NSTX-U

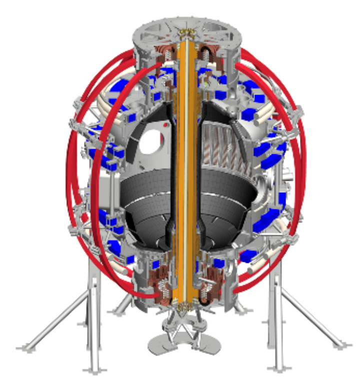
NSTX-U is sponsored by the
U.S. Department of Energy Office of Science
Fusion Energy Sciences

Preliminary results of modeling neutral beam injection on NSTX-U using neural networks (NubeamNet)

Dan Boyer, Stan Kaye

Tuesday science meeting, 12/5/2017

PPPL

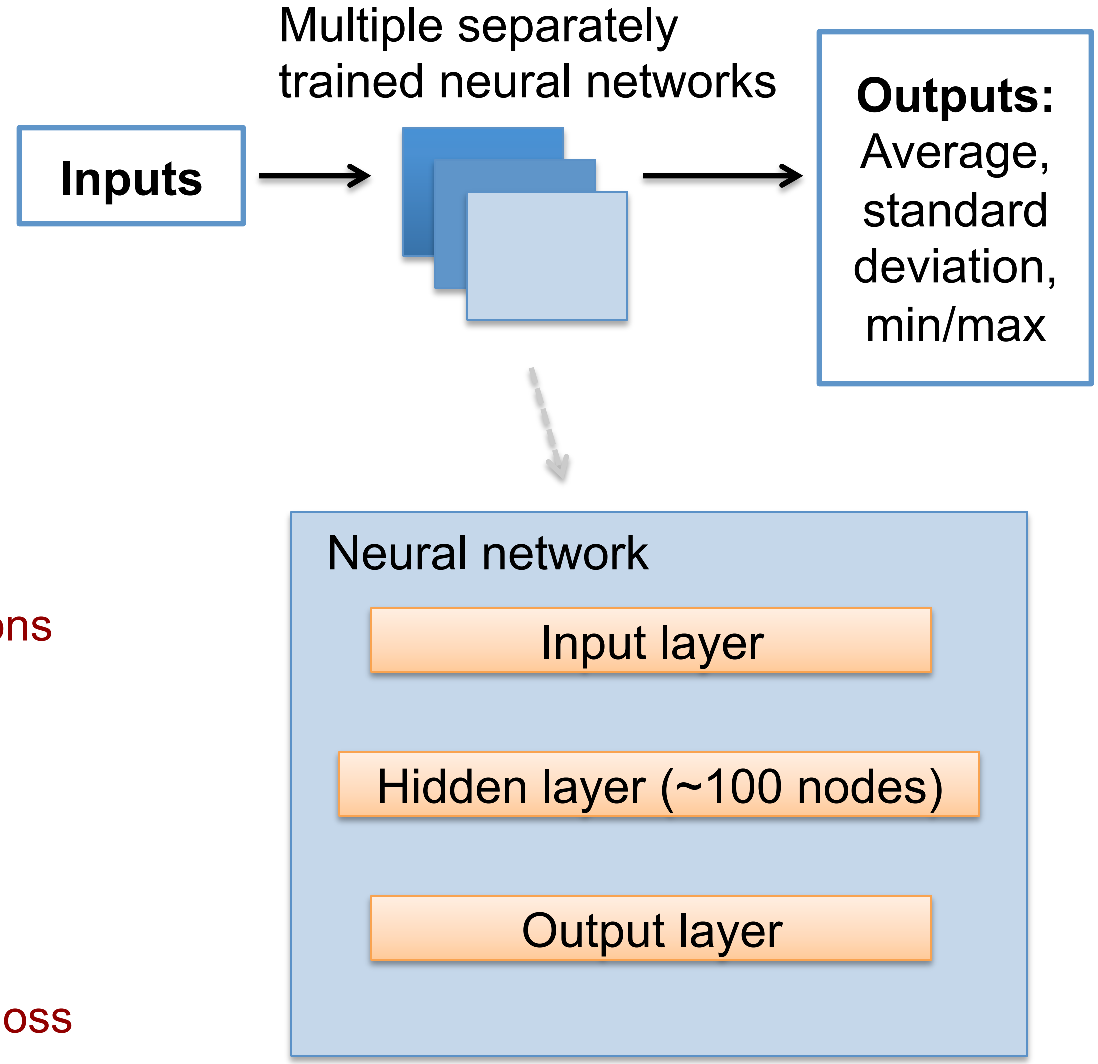


Predictive models that run in near- to faster-than- real-time will enable improved control and scenario development algorithms

- Neural networks have recently been developed for approximating the results of computationally intensive calculations
  - Meneghini NF 2017, 2014 (TGLF, EPED), Citrin NF 2015 (QuaLiKiz)
- NUBEAM often takes 30% or more of TRANSP time
  - Lower fidelity settings can speed up results but results become noisy
- **Can a neural network be trained to reproduce the result of NUBEAM?**
- Potential applications
  - Fast but realistic beam calculations for control-oriented simulations or use in real-time predictive control algorithms
  - Fast predictions to optimize neutron rate matching in TRANSP runs
  - Prediction of fast ion pressure profile for kinetic EFITs
    - Fast enough iterations for real-time implementation
  - Control room tools for P.O or S.L. to explore beam timing options prior to shot

Inputs, outputs, and topology of the neural network model

- Inputs:
  - Profiles:
    - T_e, n_e, q
    - *fast ion diffusivity*
  - Scalars:
    - Beam powers
    - *Edge neutral density*
    - Z_{eff}
- Outputs:
  - Profiles:
    - Beam heating to ions/electrons
    - Beam driven current
    - Beam torque
    - Fast ion pressure
  - Scalars:
    - Neutron rate
    - Shine through
    - Charge-exchange and orbit loss



A data set was prepared based on the TRANSP runs performed between NSTX-U shots (BEAST)

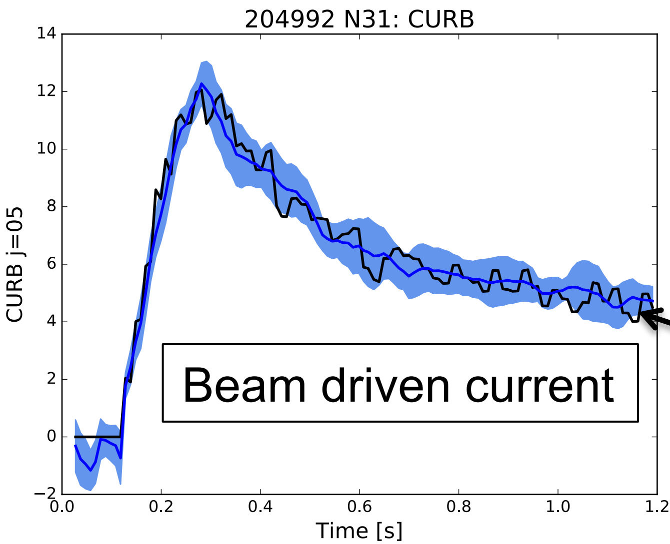
- Expanded the dataset with a **scan of Z_{eff}, anomalous fast ion diffusivity, and edge neutral density**
  - Randomly selected **~2000 cases** from the grid scan to actually run for initial testing
  - Used **low fidelity settings** for speed for initial testing
    - Results are noisy but **NN can smooth** them
- Projected profiles onto **basis functions**
    - Reduced 20 grid points per profile **to 4 mode coefficients per profile**
    - Reduces training time, also results in **smooth-in-x profiles**
- Assigned 80 of ~300 shots in the dataset to the **'testing' data set**
  - No data from any simulations of these shots is used in training the model
- Total of **~200k time slices**

The beam slowing down time causes NUBEAM results to depend on time history…

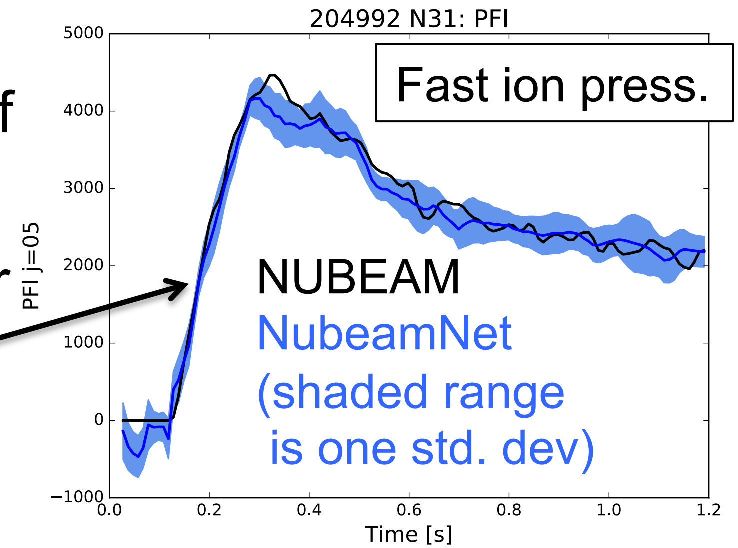
- **Simplest** approach to modeling:
  - Ignore time history, assume **steady-state**, only use instantaneous values of inputs
  - Probably not always suitable for planned applications
    - e.g., Beam modulation during control

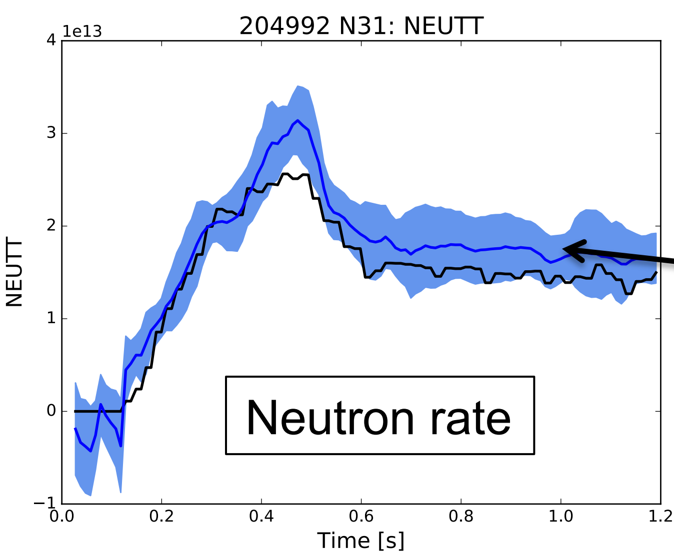
- The **next simplest** approach:
  - Expand inputs with **filtered beam powers**
    - **Multiple time constants** to account for changes in slowing down time
  - Not accounting for time history of plasma parameters
    - Fewer inputs, fewer nodes to train on
    - Plasma parameters evolve fairly slowly compared to slowing down time and beam modulation time

Time traces of NN compare well with NUBEAM for shots in testing data set

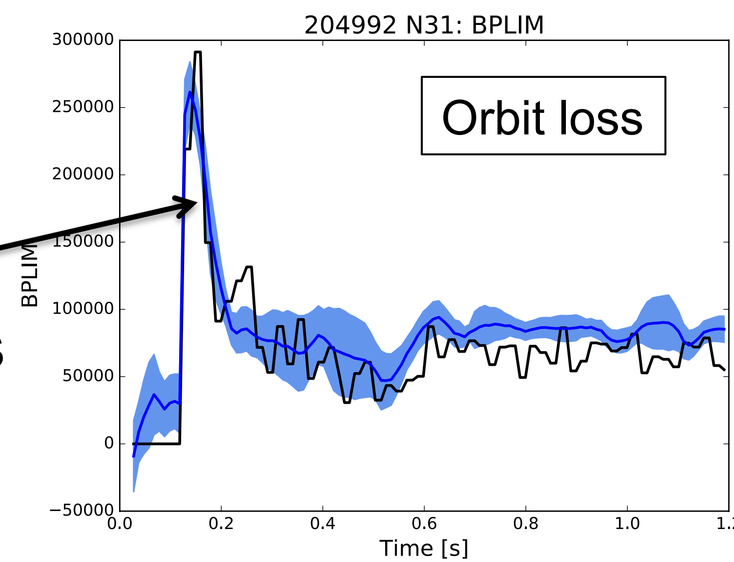


- Good matching of time history of current drive and fast ion pressure near axis

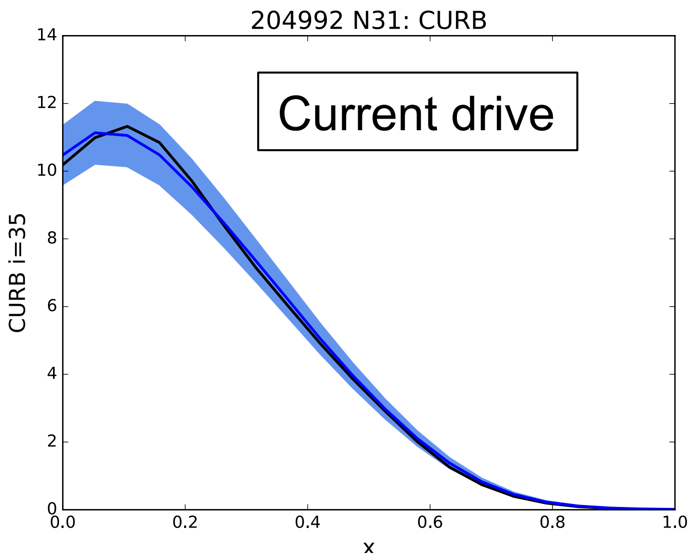


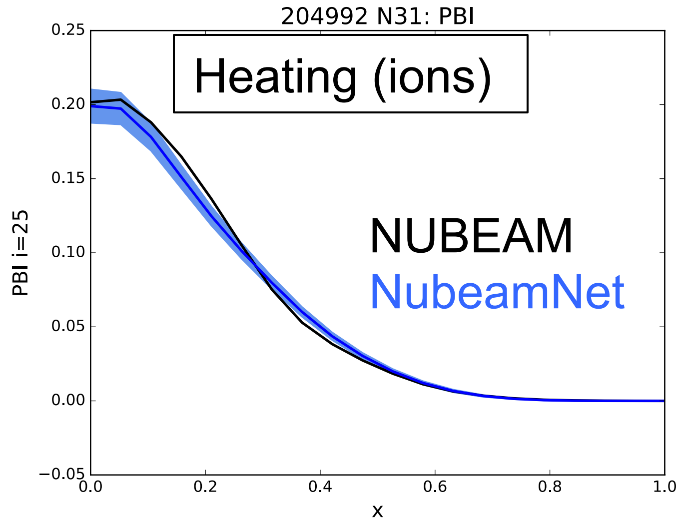


- Also good matching of scalars like neutron rate and orbit loss

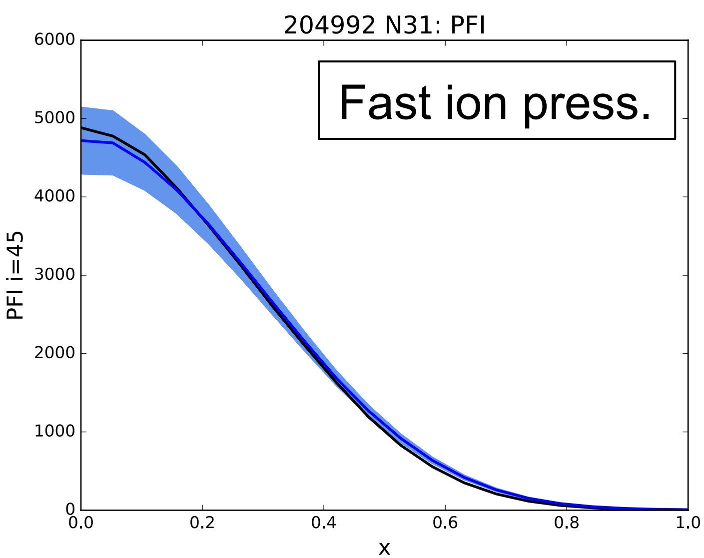


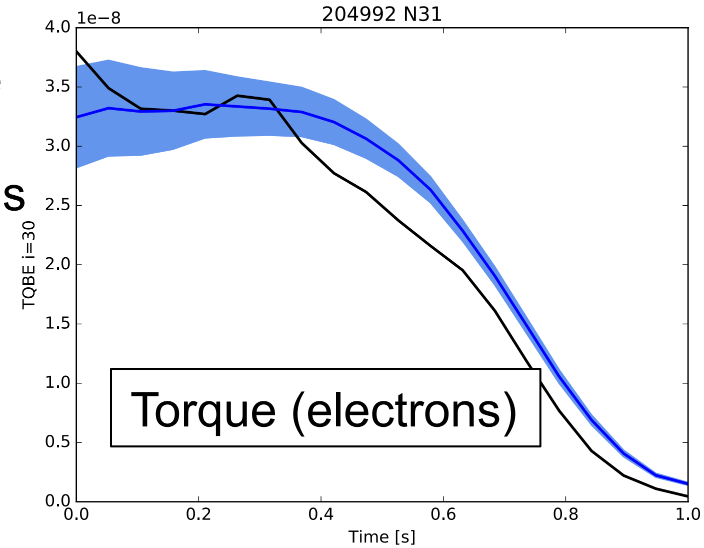
Profiles show good agreement between NUBEAM and neural network prediction



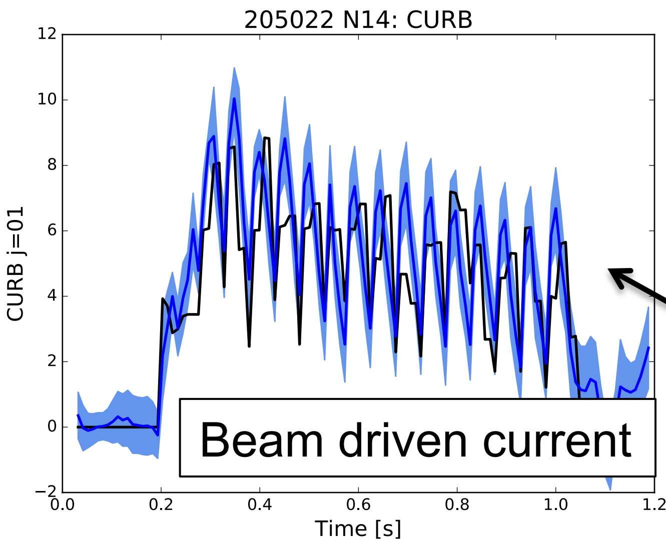


- NUBEAM profiles averaged over 3 slices
- Good matching, smooth profiles predictions
- Torque from NUBEAM for these runs is spatially noisy – NN smooths this out

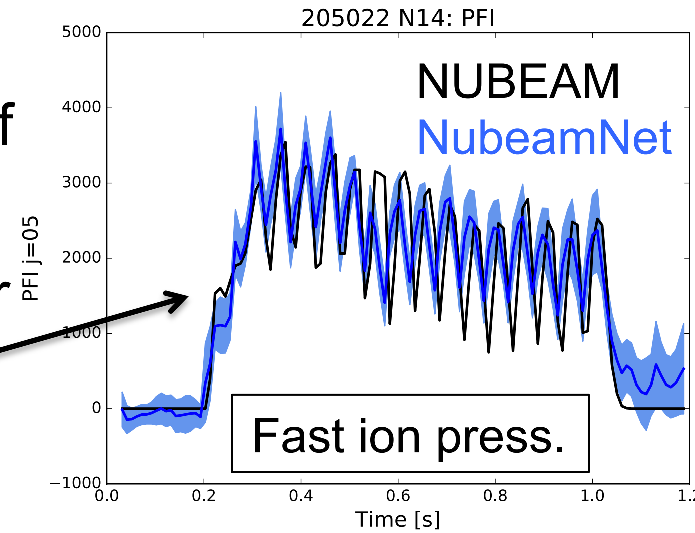


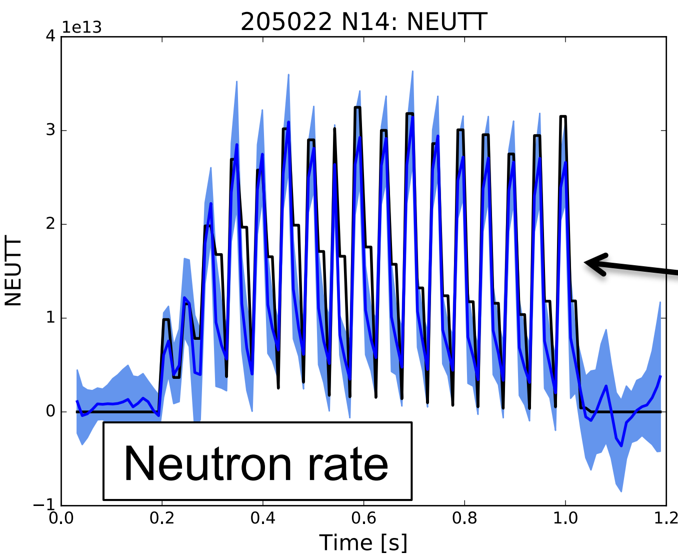


Time traces compare fairly well during beam blip shots in testing data set

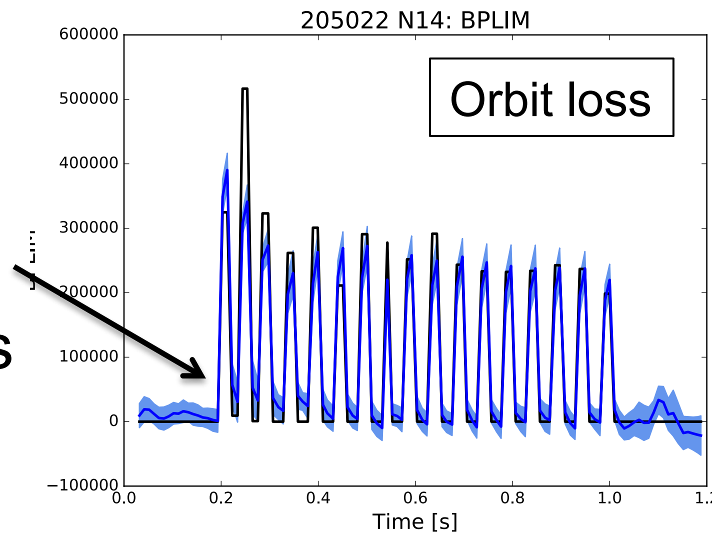


- Good matching of time history of current drive and fast ion pressure near axis



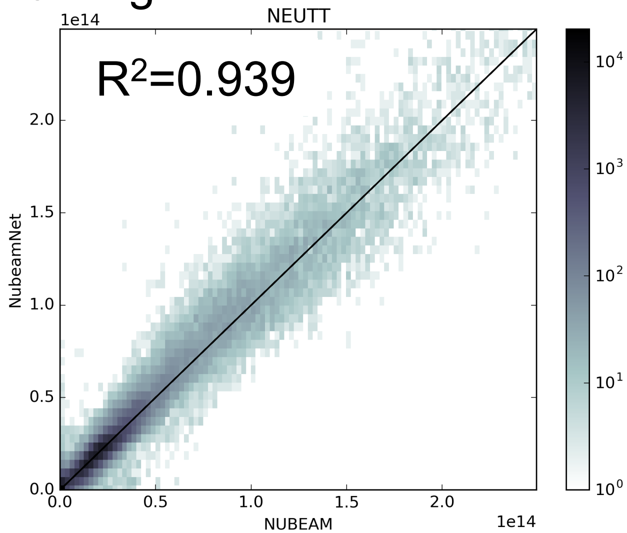


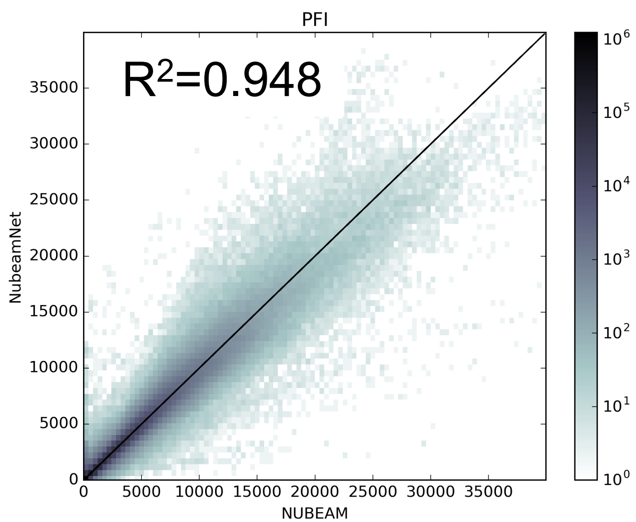
- Also good matching of scalars like neutron rate and orbit loss



Regression plots for training and testing data set show good fitting and generalization

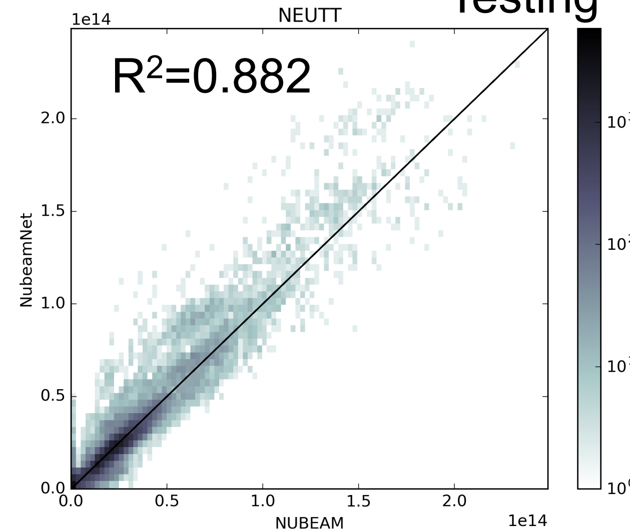
Training

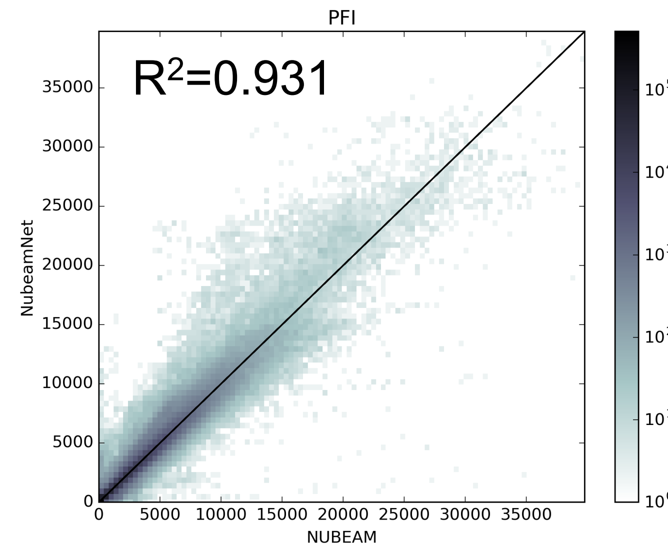




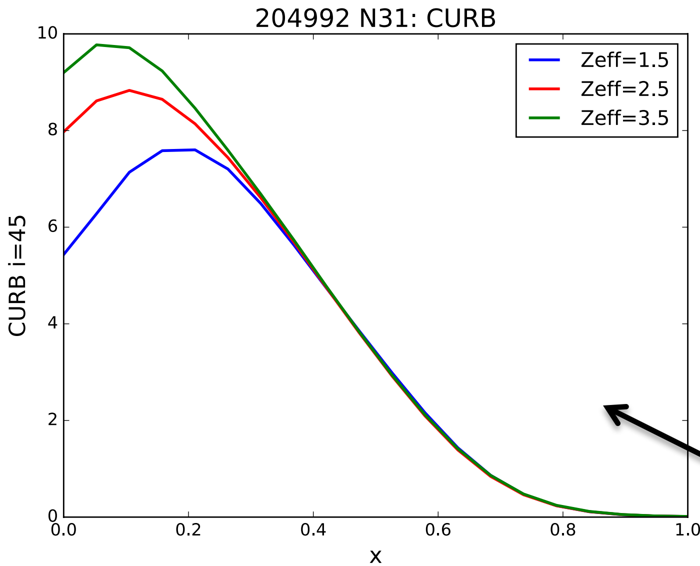
- **Log scale histograms**
- R^2 drops in testing data set but not too bad
  - Will continue to optimize neural network topology, add more data, etc. to improve generalization
- Some parameters were pretty noisy, resulting in lower R^2 (>0.8)

Testing

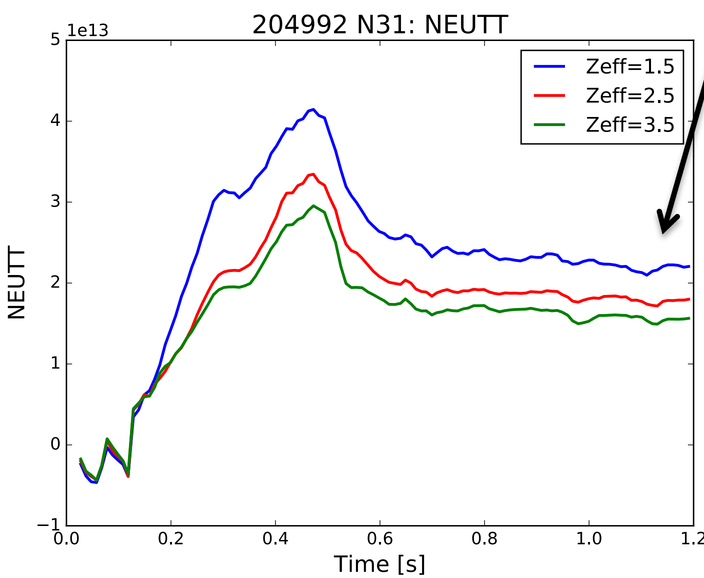


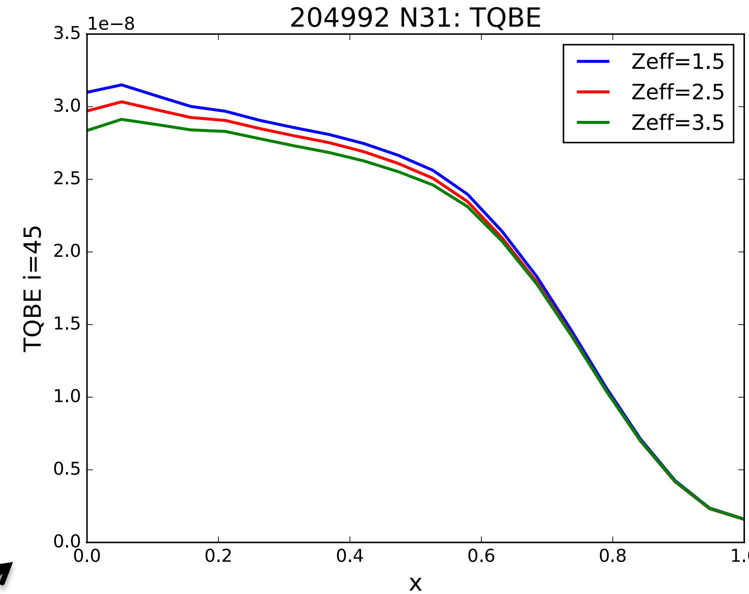


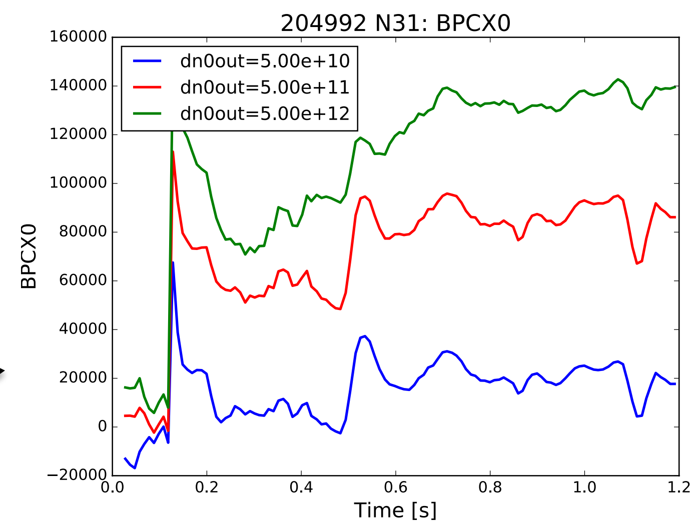
Neural net enables rapid scans of parameters (<<1s per shot)



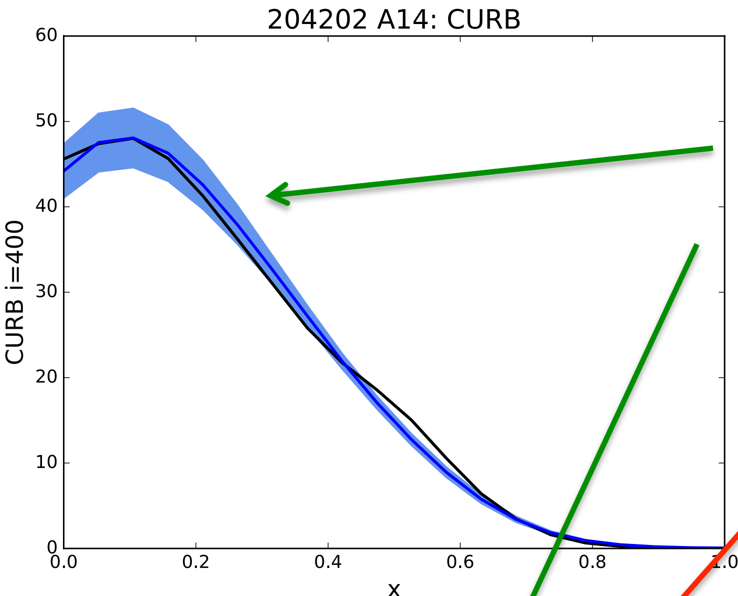
- Scans of single parameters with other inputs fixed
- Z_{eff} affects current drive, neutron rate, small effect on torque (electrons)
- Edge neutral density effects CX losses



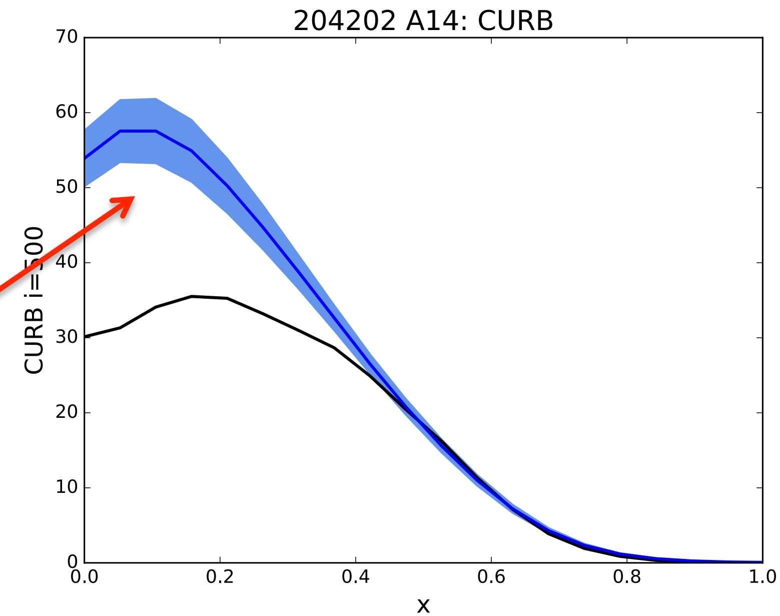


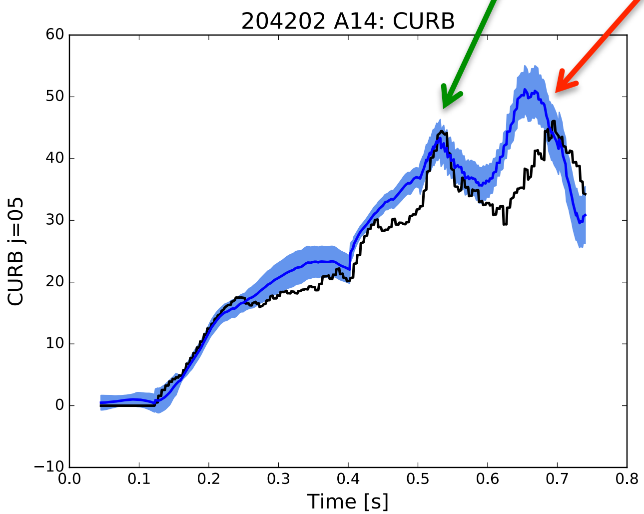


Testing on a higher fidelity run that was not included in training data set



Sometimes great, sometimes not so great…





- Results may improve with additional low fidelity training data
- However, probably will need to train on higher fidelity scans

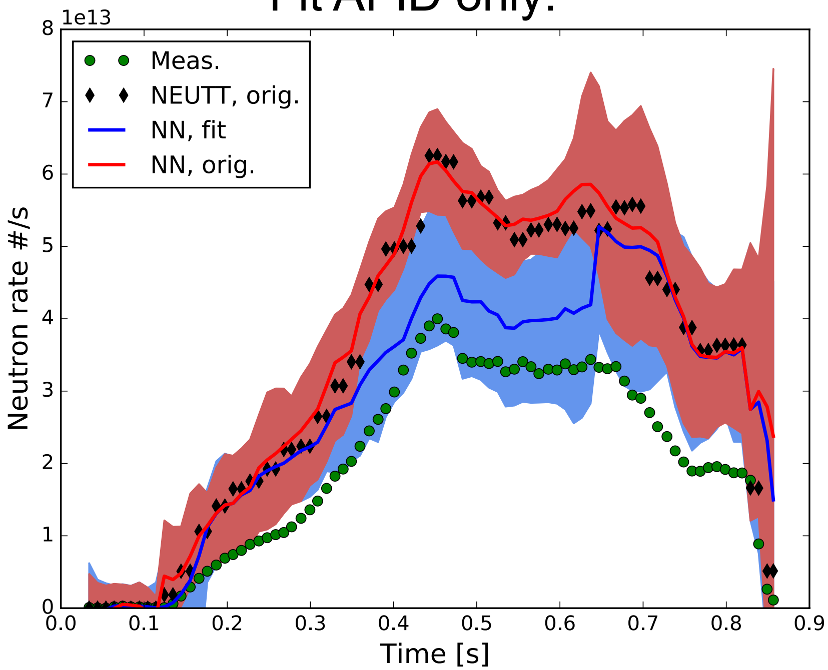
Example application: Fitting free/uncertain parameters to match measured neutron rate

- One of the steps in interpretive TRANSP runs is to match the predicted and measured neutron rates
  - Find values of fast ion diffusivity, external neutral density, and/or Z_{eff} since these are not well constrained
- Typically done with scans
  - Parameters can be time-varying so its hard to match the neutron rate at all times
- Recently added a feedback algorithm in TRANSP
  - Adjusts fast ion diffusivity based on error in neutron rate prediction
  - Automates the matching process
    - Useful for between shots (BEAST) runs

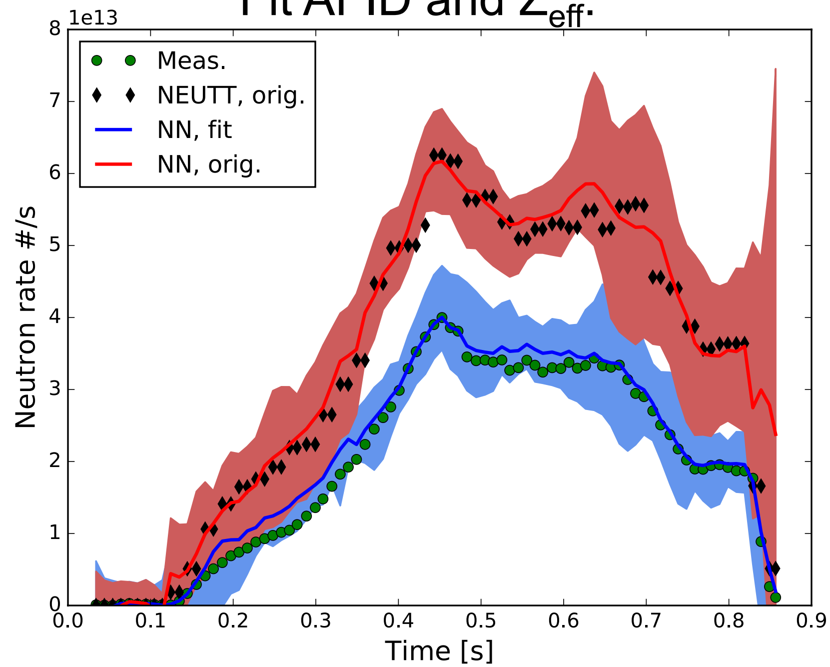
Fast execution time of neural network enables optimization of free parameters – could provide ‘feedforward’ for AFID controller

- Find fast ion diffusivity profile, edge neutral density, and/or Z_{eff} that **minimizes neutron matching error**
  - More free parameters than errors to minimize
  - Solution: **Regularize by weighting ensemble uncertainty**
    - i.e., find the solution that best matches neutron rate while staying in the range of inputs that the model has confidence in

Fit AFID only:



Fit AFID and Z_{eff}:



Future work

- More data, more devices…
  - Generate more runs, use higher fidelity runs, poach existing runs…
  - Apply approach to DIII-D, KSTAR, etc.
- w/ S. Sabbagh and Columbia KSTAR collaboration:
  - Use NubeamNet prediction of fast ion pressure profile in kinetic EFIT iterations to reduce error bars while avoiding the need for TRANSP/ NUBEAM in the loop
- Develop/test/deploy AFID fitting for routine use with TRANSP runs
- Implement NubeamNet in PCS for real-time applications
  - Real-time kinetic EFIT, profile control
  - Power balance monitoring

Discussion

- Other outputs of NUBEAM that would be useful to include?
- Suggested settings for high fidelity scans?
- Other potential applications of the model or modeling approach?
  - RF codes?