Atmospheric Retrievals in a Machine Learning Context: A Radiometric Story Over the Ocean

Mario Echeverri Bautista[1], Anton Verhoef[1], Ad Stoffelen[1], Maximilian Maahn[2]

(1) KNMI
(2) Leipzig University, Institute for Meteorology

Koninklijk Nederlands Meteorologisch Instituut
Ministerie van Infrastructuur en Milieu

UNIVERSITÄT LEIPZIG

Outline

- The context:
  - Machine Learning
  - Geo-Science: e.g. Earth Observation & Big Data
- Open Source & Community Dev.
- A Research Workflow:
  - How?
- Our Working Example
- Wrap up

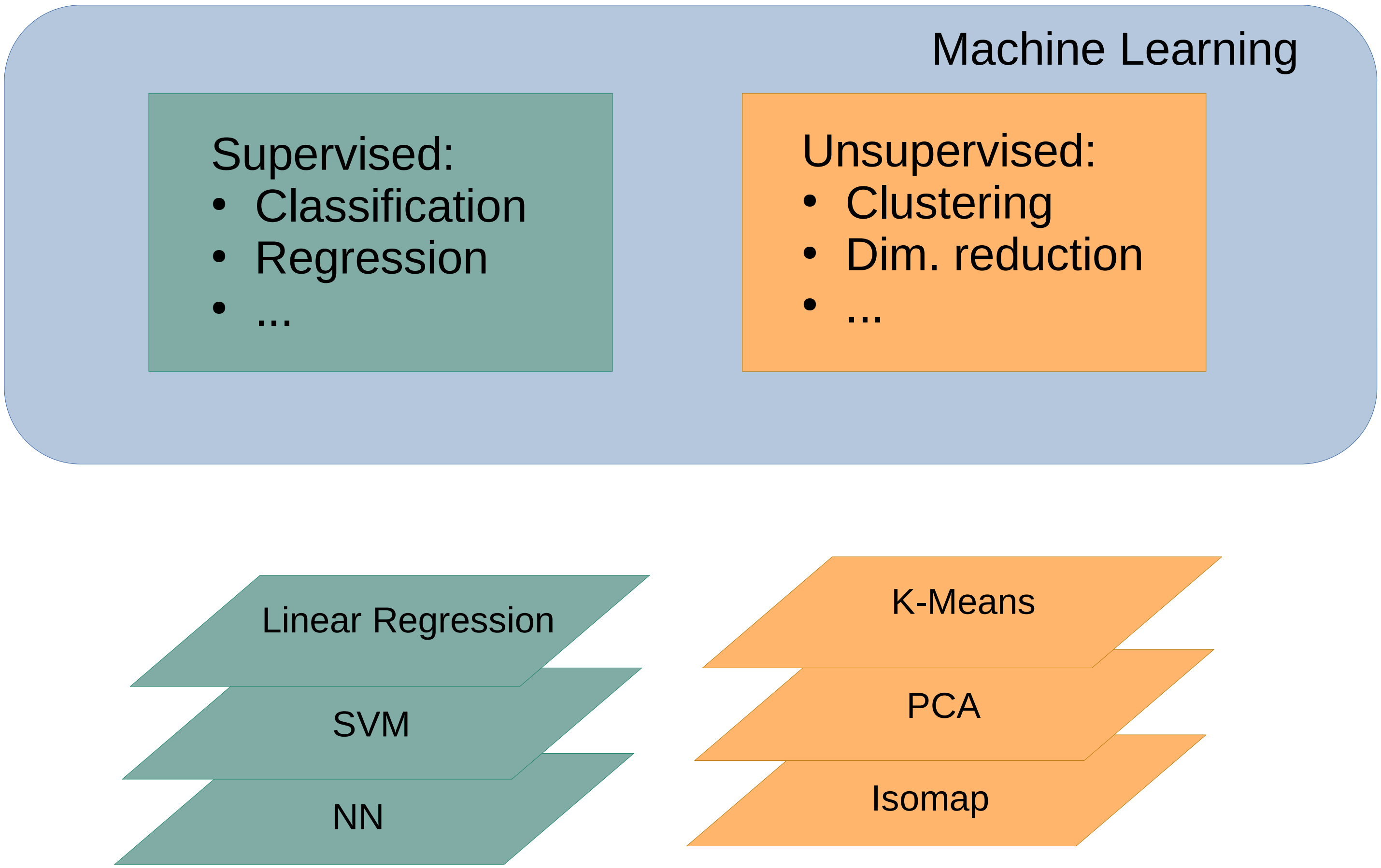
The Machine Learning Context

Machine Learning

Supervised:
- Classification
- Regression
- ...

Unsupervised:
- Clustering
- Dim. reduction
- ...

The Machine Learning Context



The Machine Learning Context

Machine Learning

Supervised:
- Classification
- Regression
- ...

Unsupervised:
- Clustering
- Dim. reduction
- ...

Linear Regression

SVM

NN

K-Means

PCA

Isomap

The Geo-Science Context

e.g. Earth Observation using MWI radiometers and Optimal Estimation:
SIC, NS Wind speed, SST, etc.

Multidimensional datasets:
- Observations
- Apriori data

Forward model:
- Physics based

The Geo-Science Context

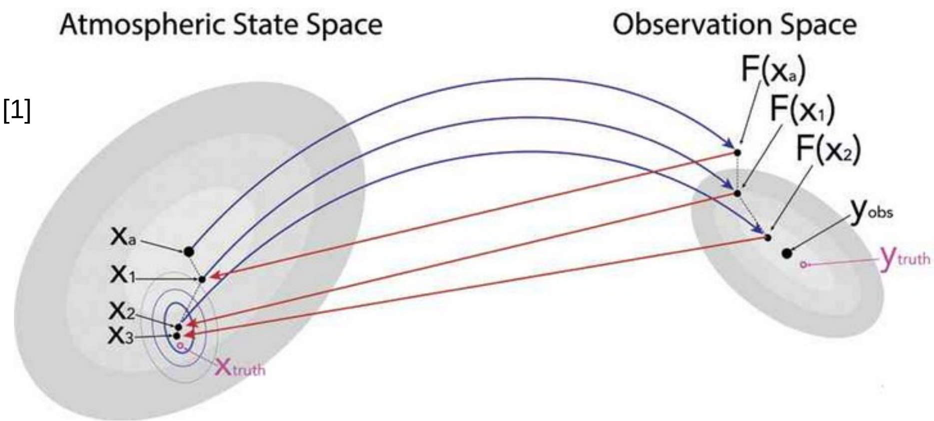
e.g. Earth Observation using MWI radiometers and Optimal Estimation:
SIC, NS Wind speed, SST, etc.

Multidimensional datasets:
- Observations
- Apriori data

Forward model:
- Physics based

(

From [1]



)

The Geo-Science Context

e.g. Earth Observation using MWI radiometers and Optimal Estimation:
SIC, NS Wind speed, SST, etc.

Multidimensional datasets:
- Observations
- Apriori data

Forward model:
- Physics based

Split-Apply-Combine

Masking & Missing data

Time series

Model specific

The Geo-Science Context

e.g. Earth Observation using MWI radiometers and Optimal Estimation:
SIC, NS Wind speed, SST, etc.

Multidimensional datasets:
- Observations
- Apriori data

Forward model:
- Physics based

Pangeo

Split-Apply-Combine

Masking & Missing data

Time series

Model specific

The Geo-Science Context

e.g. Earth Observation using MWI radiometers and Optimal Estimation:
SIC, NS Wind speed, SST, etc.

Multidimensional datasets:
- Observations
- Apriori data

Forward model:
- Physics based

Pangeo

Split-Apply-Combine

Masking & Missing data

Time series

Model specific

- **CRTM**
- **RTTOV**

The EUMETSAT Network of Satellite Application Facilities

NWP SAF
Numerical Weather Prediction

Open Source / Community Development

xarray

Iris

DASK

jupyter

Pangeo

- Pyresample
- Satpy
- Many more...

PyTROLL

NumPy

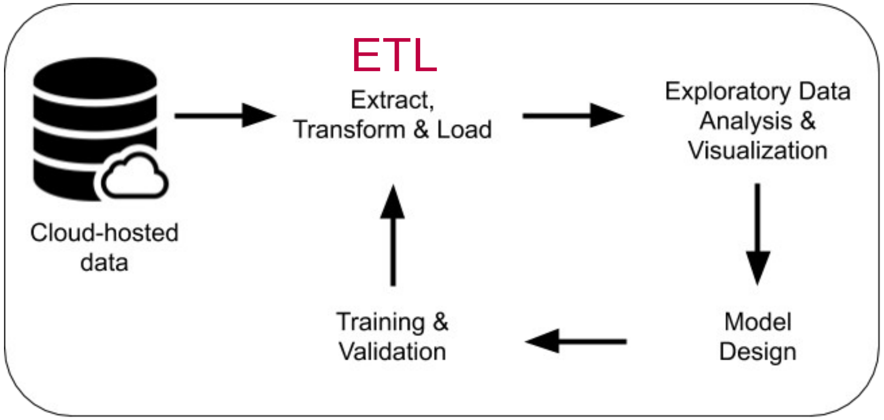
pandas

matplotlib

And more...

SciPy.org

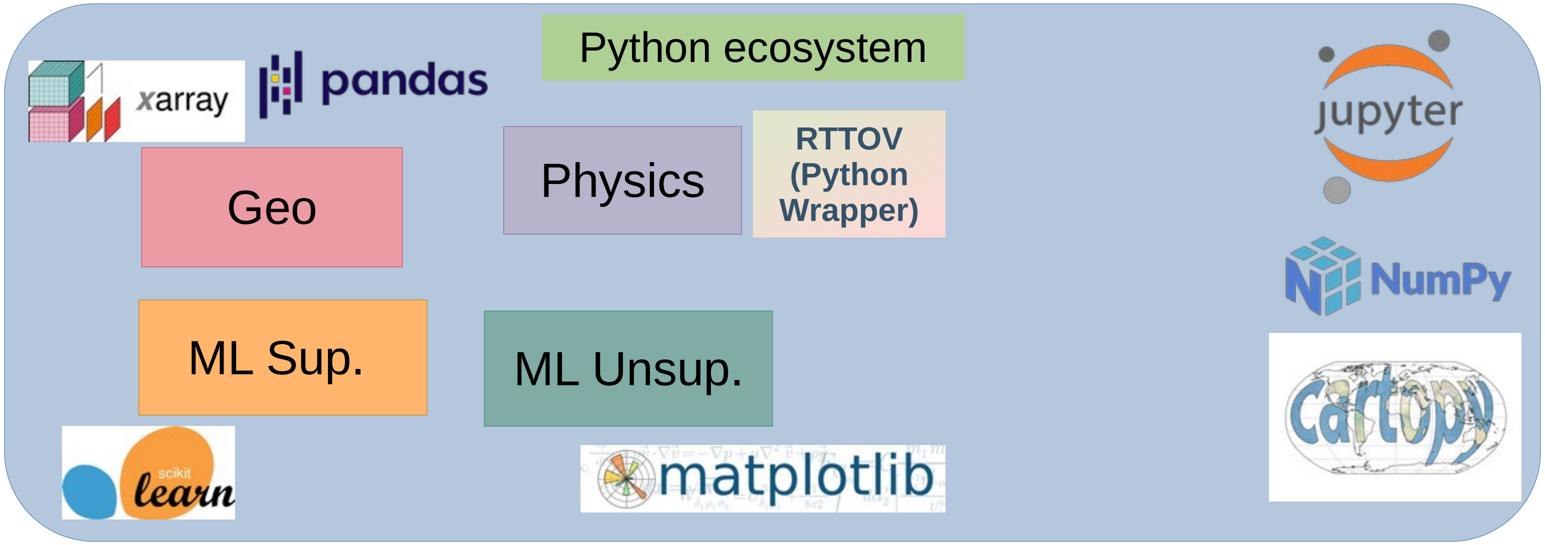
A research workflow [5]

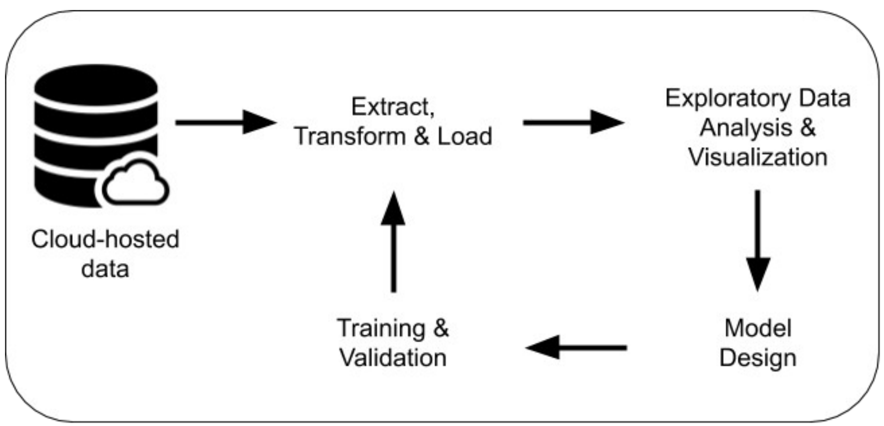


- 80 / 90 % of the time is spent in ETL, the rest is actual data analysis / use

- Open source / Community development provides “key improvements to our ability to share, reproduce and scale ML workflows in geosciences.”

How?

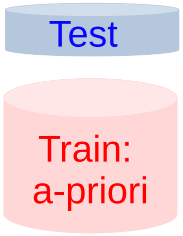

Python ecosystem
xarray
pandas
Geo
Physics
RTTOV (Python Wrapper)
ML Sup.
ML Unsup.
scikit learn
matplotlib
jupyter
NumPy
Cartopy

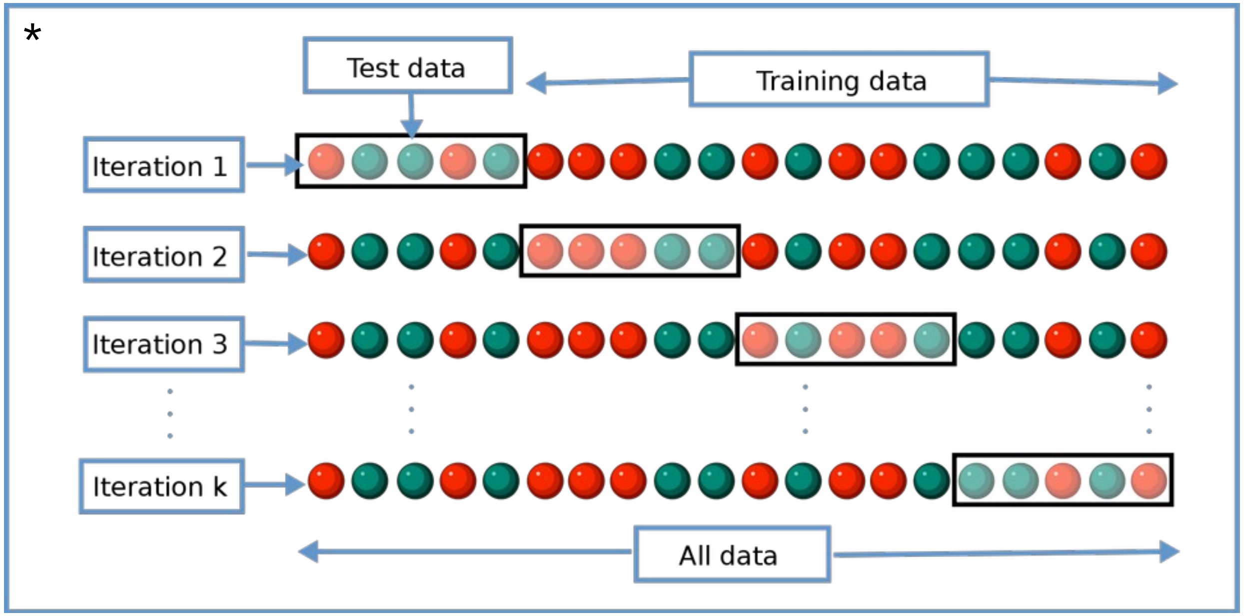

Cloud-hosted data
Extract, Transform & Load
Exploratory Data Analysis & Visualization
Model Design
Training & Validation

PyOpEst: Python Optimal Estimation

- Open source tool developed by M. Maahn, [1]
- Object oriented, based on Pandas data structures
- Originally developed in the context of ground based radiometers
- Now used in the context of onboard radiometers:
  - We have improved the speed of the Jacobians computation
  - We have expanded its scope by allowing the use of an external tool to compute Jacobians.
  - An open source example of **PyOpEst** + **RTTOV (Python's wrapper for it)** is now available: https://github.com/deweatherman/RadEst
  - Plug and play tool

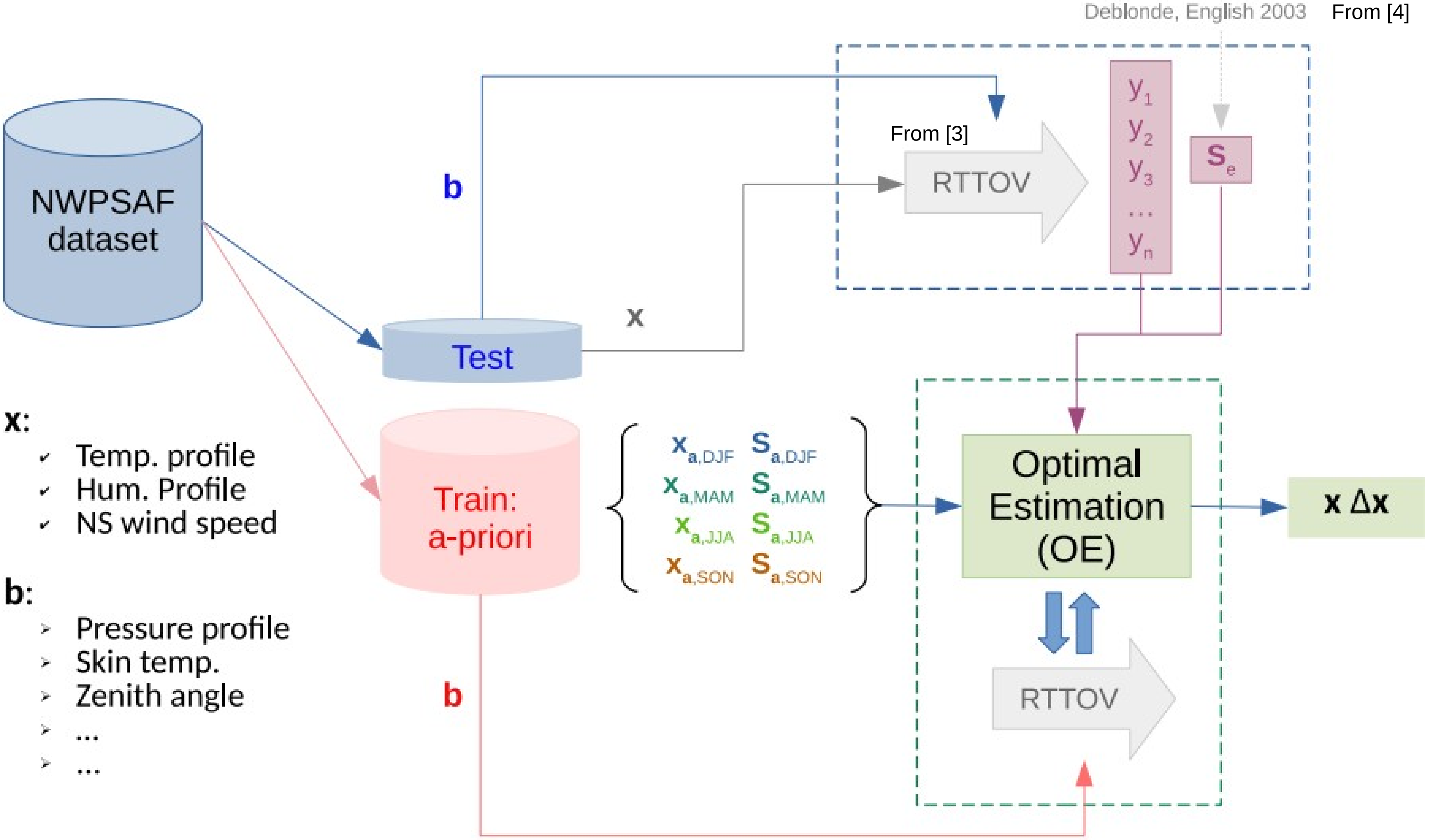
Our working example: K-Fold Cross validation





* By Gufosowa - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=82298768

Our working example: K-Fold Cross validation



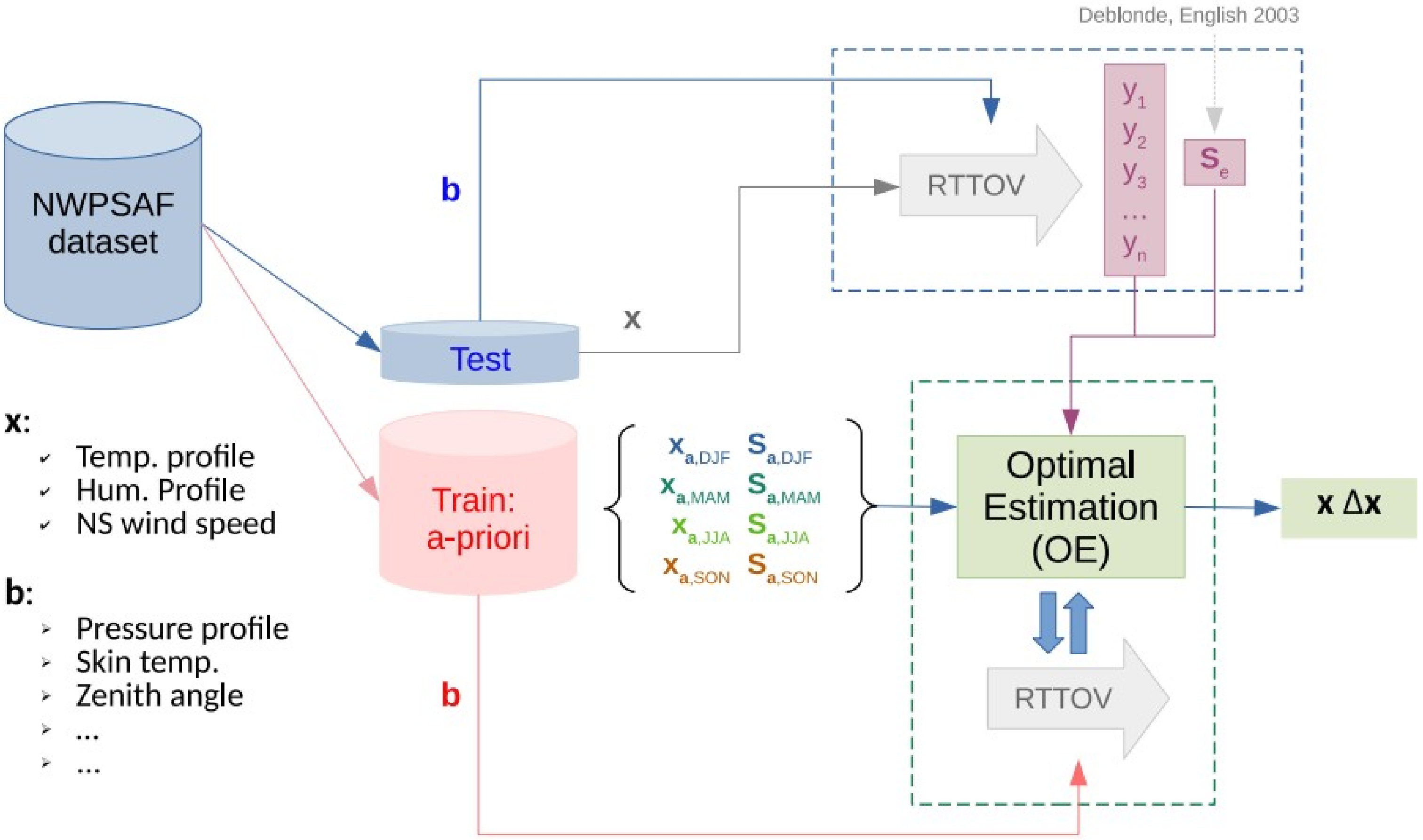
x:

- Temp. profile
- Hum. Profile
- NS wind speed

b:

- Pressure profile
- Skin temp.
- Zenith angle
- …
- …

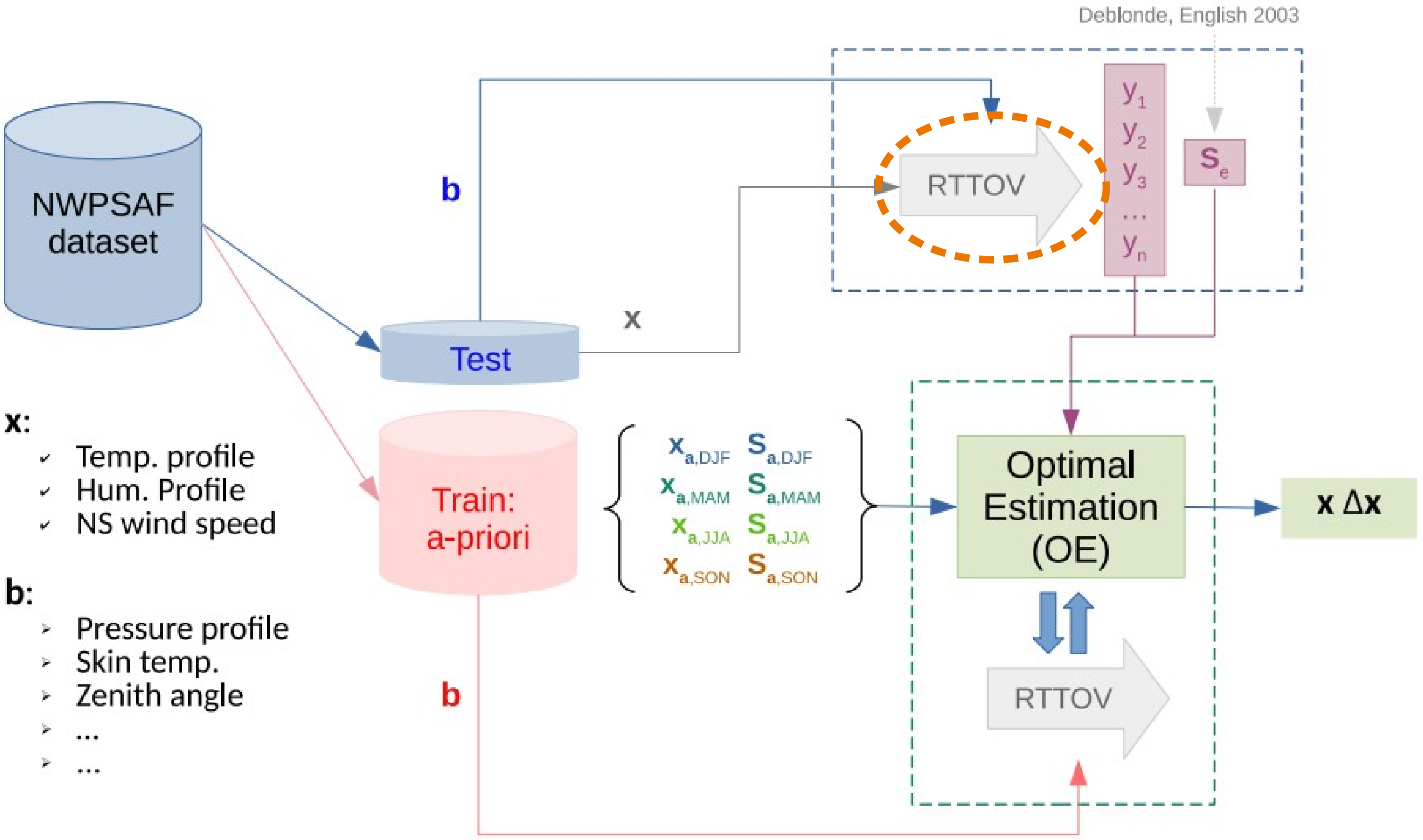
Our working example: K-Fold Cross validation



x:
- Temp. profile
- Hum. Profile
- NS wind speed

b:
- Pressure profile
- Skin temp.
- Zenith angle
- …
- …

Our working example: K-Fold Cross validation



x:

- Temp. profile
- Hum. Profile
- NS wind speed

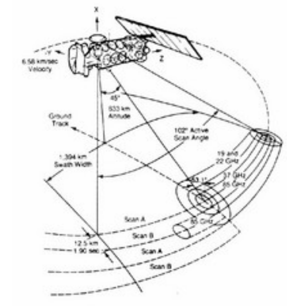
b:

- Pressure profile
- Skin temp.
- Zenith angle
- …
- …

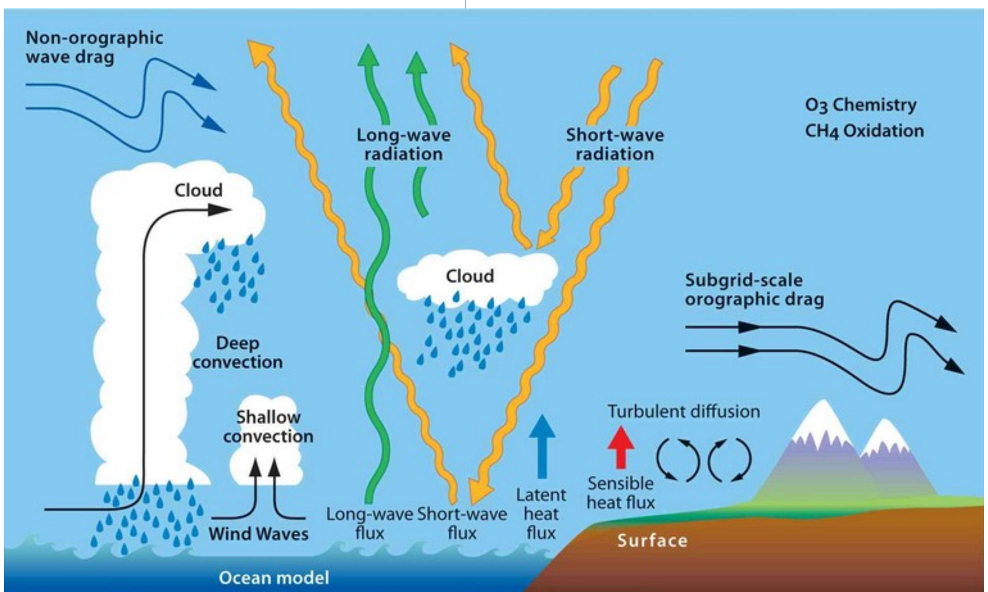
Forward model (oversimplified!)

Top of the Atmosphere: TOA

Observations: Brightness temperature (TB)



atm: Temp., Hum., Press., etc.



...

A **forward** model connects the unknowns with the observations:

TB = **F**(*f* ,θ , *b*, *atm* , *NSWS*)

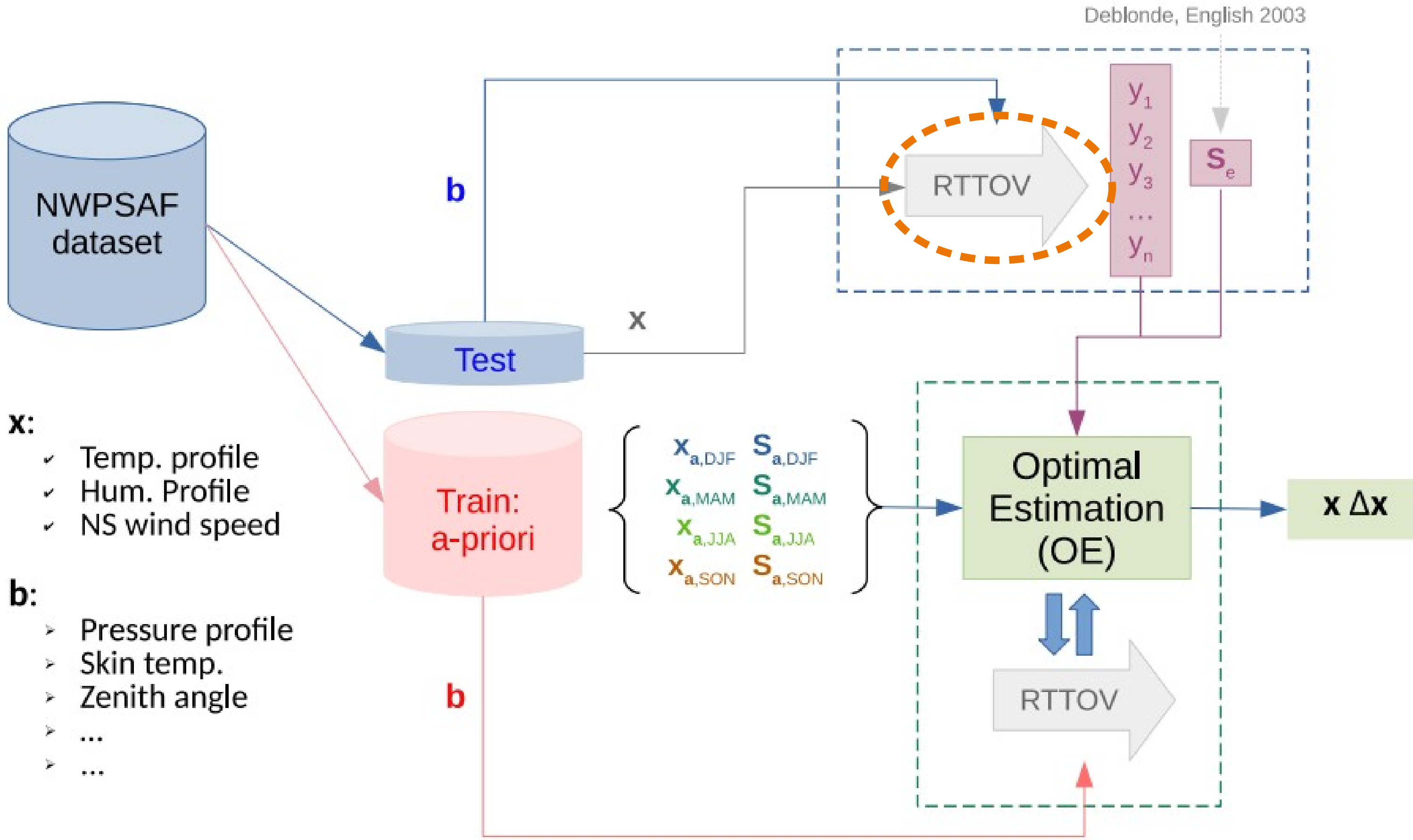
Surface of the ocean

Surface unknowns: Wind speed at 10m height

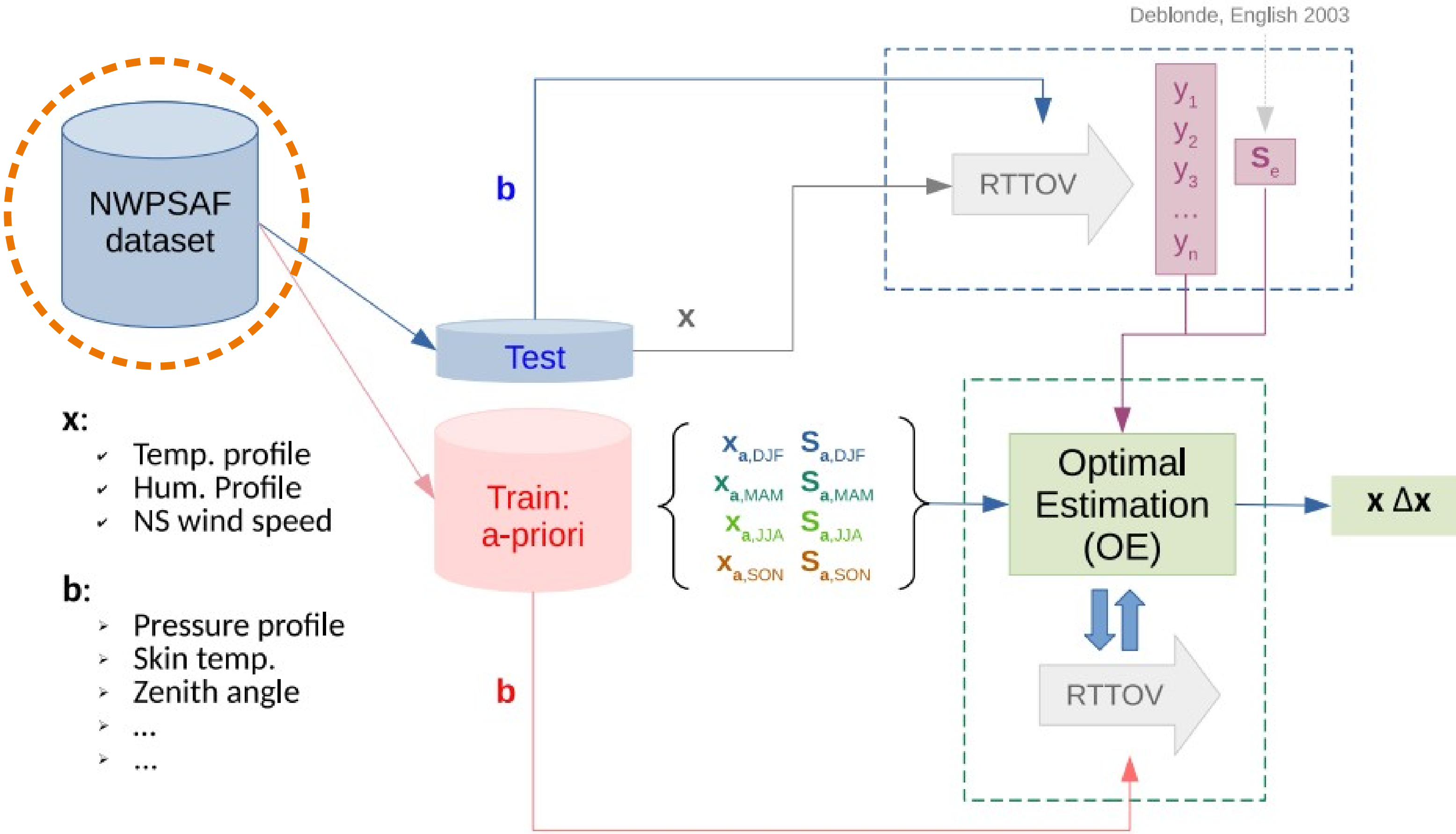
NSWS: Near Surface Wind Speed

*From: https://www.ecmwf.int/en/research/modelling-and-prediction/atmospheric-physics

Our working example: K-Fold Cross validation



Our working example: K-Fold Cross validation



x:
- Temp. profile
- Hum. Profile
- NS wind speed

b:
- Pressure profile
- Skin temp.
- Zenith angle
- …
- …

A-priori data: “Diverse profile datasets from the ECMWF-137-level short-range forecasts”

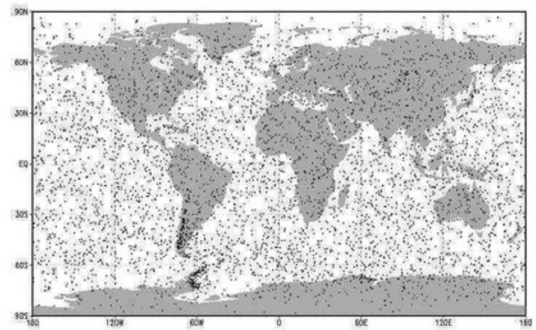
From [2]

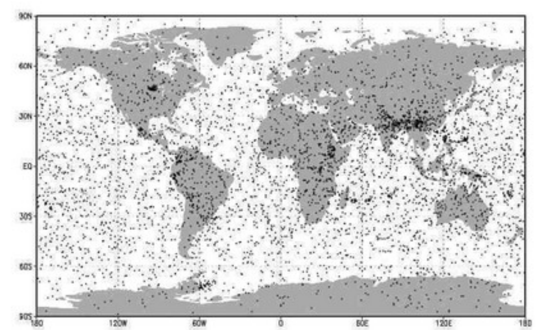
Atmospheric variables (given on model levels)

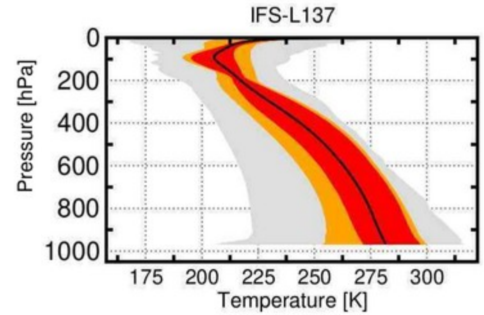
Variable name	Unit
Temperature	K
Specific humidity	kg kg^{-1}
Ozone mixing ratio	kg kg^{-1}
Fractional cloud cover	
Cloud liquid water content	kg kg^{-1}
Cloud ice water content	kg kg^{-1}
Rain rate	kg m^{-2} s^{-1}
Snow rate	kg m^{-2} s^{-1}
Vertical velocity	Pa s^{-1}

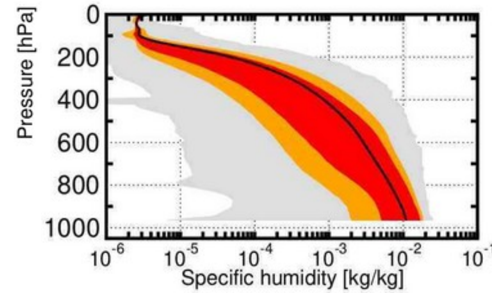
Surface variables

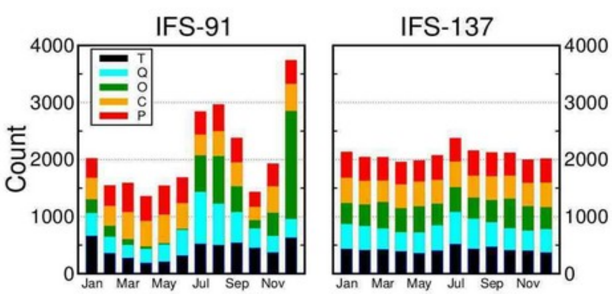
Variable name	Unit
Logarithm of surface pressure	Pa
Surface geopotential	m2 s^{-2}
Surface skin temperature	K
2-meter temperature	K
2-meter dew point temperature	K
10-meter wind speed U component	m s^{-1}
10-meter wind speed V component	m s^{-1}
Stratiform precipitation at surface	m
Convective precipitation at surface	m







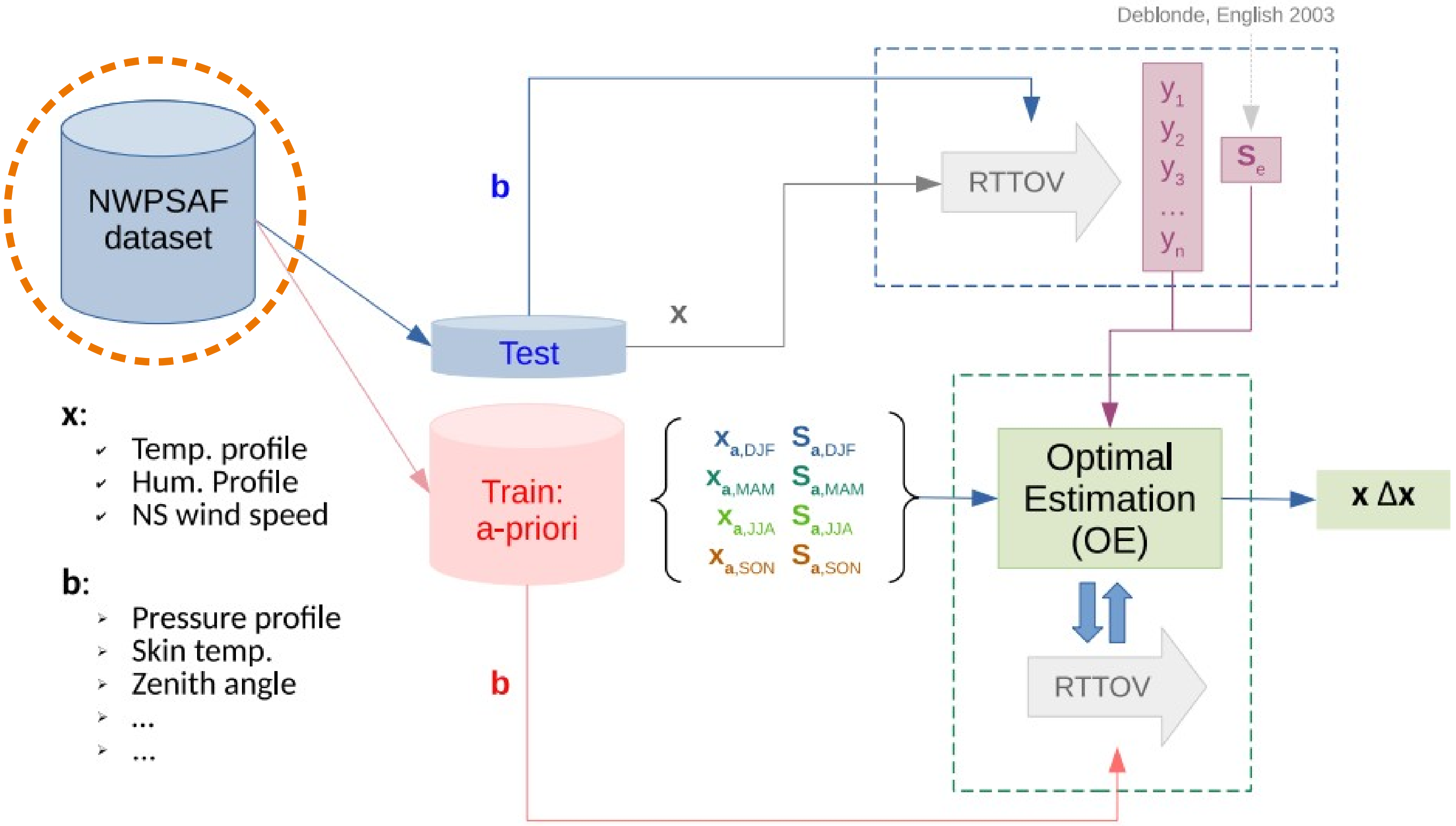




Black line: mean
Gray: min/max
Orange: 10th-90th percentiles
Red: 25th-75th percentiles

Our working example: K-Fold Cross validation

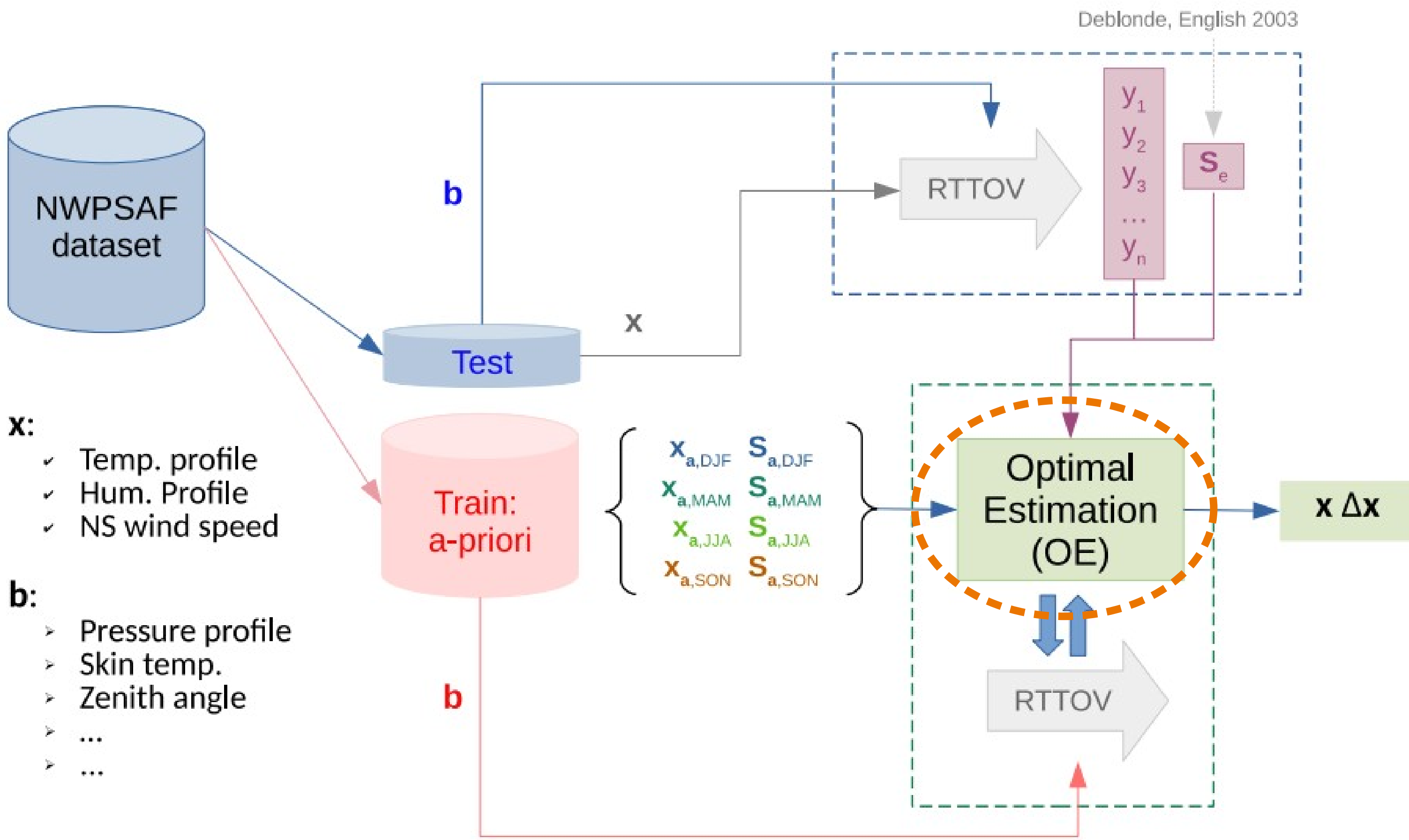
jupyter xarray pandas matplotlib NumPy Cartopy scikit learn



x:
- Temp. profile
- Hum. Profile
- NS wind speed

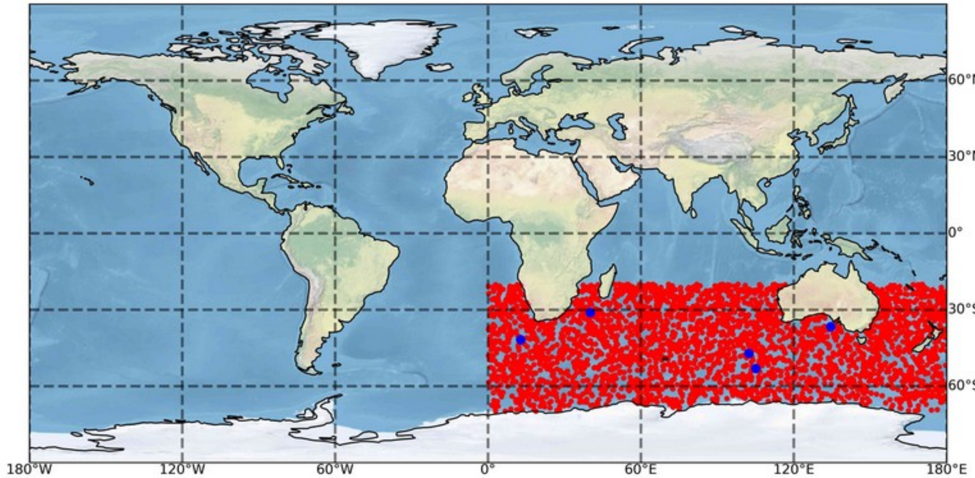
b:
- Pressure profile
- Skin temp.
- Zenith angle
- …
- …

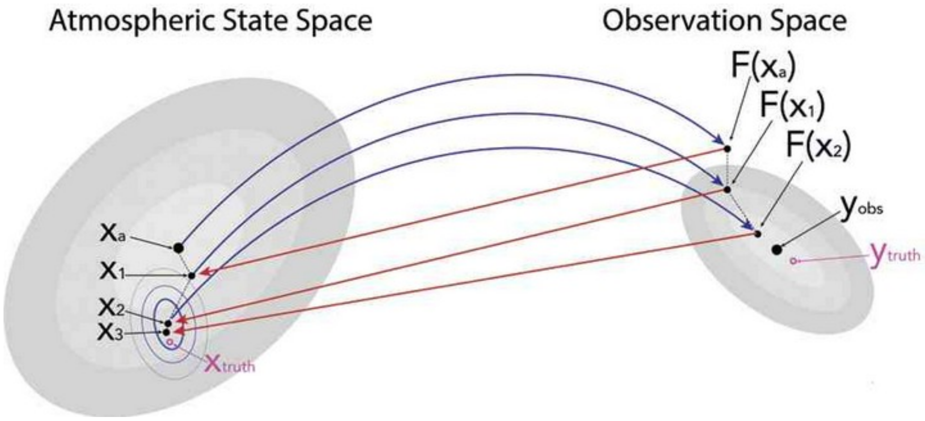
Our working example: K-Fold Cross validation

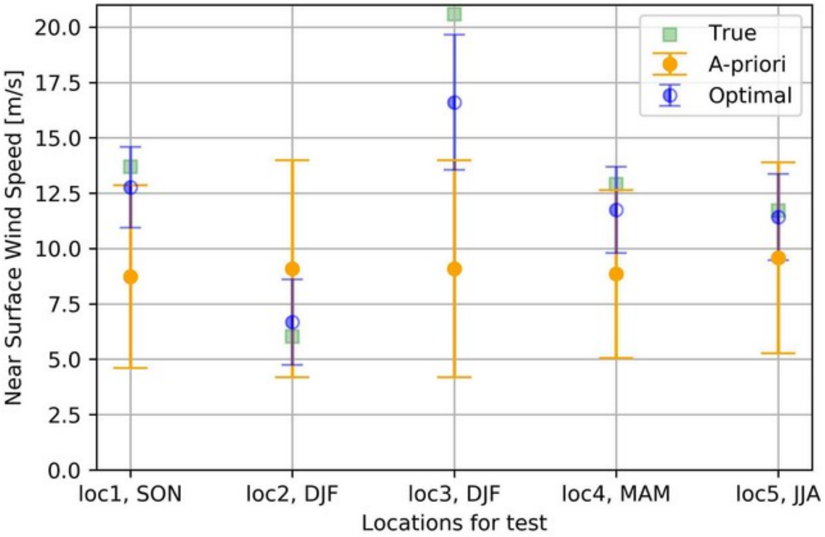


Testing with synthetic data: some inputs

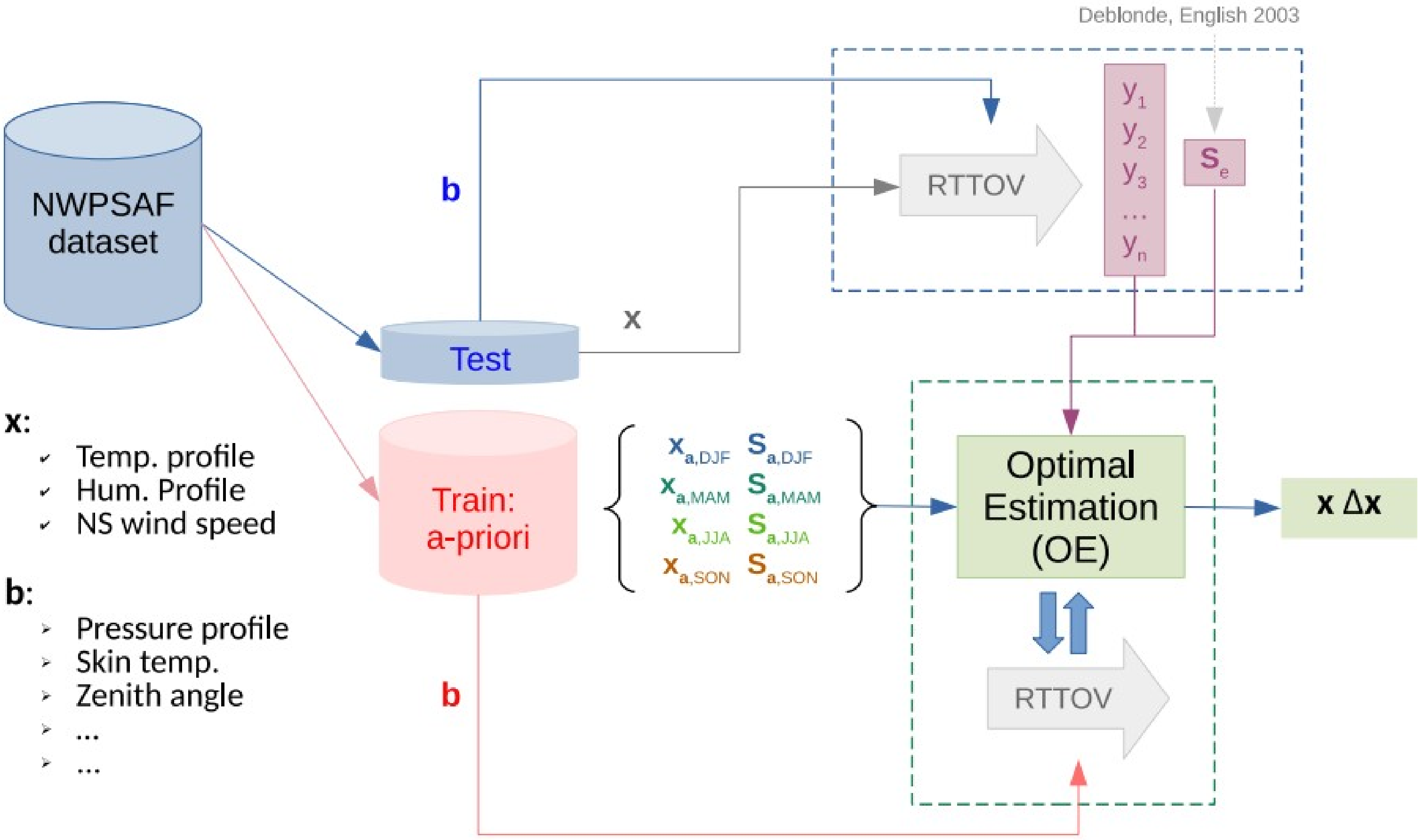
Locations for synthetic experiment:



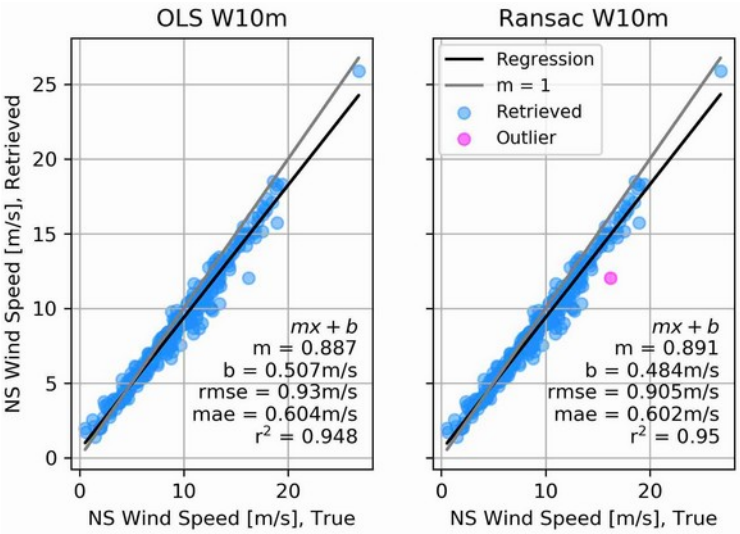


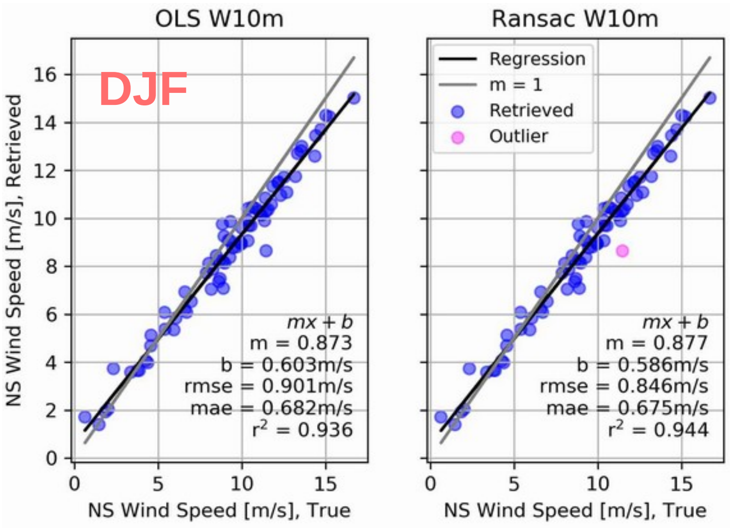


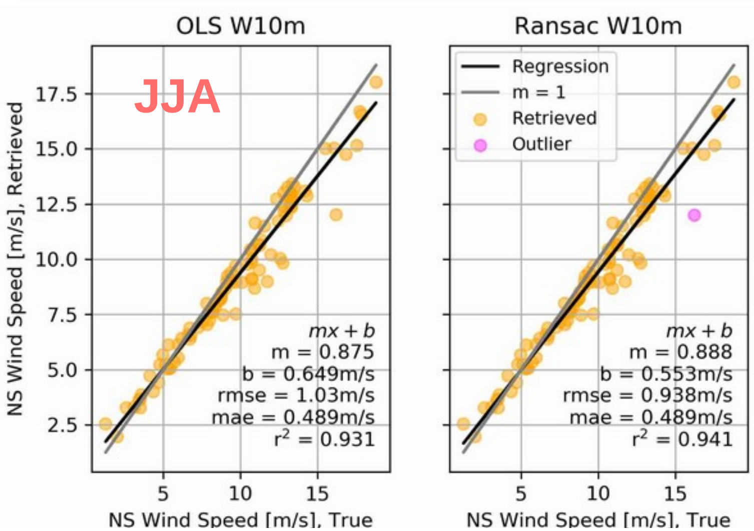
Our working example: K-Fold Cross validation

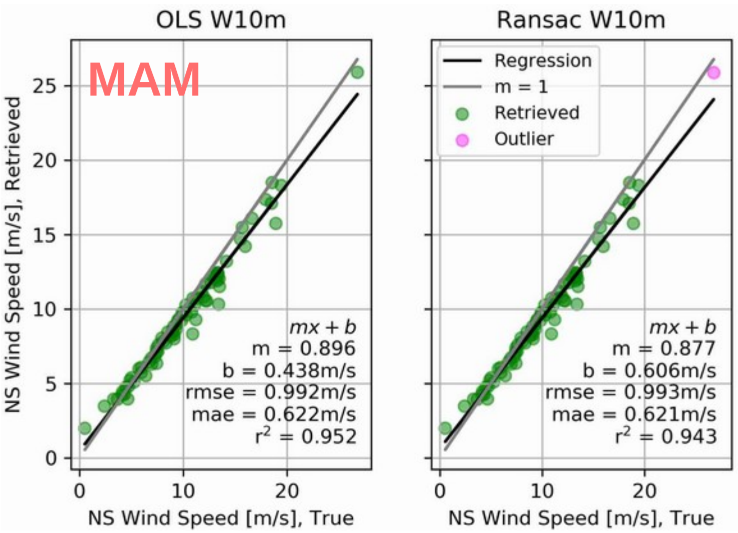


SSMIS F16:
slope mean 0.878, std 0.006





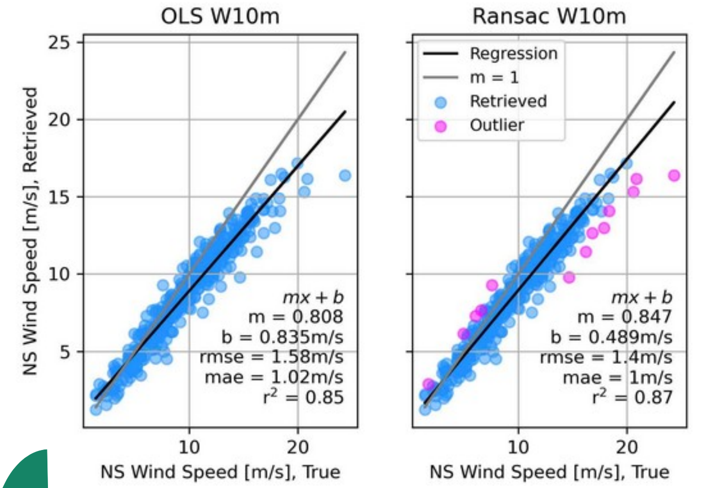




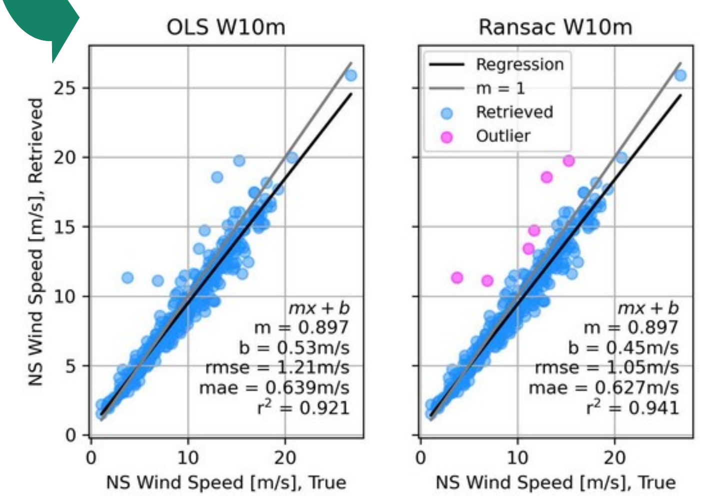
So?

- **What can we do?**
  - e.g. Sandbox for hyperparameters tunning:
    - RTTOV parameters
    - Constraining variables
    - Channels
  - e.g. Platform for prototyping and communication:
    - Scripting like environment (Jupyter) offers a lot of flexibility
    - Deployment (via Conda environments for example)
    - Python's prevalence and support
  - Open Source & Community development
    - Code available: check it, propose & improve:

https://github.com/deweatherman/RadEst



RTTOV/FASTEM bug detected!



References

[1] M. Maahn et al, “Optimal Estimation Retrievals and Their Uncertainties What Every Atmospheric Scientist Should Know”, BAMS, Vol. 101, Issue 9, Sept. 2020, https://doi.org/10.1175/BAMS-D-19-0027.1

[2] R. Eresmaa and A. P. McNally, “Diverse profile datasets from the ECMWF 137-level short-range forecasts” European Centre for Medium-range Weather Forecasts, 2014. https://nwp-saf.eumetsat.int/site/download/documentation/rtm/nwpsaf-ec-tr-017.pdf

[3] RTTOV Documentation, https://nwp-saf.eumetsat.int/site/software/rttov/documentation/

[4] Deblonde, English, “One-Dimensional Variational Retrievals from SSMIS-Simulated Observations”, AMS, Vol. 42, pp 1406-1420, March 2003. https://doi.org/10.1175/1520-0450(2003)042<1406:OVRFSO>2.0.CO;2

[5] J. Hamman et al, “Pangeo ML - Open Source Tools and Pipelines for Scalable Machine Learning Using NASA Earth Observation Data”, accepted proposal, Advancing Collaborative Connections for Earth System Science (ACCESS) Program, NASA, 2019.

References

ML and Geo packages:

ML:
- https://pypi.org/project/scikit-learn/
- https://pypi.org/project/keras/
- https://pypi.org/project/tensorflow/
- https://pypi.org/project/opencv-python/
- https://pypi.org/project/matplotlib/

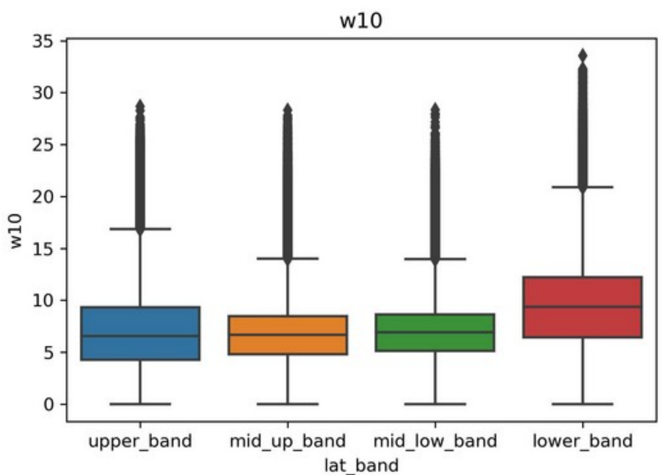
Geo:
- https://pangeo.io/
- https://pytroll.github.io/
- https://www.scipy.org/
- https://pypi.org/project/Cartopy/

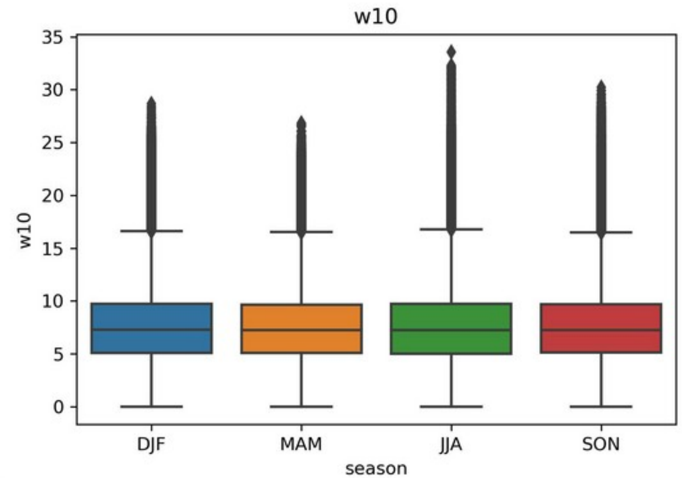
Optimal Estimation:
- https://github.com/maahn/pyOptimalEstimation
- https://github.com/deweatherman/RadEst

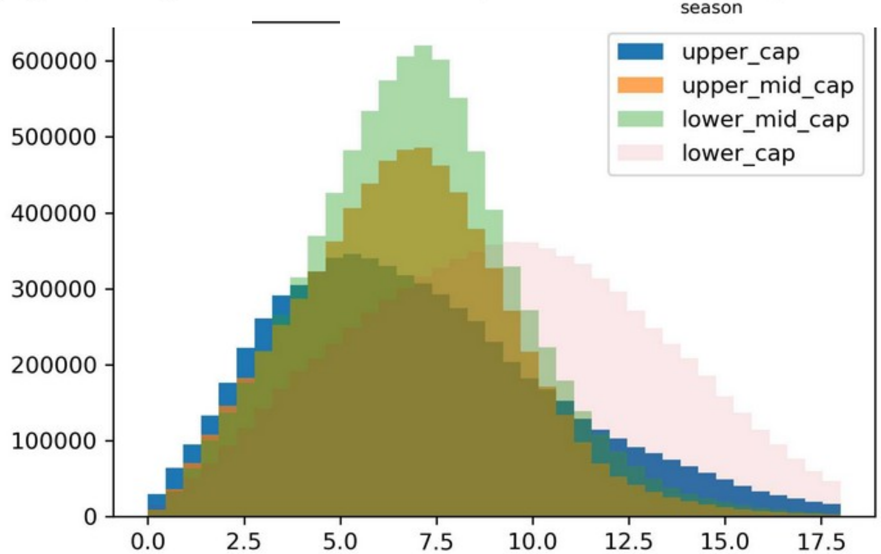
- Thanks!

Support slides

Exploratory Data Analysis and Visualization : ERA5 data

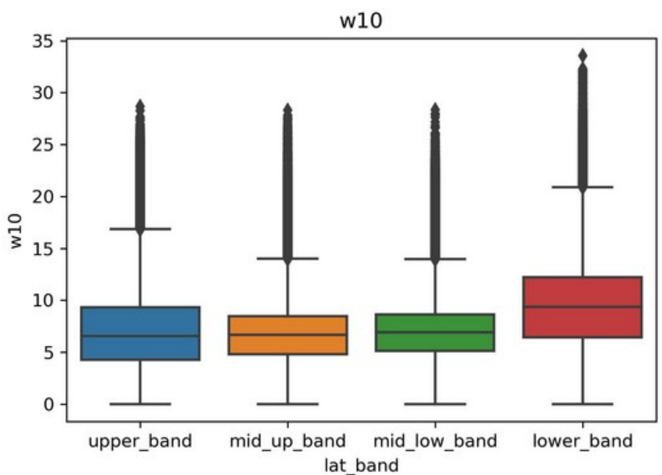


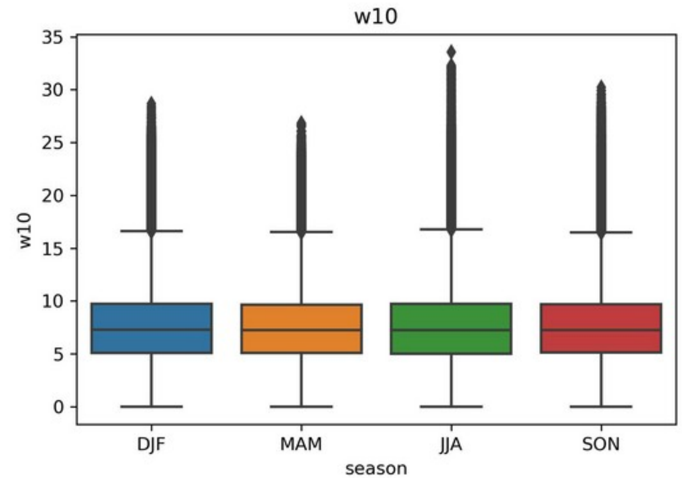




* https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=overview,
https://doi.org/10.24381/cds.adbb2d47

Exploratory Data Analysis and Visualization : ERA5 data

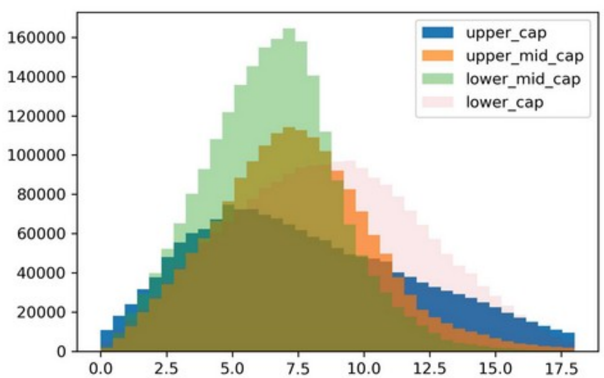




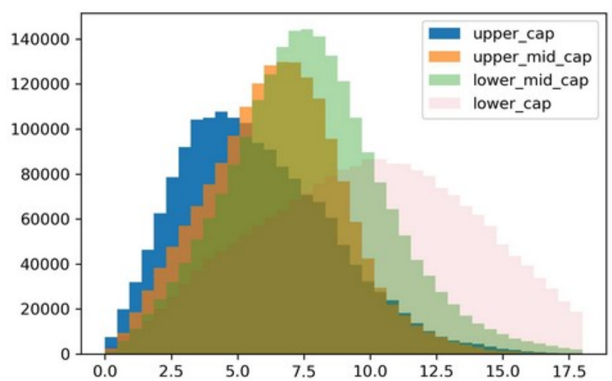
DASK

seaborn

DJF



JJA



MAM