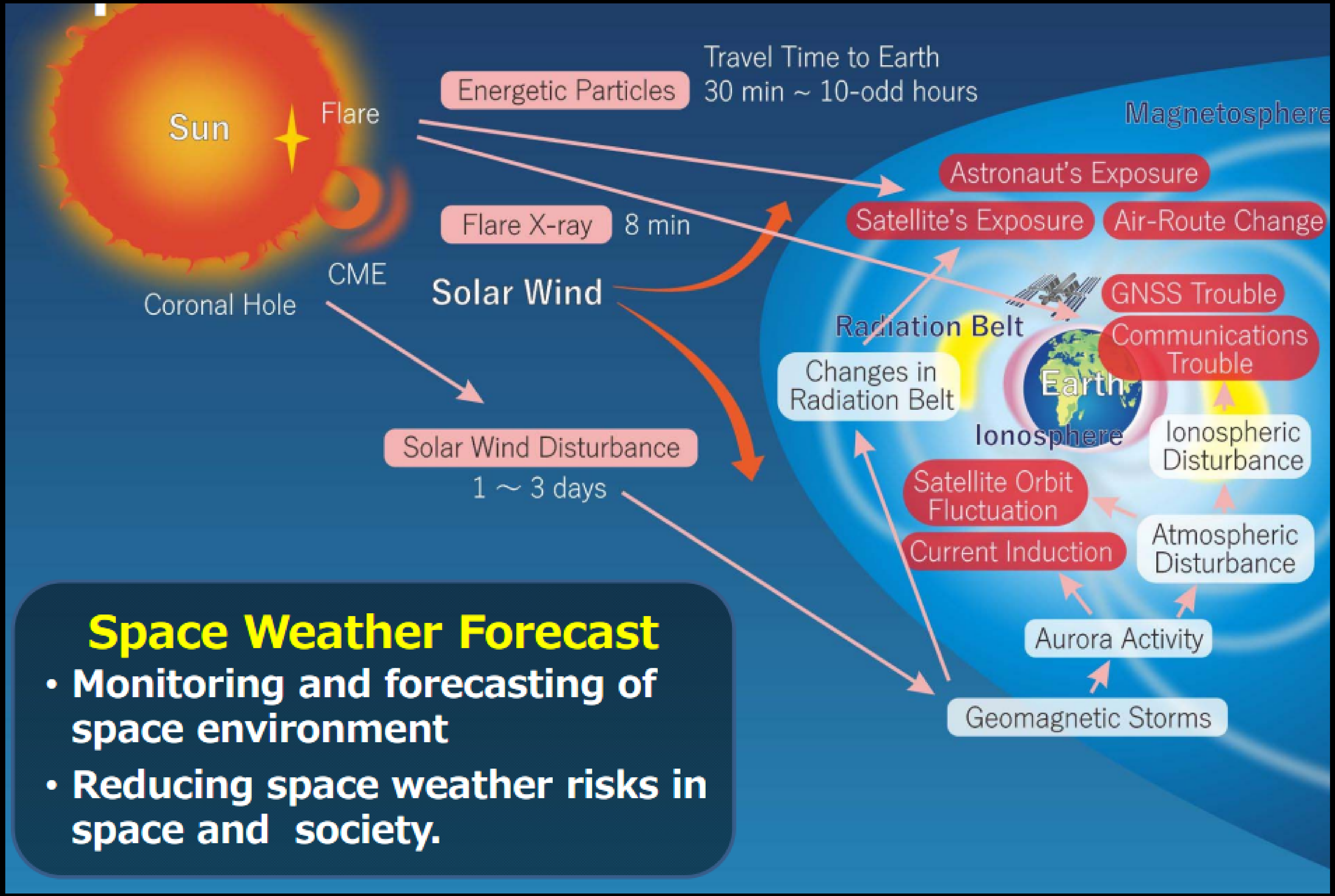


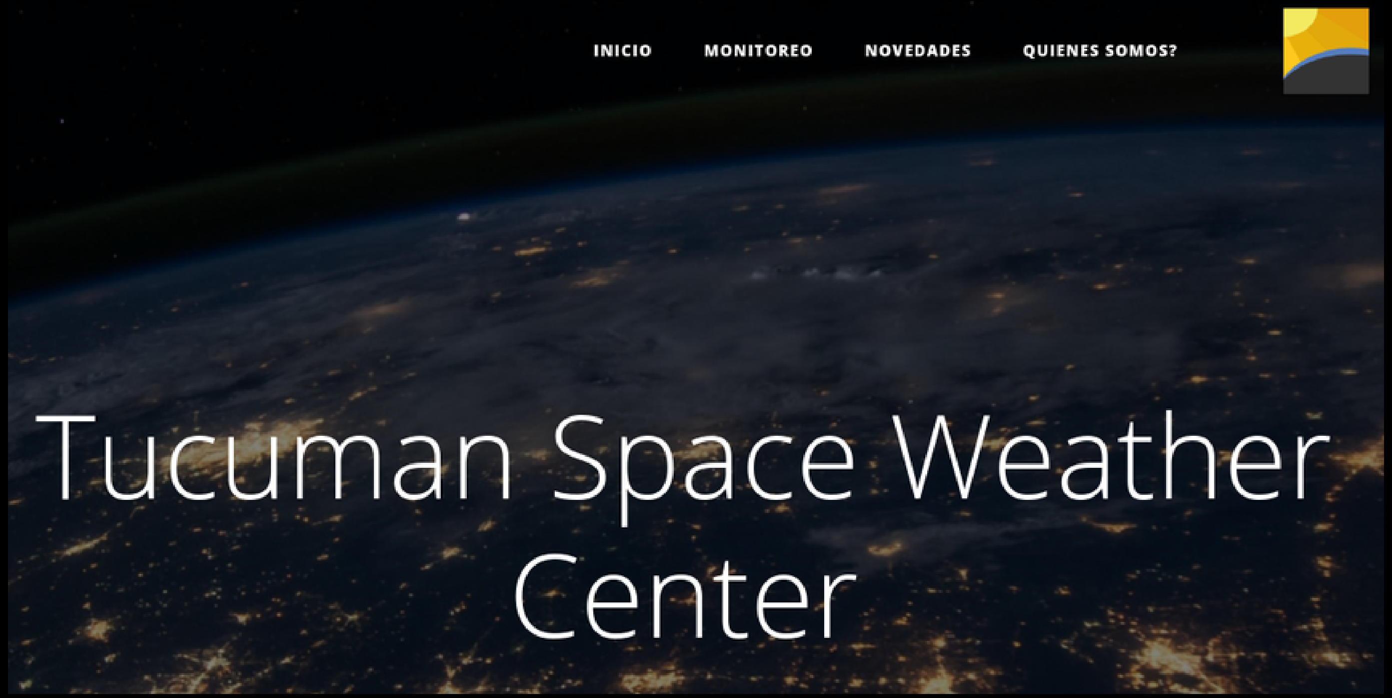
Other methods

- Bayesian Optimization
- Evolutionary Optimization.

https://towardsdatascience.com/hyperparameters-tuning-from-grid-search-to-optimization-a09853e4e9b8

Something we are working on …





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

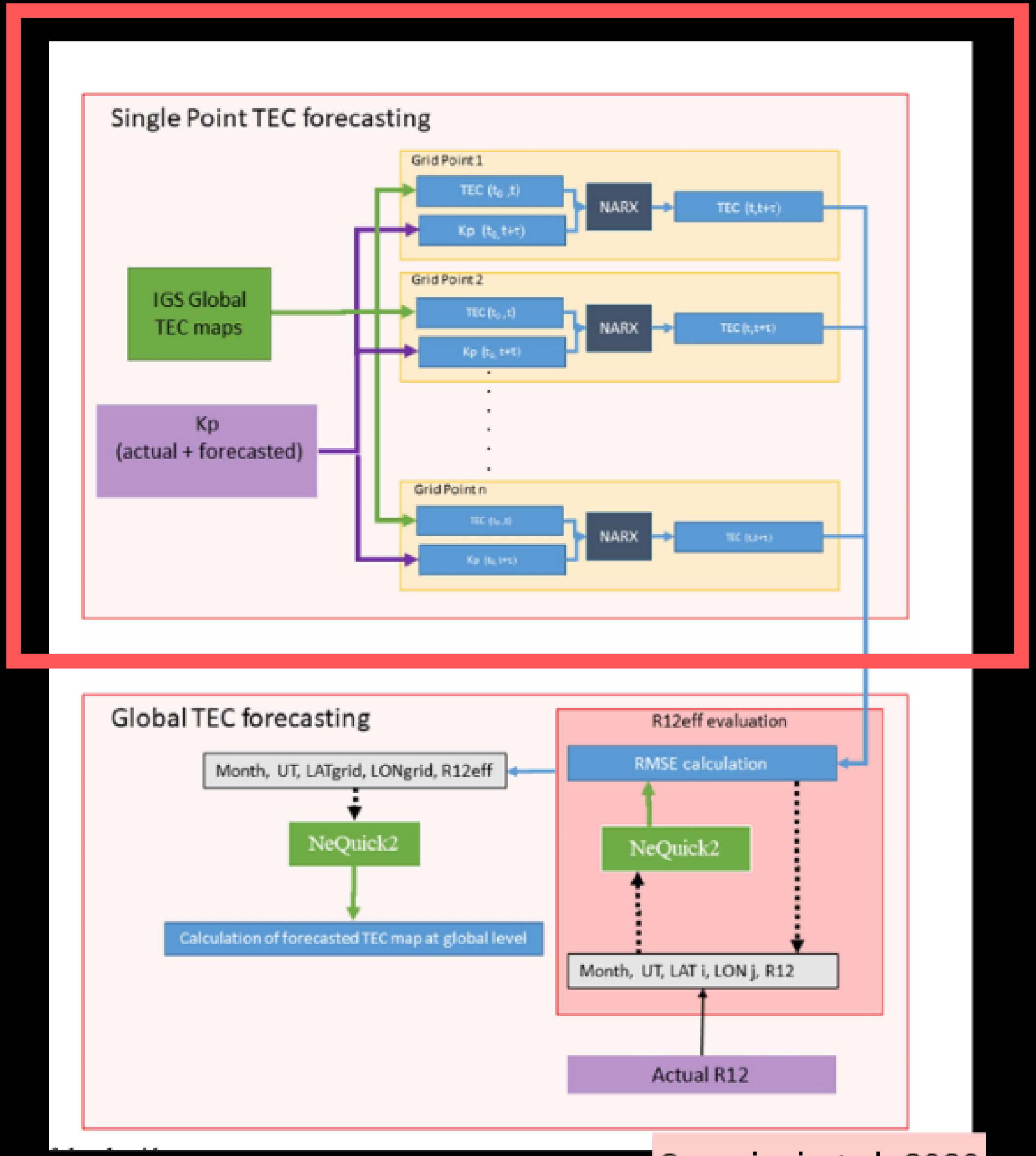
The problem: 24hs forecasting of TEC given information regarding geomagnetic conditions

GLOBAL TEC FORECASTING

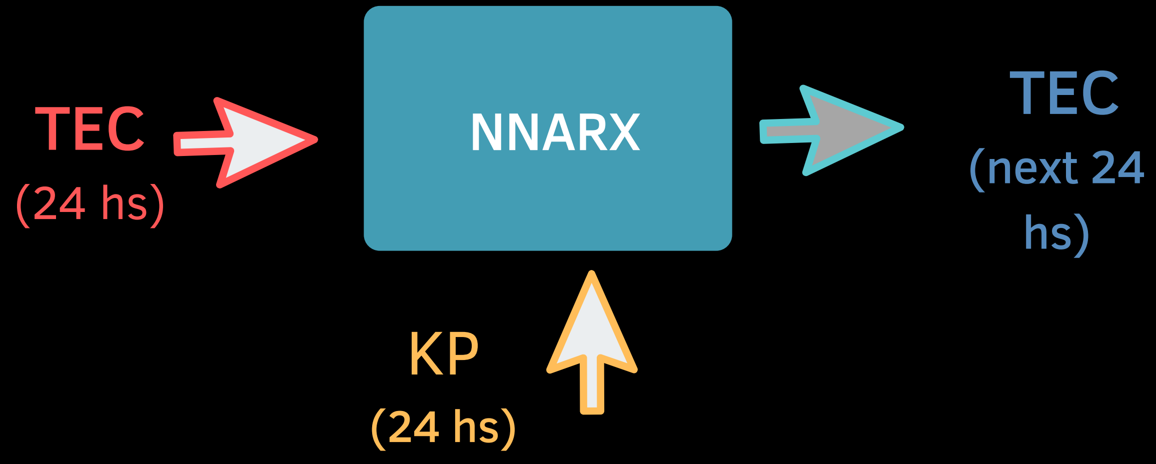
In prep

TSWC, 2022

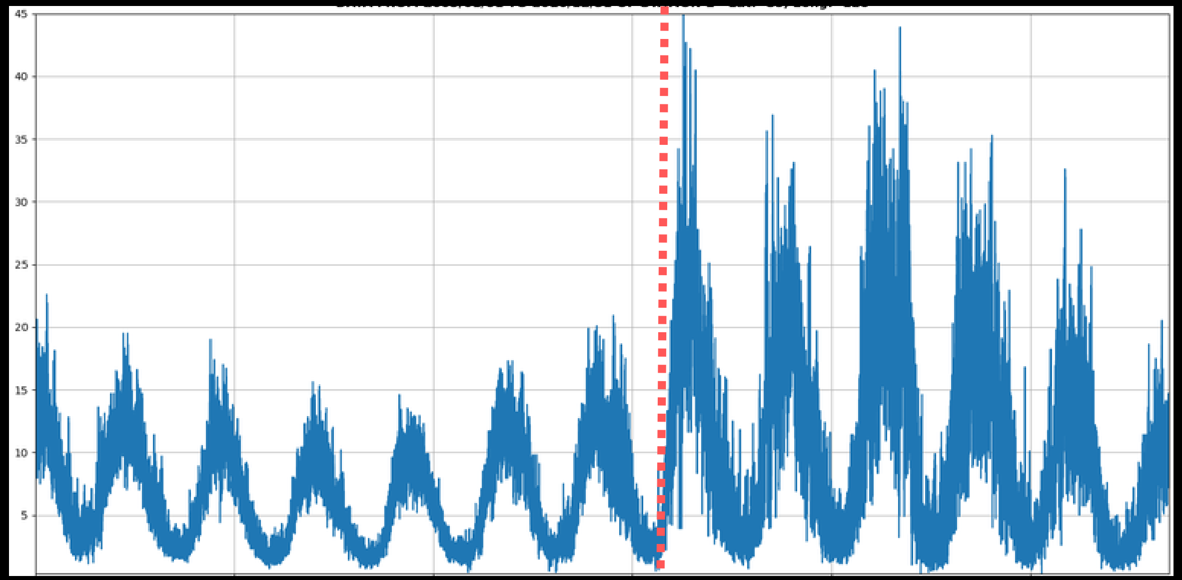
- 24 hs ahead
- Dataset: 2005 - 2016
- **Input**: Global Ionospheric Map (GIM) from IGS. Spatial-temporal res 2.5° (lat) - 5° (lon) - 2 h
- **External forcing** (* SWx): Kp index
- Loosely physics-informed ML

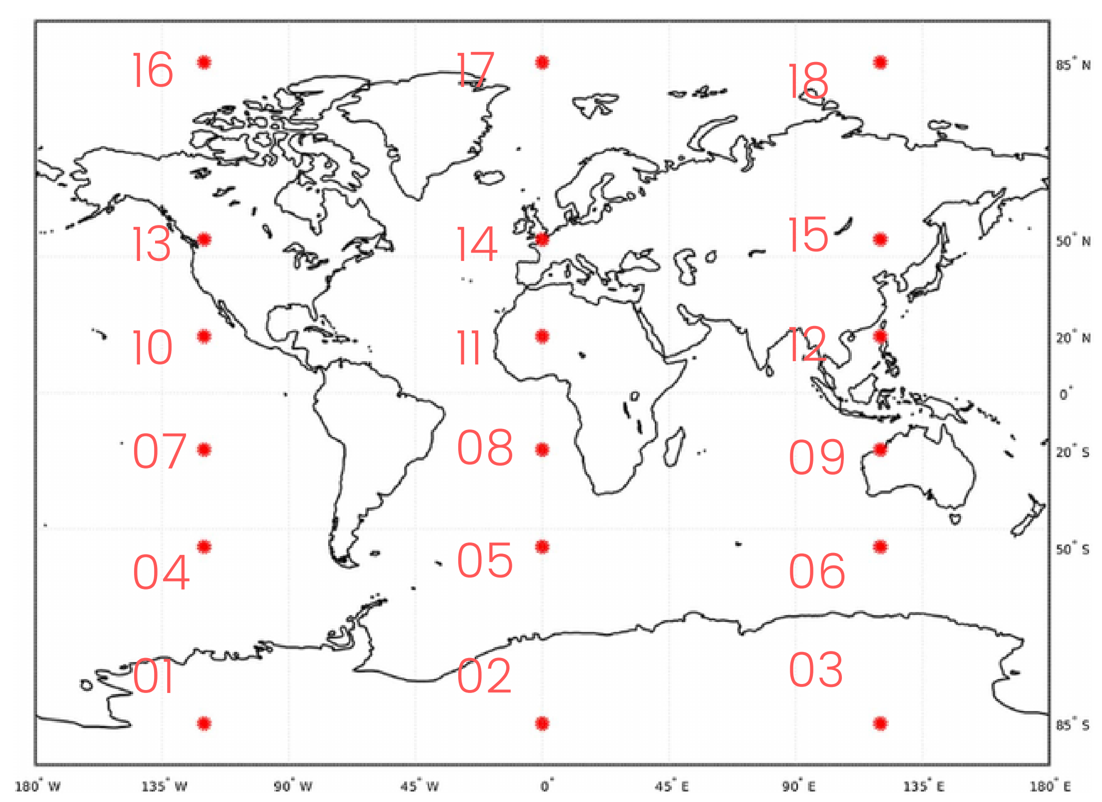


Cesarioni et al, 2020



St 01 - dataset (2005 - 2016)



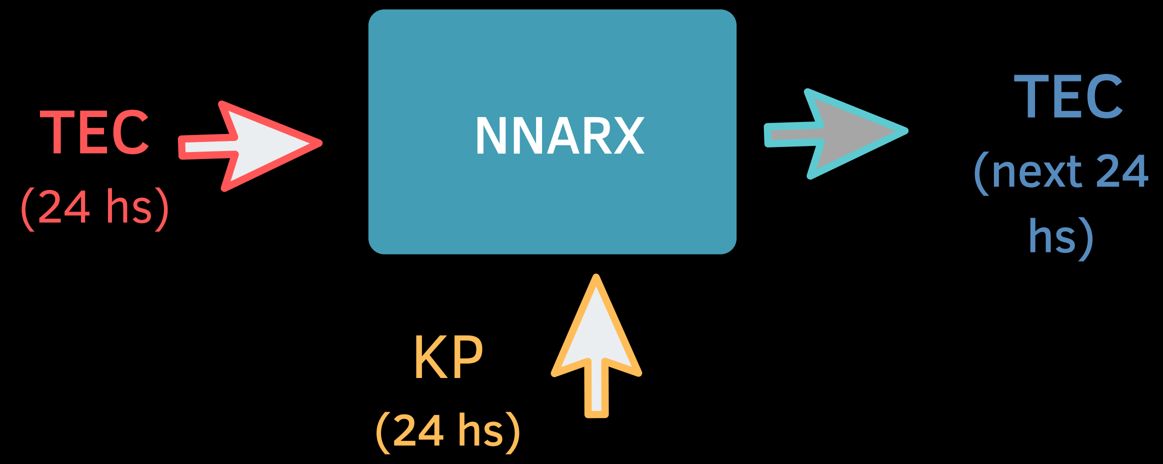


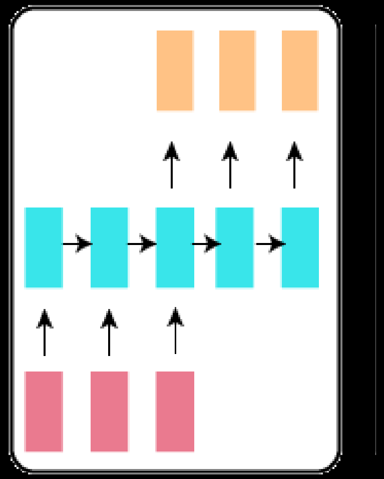
(Molina et al, ESWW 2021)

GLOBAL TEC FORECASTING

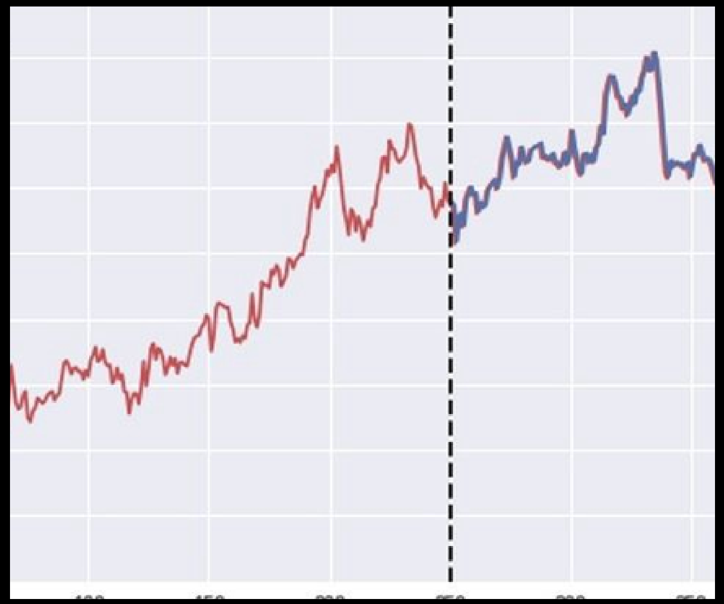
In prep

- Train: 99 %(99/1) Test: 1 %
- Re-sampling:
  - GIM 2 hs resolution
  - Kp 3hs resolution > K Nearest-neighbor interpolation
- No missing values
- Kernel initializer: GlorotNormal distribution

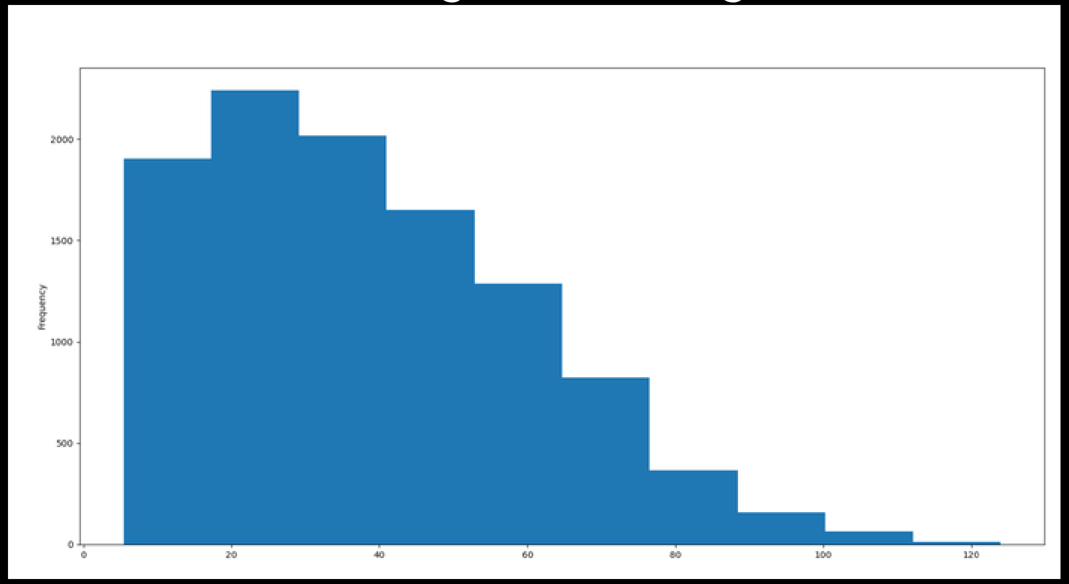




- DL modeling
  - 24 hs (before) to forecast 24 hs (ahead) - 24 hs = 12 samples
  - supervised ML
  - 3 methods
  - time-series



TEC - single ST Histogram



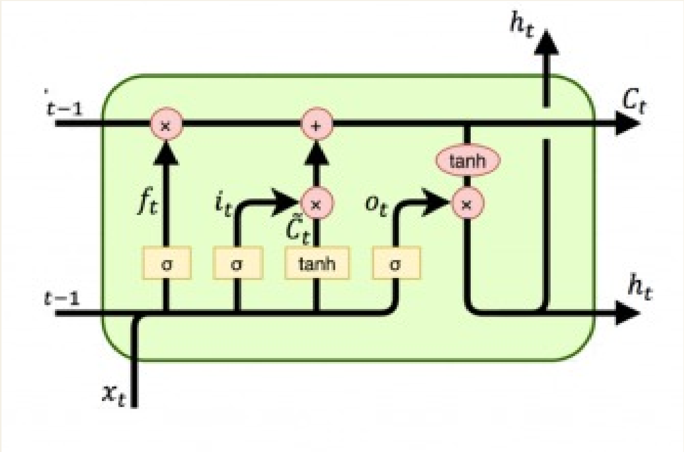
(Molina et al, ESWW 2021)

GLOBAL TEC FORECASTING

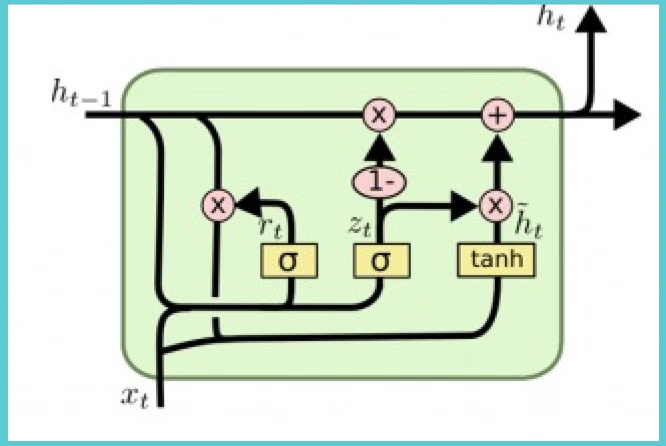
TSWC, 2022

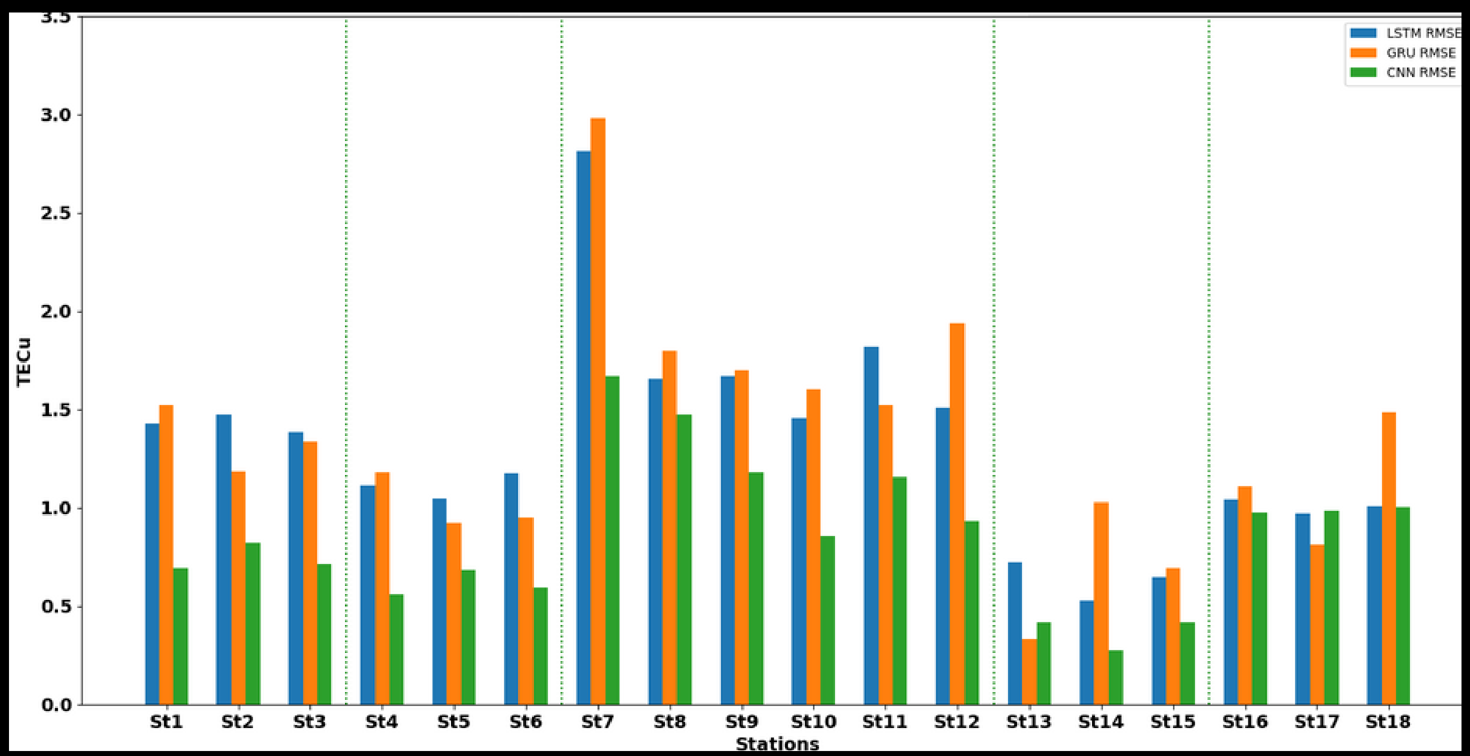
(Molina et al, ESWW 2021)

LSTM

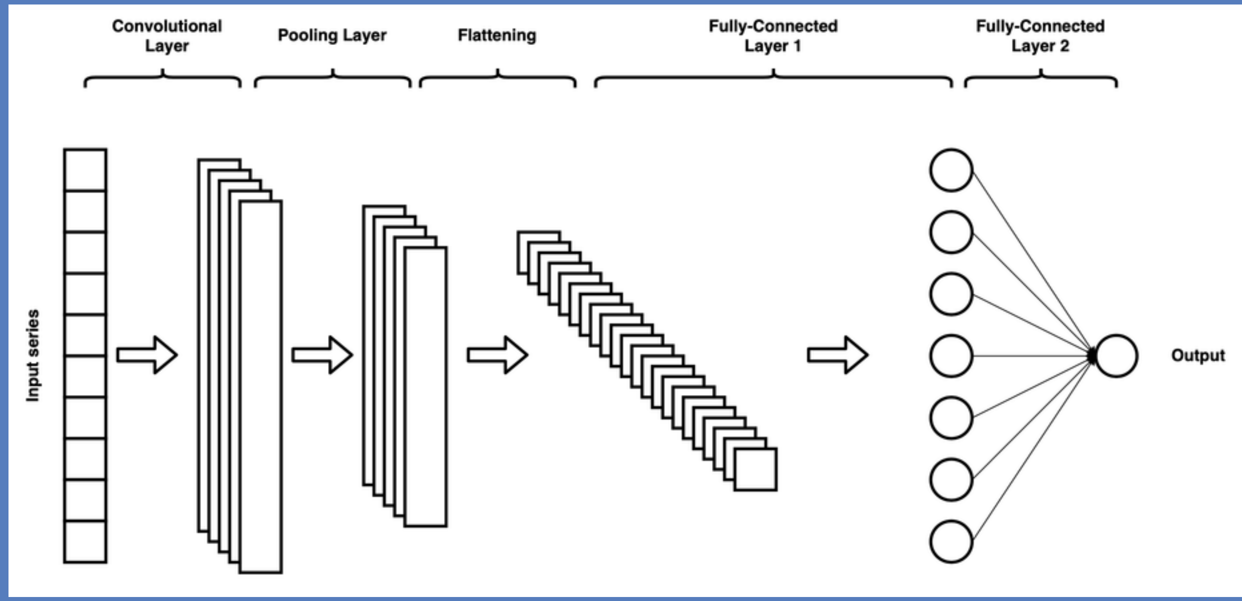


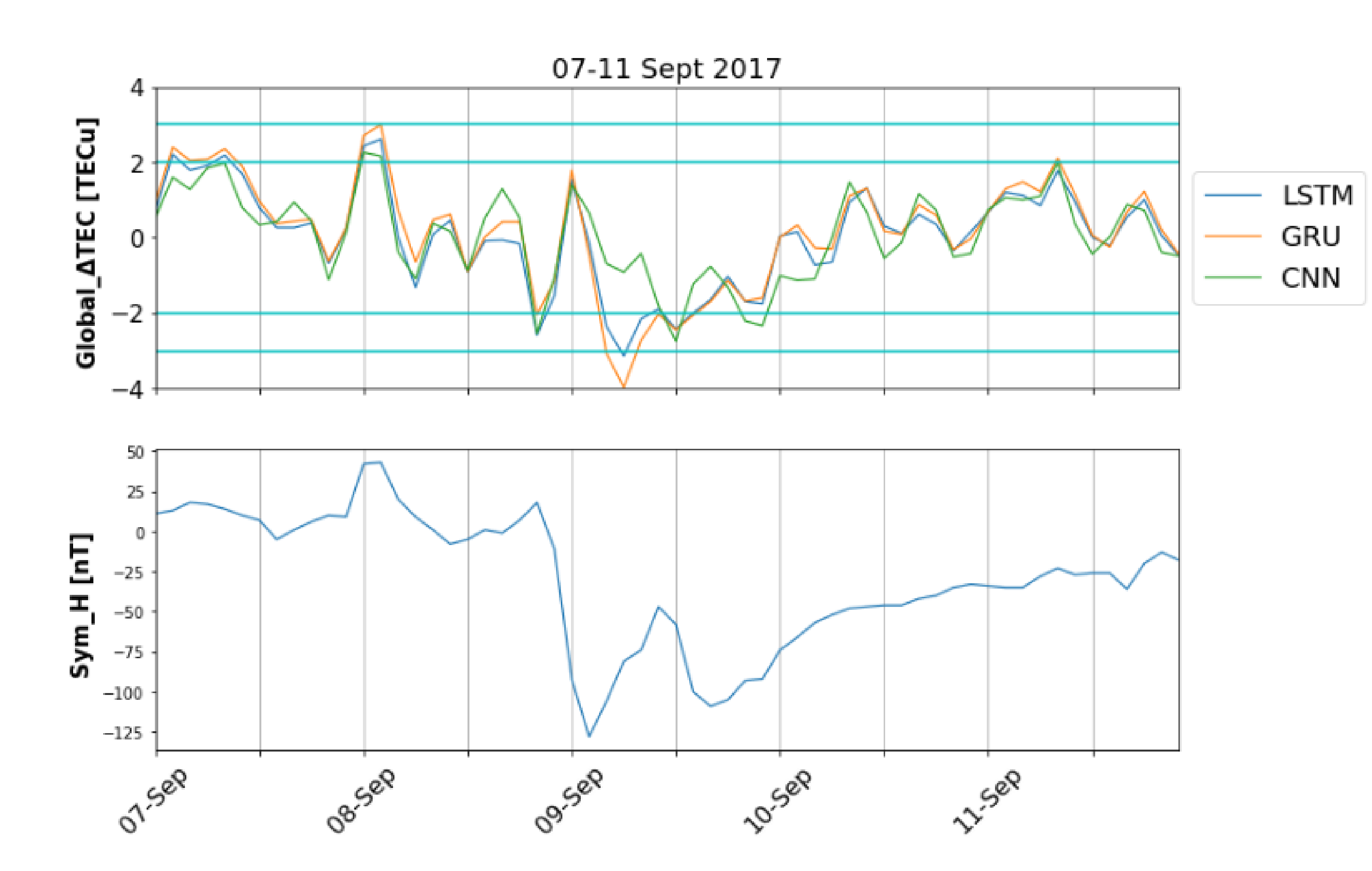
GRU





CNN - 1D





In prep

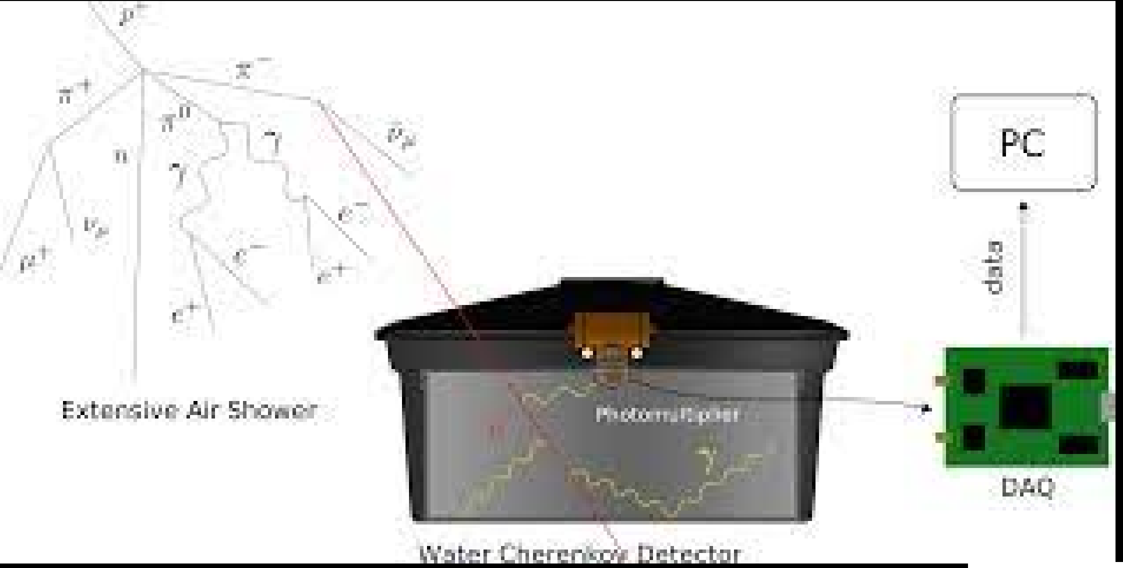
Something (else) we are working on ...

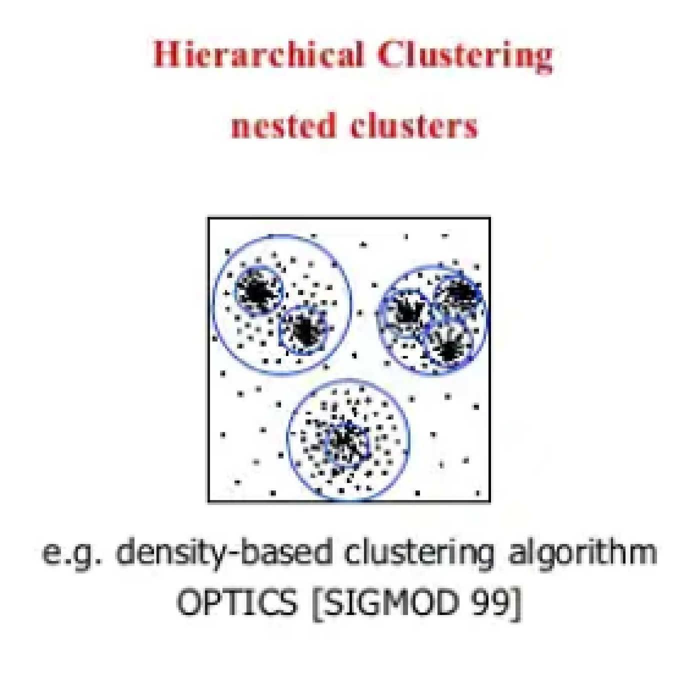
Ticiano Torres Peralta

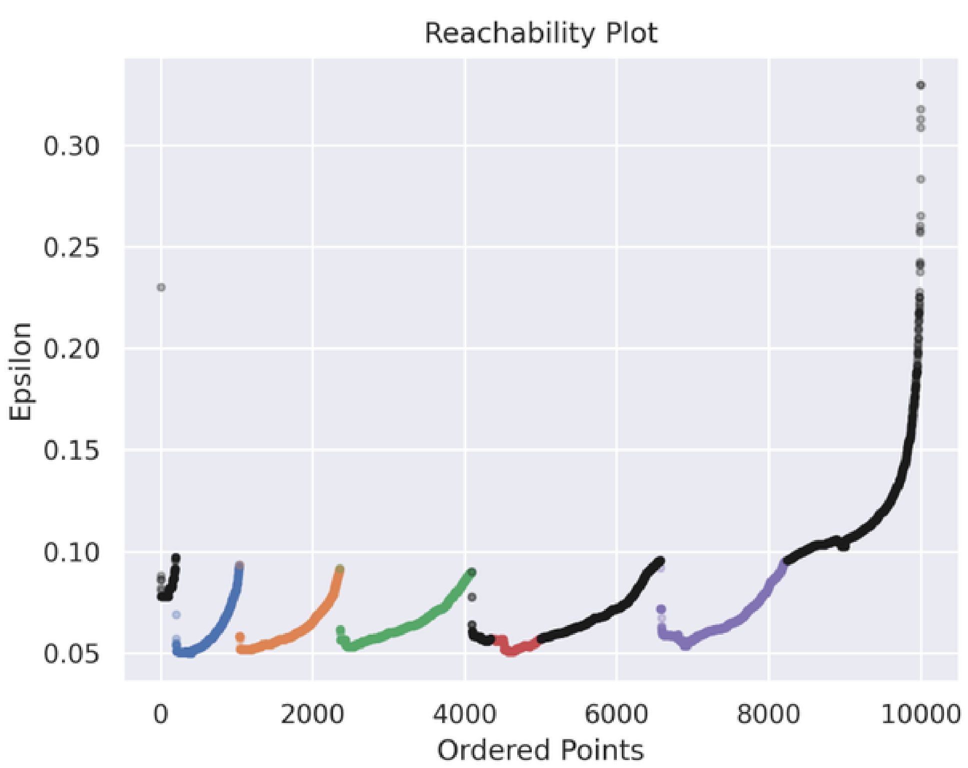
Clustering: Ordering points to identify the clustering structure (OPTICS)

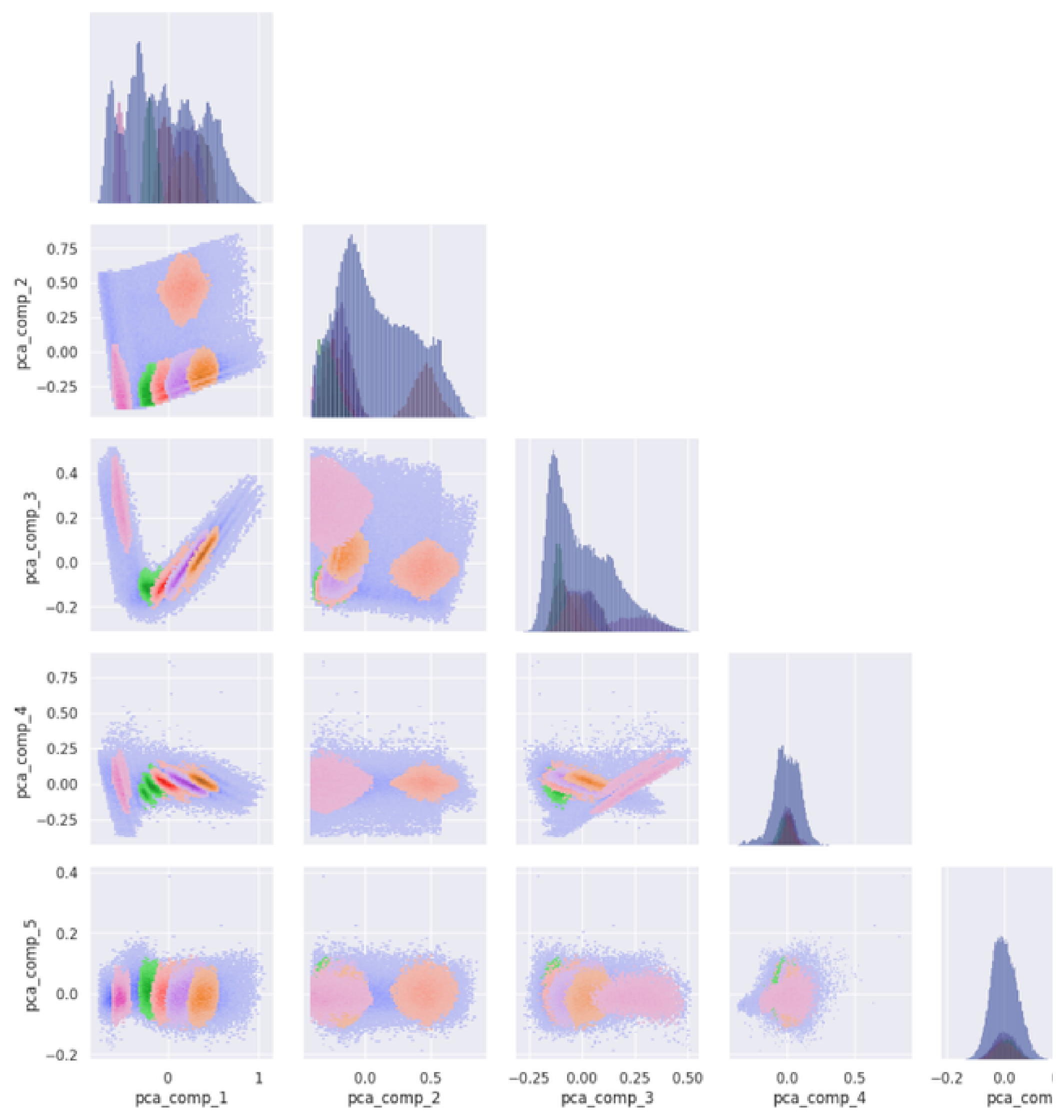
http://lagoproject.net/

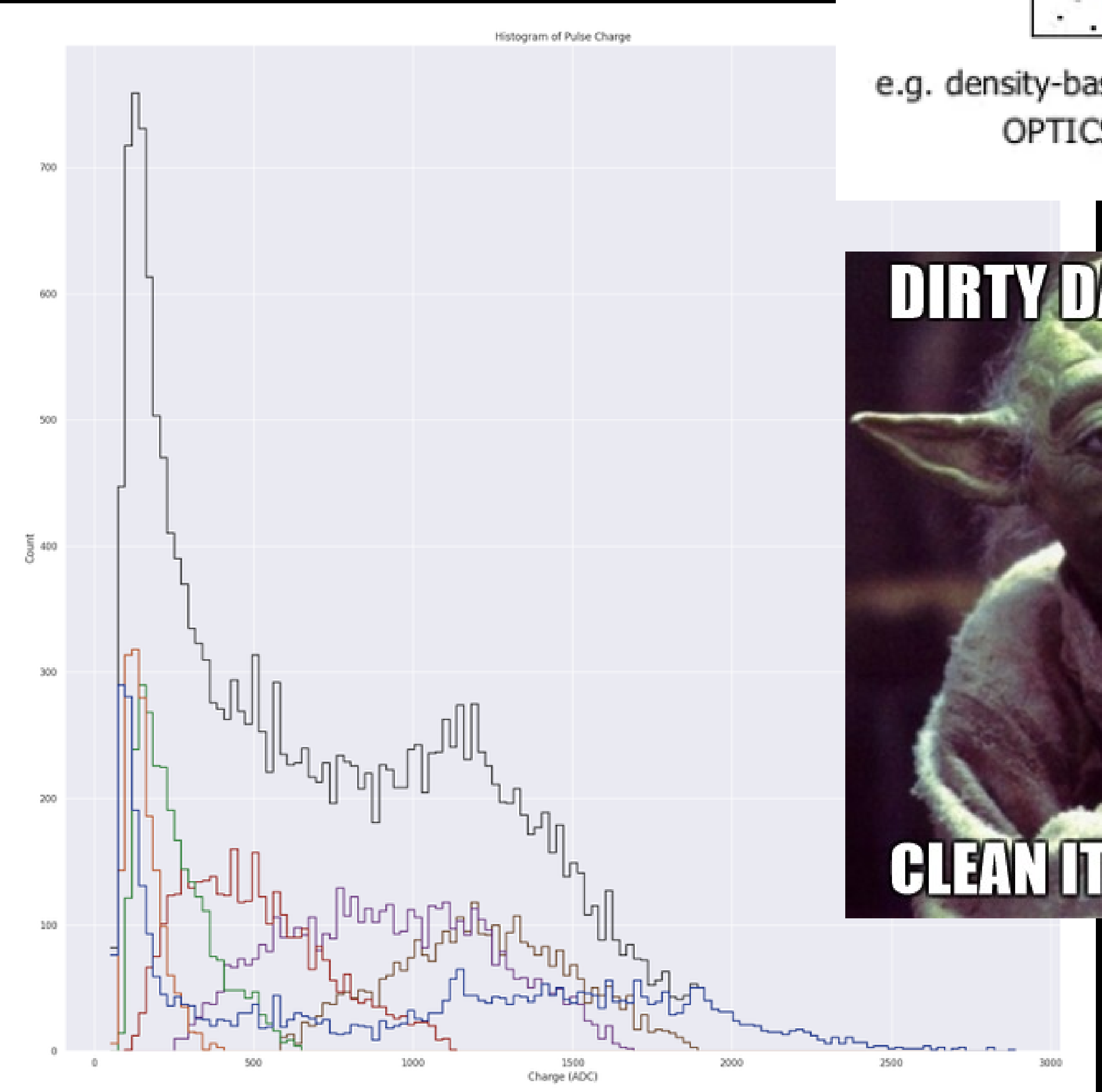
In prep











TSWC, 2022

https://indico.ictp.it/event/9840/

About Visit Women in Science Marie Curie Library

The Abdus Salam International Centre for Theoretical Physics

IAEA

DONATE

Research Scientific Calendar Programmes

Europe/Rome Administration

Search

Search in Conferences:

- Overview
- Programme
- Speakers
- Apply here

International Workshop on Machine Learning for Space Weather: Fundamentals, Tools and Future Prospects | (smr 3750)

Starts 7 Nov 2022
Ends 11 Nov 2022
Central European Time

Buenos Aires - Argentina

Organizer(s)

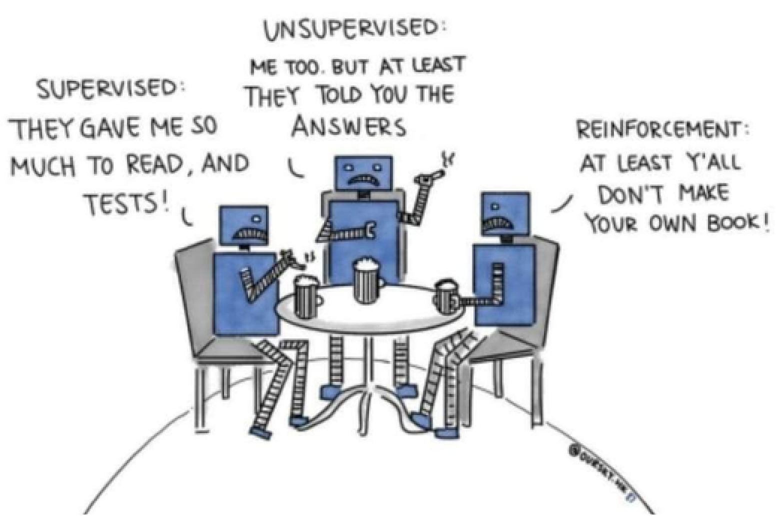
Patricia Doherty (Boston College), Sharafat Gadimova (UNOOSA), María Graciela Molina (FACET-UNT / CONICET), Yenca Migoya Orué (ICTP), ICTP Scientific Contact: Bruno Nava (ICTP STI Unit)

Secretary Elizabeth Brancaccio

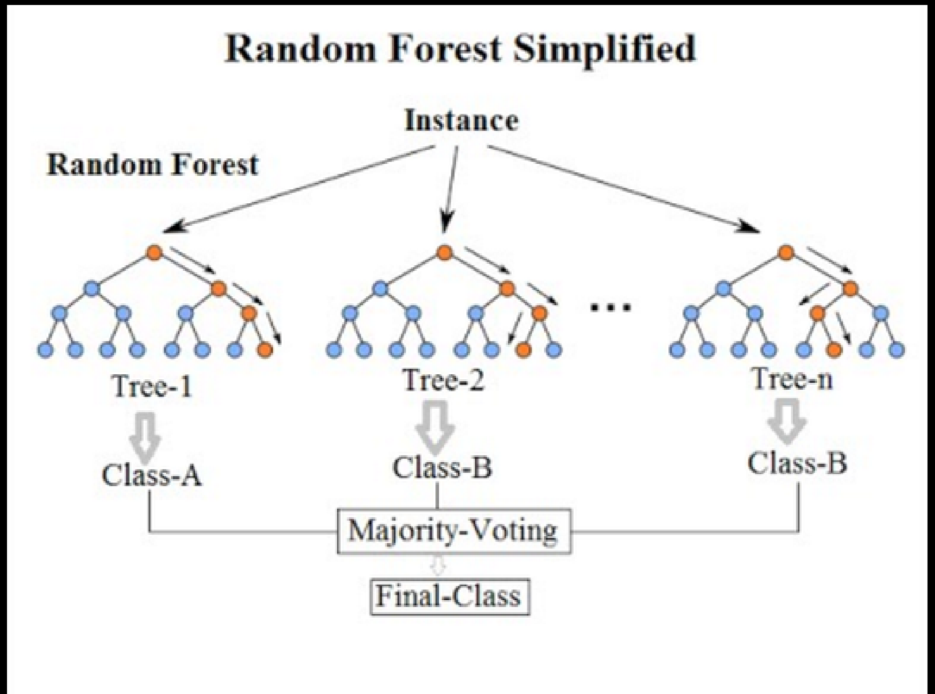
Organizers

Patricia Doherty (Boston College), Sharafat Gadimova (UNOOSA), María Graciela Molina (FACET-UNT / CONICET), Yenca Migoya Orué (ICTP), ICTP Scientific Contact: Bruno Nava (ICTP STI Unit)

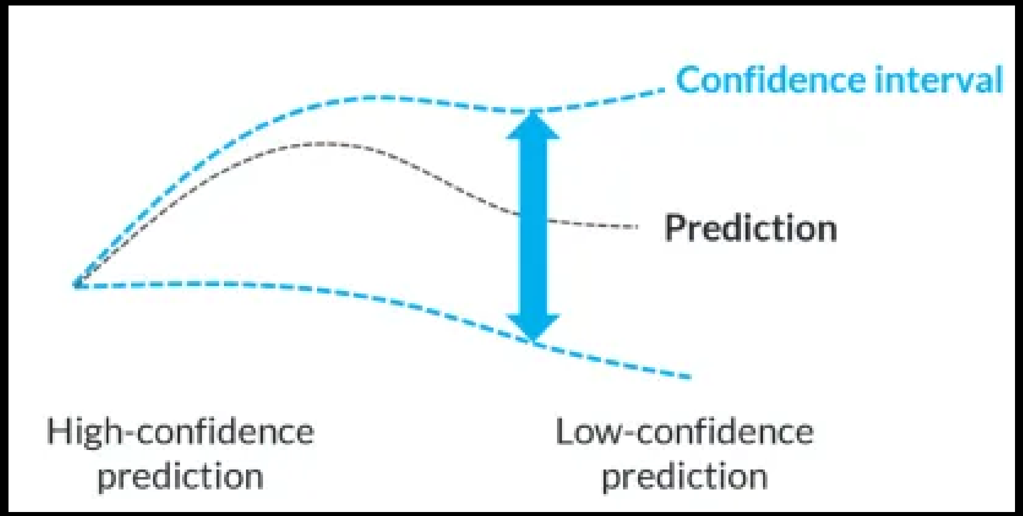
MACHINE LEARNING



- Ensemble techniques

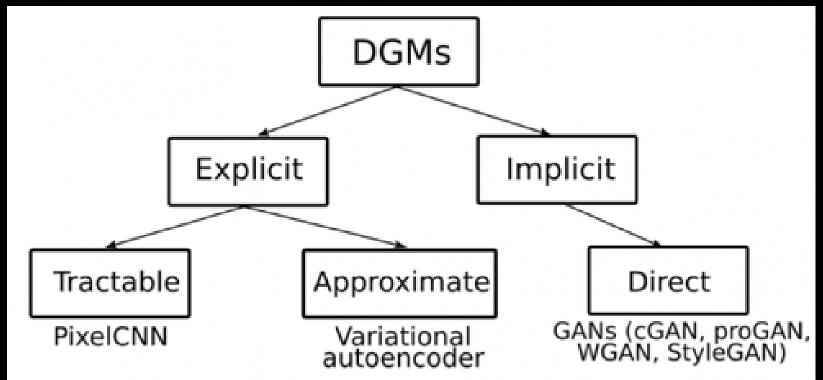


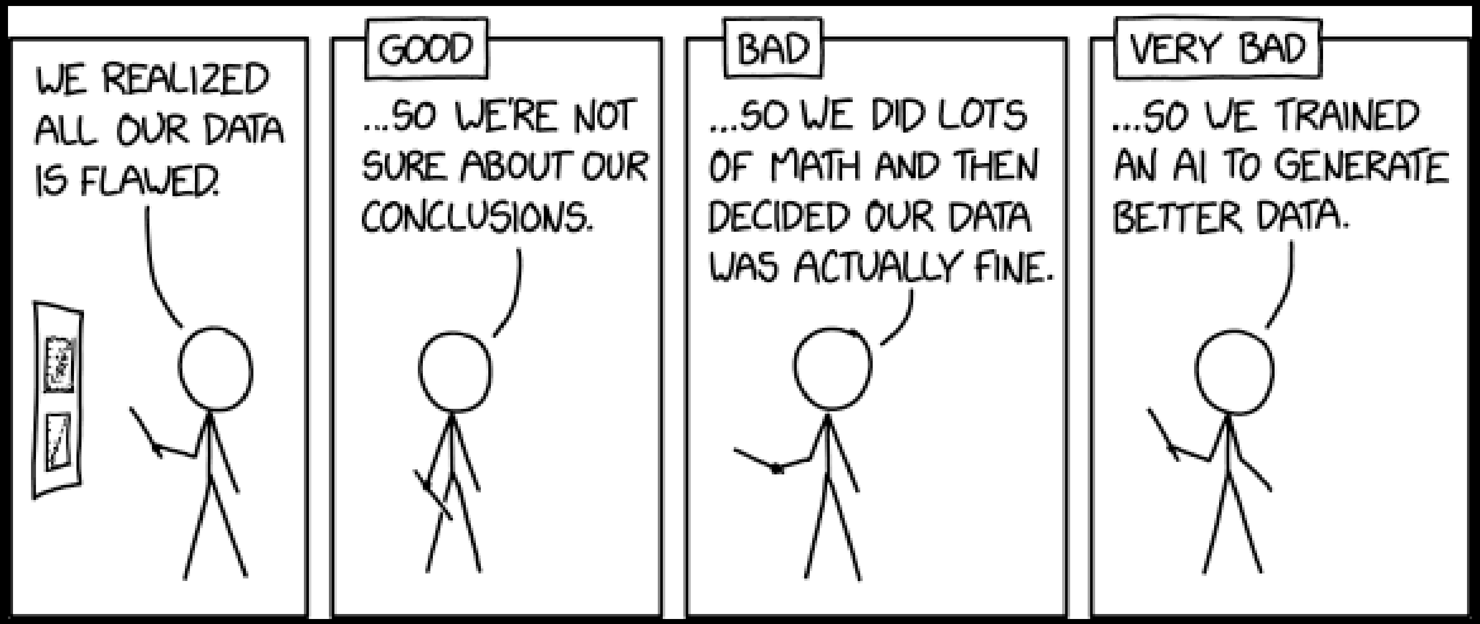
- Uncertanty Quantification (BNNs) - trustworthiness

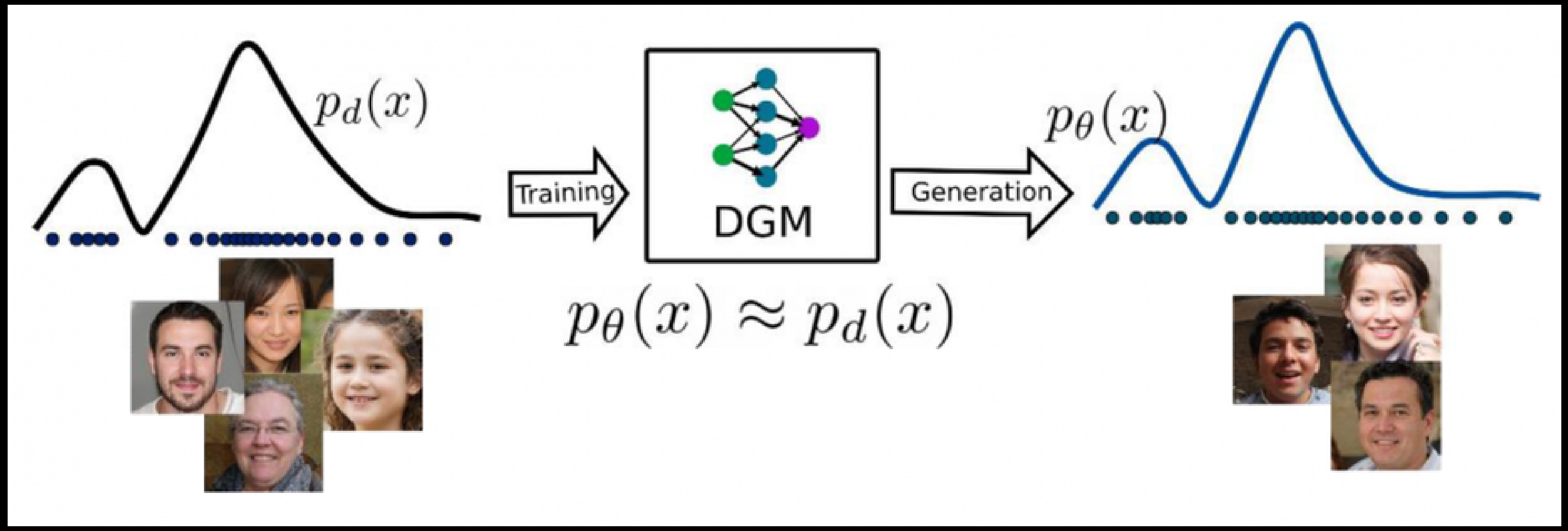


- Transformers
- Transfer Learning !

- Deep Generative Models







- (ML in production) Real -time -> re-training, incremental training, etc

and much more!!!

OPEN DISCUSSION