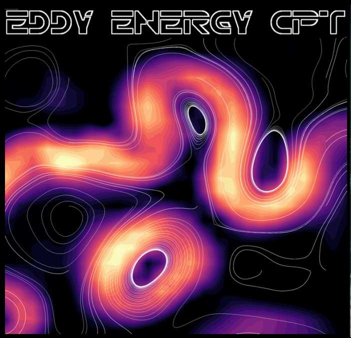
Laure's role in M²LInES

Research interests

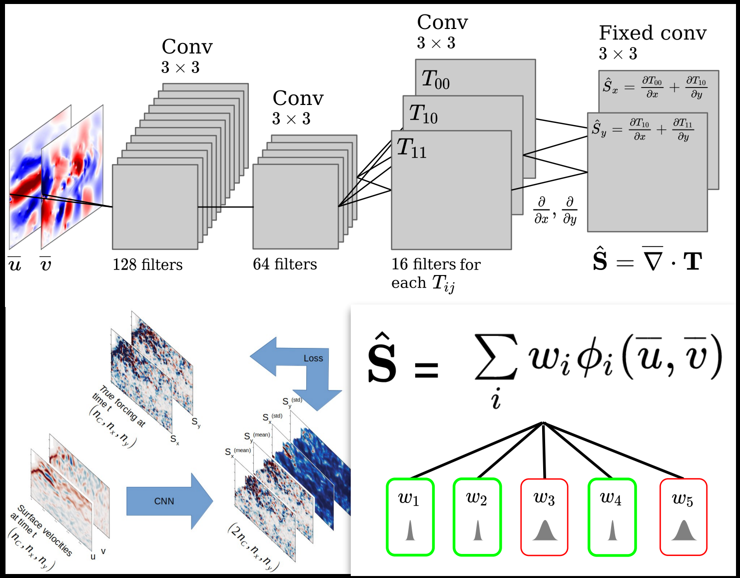
OHC, Zanna et al, 2019



Main scientific contributions in the project:

New parameterization of ocean mesoscale momentum & energy + air-sea coupling

Scale-aware & flow-aware + interpretable ML parametrization models



Alistair's role in M²LInES

What Alistair does in real life:

- Develops numerical simulation codes for ocean, sea ice and icebergs
  - MITgcm, GOLD, MOM6, SIS2, KID, …
- Co-led development of recent models (distinguishing configuration from code) at GFDL

M²LInES at GFDL:

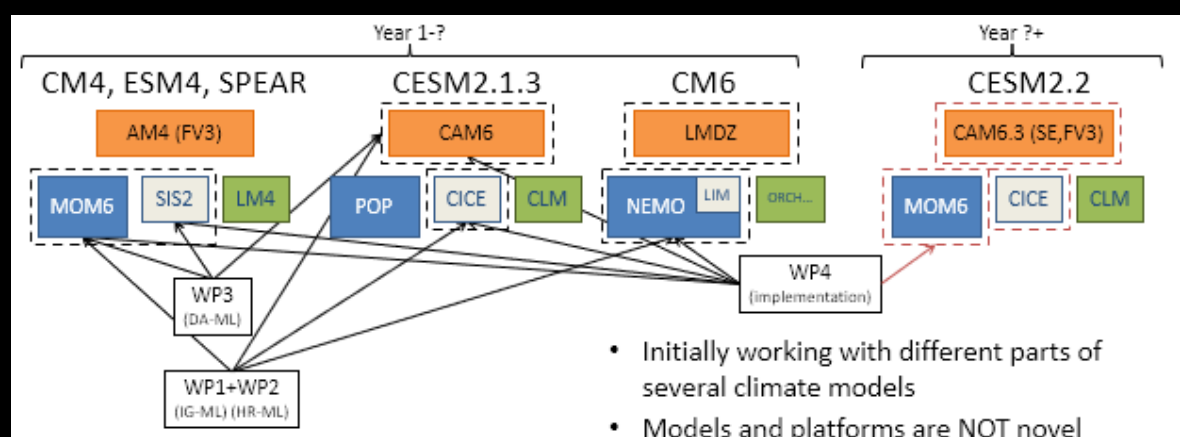
- Implement existing NN's in MOM6 (BZ19 & ZB20), and then new NN's from other projects in MOM6+SIS2 (Alistair)
- Sea-ice DA (Mitch)
- Ocean DA (Feiyu)
- BL, waves and mixing LES (Brandon)

Implementation:

- Getting NNs into GCMs
  - Software question (technical)
  - Fidelity questions (science)
    - Accuracy (how well)
    - Stability (does it always work)
    - Feedbacks (is it right)
- Transferability of learned parameterizations
  - as above

Coordinating between models:

- NEMO, MOM6 (Julien, Julie, Alistair, Feiyu, Brandon)
- CICE, SIS2, LIM (Marika, Mitch, Jul.'s)
- CAM4.5, CAM5, LMDZ? (Judith, Julien, Paul, …)



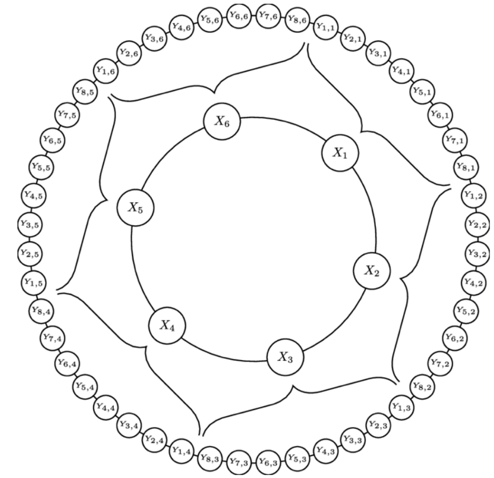
L96 Model to illustrate our goals

The truth

Y's= fast variables; subgrid/unresolved on a coarse-grid

$$\frac{d}{dt}X_k = -X_{k-1}\left(X_{k-2} - X_{k+1}\right) - X_k + F - \left(\frac{hc}{b}\right)\sum_{j=0}^{J-1} Y_{j,k}$$

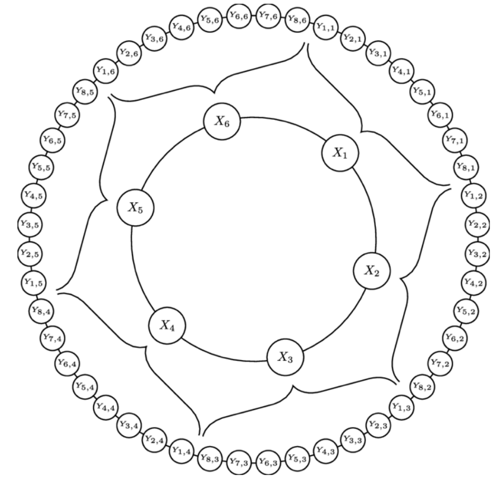
$$\frac{d}{dt}Y_{j,k} = -cbY_{j+1,k}\left(Y_{j+2,k} - X_{j-1,k}\right) - cY_{j,k} + \frac{hc}{b}X_k$$



L96 Model to illustrate our goals

The truth

Y’s= fast variables; subgrid/unresolved on a coarse-grid



$$\frac{d}{dt}X_k = -X_{k-1}\left(X_{k-2} - X_{k+1}\right) - X_k + F - \left(\frac{hc}{b}\right)\sum_{j=0}^{J-1} Y_{j,k}$$

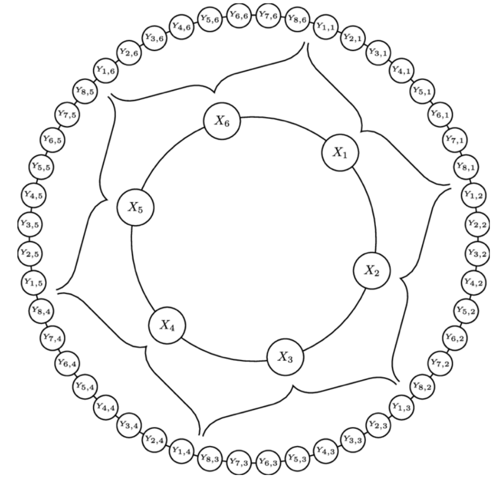
$$\frac{d}{dt}Y_{j,k} = -cbY_{j+1,k}\left(Y_{j+2,k} - X_{j-1,k}\right) - cY_{j,k} + \frac{hc}{b}X_k$$

A general circulation model (GCM) : coarse-grid, can’t resolve Y’s, instead uses a function P(X) that mimics effects of Y’s on the temporal tendency of X

$$\frac{d}{dt}X_k = -X_{k-1}\left(X_{k-2} - X_{k+1}\right) - X_k + F - P\left(X_k\right)$$

L96 Model to illustrate our goals

The truth



Y's= fast variables; subgrid/unresolved on a coarse-grid

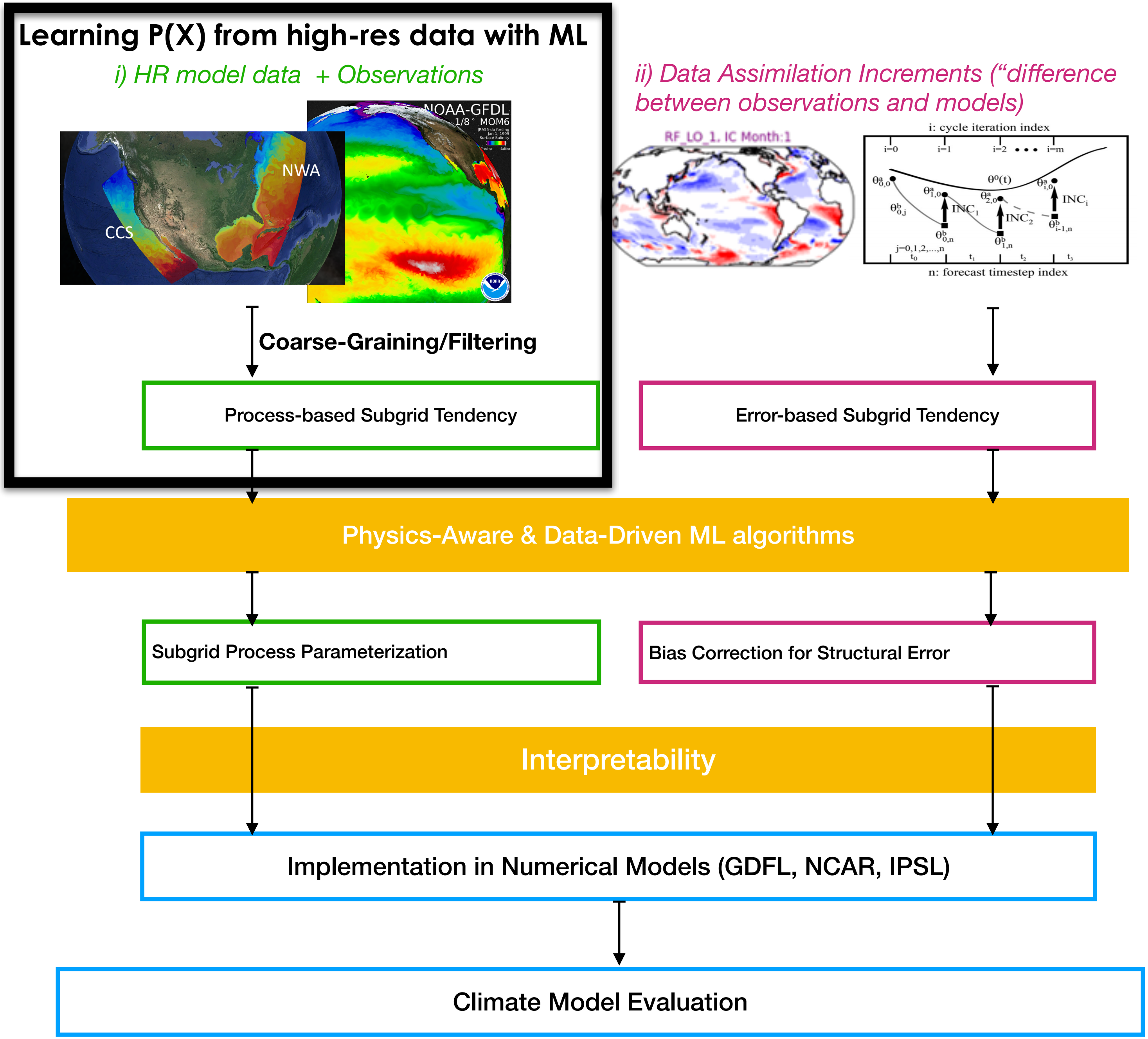
$$\frac{d}{dt}X_k = -X_{k-1}\left(X_{k-2} - X_{k+1}\right) - X_k + F - \left(\frac{hc}{b}\right)\sum_{j=0}^{J-1} Y_{j,k}$$

$$\frac{d}{dt}Y_{j,k} = -cbY_{j+1,k}\left(Y_{j+2,k} - X_{j-1,k}\right) - cY_{j,k} + \frac{hc}{b}X_k$$

A general circulation model (GCM) : coarse-grid, can't resolve Y's, instead uses a function P(X) that mimics effects of Y's on the temporal tendency of X

$$\frac{d}{dt}X_k = -X_{k-1}\left(X_{k-2} - X_{k+1}\right) - X_k + F - P\left(X_k\right)$$

Can we learn P(X) from data, and couple it to the coarse-resolution model (GCM) to improve its performance?



Proposal Work Plan

WP1: ML models

WP2: process-based closures

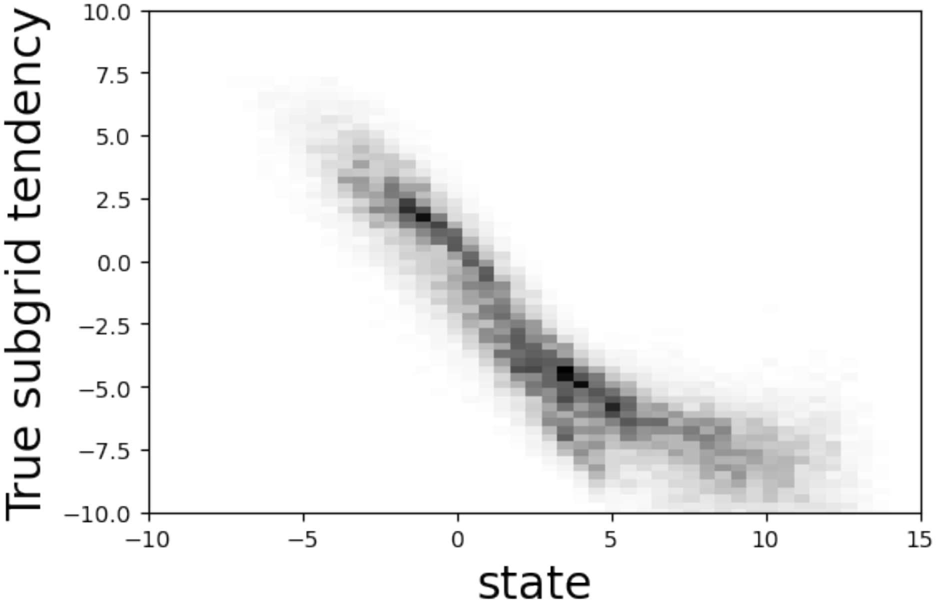
WP3: error-based closures

WP4: implementation/evaluation

Learning P(X) with ML from High-res data

Diagnosed effects of the fast variables

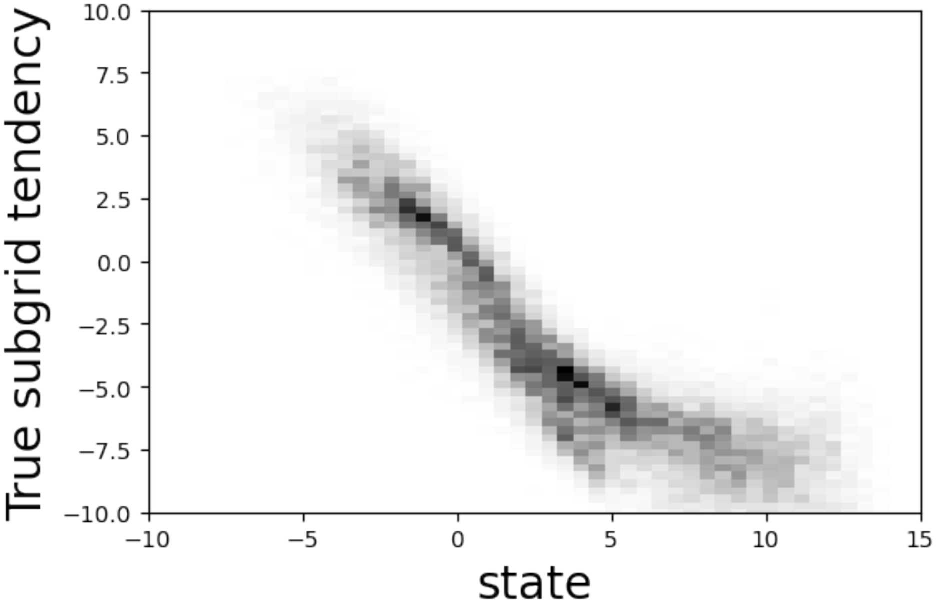
Y on dX/dt as a function of the state X



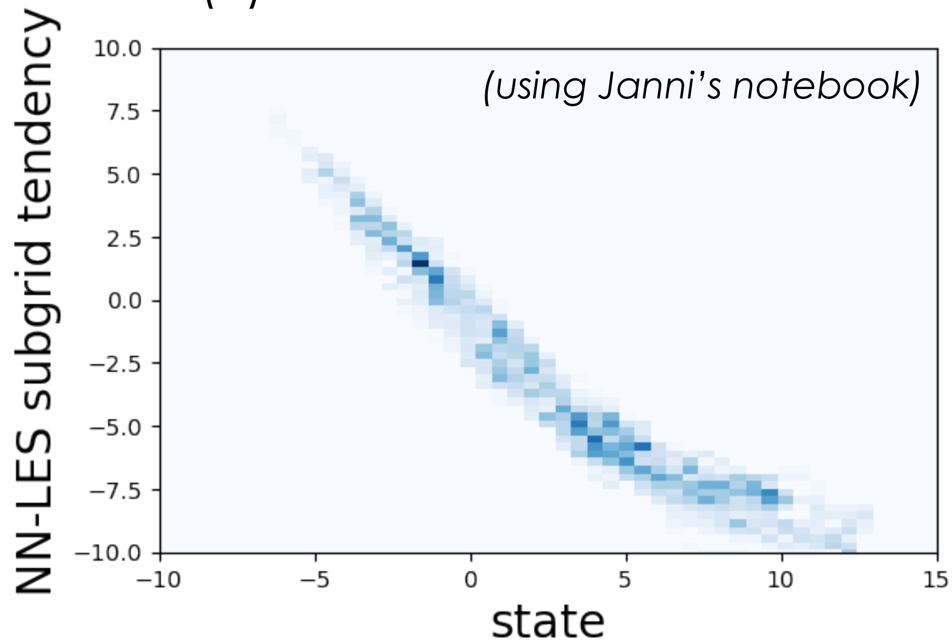
$$\frac{d}{dt}X_k = -X_{k-1}\left(X_{k-2} - X_{k+1}\right) - X_k + F \boxed{- \left(\frac{hc}{b}\right)\sum_{j=0}^{J-1} Y_{j,k}}$$

Learning P(X) with ML from High-res data

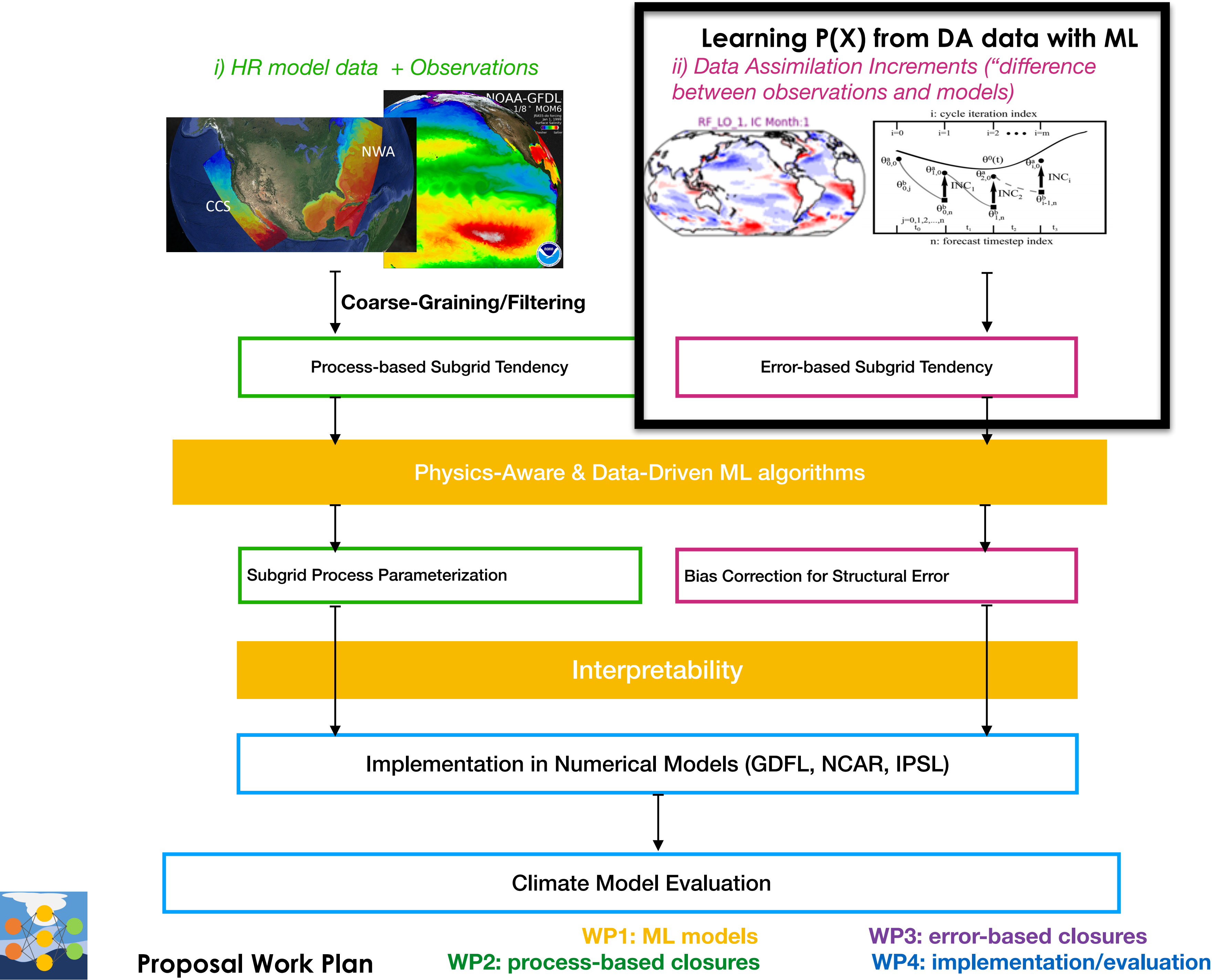
Diagnosed effects of the fast variables Y on dX/dt as a function of the state X



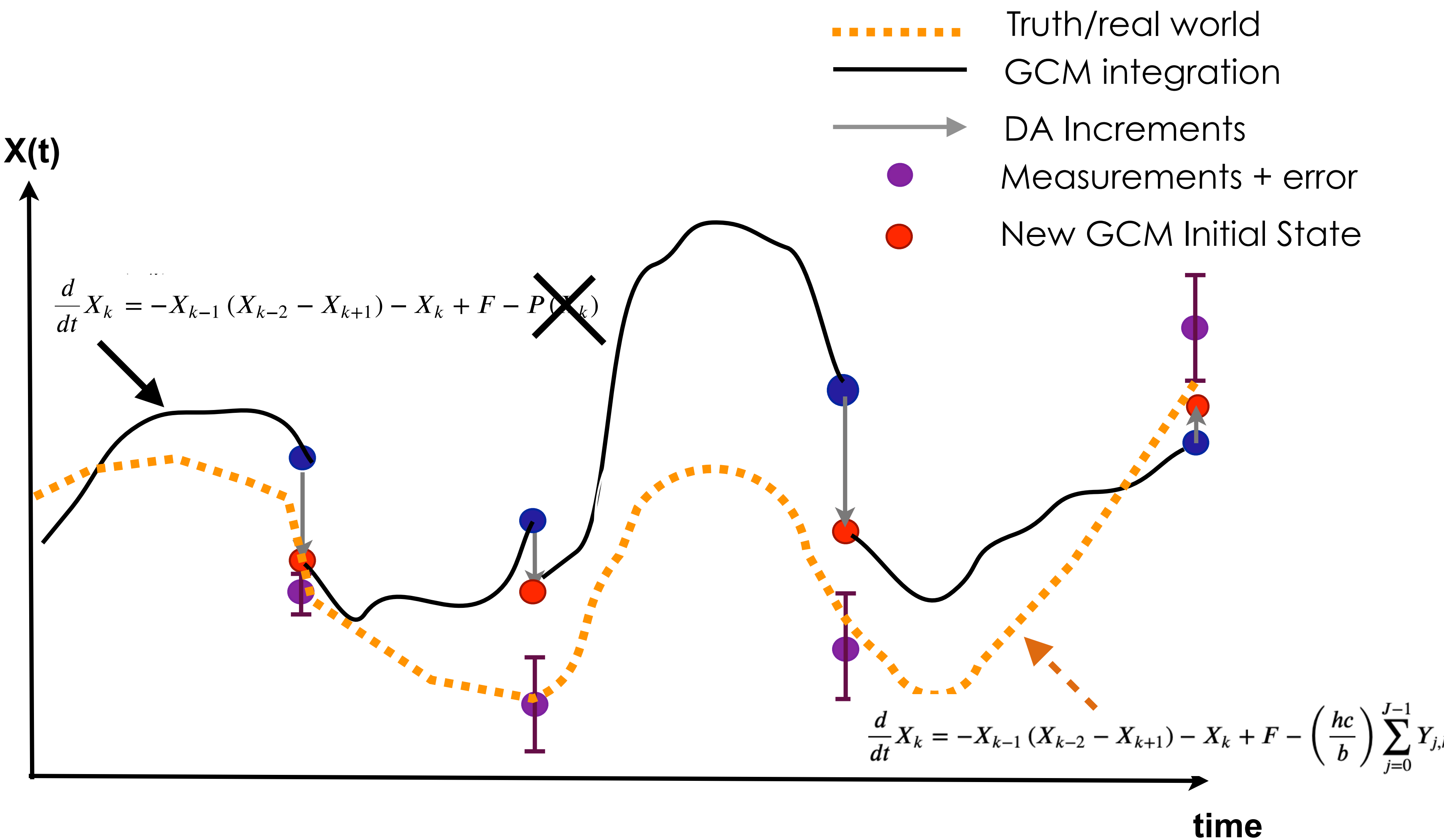
Learned effects of Y on dX/dt = P(X) as a function of X



$$\frac{d}{dt}X_k = -X_{k-1}\left(X_{k-2} - X_{k+1}\right) - X_k + F \boxed{- \left(\frac{hc}{b}\right)\sum_{j=0}^{J-1} Y_{j,k}}$$


i) HR model data + Observations
NOAA-GFDL
1/8° MOM6
NWA
CCS
Learning P(X) from DA data with ML
ii) Data Assimilation Increments (“difference between observations and models)
RF_LO_1, IC Month:1
i: cycle iteration index
n: forecast timestep index
Coarse-Graining/Filtering
Process-based Subgrid Tendency
Error-based Subgrid Tendency
Physics-Aware & Data-Driven ML algorithms
Subgrid Process Parameterization
Bias Correction for Structural Error
Interpretability
Implementation in Numerical Models (GDFL, NCAR, IPSL)
Climate Model Evaluation
Proposal Work Plan
WP1: ML models
WP2: process-based closures
WP3: error-based closures
WP4: implementation/evaluation

Data Assimilation (DA) "Review"



Learning P(X) with ML from DA Increments

Our (=Alistair's) notebooks are based on Janni's ML Notebook + Feiyu's DA Notebook

- Using "standard settings" for the system DA to get a reasonable structure of the DA increments then train a NN to learn the "missing physics" that the DA increments captured

- Sensitivity to observations sampling

- Sensitivity due to a structural model bias (by changing F of the GCM)