Machine Learning for Online Applications and Control

Jonathan Edelen, Nathan Cook, Christopher Hall, Michael Keilman, Paul Moeller, and Stephen Webb

Kevin Brown, Philip Dyer, Auralee Edelen, and Dean Edstrom



radiasoft.net 3380 Mitchell Ln Boulder CO 80301



Fermilab

SLAC NATIONAL ACCELERATOR GARD ABP Workshop 2: Working Group #1 (advanced accelerator instrumentation and controls)

15 April 2020 – Fermilab (Zoom)

Work supported by the US Department of Energy under Award No.'s: DE-SC0019979 & DE-SC0019682



Office of Science

Grand Challenges and How We're Addressing Them

- Grand challenge #1 (beam intensity):
 - Better online models and tuning algorithms will enable accelerators to operate closer to the ideal configurations
- Grand challenge #2 (beam quality):
 - Improved controls will help operational machines realize theoretical limits on beam quality
- Grand challenge #3 (beam control):
 - Better online models and tools integrated with accelerator operations directly address the need for better beam control
- Grand Challenge #4 (beam prediction):
 - Online modeling using as-built simulations augmented with measured data from the machine is a powerful tool for beam-prediction



How do better online tools impact the DOE mission?

- The primary scientific mission of the ABP thrust is to address and resolve the Accelerator and Beam Physics Grand Challenges. Other equally important ABP missions are associated with the overall DOE HEP missions:
 - Advance physics of accelerators and beams to enable future accelerators
 - Uniform tools enable easier collaboration and allow scientists and engineers to more quickly solve challenging control problems
 - Develop conventional and advanced accelerator concepts and tools to disrupt existing costly technology paradigms in coordination with other GARD thrusts
 - The next generation of accelerators requires advanced controls to meet their performance metrics
 - Guide and help to fully exploit science at the GARD beam facilities and operational accelerators
 - Improved controls at facilities maximize beam time for users and enable more cutting edge physics
 - Educate and train future accelerator physicists
 - A library of working ML examples lowers the barrier to entry for scientists solving problems in accelerators



Machine agnostic tools will help the community advance together

- Some online modeling, analysis, and tuning programs in use or under development
 - MATLAB Middle Layer / Accelerator Toolbox (Spear3 + NSLS-II + Others)
 - OCELOT (DESY + SLAC + Others)
 - GPU Accelerated PARMILA (LANL)
 - RHIC Lattice Translator (BNL + Plans for EIC)
 - LUME, Simulacrum (SLAC)
 - PyEpics (in use at APS / JLab / SLAC)
 - LCLSTools (SLAC)
- A proposed path forward: bridge gaps between accelerator facilities by integrating tools into an intuitive interface with a library of worked examples
 - Browser-based GUIs lower the barrier to entry and enable more seamless collaboration
 - Library of successful examples will allow users to quickly build robust custom solutions
 - RadiaSoft plans to test and deploy such tools at BNL, JLab and Fermilab

Aradiasoft

Surrogate modeling enables rapid optimization of high brightness photo injector



Generate ML Model using Sparse Random Sample

A. Edelen, et al., PRAB 23, 044601 (2020)

Aradiasoft

Surrogate modeling enables rapid optimization of high brightness photo injector



Generate ML Model using Sparse Random Sample

A. Edelen, et al., PRAB 23, 044601 (2020)

radiasoft

Verifying Pareto front from the neural network



A. Edelen, et al., PRAB 23, 044601 (2020)



Machine learning techniques enable rapid assessment and correction of machine settings to achieve desirable phase space parameters





Local optimizer alone was unable to converge \rightarrow able to converge after initial settings from neural network

A. Scheinker, A. Edelen, et al., PRL 121, 044801 (2018)

Aradiasoft

A controls toolbox powered by Sirepo



A controls toolbox powered by Sirepo





Prototype web tools for controls











Aradiasoft

KMeans Number of Runs 6

DBSCAN Max Sample Distance 0

2













Data Import + Selection / Algorithm Selection / Cluster Identification / Advanced Visualizations

Curve Fitting

Equation Entry / Plotting / Fit Analysis / Higher Dimensional Fits

Frequency Analysis

1-D FFT / Peak Identification / 2-D FFT / Frequency Spectrograms







Data Import	Data Source						
Simulations / Machine Interface / Direct	Data Source Input/Output Files Elegant Simulation			DTL Tank			
Import	Lattice Errors		~	RF Cavity Settings	^		
	Phase Error [deg] 1	25		Change P0 🕄	es 🛛		
Initial Data	Amplitude Error 0	0.15		Frequency [Hz]	201.24		
Visualization	dx Error [m]	0.25		End1 Focus 🔁	ies		
Plotting Tools / Data	dy Error [m] 🕄	0.25		End2 Focus 🖲	ies		
Cleaning + Reduction /	dxp Error [rad] 🔀	0.25		Phase [deg]	58.0		
Splitting (training, test, validation)	dyp Error [rad]	0.25		Lock Phase 🖲	No		
	dp Error 🚯	1.5e-4					
				Elegant Simulation	^		
Machine Learning	Files		•	Iterations	500		
Selection	Inputs File	inputs.csv -		Running Simulation			
Learning Paradigm / Model Selection +	Outputs File	outputs.csv -			End Simulation		
Construction / Loss Functions / Validation	Inputs Scaler Outputs Scaler Save	Max-Abs Scaler Min-Max Scaler ✓ Robust Scaler Standard Scaler None Changes Cancel					

Aradiasoft

Data Import Simulations / Machine Interface / Direct Import

> Initial Data Visualization

Plotting Tools / Data Cleaning + Reduction / Splitting (training, test, validation)

Machine Learning Selection

Learning Paradigm / Model Selection + Construction / Loss Functions / Validation







Aradiasoft

Data Import	Model
Simulations / Machine	Display
Import	Neural
Initial Data Visualization	Add Lay
Plotting Tools / Data Cleaning + Reduction / Splitting (training, test, validation)	Neura
Machine Learning Selection	
Learning Paradigm / Model Selection +	

Model Type	Neural Network	•	Optimizer	Adam 🔻		
Display Graph			Losses	Mean Squared Error		
			Epochs	500		
Neural Network Layers		^	Batch Size	50		
Add Layer		•	Shuffle Before Epoch	Yes		
			🖾 Data Source 🛛 🗧	Parti Binary Cross-Entropy Categorical Crossentropy	0 -	
Neural Network	RMSProp Adagrad Adadelta	^	Neural Network	Categorical Hinge Cosine Proximity Crossentropy Hinge		•
Optimizer	Adam ✓ Adamax Nesterov Adam Mean Squared Error • 1000 50		Optimize	Kullback Leibler Divergence log(cosh(x)) Mean Absolute Error Mean Absolute Percentage Error ✓ Mean Squared Error Mean Squared Logarithmic Error		
Losses		•	Losse			
Batch Size			Epoch	IS Sparse Categorical Crossentropy Squared Hinge		
Shuffle Before Epoch			Batch Siz	:e 50		
			Shuffle Before Epoc	h Yes		

•

Neural Network



Construction / Loss Functions / Validation ▲

Neural Network Model • \mathbf{A} -• Model Type Neural Network Optimizer Adam • Mean Squared Error **Display Graph** Losses Interface / Direct 500 Epochs Import Neural Network Layers ~ Batch Size 50 Shuffle Before Epoch Add Layer Yes • (None, 9) input: dense_1_input: InputLayer output: (None, 9) Neural Network Layers \mathbf{A} Plotting Tools / Data input: (None, 9) Activation Funciton Dimensionality Activation dense 1: Dense Alpha Noise Dropout output: (None, 10) relu ✓ Densely Connected NN - X Rectified Linear Unit (relu) 10 Splitting (training, test, Dropout Flatten Gaussian Noise Dropout validation) input: (None, 10) Gaussian Noise dropout_1: Dropout 🛏 output: (None, 10 Rate = 0.5Save Changes Cancel (None, 10) input: dense 2: Dense Machine Learning output: (None, 10) softplus Exponential Linear Unit (elu) Neural Network Layers Selection Softmax Scaled Exponential Linear Unit (selu) (None, 10) Softplus input: pha_dropout_1: AlphaDropou Dimensionality Layer Softsign (None, 10) output: Rate = 0.5Learning Paradigm / ✓ Rectified Linear Unit (relu) Densely Connected NN -10 Hyperbolic Tangent Model Selection + Siamoid Add Layer Hard Sigmoid • (None, 10) input: Construction / Loss Exponential (base e) dense_3: Dense output: (None, 10) Linear (identity) hard_sigmoid Functions / Validation Save Changes Cancel (None, 10) input: gaussian_dropout_1: GaussianDropout output: (None, 10) Rate = 0.5



Data Import

Simulations / Machine Interface / Direct Import

> Initial Data Visualization

Plotting Tools / Data Cleaning + Reduction / Splitting (training, test, validation)

Machine Learning Selection

Learning Paradigm / Model Selection + Construction / Loss Functions / Validation







Transfer learning enables portable solutions between accelerators

- Case Study: The Fermilab linac
- Neural networks trained on data from DTL Tanks 2, 3, and 4 for 1k epochs
 - Model from tank 2 is trained on data from tanks 3 and 4 for 1k epochs
 - Transfer learning trains faster and reaches a better overall solution





radiasoft

Inverse models for Beam steering in the ATR

- Optimization (top)
 - 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
 - Direct optimization is time consuming on average took 2k iterations to converge
- Machine Learning (bottom)

radiasoft

- Build inverse model of bpmreadings to corrector settings
- Make feed-forward correction
- Inverse models are fast and effective
 - Single iteration generates a solution almost as effective as optimization



What are some of the challenges for ML and online tools?

- Machine Learning means data: but not just any data
 - Carefully curated, archived, and cleaned
 - Large amounts of data can be required
- Data rates are a potential challenge for browser based tools
- Data archiving
 - Identifying correct parameters and ensuring time alignment or pulse ID
- Machine models
 - Many machines do not have up-to-date as-built models
 - Improving these models reduces demand on data-collection
 - It may not be possible to build models for old machines as survey data or field maps may not exist

Aradiasoft

Roadmap

Year 1 Year 2 Year 3 Year 4 Year 5 121416 18 20 $\mathbf{2}$ 3 58 9 10 11 131517194 6 7Machine agnostic toolbox Browser based tool development Control system connections Automate deployable code generation Milestone 1: Accelerator focused ML toolbox Test deployment at single lab Roll out at two or more labs Milestone 2: Toolbox integration across DOE Algorithm development and deployment Surrogate model libraries Tuning algorithm development Integration with tracking codes Milestone 3: Libraries of working example Surrogate model testing and deployment Tuning algorithm testing and deployment Milestone 4: ML tools integrated with operations



Our aim is to bridge gaps between accelerator facilities by developing machine agnostic tools that integrate machine learning, accelerator simulation codes, and accelerator operations

- Grand challenge #1 (beam intensity):
 - Better online models and tuning algorithms will enable accelerators to operate closer to the ideal configurations
- Grand challenge #2 (beam quality):
 - Improved controls will help operational machines realize theoretical limits on beam quality
- Grand challenge #3 (beam control):
 - Better online models and tools integrated with accelerator operations directly address the need for better beam control
- Grand Challenge #4 (beam prediction):
 - Online modeling using as-built simulations augmented with measured data from the machine is a powerful tool for beam-prediction









Office of Science

Aradiasoft