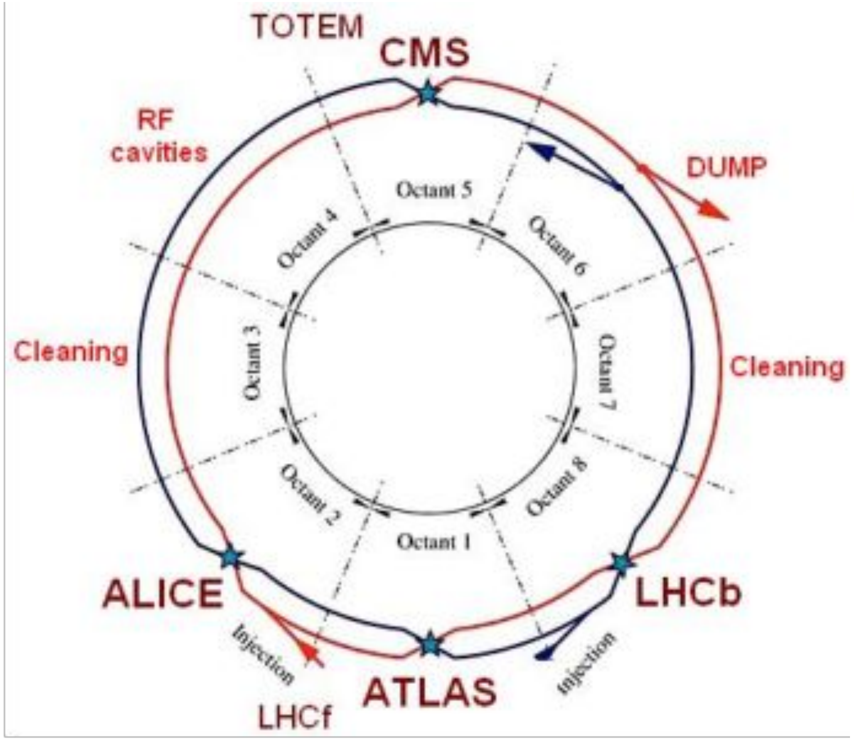
Machine Learning Optimizing beam lifetimes in the LHC

Speaker: Loïc COYLE
T. Pieloni (EPFL-LPAP) and B. Salvachua (LHC-Operation)
Acknowledgement : LHC OP, G. Azzopardi, M. Schenk, T. Tydecks

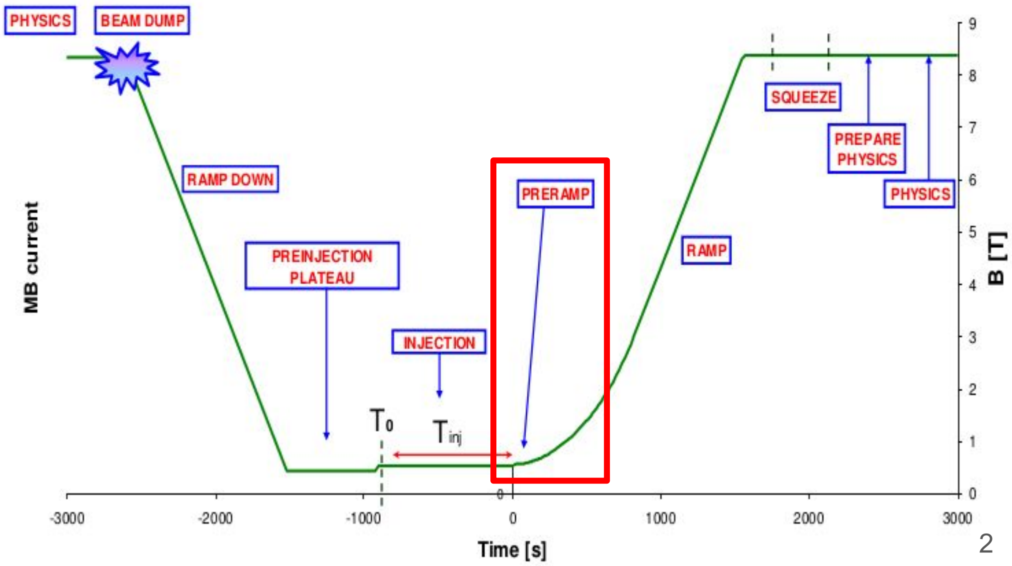
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Context



Acceleration 450 GeV → 6.5 TeV

Beam mode : PRERAMP



Motivation

Unexplained losses at different stages of the cycle

Help to shed light on the impact of parameters on the proton losses

Help optimize and determine operational setups

Goal:

First trial of using machine learning surrogate models for LHC optimization

Try to predict from initial setting the beam lifetimes over a physical cycle to perhaps feedback online corrections to optimize it.

For use in collider optimization → maximising integrated luminosity reach and reducing proton losses

Improve the understanding of proton losses to feed to numerical models for future projects studies

Framing the problem

Reinforcement learning not really feasible

Simulation:

- Particle tracking simulations extremely expensive
- Simulations have trouble with modelling coherent instabilities

Online optimization (BO) not feasible either

→ Data driven supervised learning surrogate model

Surrogate model of the beams' intensity lifetime

Dataset:

- Tunes H/V/B1/B2
- Chromaticity H/V/B1/B2
- Octupole magnet current B1/B2

The operational knobs of the machine

Target: Lifetimes B1/B2

Time span:

- 01-01 to 12-31 2017
- 01-01 to 11-01 2018

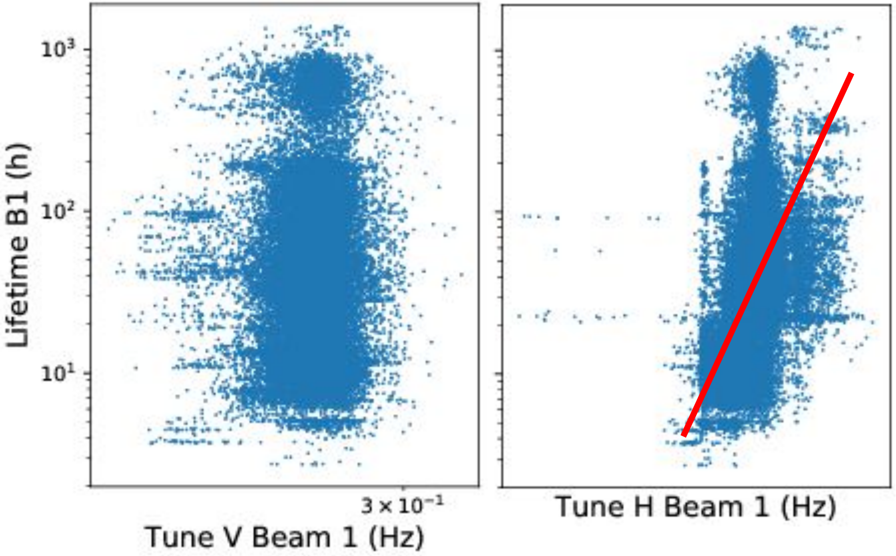
Dataset size: ~50000 points

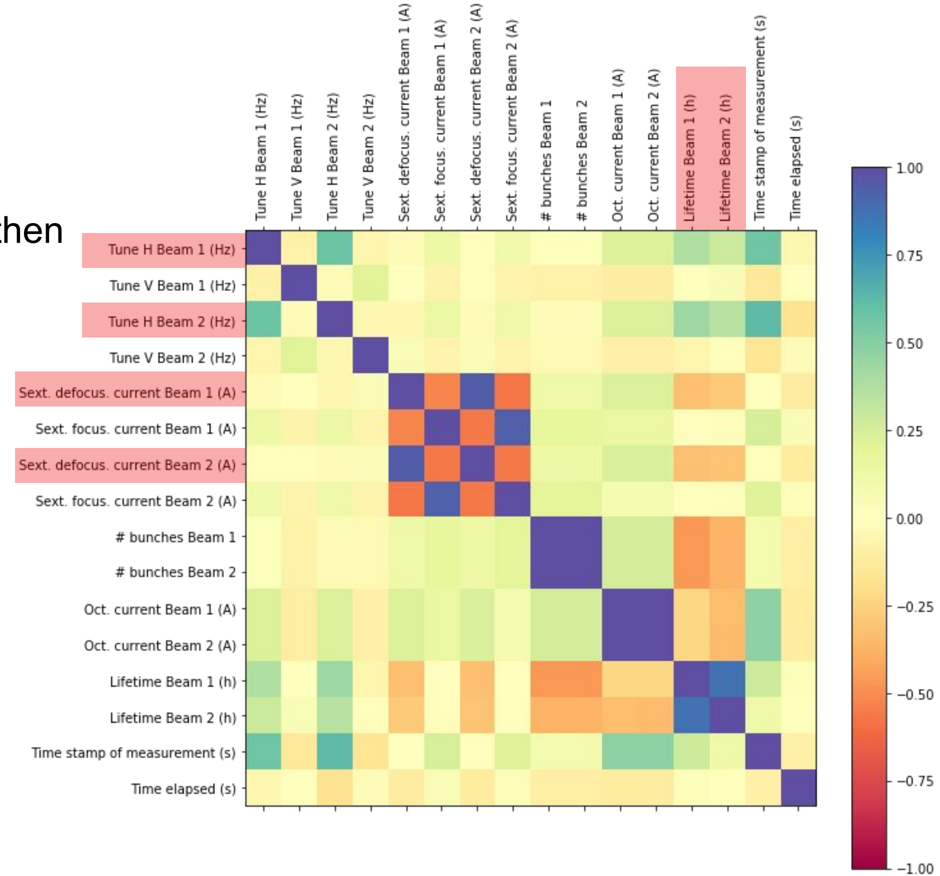
Data Visualized

Spearman correlation coefficient:
Measure of 'fitting' with monotonous function.

- H tunes of B1 & B2
- Sextupole currents

Lifetime B1 as correlated with B2 measurements then with B1 ? Cross beam dependence.

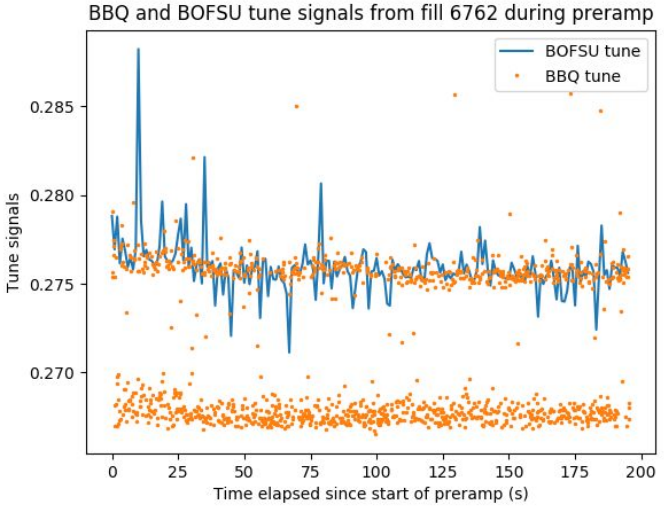


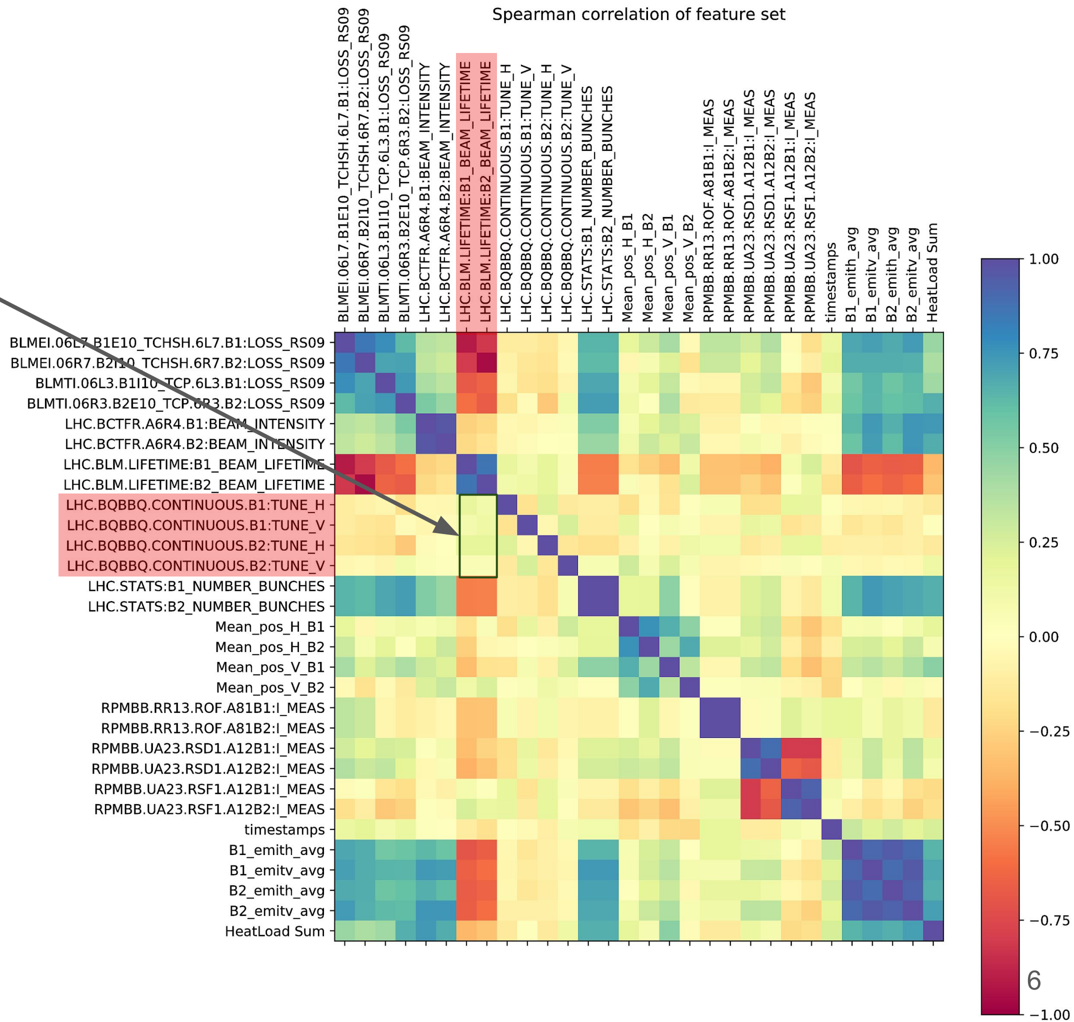


Some surprises...

Unexpected loss of correlations with tunes

Systematic under evaluation of the tune.
Preprocessing of measured data fundamental





Choosing the model

Model	Beam 1	
	R-2	MSE
Linear	0.666	9482.435
Ridge	0.664	9521.182
Lasso	0.655	9773.588
ElasticNet	0.141	24376.165
Multi-Layer Perceptron	0.833	4856.423
RandomForest	0.996	101.971
GB Decision Trees	**0.997**	**76.577**

Linear models :
Determine the ω coefficients which minimise :

$$\min_{w} ||X\omega - y||_2^2.$$

With varying degrees of regularization

http://scikit-learn.org/stable/modules/linear_model.html

Fully connected **neural network**
http://scikit-learn.org/stable/modules/neural_networks_supervised.html

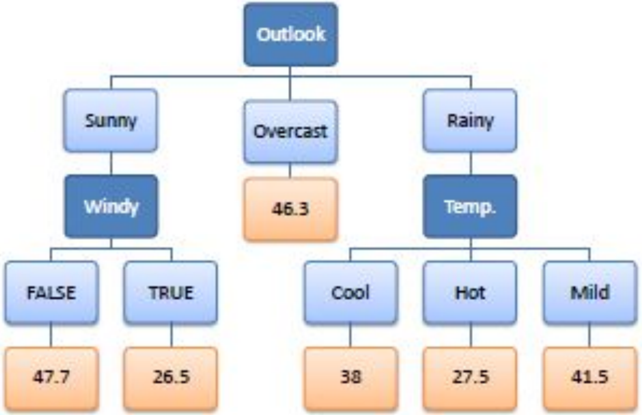
---

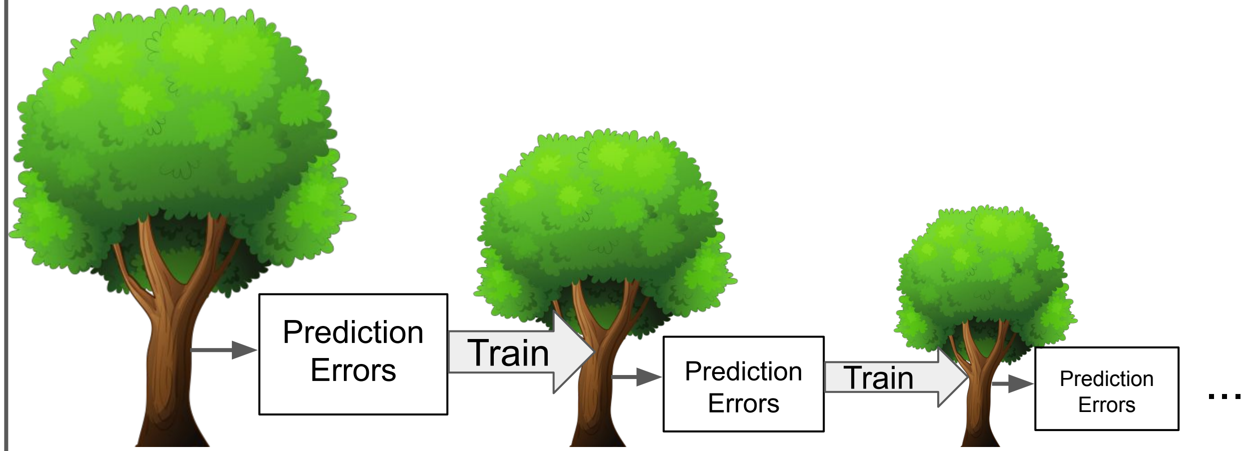
Decision Tree based methods

GBDT from https://github.com/Microsoft/LightGBM

Gradient Boosted Decision Trees

outlook	temp.	humidity	windy	time
rainy	hot	high	false	25
rainy	hot	high	true	30
overcast	hot	high	false	46
sunny	mild	high	false	45
sunny	cool	normal	false	52
sunny	cool	normal	true	23
overcast	cool	normal	true	43
rainy	mild	high	false	35
rainy	cool	normal	false	38
sunny	mild	normal	false	46
rainy	mild	normal	true	48
overcast	mild	high	true	52
overcast	hot	normal	false	44
sunny	mild	high	true	30

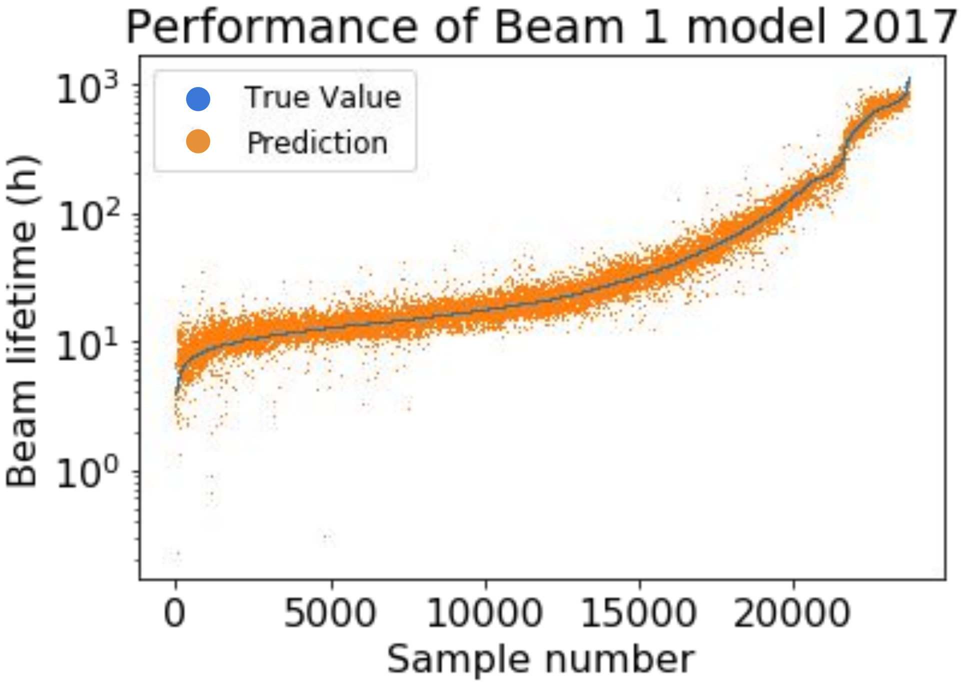




Available implementations:
https://github.com/Microsoft/LightGBM
https://github.com/dmlc/xgboost

Consistently in the top ~% of kaggle competitions

Model Performance



Performance of trained model on **unseen data**.

Model can **accurately predict the value of lifetime**.

Presence of a few outliers

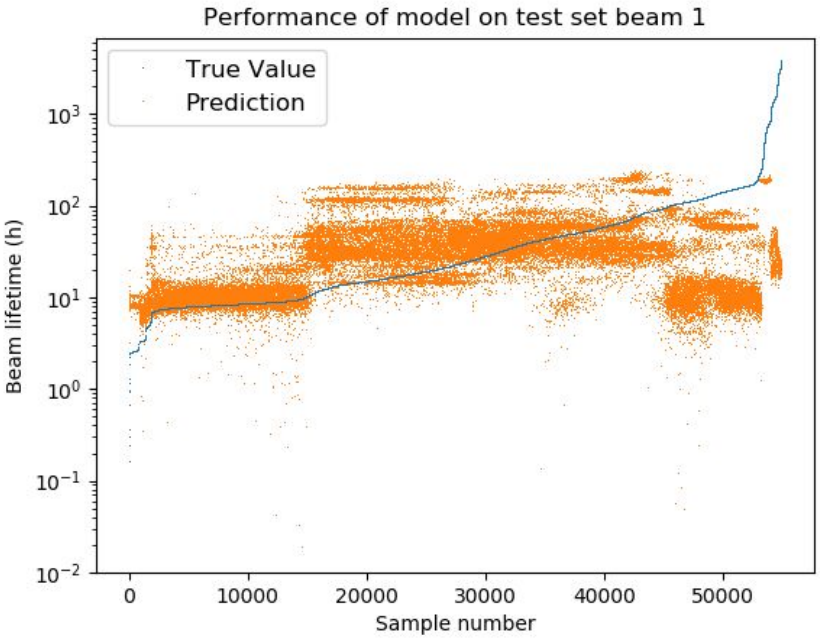
Same for beam 2

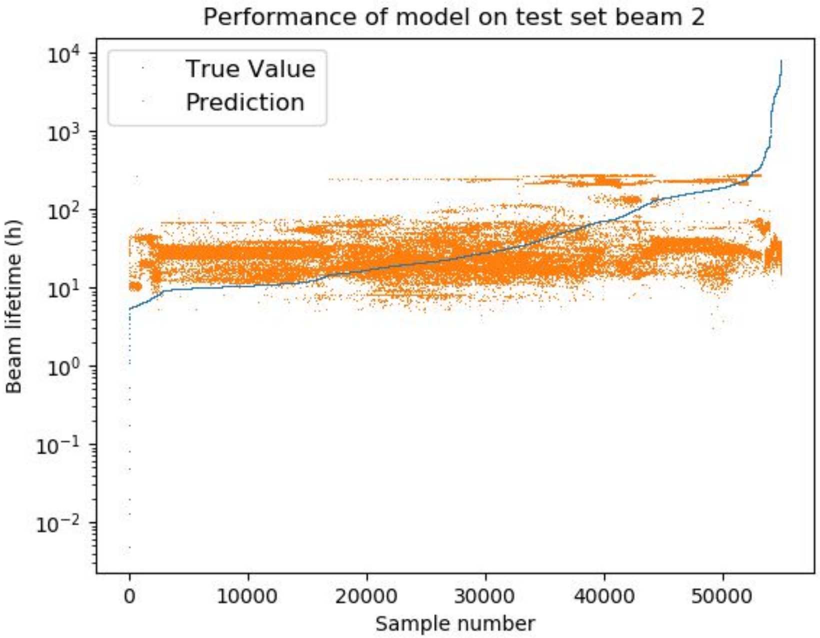
Robustness of model ?

2017 model on 2018 data

Current trained sample with 2017 data cannot fully predict 2018 lifetimes:

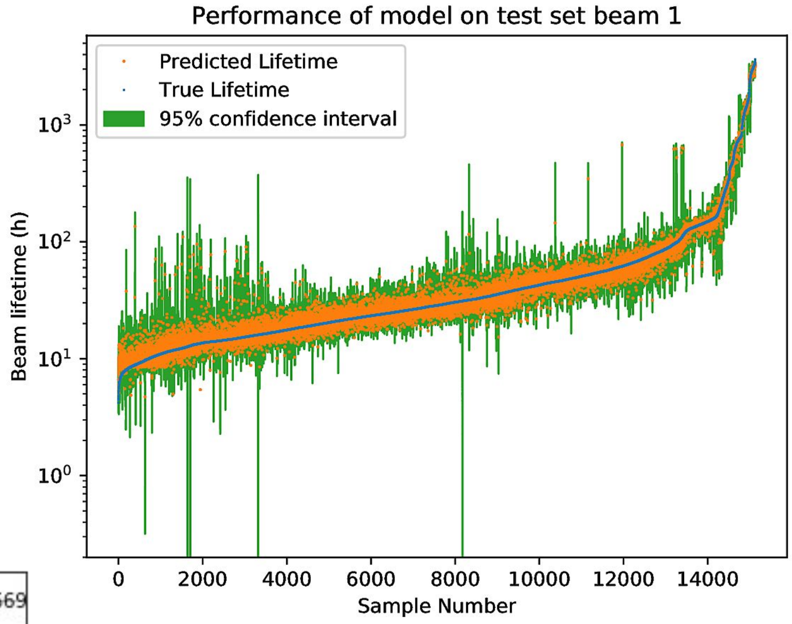
- Missing parameters that will describe better the data
- Retrain every year ? for how long ? dedicated data ?

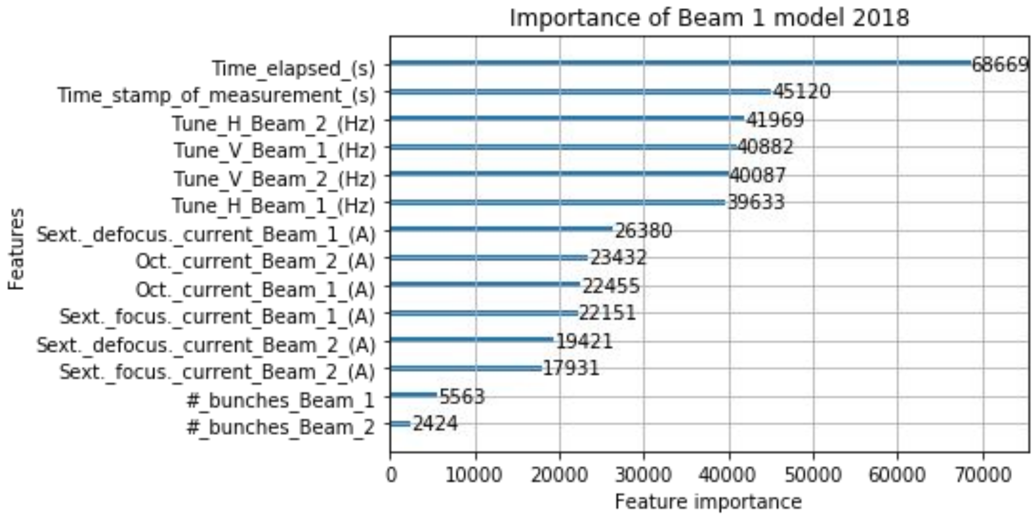




2018 model

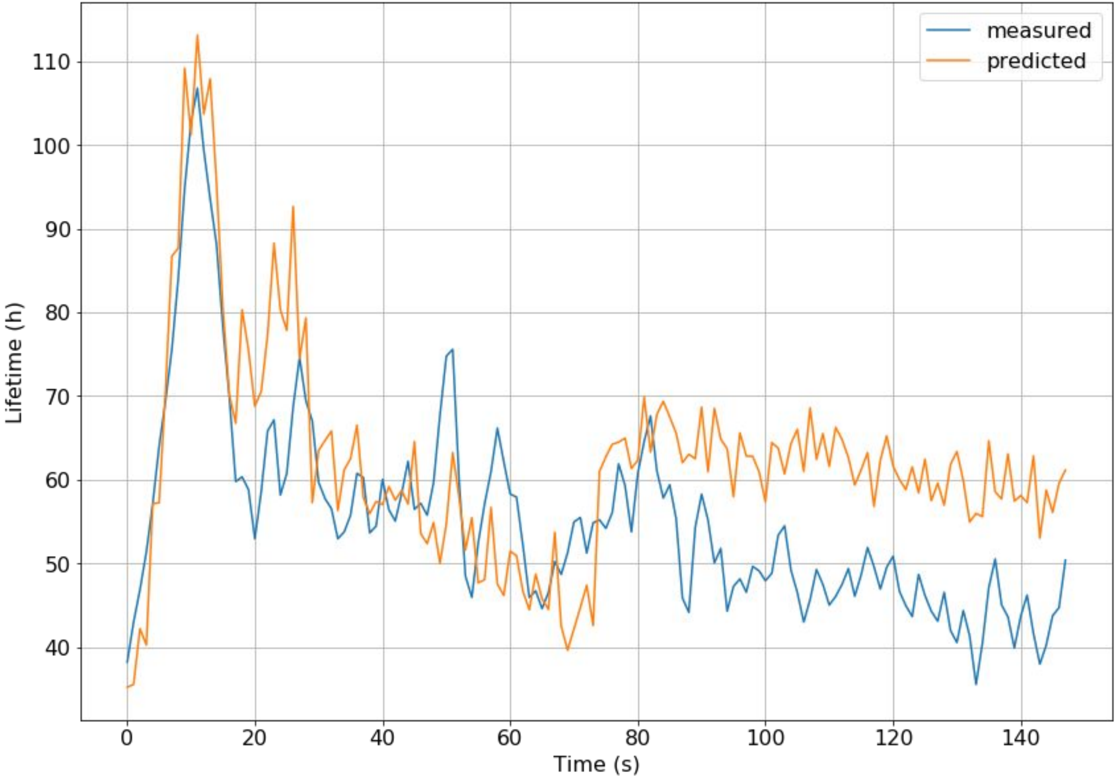
Same model trained on 2018 data.
Using bootstrapping to get an approximation of the 95% confidence interval.
→ trained 100s of models on slightly different datasets → percentile of each prediction points





Feature importance: number of times the splits in the decision trees occur on each feature
Correlation with tunes from both beams ?
Strong dependence on time ?

Proof of concept : surrogate model for optimization

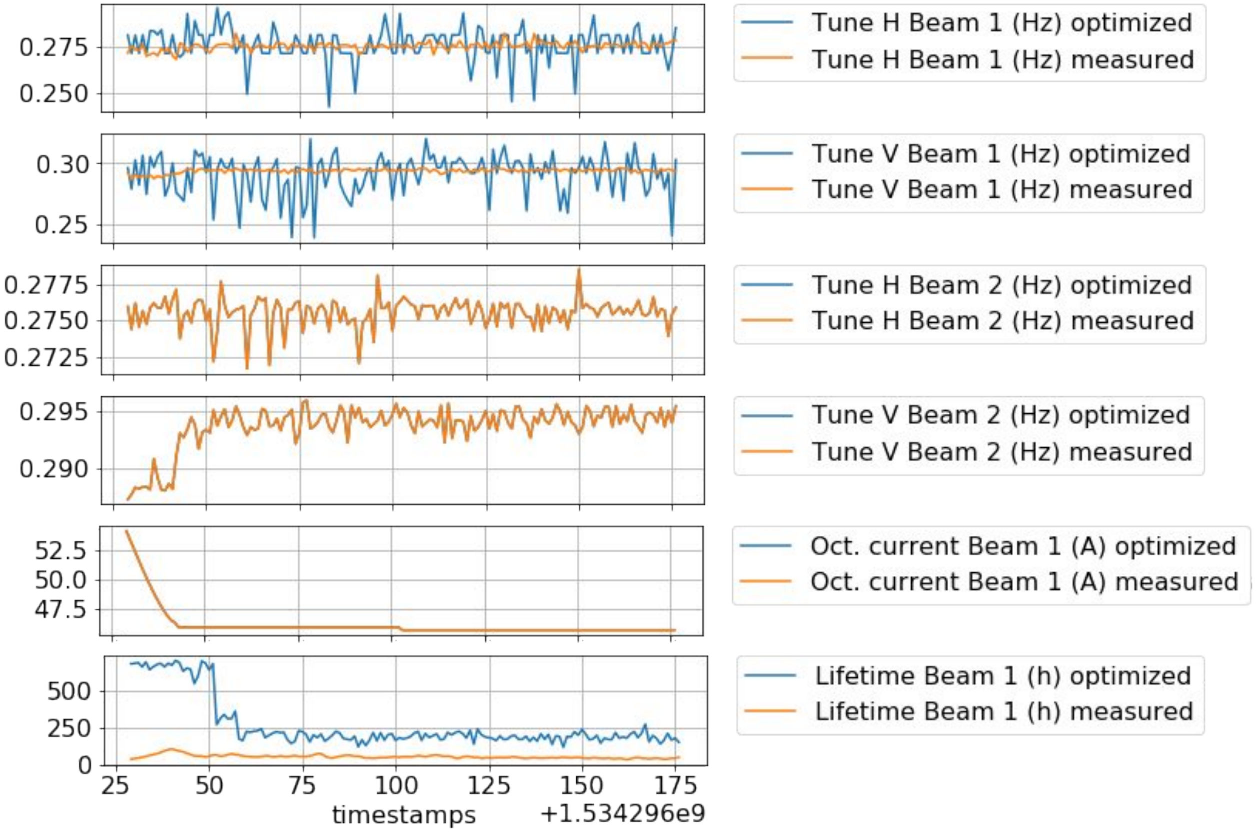


Example fill: 7056

Predicted and measured lifetime B1 evolution.

Prediction is slightly off towards the end but correct trends.

Proof of concept : surrogate model for optimization



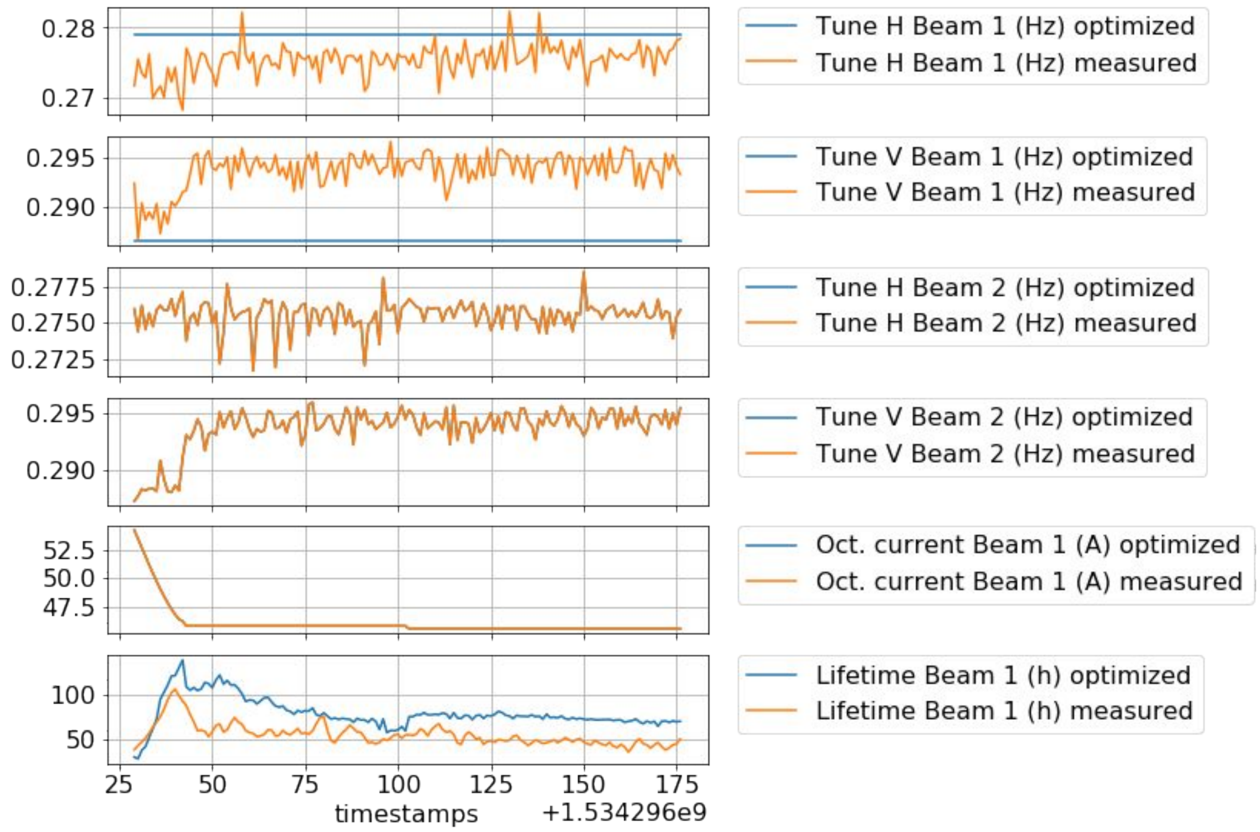
Optimization beam 1 tunes, prediction of beam 1 lifetimes.

Very erratic evolution of parameters. Not feasible.

Large lifetime gain early on tapers off during the fill

To be taken with a grain of salt

Proof of concept : surrogate model for optimization



Optimization beam 1 tunes, prediction of beam 1 lifetimes.

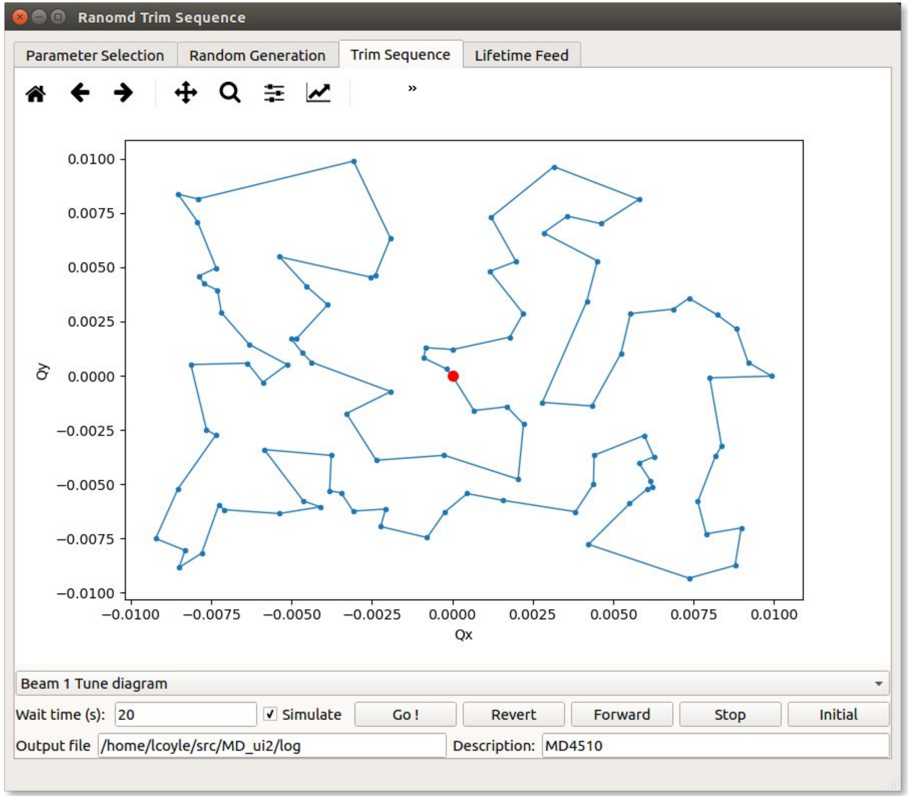
Find the best beam 1 tune trim i.e. constant value.

More realistic optimized lifetimes.

Recommended settings:

- $Q_h = 0.279$
- $Q_V = 0.286$

Dedicated experiment



Generation and control of trim sequences :

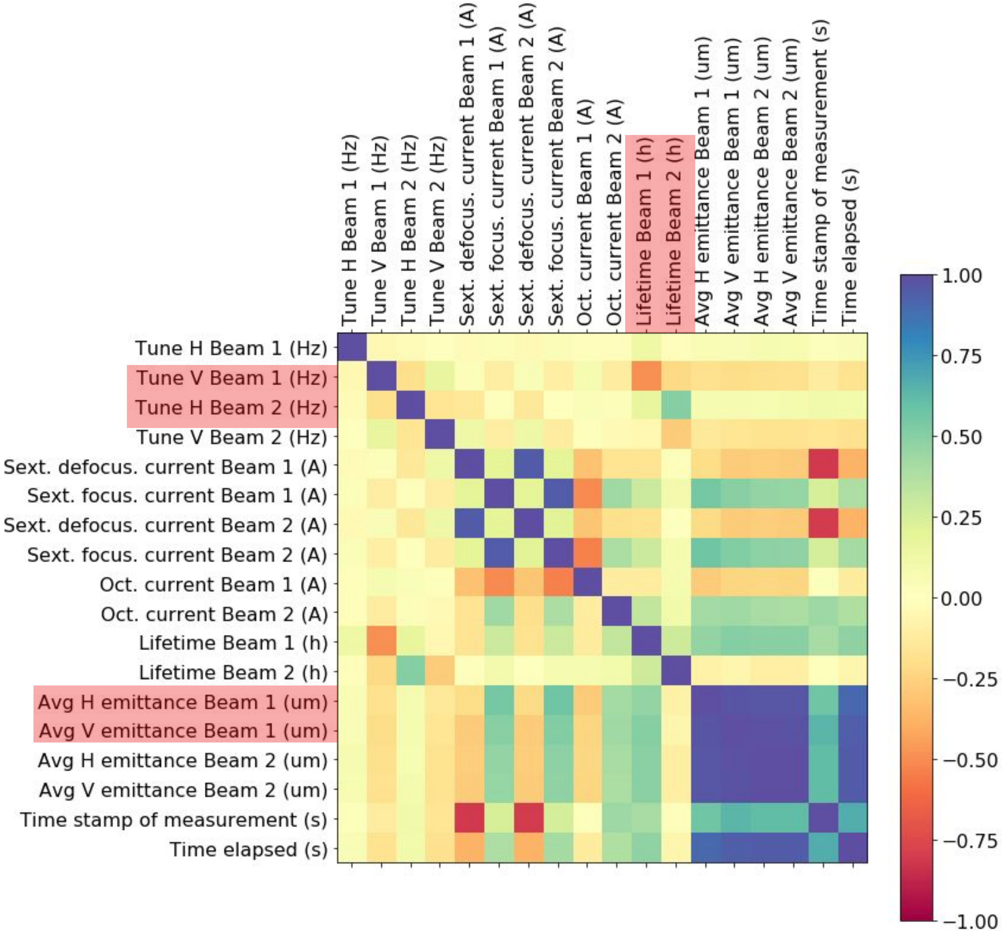
- Random tune trims around +/- 0.01 of the nominal injection tunes.
- The path of the trims was optimized to avoid far jumps crossing many lines.
- We move from one point to the other, both beams and H/V changed at the same time
- Scans repeated with different machine settings i.e. octupole/sextupole currents...
- Different scans for both beams and H/V planes to avoid fake cross correlations

MD data

Lost most of the cross beam dependence as both beams were scanned independently.

Very different correlations for both beams.

Beam 1 trade off with emittance

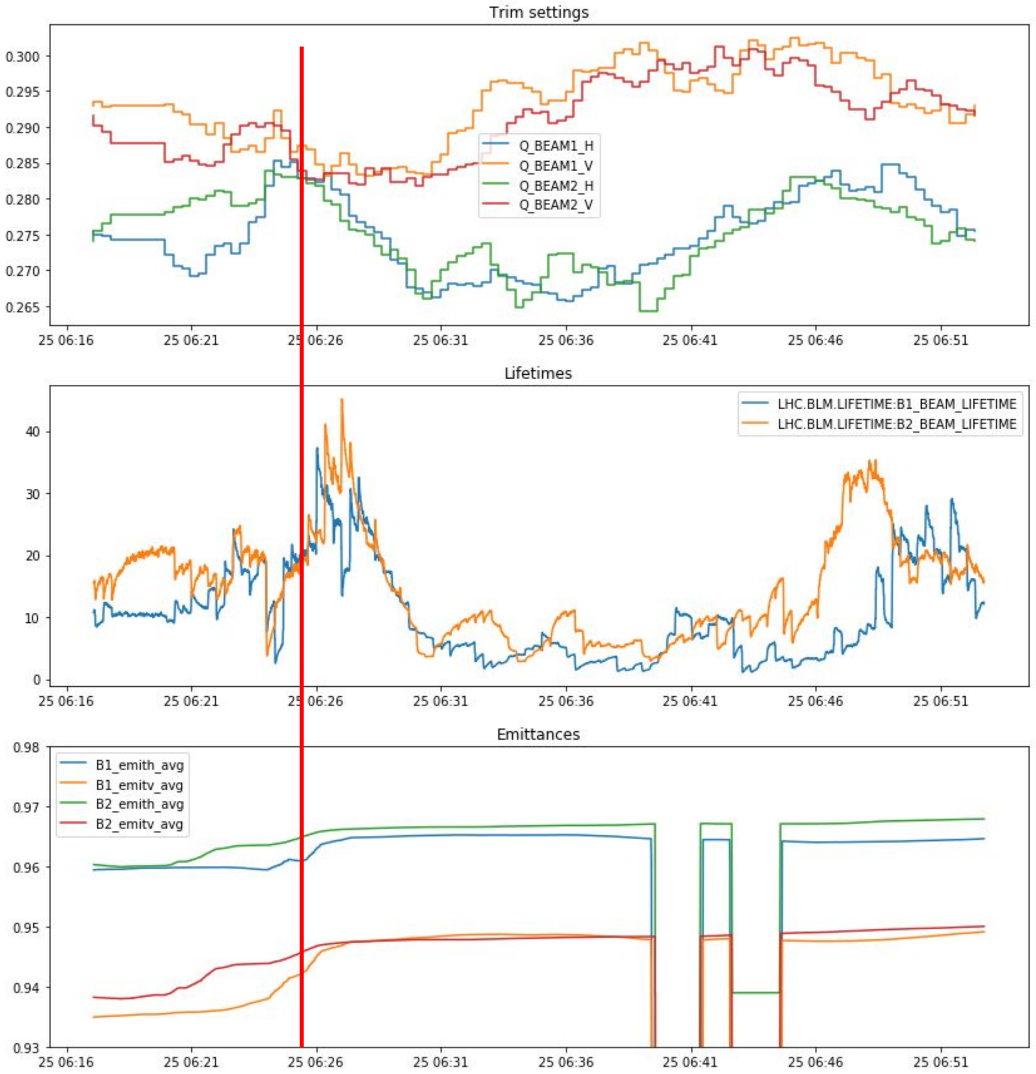


Experimental data
Main trends

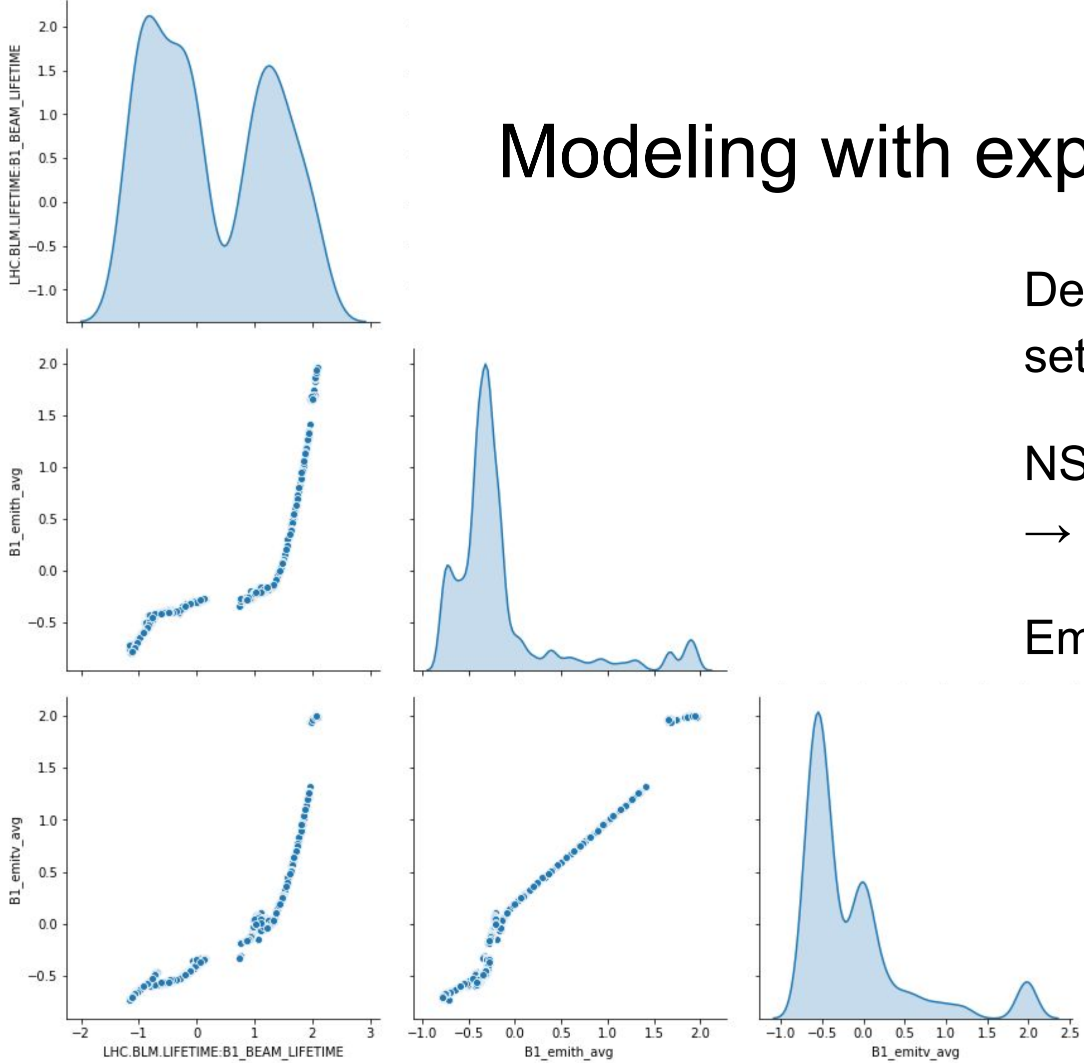
For the second trim sequence :

Tunes close to resonance, small ΔQ → High lifetimes and emittance increase.

Multi-objective optimization problem



Modeling with experiment data



Determining the pareto optimal settings from the MD model.

NSGAII multi-objective optimization → lifetime/emittance trade off

Emittance H/V ~ linear

Recommended settings vs data

Model: Gaussian Process B1

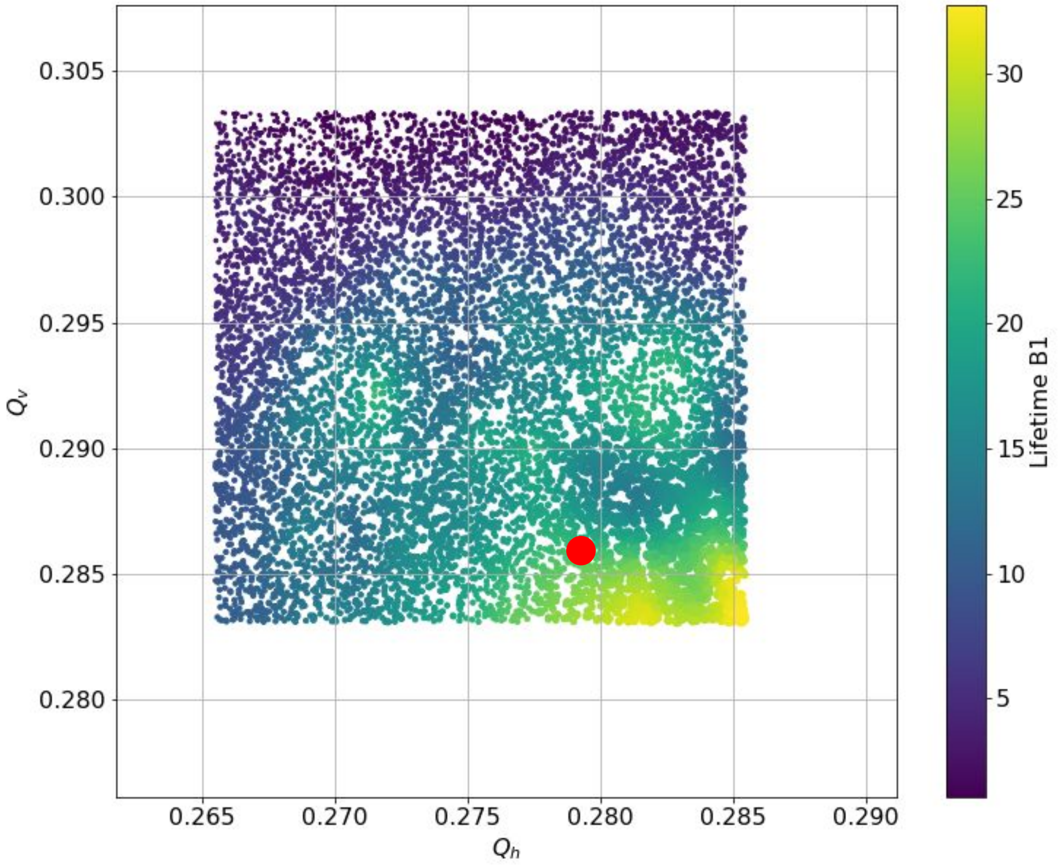
Multiple output:

Lifetime & emittances

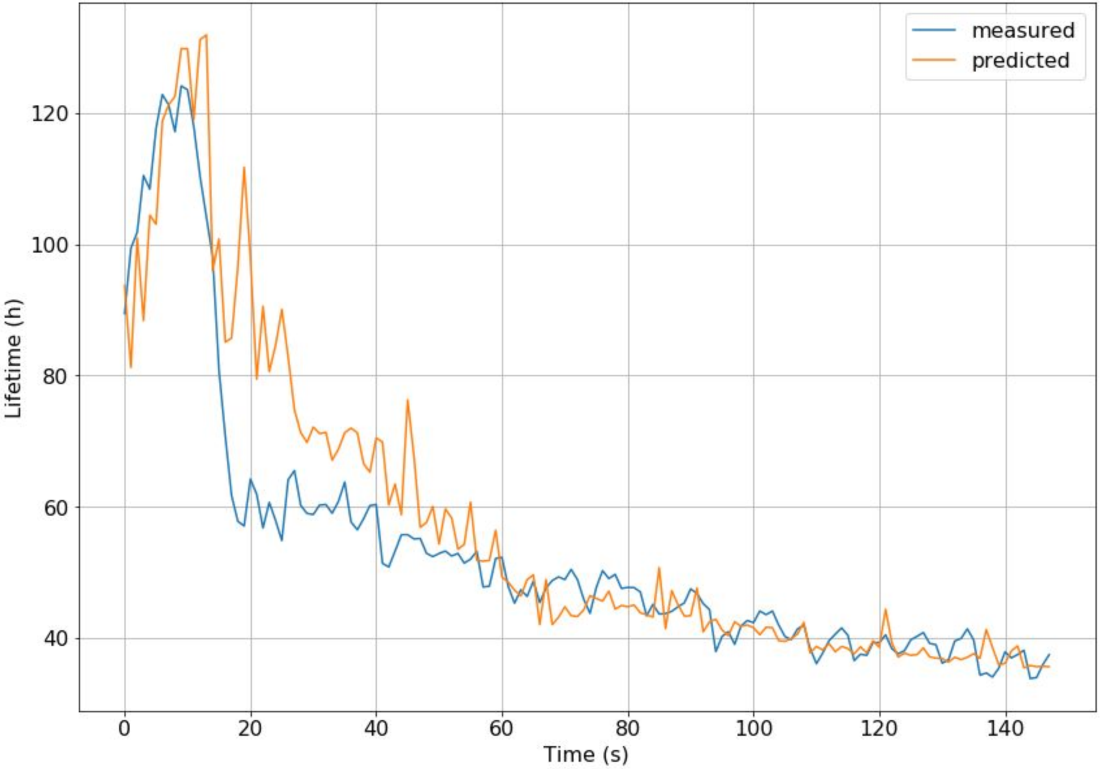
Recommended settings:

- $Q_h = 0.279$
- $Q_V = 0.286$

Seems to agree tentatively with MD data.



Proof of concept : surrogate model for optimization

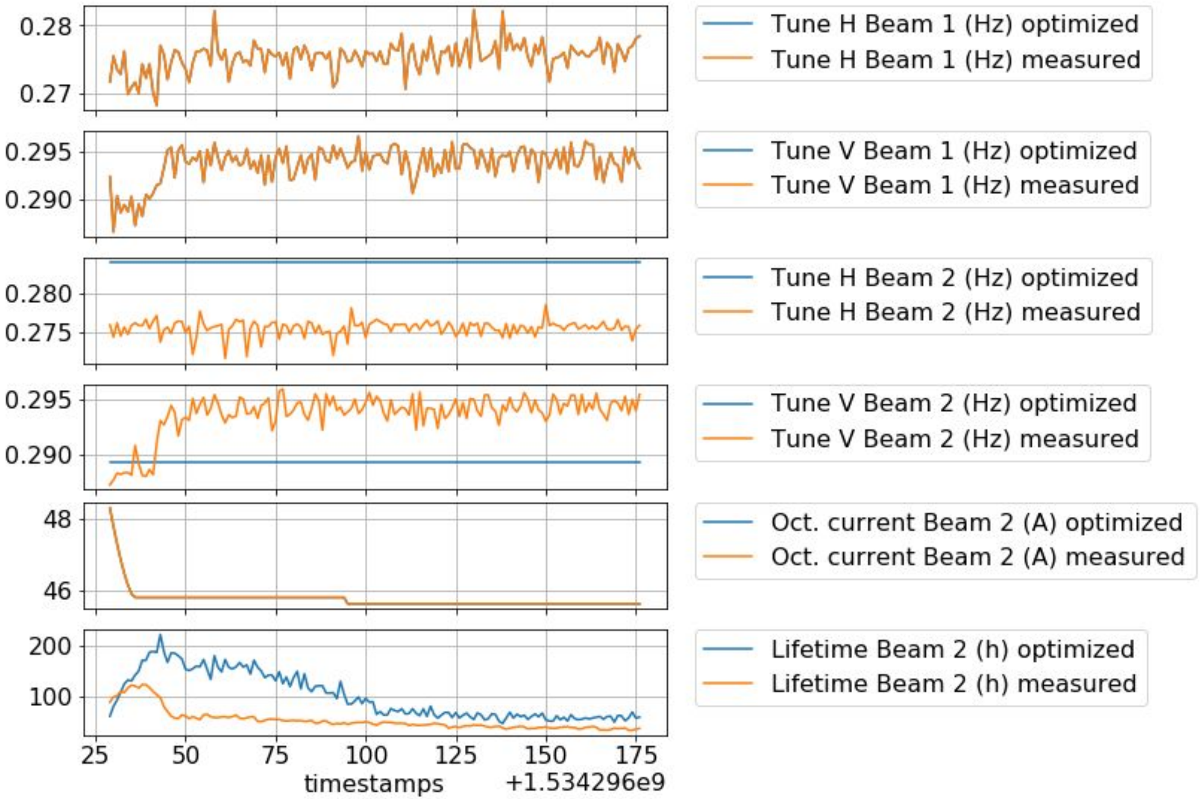


Example fill: 7056

Predicted and measured lifetime B2 evolution.

Proof of concept : surrogate model for optimization

BEAM 2



Optimization beam 2 tunes, prediction of beam 2 lifetimes.

Recommended settings:

- $Q_h = 0.283$
- $Q_V = 0.289$

BEAM 2

Recommended settings vs data

Model: Gaussian Process B2

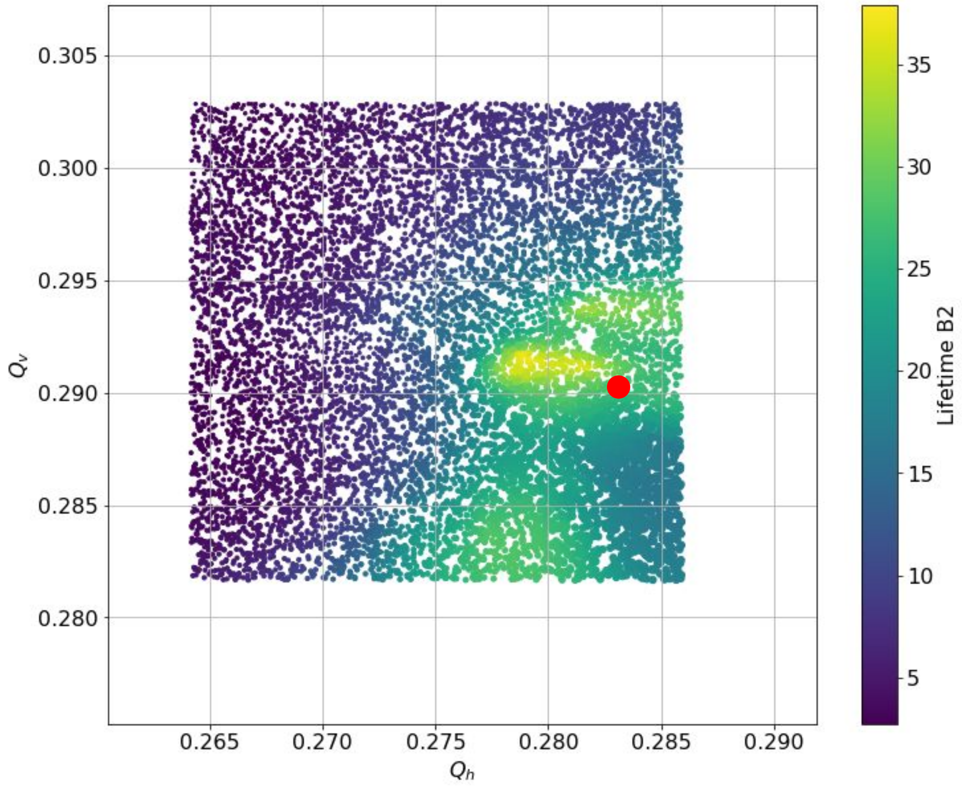
Multiple output:

Lifetime & emittances

Recommended settings:

- $Q_h = 0.283$
- $Q_V = 0.289$

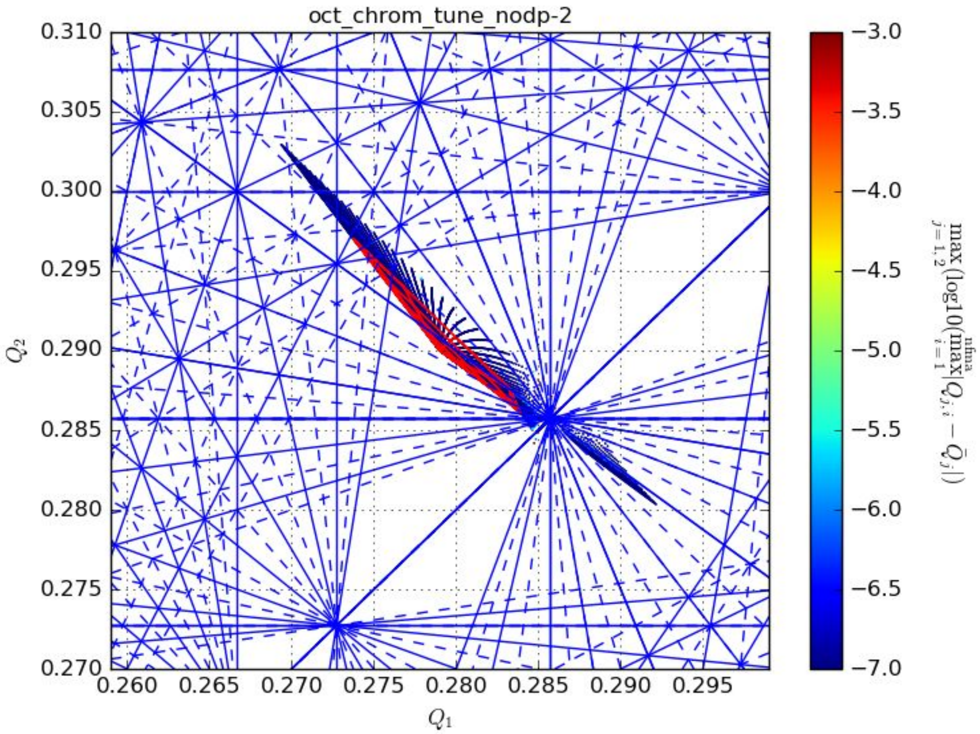
Seems to agree tentatively with MD data.



Simulations

Using simulation to understand the validity of “optimized” working point.

Long term particle tracking with Sixtrack



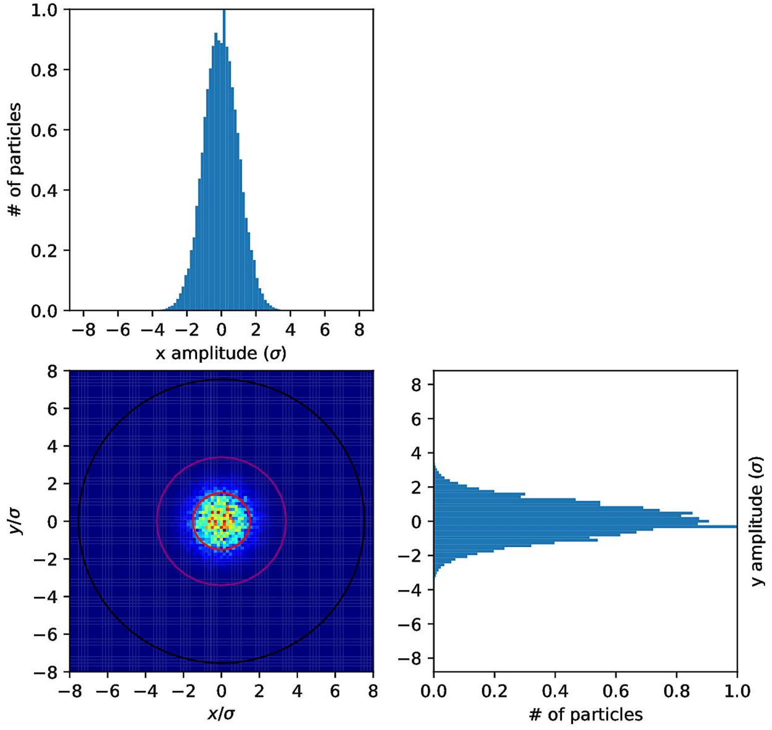
Extension to machine learning :

- Use simulations to produce additional machine learning dataset.
- Can explore at will input parameter space

Trained model could be used instead of time consuming tracking simulations.

Simulations

Using simulation to understand the validity of “optimized” working point.
Long term particle tracking with Sixtrack



Extension to machine learning :

- Use simulations to produce additional machine learning dataset.
- Can explore at will input parameter space

Trained model could be used instead of time consuming tracking simulations.

Conclusion

- Model successfully learns the expected trends
- Preprocessing and quality of data is crucial
- Interesting results:
  - Difference of tune measurement devices
  - Proof of concept of fill optimization → a recommended set of parameters to improve lifetimes
  - Dependency with time elapsed under investigation → missing variables ?
- Numerical models to support observations have been set-up
- MD → high quality data
- Multi output modeling → preliminary multi objective optimization

Outlook

- Multi-Objective lifetime/emittance model
- Larger dataset, introduce parameters at the bunch level, larger ranges of parameters must be explored (operation & MD ?)
- Improve the diagnostic of the system and the pre-processing of the available data: many quantities need to be introduced
- Go beyond just PRERAMP
- Define an “online” use to help operators with operational choices