The Abdus Salam International Centre for Theoretical Physics

UNIVERSITÀ DEGLI STUDI DI TRIESTE

APPLIED DATA SCIENCE & ARTIFICIAL INTELLIGENCE

GNN4CD: a graph neural network RCM emulator for efficient high-resolution climate projections

Valentina Blasone

ESP Seminar: Mar 24, 2026

Why deep learning?

Long or/and **many** climate model simulations at **high-resolution**

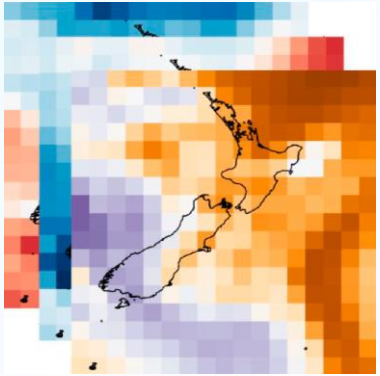
→

High computational **cost**

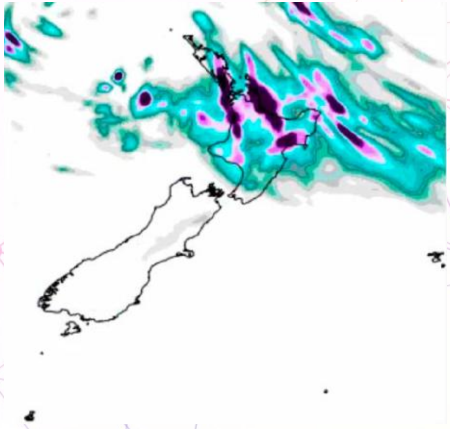
Deep learning exploits the available data and can learn to derive climate projections in a **computationally efficient** way

RCM emulator

A deep learning model that can produce climate projections (e.g. temperature, precipitation) at high resolution, given as input atmospheric climate simulations at low resolution



RCM emulator
(deep learning model)



Training the RCM emulator

- **Supervised** deep learning models are **trained** by providing examples of input and desired output (target)
- The model learns the optimal relationship between input and target by minimising a **loss function**
- The trained model can the be used in **inference** to get accurate estimates of the output quantity starting from meaningful inputs

Open research questions

State-of-the-art RCM emulators use convolutional neural networks (**CNNs**) or **Generative AI** and **train on climate simulations**

- How to be completely independent on climate simulations?
- How to deal with non-regular grids non-rectangular domains?
- How to achieve spatial transferability?
- How to get sub-daily res. and improve precipitation estimates?

GNN4CD

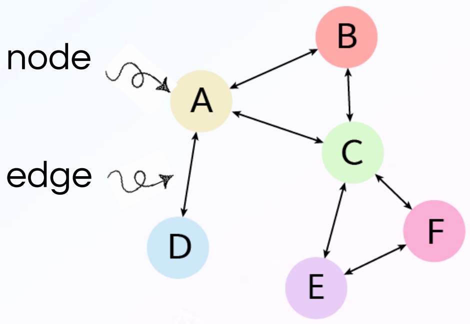
GNN4CD (Graph Neural Networks for Climate Downscaling) is a new RCM emulator that addresses these open research questions

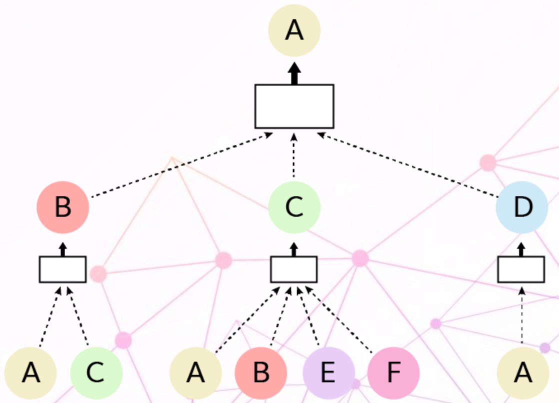
- Exploit graph neural networks (GNNs) → flexibility
- Train on reanalysis + observations → efficiency
- Use a high-resolution target → sub-daily resolution
- Use a custom loss function → account for extremes

Graph neural networks

GNNs are DL models for **graph-structured** data. Representations are learnt by exchanging information between neighbouring nodes

By **stacking layers**, they capture more complex interactions





GNN layer

Each layer performs **message passing** + **aggregation** + **update**

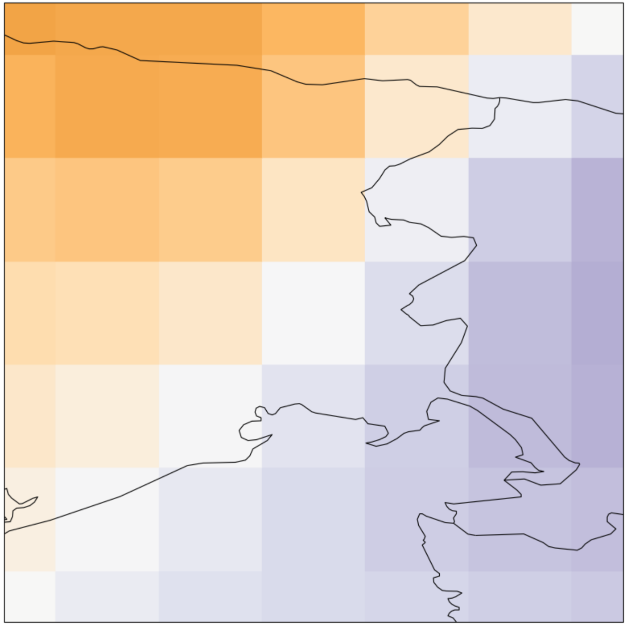
$$\mathbf{h}_i^k = f^k\left(\mathbf{W}_k \cdot \frac{\sum_{j \in \mathcal{N}_i} \mathbf{h}_j^{k-1}}{|\mathcal{N}_i|} + \mathbf{B}_k \cdot \mathbf{h}_i^{k-1}\right)$$

Graph convolutional operator

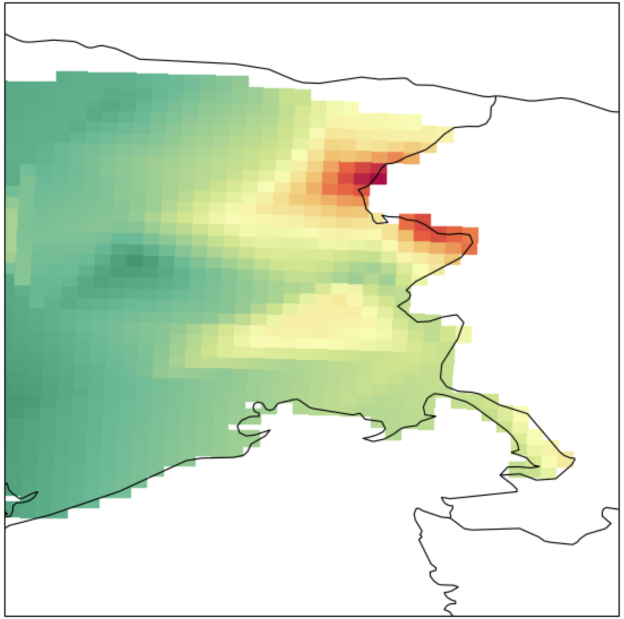
A GNN model works with graphs with an **arbitrary number** of **nodes and edges** (only the feature dimension must be fixed)

Building the graph

Low-resolution input

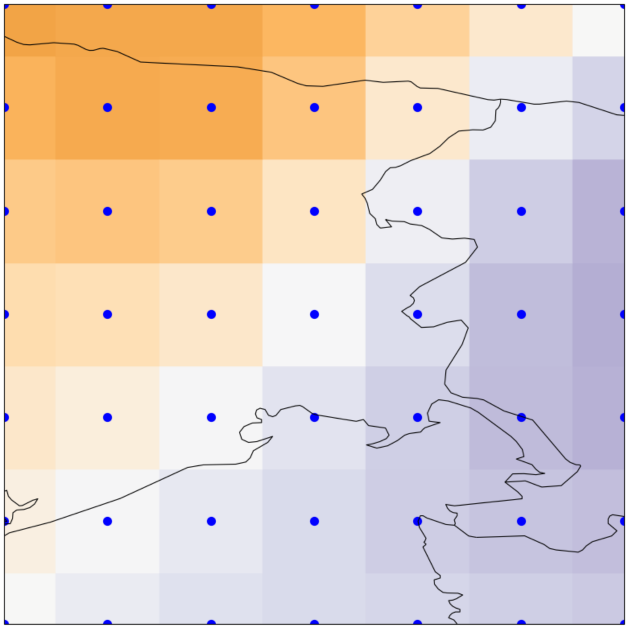


High-resolution output

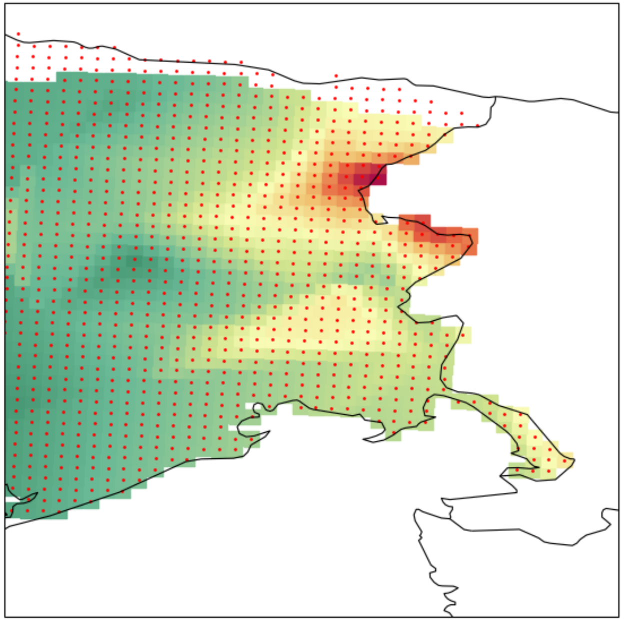


Building the graph

Low nodes

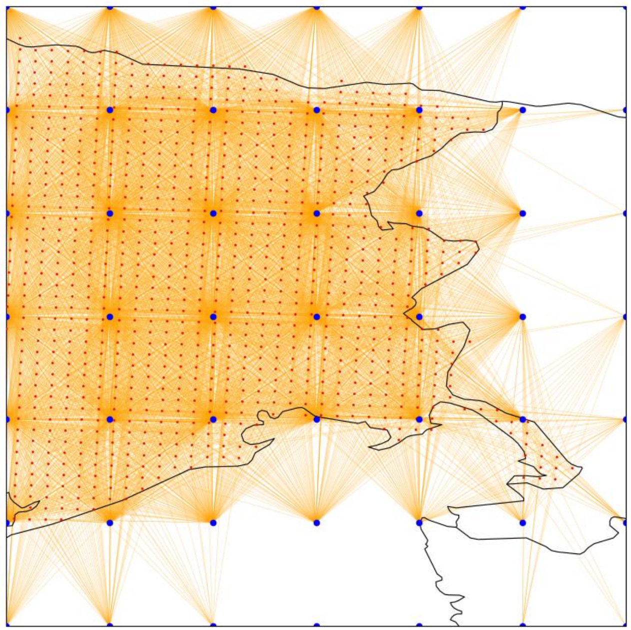


High nodes

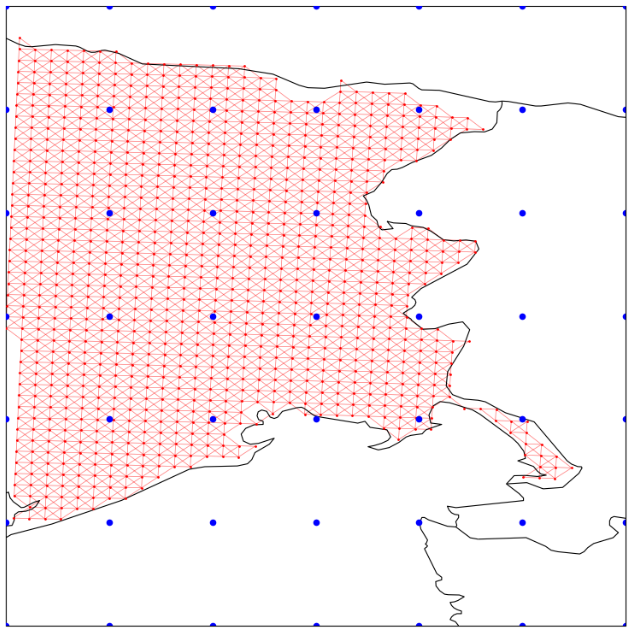


Building the graph

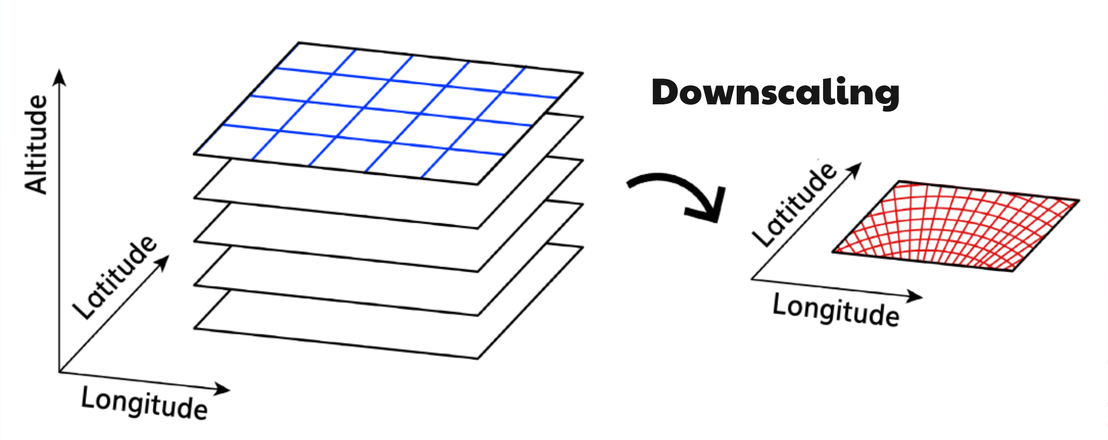
→ Low-to-High edges



↔ High-within-High edges



The downscaling task

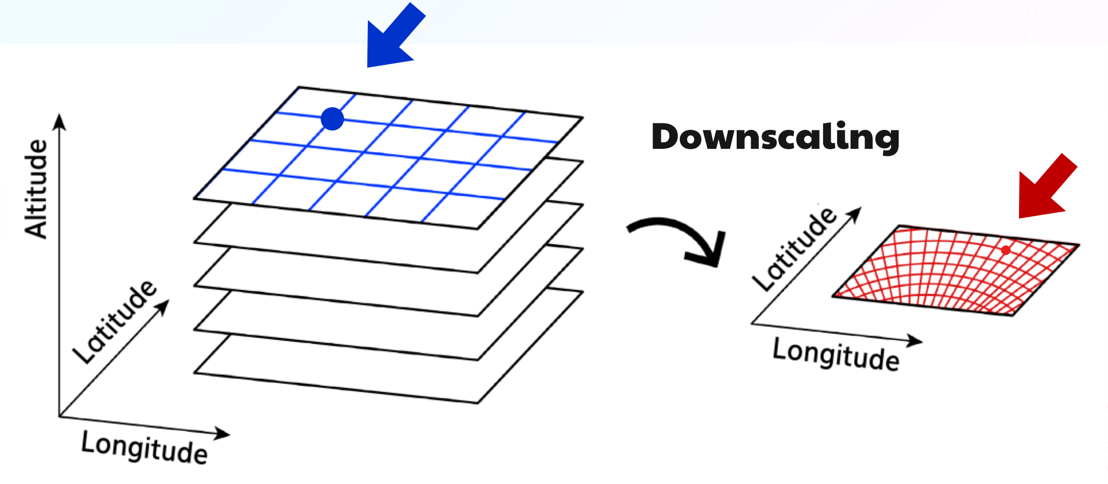


Low-resolution atmospheric variables

High-resolution precipitation

+ elevation and land use predictors

The downscaling task



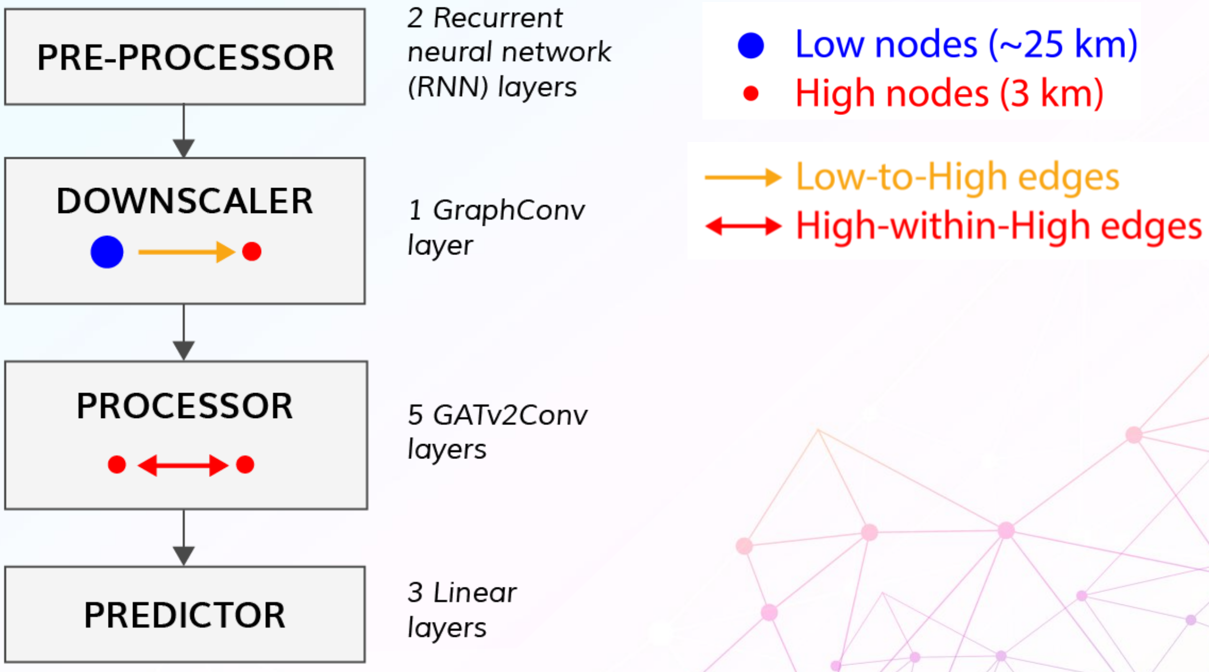
Time series of predictors

$[t - \mathrm{L}, \dots, t]$

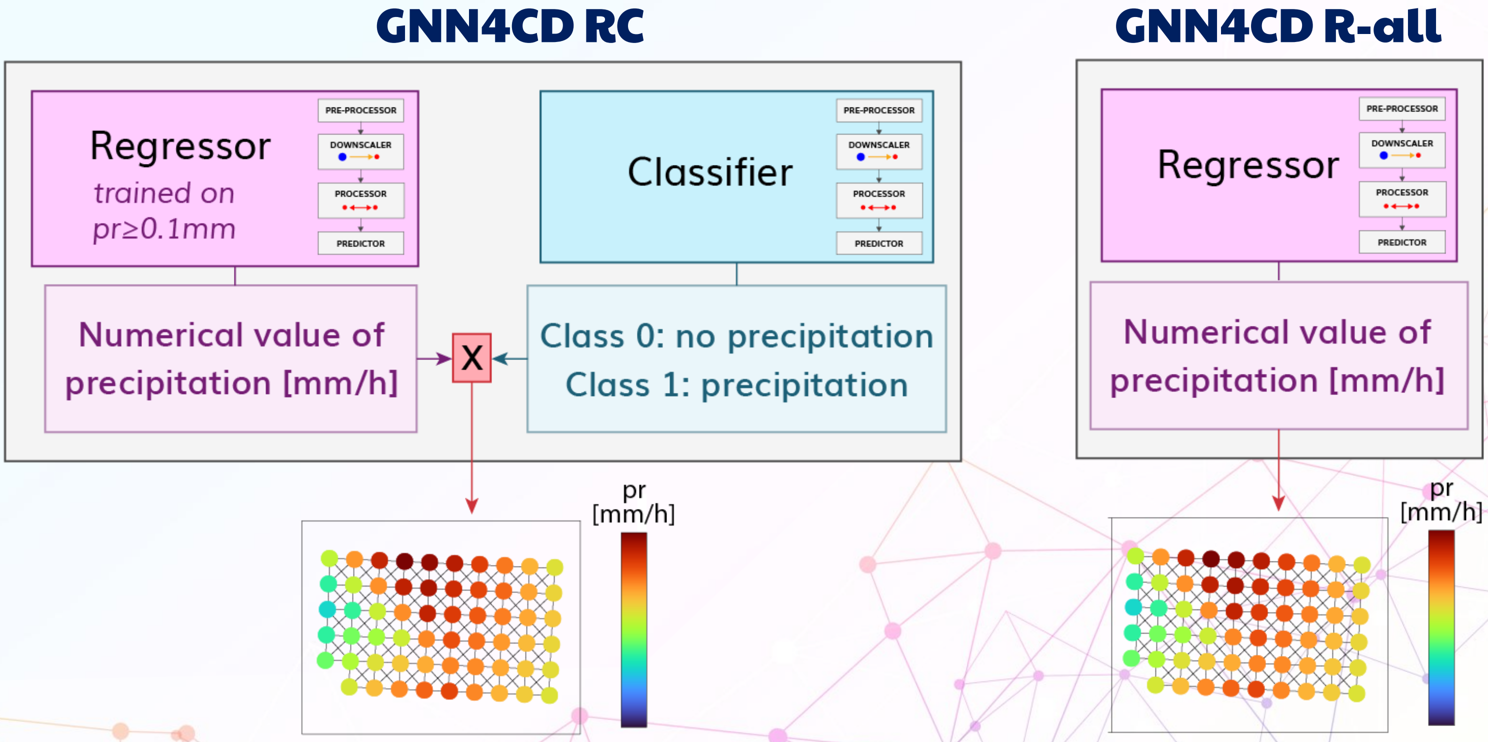
Single time output

$[t]$

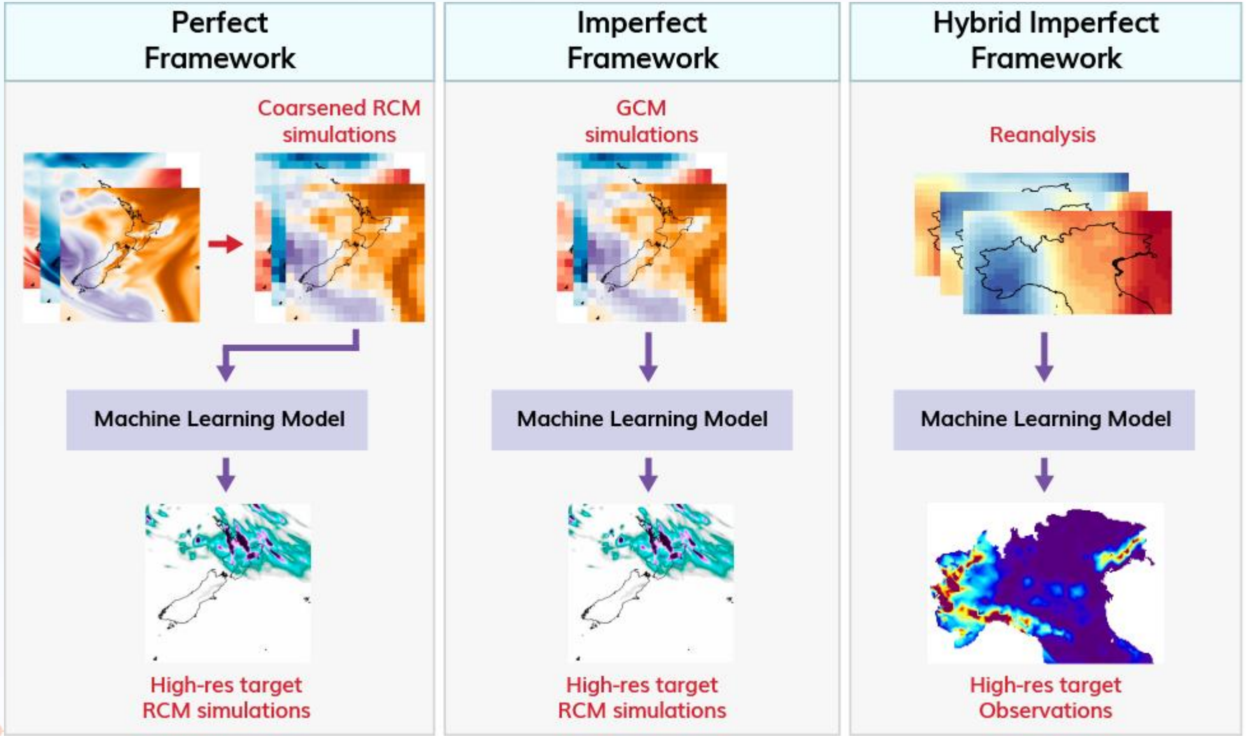
GNN-based architecture



Model design



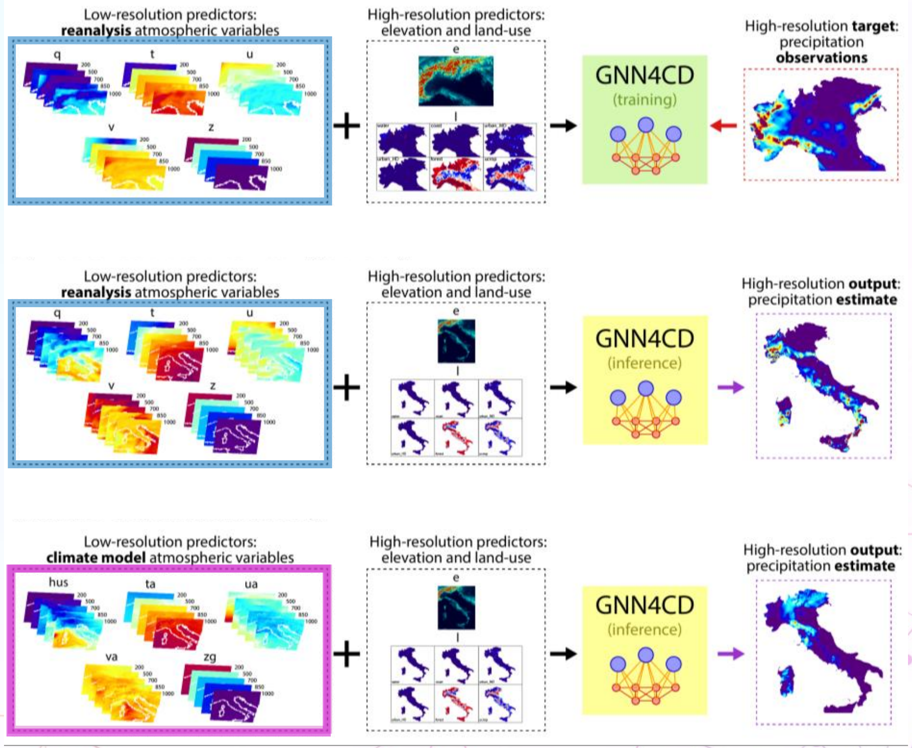
Training frameworks



Hybrid imperfect framework

Training

Inference



Reanalysis to observation downscaling

Reanalysis to observation downscaling

RCM emulation

Hybrid imperfect framework

Main advantages:

- Cost-effective
- Learns from a better representation of the present-day conditions and develops a broader foundation in atmospheric dynamics

Possible limitations

- Domain mismatch between training and inference
- Observational dataset may contain uncertainties and biases
- Reanalysis and observations are limited to the present day

Predictors and target

	Variable	Symbol	Unit	Pressure Levels [hPa]	Space	Time
P	Specific humidity	q, hus	[kg kg-1]	1000; 850; 700; 500; 200	0.25°	1hr
	Temperature	t, ta	[K]	1000; 850; 700; 500; 200	0.25°	1hr
	Eastward wind	u, ua	[m/s]	1000; 850; 700; 500; 200	0.25°	1hr
	Northward wind	v, va	[m/s]	1000; 850; 700; 500; 200	0.25°	1hr
	Geopotential	z, zg	[m^2/s^2]	1000; 850; 700; 500; 200	0.25°	1hr
	Elevation	e	[m]	Surface	3km	-
	Land-use	l	[%]	Surface	3km	-
T	Precipitation	pr	[mm]	Surface	3km	1hr

Low-resolution predictors

High-resolution predictors

High-resolution target

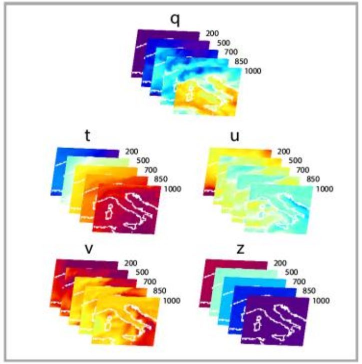
Datasets

Reanalysis to observation downscaling

Input datasets

Target dataset

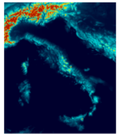
ERA5 REANALYSIS
(~25 km)

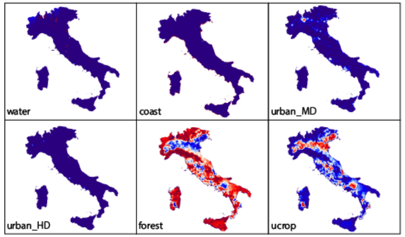


HUMIDITY, TEMPERATURE, WIND, GEOPOTENTIAL

4D: lon, lat, altitude, time (hourly)

GMTED2010 + CLM4.5
(3 km)

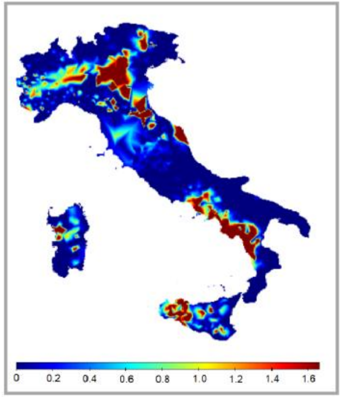




ELEVATION + LAND USE

2D: lon, lat

GRIPHO OBSERVATIONS
(3 km)



PRECIPITATION

3D: lon, lat, time (hourly)

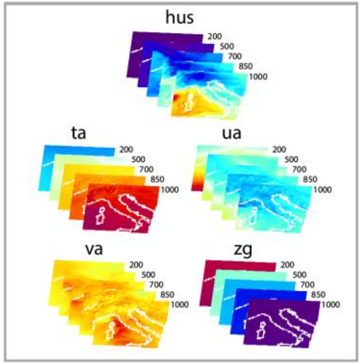
Datasets

Input datasets

Target dataset

RegCM SIMULATIONS with RCP8.5 scenario

(~25 km)

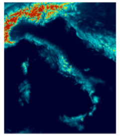


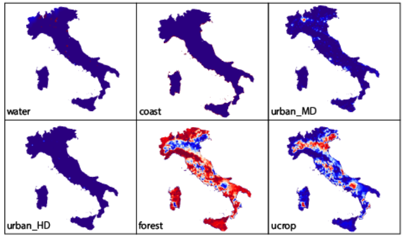
HUMIDITY, TEMPERATURE, WIND, GEOPOTENTIAL

4D: lon, lat, altitude, time (hourly)

GMTED2010 + CLM4.5

(3 km)



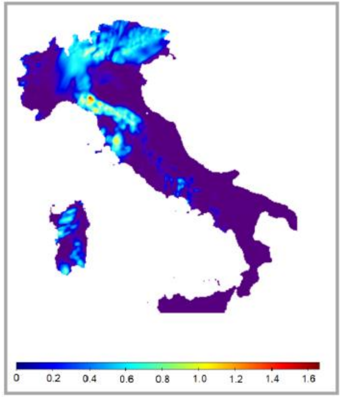


ELEVATION + LAND USE

2D: lon, lat

RegCM SIMULATIONS with RCP8.5 scenario

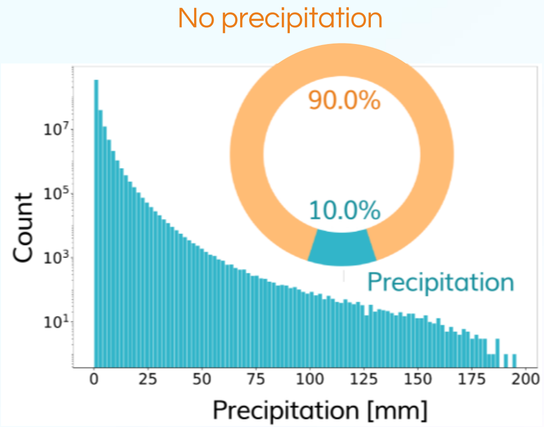
(3 km)



PRECIPITATION

3D: lon, lat, time (hourly)

Loss functions



Regressor: Quantised MSE (QMSE)

$$\mathrm{QMSE} = \sum_{j}^{B} \frac{1}{|\Omega_j|} \sum_{i \in \Omega_j} (y_i - \hat{y}_i)^2$$

$$\bar{\alpha}\text{-}QMSE = MSE + \bar{\alpha} \cdot QMSE$$

B: number of bins (defined on the training data)
j: bin index, from 1 to B
Ω_j: target indices whose values fall within bin j
$y_i, \hat{y}_i$: predicted, ground-truth values

Classifier: Focal Loss (FL)

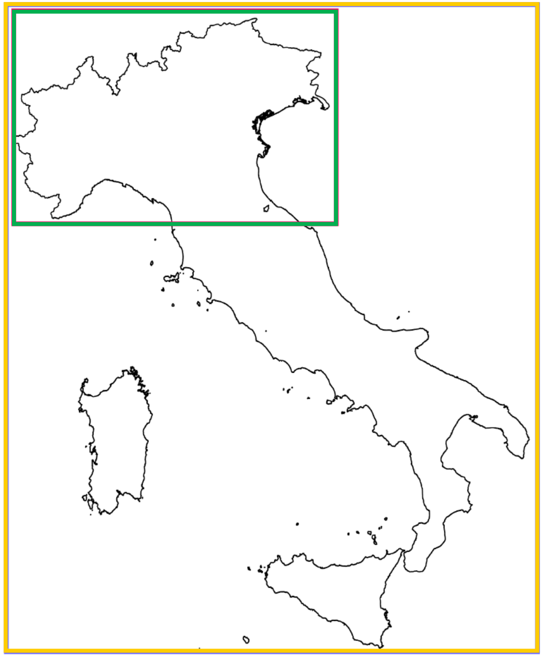
$$FL(p_t) = -\alpha_t (1 - p_t)^{\gamma} \cdot log(p_t)$$

$$p_t = \begin{cases} p & if\ y = 1 \\ 1 - p & otherwise \end{cases}$$

$$\alpha_t = \begin{cases} \alpha & if\ y = 1 \\ 1 - \alpha & otherwise \end{cases}$$

$y \in \{0,1\}$: ground-truth class
$p \in [0,1]$: model probability of class $y = 1$
γ, α: hyperparameters

Spatial and temporal domains



training inference

Northern Italy: ~400 Low and ~14000 High nodes

15 years: 2001-2015 (2007 used for validation)

Moderately long training: 50 epochs ~ **24h**

Italy: ~1000 Low and ~33000 High nodes

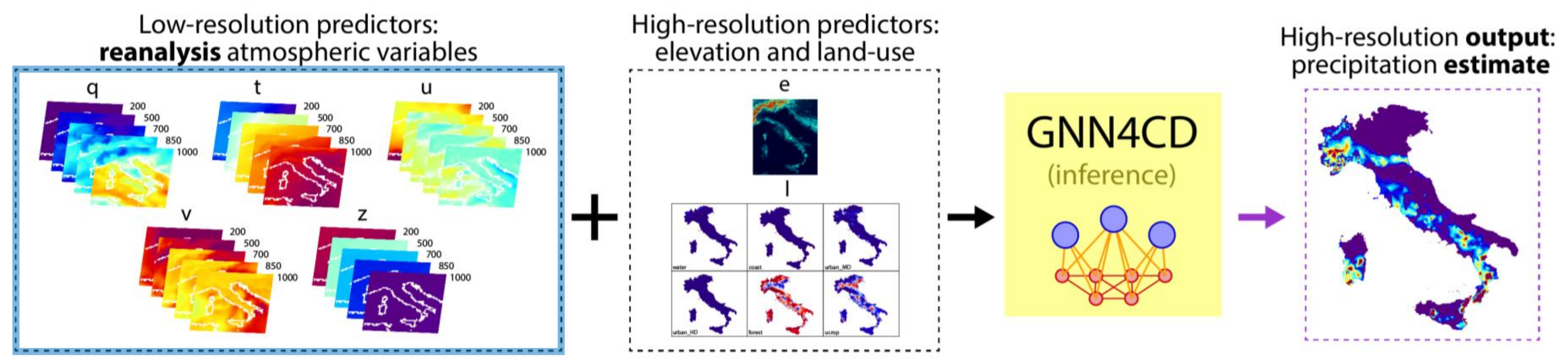
- Reanalysis to observation downscaling: **1 year** (2016)
- RCM emulation: three periods of **9-10 years** each

historical (1996-2005), *mid-century* (2041-2049), *end-of-century*(2090-2099)

Fast inference: estimates for one year ~ **few minutes**

Results

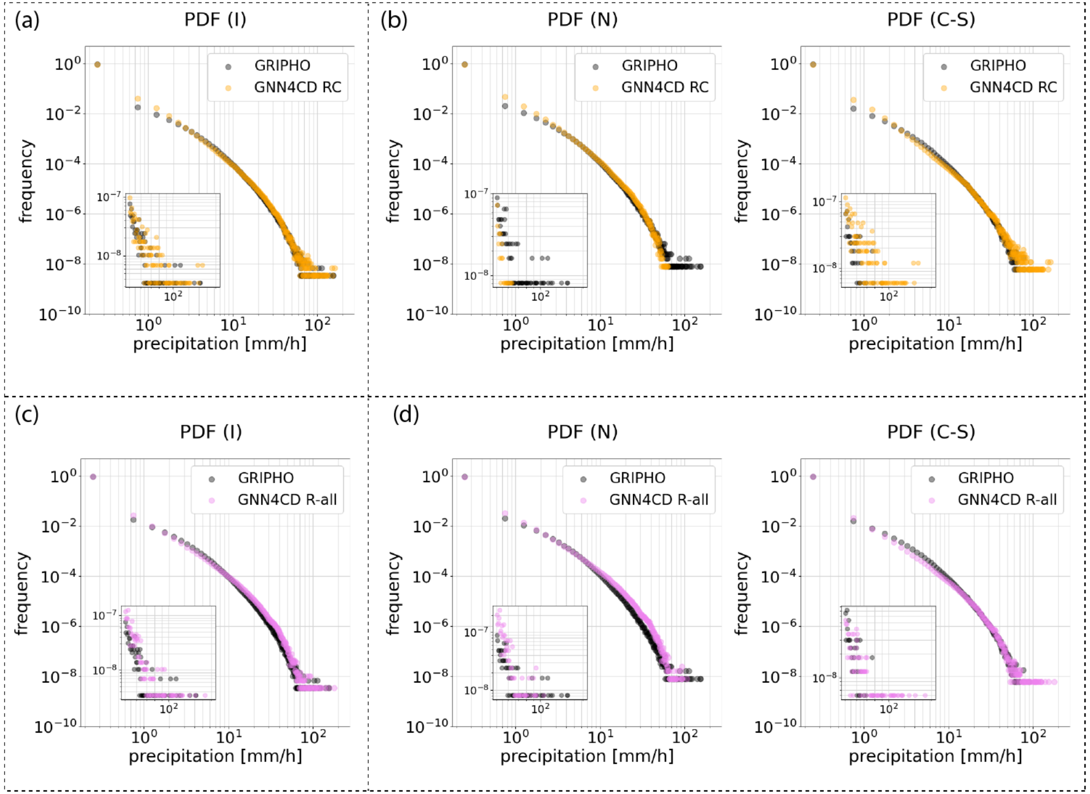
Reanalysis to observation downscaling



Comparison with **GRIPHO** ground truth

PDFs

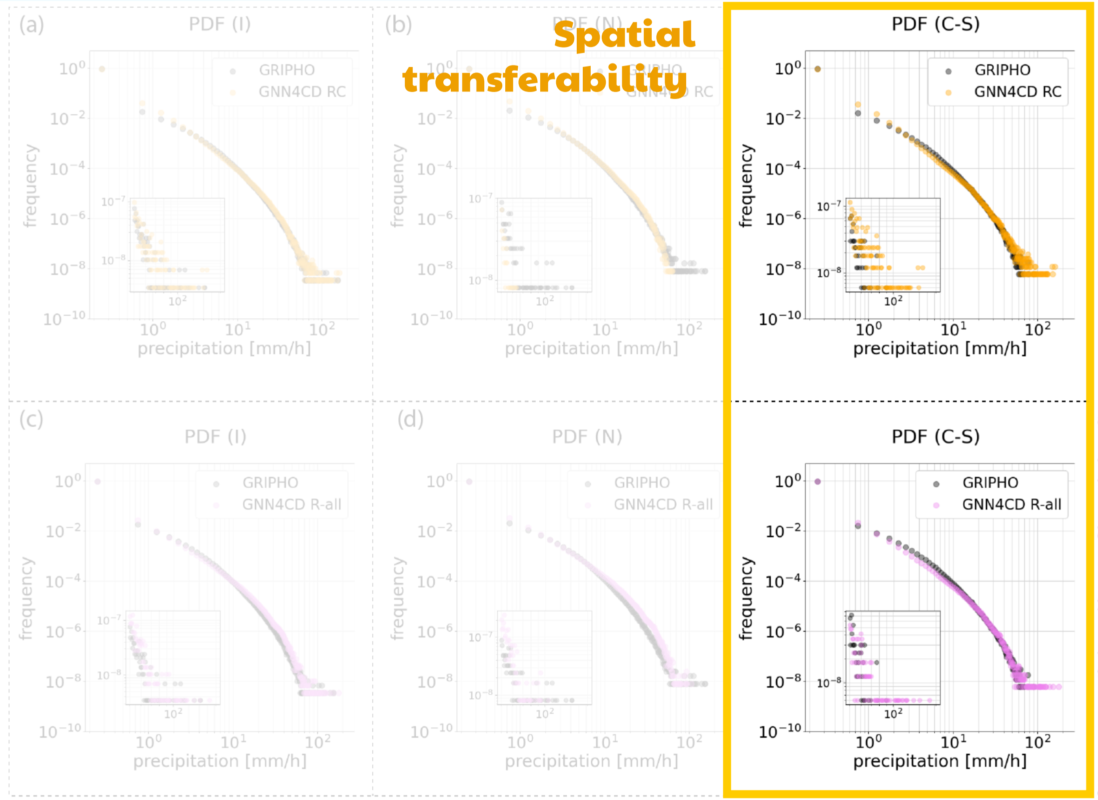
Reanalysis to observation downscaling



(I): Italy
(N): Northern Italy
(C-S): Central-South Italy

PDFs

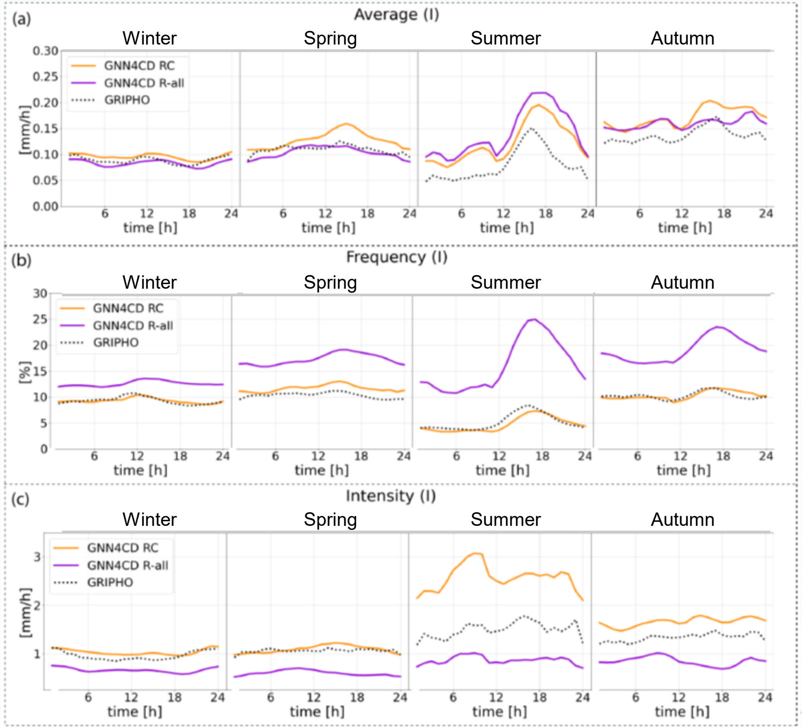
Reanalysis to observation downscaling

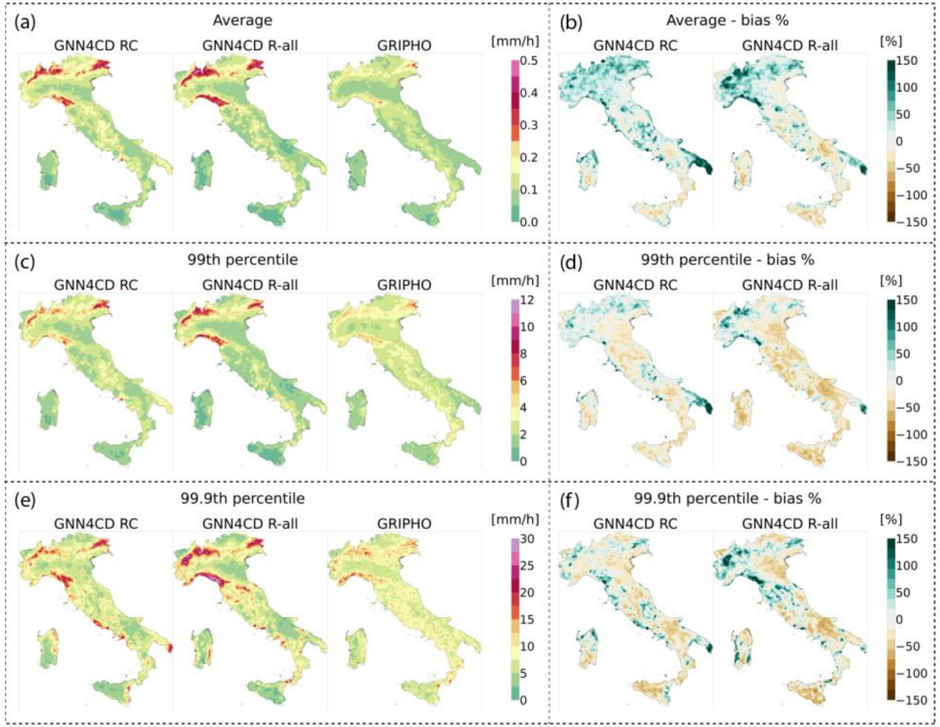


(I): Italy
(N): Northern Italy
(C-S): Central-South Italy

Diurnal cycles and maps

Reanalysis to observation downscaling

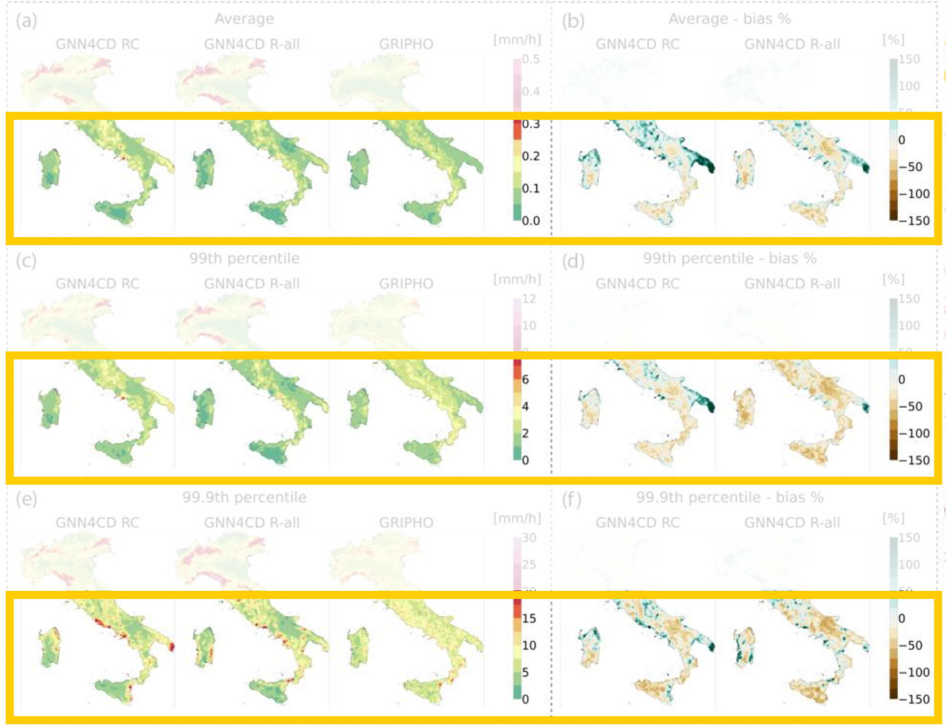




(I): Italy, (N): Northern Italy, (C-S): Central-South Italy

Diurnal cycles and maps

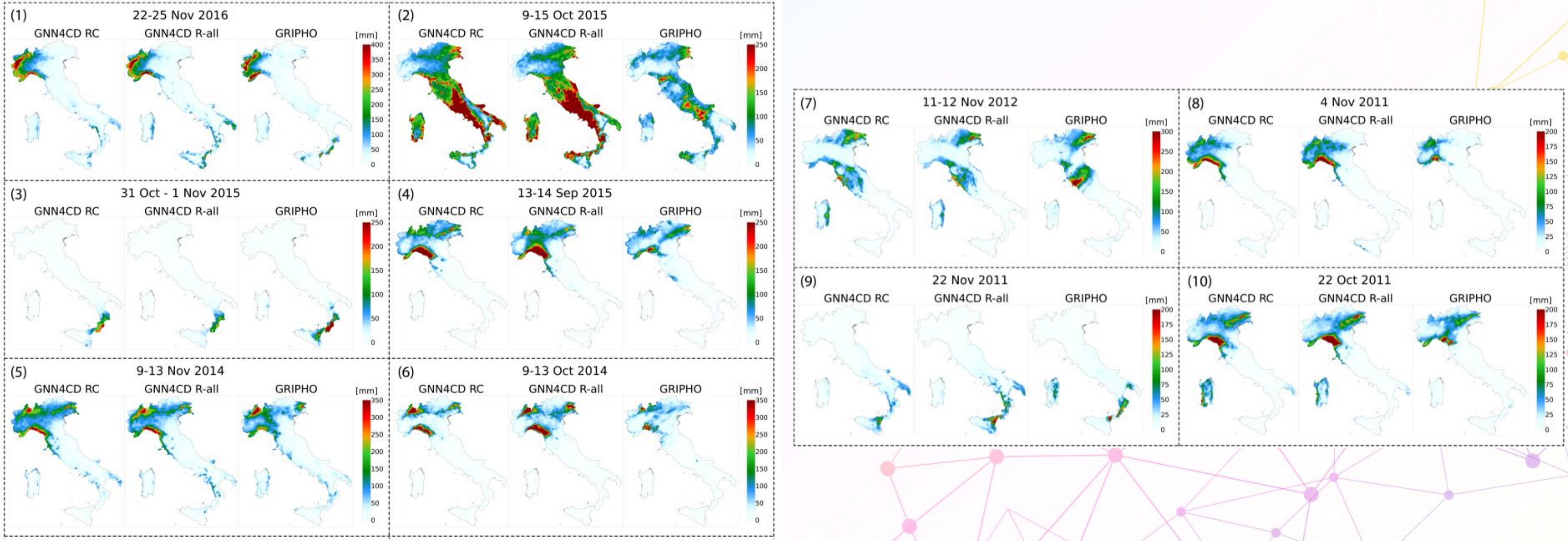
Reanalysis to observation downscaling



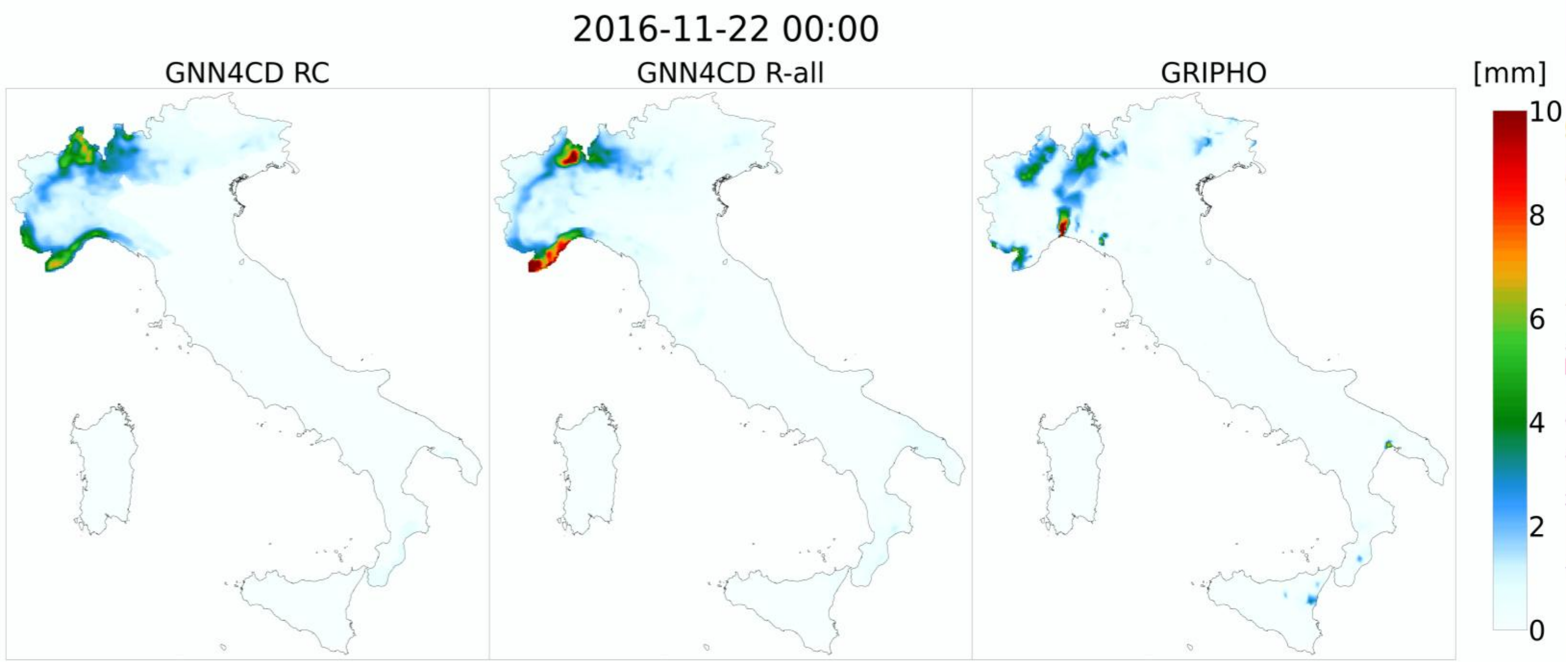
(I): Italy, (N): Northern Italy, (C-S): Central-South Italy

Floods

Reanalysis to observation downscaling

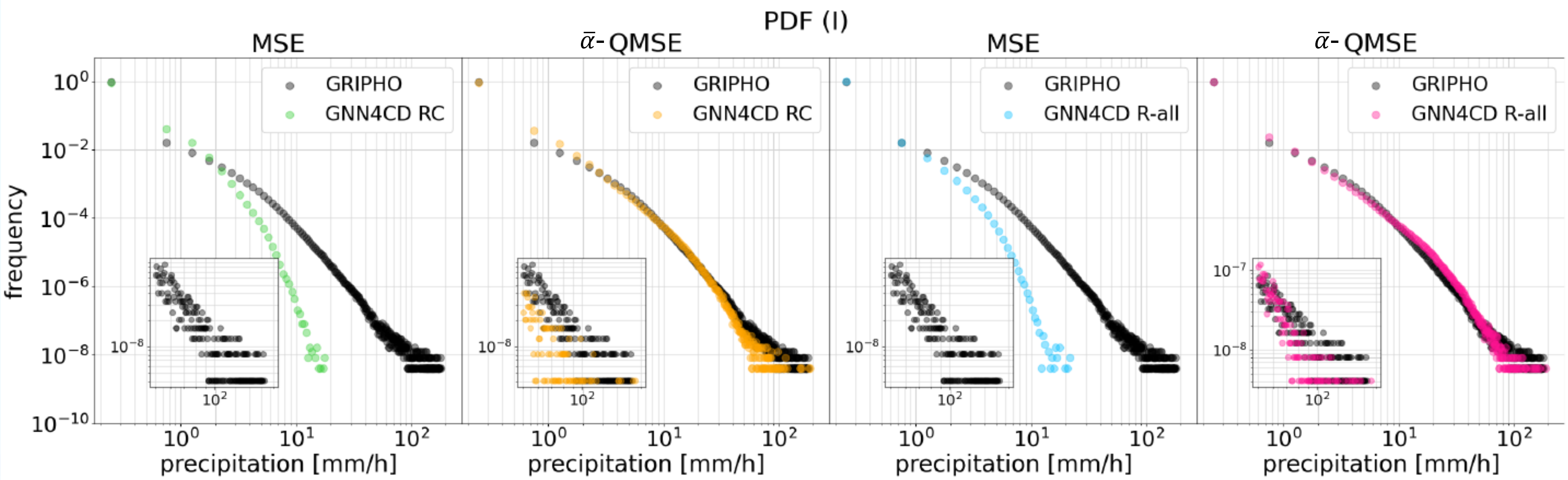


Floods: 22-25 Nov 2016

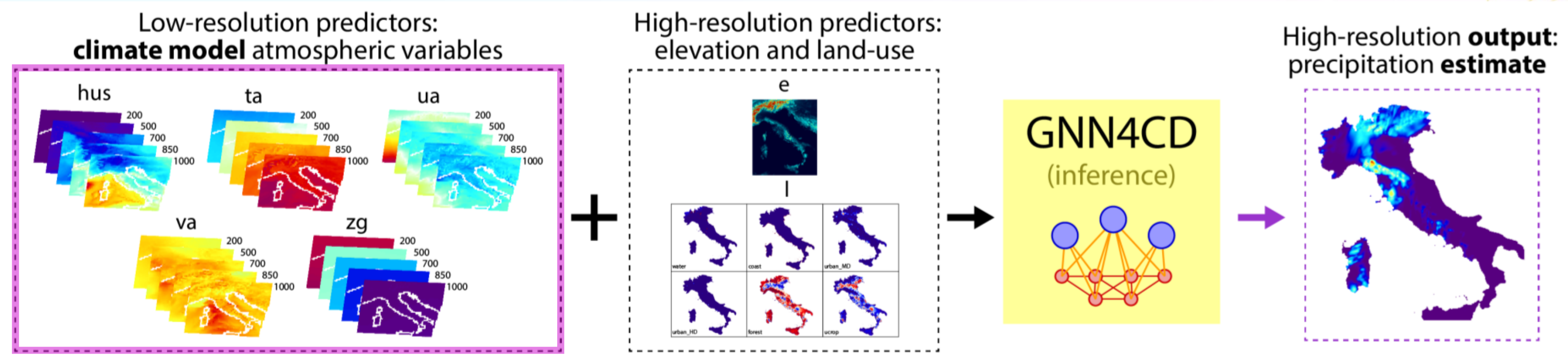


QMSE vs MSE

Reanalysis to observation downscaling



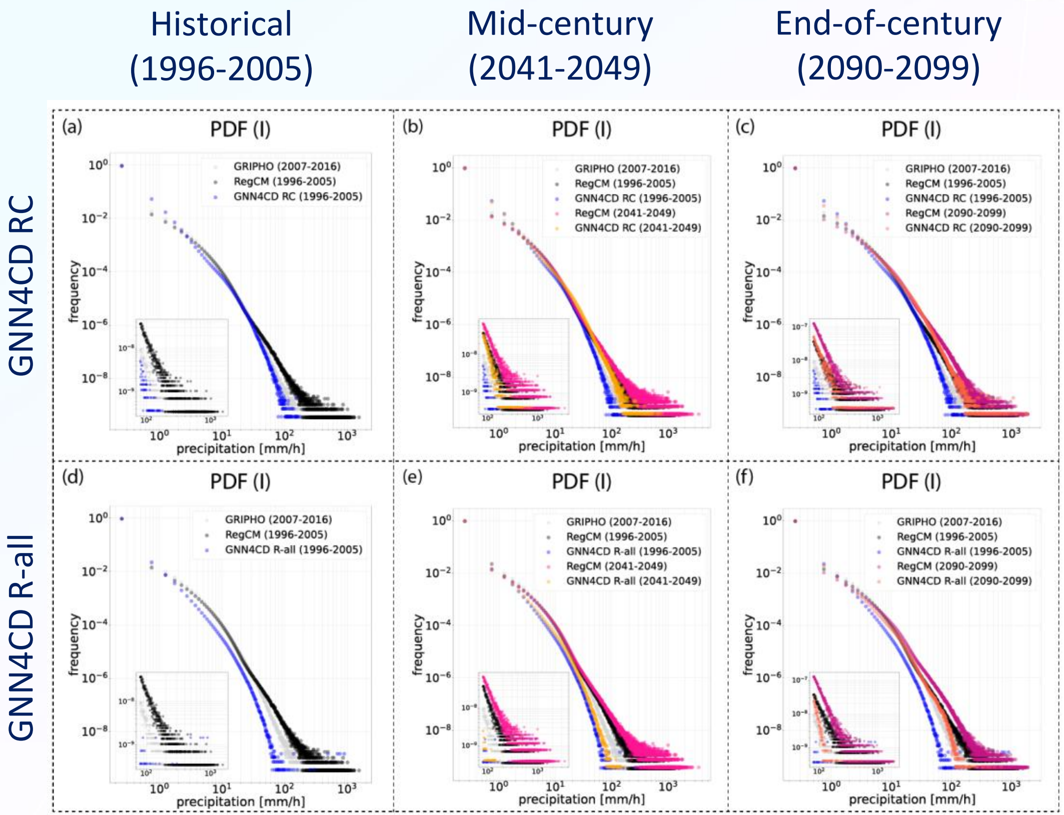
Results



No ground truth → comparative analysis with RegCM outputs
Evaluate if the emulator captures the effects of climate change

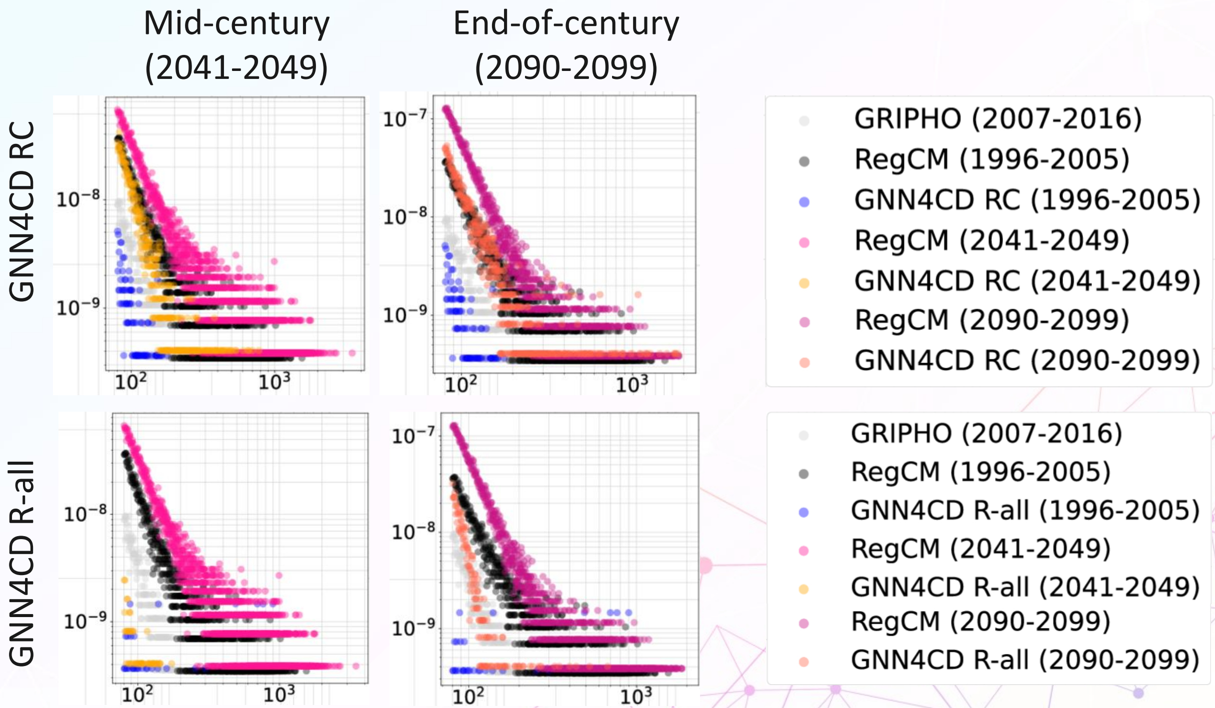
PDFs

RCM emulation



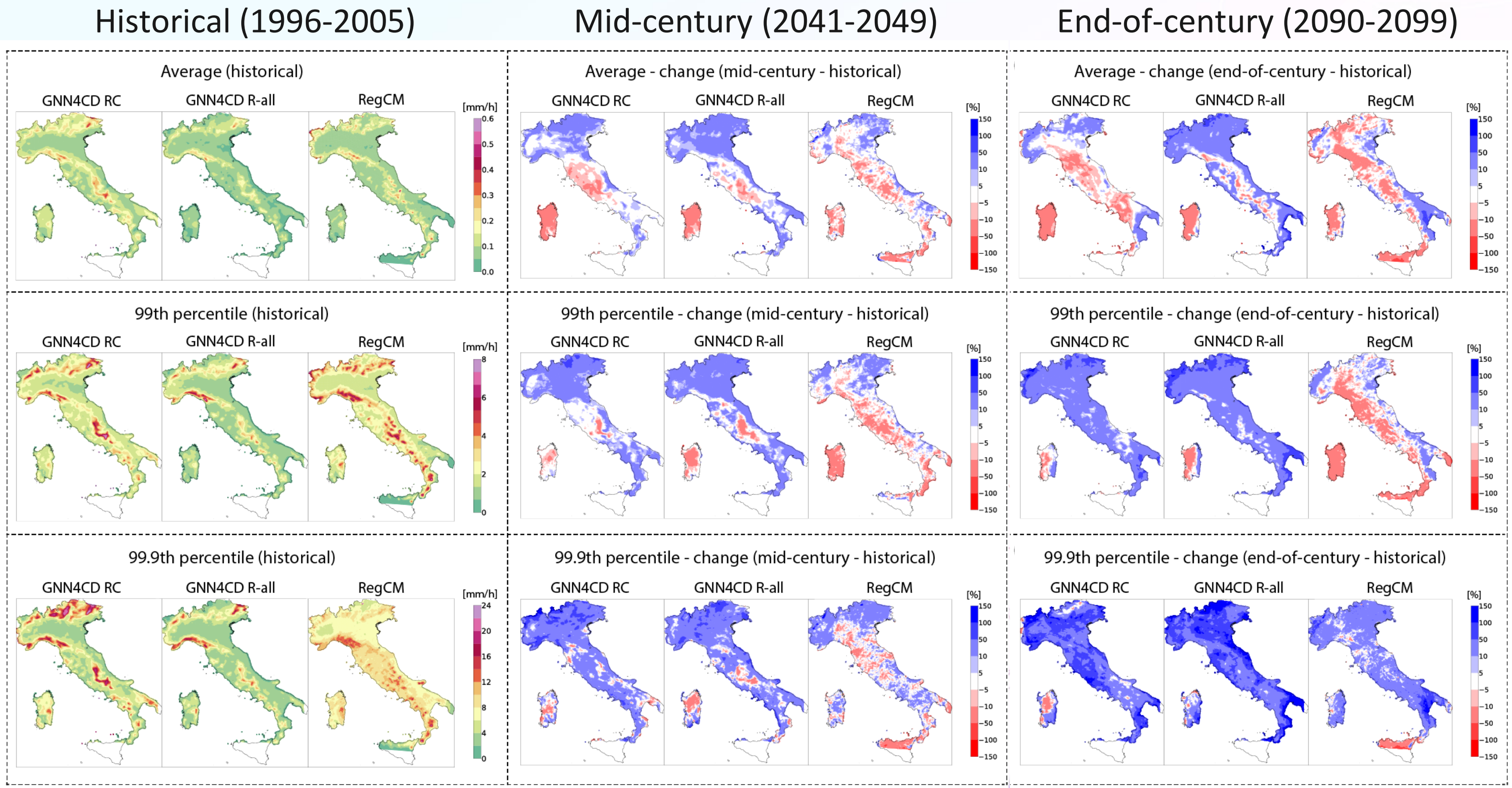
(I): Italy

PDFs: tail shift



Remarkable result as GNN4CD was not trained on any future precipitation data

Change

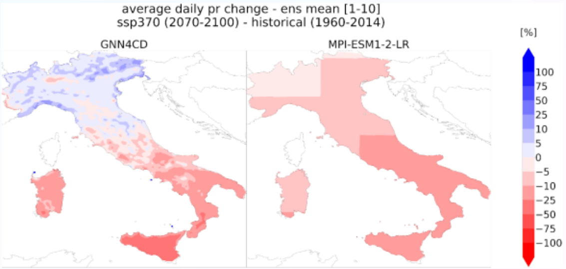


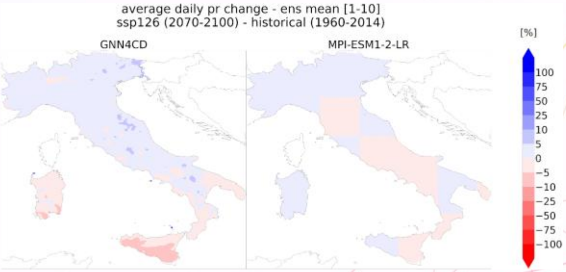
Historical (1996-2005)
Mid-century (2041-2049)
End-of-century (2090-2099)
Average (historical)
Average - change (mid-century - historical)
Average - change (end-of-century - historical)
99th percentile (historical)
99th percentile - change (mid-century - historical)
99th percentile - change (end-of-century - historical)
99.9th percentile (historical)
99.9th percentile - change (mid-century - historical)
99.9th percentile - change (end-of-century - historical)
GNN4CD RC
GNN4CD R-all
RegCM
[mm/h]
[%]

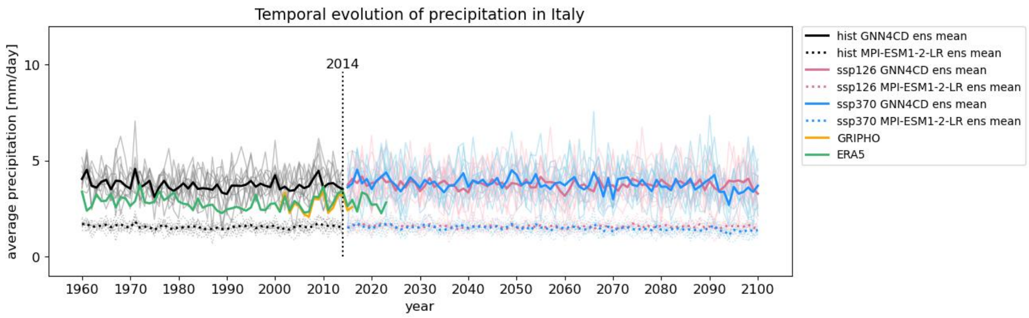
Ongoing work

Use the trained GNN4CD to downscale **GCM** simulations ensembles

- 250km → 3km
- No retraining
- No ground-truth

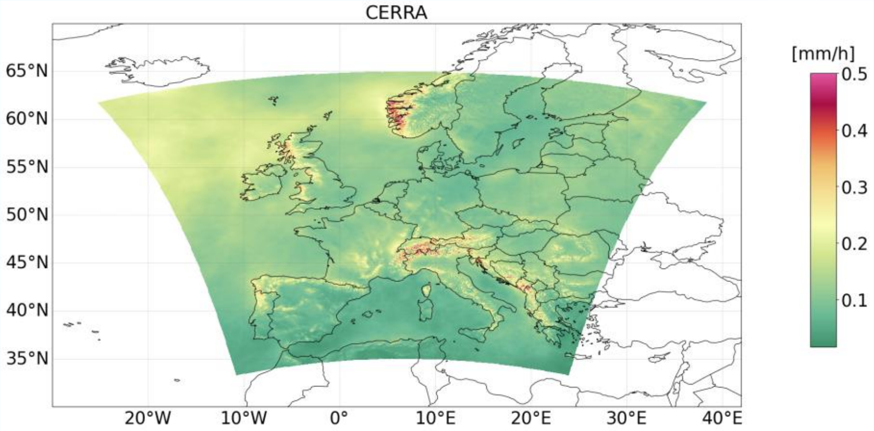


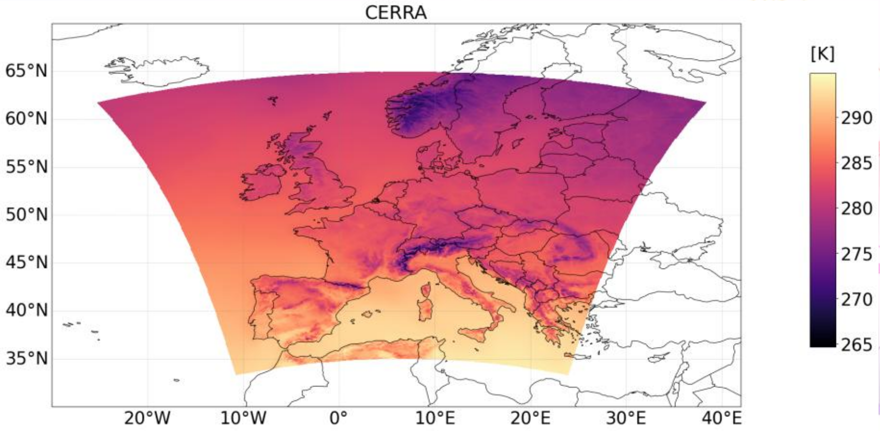




Ongoing work

Train on **CERRA** reanalysis (5.5km) → extended temporal and spatial domains





Ongoing work

Use GNN4CD to build a RCM emulator of **RegCM5**

Training strategies:

- Perfect framework
- Imperfect framework
- Hybrid imperfect framework + fine tuning

Resolution jumps:

- GCM → RCM
- RCM → CP-RCM
- GCM → CP-RCM

Thank you!