Machine learning validation

Evaluating ML models and avoiding leakage

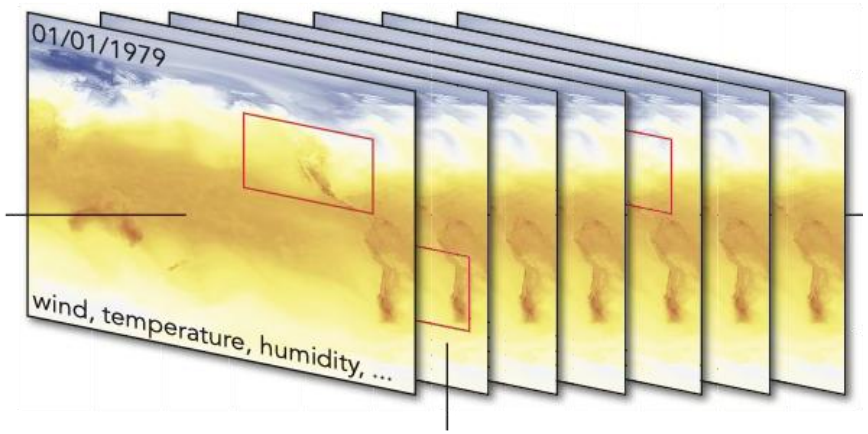
Ilaria Luise, Julian Kuehnert, Jesper Dramsch

ECWMF Bonn

ilaria.luise@ecmwf.int

ECMWF

Forecasting models

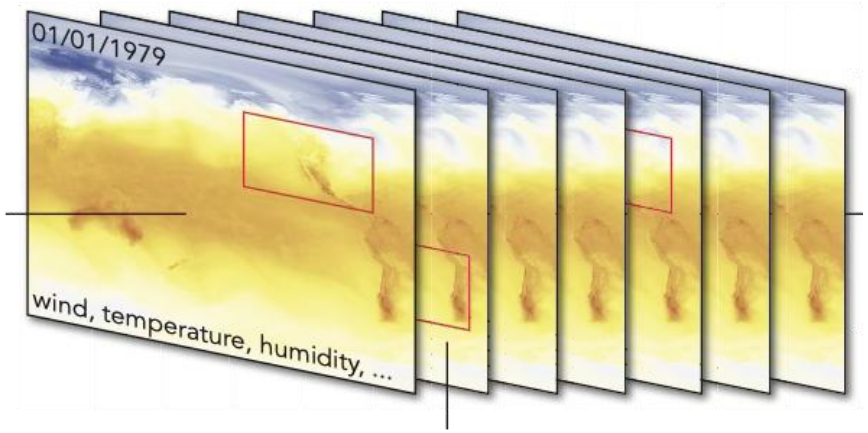


Training/Validation data

Model inference

Prediction

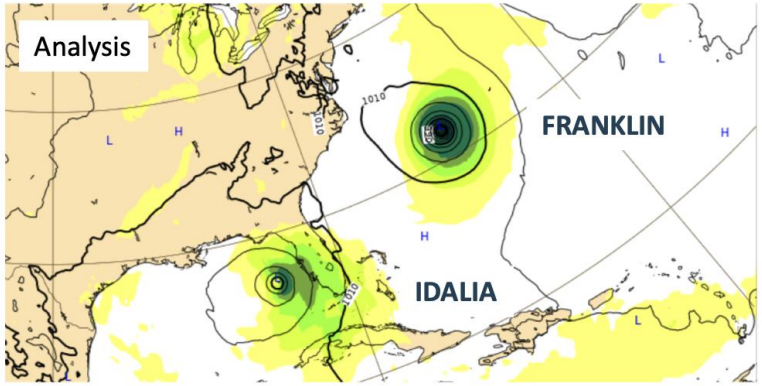
Forecasting models



Training/Validation data

Model inference

Prediction



New data

Model inference

Prediction (?)

The predictions

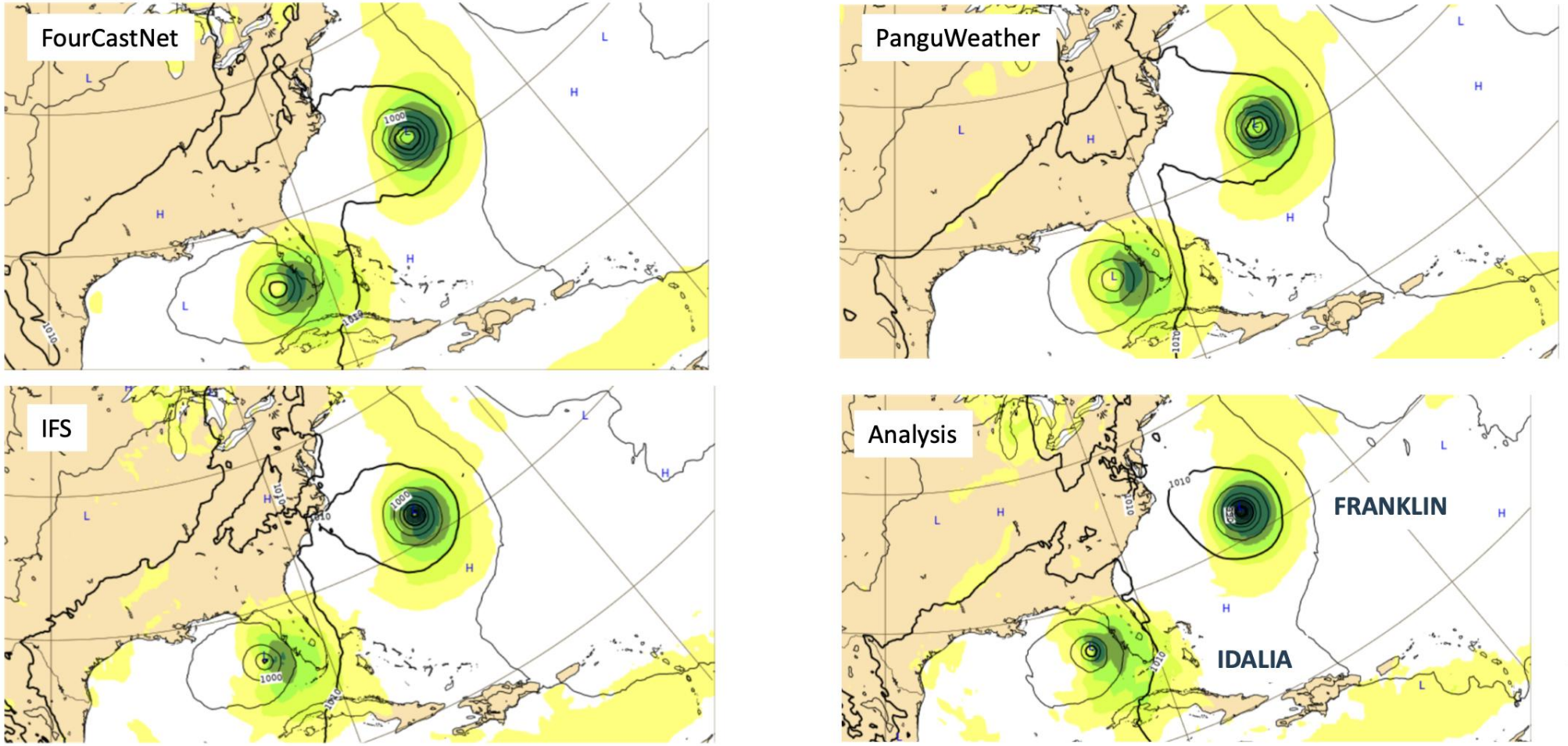


Figure from Zied Ben-Bouallegue, ECMWF

Introduction

How *generalizable* is the model?

- Choose the training dataset wisely
- Choose how to validate your predictions even more wisely
- Problems & mitigations

Part 1

How "good" is the model at *doing* the task(s) I want?

(Focus on Weather & climate modelling)

- Which tests can I put in place to probe my model?
- What should I pay attention to when putting in place my ML evaluation pipeline?
- Is my model learning physics?

Part 2

How do we ensure our **models work** on **unseen data** in the future?

Basic Validation Strategies

Example

Will it rain tomorrow? Yes/No

Suppose we have **5 years of daily weather data (2018–2022)**

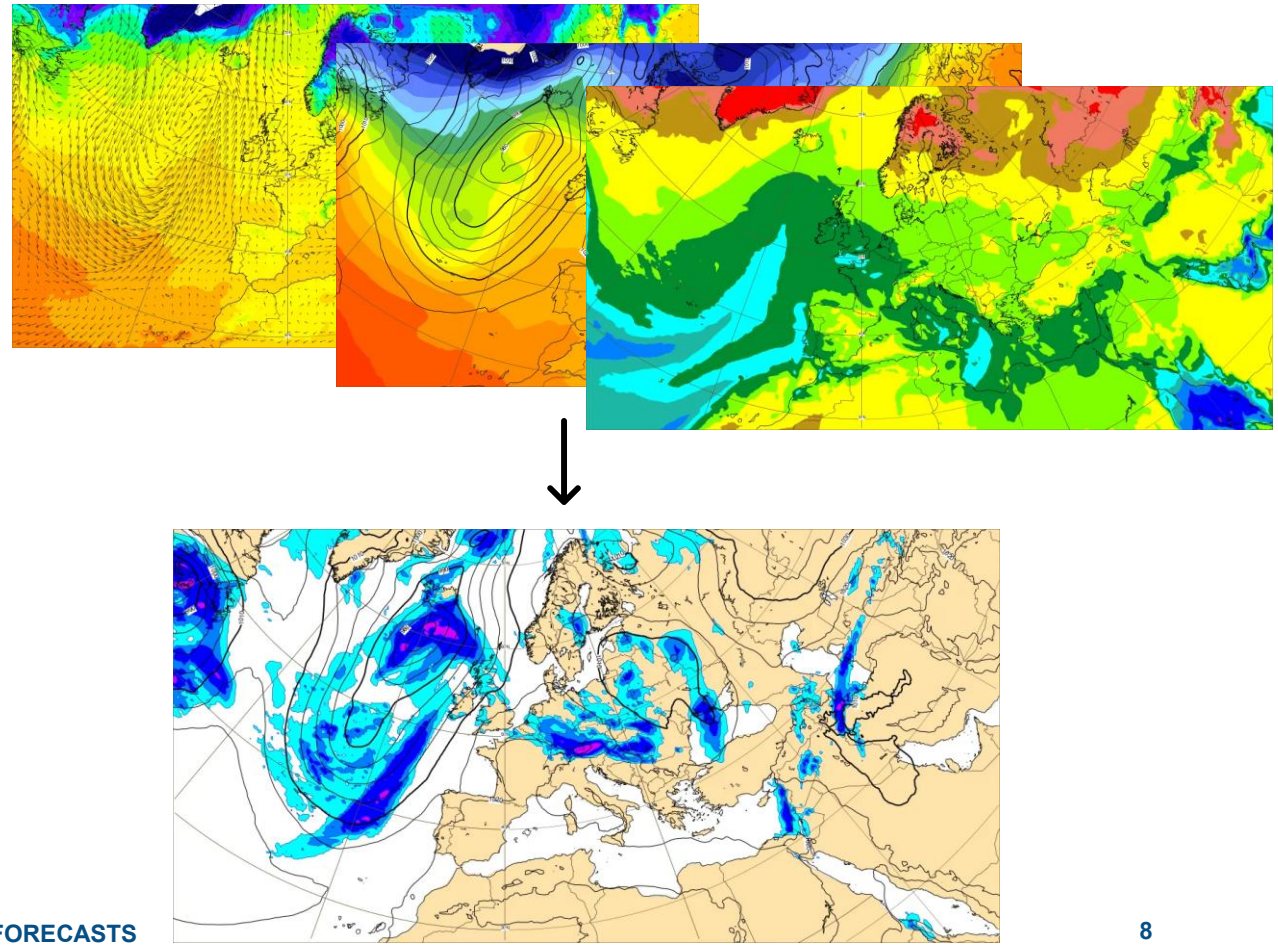
For each day, we collect:

- Temperature today
- Humidity
- Air pressure
- Wind speed

- Rain today (Yes/No) → Labels

Target variable:

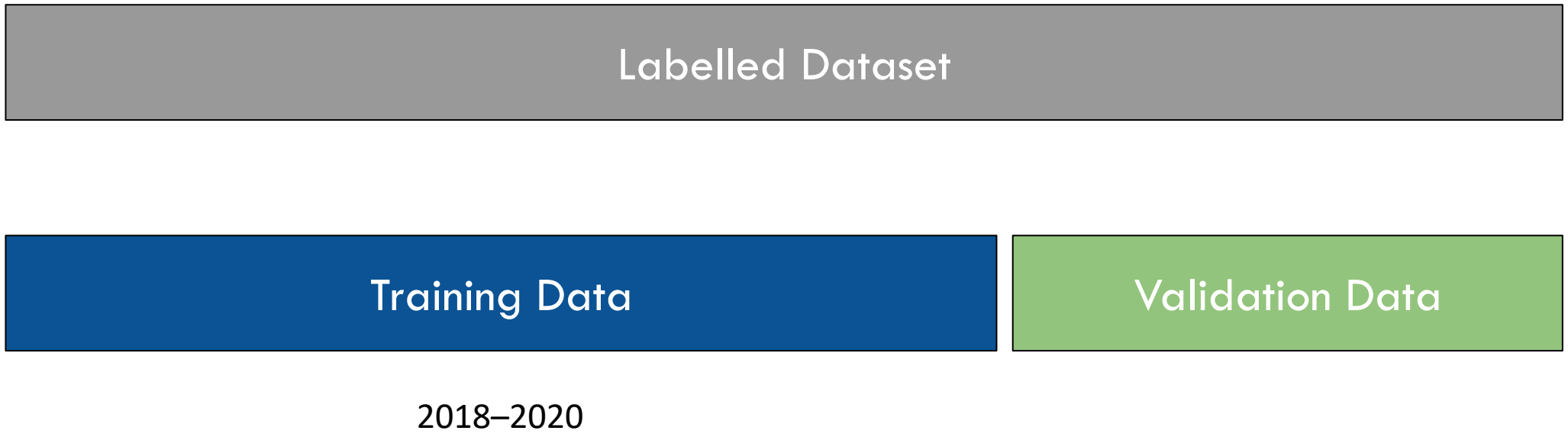
- Rain tomorrow (Yes/No)



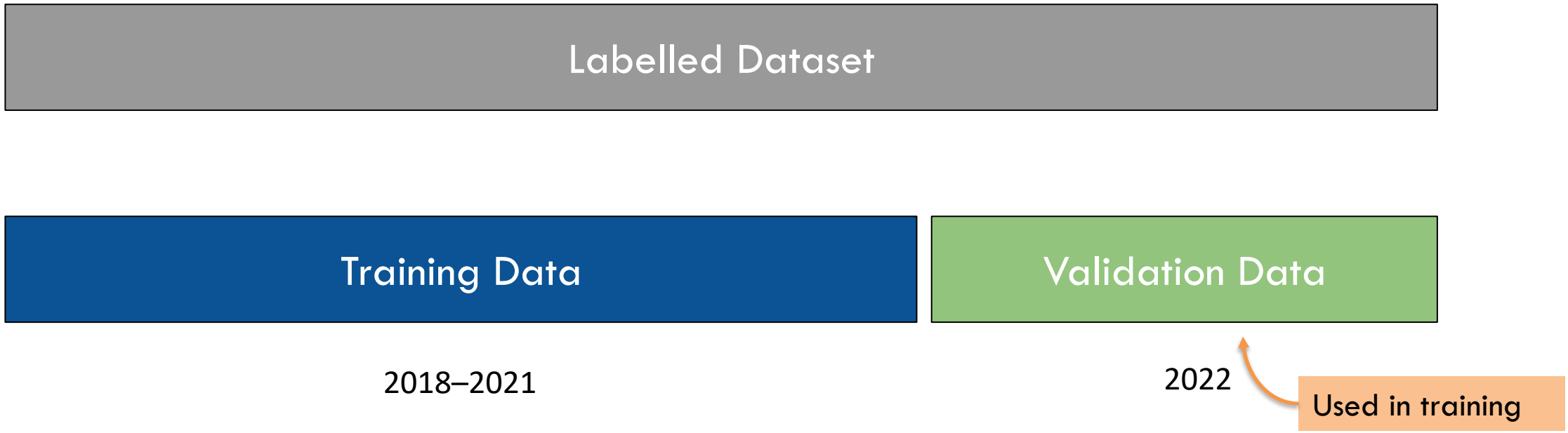
Obtaining Data to Test & Validate On

Labelled Dataset

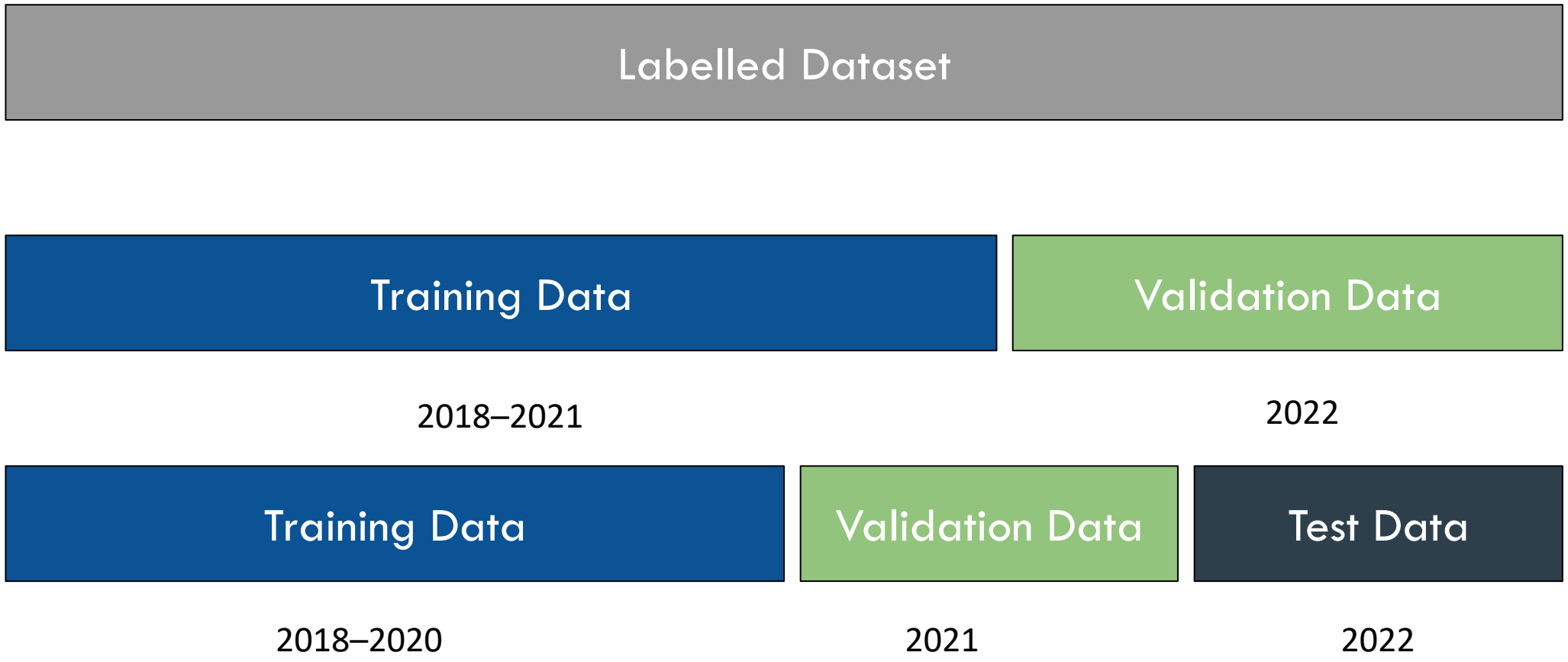
Obtaining Data to Test & Validate On



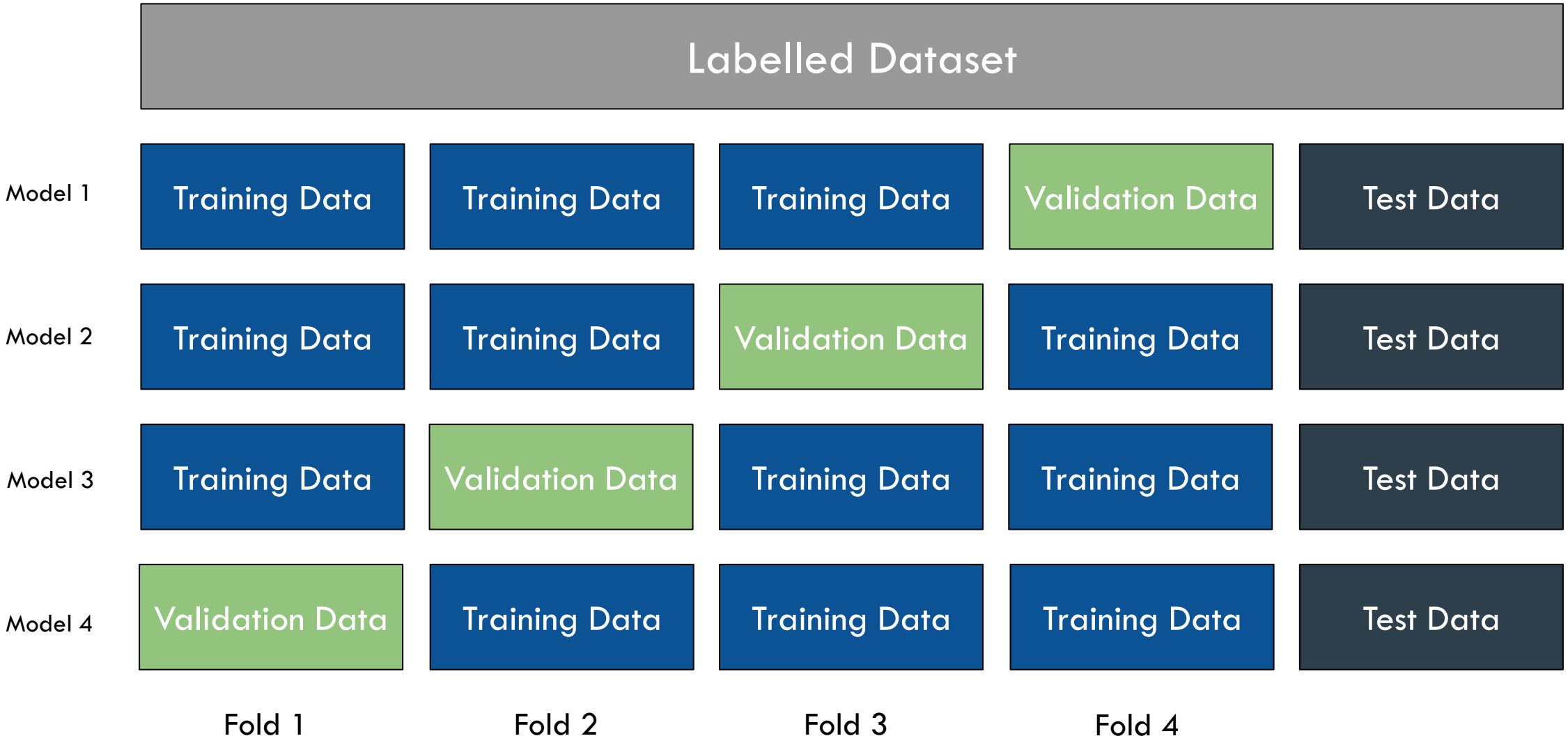
Obtaining Data to Test & Validate On



Obtaining Data to Test & Validate On

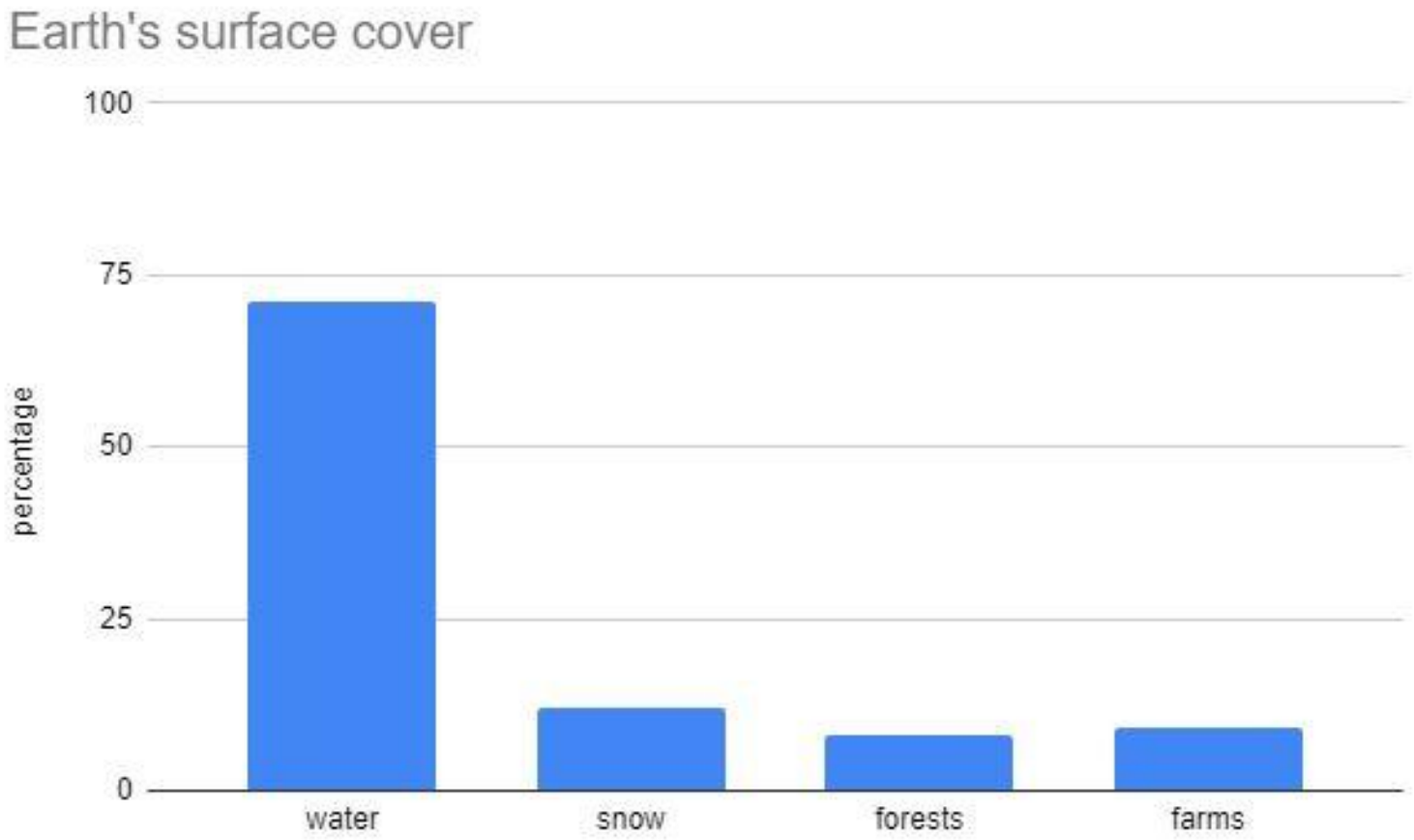


Cross-Validation

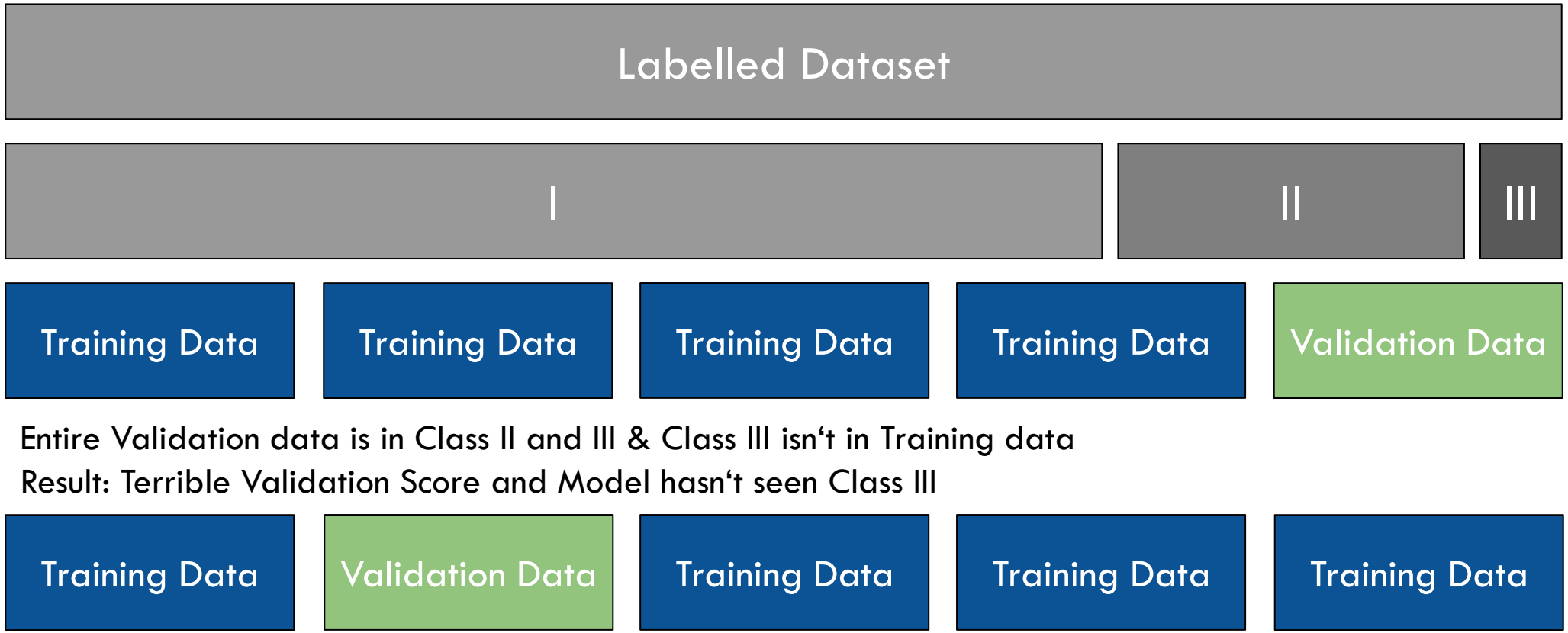


Imbalanced and Heterogeneous Data

Class Imbalance



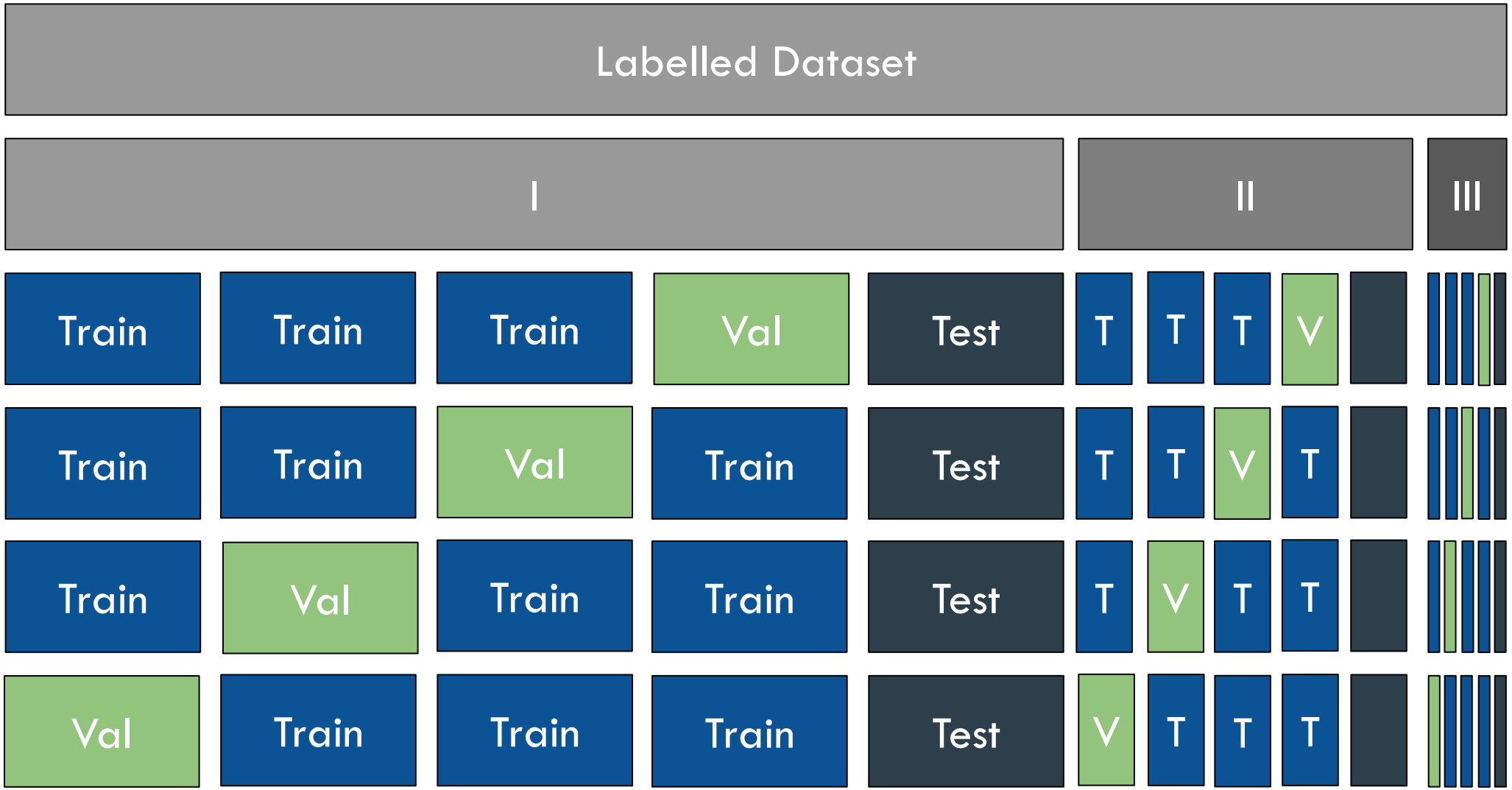
Why not use Random Sampling like before?



Entire Validation data is in Class II and III & Class III isn‘t in Training data
Result: Terrible Validation Score and Model hasn‘t seen Class III

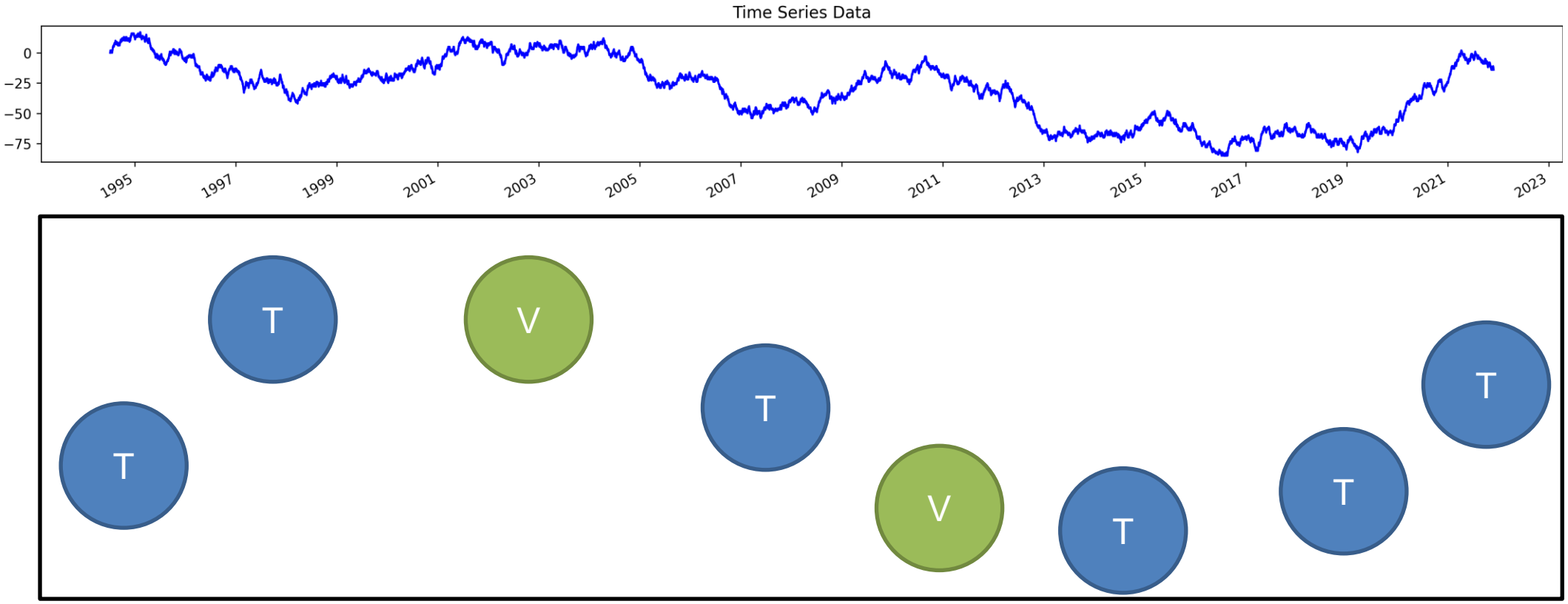
Entire Validation data is in Class I
Result: Great Validation Score but no validation of Class II & III at all

Stratification for Imbalanced Data



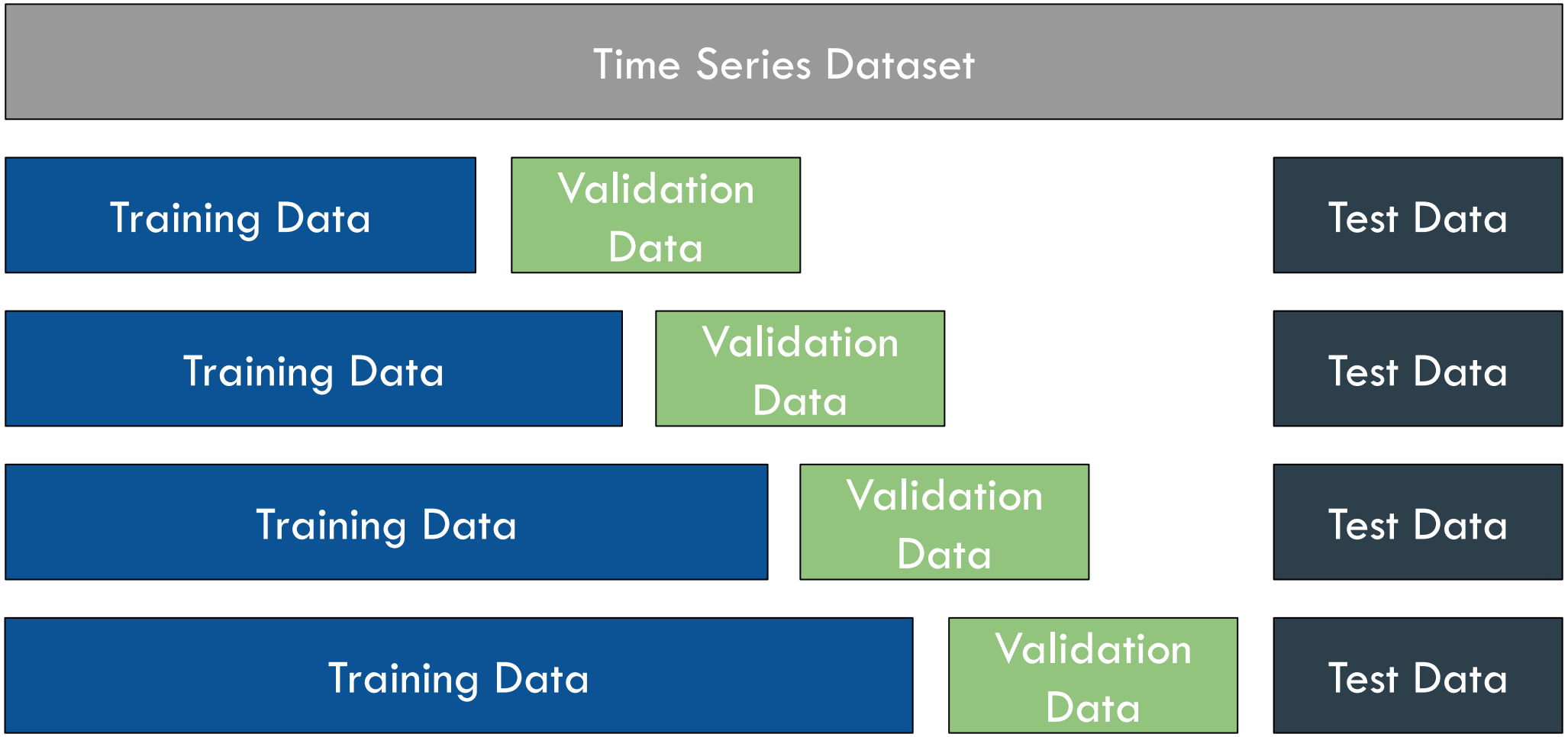
Correlated and Connected Data

Time Series Data



- ***Random Splits on Time Series Data equates to interpolation***
- **Bad on standard time series problems**
- **Devastating on forecasting problems**

Validation on Time Series Data



Choose *continuous & adjacent* training/validation datasets (no random sampling inside the training dataset)

Validation of Geospatial Data

- Geospatial Data Examples
  - Stations
  - Satellite Data
  - Weather Radar
- Geospatial Data is spatially correlated
- Problems with random split of data:
  - Clustering of Validation Locations
  - Overlap of Validation and Training Locations

    (solution: choose different time periods)

Validation of Geospatial Data

- Geospatial Data Examples
  - Stations
  - Satellite Data
  - Weather Radar
- Geospatial Data is spatially correlated
- Problems with random split of data:
  - Clustering of Validation Locations
  - Overlap of Validation and Training Locations

    (solution: choose different time periods)

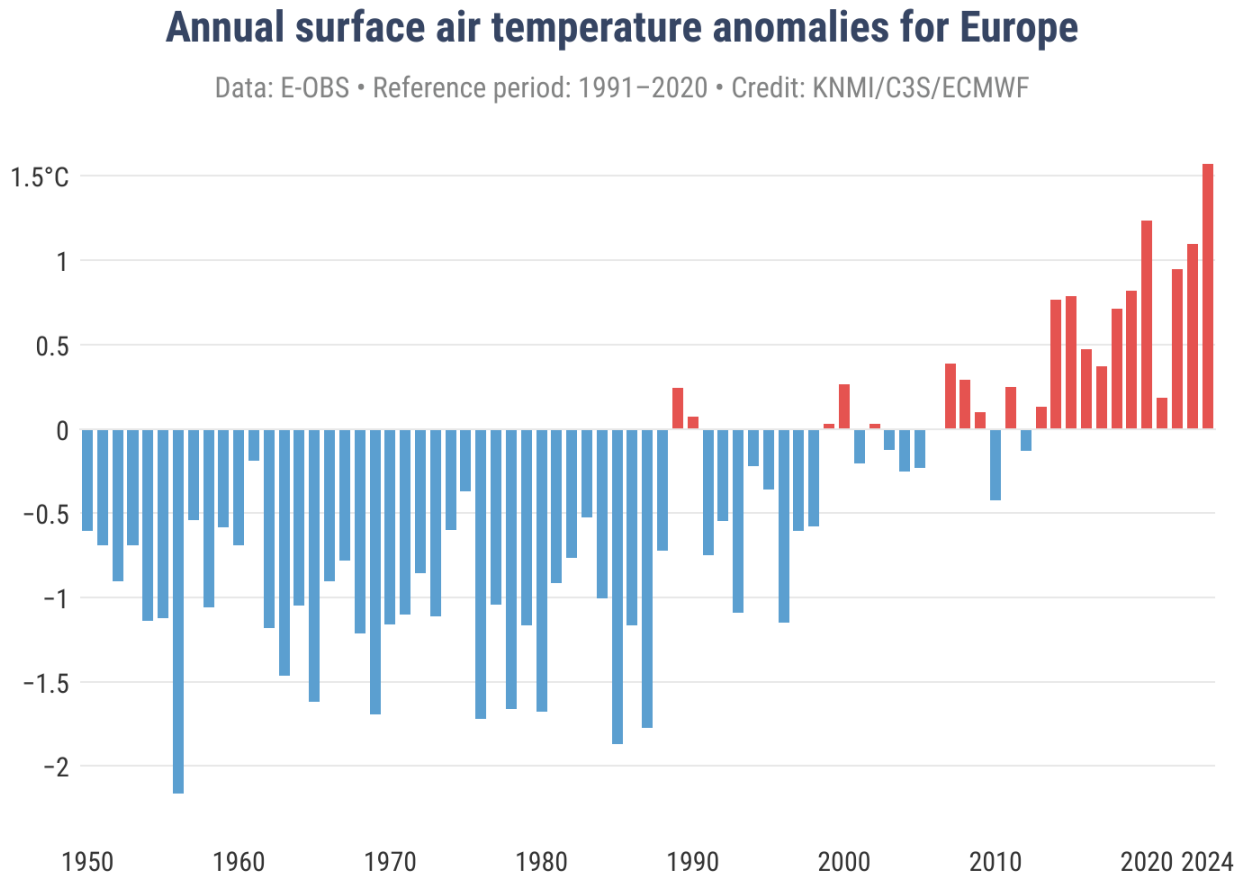
Problems & mitigations

Distribution Drifts & Data Leakages

Data Drifts

Shifts in Input Data distributions

- ***Examples***
  - Global Temperature through Climate Change
  - Land Cover Change through Urbanisation
- Mitigation Strategies
  - Monitoring of Input Data Distribution
    - Continuous, e.g. Kolmogorov-Smirnov test
    - Categorical, e.g. Chi-squared test
  - Automatic Retraining of ML Models
    - Define Threshold for Monitored Metrics
    - Implement periodic retraining

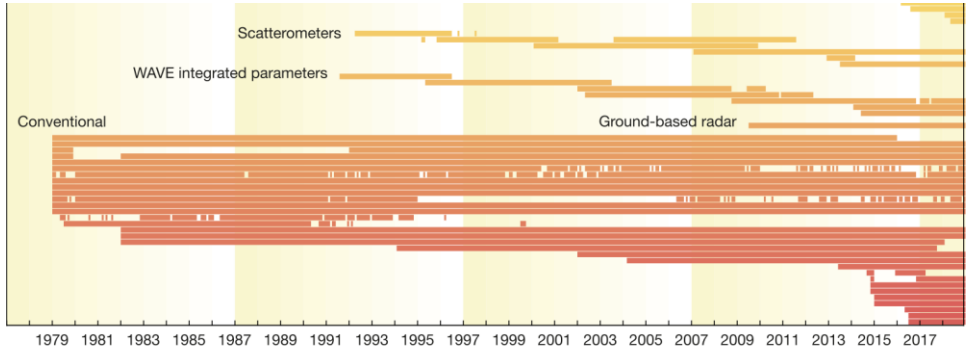


Target Drifts

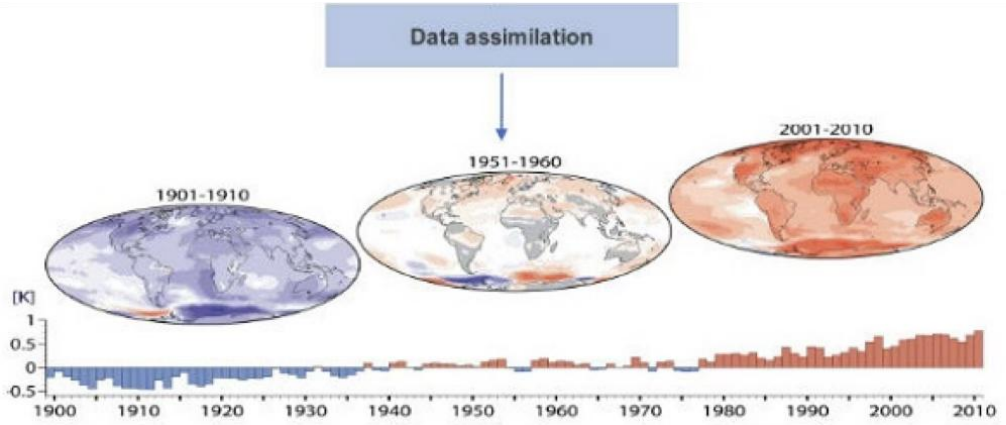
Shifts in the Target / Label Data distribution

Example:

Shift in ERA5 biases pre-and after the satellite era



- Mitigation Strategies
  - Monitoring of Output Data Distribution
  - Automatic Retraining of ML Models
  - Anticipate Class Changes if Probable
    - Set Up Pipelines for Label adaption
    - Make it Easy to Change Label Processing



Concept Drifts

Concept drift is a change in the input-output relationships an ML model has learned

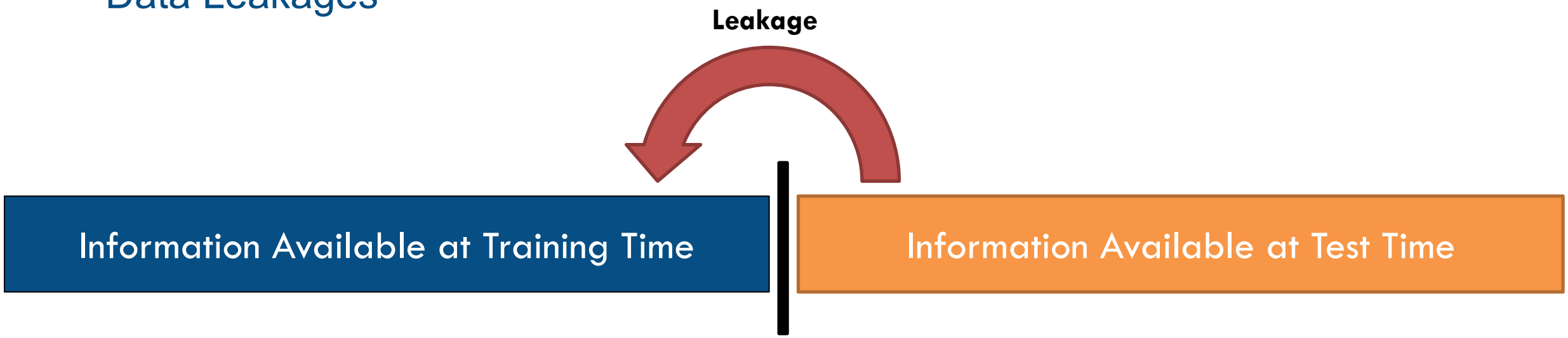
Example:

Tipping points that change “physics”

(a.k.a. the correlations across variables)

- Devastating for Machine Learning
- Mitigation Strategies
  - Monitor raw Model Metrics
  - Set up Alerts for Deterioration
  - Be prepared to take Model out of Production

Data Leakages



Results in overly optimistic performance metrics (high accuracy) during training but poor generalization to new, real-world data.

Examples:

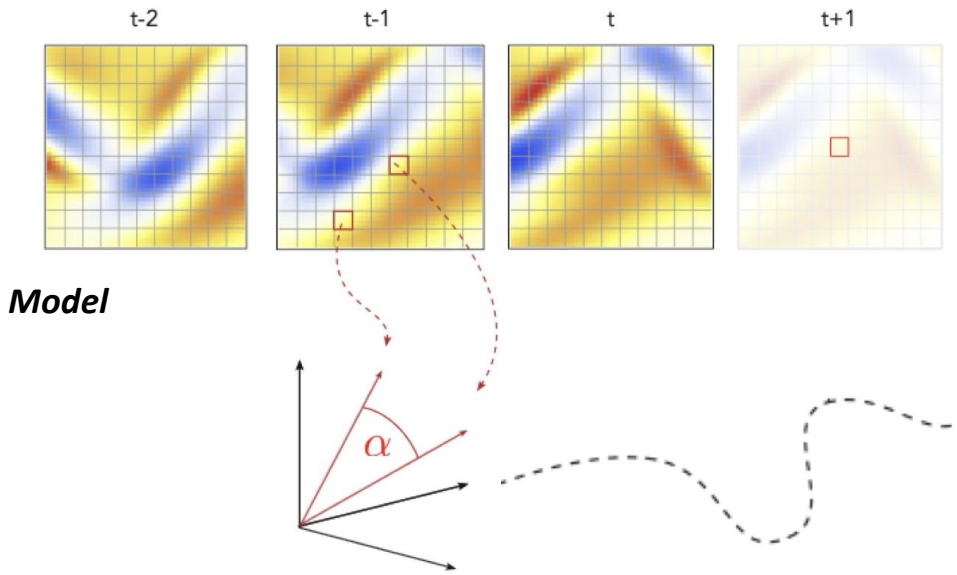
- **Target leakage**: Models include data that will not be available when the model is used to make predictions.
- **Train-Test Contamination:** Information from the validation or test set leaks into the training data during preprocessing steps like normalization, standardization, or imputation (e.g., scaling the entire dataset before splitting)

- **Solution:** process training and validation datasets independently. An independent test dataset help understanding Target Leakage

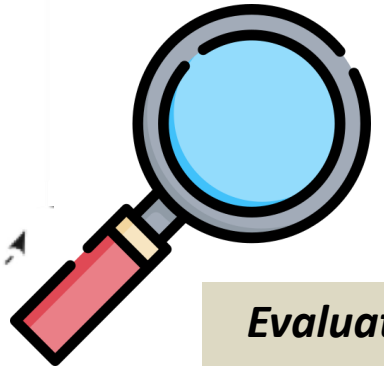
Part 2: How to Evaluate ML models in W&C

Evaluation

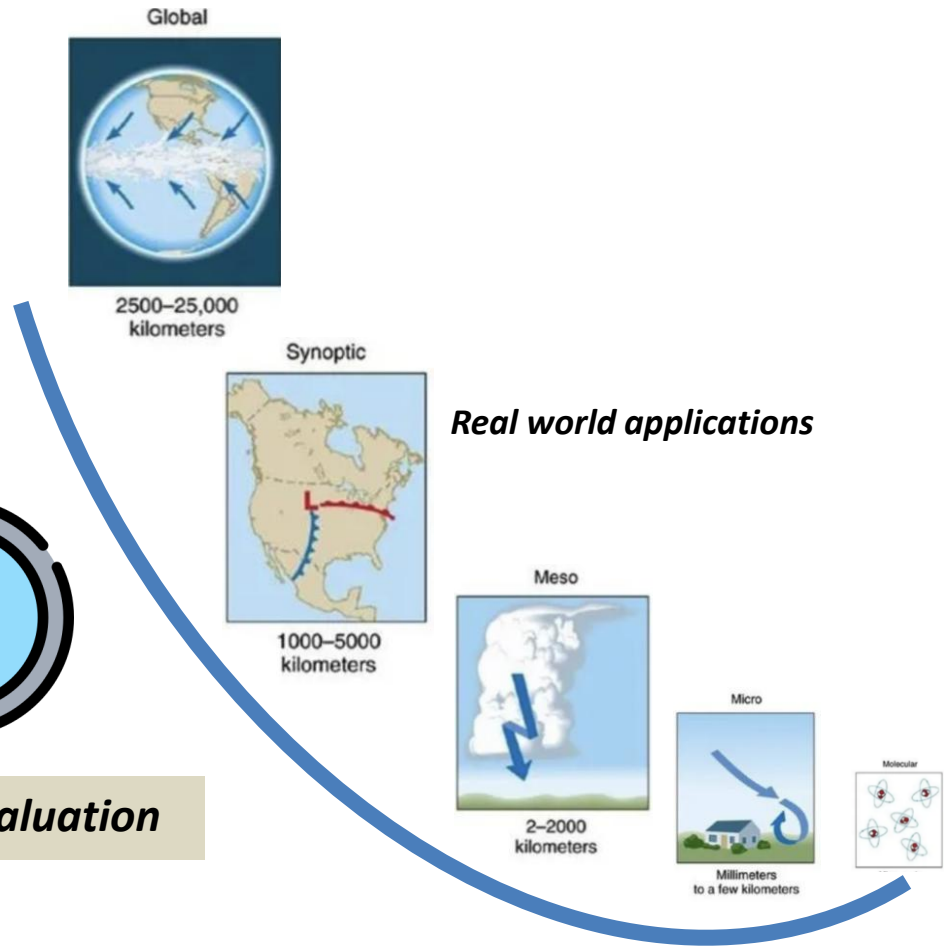
Data driven models encode the information in the feature space of the network



Model



Evaluation

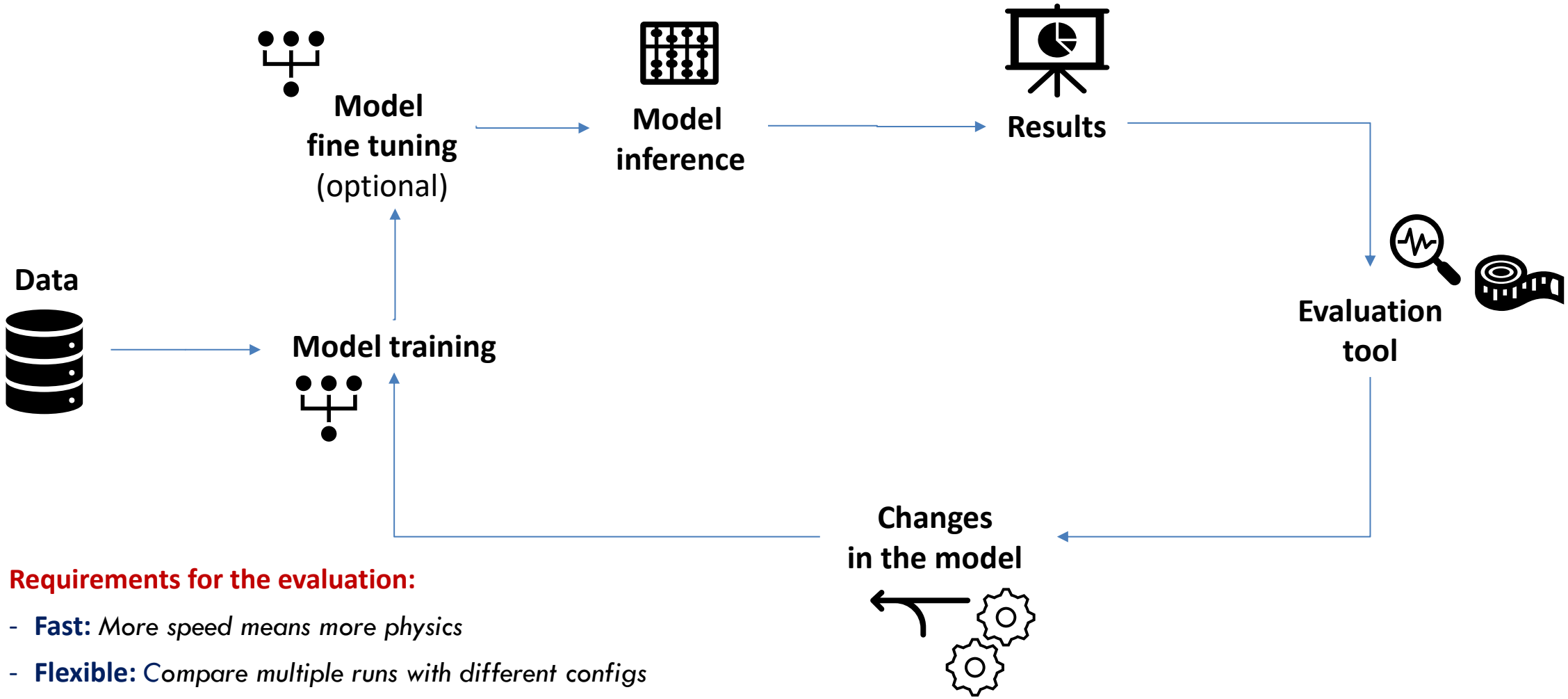


Real world applications

Set of procedures and tests that we put in place to judge whether the information encapsulated in a model is accurate, meaningful, and useful for real-world tasks.

ECMWF

The R&D cycle in a nutshell



Requirements for the evaluation:

- **Fast:** *More speed means more physics*
- **Flexible:** *Compare multiple runs with different configs*
- **Focused:** *Focus on key metrics and plots*

ECMWF

Two complementary types of evaluation

Short term evaluation

- ✓ **Fast, Flexible and Focused**

Requirements:

- ✓ **Easy way to compare multiple runs**
- ✓ **Main scores and plotting**

Strategy:

implementation as close as possible to model output

Long term evaluation

- ✓ **Extensive analyses**
  - e.g. power spectra
  - e.g. analysis against observations
- ✓ **Case studies**
- ✓ **Physical consistency**
- ✓ **Explainable AI**

Input

NetCDF/GRIB

Strategy:

Use existing tools

+

write packages of ***converters*** to the different grib or netCDF formats used by these tools

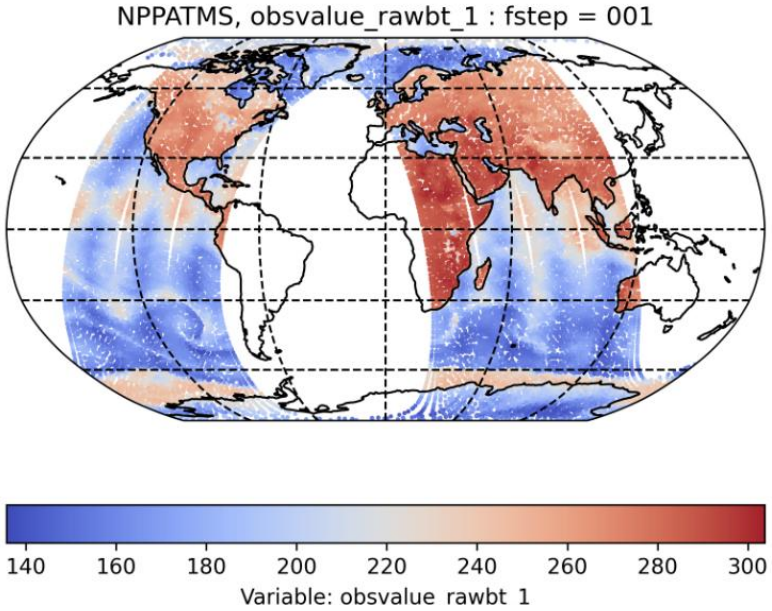
ECMWF

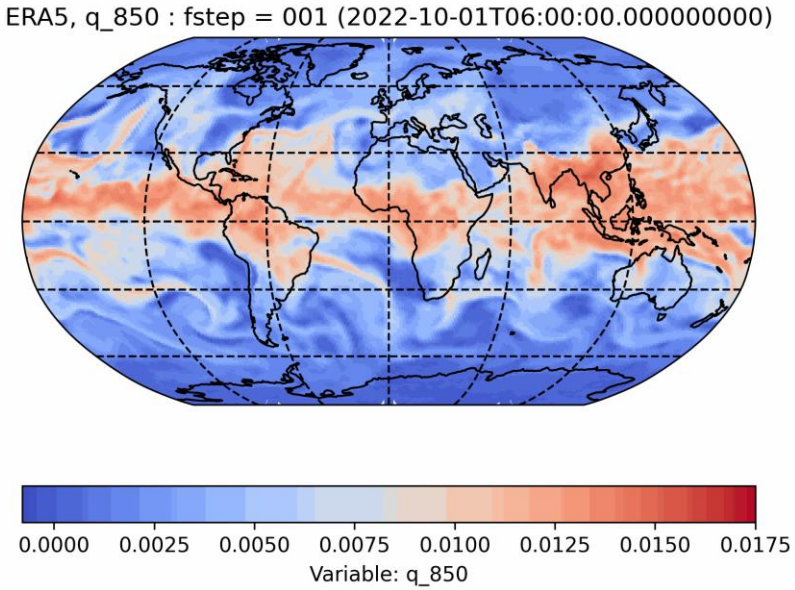
What should we look at?

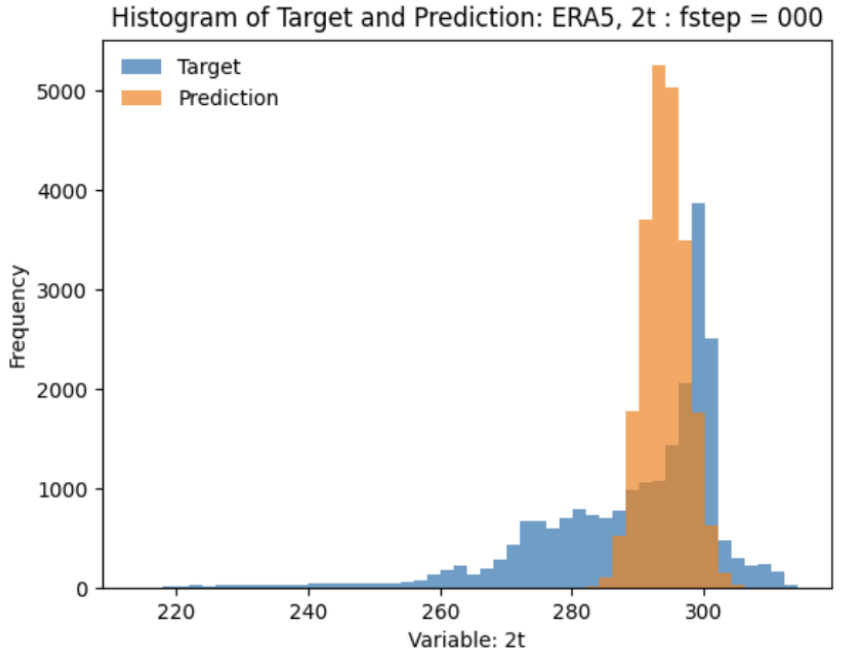
Plotting

Motivation

- Spot model artifacts → **Maps & Histograms**
- Inspect spatial structure and field quality metrics
- Time evolution → **Videos**





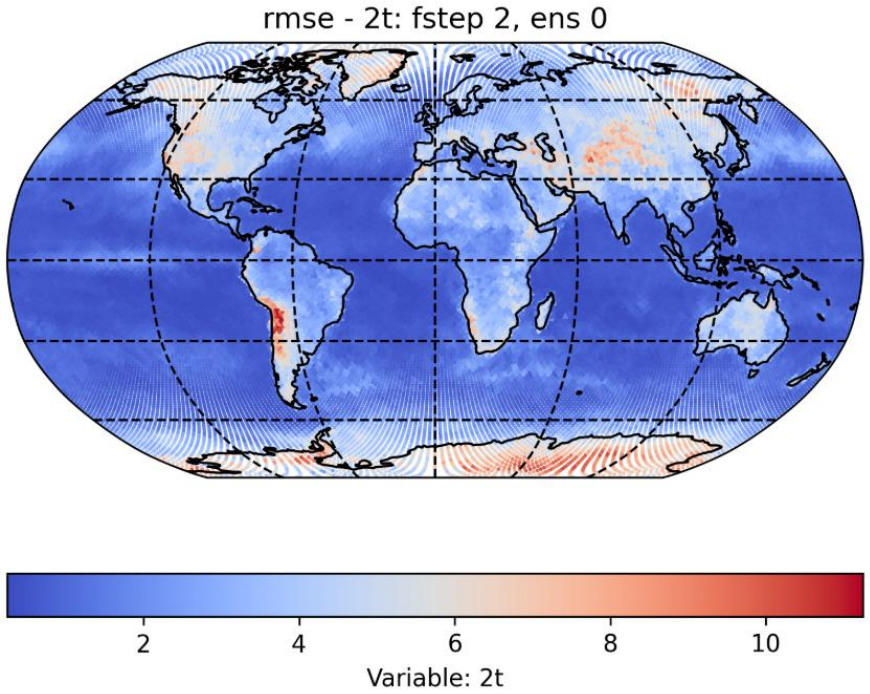


What should we look at?

Scoring

Motivation

- **Error quantification** → Lower error growth = better learned dynamics
  - **RMSE:** Lower RMSE = better large-scale dynamics understanding
  - **ACC:** Slower ACC decay over lead time = better temporal representation



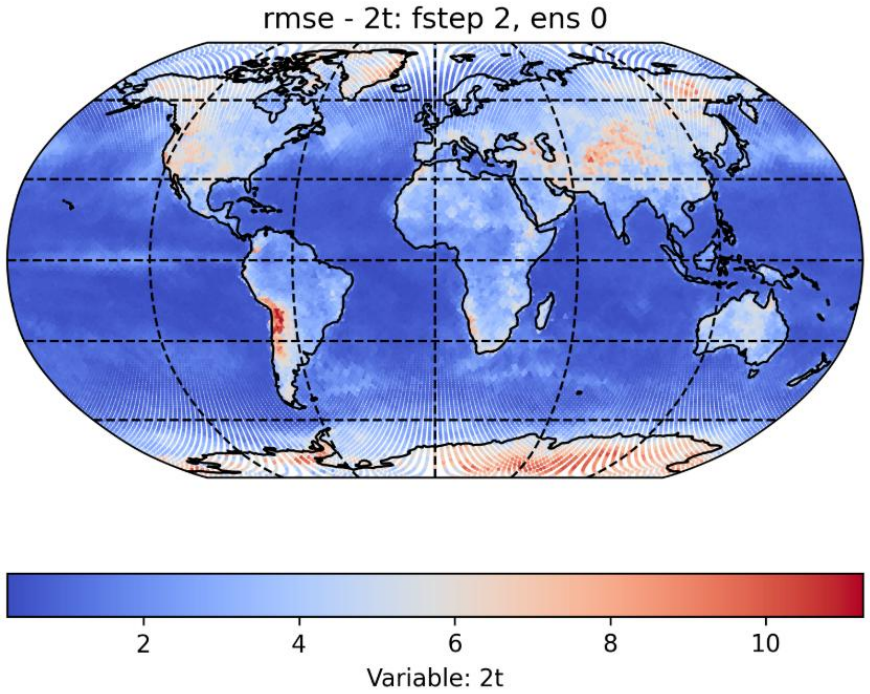
ECMWF

What should we look at?

Scoring

Motivation

- **Error quantification** → Lower error growth = better learned dynamics
  - **RMSE:** Lower RMSE = better large-scale dynamics understanding
  - **ACC:** Slower ACC decay over lead time = better temporal representation



Why is it important?

- Condense performance info into single numbers
- Easier comparisons against other models

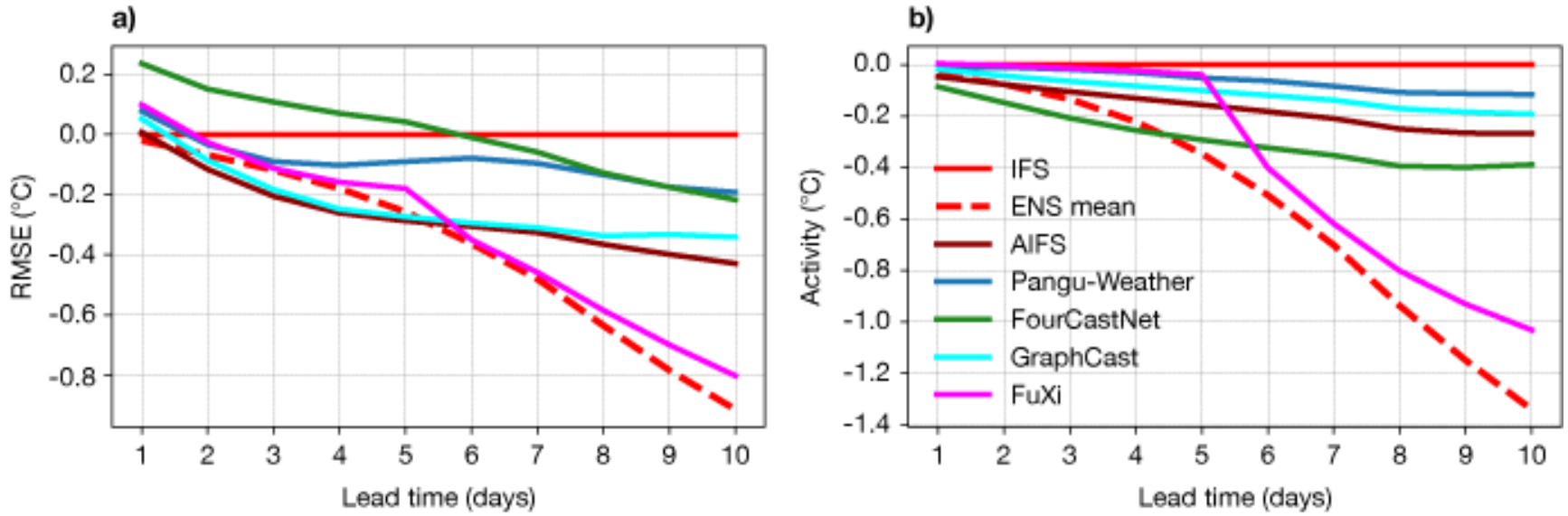
ECMWF

Which ingredients do we need?

Temporal Scoring

Motivation:

- **Time evolution** ➡ ead-time degradation curves
  - **Forecast Rate of change:** slower decay indicates stronger latent temporal modeling
  - **Activity:** deviation from climatology

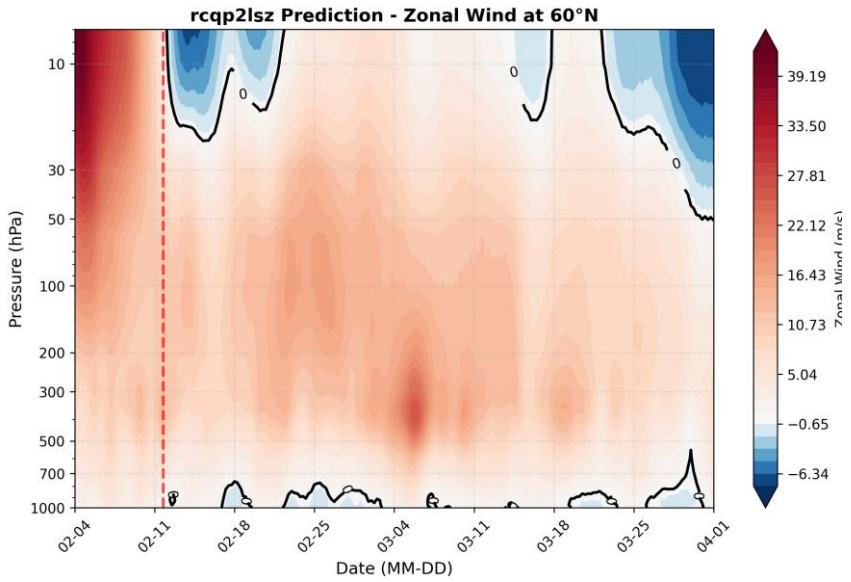


Why is it important?

Test latent temporal modeling & probe how much the model learned the correlation across variables (generalizability)

ECMWF

Is the model learning physics?



Physical Consistency Metrics

Conservation Laws

- Total mass
- Moisture
- Energy
- Potential vorticity

Good performance:

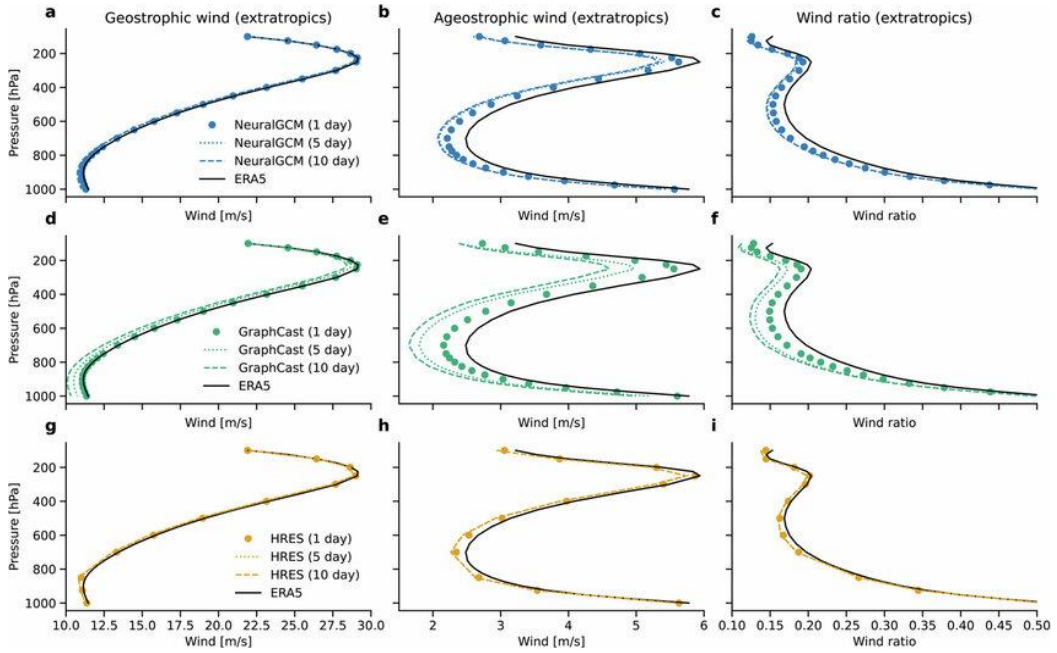
- Small long-term bias
- No runaway energy growth

Balance Constraints

- Geostrophic balance
- Hydrostatic balance
- Divergence vs vorticity ratios

Good performance:

- Ensure the latent space respects physical structure, not just statistics.



DOI: 10.1038/s41586-024-07744-y

Bonus: self supervised learning & evaluation

Our atmospheric analogy

SSL in W&C: **physical structures & regimes emerge**

How we evaluate our SSL model for W&C

- ***Freeze the backbone***
- Train lightweight head ***(linear might be too simple?)***
- Measure performance on downstream tasks
  (no end-to-end retraining)

Evaluated on few-shot W&C tasks

- RMSE for forecasting
- RMSE against SYNOP
- Maps and physical consistency

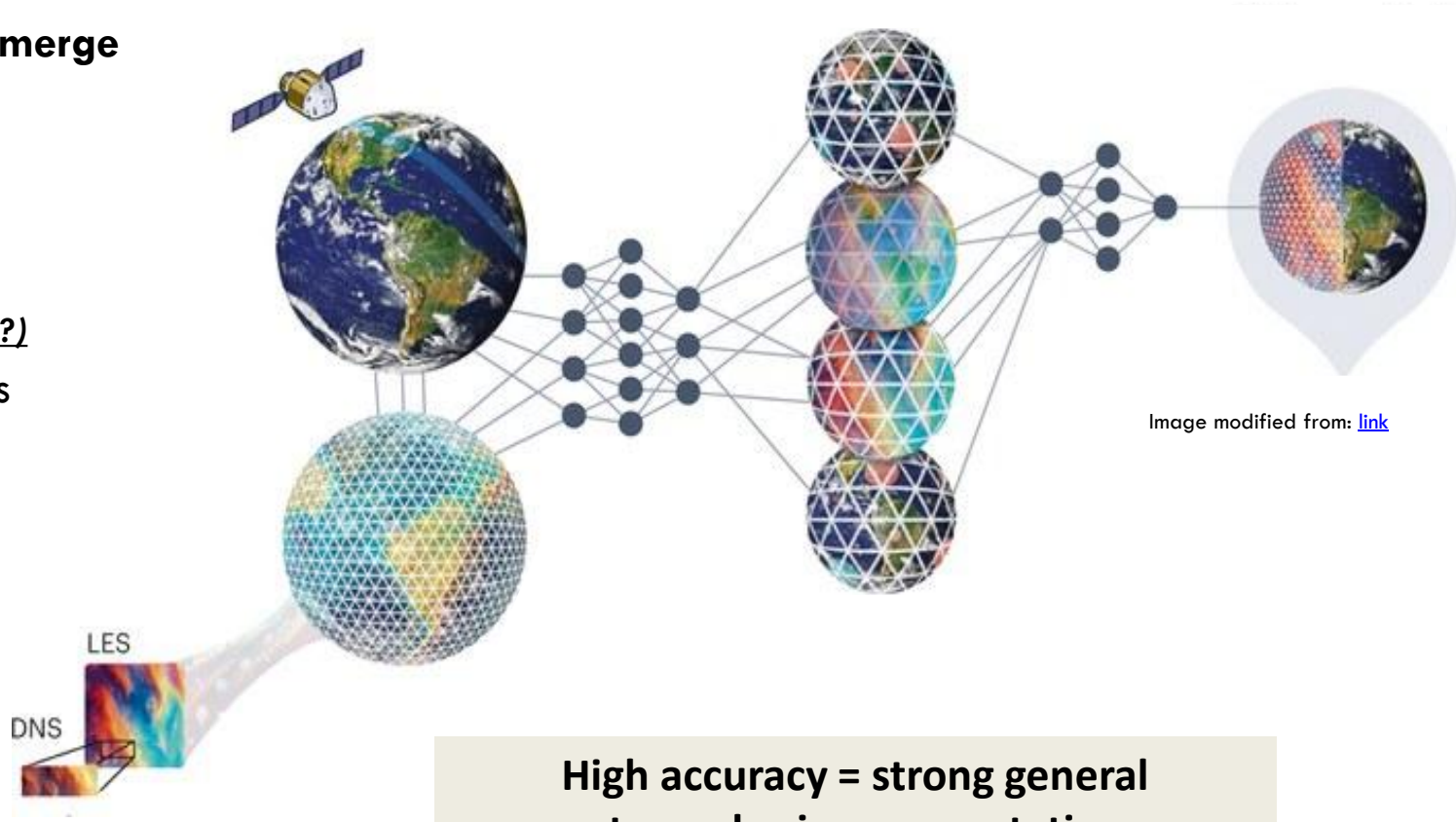


Image modified from: link

High accuracy = strong general atmospheric representation

ECMWF

Bonus: self supervised learning & evaluation

Our atmospheric analogy

SSL in W&C: **physical structures & regimes emerge**

How we evaluate our SSL model for W&C

- ***Freeze the backbone***
- Train lightweight head ***(linear might be too simple?)***
- Measure performance on downstream tasks
  (no end-to-end retraining)

More complex few-shot tasks

- Classification → e.g. rain/no-rain?
- Segmentation → land/sea reconstruction
- Tracking → Cyclone tracking

Main idea:

If a frozen representation performs well with a simple probe

↓

it has learned a general, reusable atmospheric structure — not task-specific shortcuts.

ECMWF

Conclusion

Conclusion

- Split Data Immediately into: Training, Validation, Test
- Try Cross-validation for Best Results
- **Beware of Correlated Data when Splitting**
  - Time Series & Geospatial Data are Always Correlated
- **Beware of data drifts & Data leakages**
- **A quick evaluation pipeline is essential for efficient ML R&D**
  - Focus on a few key metrics first
  - Look also at 2D maps, not only scores