Barcelona Supercomputing Center
Centro Nacional de Supercomputación
EXCELENCIA SEVERO OCHOA

Learning to simulate precipitation with Deep Neural Networks

Carlos Gómez Gonzalez, Markus Donat, Kim Serradell
carlos.gomez@bsc.es

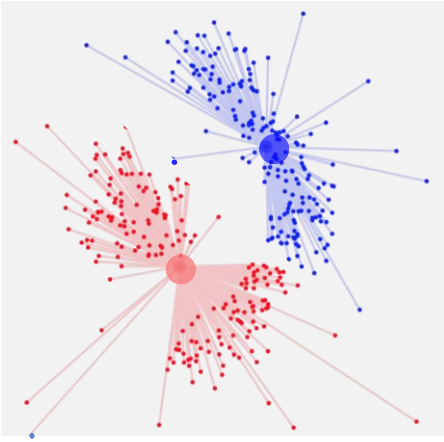
May 29 2020

6th ENES HPC Workshop 2020

STARS POST-DOCTORAL PROGRAMME

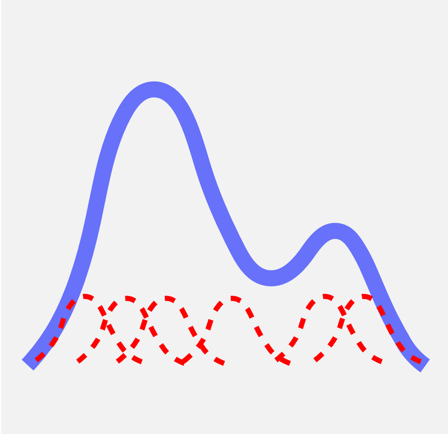
Machine Learning

Unsupervised



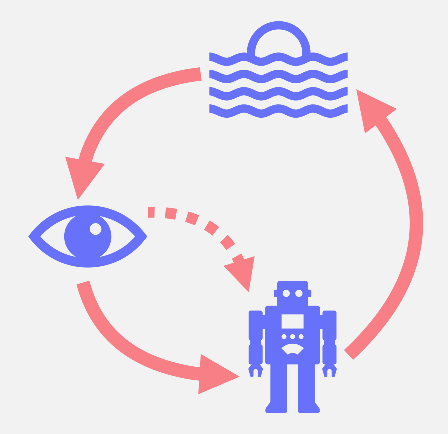
Clustering

Dimensionality reduction

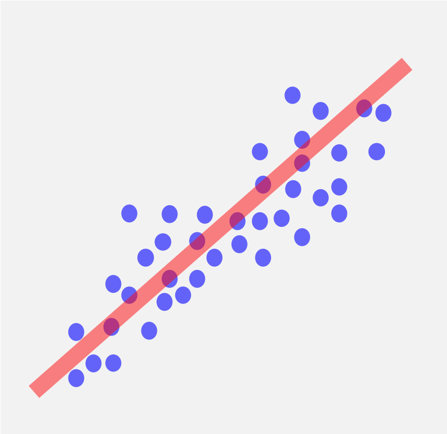


Density estimation

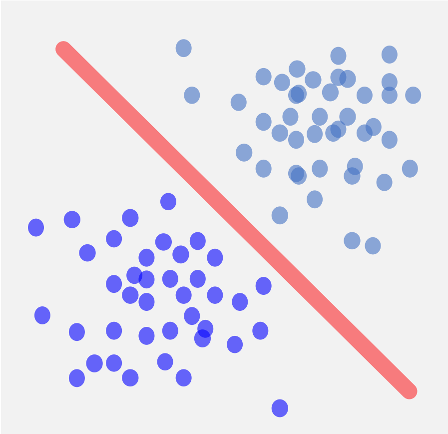
Reinforcement



Supervised



Regression



Classification

Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Supervised Learning

$$f : \mathcal{X} \to \mathcal{Y}, \quad (x_i, y_i)_{i=1,\ldots,n}$$

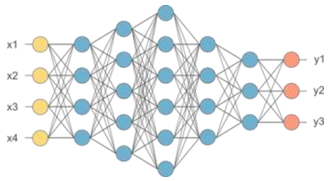
$$f = \arg\min_{f_\theta, \theta \in \Theta} \sum_{i=1}^{n} \mathcal{L}(y_i, f_\theta(x_i)) + g(\theta)$$

Supervised Learning

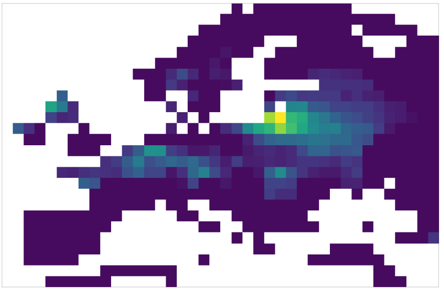
$$f : \mathcal{X} \to \mathcal{Y}, \quad (x_i, y_i)_{i=1,\ldots,n}$$

$$f = \underset{f_\theta,\theta\in\Theta}{\arg\min} \sum_{i=1}^{n} \mathcal{L}(y_i, f_\theta(x_i)) + g(\theta)$$

Training data

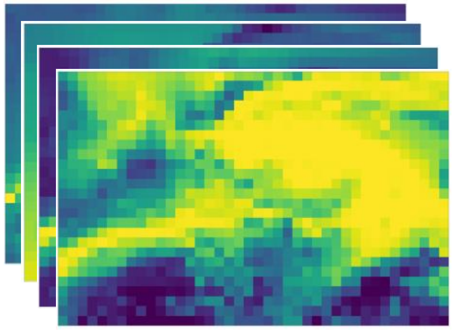


Model architecture

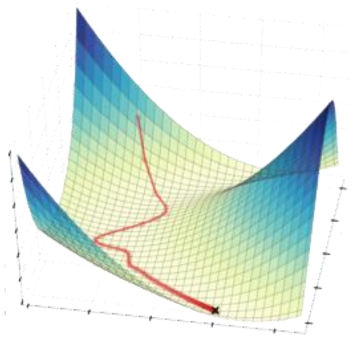


y_i

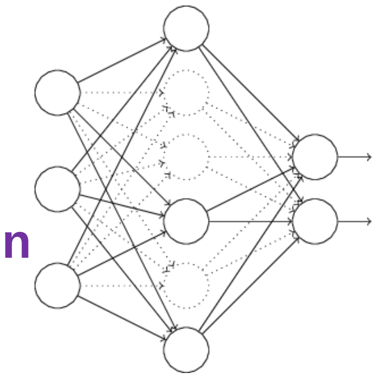
x_i



Supervised Learning



$$f : \mathcal{X} \to \mathcal{Y}, \quad (x_i, y_i)_{i=1,\ldots,n}$$



Optimization

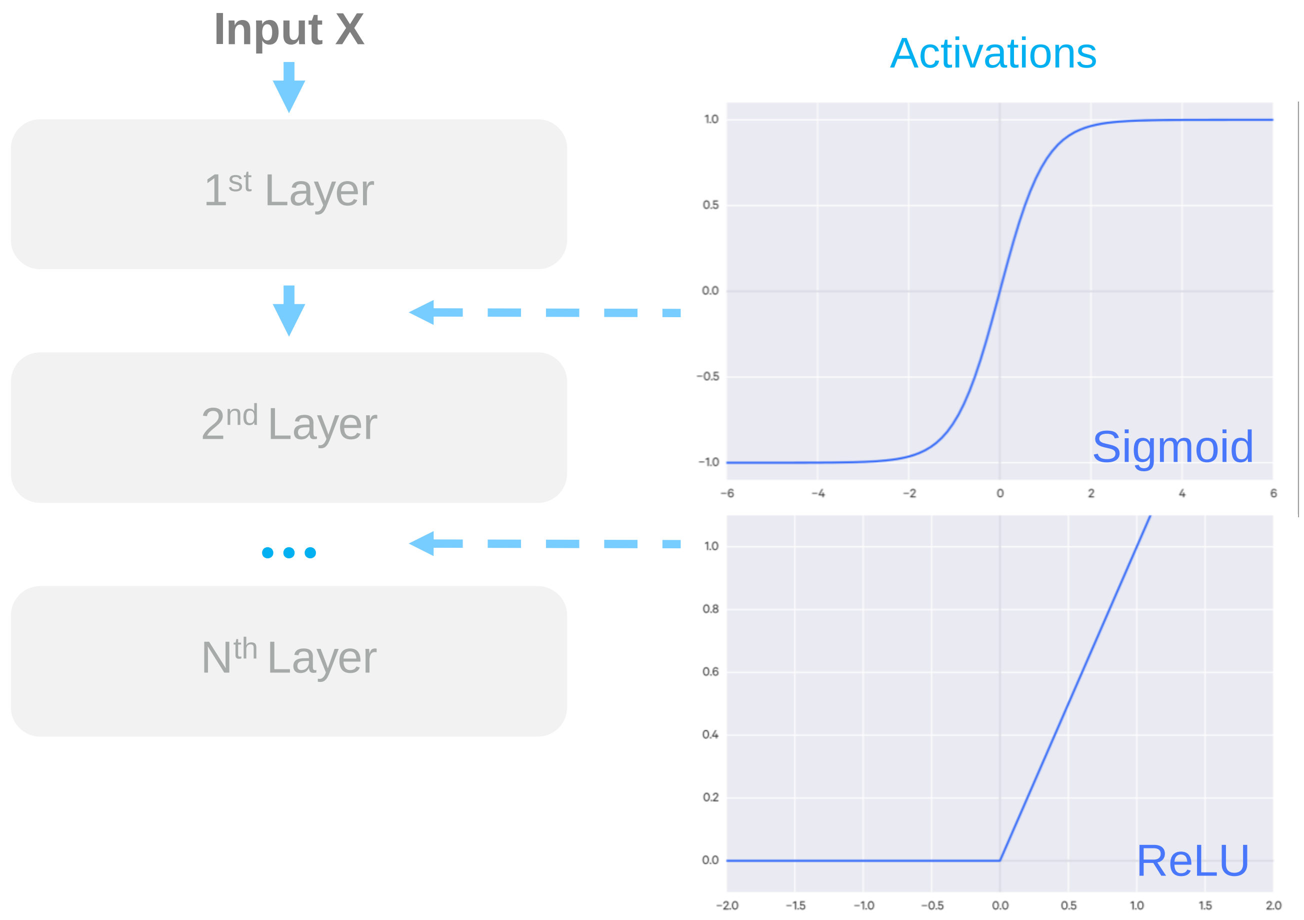
Loss function

Regularization

$$f = \underset{f_\theta,\theta\in\Theta}{\arg\min} \sum_{i=1}^{n} \mathcal{L}(y_i, f_\theta(x_i)) + g(\theta)$$

Deep Neural Network

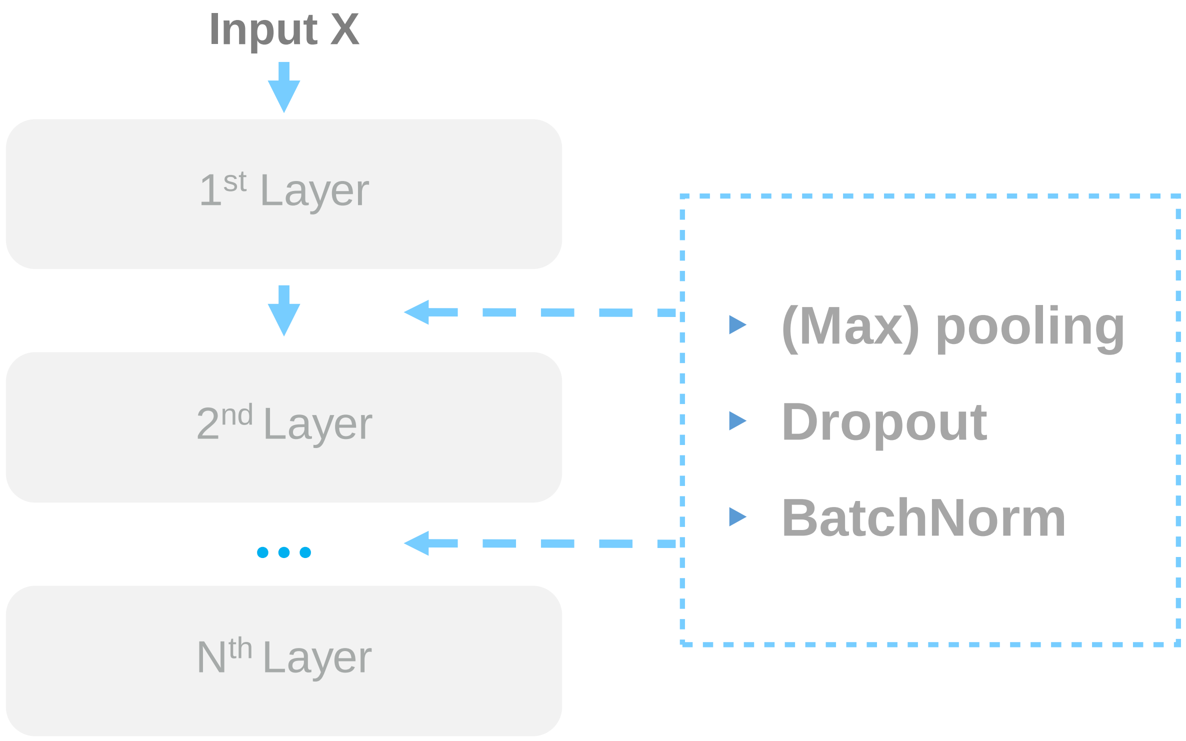
Succession of simple
linear data transformations
interleaved with simple
non-linearities



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Deep Neural Network

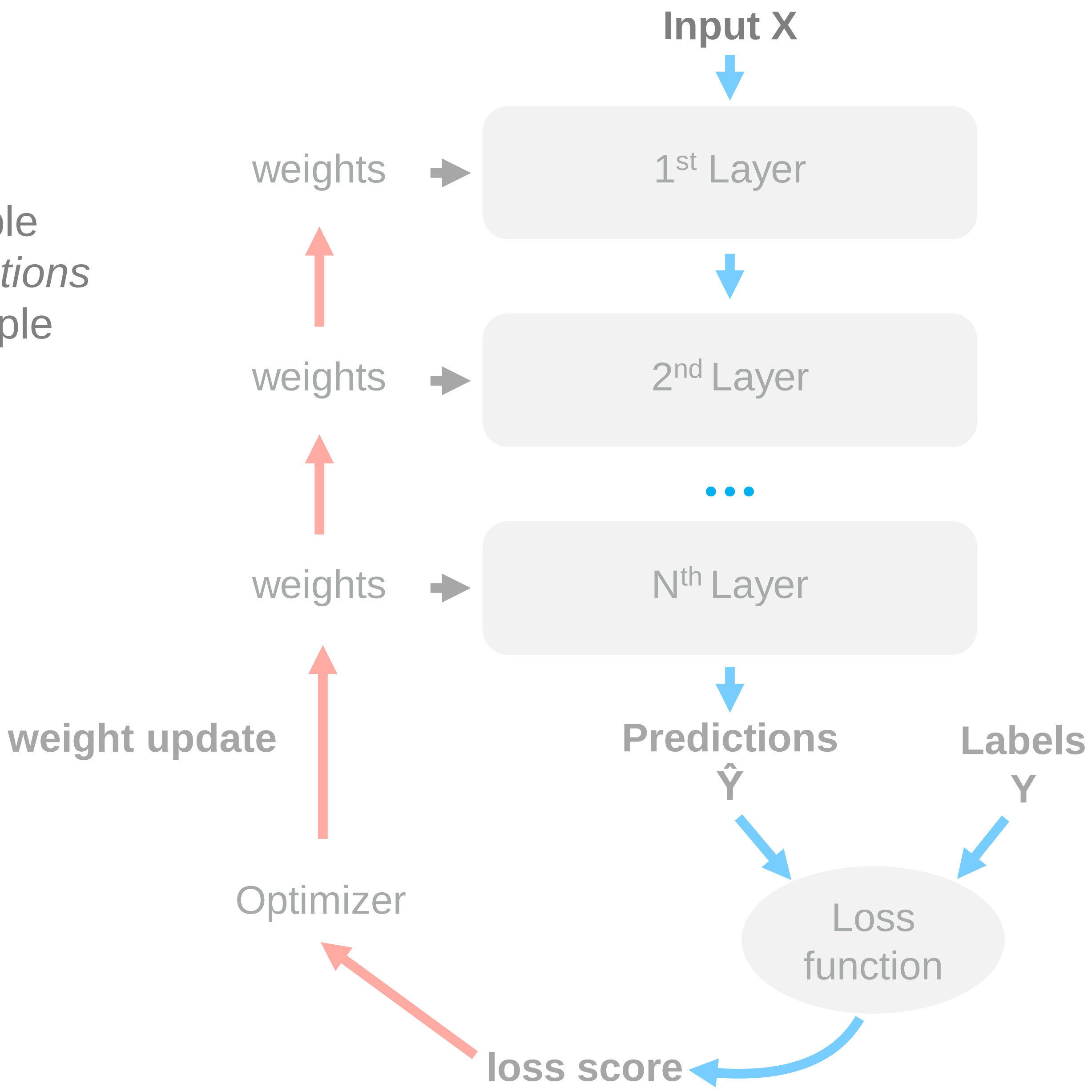
Succession of simple
linear data transformations
interleaved with simple
non-linearities



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Deep Neural Network

Succession of simple
linear data transformations
interleaved with simple
non-linearities

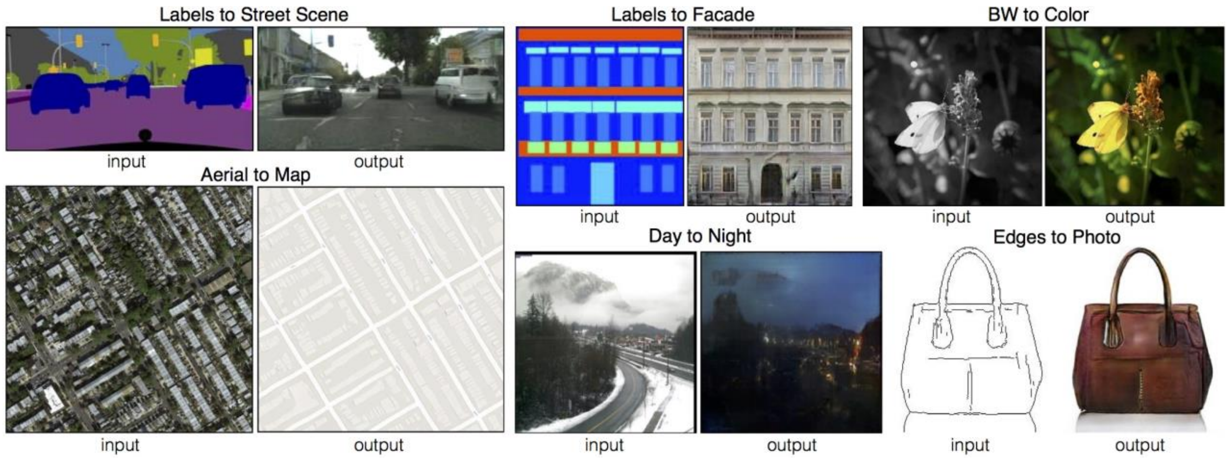


Learning transfer functions for simulating precipitation fields

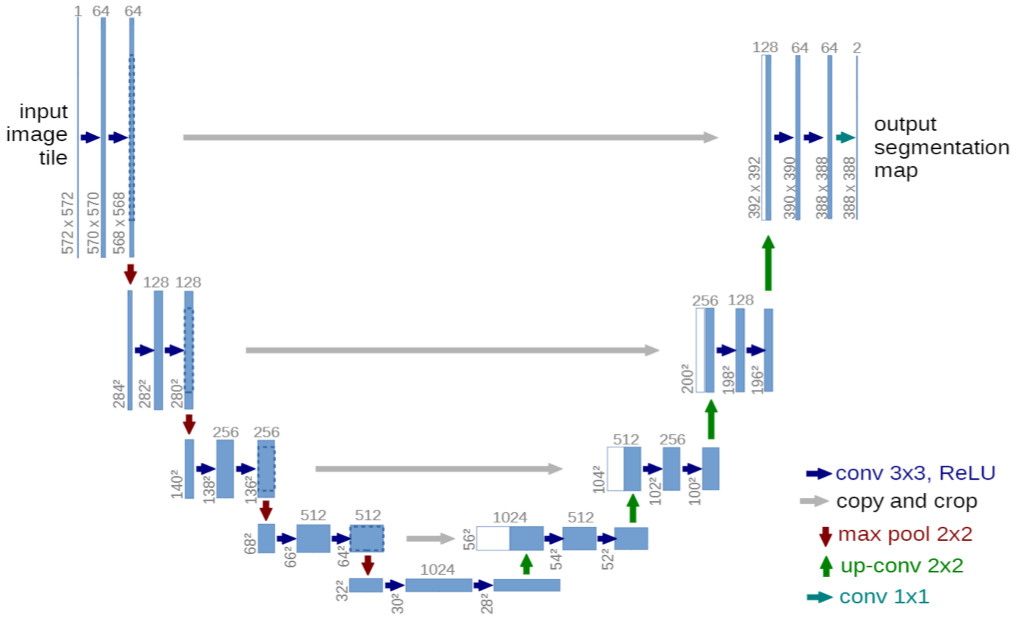
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Image to image translation

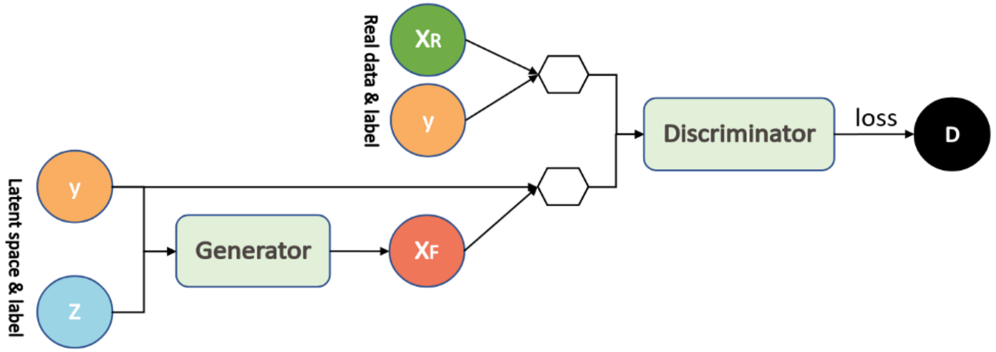
Learning the mapping (transfer function) between an input image and an output image (or between data modalities)



Isola et al. 2017



U-NET, Ronneberger et al. 2015

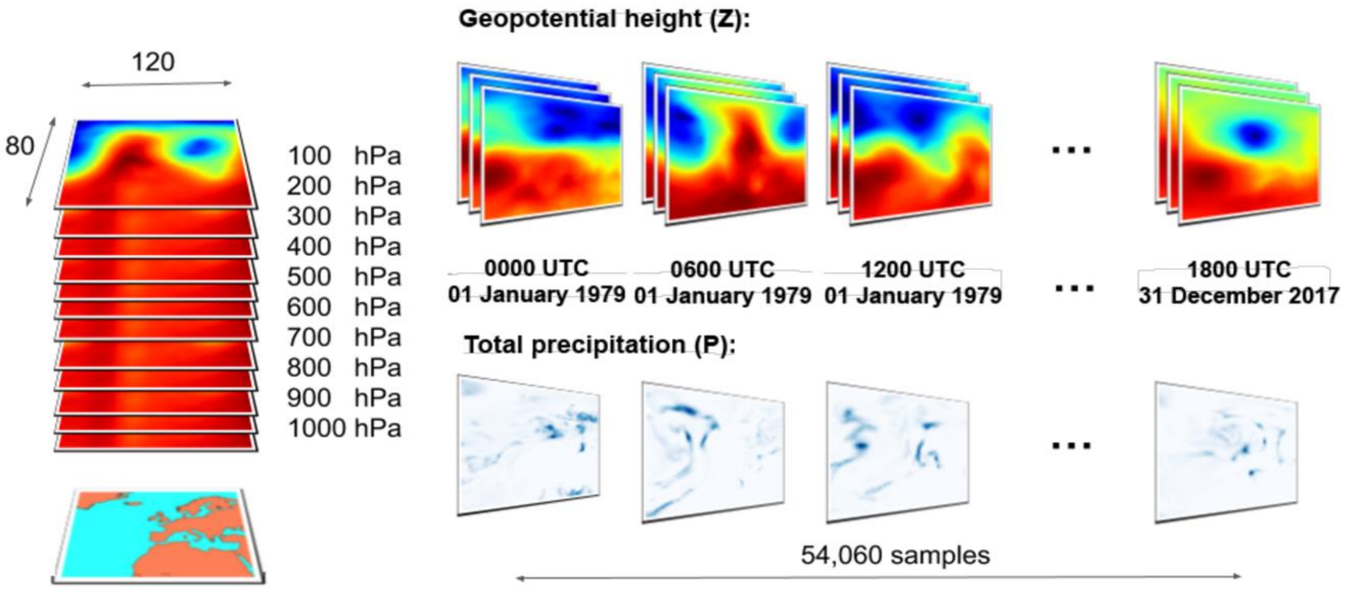


cGAN, Mirza & Osindero 2014

... and other generative models proposed in the last few years

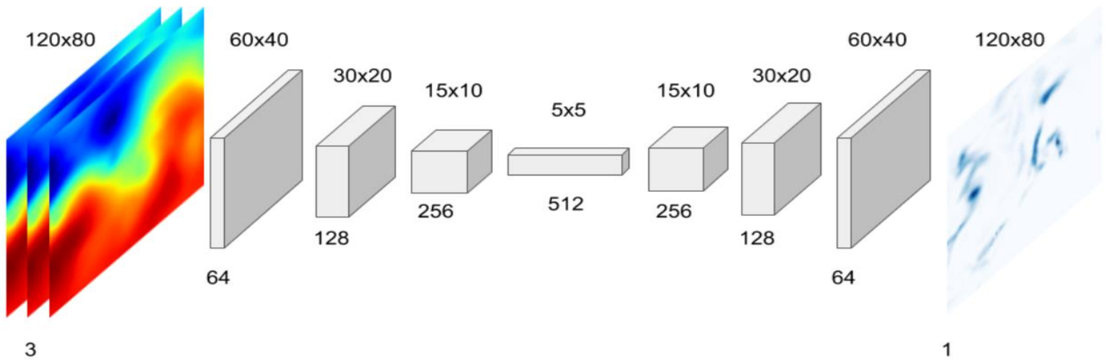
Transfer funcions for precipitation

ERA Interim geopotential



Rozas et al. 2019

"A data-driven approach to precipitation parameterizations using convolutional encoder-decoder neural networks"



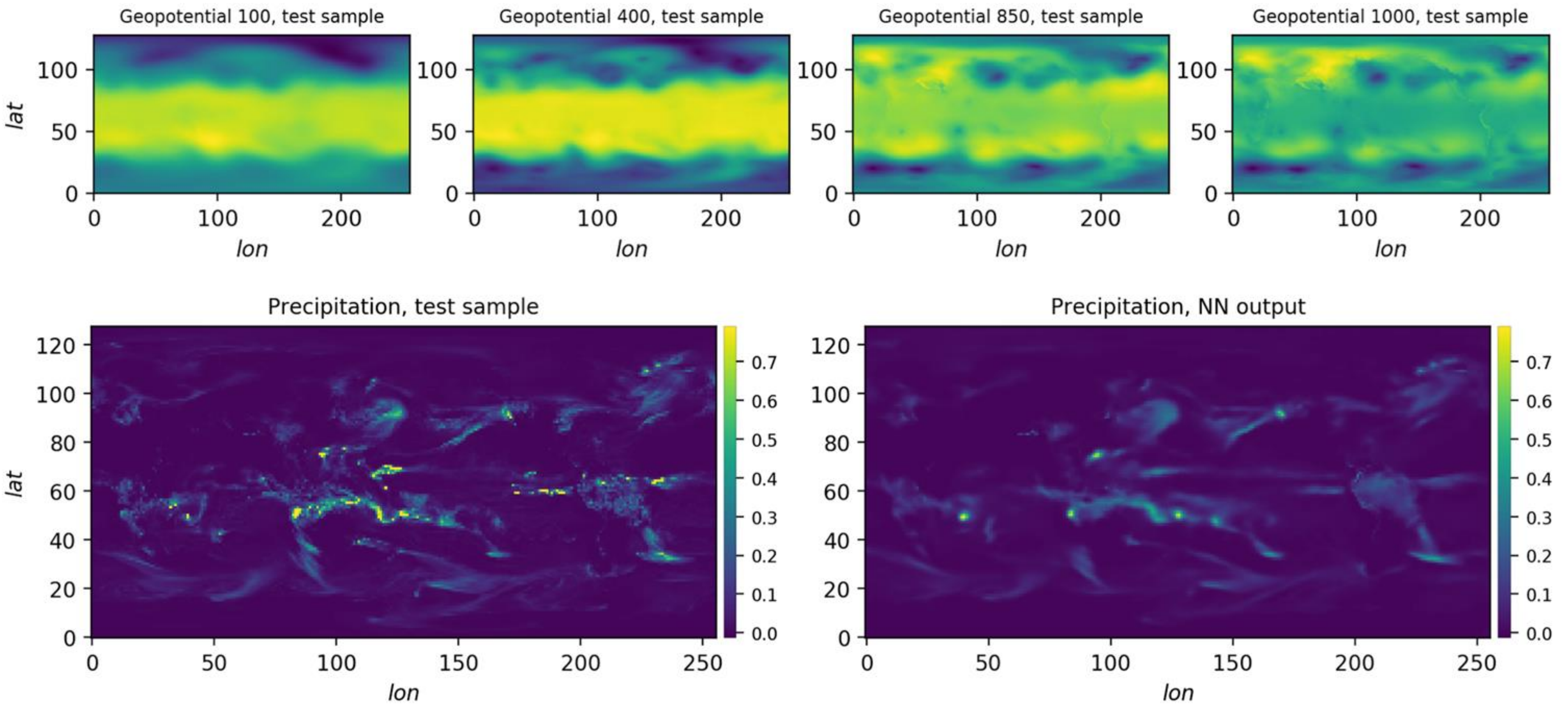
Models for I2I translation tested:

Segnet, VGG16, U-NET

ERA5 first tests

From ERA 5 (WeatherBench) geopotential to ERA 5 precipitation

X



Y

Ŷ

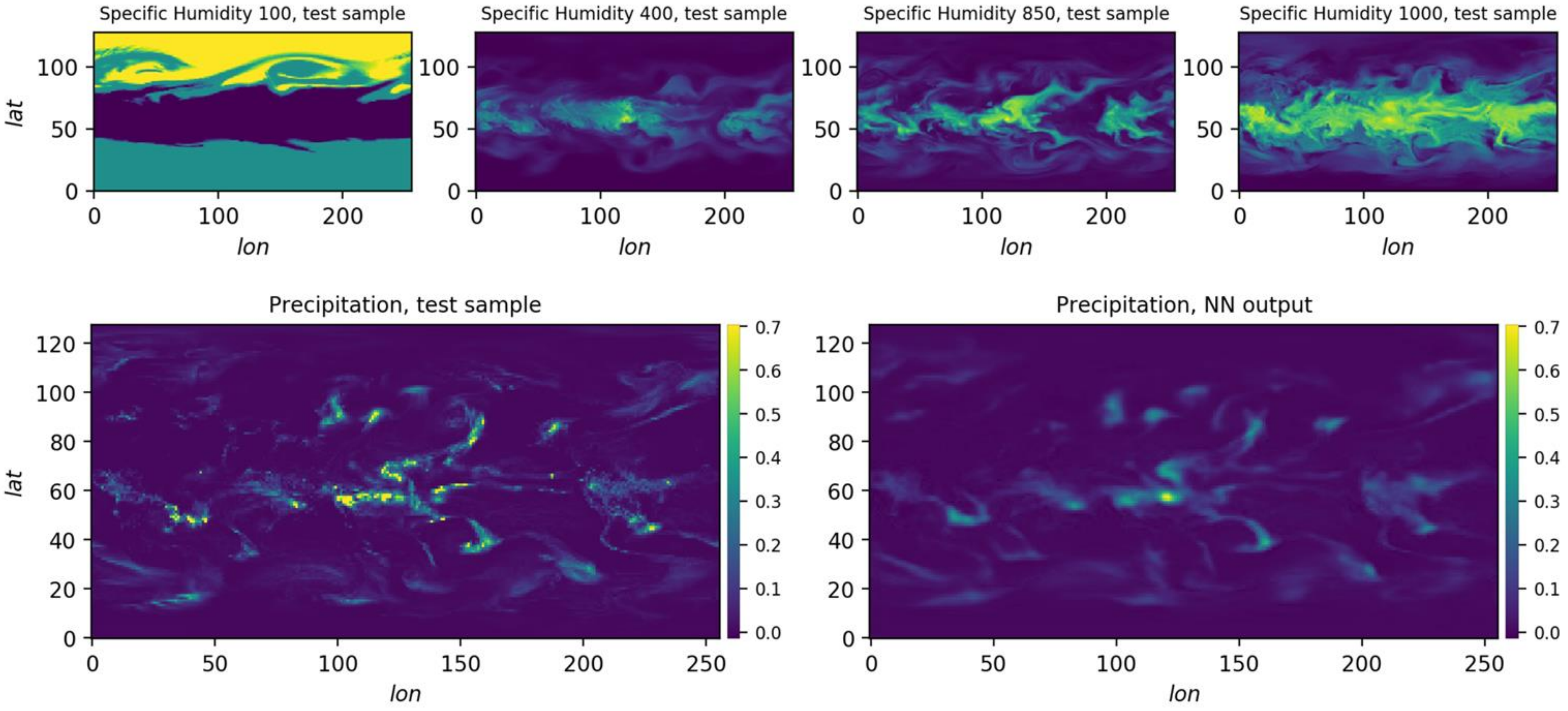
ERA5 first tests

From ERA 5 (WeatherBench) specific humidity to ERA 5 precipitation

X

Y

Ŷ

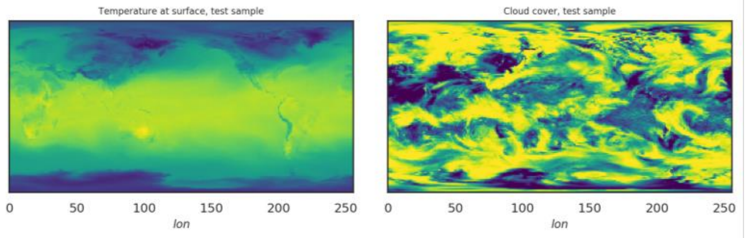


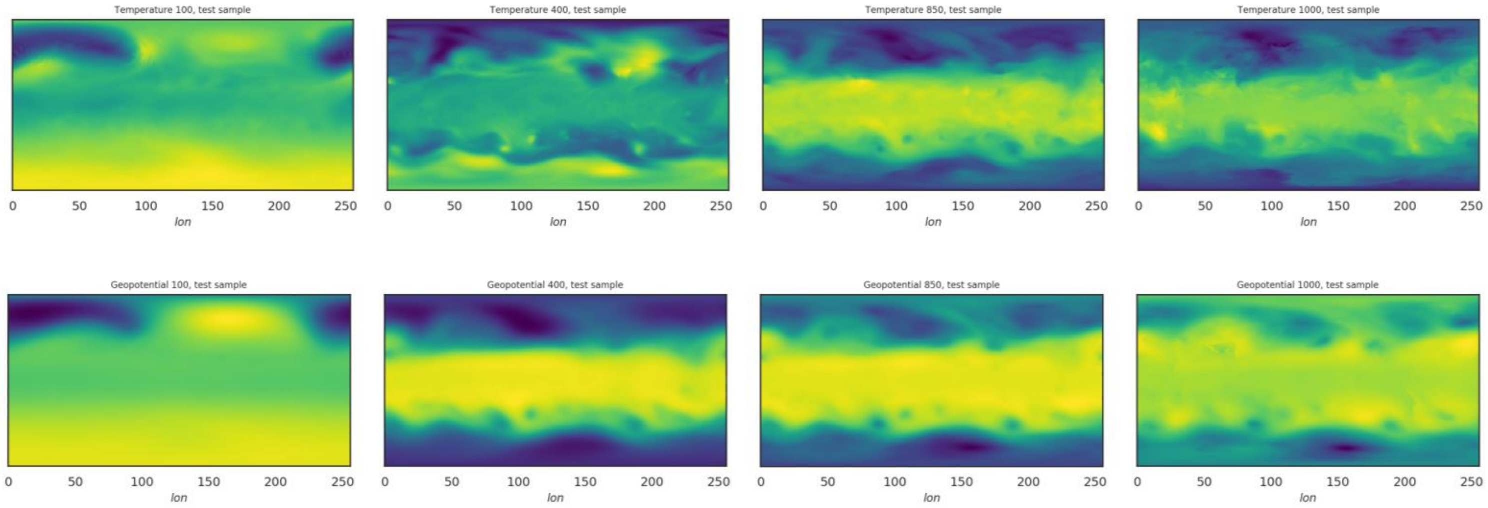
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

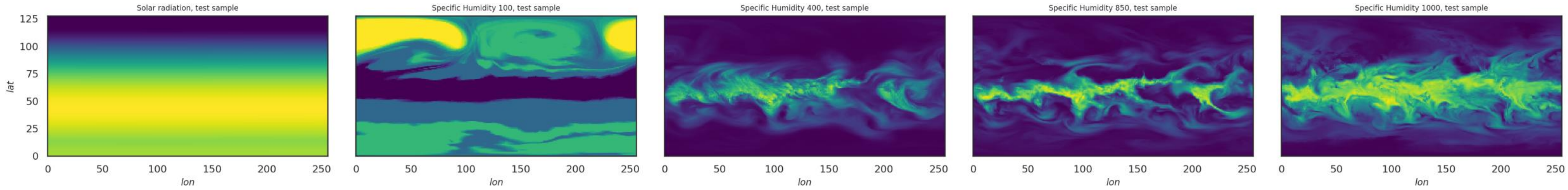
Adding ERA 5 variables

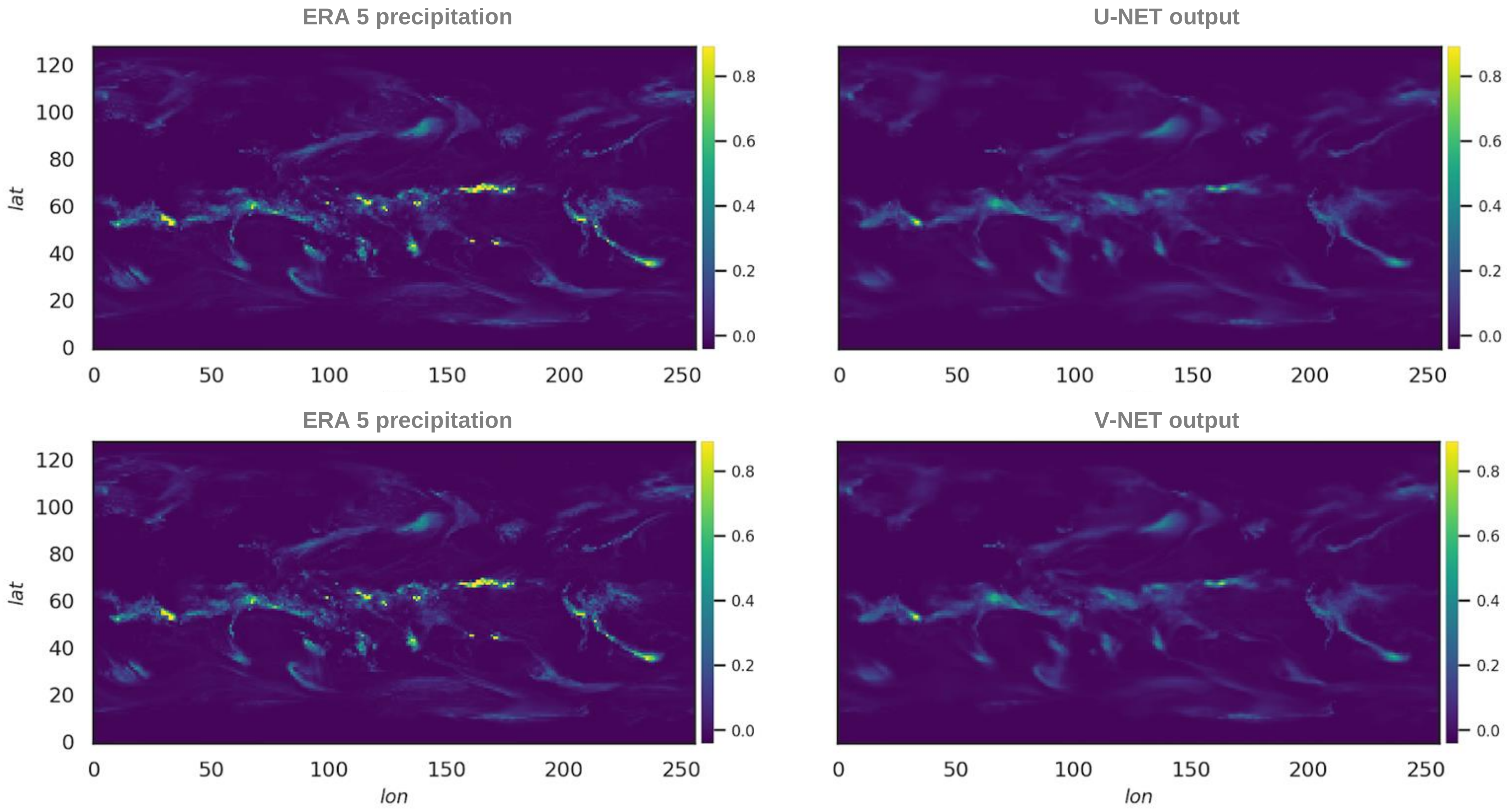
At this point, we include 15 different variables/layers:

- Temperature at Surface
- Temperature 100, 400, 850, 1000
- Cloud cover
- Geopotential 100, 400, 850, 1000
- Specific Humidity 100, 400, 850, 1000
- Solar radiation





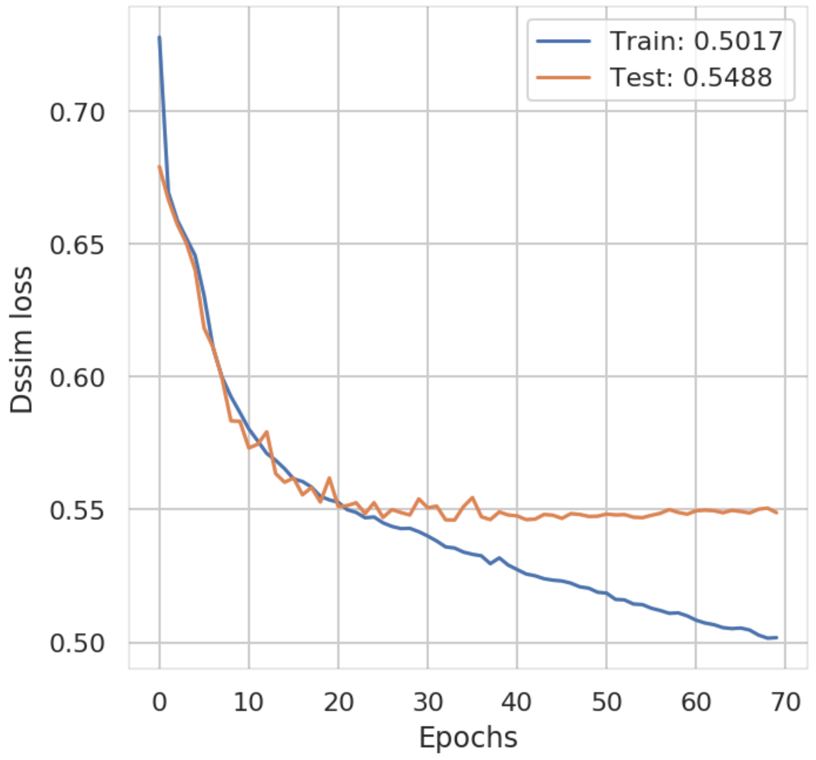




Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Distributed deep learning

- Training using the BSC CTE-Power 9 cluster, using the 4 V100 GPUs of a single node
- Escalable to multiple nodes




```python
from custom_unet import custom_unet
from train_model import traininig_logic
import tensorflow as tf

# strategy = tf.distribute.MirroredStrategy(['/gpu:0', '/gpu:1'])
strategy = tf.distribute.MirroredStrategy()

print ('Number of devices: {}'.format(strategy.num_replicas_in_sync))

BUFFER_SIZE = len(x_train)
BATCH_SIZE_PER_REPLICA = 64
GLOBAL_BATCH_SIZE = BATCH_SIZE_PER_REPLICA * strategy.num_replicas_in_sync

with strategy.scope():
    model_mae = custom_unet(x_train[0].shape, filters=32,
                            use_batch_norm=True, dropout=0.3,
                            dropout_change_per_layer=0.0,
                            num_layers=4, output_activation=None)

    traininig_logic(model_mae, x_train, y_train, x_valid, y_valid, x_test, y_test,
                    epochs=50, batchsize=GLOBAL_BATCH_SIZE, verbose=1,
                    optimizer='adam', loss='mae', lr=0.001,
                    plot='plt',
                    savetoh5=True,
                    savetoh5_path='./tmpdata/nn_VARStoPRLR_stand_Unet_mae.h5',
                    checkpoints=False,
                    checkpoint_dir='./training_checkpoints',
                    early_stopping=False, patience=5, min_delta=0.01,
                    returns=False)

    # with tf.device('/gpu:0'):
    y_test_hat_mae = model_mae.predict(x_test)
```

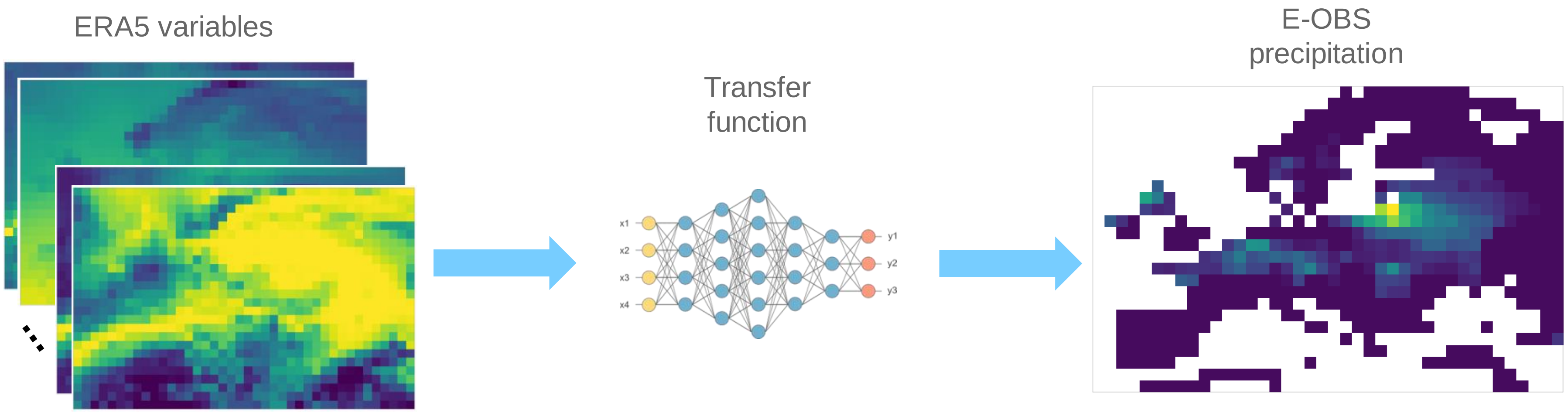
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

From ERA 5 reanalysis to E-OBS precipitation

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

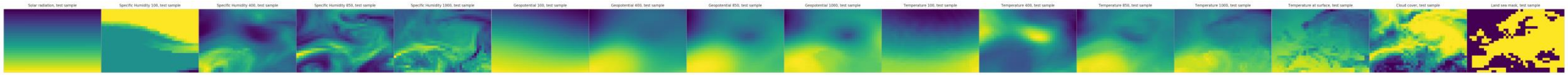
ERA 5 to E-OBS

- ERA5 reanalysis data (WeatherBench data at 1.4 deg, 1 hourly resampled to daily)
- E-OBS daily gridded precipitation (regridded to 1.4 deg)
  - Predicting the ERA5 precipitation is a rather methodological exercise
- Data from 1979 to 2018 (~14.6 k samples)
- Implementation of various models including deep neural networks for learning transfer funcions
- Comparison in terms of MSE and Pearson correlation



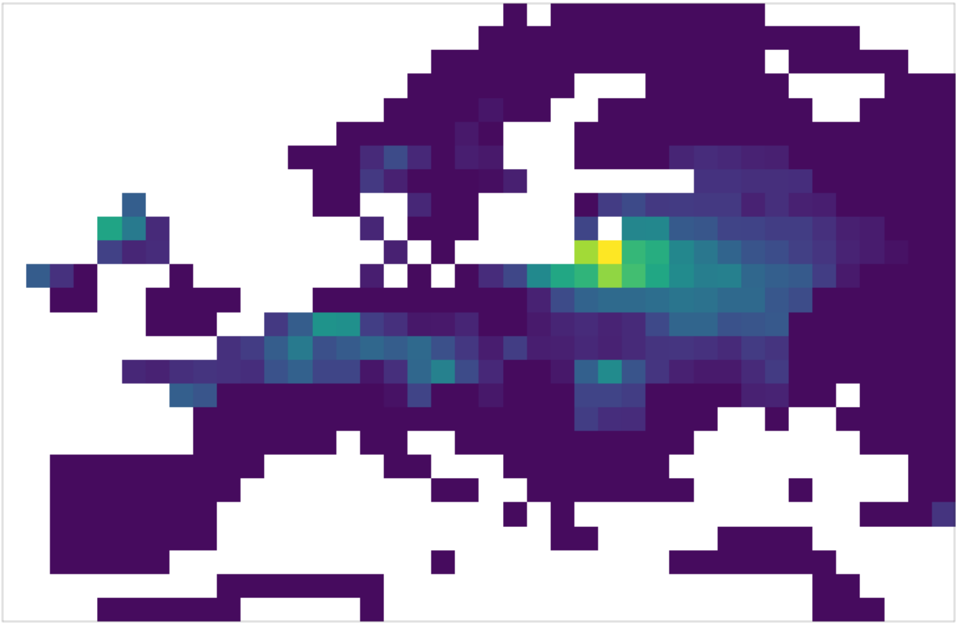
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Data



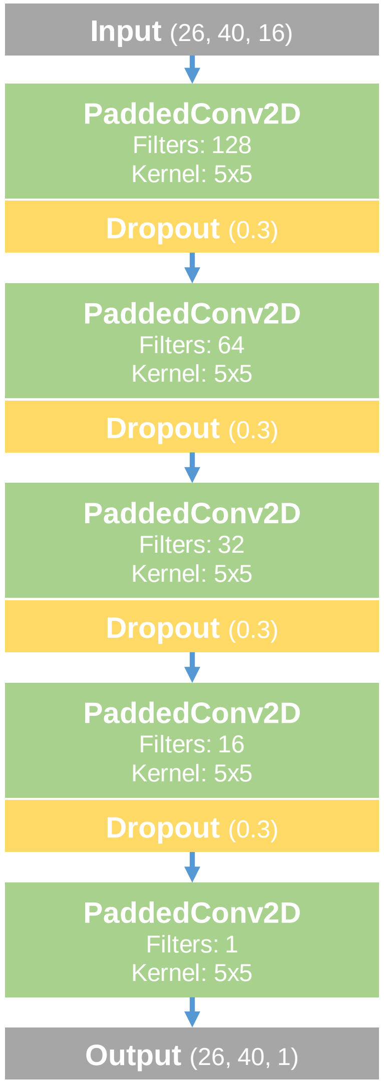
- 16 slices: 15 variables/levels plus a land sea mask (proper standardization)
- Dealing with NaNs (over the ocean) in E-OBS data:
  - NaNs to non-physical value
- ~14k samples, train/valid/test splitting
- Models, px-wise vs convolutionals:
  - Linear Regression (16 variables -> precipitation)
  - Random Forest (16 variables -> precipitation)
  - All (2D) convolutional network
  - U-NET (2D convolutions)
  - V-NET (3D convolutions)

E-OBS precipitation

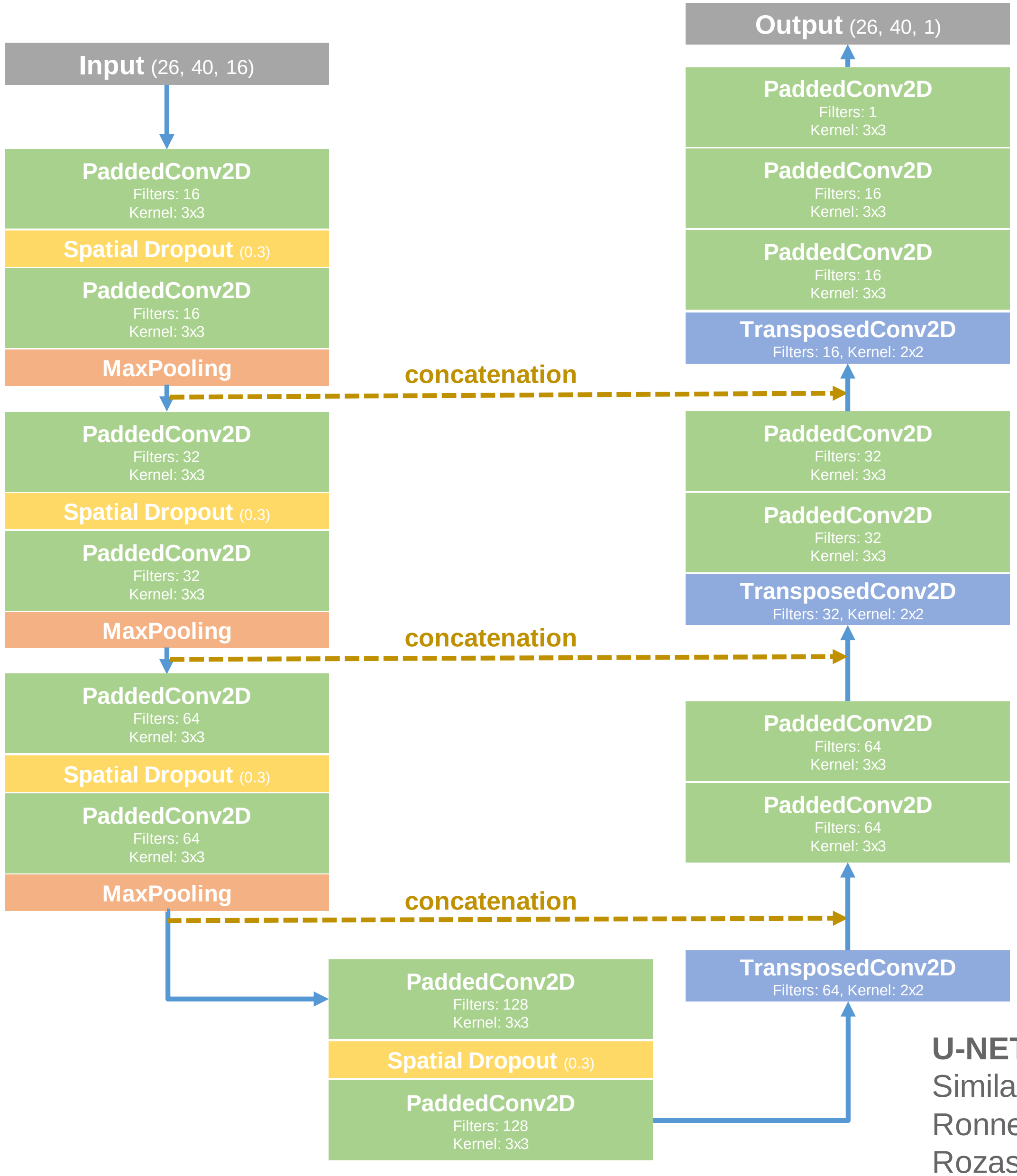


Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Models



All convolutional net
Similar to:
Springerber et al. 2015
Rasp et al. 2020



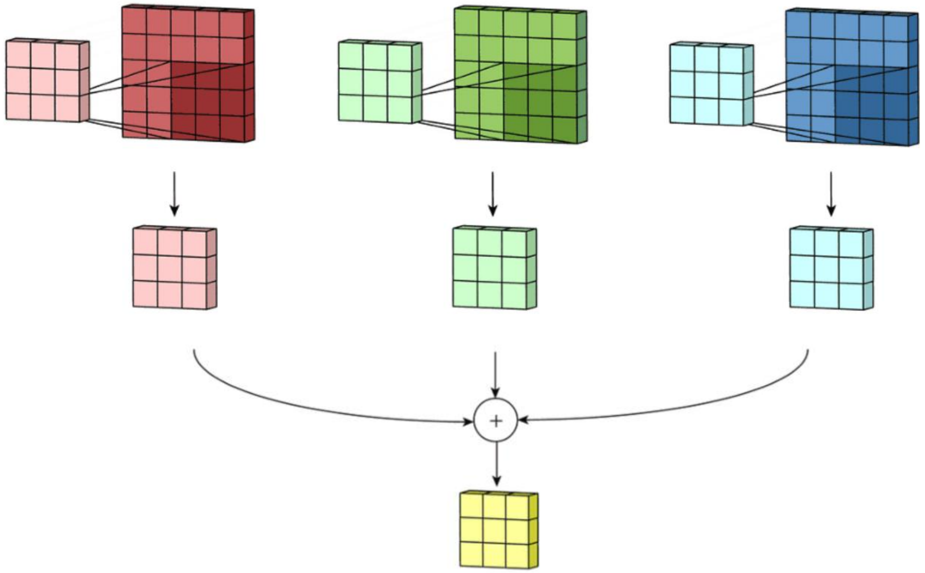
U-NET inspired network
Similar to:
Ronneberger et al. 2015
Rozas et al. 2019

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

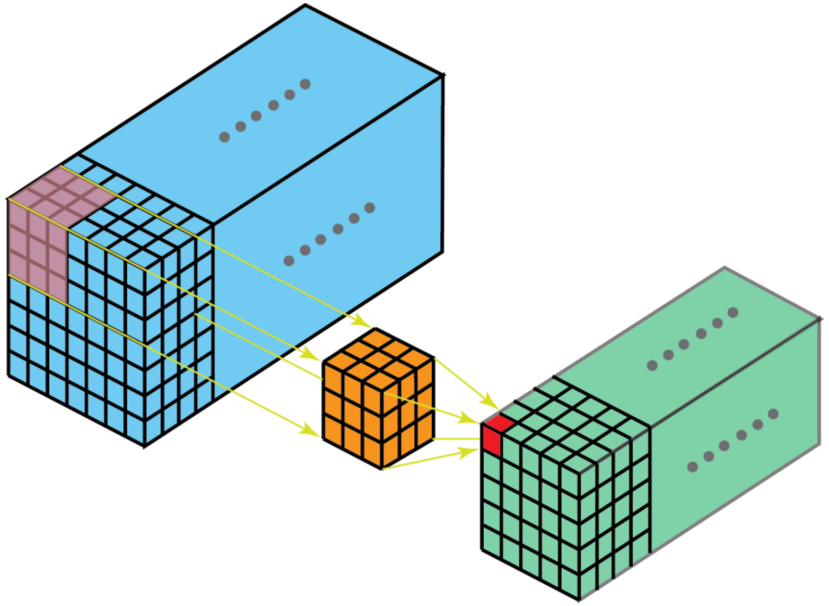
Models

V-NET (Milletari et al. 2016) similar to U-NET but using volumetric (3D) convolutions.

2D convolution with several channels (e.g., RGB)



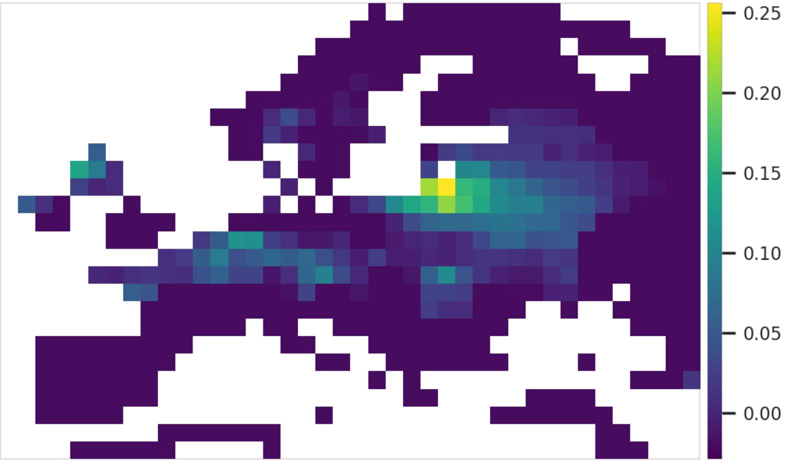
3D convolution



Tran et al. 2015

Model comparison

E-OBS ground truth
(single timestep)

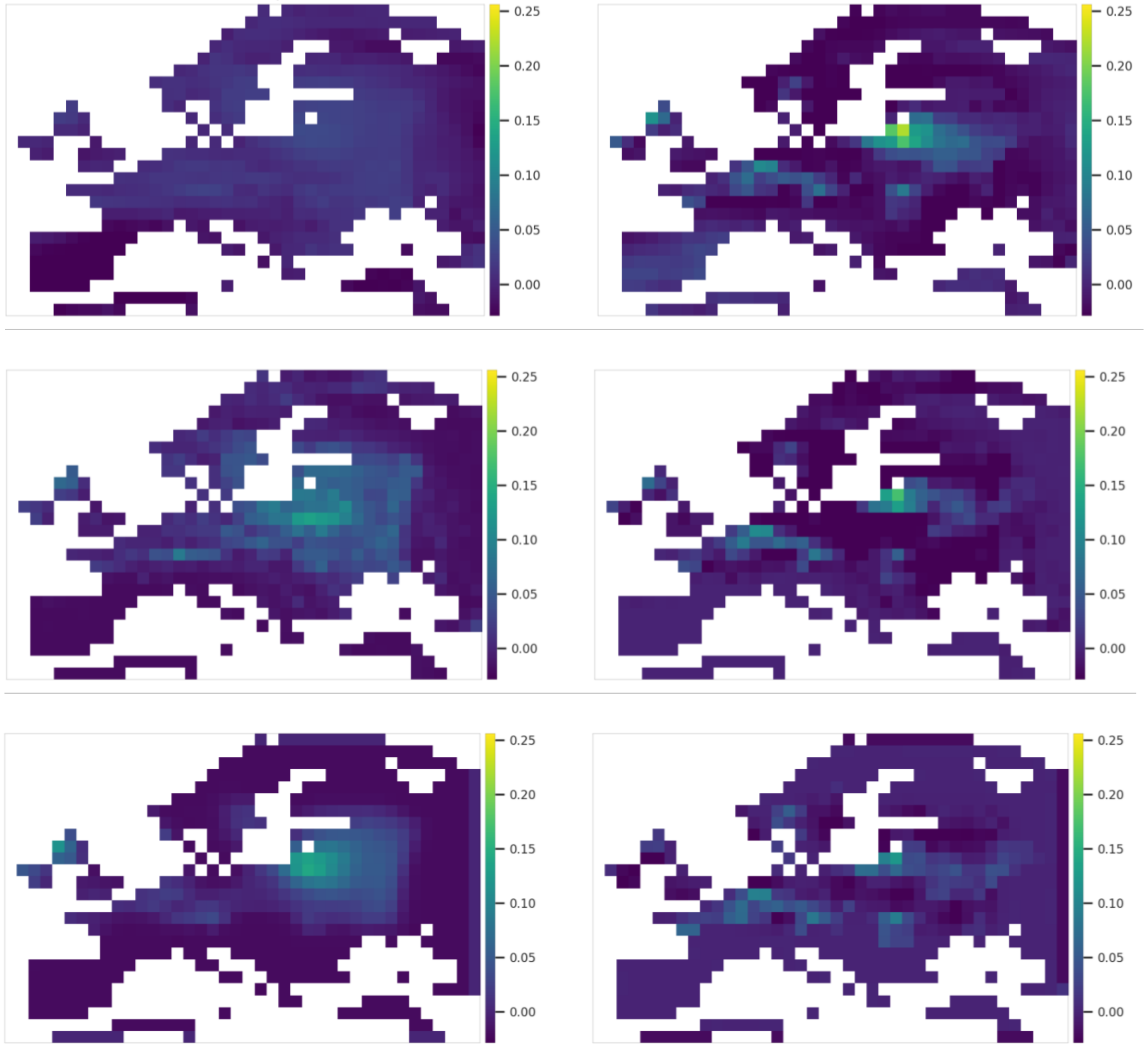


Model output | Residuals

Linear regression

Random forest regression

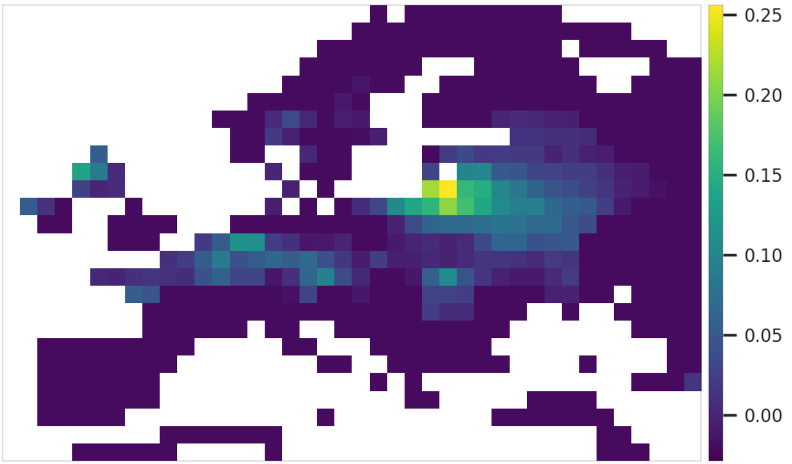
All convolutional network



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Model comparison

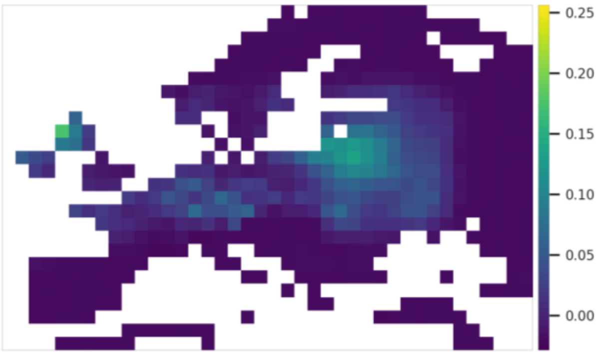
E-OBS ground truth
(single timestep)

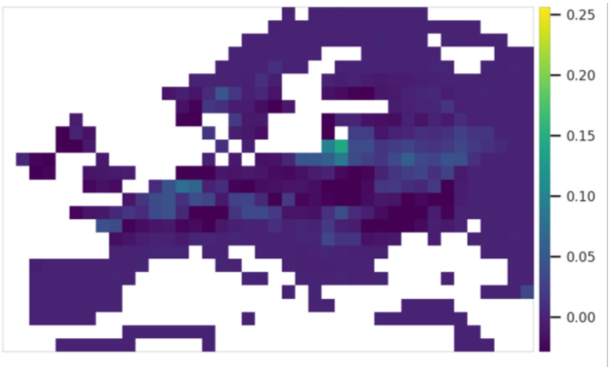


Model output

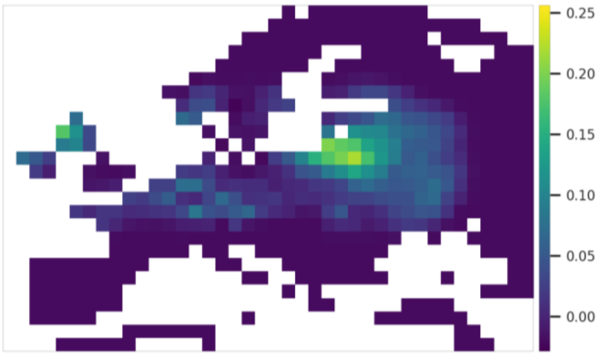
Residuals

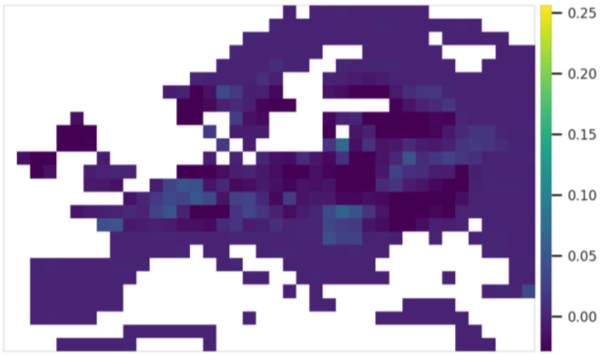
U-NET





V-NET

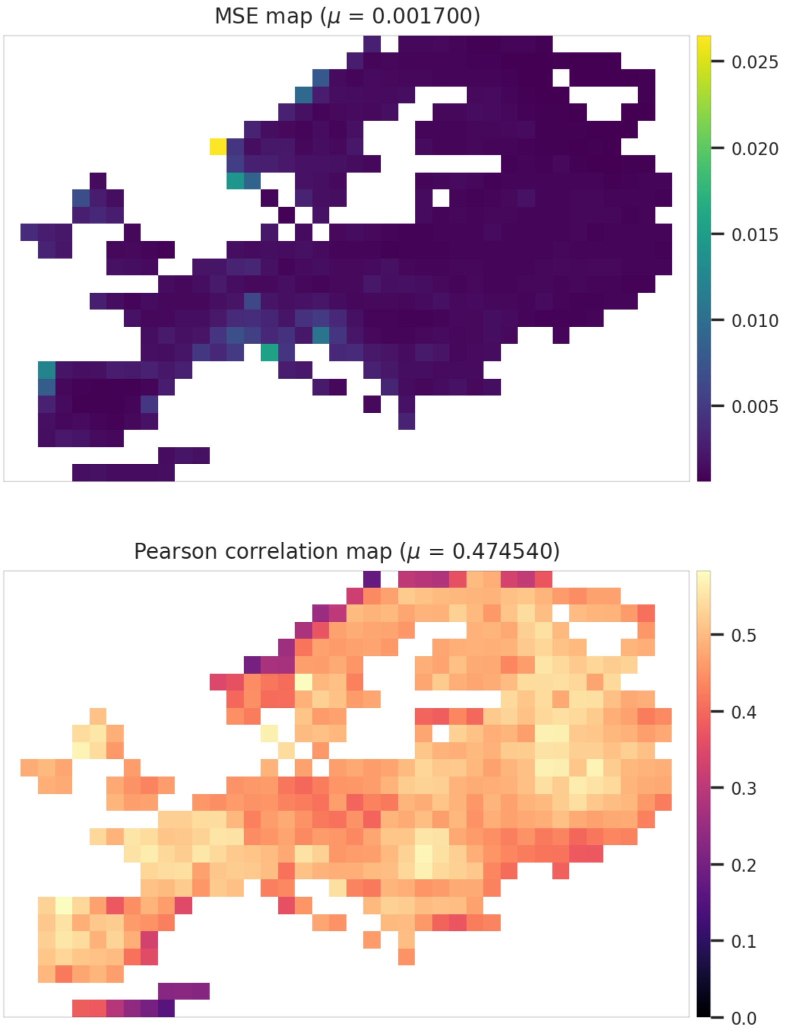




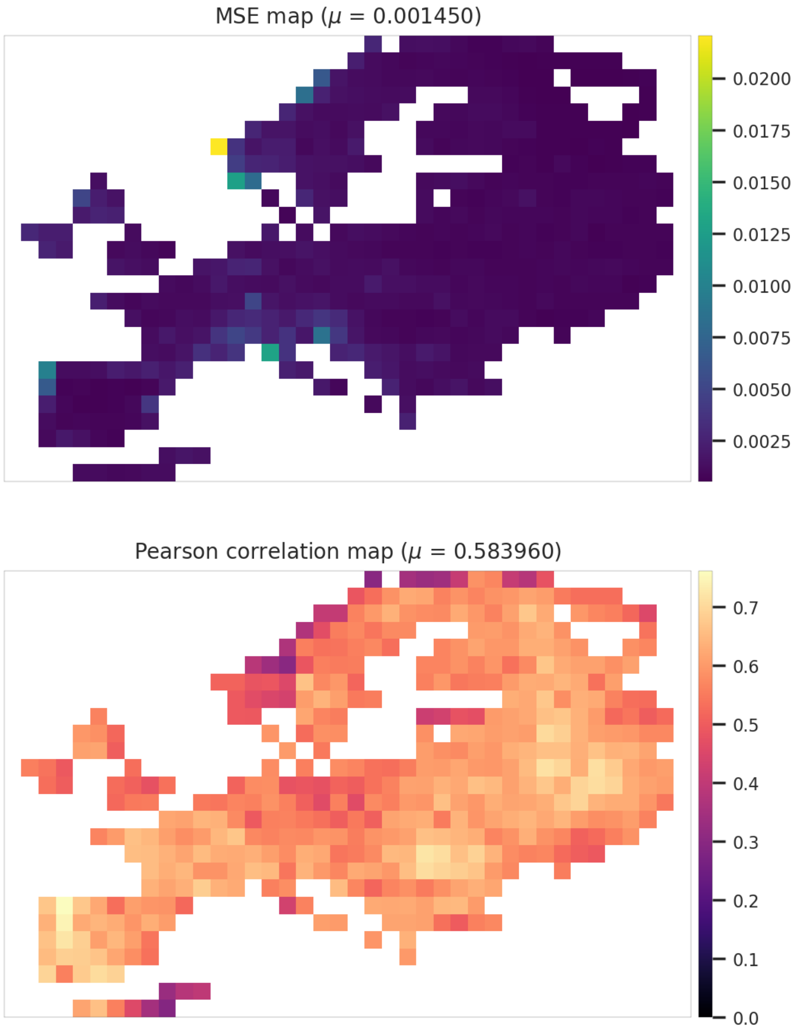
Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Model comparison

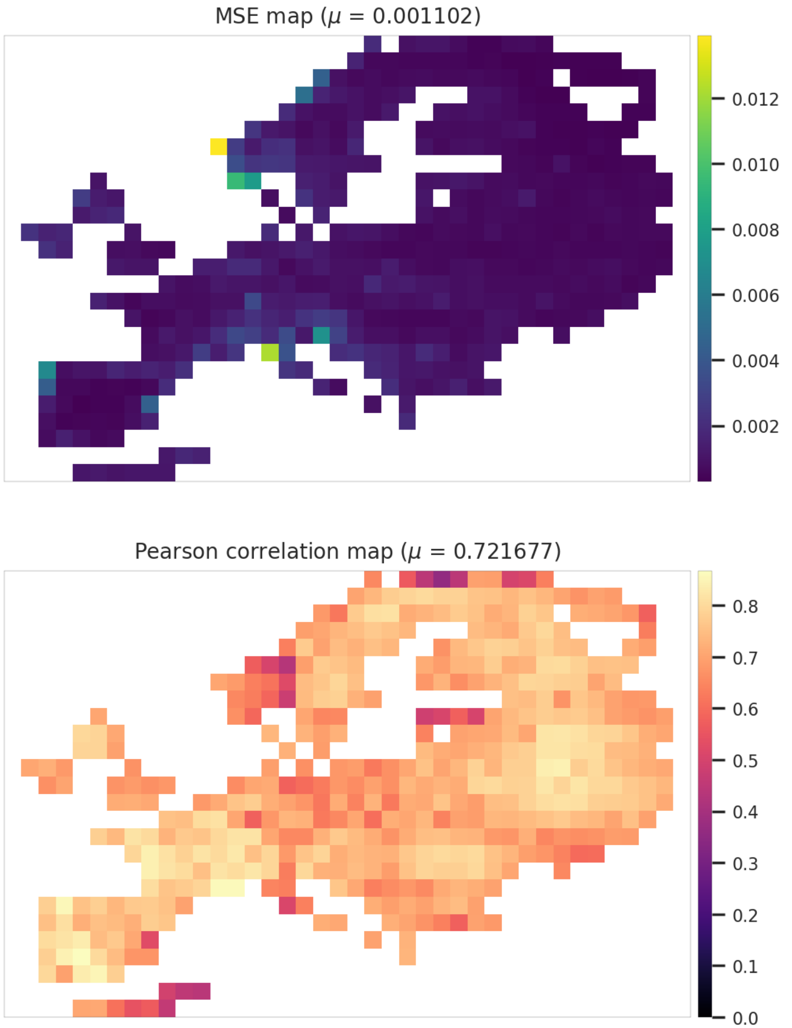
Linear Regression



Random Forest



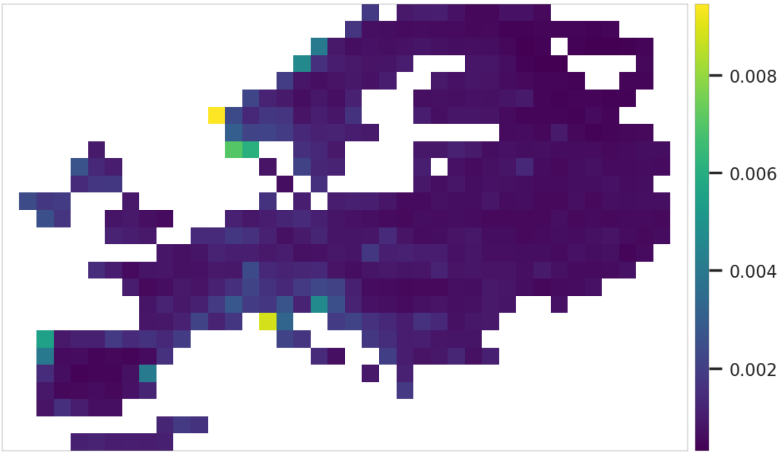
All convolutional network



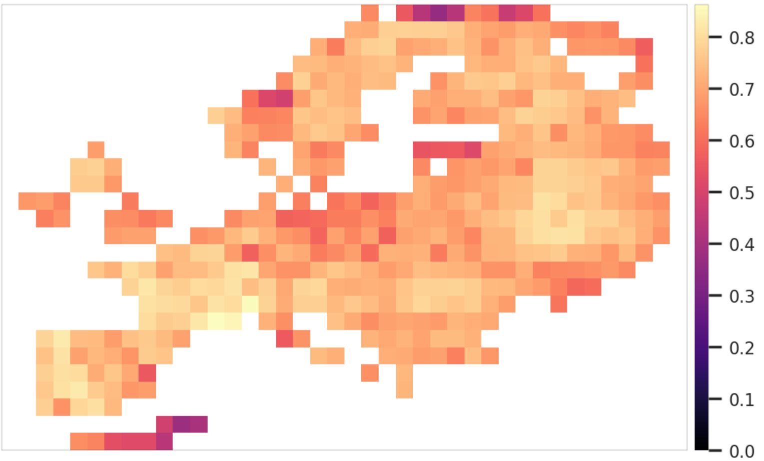
Model comparison

U-NET

MSE map (μ = 0.001036)

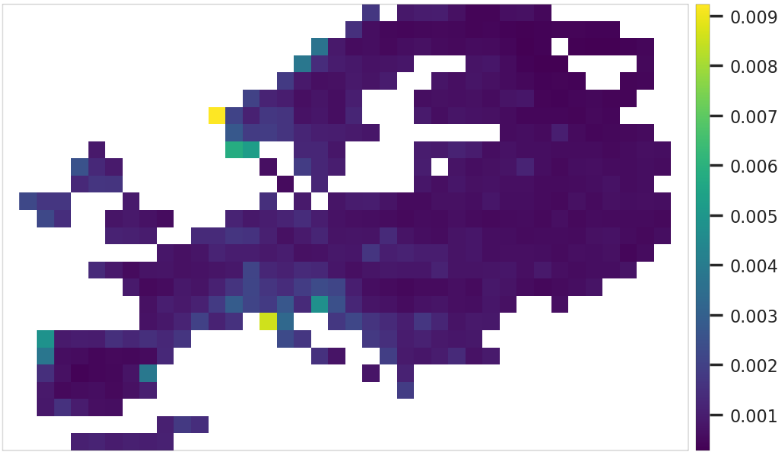


Pearson correlation map (μ = 0.707368)

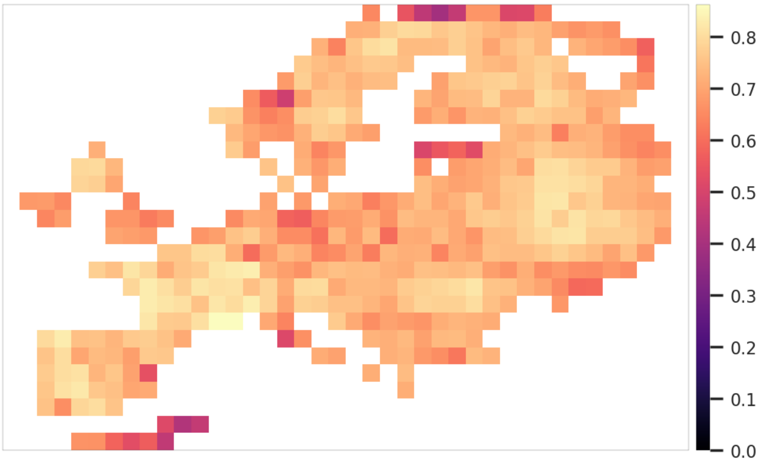


V-NET

MSE map (μ = 0.000973)



Pearson correlation map (μ = 0.721204)



Model comparison

Model	MSE	Pearson correlation	
Linear regression	1.70E-03	0.47	
Random forest regression	1.45E-03	0.58	
All convolutional network	1.10E-03	**0.72**	(~320 k pars)
U-NET	1.04E-03	0.71	(~500 k pars)
V-NET	**9.73E-04**	**0.72**	(~1.4 M pars)

Conclusions and next steps

- Deep neural networks (in a supervised context) yield impressive results on I2I tasks using NWP fields
- Same experiments with 0.25 deg E-OBS precipiation and ERA 5 variables
- Different strategies for exploiting multiple variables more independently
- Compare current results with generative models (conditional GANS)
- Validation with external observational precipitation data
- Downscaling
  - ERA 5 at 14 deg -> E-OBS original 0.25 degree resolution (Baño-Medina et al. 2019)
  - Use the sparse station measurements
- Forecasting
  - Use lead time to forecast future states (almost for free)
  - Global precipitation data?

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

Thank you

carlos.gomez@bsc.es