­02/02/2023

**Paper:** [**https://doi.org/10.1140/epjp/s13360-021-01348-5**](https://doi.org/10.1140/epjp/s13360-021-01348-5)

Problem, usually correction is made few times a year, multiple magnets at a time. Goal, predict individual magnet errors as misalignment, sextupoles…

**INPUT:** deviation of the optic measurement from design \Delta(x)

**OUTPUT:** In this case, effective quadrupole field errors

source of the problem (change in intensity of field? position of magnet? Intensity of current of electromagnet?)

* Correction of the problem compensating predicted errors

How does mad-X work

**Pregunta**: The paper does not talk about the correlation between change in field and an applicable correction ie repositioning of magnet, intensity change

03/02/2023

**Paper: Thesis Tobias Persson**

Summary on beam optics theory.

Summary on beam measurements.

* Exciting the beam
* Phase
* beta
* K modulation
* Dispersion

Summary on beam corrections and procedures.

**Expected workflow:**

1 Generation of possible magnet errors using MAD-X (ARTIFICIAL Y)

2 Simulation using MAD-X => OPTIC MEASUREMENTS (X)

3 Algorithm training and selection Y\_pred = f(X)

4 Validation using model data and new EXPERIMENTAL DATA

Y => Deviation in magnetic field

X => beta(?)

Today I read and understood most of the theoretical part of the thesis, set up my github and played with MADX simulation data an OMC3 software.

06/02/2023

Ideas:

1. Non-linear optic errors

2. Measurement=>optic functions

08/02/2023

Running first MADX script.

Dont request many files from afs => DDOS.

Understand how to generate data from MADX.

Reading documentation for OMC python package

09/02/2023

This project is highly dependant on accelerator physics knowledge. I have to study and understand the problem. Lots of reading.

Trying to read .npy data from Elena does not work. Trying to run her script doesnt work either because of library dependencies.

13/02/2023

Library dependencies solved, VS code SSH setup, trying to understand elenas python script and MADX.

**Questions for Elena:**

1. **Data shape. 10 simulations (different seeds for the random distributions of errors I guess), 610 Measurements of (beta, mu, n) 8 possible arc magnet errors 2 possible sextupole errors for two beams (1,2) and two planes (x, y)**
2. **The data is for one turn or multiple turns, does this matter?**
3. **Weird learning curves, is this normal? (Not important)**

14/02/2023

**beta\*=> IMPORTANT PARAMETER** for focusing near triplet magnets, (collision)

**mu =>** Allows us to calculate actual beta parameter

**n =>** dispersion parameter (beam quality)

**Triplet error =>** Error in near colision triplet magnets (independent)

**Arc error =>** Error in circuit of magnets

**MQT error =>** ??

**Misaligment errors =>** Cant be fixed simulated to make data more realistic

**Possible improvements:** If you take your data as a matrix instead of a vector you have additional information on the mixed effects of magnets therefore deep learning with a multidimensional input would be interesting this is something I know how to do and would not be extra hard, maybe I have more idea than elena in this topic and can improve.

If everything is a lineal relationship with noise then it does not make much sense, I would like to know how important are the non linearities and how we can model then in MADX this is a field where ML can improve the existing tech.

**Possible improvements:**

* **CPYMAD allows for simulation and processing at the same time, no TWISS FILE**
* **Non linearities => Deep learning and comparing with Linear optics simulations**
* **FPD Metric for model performance (ML Group meeting) => K1L K2L K3L…**
* **Up to date optics, models, errors(?)...**

Opening the original .npy can be useful, I dont have to simulate everything

16/02/2023

Three files for sim errors =>

* Before matching
* After matching
* Common errors

How to obtain this measurements using cpymad instead of older madx wrapper

20/02/2023

Translation to cpymad complete, only for triplet errors. Still saving files because you need to read the errors from B1 when using B2. Not reading from python. Program still crashing and slow as hell. **Expected time to generate 10K samples, 7 hours aprox, manageable with 5 parallel simulations.** Trying to manage crashing, and do sanity checks!!!! Try higher number of processes as possible,

21/02/2023

Computer crash: Computer crashes after 5 iterations even without parallel computing.

PROBLEM: **SWAPING**, memory gets filled and everything stats running slow until it crashes

SOLUTION: Optimize, increase performance… All of them

10K samples => 138 hours, without parallel simulations, not manageable.

Optimization, what parameters get saved????

Debugging script and watch what variables get saved.

23/02/2023

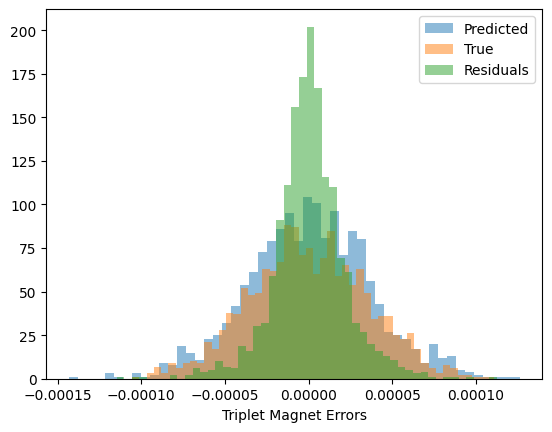
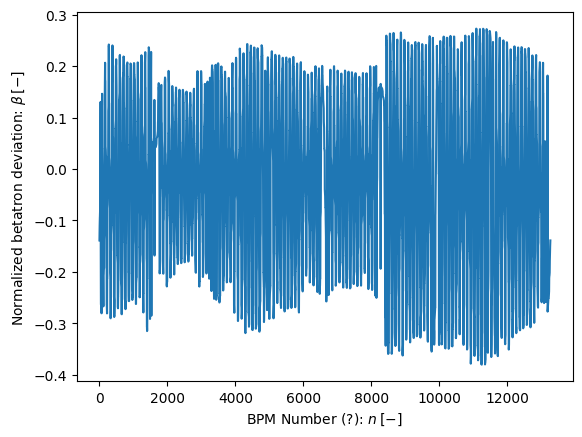
**State of the project:**

1. **Coding**
   1. Running Elenas code on updated packages: Done
   2. Changing from omc madx wrapper to cpymad: Done
   3. Memory allocation issues: Fixed
   4. Obtaining more data points: Pending
   5. cpymad running without saving files: Pending
   6. Expanding to all errors: Pending
   7. Running simulation on cs-ccr-dev3 server: Pending
2. **Optics, sanity checks**
   1. Comparing optics saved by python script with the saved files: Done
   2. Plotting relative beta ideal optics w.r.t. B2: Done
   3. Plotting error histograms for more data points: Done

Results, changing from the omc3 madx wrapper to cpymad has allowed me to learn the basics of madx, understand elenas code better and will allow for faster simulation times. Updating the code not to save files has proven being harder than expected, thinking of running everything from python, to make a cleaner code, however since running times are not an issue right now I will focus elsewhere. Right now I think it would be best to focus on the optics and theoretical part of the project, understand the simulations and the data better ie matching, errors, madx saved files, sanity checks...

RAM problem: created a madx instance for each iteration => RAM explodes even if deleting this instance 14 s/data point => 20-30 Hours per 10k points not ideal 17,8 s/data point

Plotting relative beta seems fine!

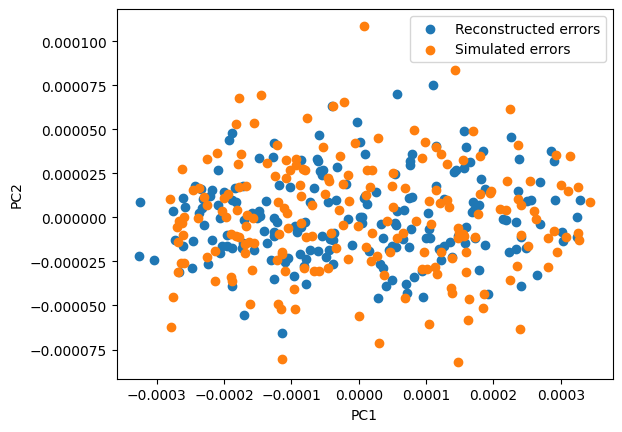
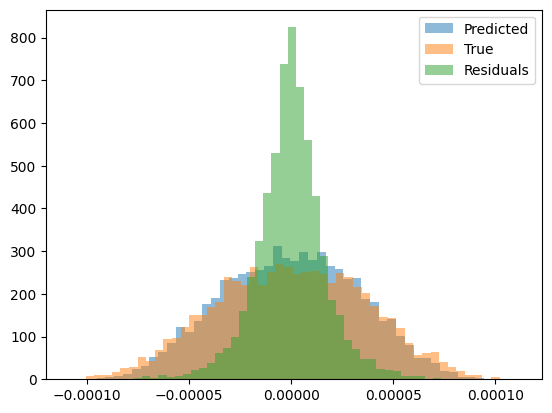


27/02/2023

No meeting, creating more data points. Trying to fix remote teleworking.

28/02/2023

200 Data points no crash, will leave simulation running on server today

03/03/2023

Creating only one madx instance and multiprocessing is really hard to implement. Going to run different instances of the program. Anyway it is faster to create multiple instances. Not going to focus in this really.

1. **Coding**
   1. cpymad running without saving files: Pending
   2. Expanding to all errors: Pending
   3. Running simulation on cs-ccr-dev3 server: Pending
   4. Running parallel code with only one instance: Pending
   5. Updating to 2020 optic settings etc: Pending
   6. Using pymadng(?): Ask Felix
2. **Optics**
   1. Reading possible problems and application: Pending

06/03/2023

Trouble connecting to remote, possible issues.

System suspension of CERN desktop due to memory issues (Most probable), change in IP address (Maybe), github ssh permission expired (I don’t think so)

Possible solutions:

Remote desktop Ubuntu-windows. Familiar.

VS Code remote development. Issues connecting.

Installing Ubuntu partition, bad idea

Using swan, easy to use different servers

TO-DO: Check if I can connect before and after suspension!

07/03/2023

Reading about MADNG: Questions, non linearities are a problem most of the time or only when the beam enters in an unpredictable caotic motion?

Reading about non linear optics:

Implemented cpymad in python class code:

13/03/2023

1. **Coding**
   1. cpymad running without saving files: Done
   2. Expanding to all errors: Pending
   3. Running simulation on cs-ccr-dev3 server: Done
   4. Running parallel code with only one instance, running on multiple instances: Problem (madx crashes when using pool)
   5. Updating to 2020 optic settings etc: Pending
2. **Optics**
   1. **Using pymadng(?): Ask Felix, hard to use**
   2. Reading possible problems and application: Pending

14/03/2023

Paralellisation done, time of execution 433 s for 10 scripts simulation 10 samples each. => 4.33 S per sample => 100 K samples is 5 days,Probably less since the longer the script runs the better.

Lets try with more scripts and see how it performes. Make sure that the data does not get overwritten.With 20 simulations 472 s => 2.36 S per sample => 66 Hours

|  |  |  |
| --- | --- | --- |
| N\_Parallel | Time Per sample | Time 100 K samples (extrapolated) |
| 10 | 4.33 s | 120 h |
| 20 | 2.36 s | 66 h |
| 30 | 1.48 s | 41 h |
| 80 | 1.4 s | 39 h |

1. **Coding**
   1. Extending for all previous magnets: Pending
   2. Updating to 2023 optic settings for 30 and 60 CM: Pending
   3. Simulation 100k data points: Pending
2. **Optics**
   1. Reread Elenas paper: Pending

16/03/2023

Fixing SSH problems definitively.

1. **Coding**
   1. Extending for all previous magnets: Done
   2. Updating to 2023 optic settings for 30 and 60 CM: Pending
   3. Simulation 100k data points: Pending
2. **Optics**
   1. Reread Elenas paper: Pending

21/03/2023

1. **Coding**
   1. Extending for all previous magnets: Done
   2. Updating to 2023 optic settings for 30 and 60 CM: Done
   3. Simulation 100k data points: Pending
2. **Optics**
   1. Reread Elenas paper: Pending

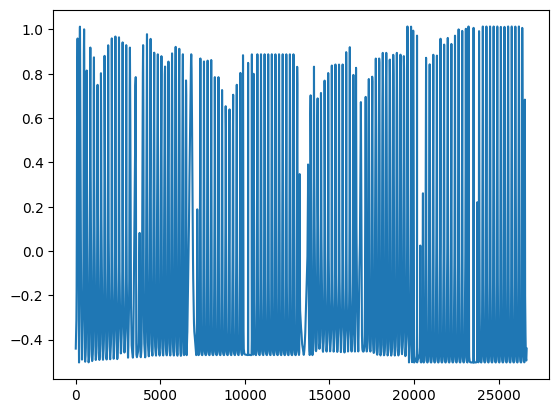
Still Have to reorganize and clean all the code.

27/03/2023

Lost some journal days. Results of simulation, bug fixing. Decision trees perform worse than ridge. Try svm.

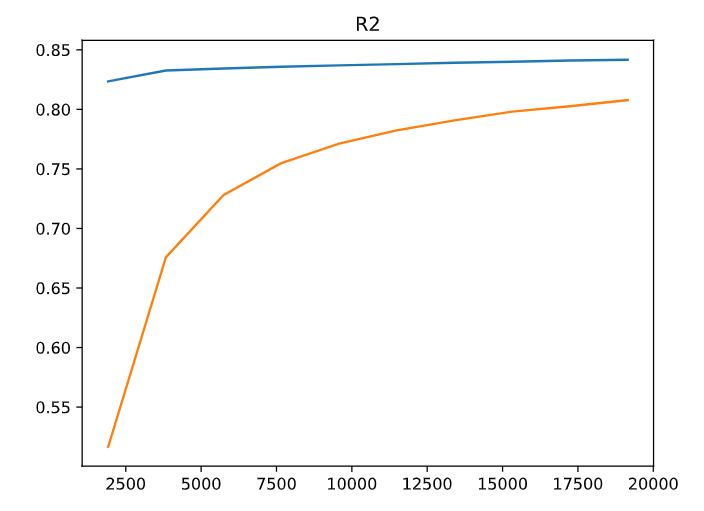
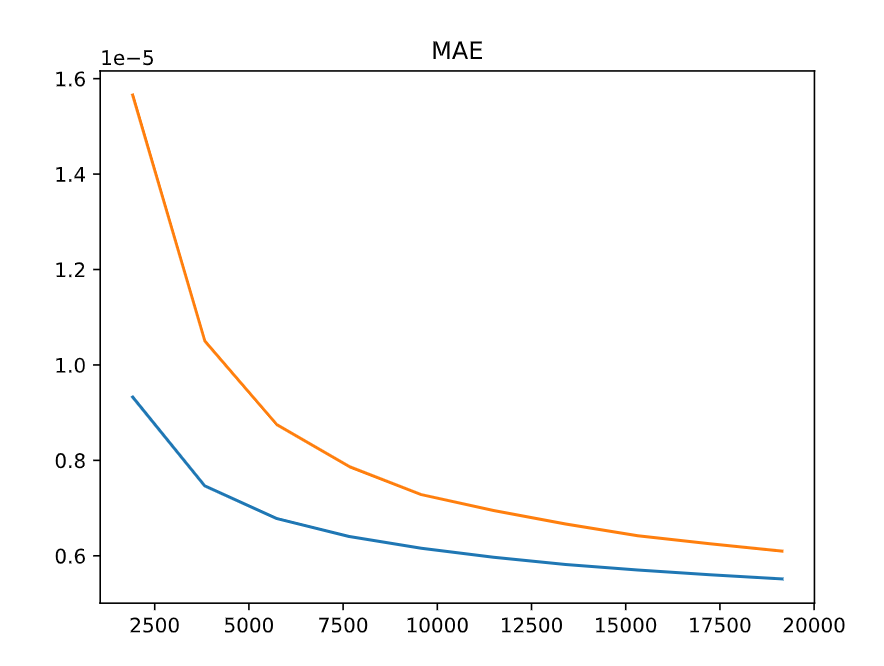
04/04/2023

Comparing before and after matching twiss



11/04/2023

Making a function to save predicted errors in a .tfs file. Played with pymadng, but first I want real results using madx. Training for 20K samples. Trying to use svr, no way.



Reasons why the sample generation seems to be failing.

* R2 too good for this amount of samples.
* 10-15% Of simulations fail
* Sometimes the MQT knob thing gives wrong outputs

All of this things can make sense.

12/04/2023

Presentation in 2 weeks.

**- Visualizaciones de la simulacion**

- Errores predecidos vs simulados para cada iman.

- Errores simulados vs usados en el algoritmo para comprobar que la simulacion esta bien y no es demasiado simple errores normalizados

**- Testing the model**

- Predecir twiss y meter las correciones de los datos reales y ver si funcionan

- Con los errores generados predecir twiss y comparar con el medido. Problema, puede generarse con distintos errores.

**- Problemas**

- R2 demasiado bueno, quizas pq el modelo es demasiado sencillo. Quizas porque ha disminuido la degenerancia del LHC.

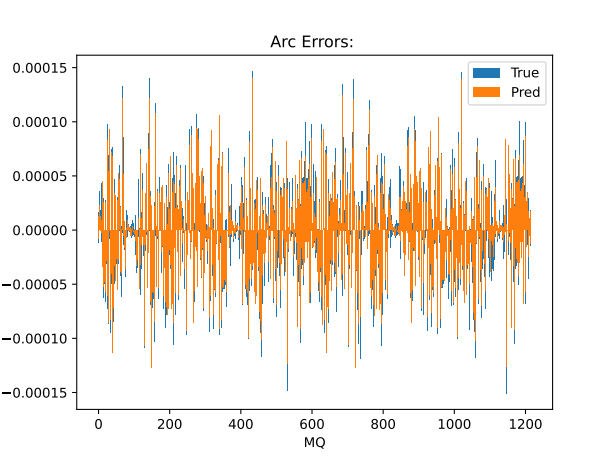
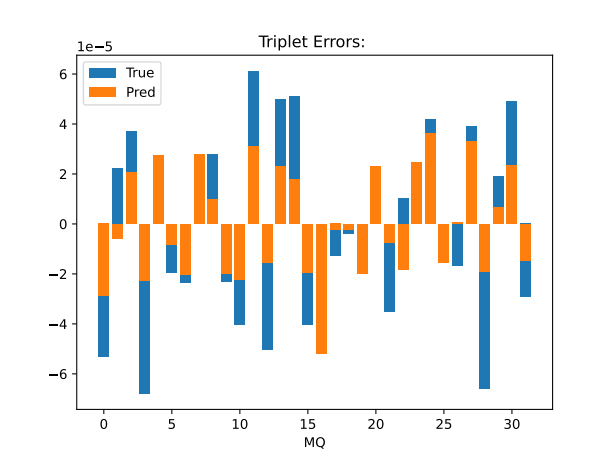
- 15% de las simulaciones fallan, quizas por usar opticas de 30cm y un beta mayor en los arcos

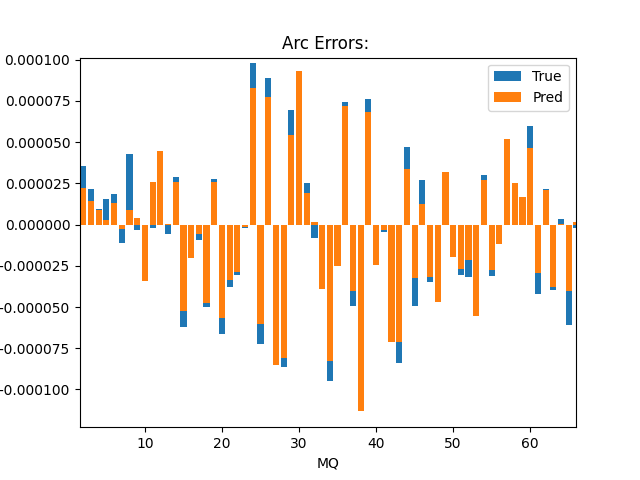
**- Ideas**

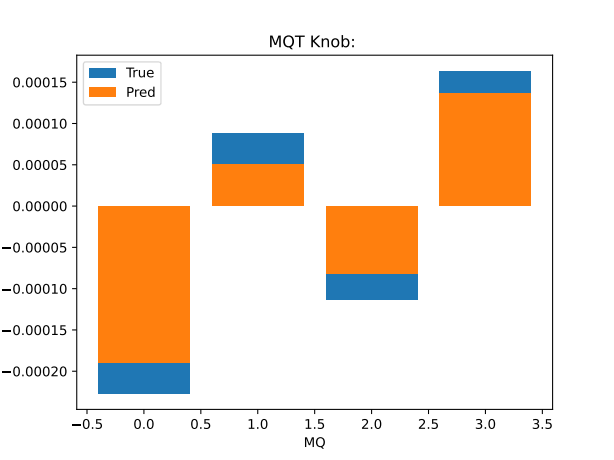
- El test con datos reales no parece optimo, quizas se podria hacer alguna forma de testearlo mas sistematicamente, predecir las correciones directamente

- madng, using gradients for training

- si estamos generando errores como se hacia en 2016, eso es lo mismo para los nuevos arcos??? quizas ha cambiado la tolerancia, ya que ha cambiado la beta

Simulated vs predicted errors for random sample:





At first glance the model performs worse for the triplet errors, this is not expected, maybe it is because the curse of dimensionality. A better way to visualize this is with the correlation plots by Elena. The simulated errors and the ones used for training seem to be the same. PROBABLY NEED MORE DATA!!!!

Testing in a random dataset, maybe not the best practice but still expected lower.

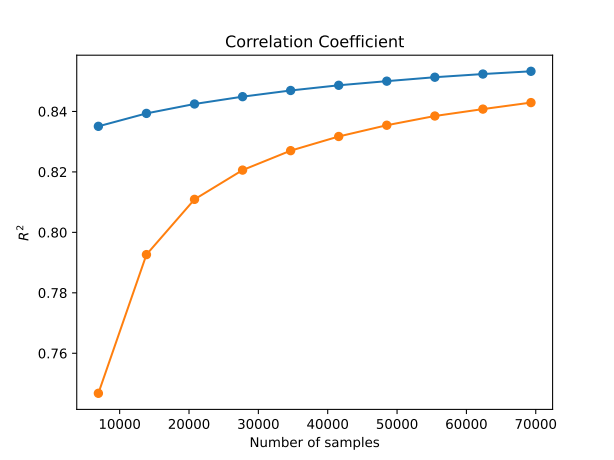
Train R2: 0.779 Test R2: 0.788

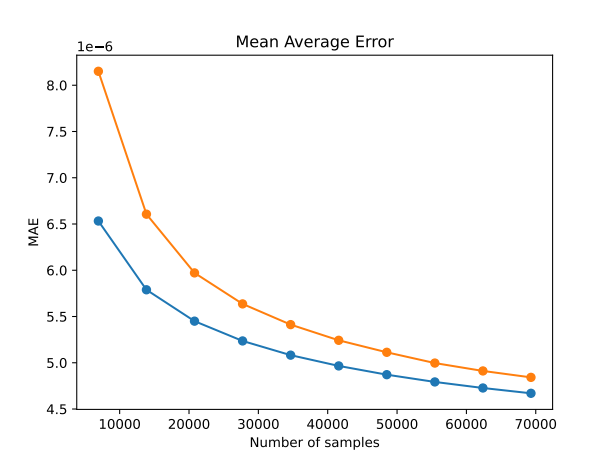
Train MAE: 8.41e-06 Test R2: 8.58e-06

Tomorrow train with more data, next days plot different results. Hope for the best about the shift, as long as its not on friday working hours I am good, lets hope.

13/04/2023

Training with 70 k samples 80/20 split for test, results are still improving, will make plots to understand the results better.





Figures I need to add to the presentation:

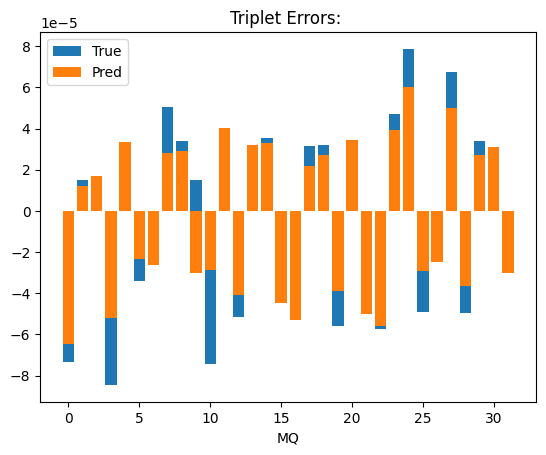
* Data pipeline, Done
* MAE and R2, Done
* Errors generated VS predicted histogram for different magnets, Done
* Twiss, Done
* Backup: Some error comparison and performance for different types of errors

Be careful if figures are obtained from test or train data, big difference.

The presentation should have scientific type communication and peer to peer communication. How its actually going etc.

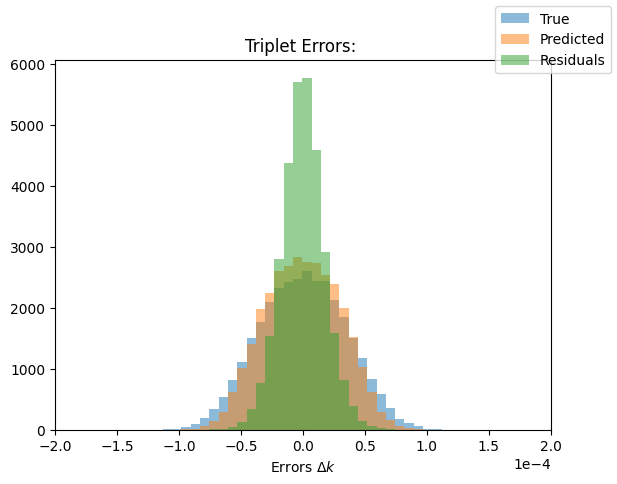
14/04/2023

Other example triplet error. This plots should take into account test vs train splits. In this example performance seems to be better, however maybe there is still worse performance on triplets.



14/04/2023

Understimation of triplet errrors especially, This might be because of ridge regression. As seen like in the next figure, still, results dont make as much sense as I would like them



IDEA

* Trying using linear regression instead of ridge.

Training: R2 = 0.8875, MAE = 3.45e-06

Test: R2 = 0.8715, MAE = 3.69e-06

Better results overall, compared to the 0.85 using ridge regression.

It seems like the understimation of the errors comes from using ridge regression.

**This last weeks summary:**

* Plotting errors, systematic understimation
* Beginning presentation
* Trying normal linear regression instead of ridge => No systematic error
* **RESULT: The systematic errors in elenas paper probably come from using ridge, ridge penalizes strong dependencies therefore making an estimator more robust to variance**
* Plotting example twiss! Some twiss are not centered in cero.

PLOT ONLY FOR TEST DATA. CHECK WEIRD TWISS

18/04/2023

RESULTS FOR RIDGE AND DIFFERENCE BETWEEN TRIPLET AND ARC PERFORMANCE

Training Triplet: R2 = 0.840, MAE = 4.752-06

Test Triplet: R2 = 0.836, MAE = 4.763e-06

Training: R2 = 0.853, MAE = 4.67e-06

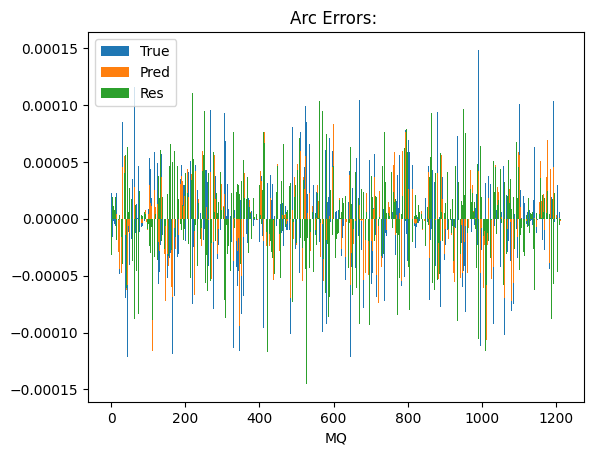
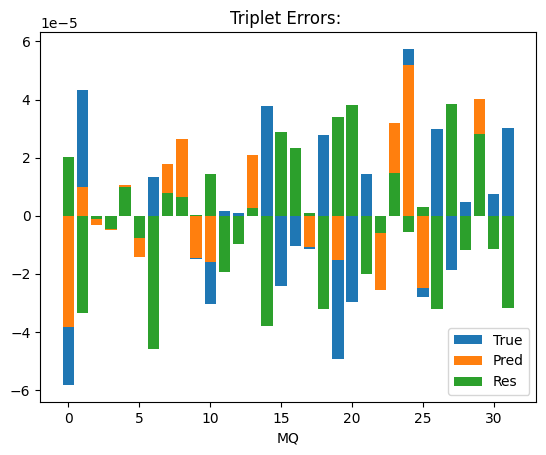
Test: R2 = 0.8430, MAE = 4.83e-06

Generated all figures for test data, obtaining BIG residuals according to histograms, maybe histograms are more centered!! In addition to the weird twiss data.

19/04/2023

RETRAIN AND TEST IN TEST DATA

Some results are much much worse, still for most samples the results are great, t**his might explain the thick histograms**. This means some samples are not being generalized by the model example sample 16. R2 and mae is great either way. Are these outliers??? **TRY USING RANSAC**



The fact that this works so well for some samples but others horribly suggest that some simulations enter a non stability region and are much harder to predict than others.

Decission trees work worse still.

21/04/23

With 45 cm optics the errors are still in the order of 1E-3

24/04/23

LINEAR 45 CM

Training: R2 = 0.8637265532726398, MAE = 4.13665951038814e-06

Test: R2 = 0.8562377544789292, MAE = 4.250593427395733e-06

RIDGE 45 CM

???

28/04/23

Ideas after presentation

1. Adding noise Done
2. Plotting beta beating histogram and training with more realistic data Done
3. Testing degeneracy training with and without increase in beta for ATS arcs Pending
4. Trying to predict \Delta p / p Chromaticity Pending
5. Recreate optics Pending
6. Introduce 2023 corrections Pending

IDEA!

Triplet performance is due to missing noise! Studying performance for different levels of noise! The parameters have not chaged!!!!

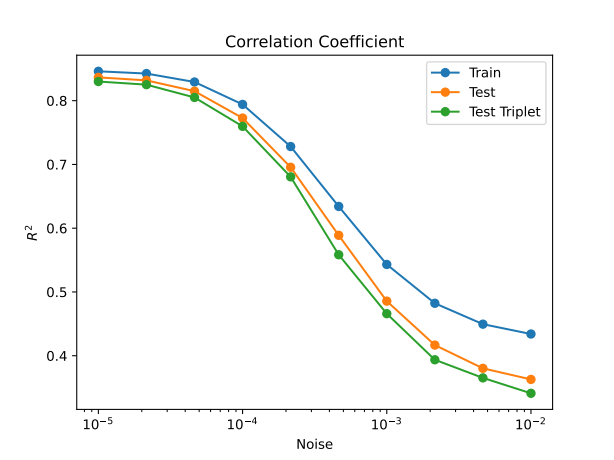
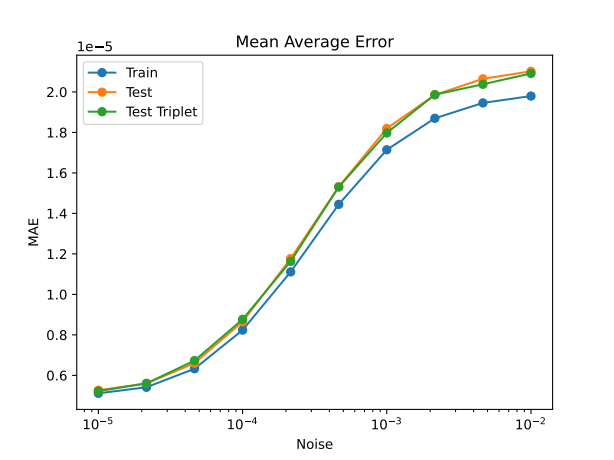
02/05/2023

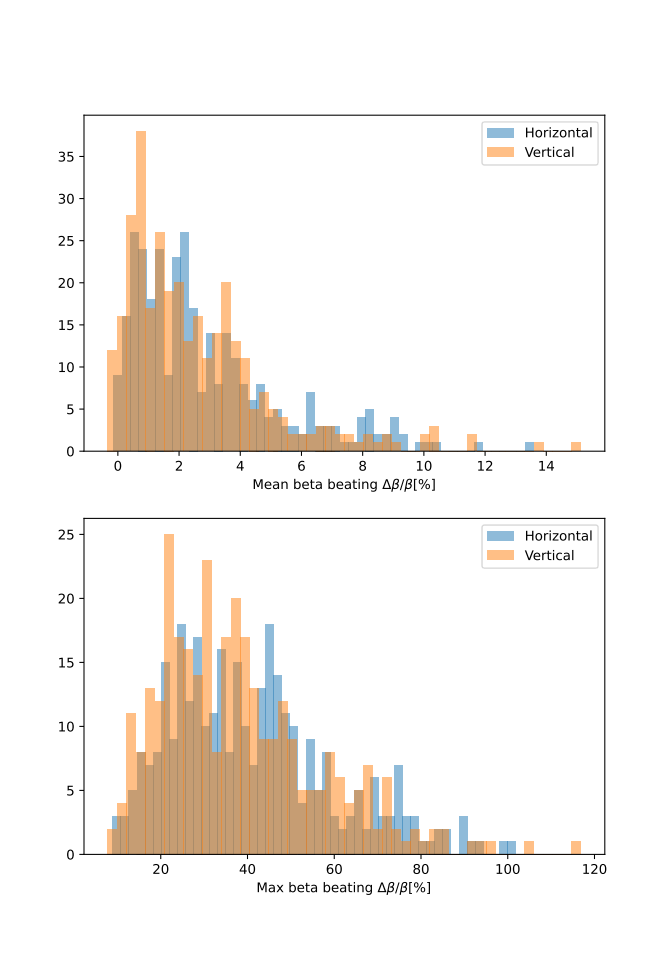
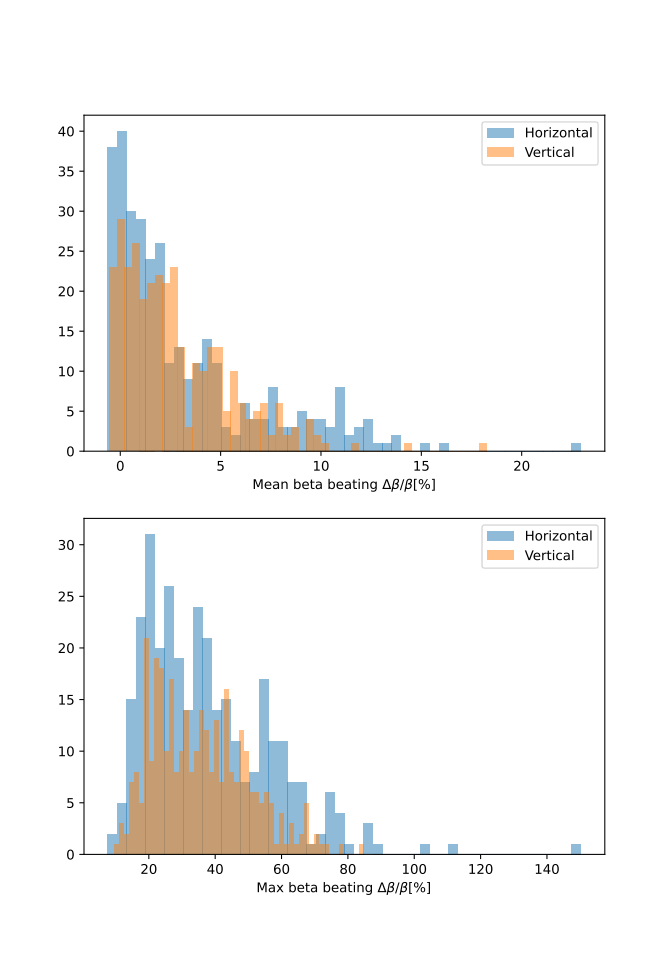
Data generation with a 30% cutoff

Effect of noise in ridge regression, used noise level is 10E-3 according to Elena.

Recreating optics is nearly complete!!!

e simulated normalized dispersion deviations,





03/05/2023

Optic recreation completed, still have to compare with a better example and with the original true errors.

**Results 30cm, ridge, 1e-3 noise for phase and disp**

Train Triplet: R2 = 0.5117993670334481, MAE = 1.7202585891941248e-05

Test Triplet: R2 = 0.4633846137835224, MAE = 1.8070333017981448e-05

Train: R2 = 0.5430435119648702, MAE = 1.7143778452768323e-05

Test: R2 = 0.4857963488172039, MAE = 1.8183475904515684e-05

**Results 30cm+bb30, ridge 10e-3 noise for phase and disp**

Train Triplet: R2 = 0.4832839275335945, MAE = 1.7812030836917395e-05

Test Triplet: R2 = 0.46073526707691526, MAE = 1.8195960713492678e-05

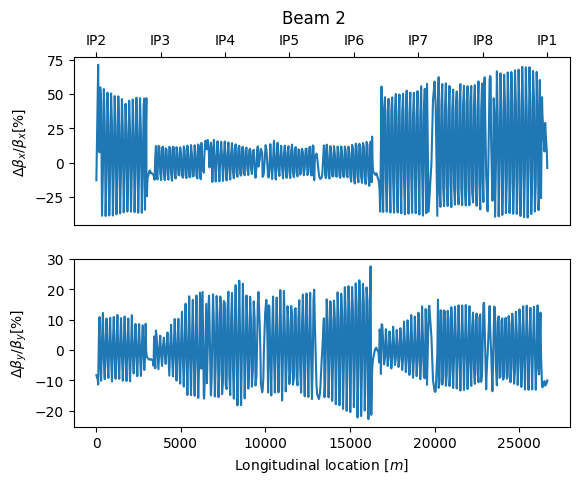
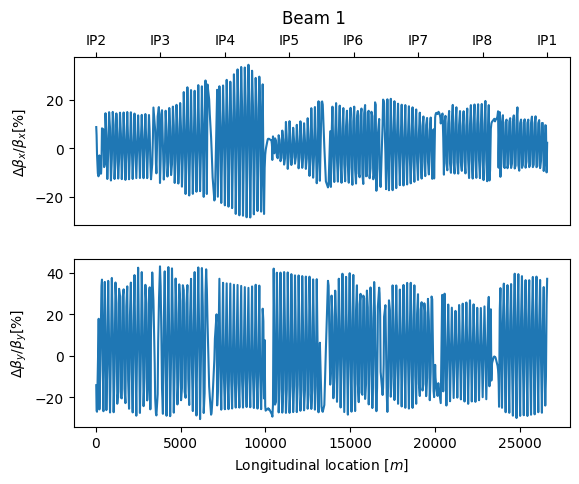
Train: R2 = 0.5216334719843492, MAE = 1.767973943729175e-05

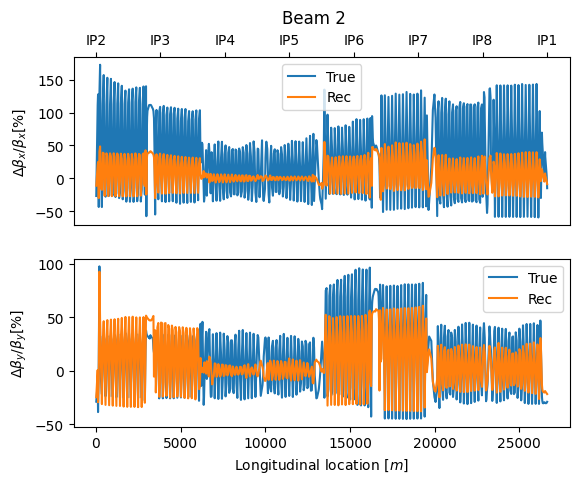
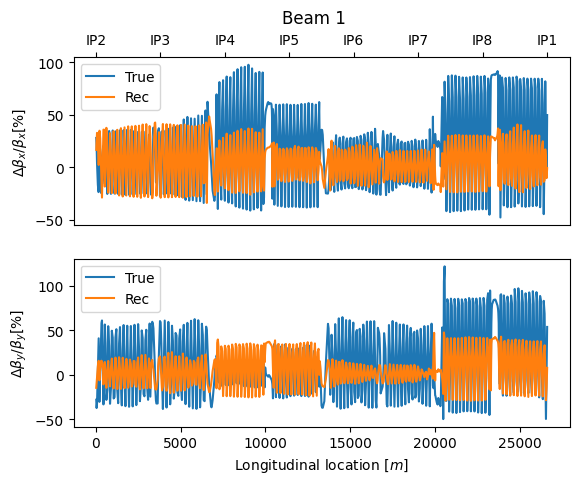
Test: R2 = 0.48848438816963746, MAE = 1.8284178935036706e-05

Worse results are due to the fact that the noise figures for Elena are different studies that the one we are using.

Ideas after presentation

1. Adding noise Done
2. Plotting beta beating histogram and training with more realistic data Done
3. Testing degeneracy training with and without increase in beta for ATS arcs Pending
4. Trying to predict \Delta p / p Chromaticity Pending
5. Recreate optics Done
6. Introduce 2023 corrections Pending

**Recreating optics sanity check:**

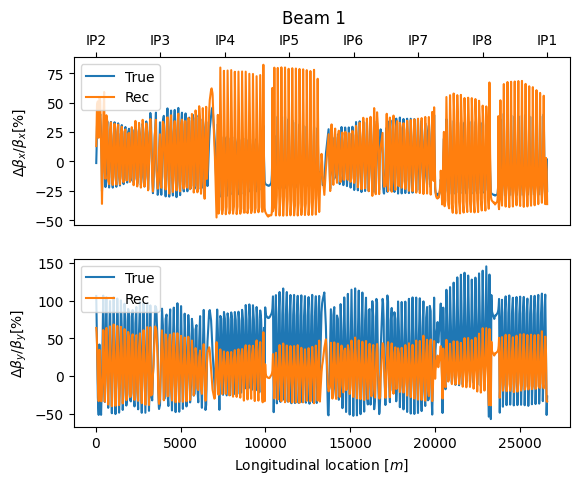
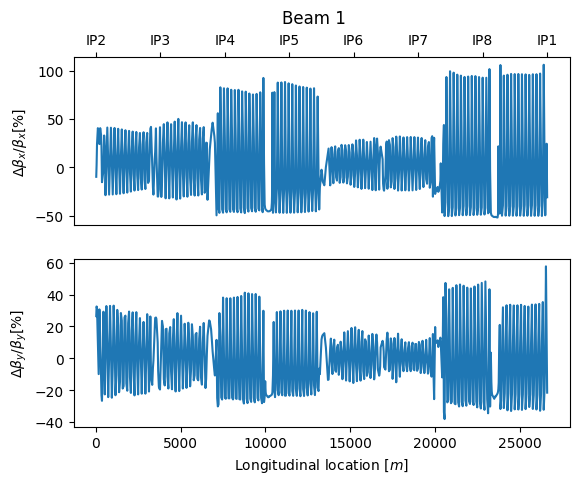
****

Something is completely wrong, a bug. The true betabeat should be the same!

- Not the noise

- Not the matching

Bug found, WRONG NOMINAL (using 45cm instead of 30cm) for BB30 data, testing with correct nominal, still no improvement

This is without adding the dipole errors in the reconstruction and without tuning an dipole

The problem is for the true errors not the predicted. Maybe mqts?

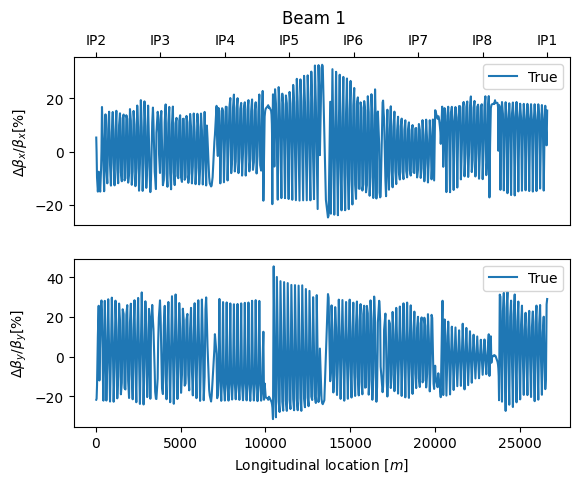
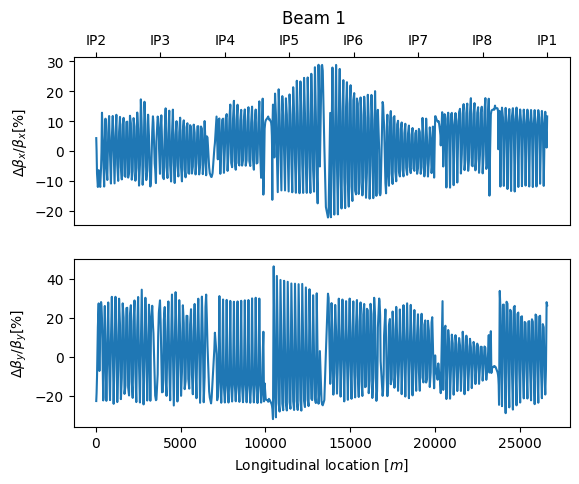
The mqts errors can not be cero!!

Introducing the mqt predicted by the ml algorithm, but we need the absolute value of the error, and also

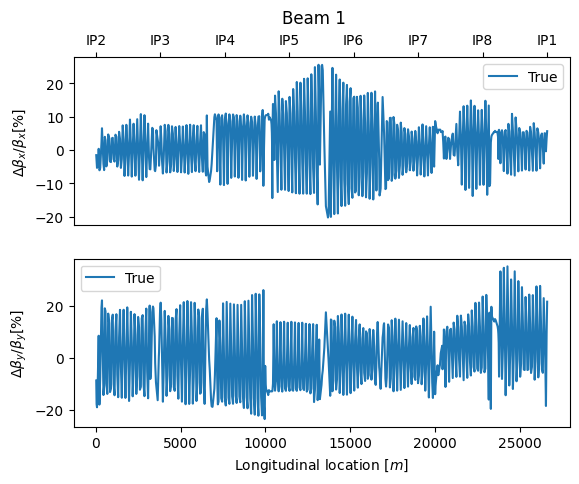
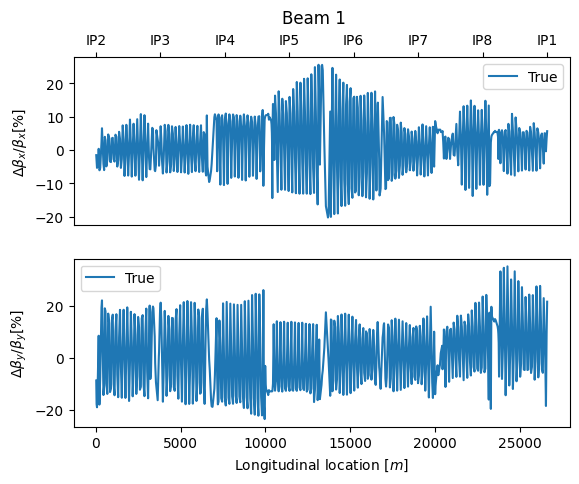
08/05/23

New test sample for reconstruction seed = 1 using the tfs data straight from data generation

**REAL RECONSTRUCTION**

****

**NOT MATCHING MATCHING**



Solution, not introducing the mqt values and matching to get them!!!

Not working!!! The matching is not really impacting, ploting different examples.

Probar nuevos algoritmos en vez de seguir por aquí!

09/05/23

Setting up training using LSTMs and CNNs. Searching for some data to use from the CERN logbook.

10/05/23

Trying to train using GPUs, HTCondor, lxplus gpu, and SWAN all give me issues. I think

- HTCondor is not as staight forward

-lxplus gpus is giving me some issues with the packages or cuda, maybe cuda not installed (???)

- Swan GPUs must be down today or something like that.

11/05/23

- Reinstalled tf in lxplus gpu,

-Training time much shorter 540 f/epoch to 25 s/epoch to for one of the models and 1/40 of the data around 50 times faster and maybe more.

12/05/23

-setting up CNN model!!!

15/05/23

-solving bugs and trying regression tree

24/05/23

Dec tree faster with MSE, choosing branch depth

Summary:

- Noise

- Twiss Recreating

- Error, mqt error generation, ask

- DL:

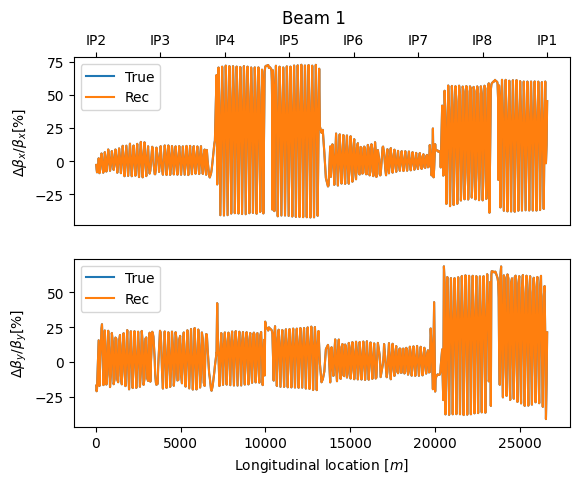
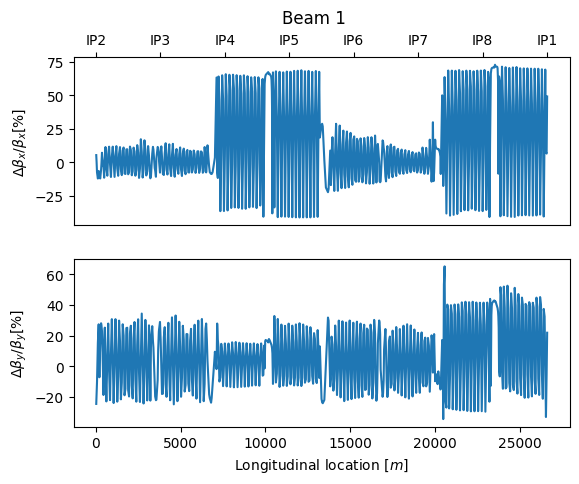
-Setting up GPU

- LSTM and CNN

Future: which measurement on the logbook would be better

Model training for beam 1  
- > assign arc errors as usual (triplet errors 1/10 of the usual strength to reflect the local corrections in + small residual errors)  
-> use the global corrections used in the machine  
  
Input: phase advance errors at every B1 BPM  
Output: arc errors B1 (including MQTs)  
  
Model training for beam 2:  
If we can out global corrections:  
-> usual errors in the arcs, 1/10 of usual triplet error values  
  
If not:  
-> same as for b2  
  
- maybe we can trim Q4 and measure triplet errors?  
  
Input: phase advance errors at every B2 BPM  
Output: arc errors B2 (including MQTs)  
  
For ‘virtual’ k-mod:  
Models trained on the data from same twisses:  
  
Input: phase advance errors everywhere  
Output: betas at BPMs: BPMSW.1L1.B1, BPMSW.1R1.B1, BPMSW.1L2.B1, BPMSW.1R2.B1, BPMSW.1L5.B, BPMSW.1R5.B1, BPMSW.1L8.B1, BPMSW.1R8.B1 (.B2 for the models trained for beam 2)

**CORRECT AND TRUE MATCH, ORIGINAL DATA GEN BUT NOT ORIGINAL, BUT CLOSE**

****

**The difference is the misalignment errors**

26/05/23

**Prepare for MD:**

+ Input correct errors: Pending

+ Test and plot virgin model with different errors simulated: Misalign, mqt, triplets… Pending

+ Find correct optics: Pending

+ Prepare and test models for MD:

- B1 Pending

- B2 Pending

+ Prepare measurement reading script and test on existing measurements Pending

+ Create best knowledge model for twiss reconstruction

- Create the model Pending

- Plot difference between twisses Pending

Could try to predict misalignments

According to Elena all errors are relative errors, the macros work like that. “All quadrupoles in the lattice are assigned a random relative gradient error obtained from uniform distribution with the same rms error σ per magnets family”

Which tune should I use??