Emulation of wildland fire spread simulation for danger mapping using deep learning

Frédéric Allaire[‡] **J.B. Filippi**[†] **Vivien Mallet**[‡]

[†] SPE, CNRS/Università di Corsica [‡] INRIA, Paris

GDR Feux, 4 Décembre 2020 Nancy-berspace

Motivation

Mass simulation and danger

- Wildland fire spread is yet useful for an active fire
- To account for danger with spread simulation would require to test every potential ignition point at any time
- At 1 simulations per Ha, 100 for ensembles
- 80 millions simulation for Corsica, 1 billion for 24 hours.
- Too costly to generate daily maps in operational context using spread models

Emulation of simulations with deep learning

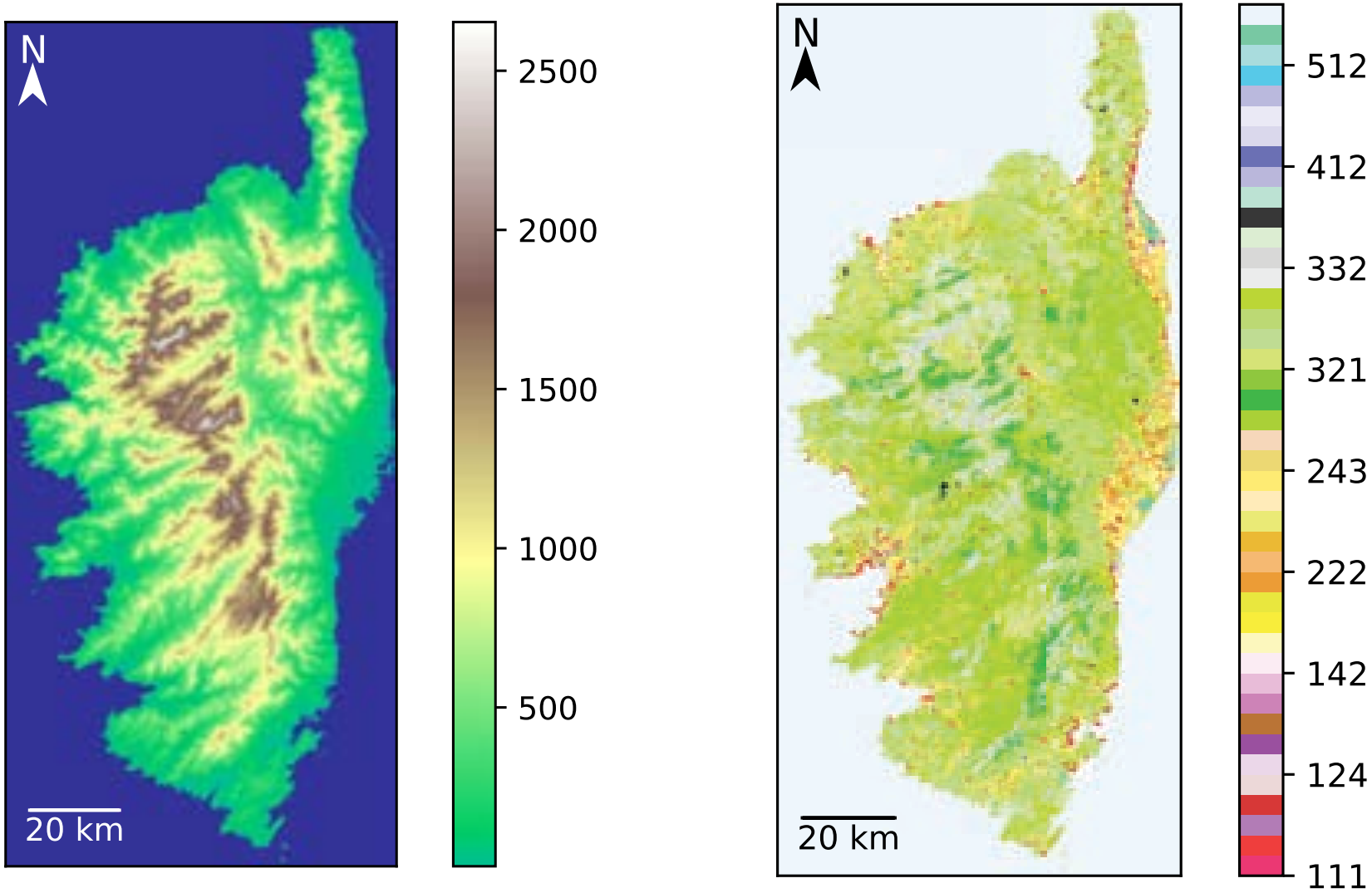
From wildfire spread simulations to danger

ForeFire simulation

- Emulated model - ForeFire.
- Fire igniting anywhere in Corsica and **spread freely** during one hour.
- All potential weather conditions

FIGURE – Example of an operational ForeFire simulation, Bigiglia 2017.

Neural Network inputs



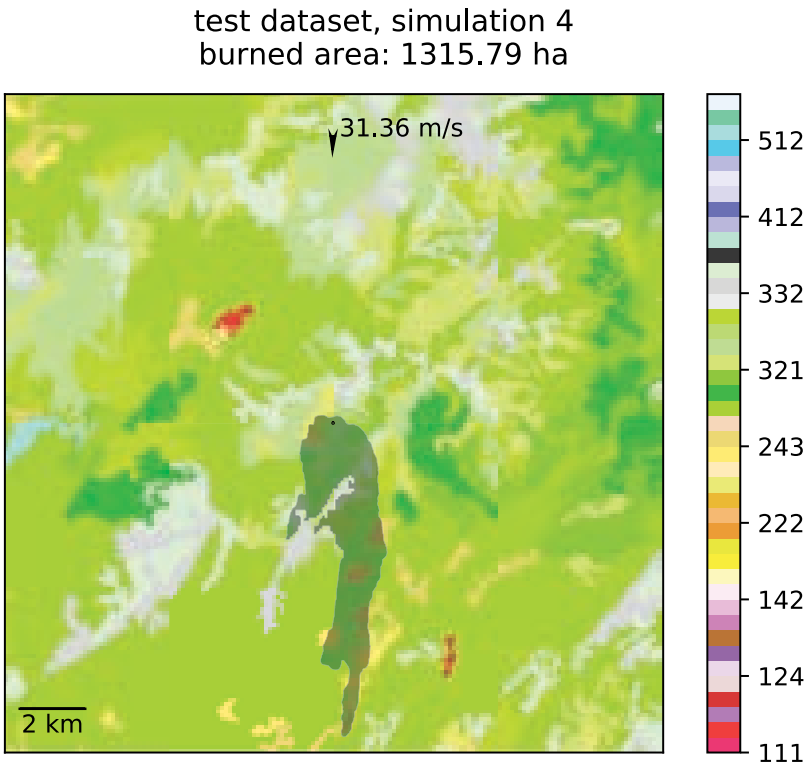
Gridded Input Data maps of Corsica used to describe the landscape in ForeFire simulations ; their spatial resolution is approximately 80 m.

Neural Network Construction

Input	Symbol	Unit	Type	Range t
Ignition point coordinates	(x, y)	m	Raw	Map of Corsica
Wind speed	(W_x, W_y)	$\mathrm{m\,s^{-1}}$	Raw	$[-35, 35]^2$
Fuel moisture content (dead fuel)	m_c		Raw	$[0.04, 0.3]$
Heat of combustion perturbation	$\blacksquare H$	$\mathrm{MJ\,kg^{-1}}$	Additive	$[-5, 5]$
Particle density perturbation	ρ_p	$\mathrm{kg\,m^{-3}}$	Additive	$[-300, 300]$
Fuel height perturbations	h	m	Multiplicative	$[0.4, 1.6]^{13}$
Fuel load perturbations	σ_f	$\mathrm{kg\,m^{-2}}$	Multiplicative	$[0.4, 1.6]^{13}$
Surface-volume ratio perturbations	S_v	$\mathrm{m^{-1}}$	Multiplicative	$[0.4, 1.6]^{13}$

Scalar Inputs In the case of perturbations, the symbol corresponds to the perturbed quantity, and the perturbation of this quantity can be either additive or multiplicative. The range indicates the boundaries of the domain of definition with two components for the wind and 13 components in the last three rows (one row per fuel type).

Neural Network Construction



Simulated burned surface after one hour returned by ForeFire.
The initial firefront of 0.45 ha in black at the center
final burned surface is the surrounding shaded shape. The input wind speed vector is represented by the arrow at the top. The simulated fire spread to the south, was partly blocked by mountains (in gray), but still burned 1316 ha.

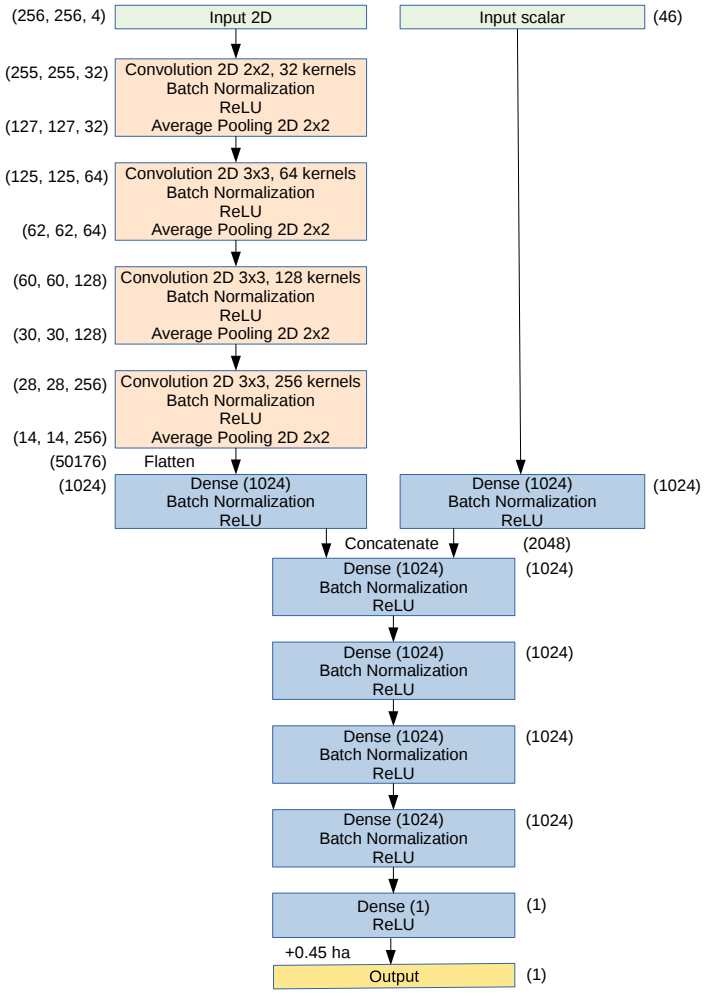
Design of experiments

Mass simulation and danger

- Consider the simulator as a “black-box”
- Build the emulator based on a synthetic dataset of input and corresponding output
- Design of experiments (DOE) to generate the datasets
- Evaluate its approximation error

large number of simulations $\sim 5.10^6$ **are required for an emulator**

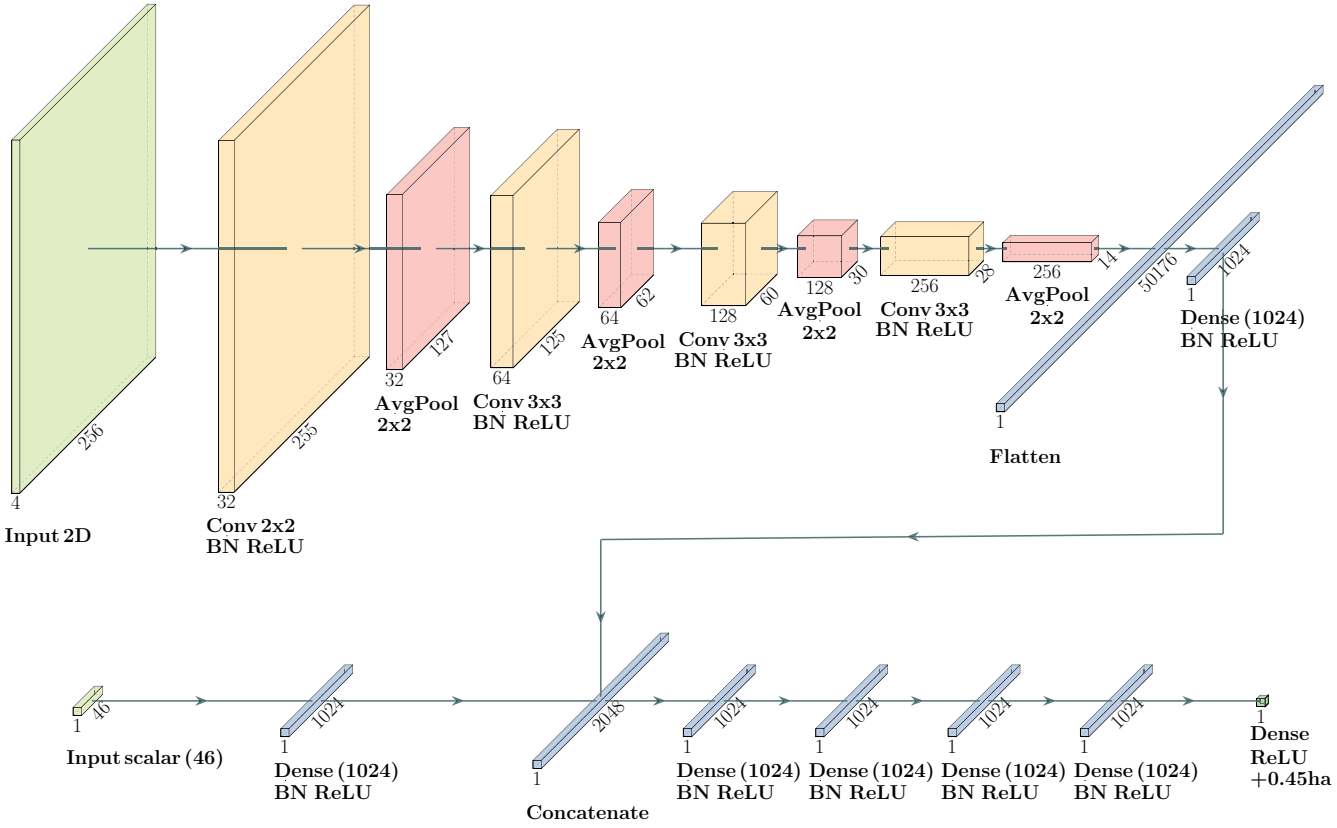
Neural network architecture



Why convolutional deep neural networks ?

Gaussian processes, polynomial chaos, high dimensional model reduction, radial basis functions are interesting alternatives, however their computational requirements (regarding time and/or memory space) can become prohibitory when there are both a high dimension ($d = 46$) and a large sample size ($\geq 10^5$)

Neural network architecture



Representation of data processing in the neural network. The blocks indicate the shape of the data. The 2D input is derived from the four fields of elevation, and fuel parameters h, σ_f, and S_v. The 46 scalar inputs are derived from the simulation parameter inputs of Table. Conv : Convolution 2D ; BN : Batch Normalization ; AvgPool : Average Pooling 2D.

Accuracy metrics and training strategy

Among a dataset of size n, $\boldsymbol{u}^i$ denotes the i-th set of simulation inputs, $y(\boldsymbol{u}^i)$ the resulting output, and $\widetilde{y}(\boldsymbol{u}^i)$ the corresponding value returned by the emulator. We use the mean absolute error (MAE), the mean absolute percentage error (MAPE) and the standardized mean square error SMSE :

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\widetilde{y}(\boldsymbol{u}^i) - y(\boldsymbol{u}^i)|, \tag{1}$$

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n} \left| \frac{\widetilde{y}(\boldsymbol{u}^i) - y(\boldsymbol{u}^i)}{y(\boldsymbol{u}^i)} \right|, \tag{2}$$

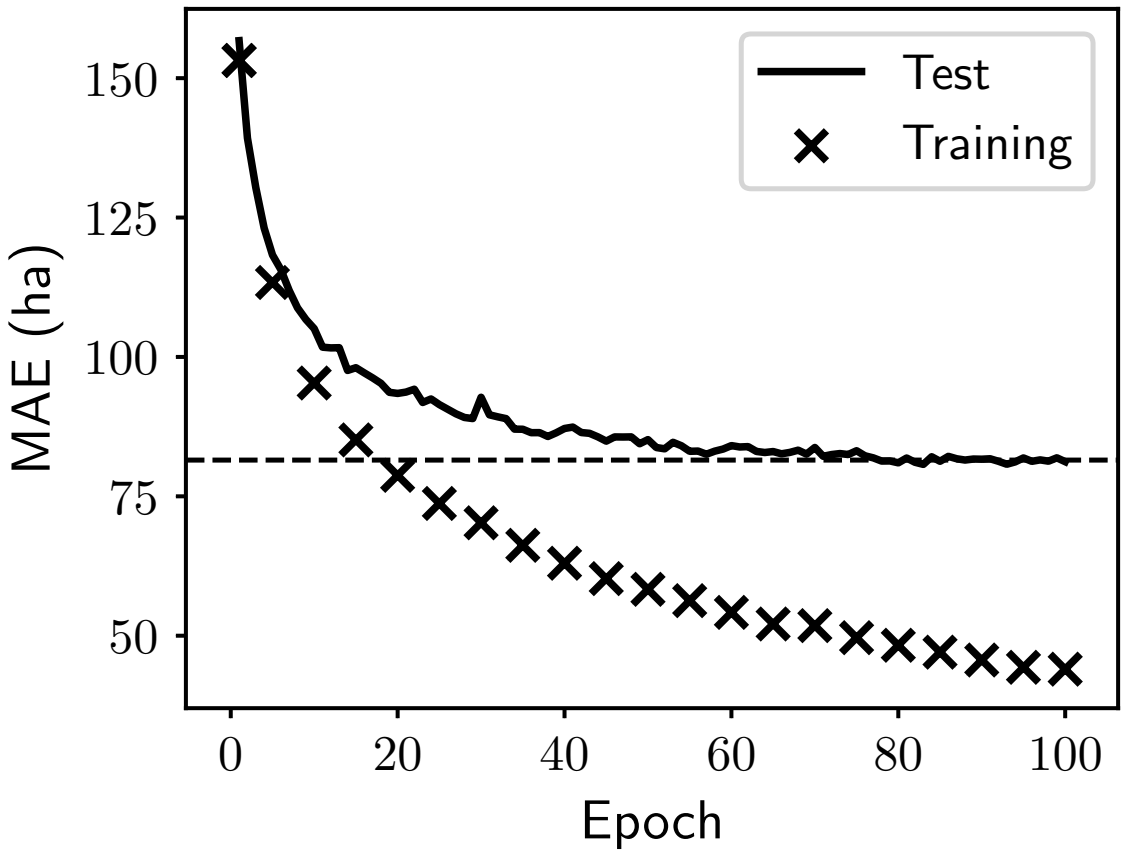
$$\text{SMSE} = \frac{\sum_{i=1}^{n} \left(\widetilde{y}(\boldsymbol{u}^i) - y(\boldsymbol{u}^i)\right)^2}{\sum_{i=1}^{n} \left(y(\boldsymbol{u}^i) - \bar{y}\right)^2}, \tag{3}$$

where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y(\boldsymbol{u}^i)$ is the sample mean of the emulated function. SMSE can be seen as a mean squared error normalized by the sample variance of y, equal to 1 if the emulator was a constant function equal to the sample mean $\bar{y}$. The lower these scores, the more accurate the emulator.

Implementation

Python scripts are used to process the data, generate the training and test datasets, build and evaluate the DNN. **Keras library**, which is a high-level neural networks API that is running on top of **TensorFlow**, is used for building the DNN.
Training and accuracy evaluation of the DNN up to the retrieval of the actual emulator are carried out on a ran on **GENCI IDRIS Jean Zay computer** (one week on 4 GPU Nodes).
The computational time of the actual emulator (**inference**) is evaluated on a **machine with 32 CPU.**
The size of the datasets are $n_{\text{train}} = 5 \times 10^6$ and $n_{\text{test}} = 10^4$. Training is carried out for 100 epochs with batches of size 400.

Learning Phase Over-Training



MAE over training. The solid curve represents the MAE for the test dataset, while the crosses represent the MAE computed for the training dataset at the end of the first epoch and after every five epochs starting from the fifth. The horizontal dotted line corresponds to MAE=81.5 ha.

Performance of emulator - training

Mean	Std	Minimum	Q1	Median	Q3	Maximum
455.7 ha	782.0 ha	0.45 ha	52.6 ha	181.0 ha	517.7 ha	24 804.4 ha

TABLE – Simulated burned surface area on training dataset of size 5×10^6.

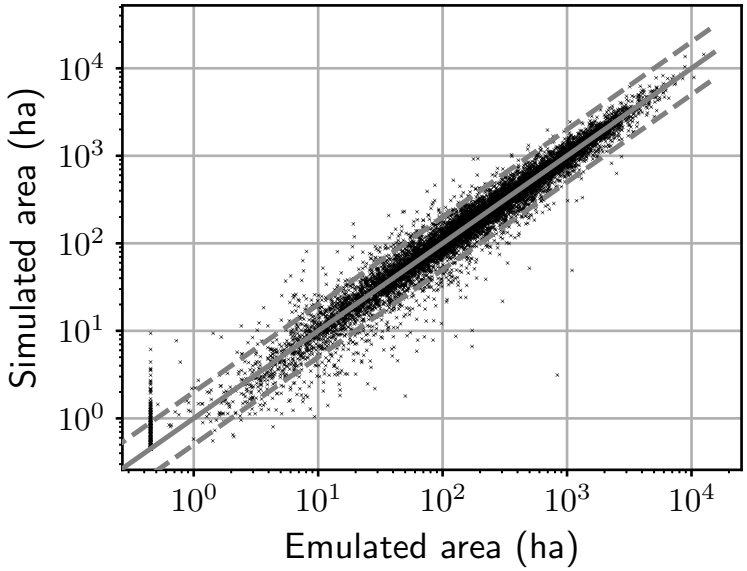
Model \ Metric	MAE	MAPE	SMSE	Bias
Mean of training	461.5 ha	2139%	100.0%	0 ha
DNN after 100 epochs	44.0 ha	23.8%	1.2%	−7.6 ha
Emulator (from DNN after 94 epochs)	45.1 ha	23.2%	1.2%	−0.9 ha

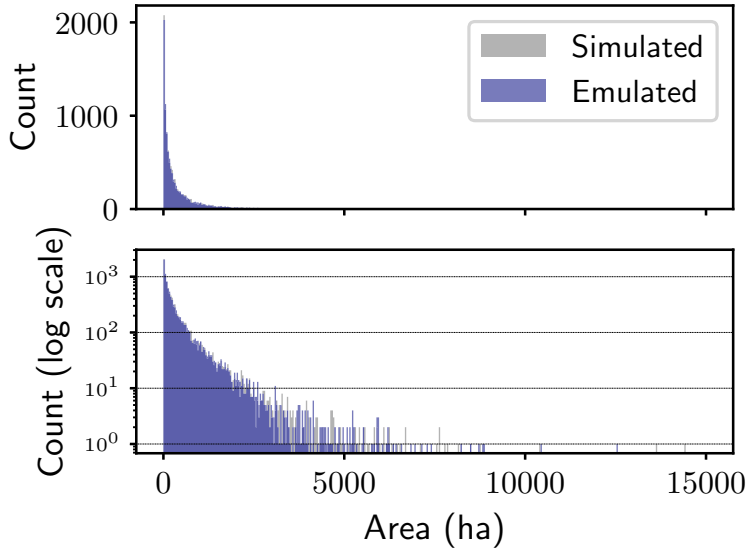
TABLE – Model error on training dataset of size 5×10^6.

Model \ Metric	MAE	MAPE	SMSE	Bias
Mean of training	461.9 ha	2266.0%	100.0%	2.2 ha
DNN after 100 epochs	81.2 ha	33.5%	6.2%	−13.1 ha
Emulator (from DNN after 94 epochs)	81.2 ha	32.8%	**6.0%**	−6.5 ha

TABLE – Model error on test dataset of size 10^4.

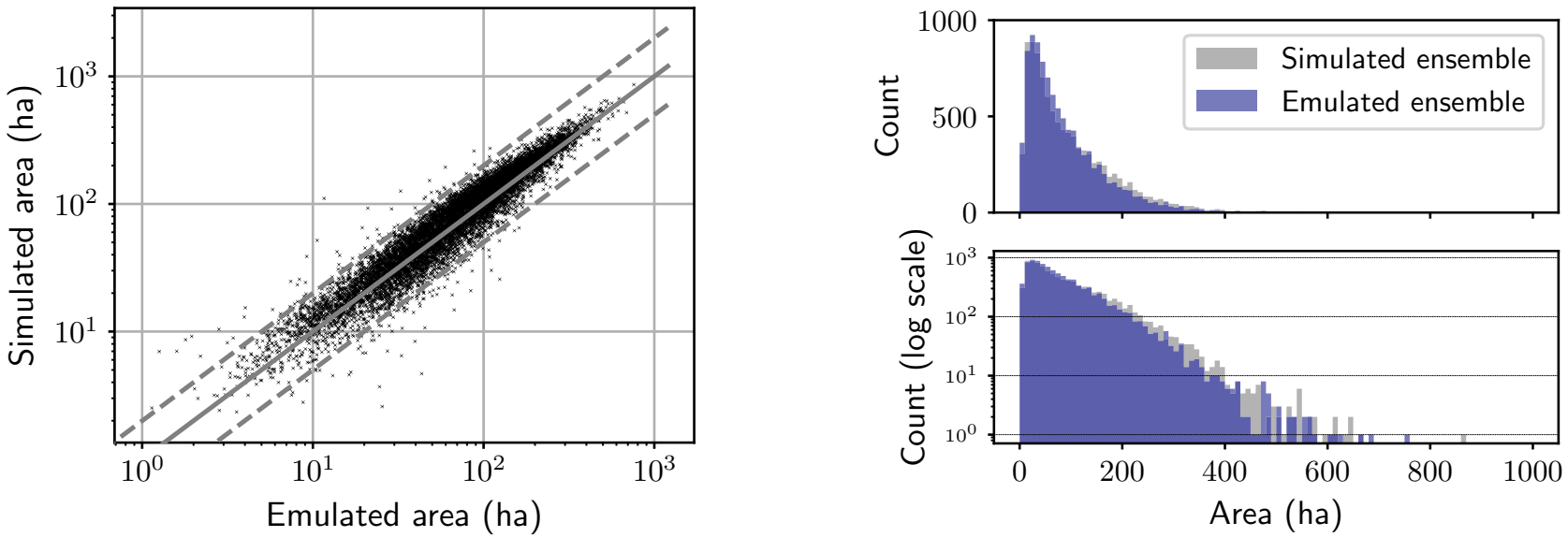
Emulated vs Simulated distributions





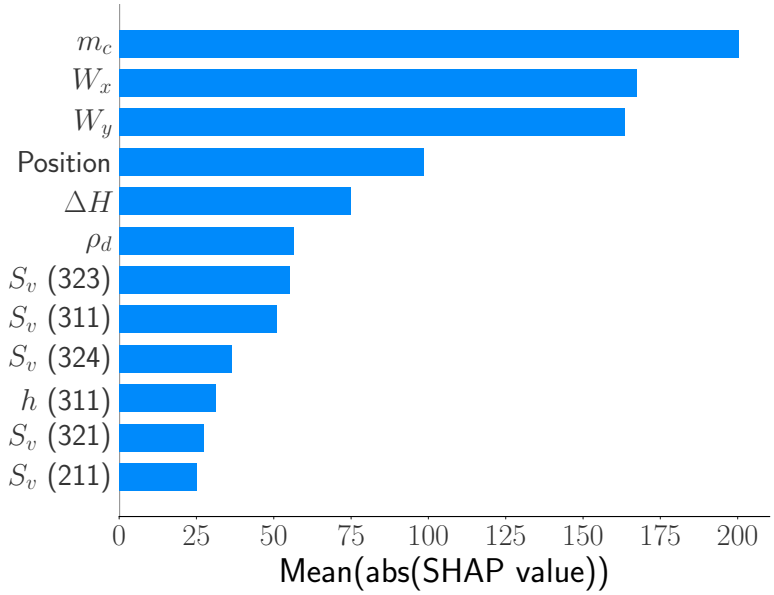
Comparison over the test dataset of size 10^4.
(a) The solid oblique gray line corresponds to a perfect match and the dotted lines correspond to an error by a factor of 0.5 and 2.
(b) Light gray : simulated area ; blue : emulated area. Both top and bottom figures represent the same distributions, they share the same abscissa axis but the bottom figure has its ordinate in log scale.

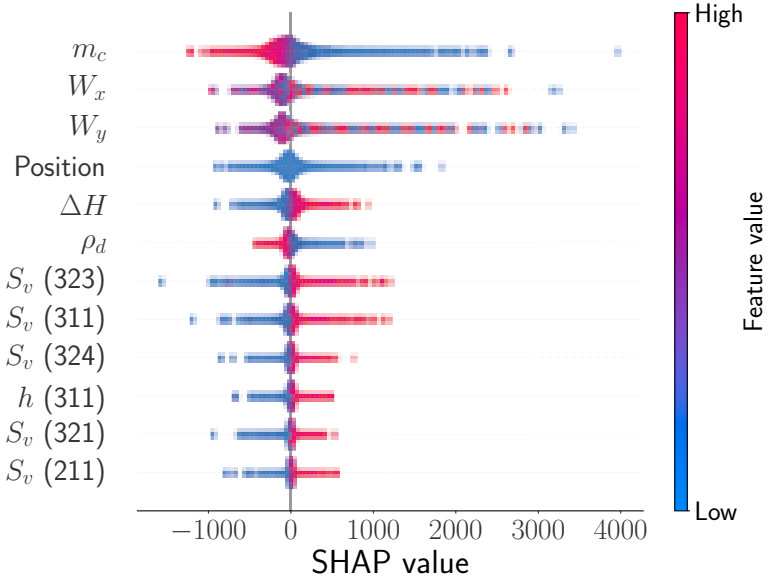
Emulated vs Simulated a specific case/event



Case of Calenzana 2017 and their emulated counterparts.
(a) The solid gray line corresponds to a perfect match and the dotted lines correspond to an error by a factor of 0.5 and 2.
(b) Light gray : simulated area ; blue : emulated area. Both top and bottom figures represent the same distributions, they share the same abscissa axis but the bottom figure has its ordinate in log scale.

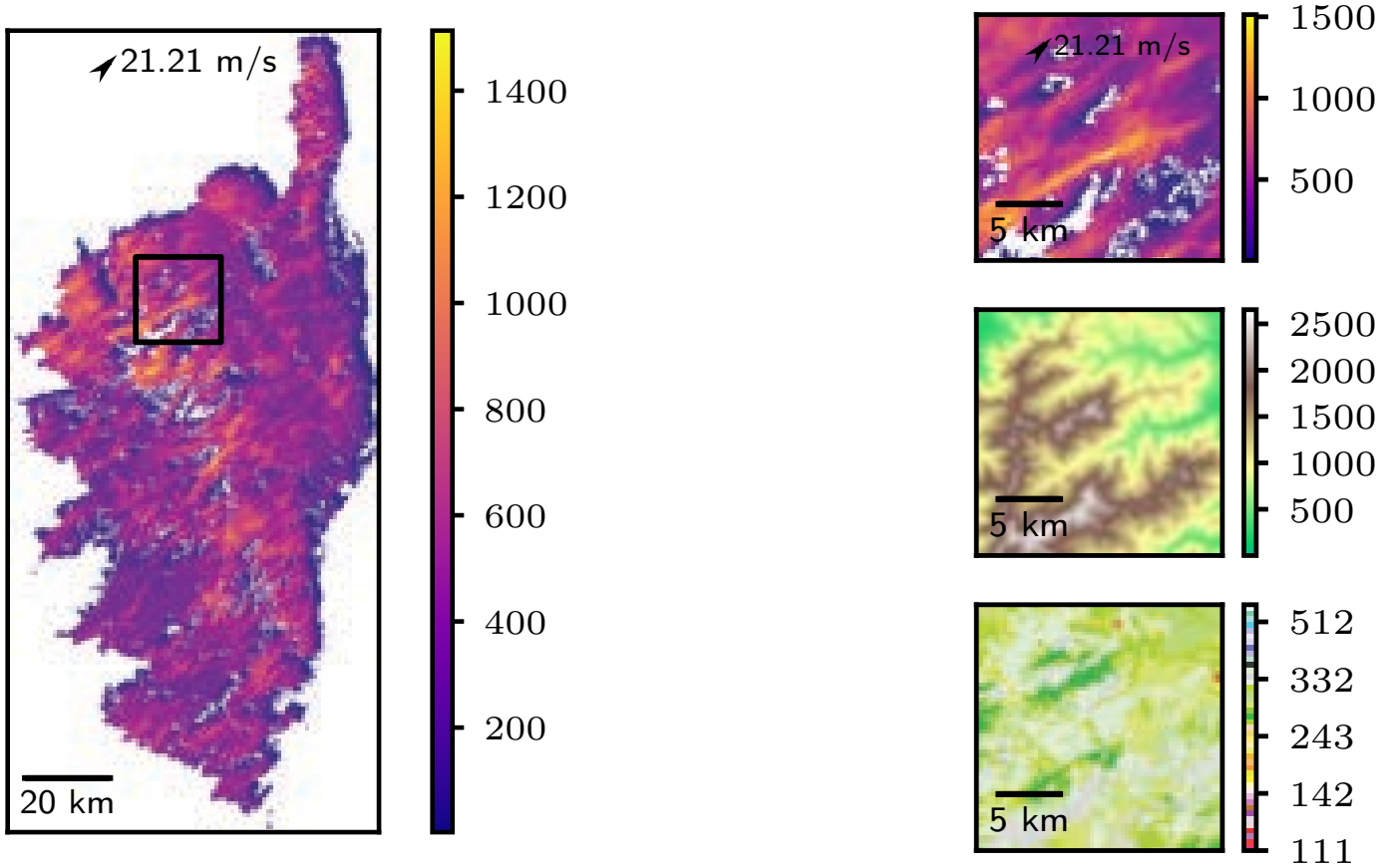
Sensitivity, SHapley Additive exPlanations test





SHAP values associated with the emulator computed for the test dataset, using the training dataset as basis. The SHAP values corresponding to the 1024 inputs resulting from the convolutional part of the DNN are summed up and this sum is identified as “Position” in the figure.

Towards Simulated Maps



Burned surface (in hectares) based on 1 million simulations (40 seconds on 32 CPUs). Spatial resolution 80 m ; From top to bottom : burned area (ha), altitude (m), land cover. § **Computation speedup vs simulation** $\geq 10^4$

PhD in 2021

Fréderic Allaire- Inria - ANR FireCaster - Co-advised with Vivien Mallet

Ack

- Funding : This work was supported by the Agence Nationale de la Recherche, France [grant number ANR-16-CE04-0006 FIRECASTER].
- This work was carried out using HPC resources from GENCI-IDRIS (Grant 2020-AD011011630).
- Numerical Wildfire Workshop - March 2021 - Cargese, Corsica

References

- Frédéric Allaire, Jean-Baptiste Filippi, and Vivien Mallet. **"Generation and evaluation of an ensemble of wildland fire simulations"**. In : International Journal of Wildland Fire 29.2 (2020), p. 160. doi : 10.1071/wf19073.
- Frédéric Allaire, Vivien Mallet, and Jean-Baptiste Filippi. **"Novel method for a posteriori uncertainty quantification in wildland fire spread simulation"**. In : Applied Mathematical Modelling 90 (Feb. 2021), pp. 527–546. doi : 10.1016/j.apm.2020.08.040.
- Paper on DeepFire Submitted. On danger maps in prep.