Adding Machine Learning to the Analysis and Optimization Toolsets at the Light Source BESSY II

Luis Vera Ramírez, Gregor Hartmann, Tom Mertens, Jens Viefhaus, Roland Müller

Helmholtz-Zentrum Berlin (HZB), Germany

17th Biennial International Conference on Accelerator and Large Experimental Physics Control Systems, 08/10/2019, New York

Motivation

At the **synchrotron radiation source BESSY II** (Helmholtz-Zentrum Berlin, HZB) both the **beamline** and the **machine groups** have started working towards setting up the infrastructure to introduce **modern analysis, optimization and automation** in order to improve the performance and the experimental setups.

We present an initial study with Machine Learning tools (started at the end of 2018) including **analysis and prediction models** with real accelerator data as well as first prototypes and use cases for **parameter tuning** with (Deep) Reinforcement Learning agents.

HZB Helmholtz Zentrum Berlin

- The **Helmholtz Association** (the largest scientific organisation in Germany) has initiated the implementation of the **Data Management and Analysis** concept across its centers.
- Further HZB colaborations:
  - Accelerating Machine Learning for physics (AMaLea) (DESY, KIT, HZDR and HZB) - *more about this at talk TUCPL06*
  - Advanced Computational Tools (ACT) (TU Darmstadt, HZB)
  - Findable, Accessible, Inter-operable and Re-usable Data Infrastructure for material research (FAIRmat/FAIRDI) (IRIS Berlin, HU Berlin, Max Planck Institute and HZB)

HELMHOLTZ ASSOCIATION

Figs: Helmholtz-Gemeinschaft Deutscher Forschungszentren - https://commons.wikimedia.org/w/index.php?curid=56032104

Table of Contents

Table of Contents

Overview: Machine Learning tools

- Standard context for measurement prediction → **supervised learning.**
- Further **unsupervised** tools for e.g. anomaly detection.
- **Reinforcement learning** allows (model-free) parameter tuning.

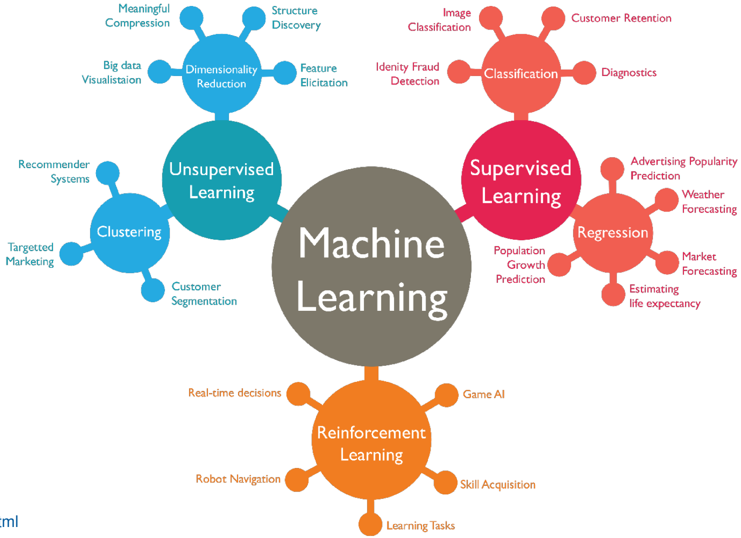


Figure from http://www.isaziconsulting.co.za/machinelearning.html

Overview: Current state at BESSY II

Device

- Injector / Ring accelerator
- Undulator
- Beamline
- Experiment

Data

- Archive, Diagnostic
- Simulations
- Online optimization

- Diagnostic
- Raytracing
- Scans/online data

- Demands
- Simulations
- Beamtimes

Methods

- SVR-RFF
- DNN
- Deep-RL-Control
- RNN, LSTM

- Autoencoder
- CNN, MLP, GBoost
- Dataloader
- Tensor product
- kNN, auto-diff.

- Reasonable random generator

Agent

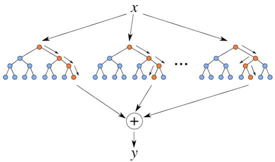
- Operator
- Beamline scientist
- (Random-) User

Table of Contents

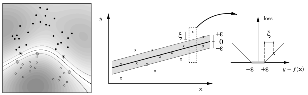
Prediction models

- Goal: **time-series-based prediction** where the input given to the model consists only of simultaneous accelerator readbacks, i.e. **without previous target measurements.** This allows us to:
  - Avoid an excessive reliance on the previous target measurements.
  - Force the model to identify further correlations and patterns in the readbacks.
  - Reuse the information and experience gained in a RL context.

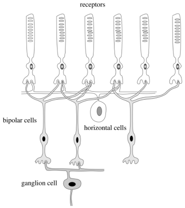
- **Ensemble methods**: Random Forests, Extremely Randomized Trees... ([Bre01], [GEW06]). For regression, **MSE** as loss $\rightarrow$ variance as impurity measure. **Self-explaining**: allows individual analysis of each variable's behavior.



- **Support Vector Regression** [Smo98] with **Random Fourier Features** ([RR08]). SVR extends traditional SVM (for classification) via Vapnik's ϵ-insensitive loss function ([Vap95]).

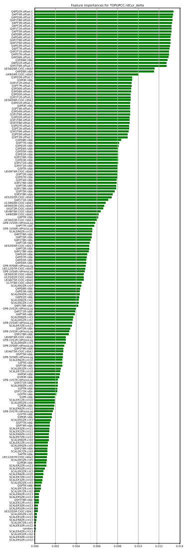


- **Neural Networks** (e.g. see [Roj96]). Feed-forward NNs for regression (i.e. MSE as loss function).



Figs. from https://dsc-spidal.github.io/harp/docs/examples/rf/, [SS03], [Roj96].

Beam lifetime prediction: Experiments



- Target: instant approximation to **beam lifetime** defined via **current decay rate** ($k = 20$): $\frac{1}{\tau} = -\frac{\mathring{I}_t}{I_t} \approx -\frac{1}{I_t} \frac{\sum_{i=0}^{k}(I_{t-i}-I_{t_0})(t-i-t_0)}{\sum_{i=0}^{k}(t-i-t_0)^2}$
- Input variables (185 after preprocessing):
  - Gap and shift of **insertion devices** undulators (21 vars)
  - **Power supply currents** and **offsets** into **quadrupoles** (58 + 38 vars) and **sextupoles** (7 vars)
  - **Collisions** with rest gas particles, **vacuum pressures** (12 vars)
  - **Local beam loss fractions** (49 vars)
- Feature importance analysis with `RandomTrees`: even distribution, but **quadrupoles** (offsets) and **insertion devices** stand out

Test set	Algorithm	RMSE				R^2		
		Avg.	Pers.	Mov. Pers.	Model	Pers.	Mov. Pers.	Model
Random 20%	`ExtraTrees`	0.201	0.099	0.091	0.068	0.757	0.794	0.885
	`SVR-RFF`				0.077			0.852 ± 0.001
	`DNN`				0.069			0.881 ± 0.001
Last 20%	`ExtraTrees`	0.231	0.096	0.079	0.195 ± 0.001	0.829	0.884	0.292 ± 0.006
	`SVR-RFF`				**0.121 ± 0.003**			**0.725 ± 0.015**
	`DNN`				**0.125 ± 0.006**			**0.707 ± 0.027**

Beam lifetime prediction: SVR-RFF and DNN with chronological split

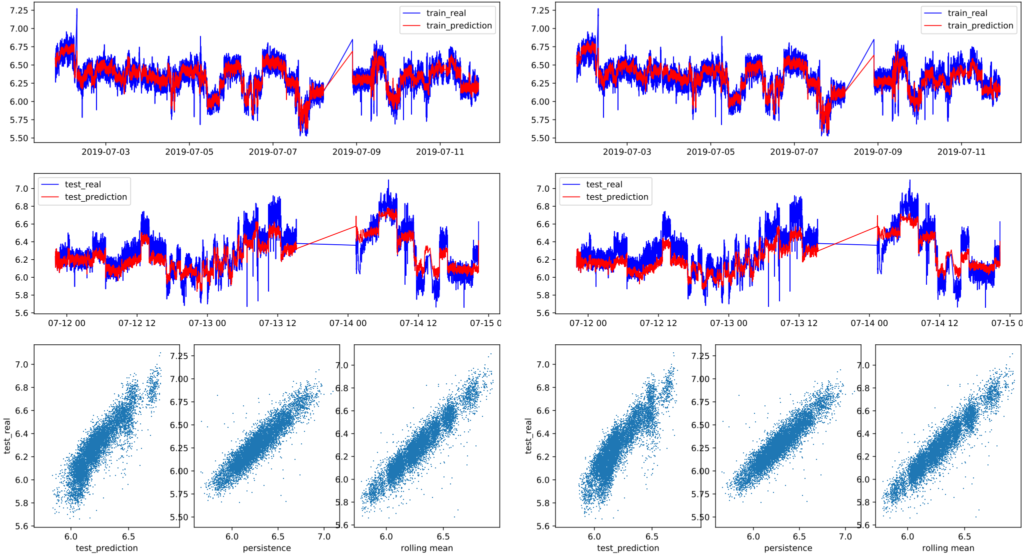


Table of Contents

- **Deep Deterministic Policy Gradient** [LHP+16]: Actor-critic Reinforcement Learning algorithm for continuous environments.
- Off-policy data and the Bellman equation used to learn the Q-function.
- Q-function used to learn the policy.
- *Approximated* with NNs.

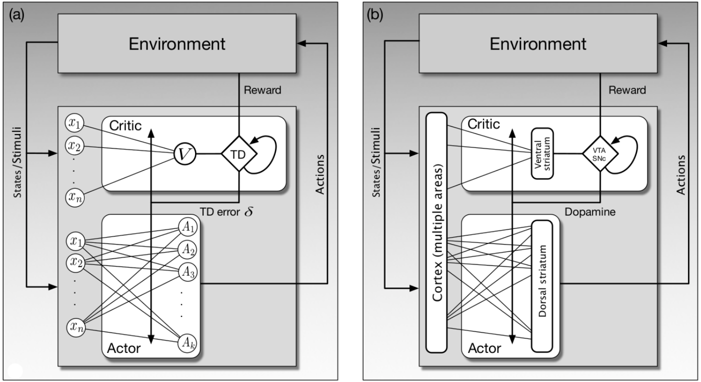


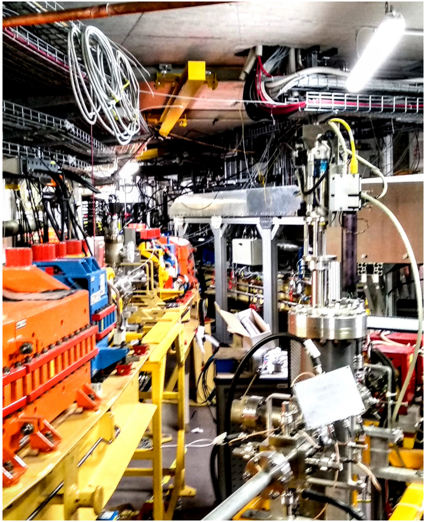
Figure: From [SB18]

Optimization of booster current: Case description

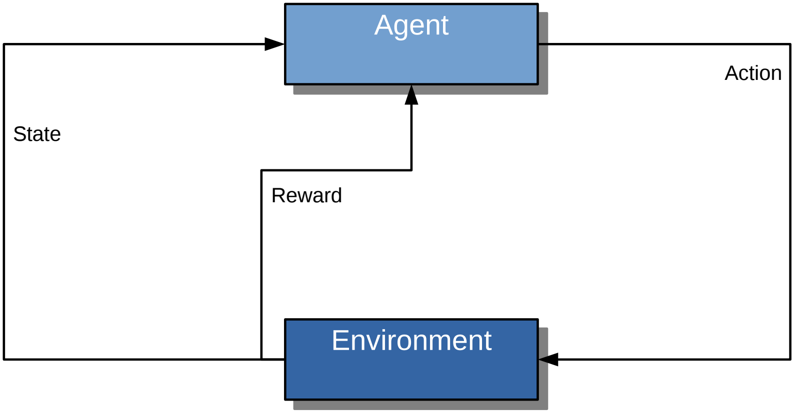
After long interruptions of the machine operation, the booster current tends to be low - as for today, **manual parameter tuning** is required.

We seek an automatized, RL-based solution.

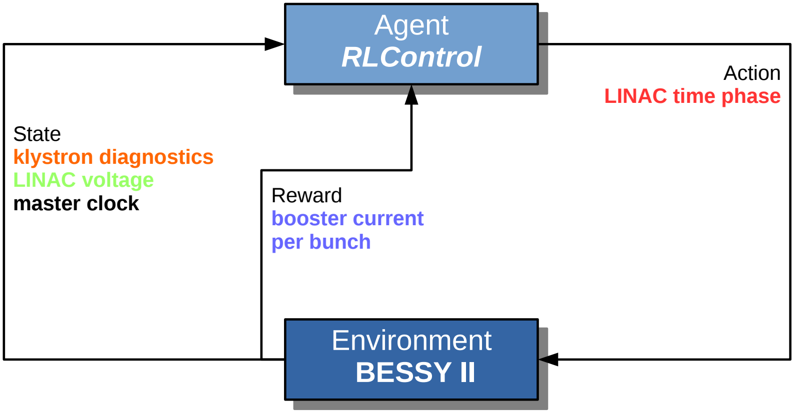
- State variables:
  - High (radio) frequency - master clock.
  - Voltage in LINAC.
  - Klystron current diagnostic measurements.
- Action variable: **time phase in LINAC.** Observations show that this parameter does not affect the injection efficiency.
- Reward: (normalized) **booster current per bunch.**



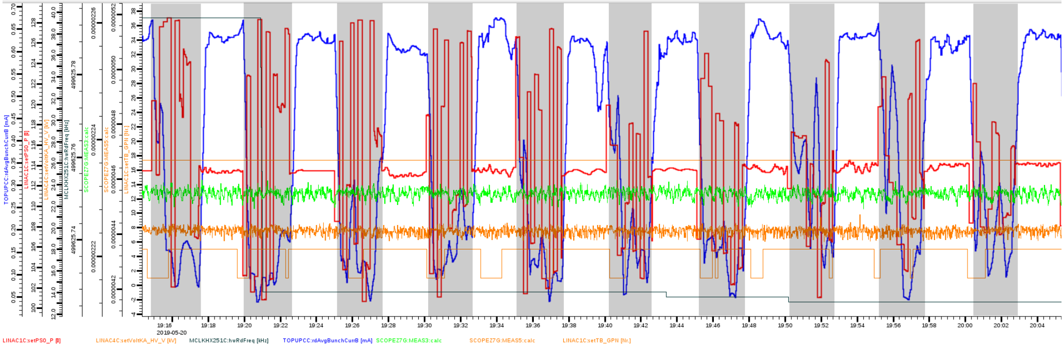
RLControl - Parameter tuning with Deep RL: Case description

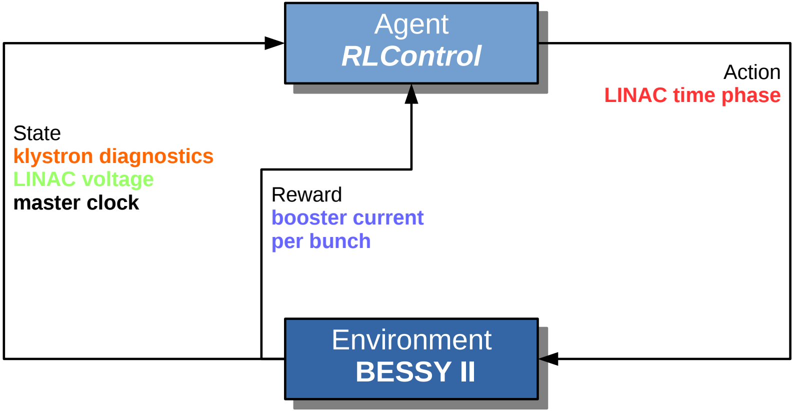


RLControl - Parameter tuning with Deep RL: Case description

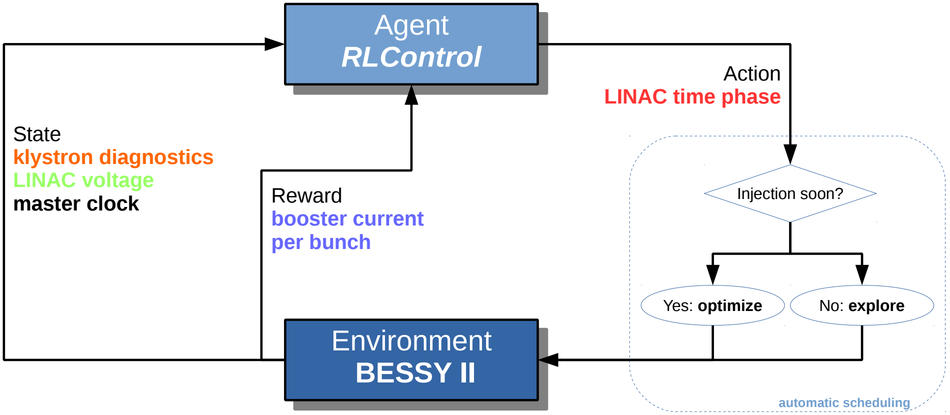


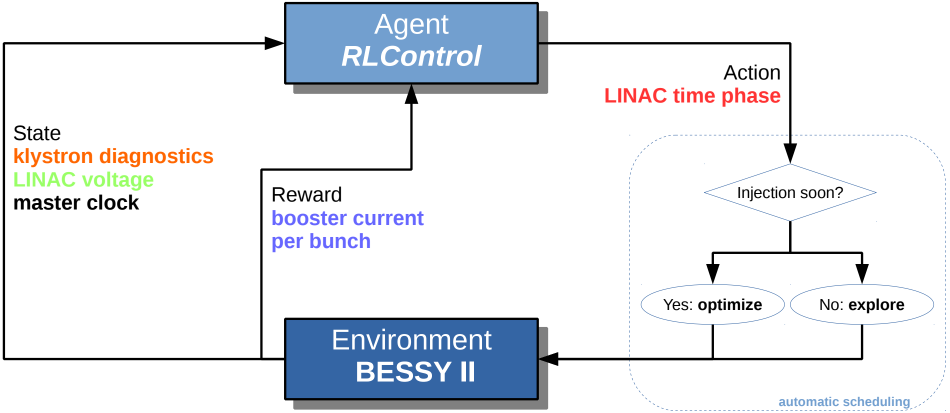
Short test (20/05/19). **Reward (booster current per bunch)** in **blue**, **action (LINAC time phase)** in **red** - remaining lines correspond to state variables. Pretraining with 30 days of historical data. **Exploration** with Parameter Space Noise ([PHD+17]) appears **shaded** - period length of ca. 2 min chosen as experiment for automatic scheduling...




Agent
RLControl
Action
LINAC time phase
State
klystron diagnostics
LINAC voltage
master clock
Reward
booster current
per bunch
Environment
BESSY II

Optimization of booster current: Automatic scheduling





Exploration is scheduled in the meantime between injections to avoid disturbing user activity - optimization activated shortly before each injection.

Optimization of booster current: Test during user operation

*Long test during user time with **automatic exploration schedule** (09/07/19).* Reward (booster current per bunch) in blue, action (LINAC time phase) in red - remaining lines correspond to state variables. Pretraining with 30 days of historical data. Exploration with automatic schedule shaded - **first hour.** The agent optimizes (and learns) successfully during the next **8.5 hours of user operation.**

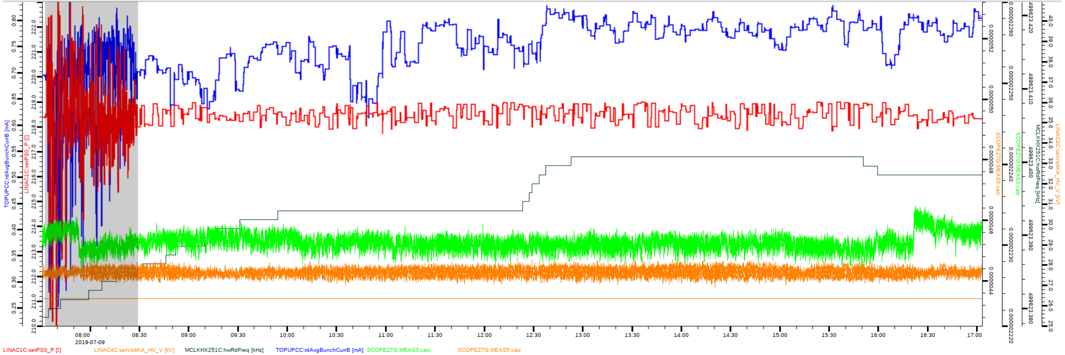
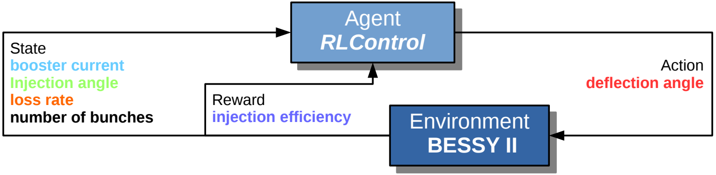


Table of Contents

Optimization of injection efficiency: Case description

Injection efficiency is known to be affected by temperature - nowadays it also needs **manual tuning**. RL-based optimization is a work in progress.



- State variables:
  - Number of **bunches** generated by the LINAC (1, 3 or 5).
  - Injection angle **mismatch**, measured by the beam position in the transfer line.
  - **Current** measured during the booster acceleration phase.
  - Measured **loss rate after extraction** from the booster.
- Action: **deflection angle into the storage ring**, generated by the 2^{nd} septum.
- Reward: **last injection efficiency** - fraction of current increase generated in the storage ring by the charge accelerated in the booster.

Optimization of injection efficiency: Preliminary tests

Short test (23/09/19). Pretraining with 23 days of historical data. **Reward (injection efficiency)** is plotted in **blue**, **actions (septum deflection angle)** is plotted in **red**. Exploration periods appear shaded. *Ad-hoc* modifications in the number of pulses (in black) and booster current (in purple) are carried out during the test - the agent manages to find and improve the optimal action regions.

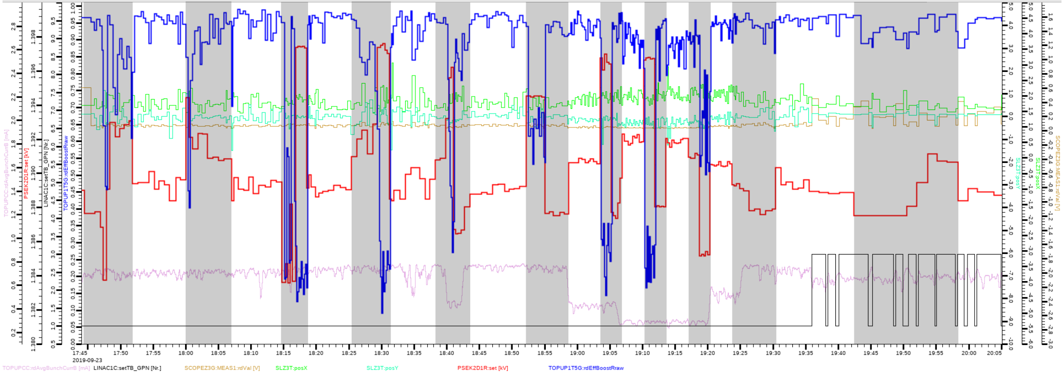
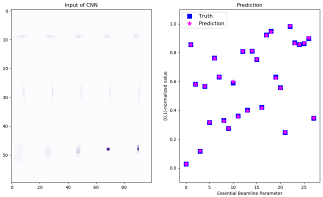


Table of Contents

Building the digital twin

- **The *accelerator/source* side: OCELOT surrogate models for RL** ([AGTZ14]). Challenge: *export* a RL-agent trained on the virtual lattice to the real accelerator.
- **The *beamline/x-ray* side:** general initiative to take advantage of ML methods also for the **retrieval of scientific data** from the measurements. Close coordination of activities developing toolsets for the accelerator and the beamlines (talk MOCPL02 - [M+19]).



E.g.. work by Dr. Gregor Hartmann et al.: **beamline raytracing inversion** (photon footprint screens $\rightarrow$ beamline parameters) with **neural networks.**

Table of Contents

Conclusion

- **Beam lifetime at BESSY II** can be **successfully predicted** in a time-series fashion through supervised learning models trained only with 185 accelerator variables readbacks - i.e. excluding previous beam lifetime measurements.
- The accelerator optimization framework *RLControl* is able to **solve a first use case (booster current) at BESSY II** through Deep Reinforcement Learning. It has also been tested even **during user operation** with the help of injection-based automatic scheduling.
- Further effort has been put into the application of *RLControl* for **more advanced use cases** such as **injection efficiency optimization**, leading also to successful preliminary tests.

Conclusion: Next steps

- Measurement prediction framework:
  - Build models for further target variables such as purity
  - Classification approach
  - Surrogate models
- *RLControl*:
  - Further tests, investigation and use cases (injection efficiency with more state and action variables, orbit correction with OCELOT ([AGTZ14]) pretraining...)
  - Bluesky integration
  - User interfaces

Table of Contents

Bibliography I

[AGTZ14] Ilya Agapov, Gianluca Geloni, Sergey Tomin, and Igor Zagorodnov. OCELOT: A software framework for synchrotron light source and FEL studies. *Nuclear instruments & methods in physics research / A*, 768:151 – 156, 2014. (c) Elsevier B.V.

[BKH16] Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *ArXiv*, abs/1607.06450, 2016.

[Bre01] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.

[GEW06] Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine Learning*, 63(1):3–42, Apr 2006.

[LHP+16] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[LTZ08] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 413–422, Washington, DC, USA, 2008. IEEE Computer Society.

Bibliography II

[LWSG13] Gilles Louppe, Louis Wehenkel, Antonio Sutera, and Pierre Geurts. Understanding variable importances in forests of randomized trees. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'13, pages 431–439, USA, 2013. Curran Associates Inc.

[M+19] Roland Müller et al. Modernization of experimental data taking at BESSY II. ICALEPCS, 2019.

[PHD+17] Matthias Plappert, Rein Houthooft, Prafulla Dhariwal, Szymon Sidor, Richard Y. Chen, Xi Chen, Tamim Asfour, Pieter Abbeel, and Marcin Andrychowicz. Parameter space noise for exploration. *CoRR*, abs/1706.01905, 2017.

[Roj96] Raul Rojas. *Neural networks : a systematic introduction*. Springer-Verlag, Berlin ;;New York, 1996.

[RR08] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1177–1184. Curran Associates, Inc., 2008.

[SB18] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

Bibliography III

[SLH+14] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 387–395, Bejing, China, 22–24 Jun 2014. PMLR.

[Smo98] Alexander Johannes Smola. Learning with kernels, 1998.

[SS03] Alex J. Smola and Bernhard Schölkopf. A tutorial on support vector regression. Technical report, STATISTICS AND COMPUTING, 2003.

[Vap95] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, Berlin, Heidelberg, 1995.

[ZM18] Xiaoqin Zhang and Huimin Ma. Pretraining deep actor-critic reinforcement learning algorithms with expert demonstrations. *CoRR*, abs/1801.10459, 2018.