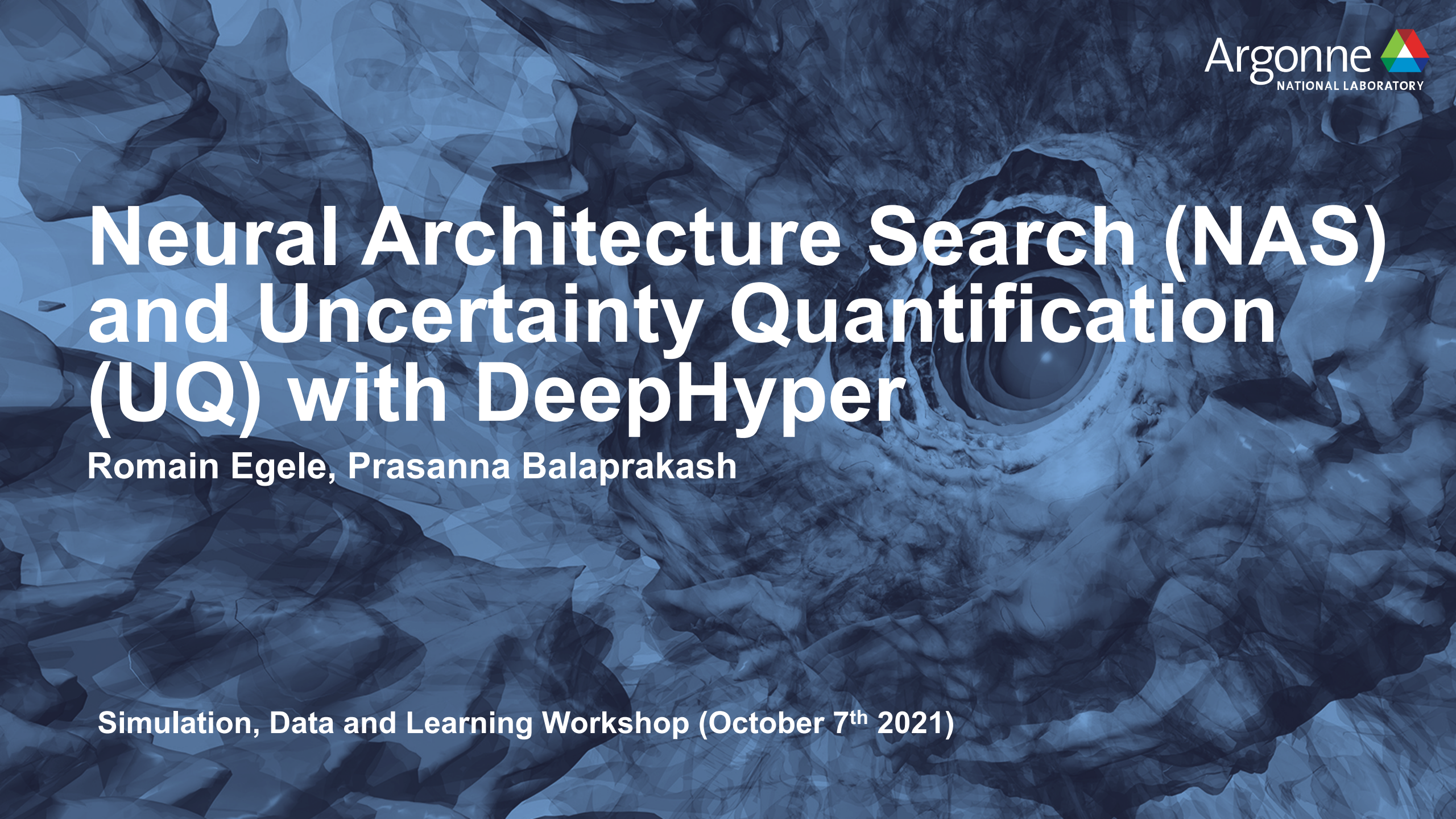


# Neural Architecture Search (NAS) and Uncertainty Quantification (UQ) with DeepHyper

Romain Egele, Prasanna Balaprakash

Simulation, Data and Learning Workshop (October 7th 2021)

# The DeepHyper Project

"Automated development of machine learning algorithms to support scientific applications"

**Prasanna Balaprakash**

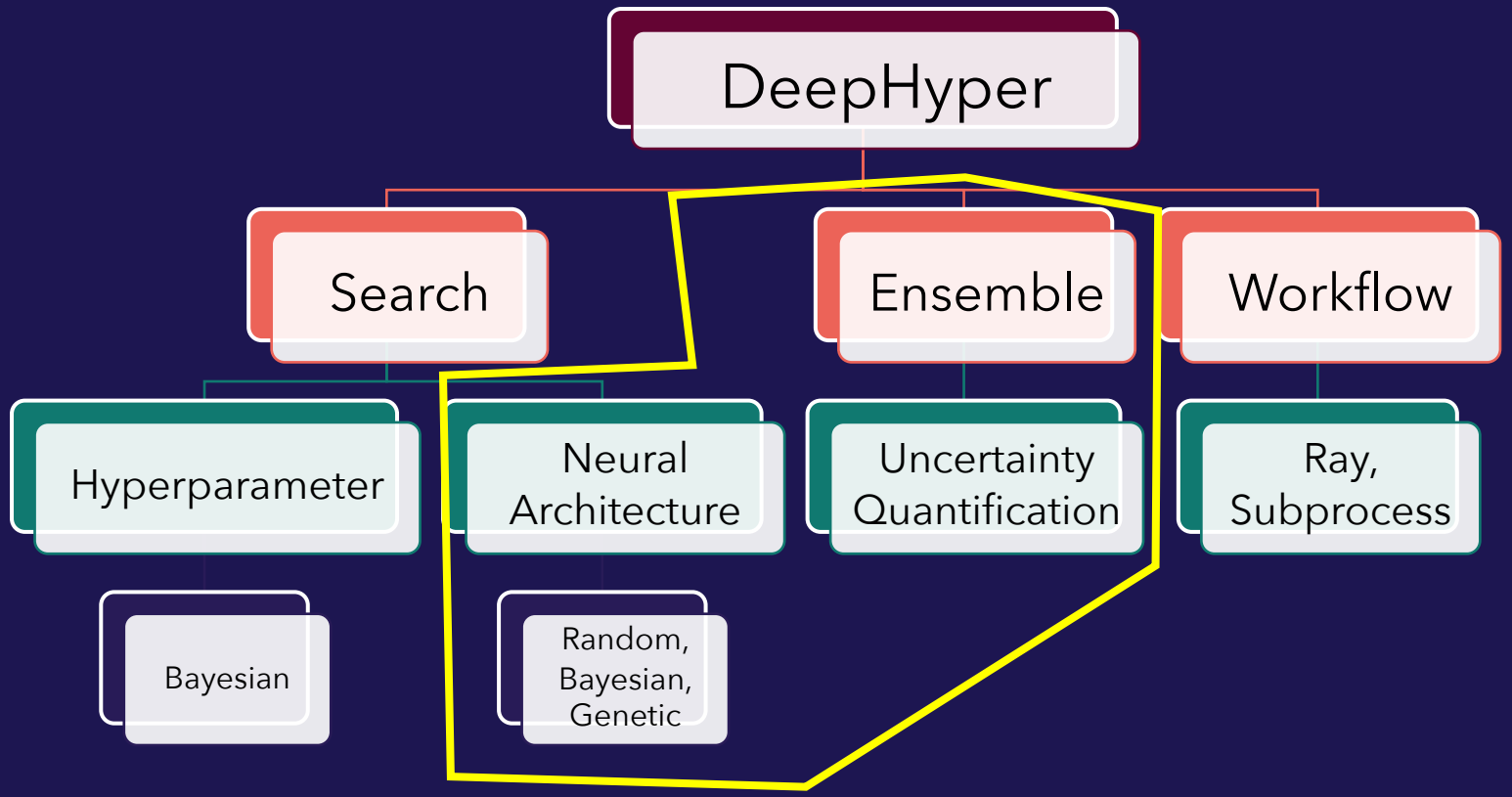
**Romain Egele**

DeepHyper

Open-Source

https://deephyper.readthedocs.io/

# DeepHyper Overview




DeepHyper documentation: http://deephyper.readthedocs.io

# Installed on ALCF systems

- **Theta**

```
$ module load conda/2021-09-22
```

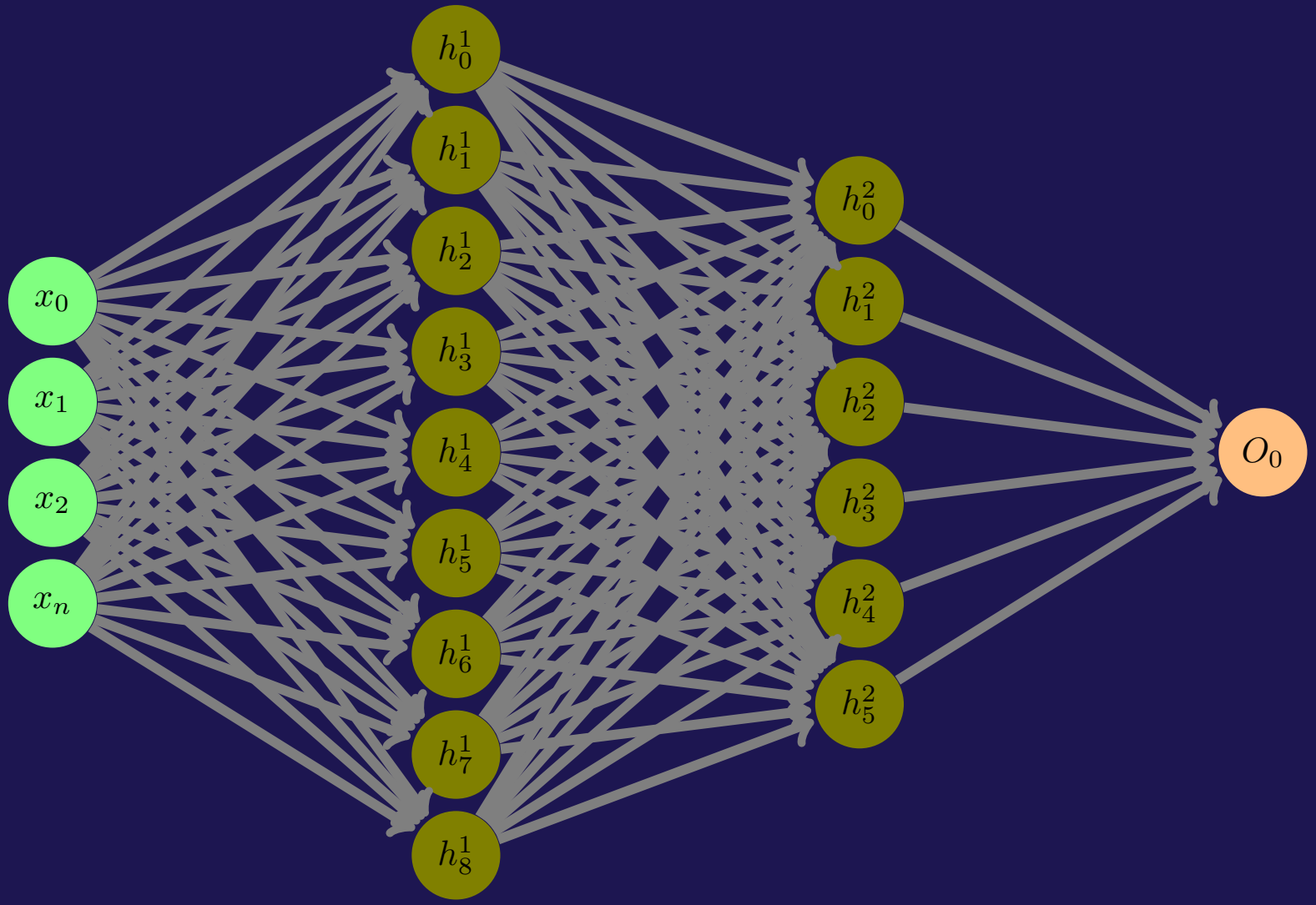
- **ThetaGPU**

```
$ module load conda/2021-09-22
```

**Warning**: After loading the module, don't forget to run `$ conda activate base`

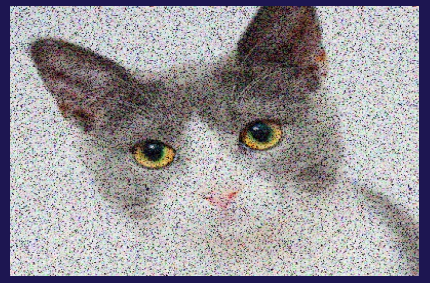
# Artificial neural networks

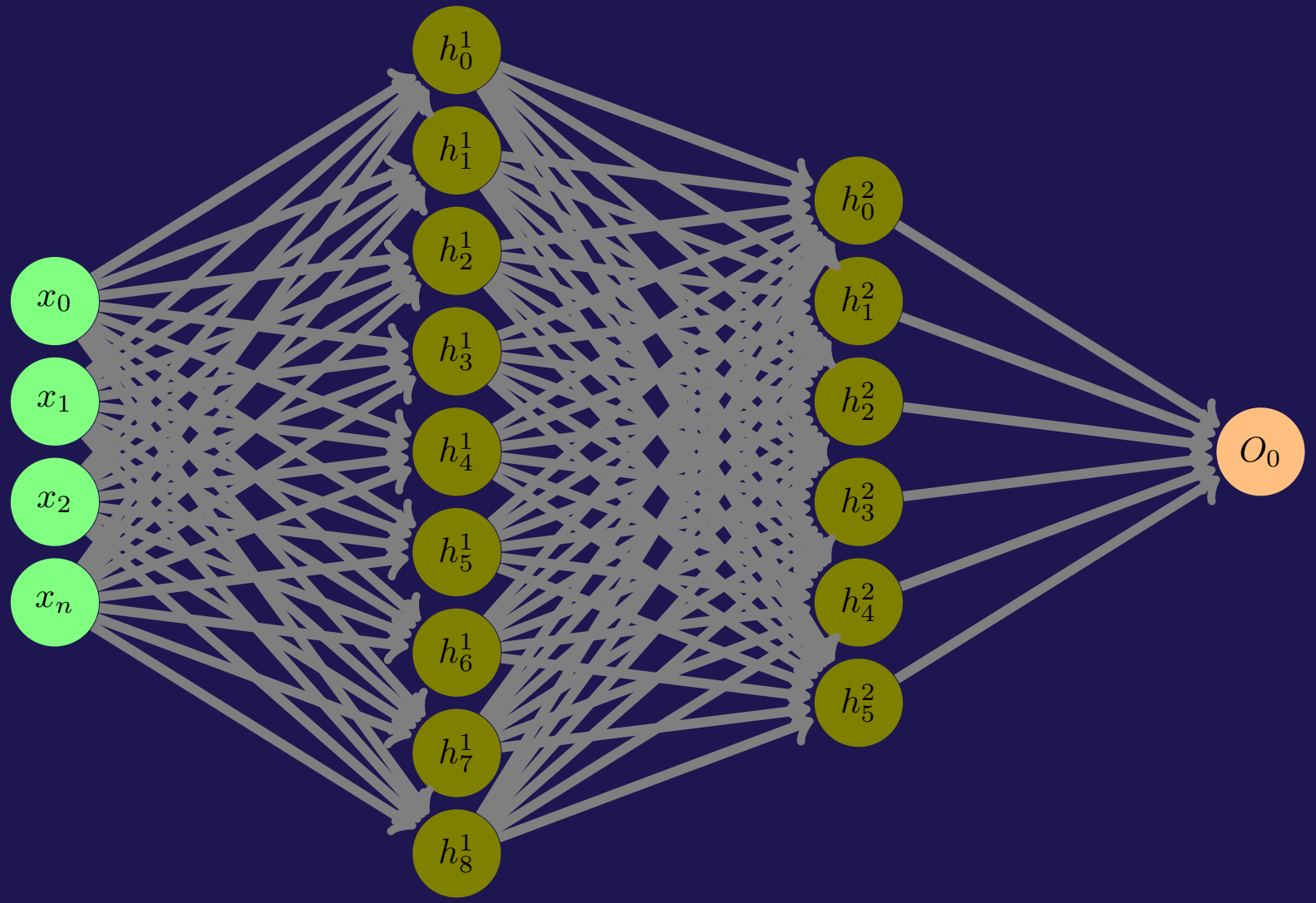
Training



# Artificial neural networks

**Inference**





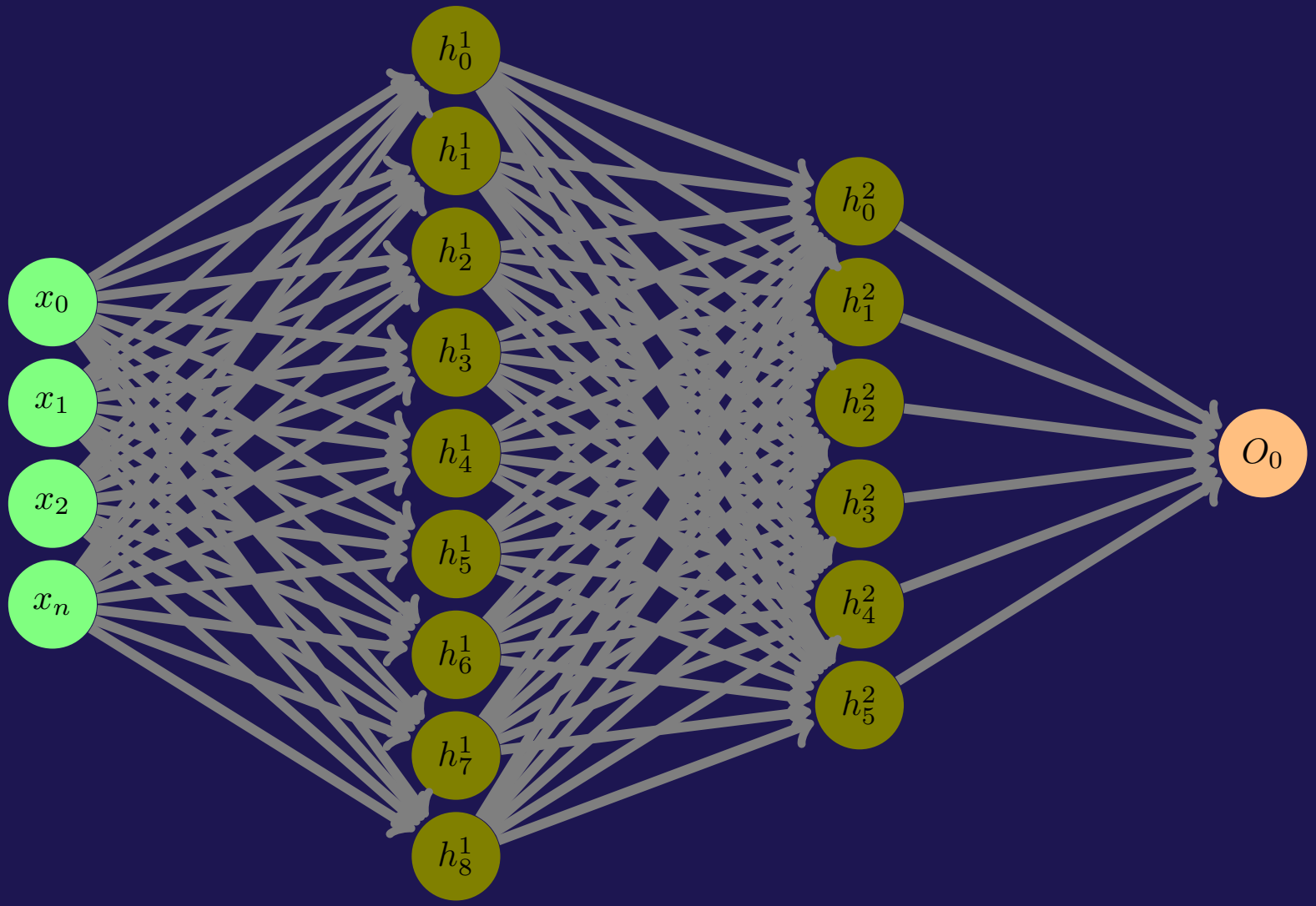

Cat

Dog

# Artificial neural networks

## Inference

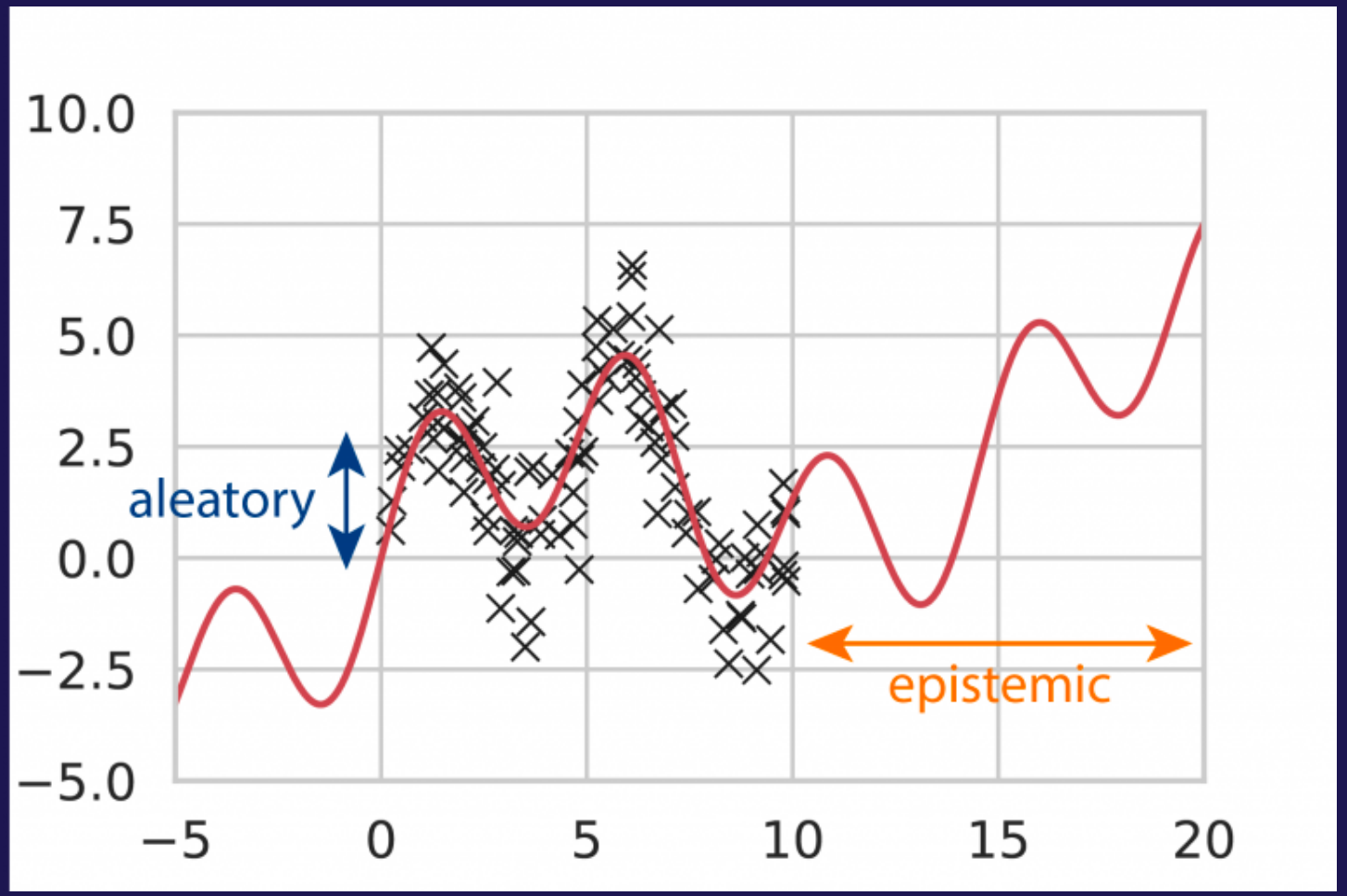
Out of distribution




Cat

Dog

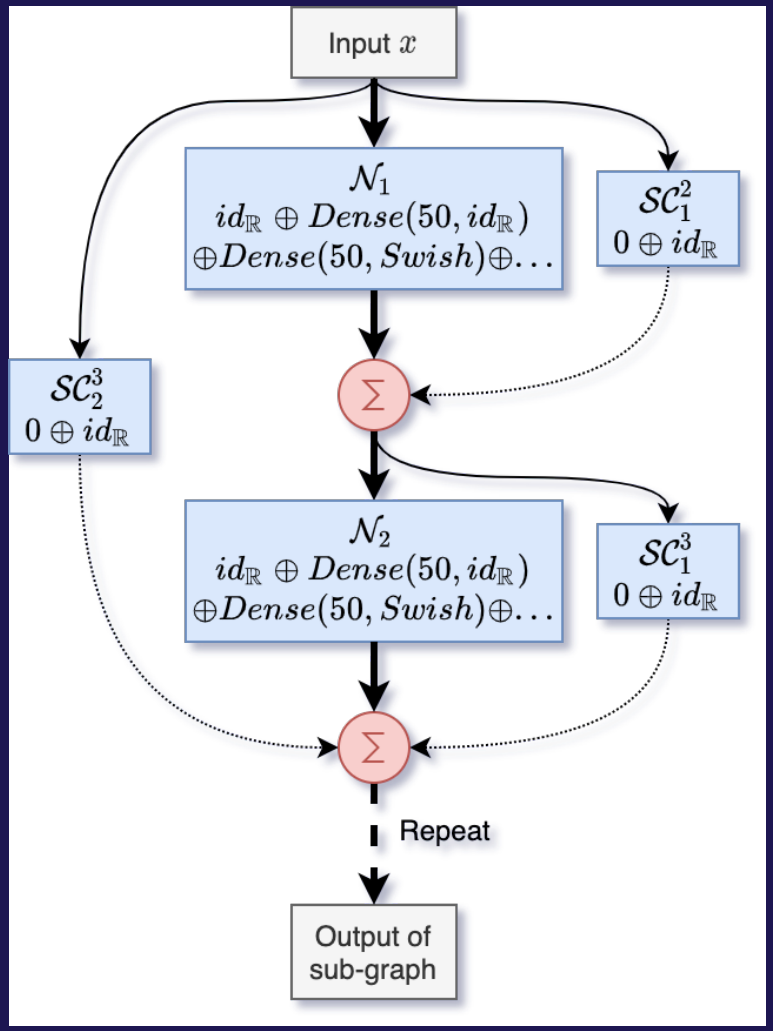
# Two major forms of uncertainty




https://www.inovex.de/de/blog/uncertainty-quantification-deep-learning/

# Neural Architecture Search

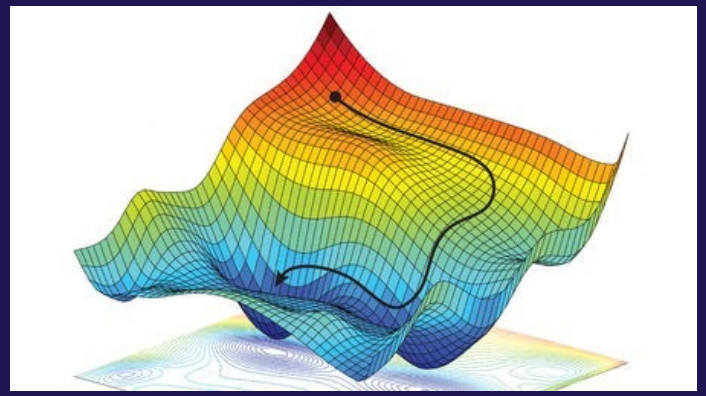
# Neural Architecture Search Space




**WARNING**: Hyperparameters are kept constant in general

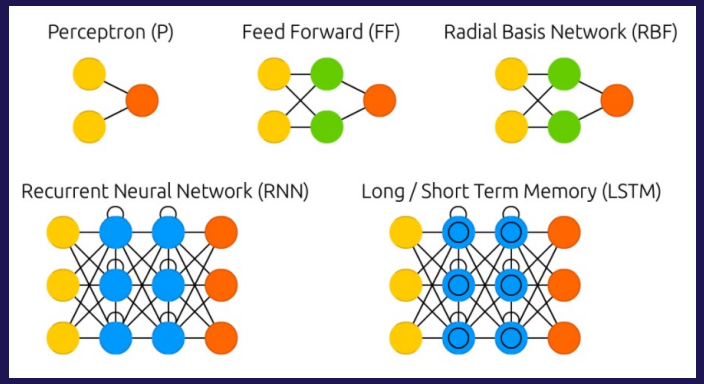
# Hyperparameters of Neural Networks

## Algorithm Hyperparameters



Optimizer: SGD, RMSprop, Adam…
Learning rate
Mini-batch size
Learning rate scheduler
Adaptative batch size
…

## Model Hyperparameters




Number of layers
Type of the layer: Fully Connected, Convolution, Recursive…
Activation function
Dropout rate
Skip connection
…

# Joint Hyperparameter and Architecture Search Problem

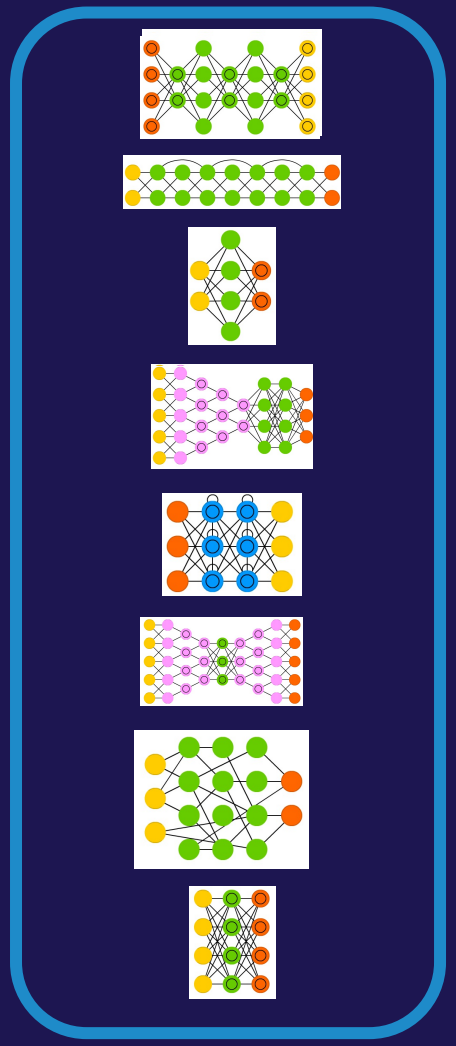
Lower-level problem: *Training data "$T$"*

$$\min_{w} \ \text{err}_T(a, h; T; w)$$

Upper-level problem: *Validation data "$V$"*

$$\min_{a,h} \text{err}_V(a, h; V; w^*)$$

# Selection of K models for Deep Ensembles with UQ



Catalogue

Ensemble

**Each neural network predicts a Probability distribution**

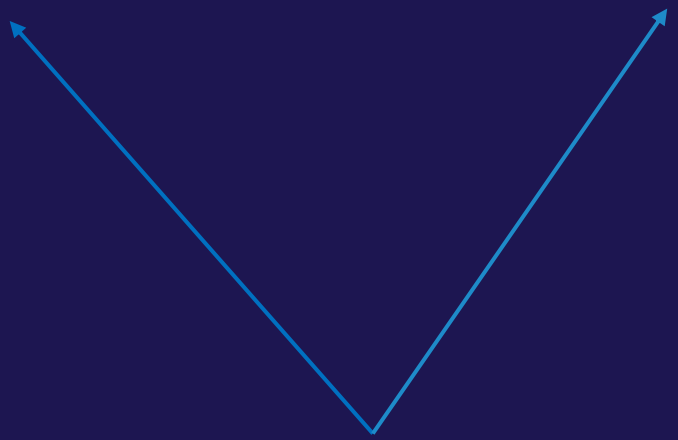
**The ensemble reflects a new mixture of distribution.**

# Decomposition of Ensemble Uncertainty

aleatoric

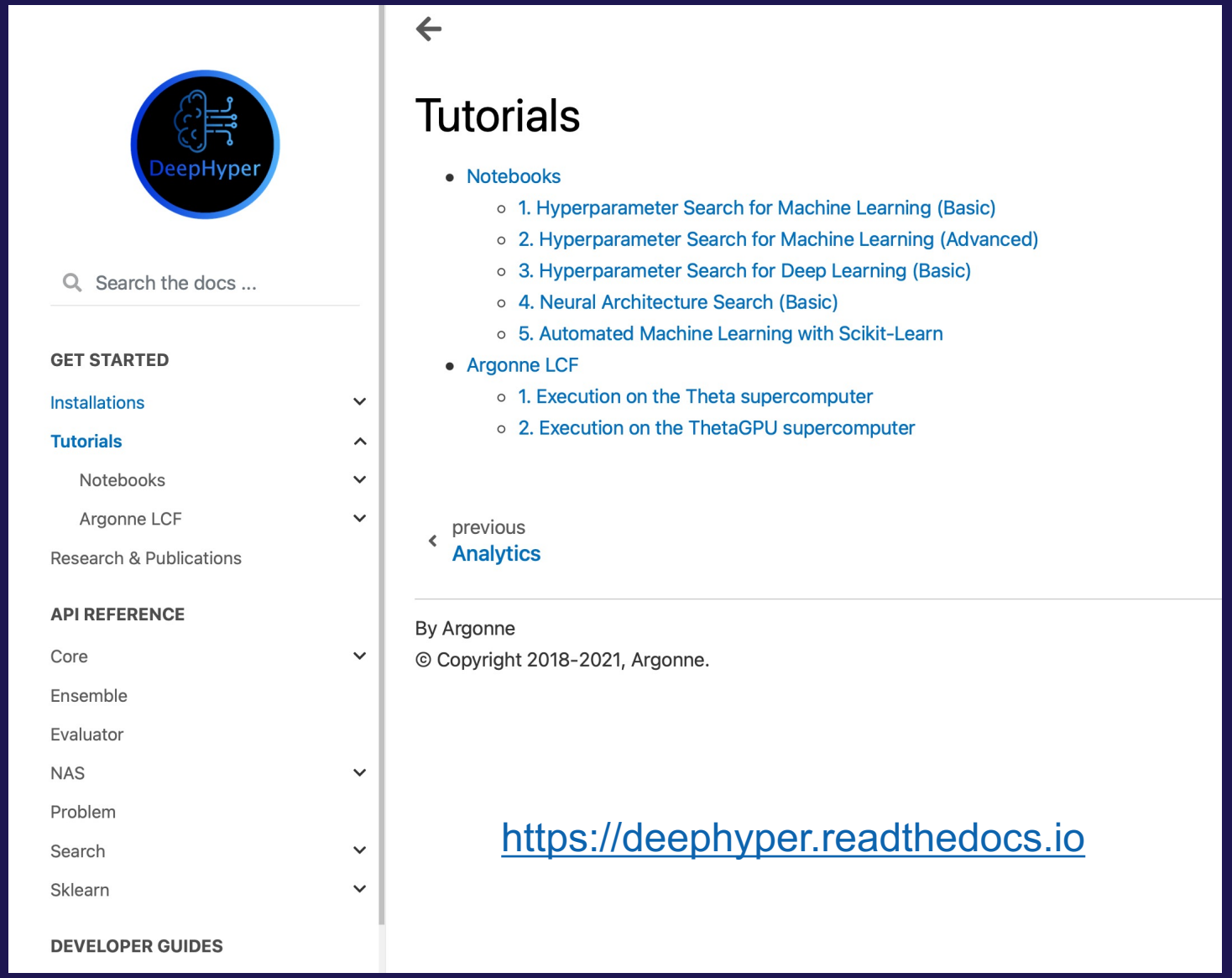
epistemic

$$\sigma_{\mathcal{E}}^2(x) = E_{p(\theta)}\left[\sigma_{\theta}^2(x)\right] + V_{p(\theta)}\left[\mu_{\theta}(x)\right]$$



Learned and predicted by the neural networks

# Learn more about DeepHyper



# A Tutorial for NAS, Joint HPS + NAS, Deep Ensemble


## From Neural Architecture Search to Automated Deep Ensemble with Uncertainty Quantification

**GITHUB: sdl_ai_workshop/03_distributedHyperOpt/02_NAS_and_more/**

### Table of Contents