

Application of machine learning techniques at the CERN Large Hadron Collider

EPS-HEP 2019, Ghent

Friday July 12, 2019

```
F.F. Van der Veken<sup>1,2</sup>,
G. Azzopardi<sup>1,2</sup>, F.H. Blanc<sup>4</sup>, L.T.D. Coyle<sup>1,4</sup>, E. Fol<sup>1,3</sup>, M. Giovannozzi<sup>1</sup>, T. Pieloni<sup>1,4</sup>, S. Redaelli<sup>1</sup>,
L. Rivkin<sup>4,5</sup> B.M. Salvachua Ferrando<sup>1</sup>, M. Schenk<sup>1,4</sup>, R. Tomas Garcia<sup>1</sup>, G. Valentino<sup>1,2</sup>,
```

¹ CERN, ² University of Malta, ³ Johann-Wolfgang-Goethe University, ⁴ Ecole Polytechnique Federale Lausanne, ⁵ Paul Scherrer Institut



Outline

1 Introduction

- 2 Collimator alignment
- Optics measurements and corrections
- Oynamic aperture studies
- Beam lifetime optimisation
- Conclusions and Outlook



EPS-HEP 2019

ML @ LHC

Introduction

Machine Learning

- ML has been used extensively in several domains
- Very recently also in accelerator physics
- Also for the LHC at CERN
- In this talk: 4 collaborations working on different topics



Introduction

Machine Learning at the LHC

- Concerning machine operation:
 - Collimator alignment
 - Recognition of faulty monitors
 - Correction of beam optics variables
- Concerning analysis of measurements and simulations:
 - Anomaly detection in tracking simulations
 - Extrapolation of tracking simulations
 - Modelling beam lifetime by operational settings



Outline

Introduction

- 2 Collimator alignment
- Optics measurements and corrections
- Oynamic aperture studies
- Beam lifetime optimisation
- Conclusions and Outlook



EPS-HEP 2019

ML @ LHC



Context

LHC Protection System

- The LHC uses a system of 100 collimators for protection
- These must be **aligned** around the two beams with a precision better than $50\mu m$
- Alignments are performed yearly before start of operation

Applications with Machine Learning

- Alignment is tedious, time-consuming, and repetitive
- Ideal situation for machine learning
 - \Rightarrow supervised learning





Setup



Spikes

- data sample taken when collimator stops moving
- spike when threshold in BLM is passed
- goal is to distinguish real spikes (beam is hit) from noise





Spike Classification

Approach

- Six ML models for spike classification were compared Logistic Regression, Neural Network, SVM, Decision Tree, Random Forest, Gradient Boost
- data (8706 samples) split into: 85% training, 15% testing
- enforce: no false positives
- false negatives are OK (because alignment will continue)
- no retraining needed unless hardware changes
- Analysis of beam crosstalk allows parallel alignments





Results







Results

Results

- ML can replace human operators for alignments
- More than three times faster!
- ML-based alignment will be default from now on



Outline

Introduction

- 2 Collimator alignment
- Optics measurements and corrections
- Oynamic aperture studies
- Beam lifetime optimisation
- 6 Conclusions and Outlook



EPS-HEP 2019

ML @ LHC





Analysis of Beam Optics

- Beam Position Monitors (BPMs) measure excited beam
- Faulty BPMs give unphysical values for optics functions

Applications with Machine Learning

- Identify and remove faulty BPMs from data
 - \Rightarrow anomaly detection by *unsupervised learning*
- Correct β-beating
 - \Rightarrow supervised learning





Approach

- $\bullet\,$ Past measurements show that $\sim 10\%$ of BPMs are faulty
- Non-physical spikes in optics are artefact of bad BPMs
 - \Rightarrow Use ML to identify faulty BPMs from harmonic analysis
 - \Rightarrow To avoid spikes in optics functions
- enforce: no false negatives (don't keep a bad BPM)
- false positives are OK (we have >1000 BPMs...)
- Four ML algorithms are compared:

K-means, DBSCAN, Local Outlier Factor, Isolation Forest











Results

- Very good recognition of faulty BPMs
- Now integrated by **default** into optics measurements at LHC
- Successfully used during commissioning and machine developments





Setup

- β-function calculated from harmonic analysis of BPMs
- β -beating is ratio of measured over designed β -function
- Corrections in the LHC are based on response matrix



Machine Learning (work in progress)

ML to replace response matrix \Rightarrow supervised learning











Approach

• Three ML algorithms are compared:

Convolutional Neural Network, Linear Regression, Ridge

- CNN (Keras with TensorFlow backend):
 - Used for image processing
 - Spatially dependent features: phase advance between neighbouring BPMs
 - Different deep layers look for different features
- Very simple model is applied: no parameter tuning, no optimisation
 - \Rightarrow Lots of improvements are possible









EPS-HEP 2019

11/21



Results

β -beating %	peak	rms
Uncorrected	32±10	11±3
Response Matrix	11±5	3±2
CNN	11±2	3.2±0.5
Ridge regression	10±2	2.9±0.8
Linear regression	9±2	2.6±1.7

- All methods demonstrate similar performance
- Linear Regression ML achieves best correction



Outline

Introduction

- 2 Collimator alignment
- Optics measurements and corrections
- Oynamic aperture studies
- Beam lifetime optimisation
- Conclusions and Outlook



EPS-HEP 2019

ML @ LHC



Context

Dynamic Aperture

- A tool to estimate beam quality
- It is the volume of the smallest connected region in phase space that remains **stable** for a certain amount of time
- Its evolution over time can be estimated with scaling laws
- DA can describe beam losses and luminosity evolution

Applications with Machine Learning

- Anomaly detection \Rightarrow unsupervised learning
- DA extrapolation \Rightarrow supervised learning





Setup

- 60 random realisations ('seeds') in LHC simulations
- Sometimes one seed gives very bad DA for one angle (because close to resonance, internal cancellations, ...)

Machine Learning

- Use ML to flag these outliers
 - ightarrow let human decide whether or not to remove
- Investigate anomaly dependence on angles or seeds





Approach

- Points are sometimes clustered in several groups
 - \Rightarrow **DBSCAN** to recognise clusters

(scaled over population, min 3 points in a cluster) points not in cluster are possible outliers

- \Rightarrow LOF to quantify outlier strength
- ⇒ Cut off at minimum threshold, and outliers can only exist as minima or maxima (not in between)





Results







Results







Results DBSCAN DBSCAN Frequency of anomaly Frequency of anomaly ò Seed number Angle











Results

- Outlier detection per angle works as expected But human verification is indeed needed!
 - → to decide whether or not to remove a particular seed (depending on behaviour of nearby angles)
- $\approx 10\times$ more outliers at large angles and seeds 1 and 52 \Rightarrow further investigation needed





Setup

- DA simulations are very CPU-intensive
 - \Rightarrow only $10^5 10^6$ turns (~ 1 minute) are achievable
- Realistic timescales are much larger ($\sim\!10$ hours)
 - \Rightarrow simulations need to be **extrapolated**
- Scaling laws exist to describe evolution over time

Machine Learning (work in progress)

- Use ML to improve fitting to scaling laws
- Recurrent Neural Network to make prediction estimates (Well-suited to predicted sequential / time-series data)





Approach

- Existing scaling laws work well to describe the data
- But not that much to **predict** (sensitivity of fit parameters)







Trying with a Neural Network

- Brute-force approach: not including any info from scaling
- Time series analysis (LSTM with Keras)







Trying with a Neural Network

- Results aren't very impressive; deeper investigation is needed
- Alternative: use a Neural Network to find optimal weights to fit to existing scaling laws



Outline

Introduction

- 2 Collimator alignment
- Optics measurements and corrections
- Oynamic aperture studies
- 6 Beam lifetime optimisation
- 6 Conclusions and Outlook



EPS-HEP 2019

ML @ LHC

Context



Beam Lifetime

- Is the time τ such that intensity $I(\tau) = \frac{1}{e}I_0$
- Real-life counterpart of DA, describing beam quality
- Strongly influenced by operational settings
- Extraction from simulation is difficult (coherent instabilities)

Applications with Machine Learning

- Avoid time- and CPU-consuming tracking simulations
- Model that directly relates lifetime to machine settings
- Ample data available, focus on 2017 and 2018
 - \Rightarrow supervised learning



Beam Lifetime Model



Approach

- Input:
 - tunes (H/V, B1/B2)
 - sextupole strengths (B1/B2)
 octupole strength (B1/B2)
 - elapsed time

- emittances (H/V, B1/B2)
- timestamps
- number of bunches (B1/B2)
- Output:
 - beam lifetimes (B1/B2, from slope of BCTs)
- Data from Run 2



Beam Lifetime Model



Result: Prediction of Lifetime (with LightGBM algorithm)















Multi-Parameter Optimisation



Optimal Settings

- Close to resonances: highest lifetime
- However this also gives emittance blow-up
- Latter is unwanted as it decreases luminosity

⇒ Multi-objective optimisation problem



Multi-Parameter Optimisation



Optimal Settings





Multi-Parameter Optimisation



Optimal Settings





Outline

Introduction

- 2 Collimator alignment
- Optics measurements and corrections
- Oynamic aperture studies
- 5 Beam lifetime optimisation





Conclusions

• Collimator Alignment:

ML is now the standard tool for collimator alignments

Optics Measurements and Correction:

- ML is now the standard tool to find faulty BPMs
- First steps are made to use ML as an alternative for the response matrix

Dynamic Aperture:

- Anomaly detection is very efficient
- Beam Lifetime:
 - First steps are made towards a model that predicts lifetime in function of the operational parameters



Outlook

- Collimator Alignment:
 - $\bullet\,$ Advanced crosstalk analysis \rightarrow more alignments in parallel

• Optics Correction:

- $\bullet \ \text{Larger dataset} \to \text{more general model} \\$
- Add more sources of errors and non-linearities
- Reinforcement Learning

• Dynamic Aperture:

- Anomaly detection by centralised supervised learning
- Improve prediction algorithms using high-precision data
- Use supervised learning on fitting weights
- Beam Lifetime:
 - Larger dataset and more operational parameters
 - ightarrow more general model



Backup Slides Collimator Alignments



EPS-HEP 2019

ML @ LHC

L-Università ta' Malta

Setup





Backup Slides Beam Lifetimes



EPS-HEP 2019

ML @ LHC

Internal Correlations



Correlations



Lifetimes depend on tunes from **both** beams!

 \Rightarrow Need to de-correlate before continuing



Internal Correlations



Correlations





Robustness of Model



Machine Development

- Use dedicated MD run:
 - to decorrelate tunes between two beams
 - to extend tune range further than only current operational settings
- This allows us to test robustness of model:
 - o does the tunes correlation matter?
 - behaviour of other beam parameters when lifetime is large?



Robustness of Model



Machine Development



- random walk over tunes
- different random walk for beam 2 at the same moment
- do this for different operational settings



Robustness of Model



Machine Development







www.cern.ch