Machine Learning Based Reconfiguration of the BNL ATR Line

Jonathan Edelen, Nathan Cook, Kevin Brown, and Philip Dyer
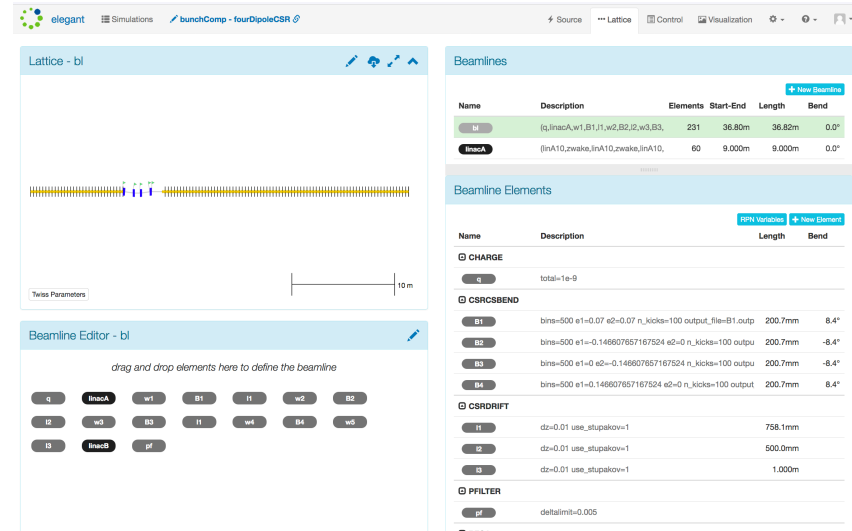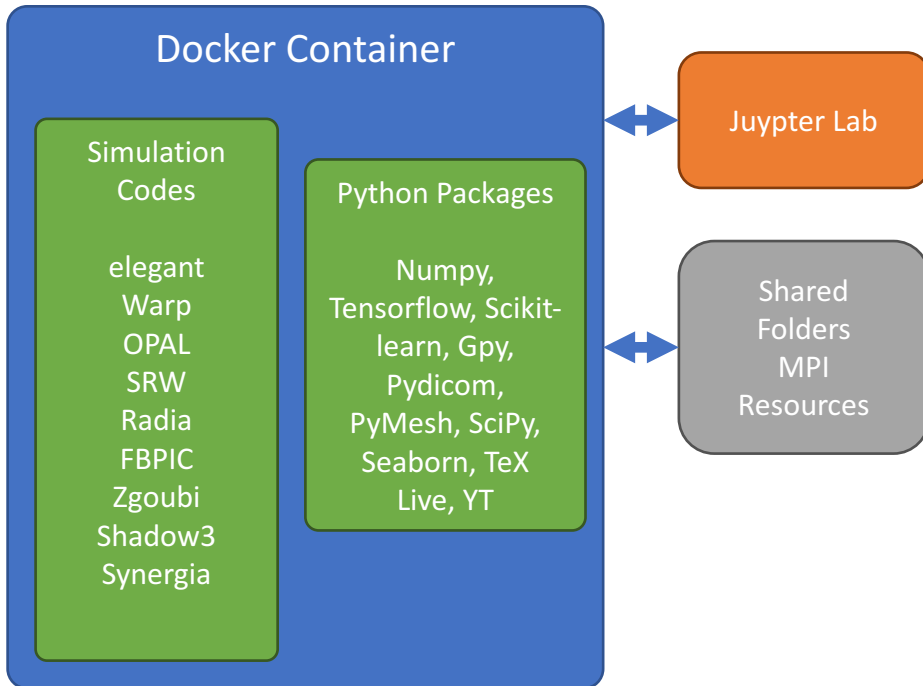
presented at ICALEPCS 2019: Data Science and Machine Learning Workshop

(Brooklyn, NY)

6 Oct 2019

Jupyter/Hub

Sirepo

Docker Container

Simulation Codes

elegant
Warp
OPAL
SRW
Radia
FBPIC
Zgoubi
Shadow3
Synergia

Python Packages

Numpy,
Tensorflow, Scikit-learn, Gpy,
Pydicom,
PyMesh, SciPy,
Seaborn, TeX
Live, YT

Juypter Lab

Shared
Folders
MPI
Resources

D.L. Bruhwiler *et al*., "Knowledge Exchange Within the Particle Accelerator Community via Cloud Computing," in *IPAC* (2019).

https://jupyter.radiasoft.org, and https://sirepo.com are free Scientific Gateways

Developing machine learning tools @ Radiasoft

Data Import	Initial Data Visualization	Machine Learning Selection
Simulation Batches	Plotting Tools	Model Selection
Machine Archiver Interface	Data Cleaning / Reduction	Learning Paradigm
Direct Data Importing	Train / Validation / Test Split	Loss Function

Developing machine learning tools @ Radiasoft
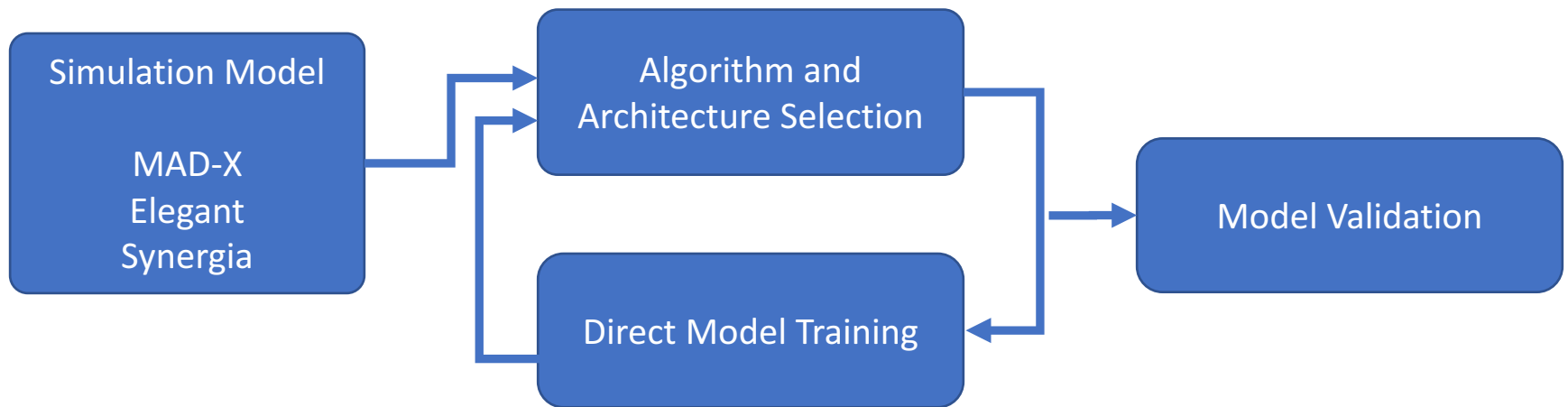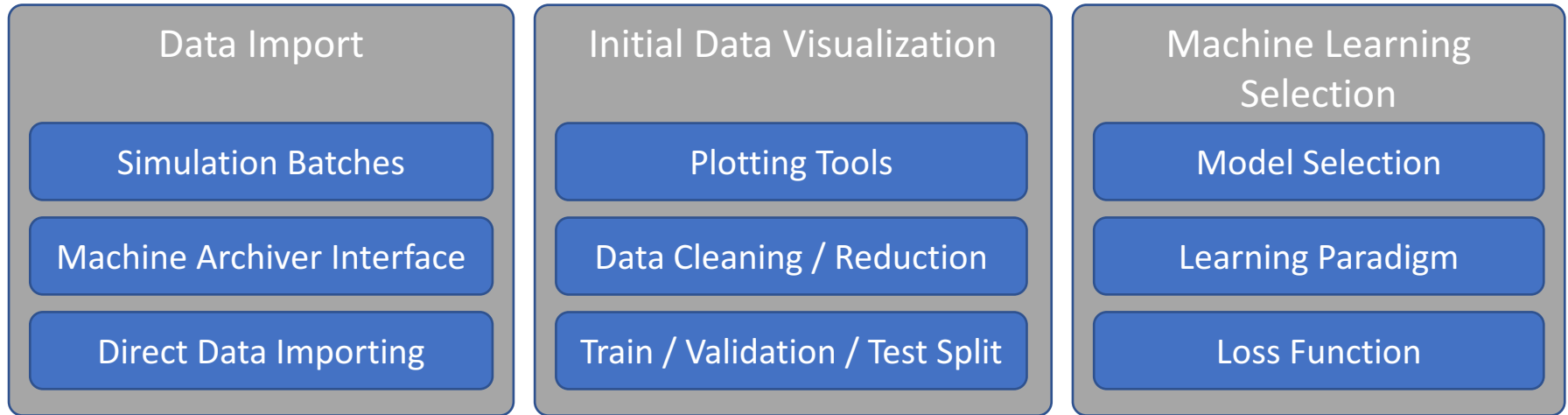
Data Import	Initial Data Visualization	Machine Learning Selection
Simulation Batches	Plotting Tools	Model Selection
Machine Archiver Interface	Data Cleaning / Reduction	Learning Paradigm
Direct Data Importing	Train / Validation / Test Split	Loss Function

Simulation Model

MAD-X
Elegant
Synergia

→ Algorithm and Architecture Selection

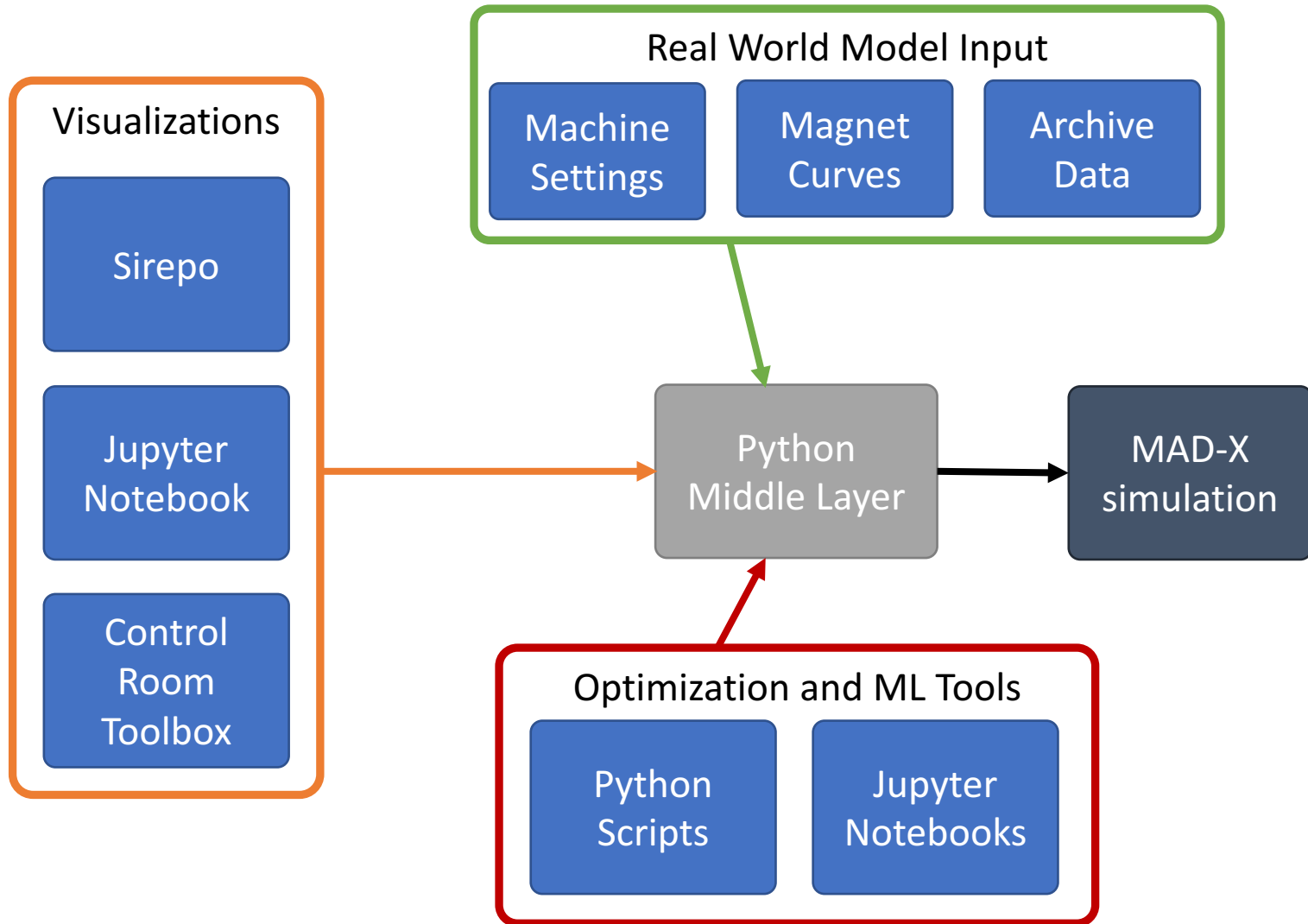Direct Model Training

Model Validation

The AGS to RHIC transfer line

- 600 Meter transfer line brings beams from the Alternating Gradient Synchrotron (AGS) to the Relativistic Heavy Ion Collider (RHIC)

- Energy scan requires re-tuning of the ATR line
  - *Match the beam trajectory*
  - *Match transverse optics*
  - *Make spin transparent*

Injection Section   Injection Section

X-line   **Dump**   Y-line

W-line

1.7 m level drop 20^0 bend

U-line

8^0 bend

4.25^0 bend

AGS Extraction (H13)

radiasoft

The AGS to RHIC transfer line
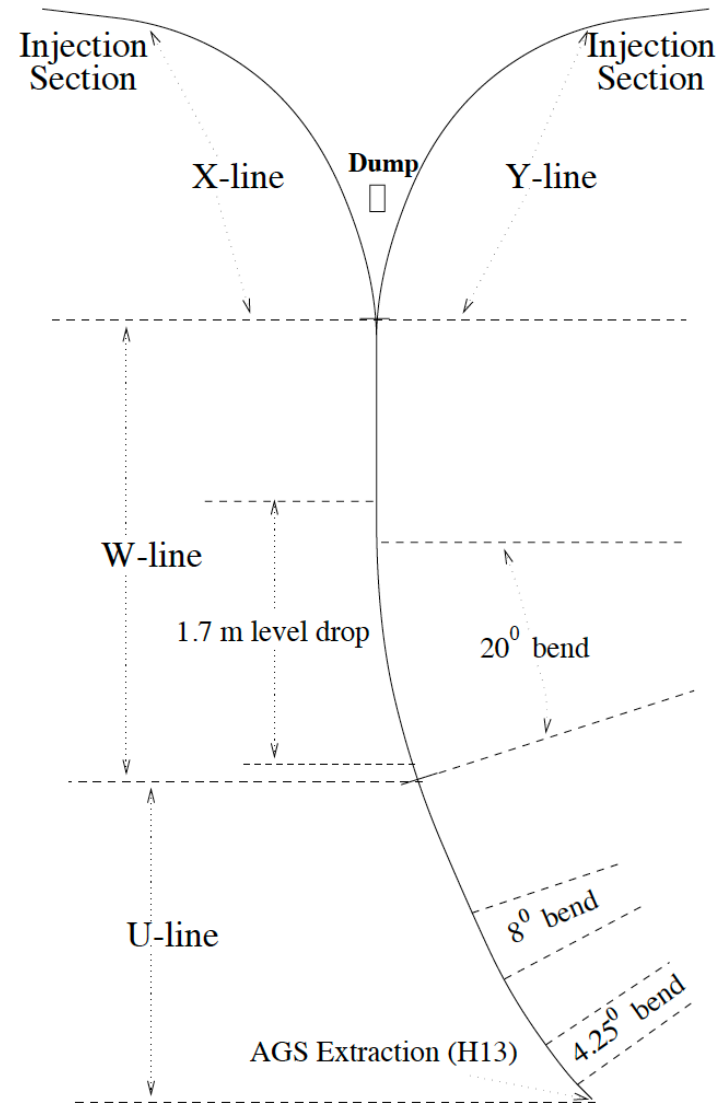
- 600 Meter transfer line brings beams from the Alternating Gradient Synchrotron (AGS) to the Relativistic Heavy Ion Collider (RHIC)

- Energy scan requires re-tuning of the ATR line
  - *Match the beam trajectory*
  - *Match transverse optics*
  - *Make spin transparent*
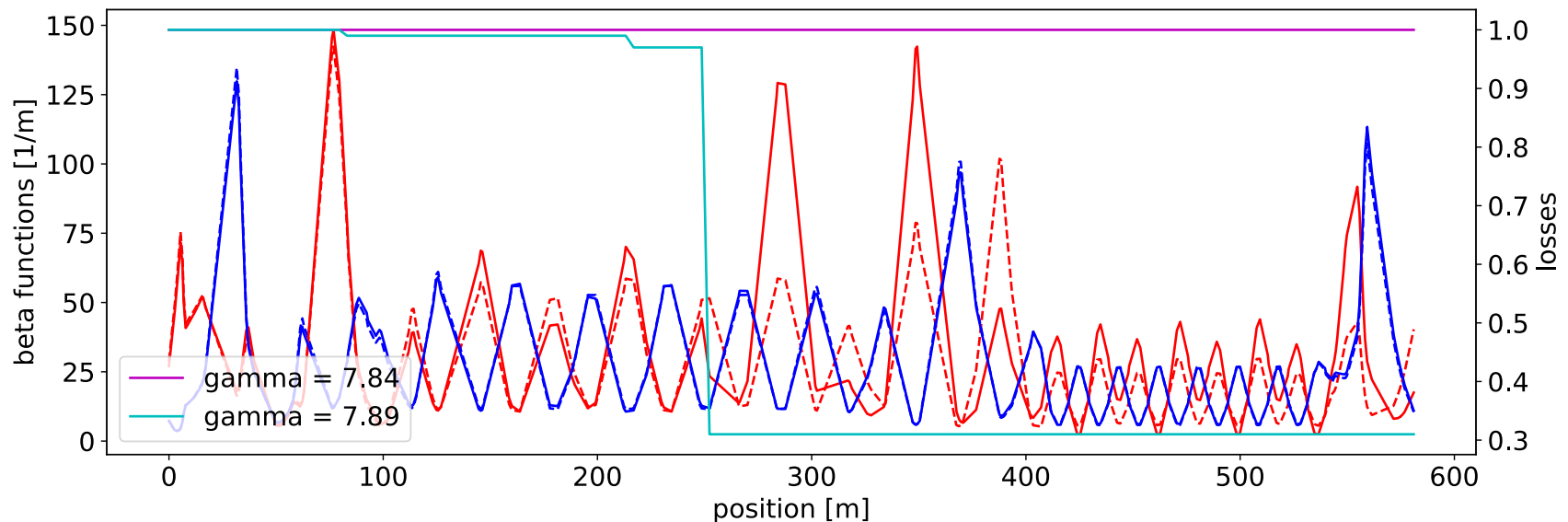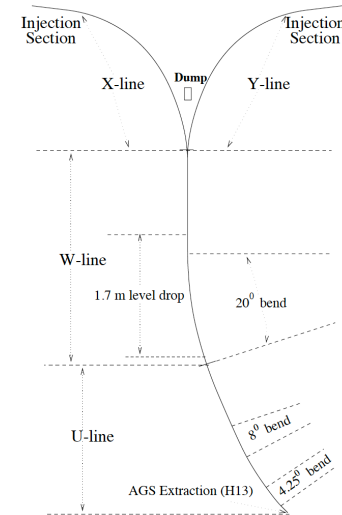
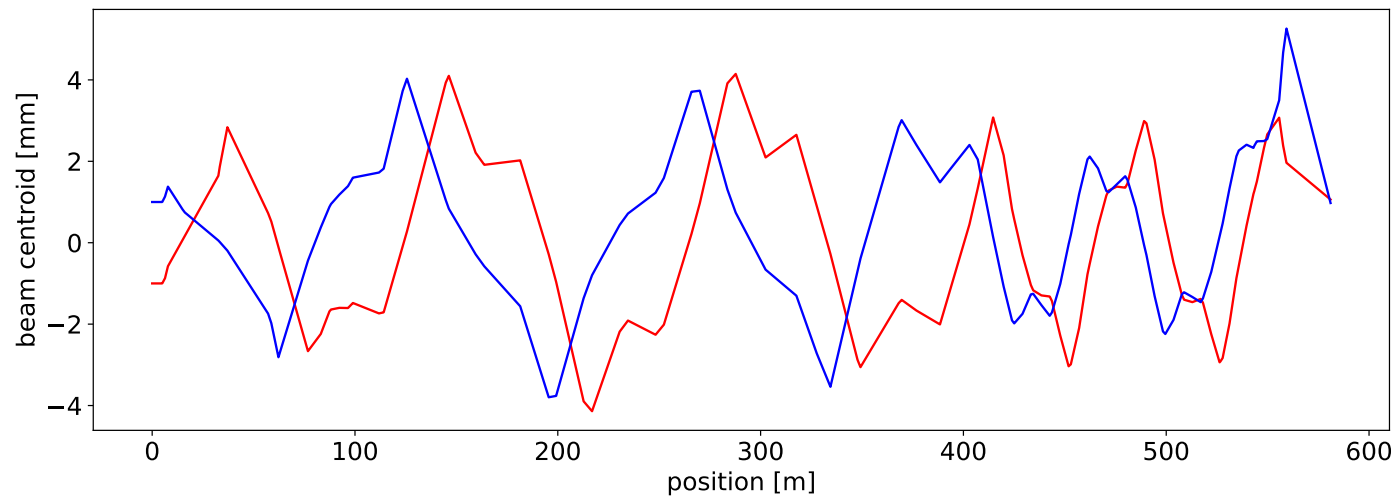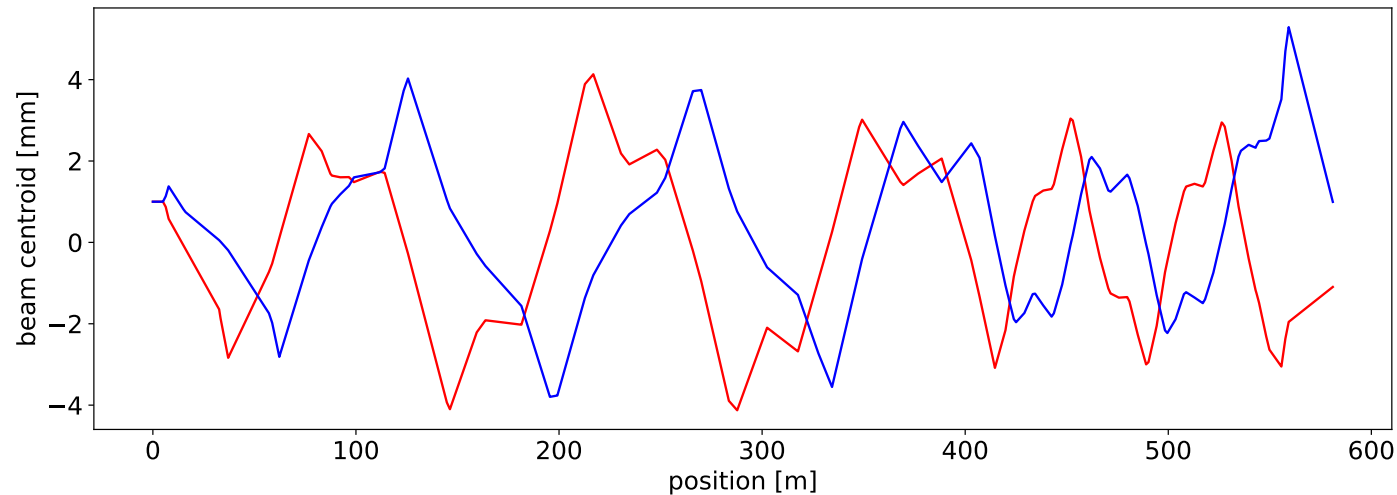# Developing a toolbox for simulations and ML
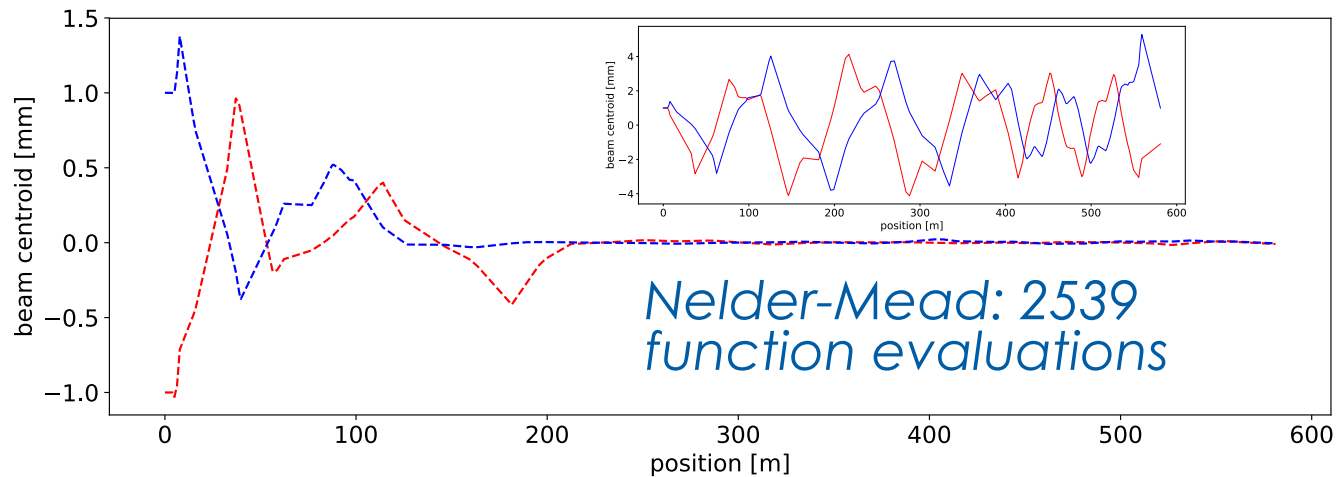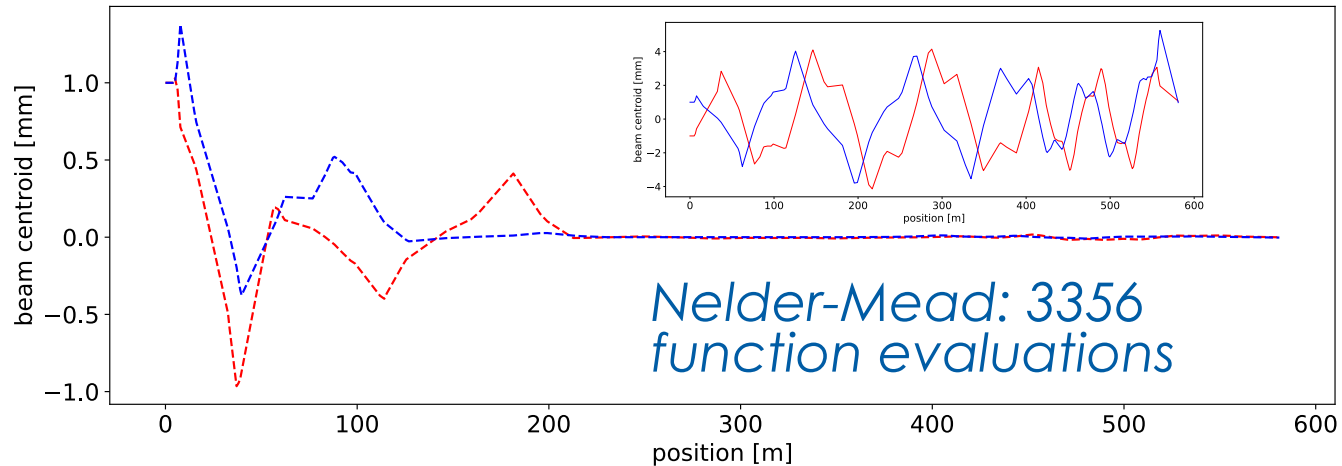
Beam steering in the ATR

- Optimization
  - *Connect MAD-X simulation to python optimization tools using our middle layer*
  - *Study convergence rate for tuning the trajectory over a range of initial offsets*

- Machine Learning
  - *Build neural network model of corrector settings to bpm-readings*
    - Use with optimization
  - *Build inverse model of bpm-readings to corrector settings*
    - Make feed-forward correction

Beam steering in the ATR with optimization

- Optimization
  - *Connect MAD-X simulation to python optimization tools using our middle layer*
  - *Study convergence rate for tuning the trajectory over a range of initial offsets*

Beam steering in the ATR with optimization

- Optimization
  - *Connect MAD-X simulation to python optimization tools using our middle layer*
  - *Study convergence rate for tuning the trajectory over a range of initial offsets*
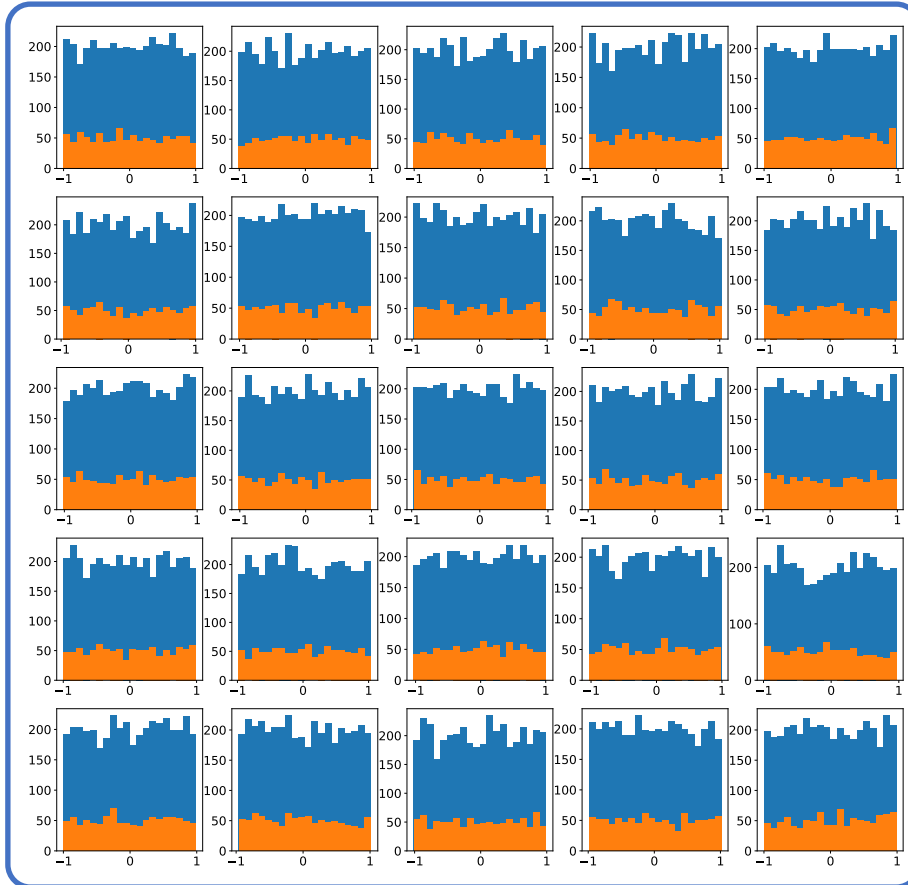
Beam steering in the ATR

- Optimization
  - *Connect MAD-X simulation to python optimization tools using our middle layer*
  - *Study convergence rate for tuning the trajectory over a range of initial offsets*

- Machine Learning
  - *Build neural network model of corrector settings to bpm-readings*
    - Use with optimization
  - *Build inverse model of bpm-readings to corrector settings*
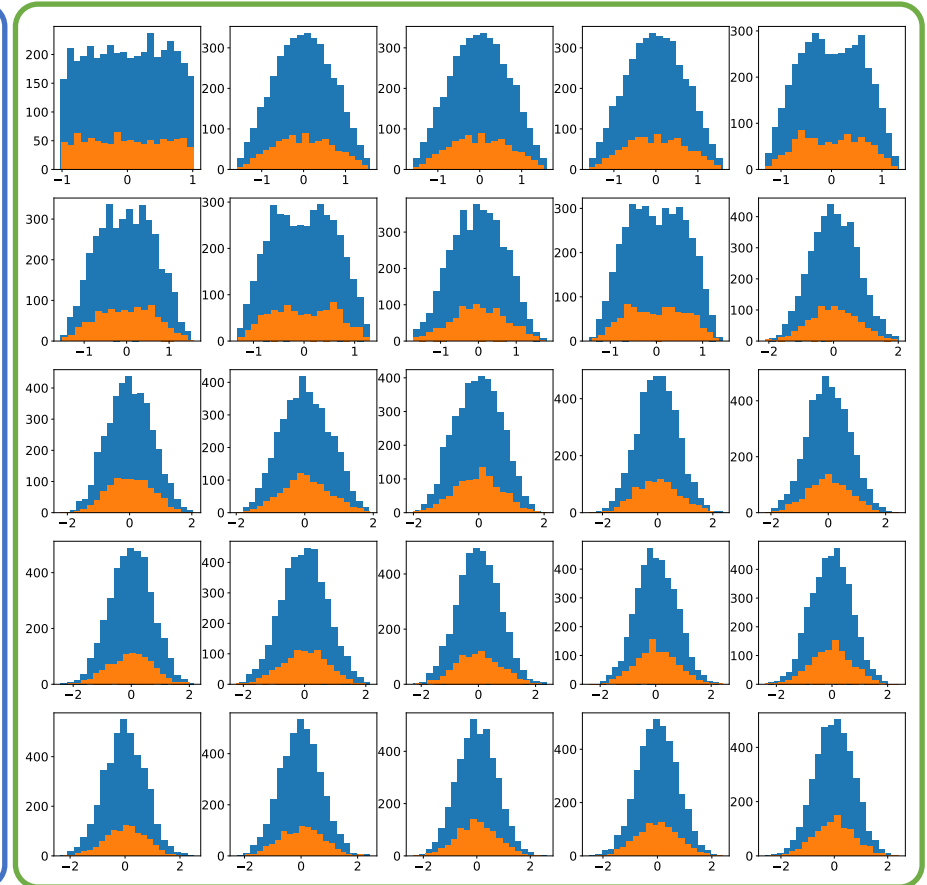   - Make feed-forward correction

Beam steering in the ATR with Machine Learning

- Forward model at a fixed energy
  - *Random input offsets and corrector settings*

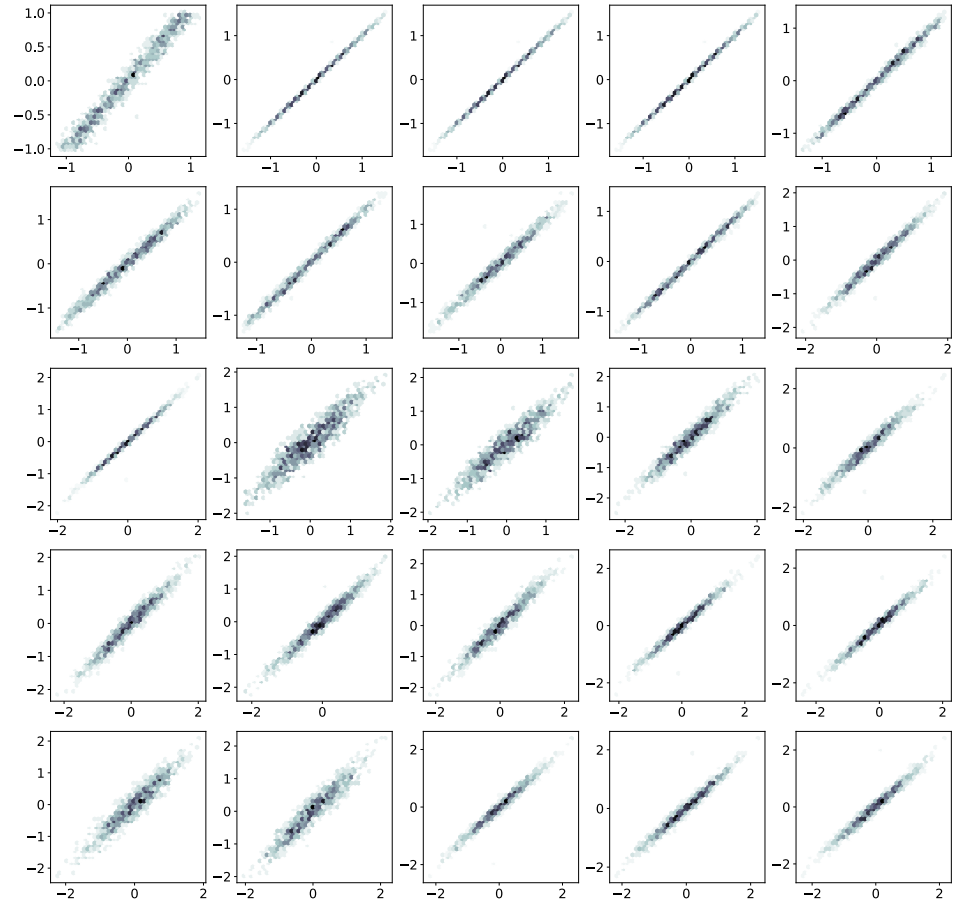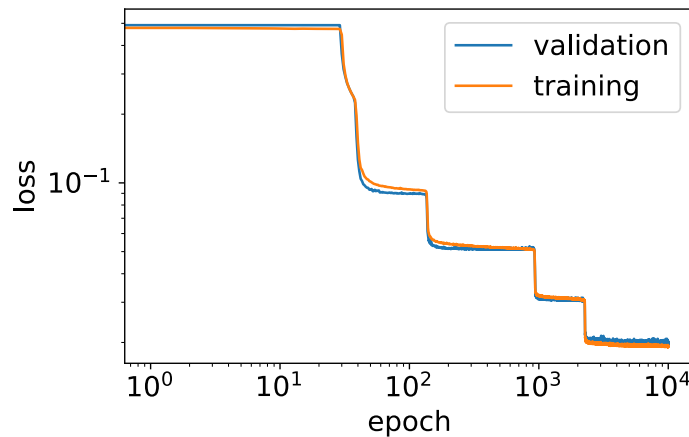Inputs to the model (beam offset and corrector settings)

Outputs from model (beam position along the beam-line)

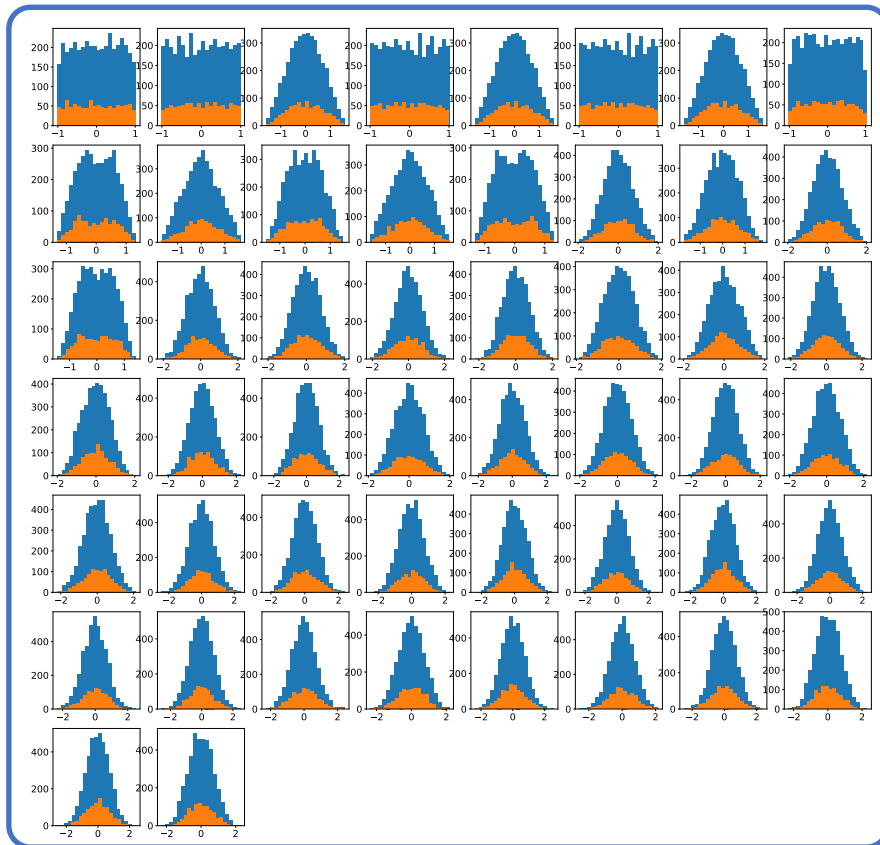# Beam steering in the ATR with Machine Learning

- 14 layers, 15 nodes per layer
- "relu" activation functions
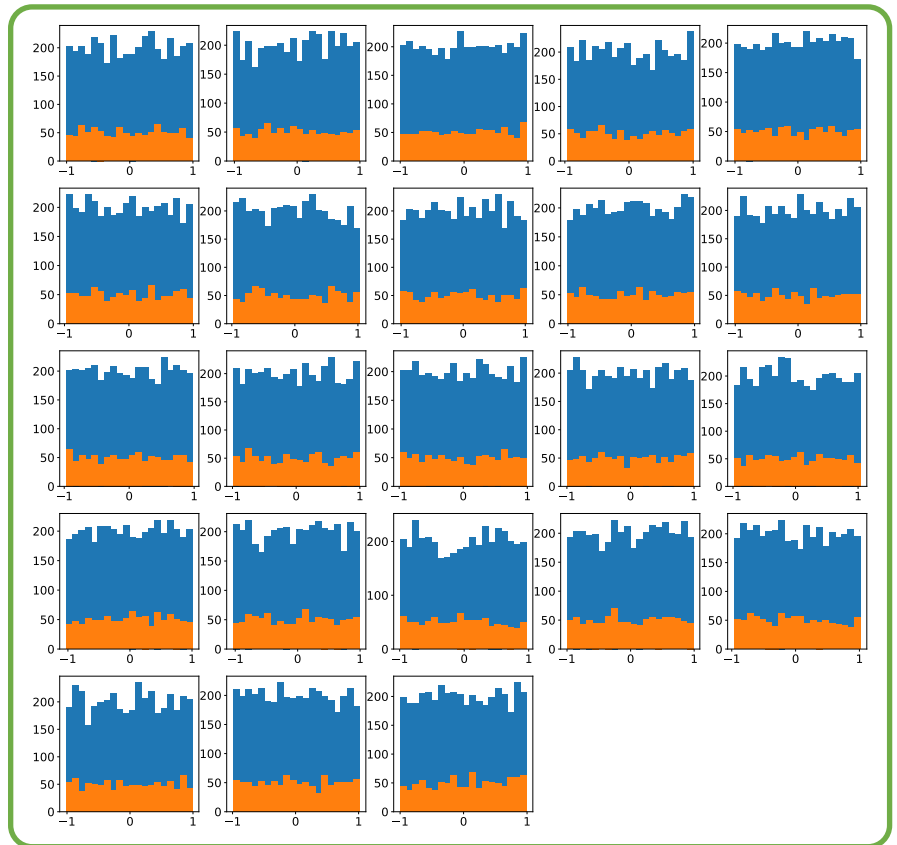- Gaussian noise layers

Beam steering in the ATR with Machine Learning

- Inverse model at a fixed energy: fundamentally more challenging problem

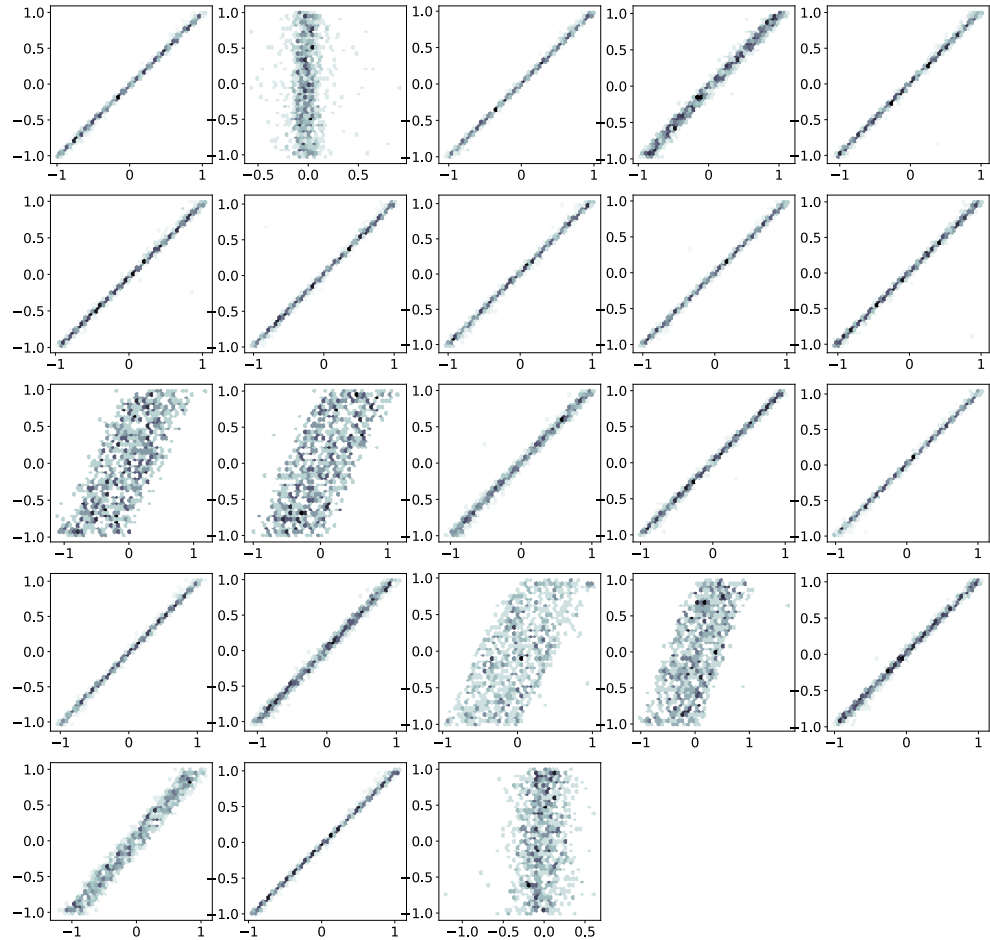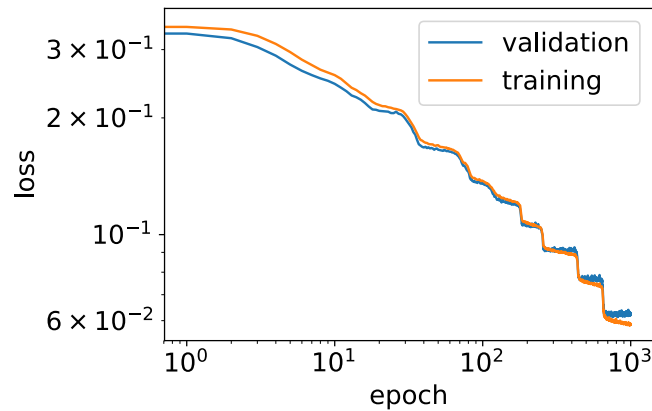Inputs to the model (beam offset and beam-trajectory)

Inputs to the model (corrector settings)

Beam steering in the ATR with Machine Learning

- Inverse model at a fixed energy: fundamentally more challenging problem

Summary and Conclusions

- We want to provide tools for the community that enable wider / easier use of advanced methods for control
  - *Continuing to pursue new collaborations*

- Developing tools to improve the adoption of ML techniques for Accelerators

- Working towards using ML for control problems at BNL and also at Fermilab

radiasoft