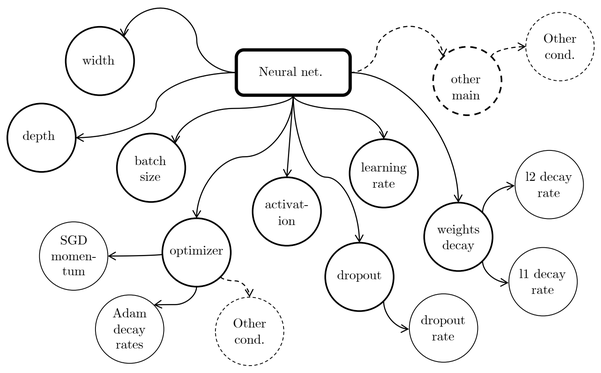
Uncertainty analysis as an ally for deep-learning-based hybridization of simulation codes

Goal oriented sensitivity analysis for cost effective neural networks

$\tau = \{\tau_i\}_{0<i\leq n_h}$ where each τ_i is one of the n_h hyperparameters.



Challenges:

- High number of possible configurations
- Different measurable spaces (categorical, integer, continuous, Boolean)
- Interactions
- Can be conditioned to each other's values

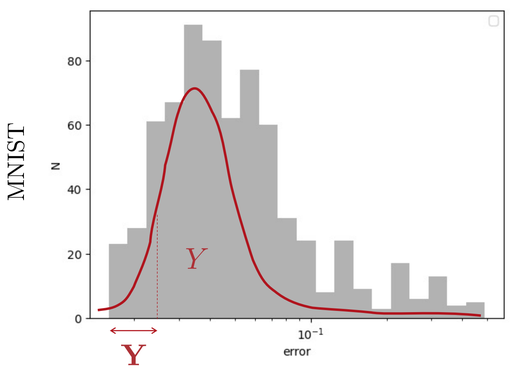
Stakes

- Impact on **error** and on **computational cost**

Challenges

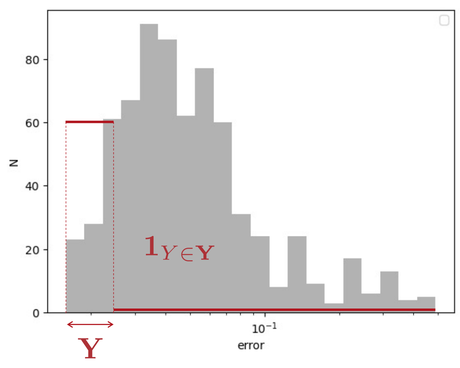
- Interpretability
- Multi-objective
- Space complexity

Traditional SA approach:



- Assess the link between Y and each X_i
- Analysis of the variance using Sobol indices (quantitative)

Goal oriented SA:



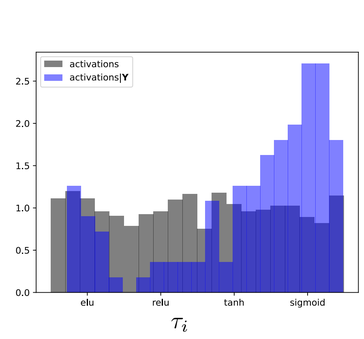
- Assess the link between $1_{Y\in\mathbf{Y}}$ and each X_i
- Several sensitivity indices can be used ...

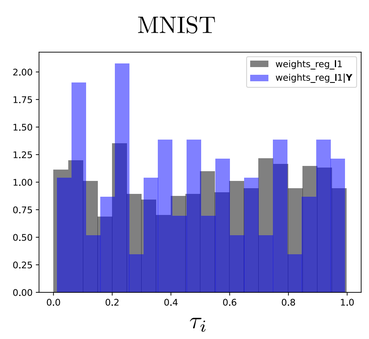
Hilbert Schmidt Independence Criterion (HSIC) based goal-oriented sensitivity analysis

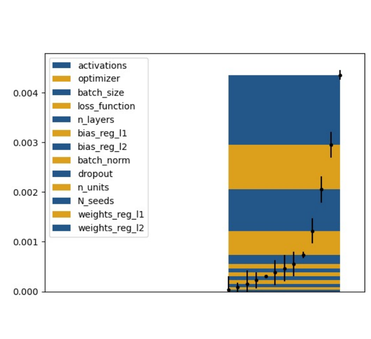
Obstacles to circumvent

- Hyperparameters can be categorical, continuous or Boolean, have different distr.
  Ex: activation function, learning rate, dropout
- They may interact
  Ex: batch size and optimizer
- Some of them are not always involved in the training
  Ex: dropout rate

MNIST





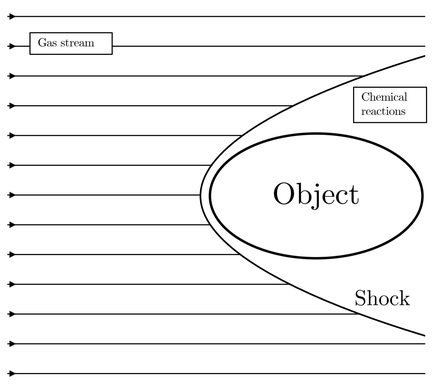


Test case	method	Test metric	# params	factor
MNIST	BO	98.42 ± 0.05	10,271,367	1
	HSIC BO	$\mathbf{98.42 \pm 0.02}$	**151,306**	**68**
Chem. Reac.	BO	2.94 ± 0.42 $(\times 10^{-4})$	1,588,215	1
	HSIC BO	$\mathbf{3.49 \pm 0.31}$ $\mathbf{(\times 10^{-4})}$	**3,291**	**482**

Bayesian Optimization (BO) vs HSIC based Bayesian Optimization (HSIC BO)

Uncertainty quantification to obtain guarantees

Atmospheric reentry coupled with chemical reactions:



U : Fluid variables $\mathbf{x}$: Chemical reactions variables

```
initialize_guess_vector_of_unknowns_on_mesh(U^0,x^0)
while convergence_criterion_not_satisfied do
    U^{n+1/2} =solve_CFD(U^n,x^n)
    for each cell i ∈ mesh do
        (U_i^{n+1}, x_i^{n+1}) =solve_chemical_reactions(U_i^{n+1/2},x_i^n)
    end
end
```

Mutation++

Original code

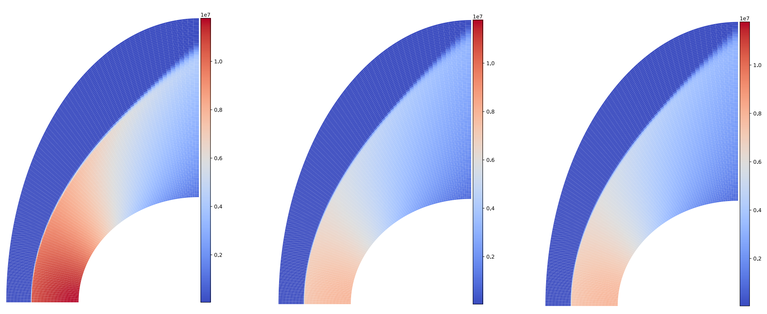
```
initialize_guess_vector_of_unknowns_on_mesh(U^0,x^0)
while convergence_criterion_not_satisfied do
    U^{n+1/2} =solve_CFD(U^n,x^n)
    (U_i^{n+1}, x_i^{n+1}) =solve_chemical_reactions_NN(U_i^{n+1/2},x_i^n)
end
```

Neural Network

Hybrid code

With BO + HSIC !

time : 81 s	time : 220 s	time : 4090 s
error : 7.74×10^{-2}	error : 6.06×10^{-7}	error : ref
		$\times 18.7$



PG (Perfect Gas) | NN (Neural Network) | MPP (Mutation ++)

Pressure field

Is this error **acceptable ?** Can we obtain **guarantees ?**

- Parameter uncertainty error: δv
- Discretization error: δ_Δ
- Model error: δ_M

How do they compare to δ_{NN} , the error coming from the approximation of Mutation++ by a neural network?

Error study:

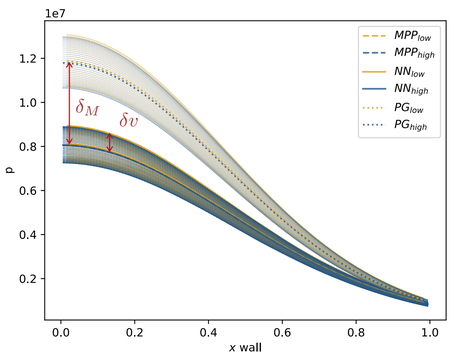
Mesh size: 90×100 (PG/NN/MPP$_{high}$) vs 30×100 (PG/NN/MPP$_{low}$)

Model: PG vs NN/MPP

Neural network approximation: MPP vs NN

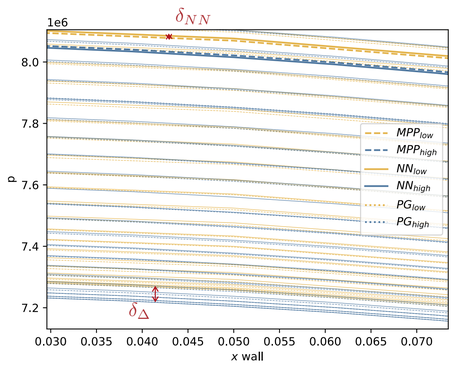
...under the uncertainty of

Upstream speed: $v_0 + \delta_v$ with $\delta_v \sim \mathcal{U}(-0.05v_0, +0.05v_0)$



Pressure at the wall of the object

- Model: δ_M
- Uncertainty: δv
- Discretization: δ_Δ
- Neural network: δ_{NN}



Pressure at the wall of the object (zoomed)

$$\delta_{NN} < \delta_\Delta < \delta v < \delta_M$$