# Implementation and testing of a deep neural network emulator for aerosol activation in E3SM

Balwinder Singh[1], Sam Silva[1], Mike Pritchard[2] and Po-Lun Ma[1]

[1]Pacific Northwest National Laboratory
[2]University of California, Irvine

Motivation

- Aerosol Activation is one of the most computationally expensive aerosol process

- Establishing a method to enable the use of deep neural networks (DNN) to replace (or enhance) physical parametrizations (such as aerosol activation) in E3SM

drydep 13%
hetfrz_classnuc_cam_calc 17%
wetdep 9%
nucleate_ice_cam_calc 4%
ma_convproc 3%
convect_deep 1%
wateruptake 6%
calcsize 1%
modal_nucl 1%
modal_coag 1%
chemdr_diags 1%
modal_gas_aer_exchange 2%
Aerosol activation and other Aerosol-Cloud Processes [41 %]
dropmixnuc 41%

Approximate computational cost of the aerosol processes
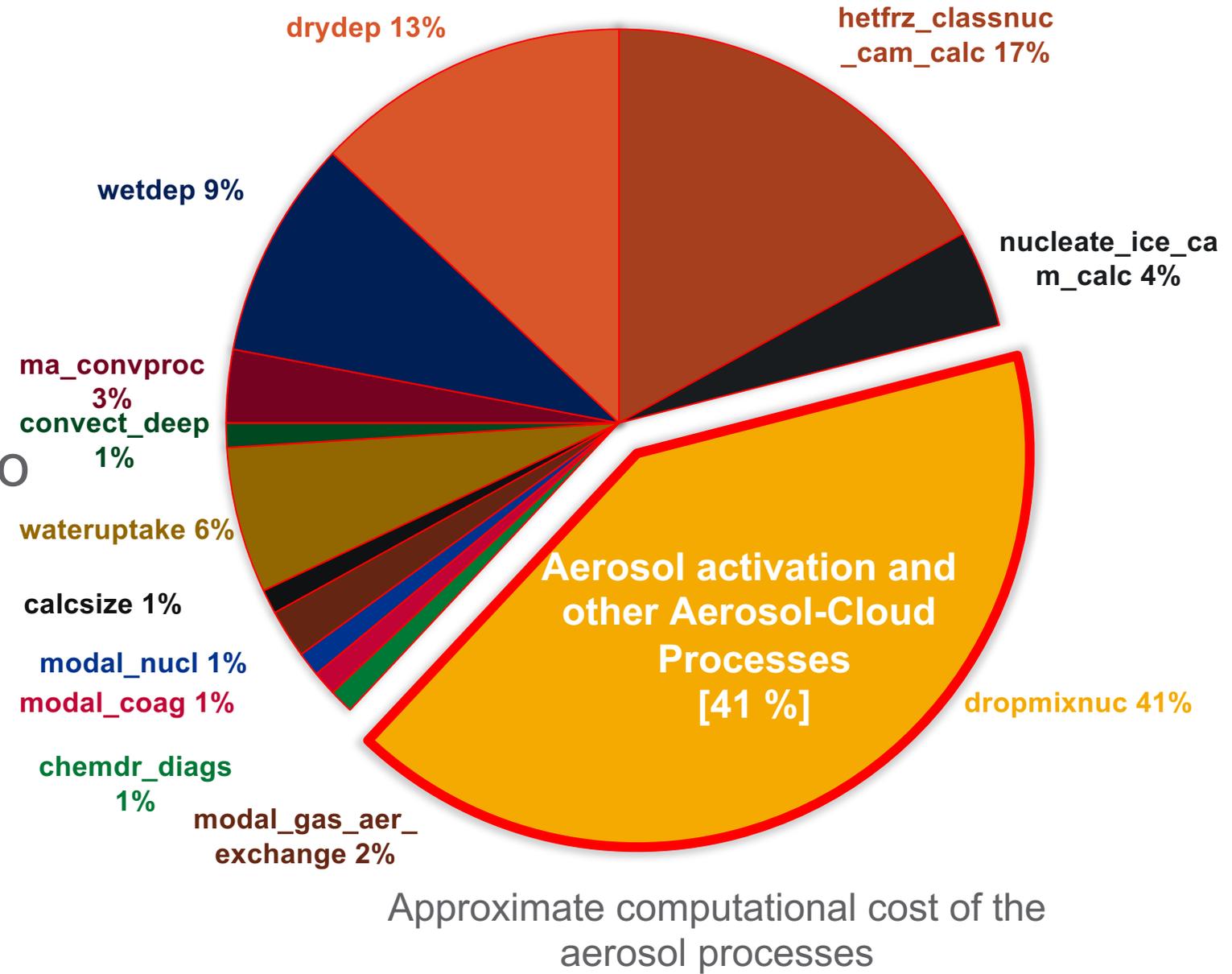
Implementation Description

➢ Fortran-Keras-Bridge (FKB) library provides a seamless interface to invoke trained DNN within Fortran codes (Ott et al. 2020)

➢ Prebuilt FKB library can be linked to any large-scale Fortran models such as E3SM

➢ Pretrained DNN networks can be made available to E3SM via a configuration text file

➢ E3SM can invoke DNN at each time step with a set of inputs to obtain output(s) from DNN

Image from: https://github.com/scientific-computing/FKB
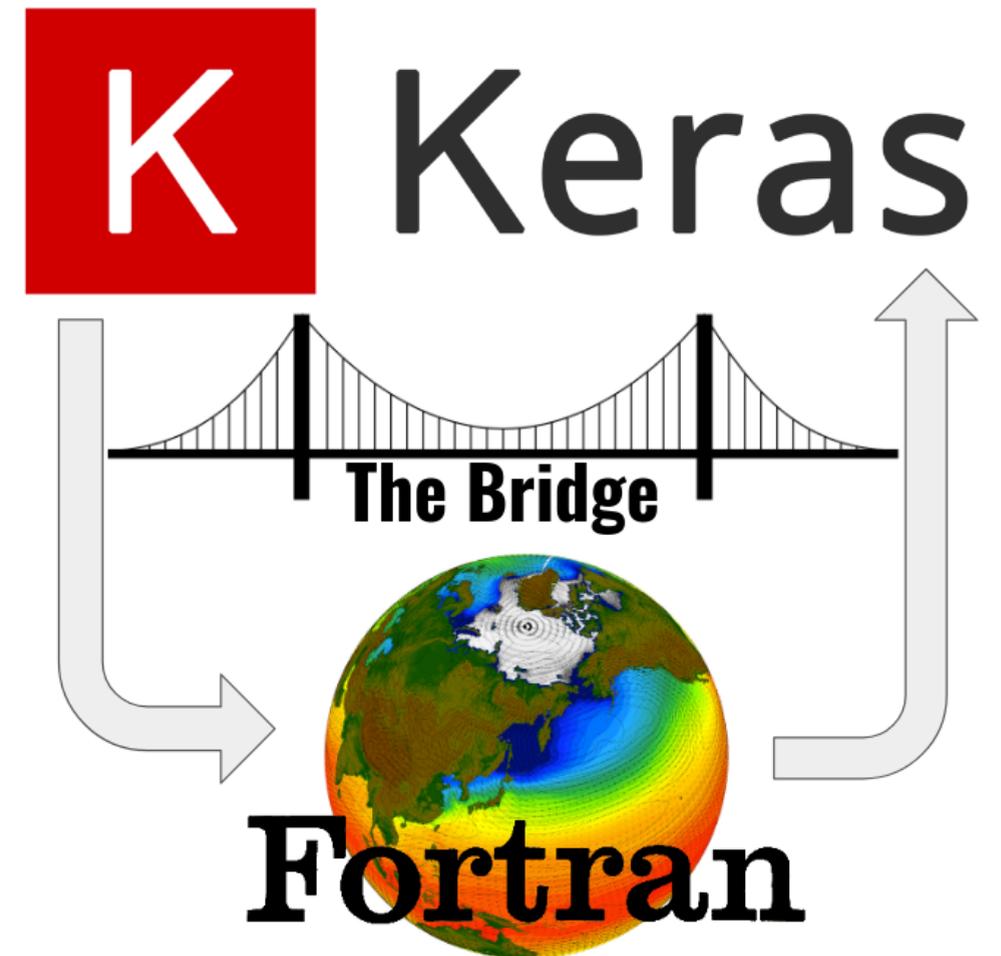
Steps for using FKB in E3SM

➢ Build FKB library on a local machine

➢ Link FKB library in E3SM machine files

➢ Add an "use" statement to access FKB provided data types and methods

➢ Load a pre-trained DNN configuration file during model initialization

➢ At each time step:
- Prepare inputs for the calling DNN
  - ✓ May involve normalizing inputs based on means and standard deviations
- Call DNN using FKB provided function
- Process DNN output to use within E3SM
 - ✓ May involve de-normalization of the output
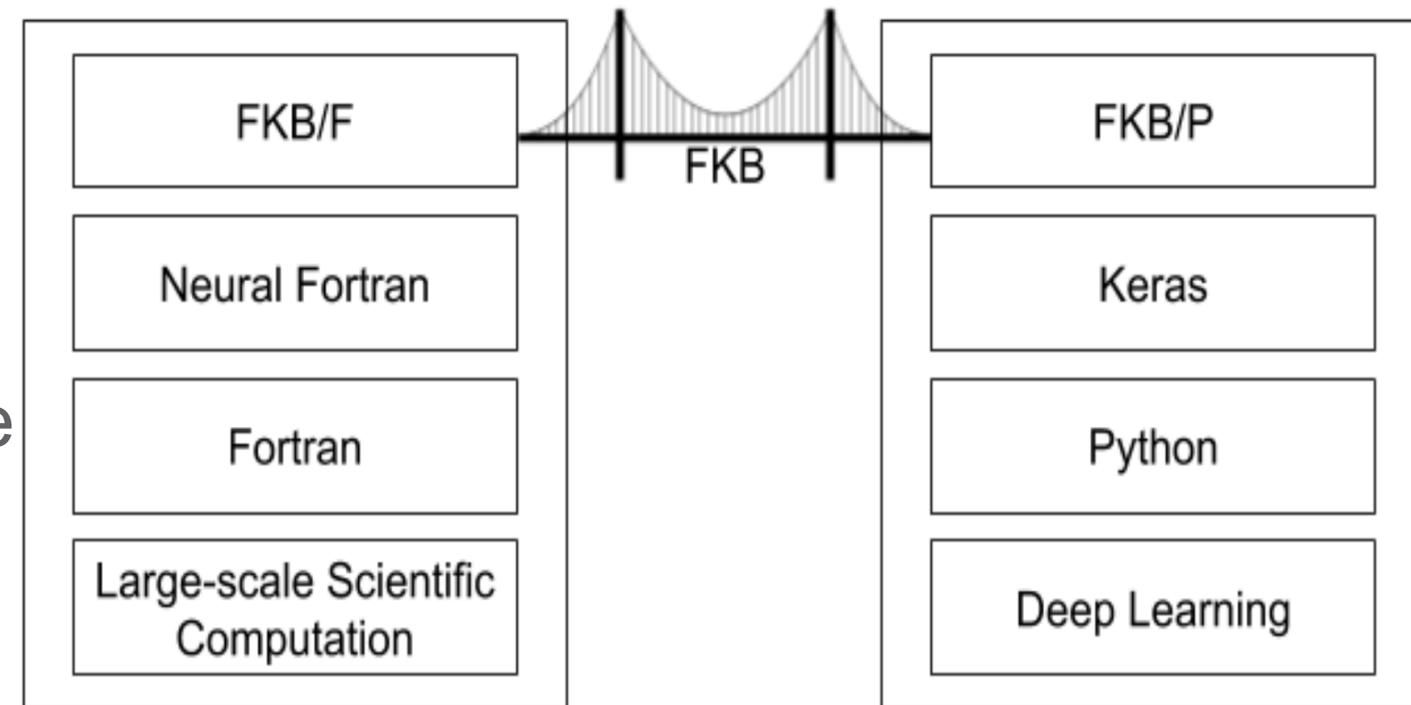
Positioning of FKB within Fortran and Python ecosystems
(Image courtesy Ott et al. 2020)

Emulating aerosol activation using DNN

- Replaces parameterized aerosol activation by pre-trained DNN

- Inputs supplied (normalized):
  - Temperature, pressure, hygroscopicity, vertical velocity, mean size of mode, accommodation coefficient, aerosol number concentration

- Output (de-normalized):
  - Activated fraction of aerosol number

- Additional quantities derived from activated fraction:
  - Activated mass fraction
  - Fluxes of aerosol number and mass
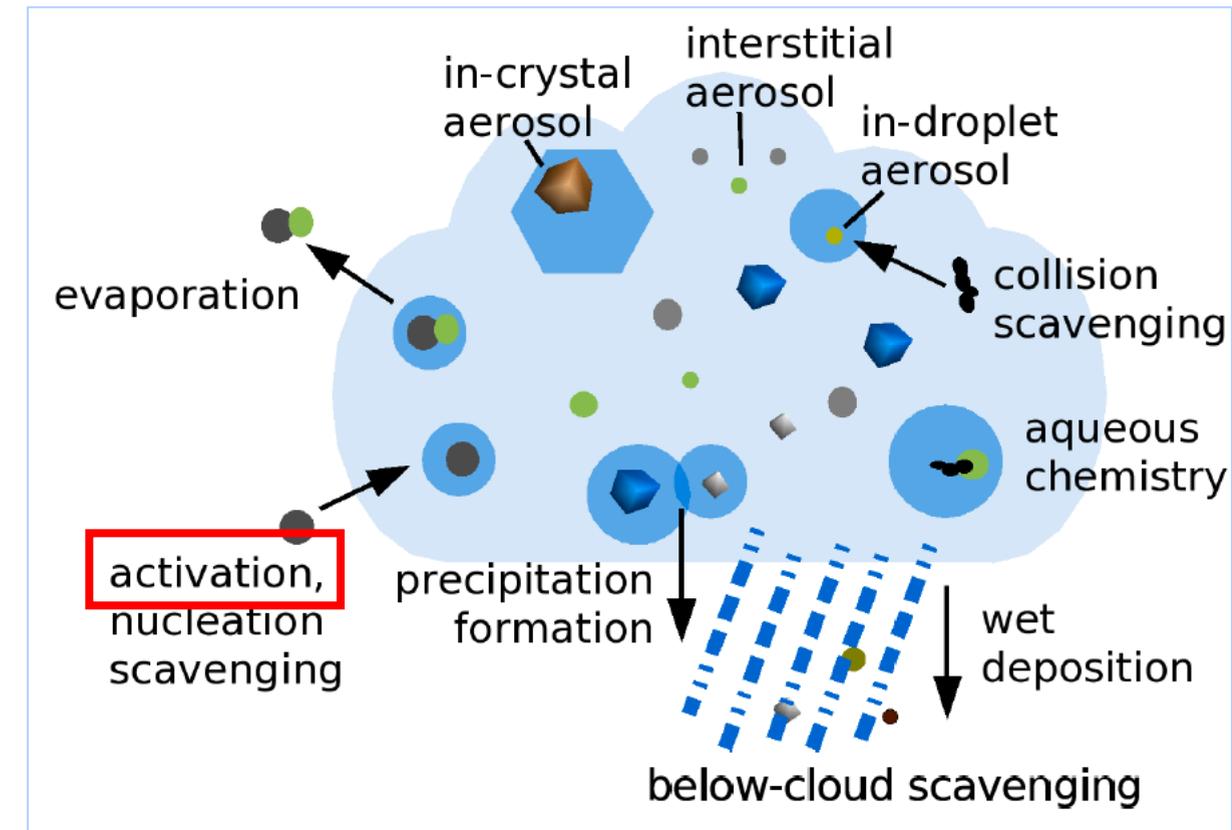
(Image courtesy: Hoose et al. 2008)

Comparing DNN and parameterized aerosol activation

Simulation Setup:

- Two E3SM atmosphere only simulation:
    - Using parameterized aerosol activation
    - Using DNN aerosol activation
- Ne30 (1-degree) resolution for 2 years

Software considerations when implementing emulators in E3SM

- Numerical considerations
  - Clipping (input)
    - Assumptions made in other parts of the model: updraft velocity
    - Inconsistent bounds: minimum hygroscopicity in the emulator (10^{-4}) vs in E3SM (10^{-10})
  - Clipping (output)
    - Fraction between 0 and 1
  - Sampling of training data
    - Comprehensive (Latin Hypercube, MCMC) vs. realistic multivariate PDF

- Computational cost
  - DNN on CPU machines is slightly cheaper than the default E3SM with ARG

New DNN-based activation in E3SM runs smoothly
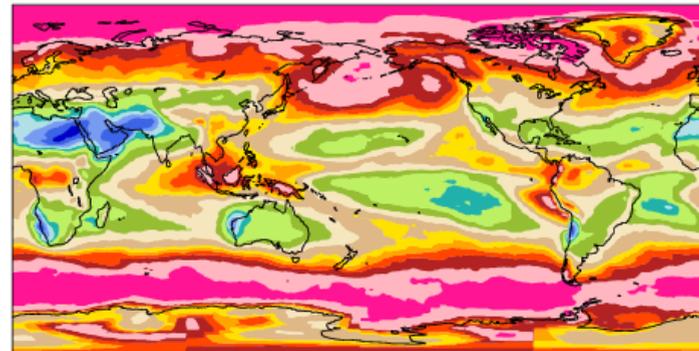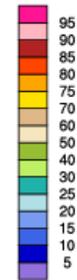
Cloud fraction

Precipitation rate

DNN

ARG
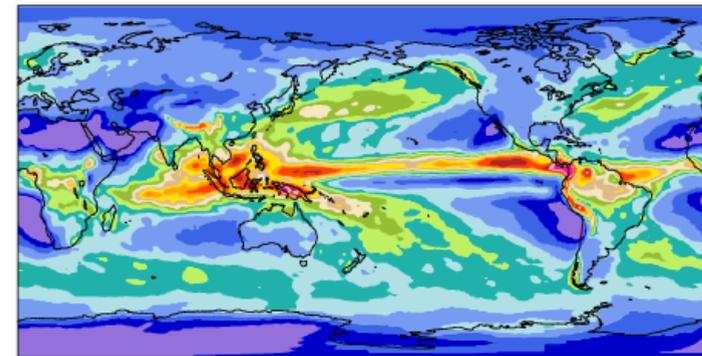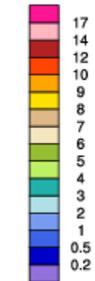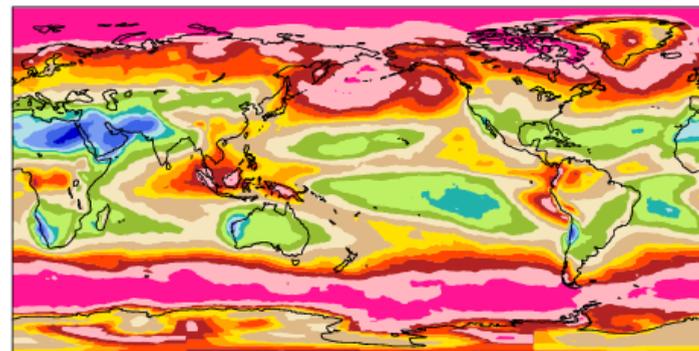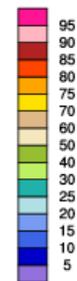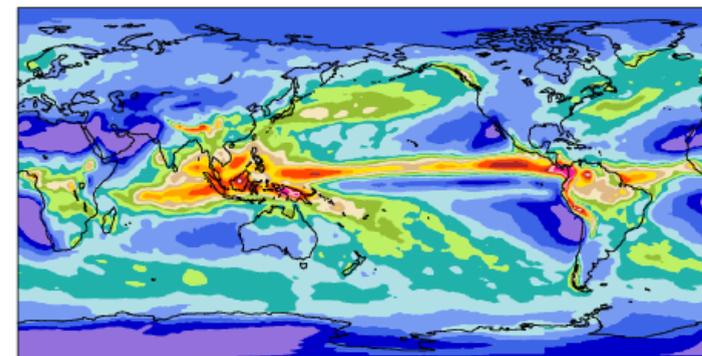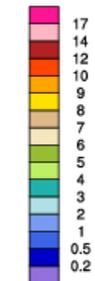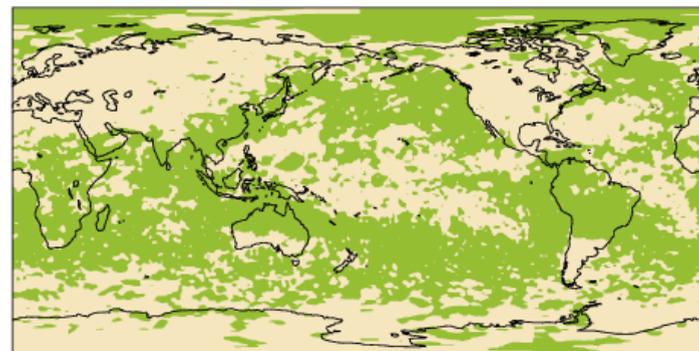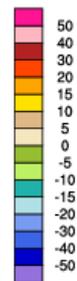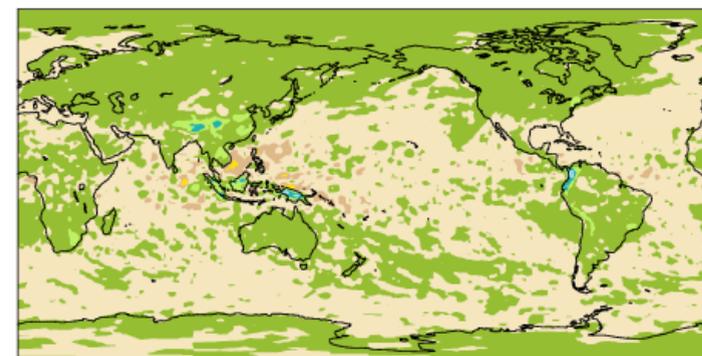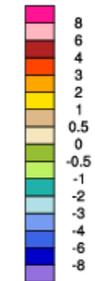
Diff

Overprediction of activation in the emulator caused by "comprehensive sampling" and single mode treatment

Column integrated droplet number concentration (x10⁹)
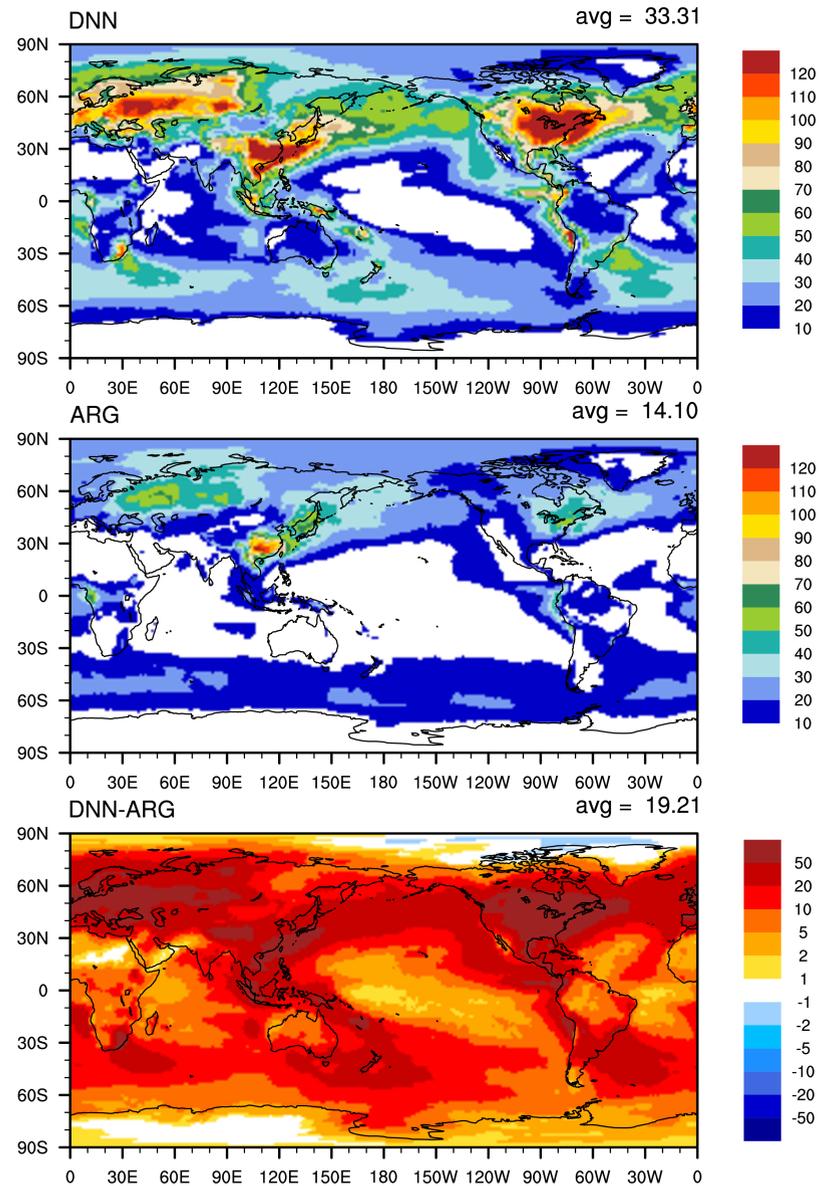
ERFaci
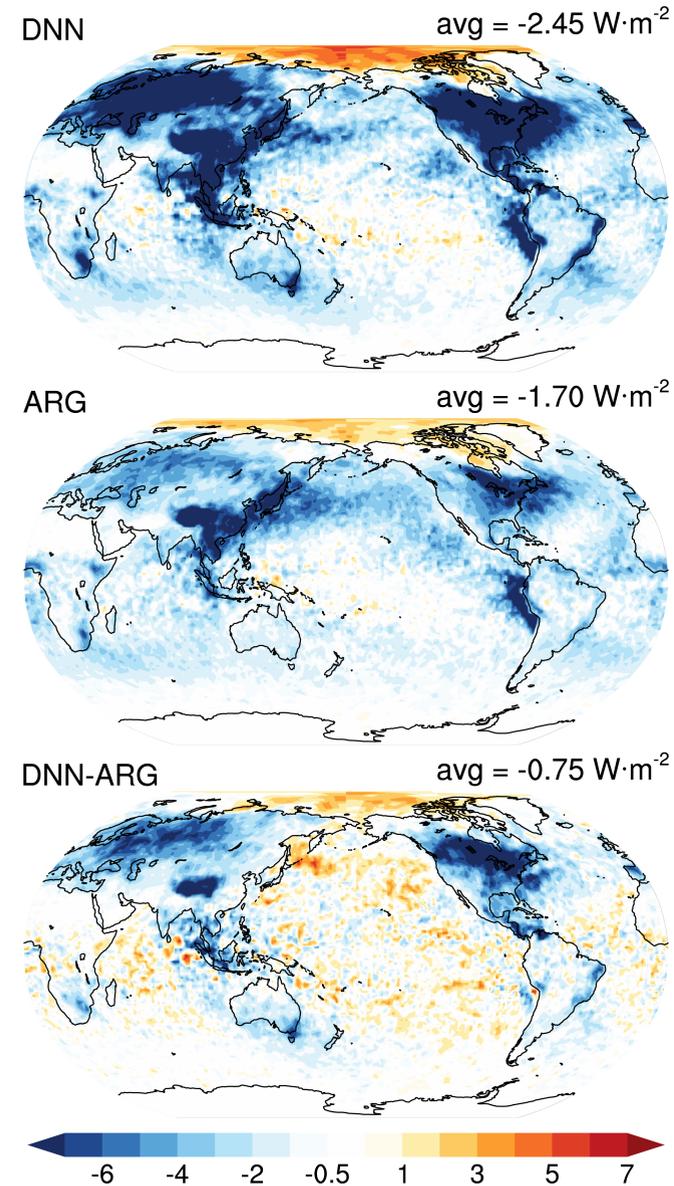
Summary and Future Work

- DNN has been successfully used to drive aerosol activation process within E3SM using FKB

- Model runs stably for long simulations using some physically based limiters

- DNN based aerosol activation reasonably simulates the activation process

- Over prediction of activation is caused by comprehensive sampling and single mode treatment. A new emulator development is on the way to fix this.

- Next Steps:
    - Further refine and generalize this approach to streamline the use of DNN in current and future versions of E3SM