Evaluating ECAL timing properties using spacal data and Machine Learning

Student: Fakanov Pavel
Supervisor: Ratnikov Fedor

Introduction

- We want to compare the perfomance of Machie Learning algorithms for two different datasets
- The goal is **not** to improve particular baseline algorithm with ML
- The goal is to use ML to extract the maximum of available in data information and evaluate limitations of the possible physics performance that are driven by behaviours of actual, test beam data
- We would like to explore how the quality of algorithms depends on the stability of the data

A few words about data

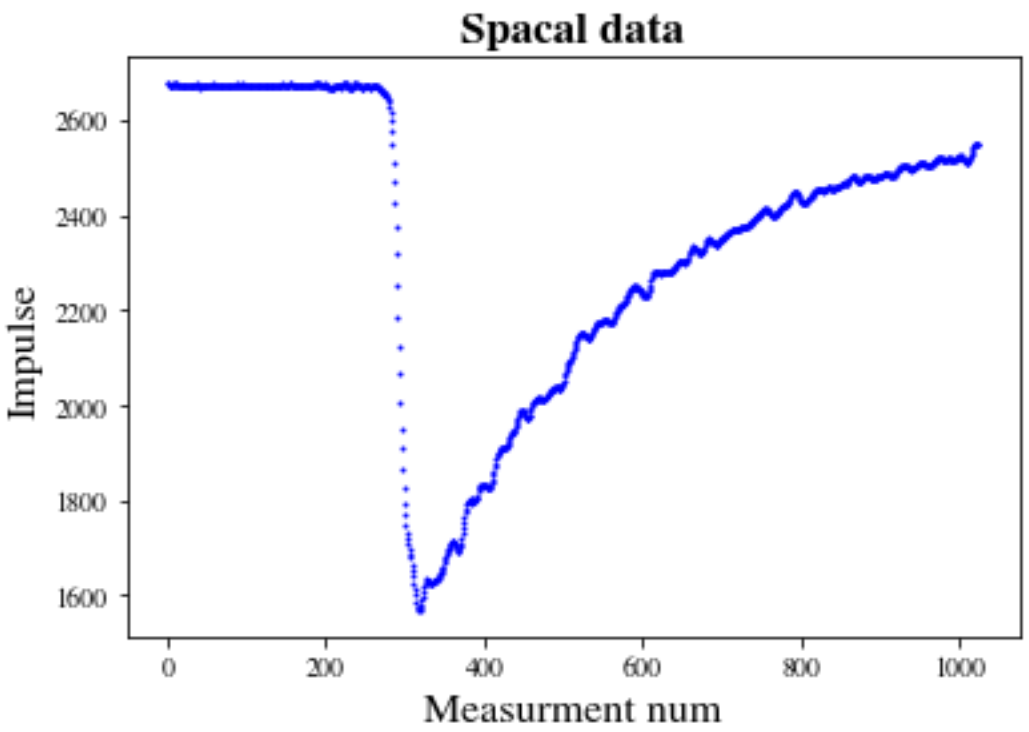
- Data obtained from the 30 GeV electron beam
- Use “output” module of the LHCb electromagnetic calorimeter
- Reference time is measured as average of two quick scintillator counters
- *7848* signals
- Each signal is *1024* impulse measurements, sampled with a step *200* picoseconds (5GHz) + the reference time of a signal

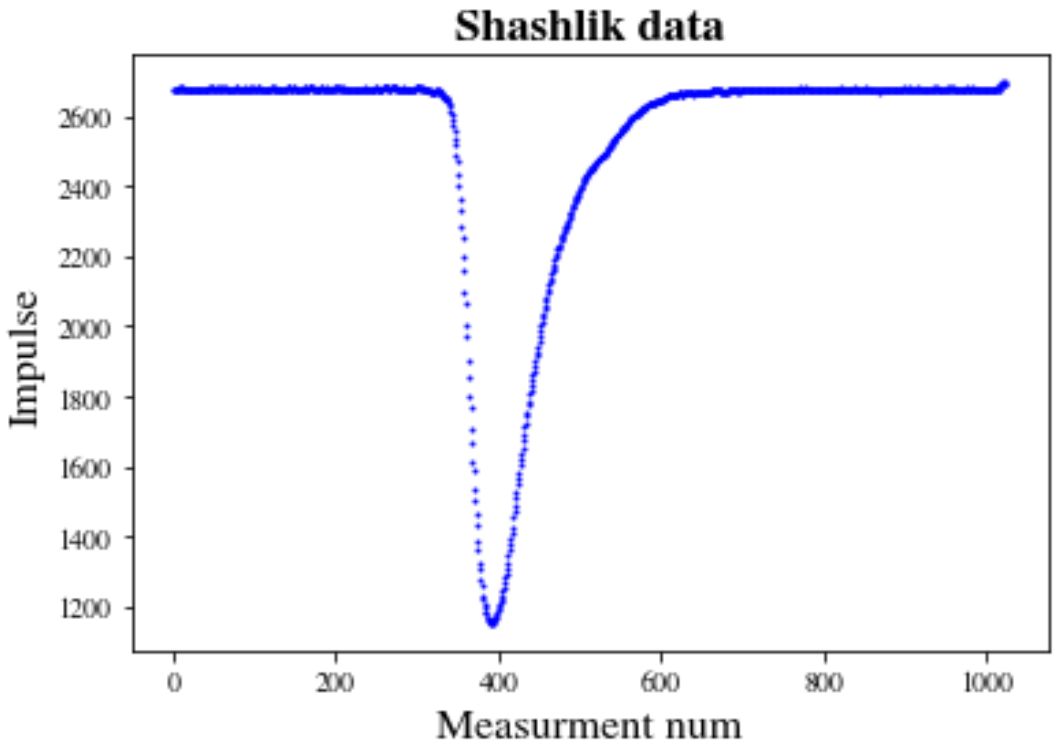
Many thanks to Yury Guz for providing data and fruitful discussions

What do we want?

- Explore possible quality of signal recovery for different sampling rates
- Explore the ability to distinguish between single signal and two close signals for different sampling rates
- Explore the ability to determine the time and amplitude of a signal in the presence of a second close signal for different sampling rates

Spacal vs Shahlik data

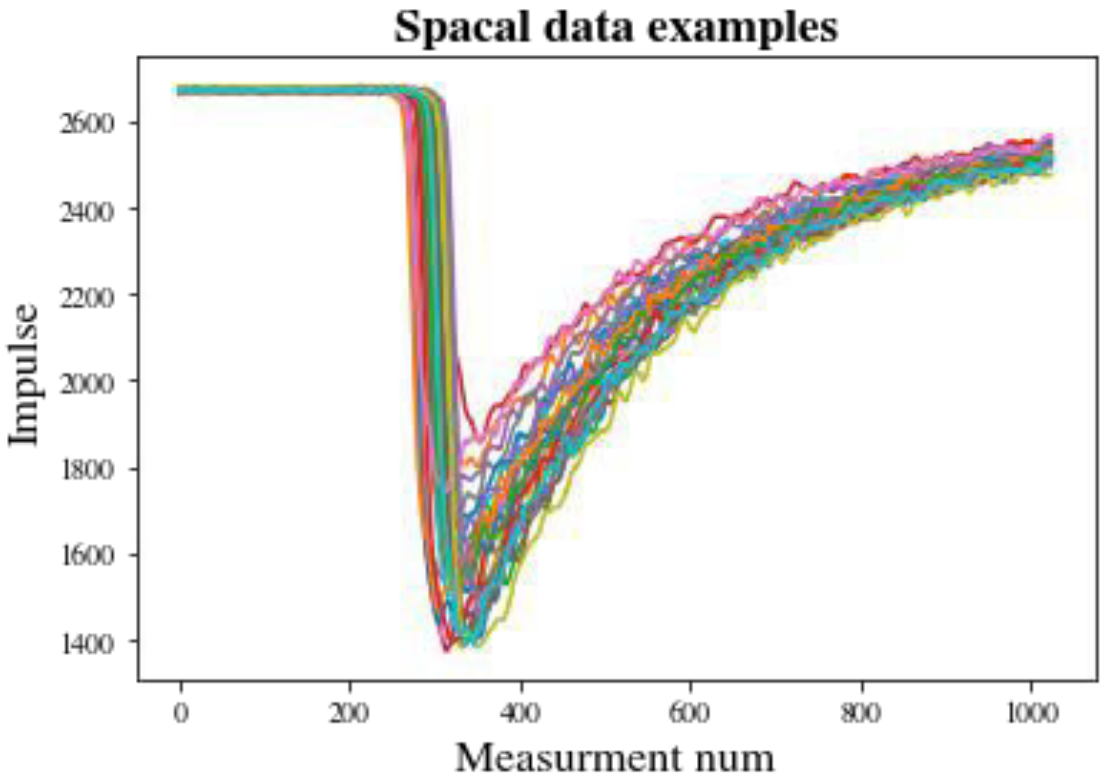


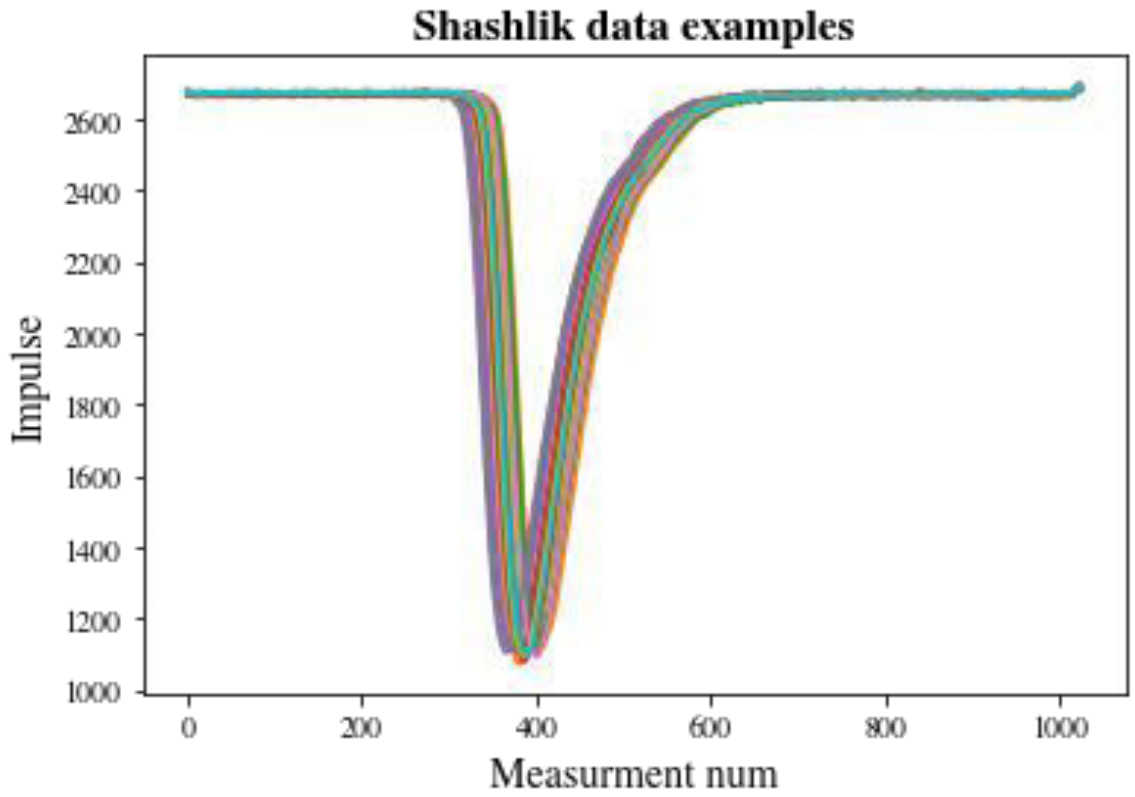


What do we want?

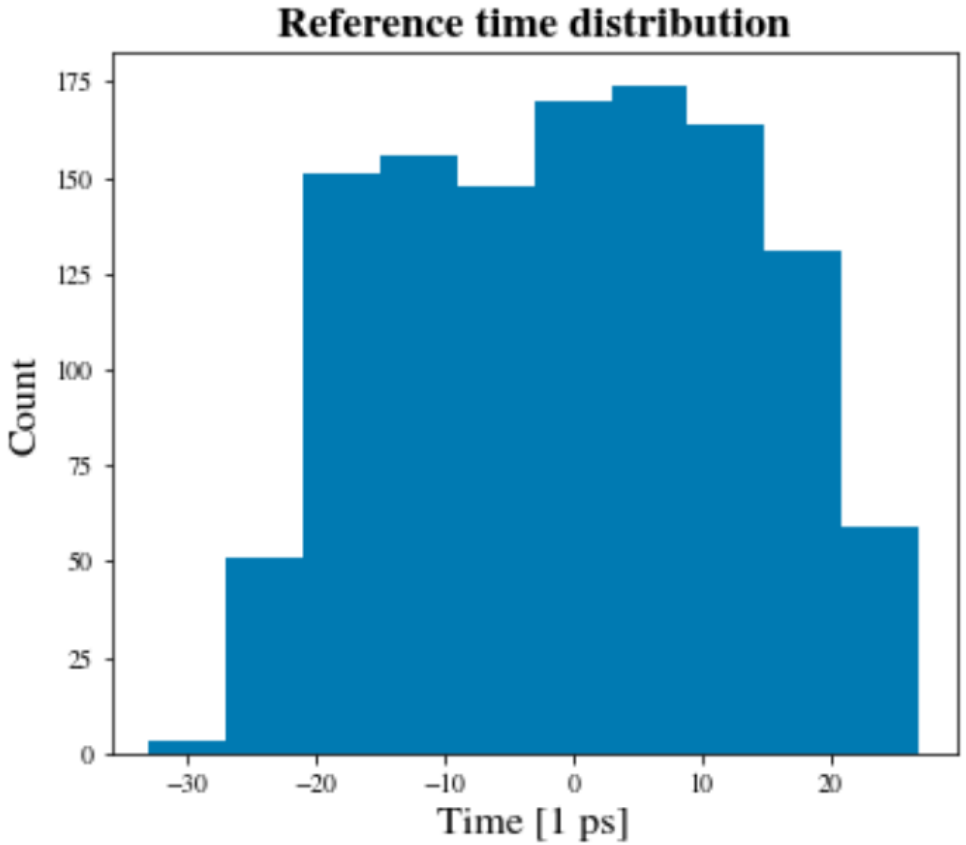
- Explore possible quality of signal recovery for different sampling rates
- Explore the ability to distinguish between single signal and two close signals for different sampling rates
- Explore the ability to determine the time and amplitude of a signal in the presence of a second close signal for different sampling rates

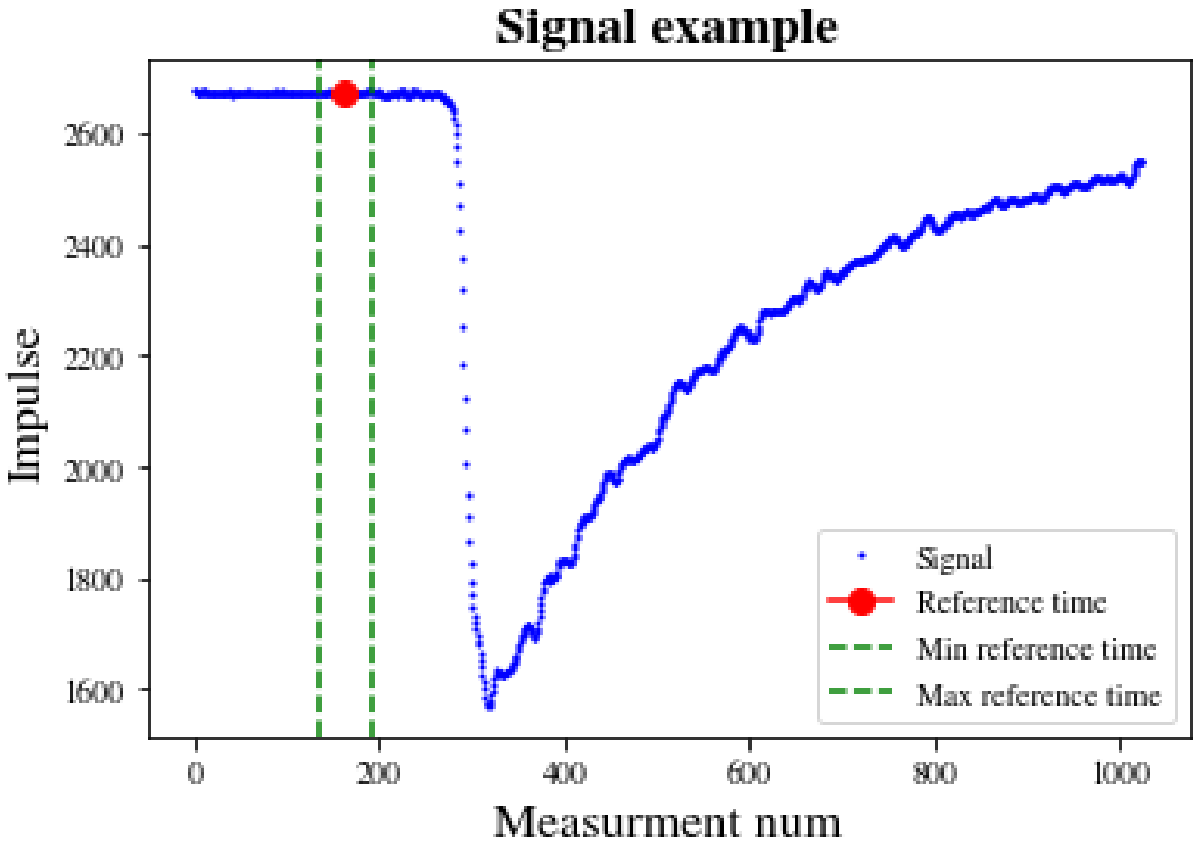
Spacal vs Shahlik data



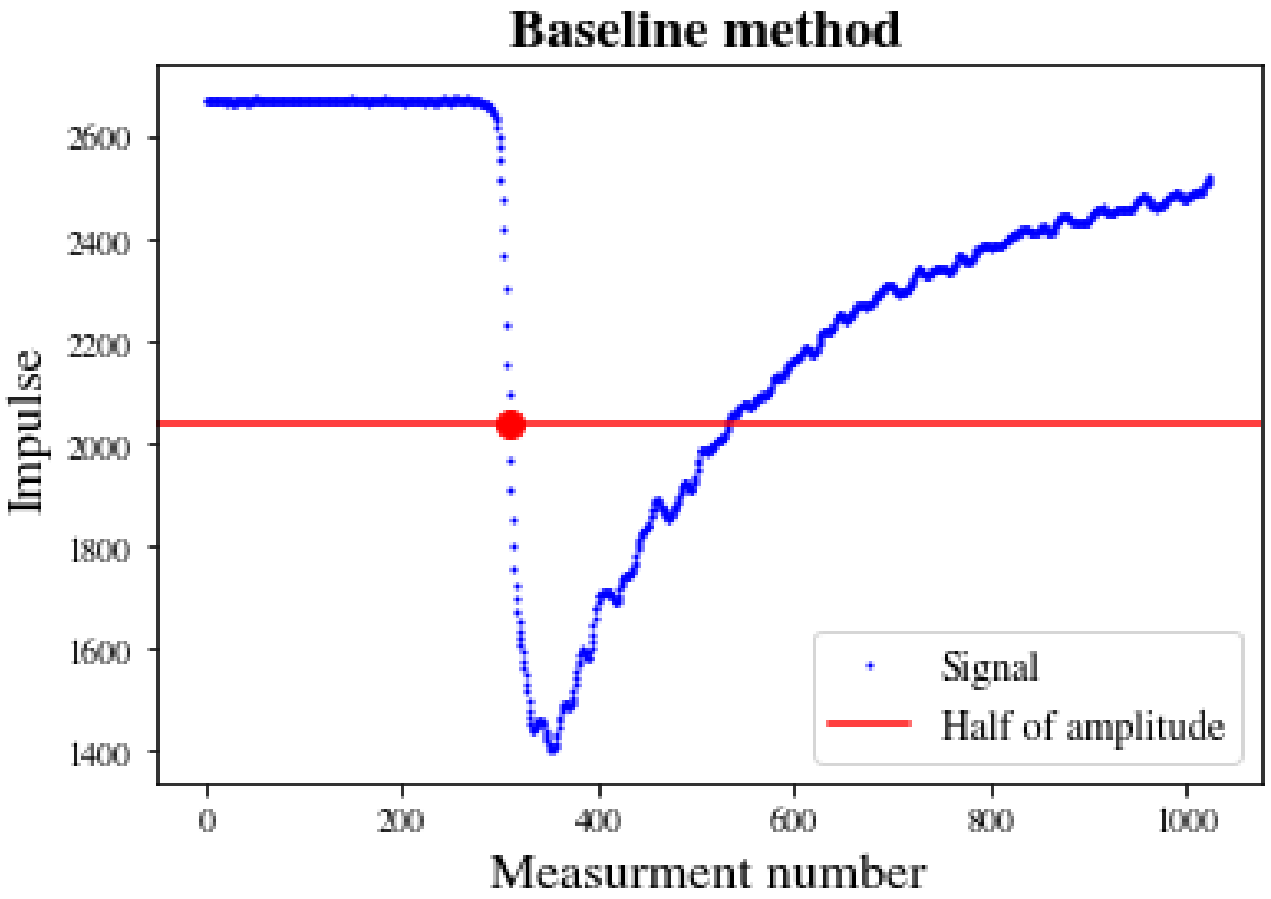


Problem 1. Predict reference time for a signal





Baseline method description

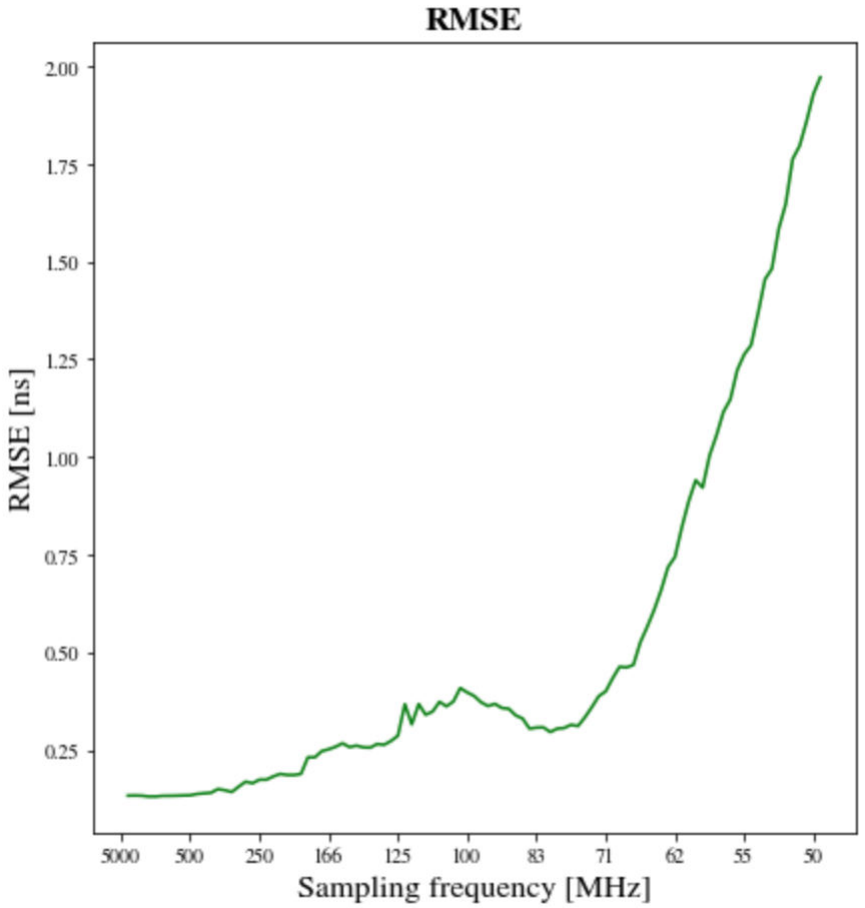



```
Baseline metric:
RMSE[ns]: 0.09752
```

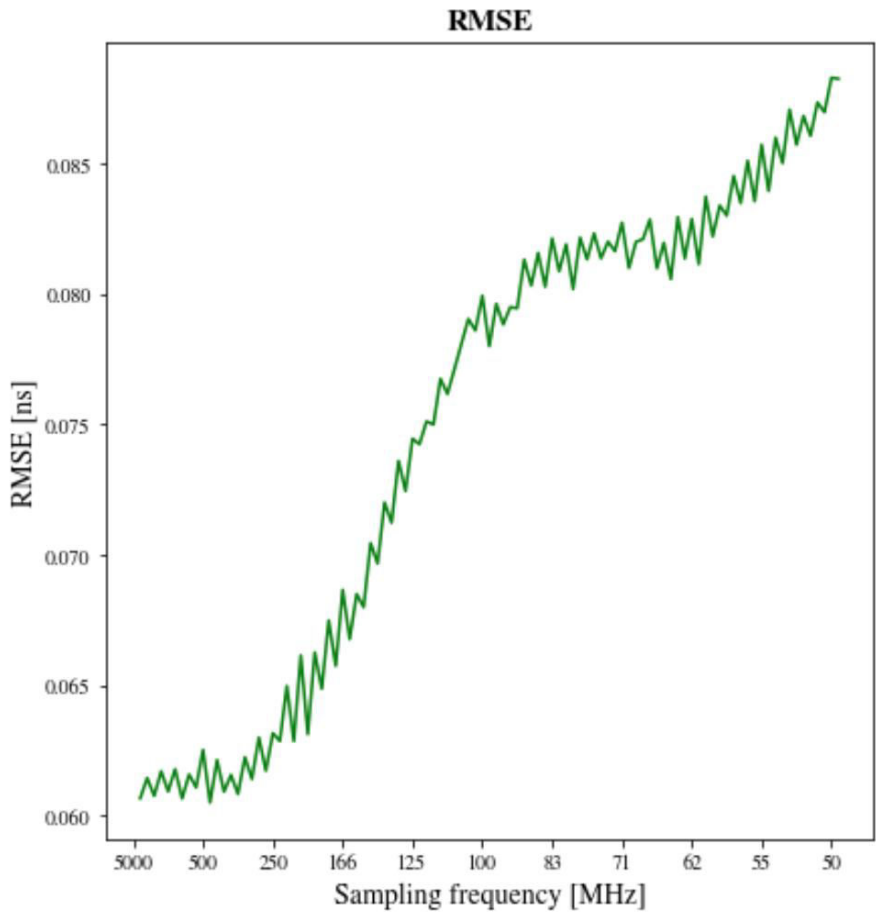
As we can see even baseline solution gives us excellent metrics

Scores dependency on sampling frequency

Spacal data

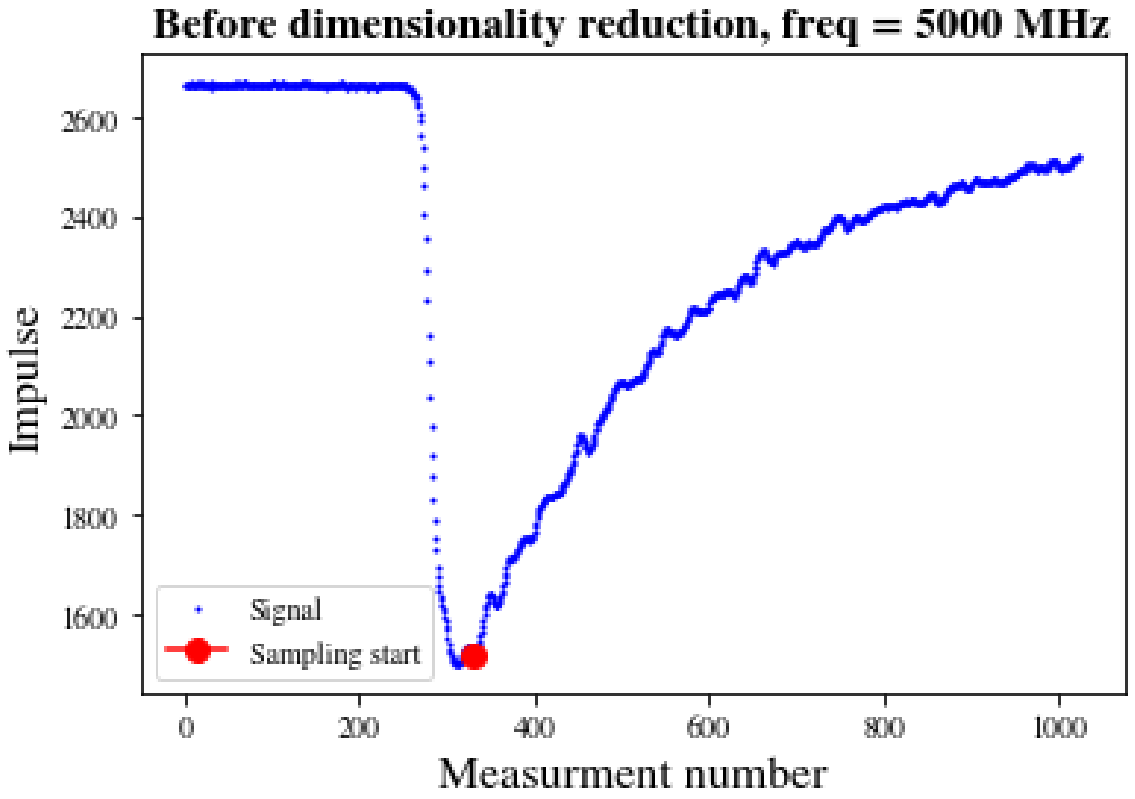


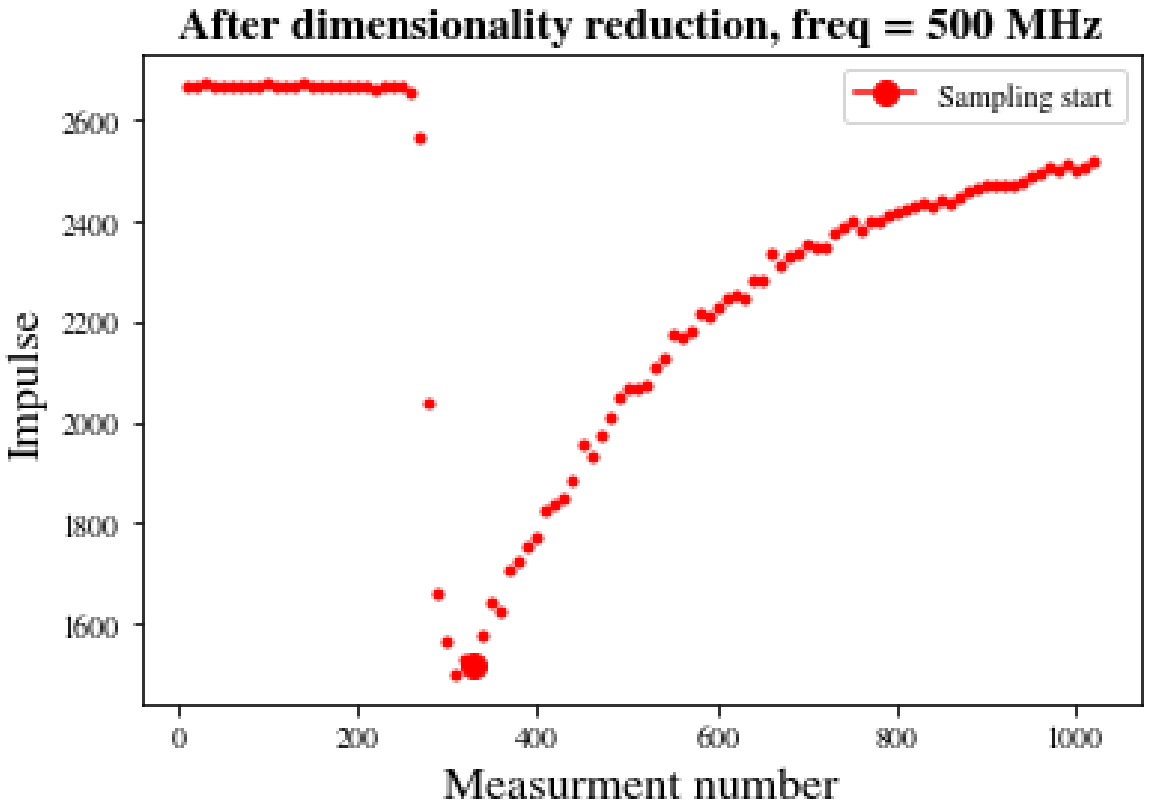
Shashlik data



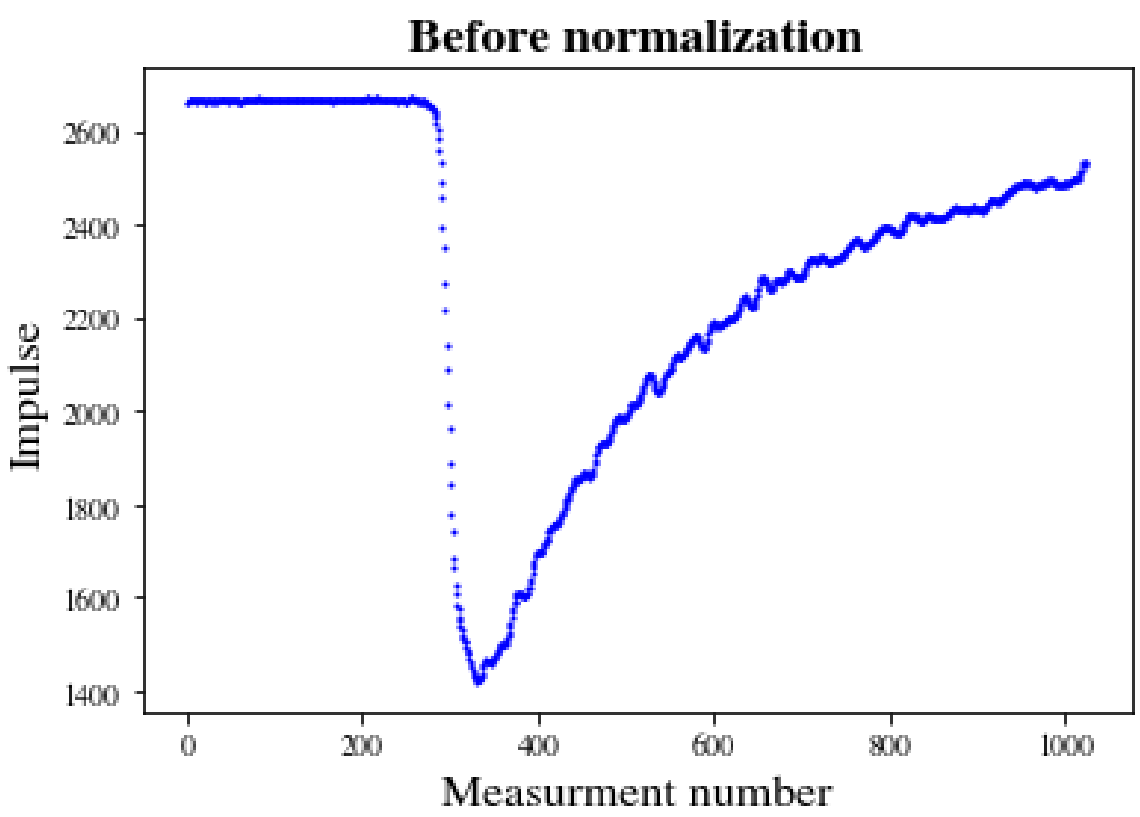
Xgboost is used as a model

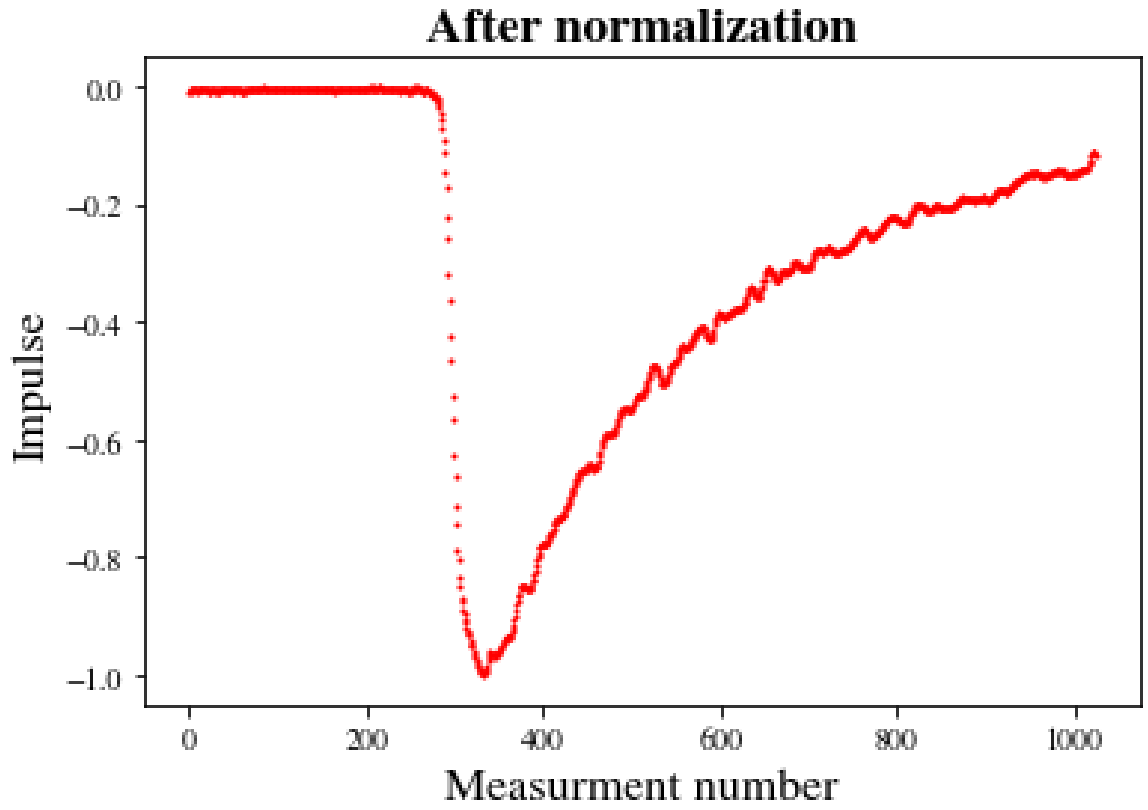
Reducing sampling frequency





Preprocessing data





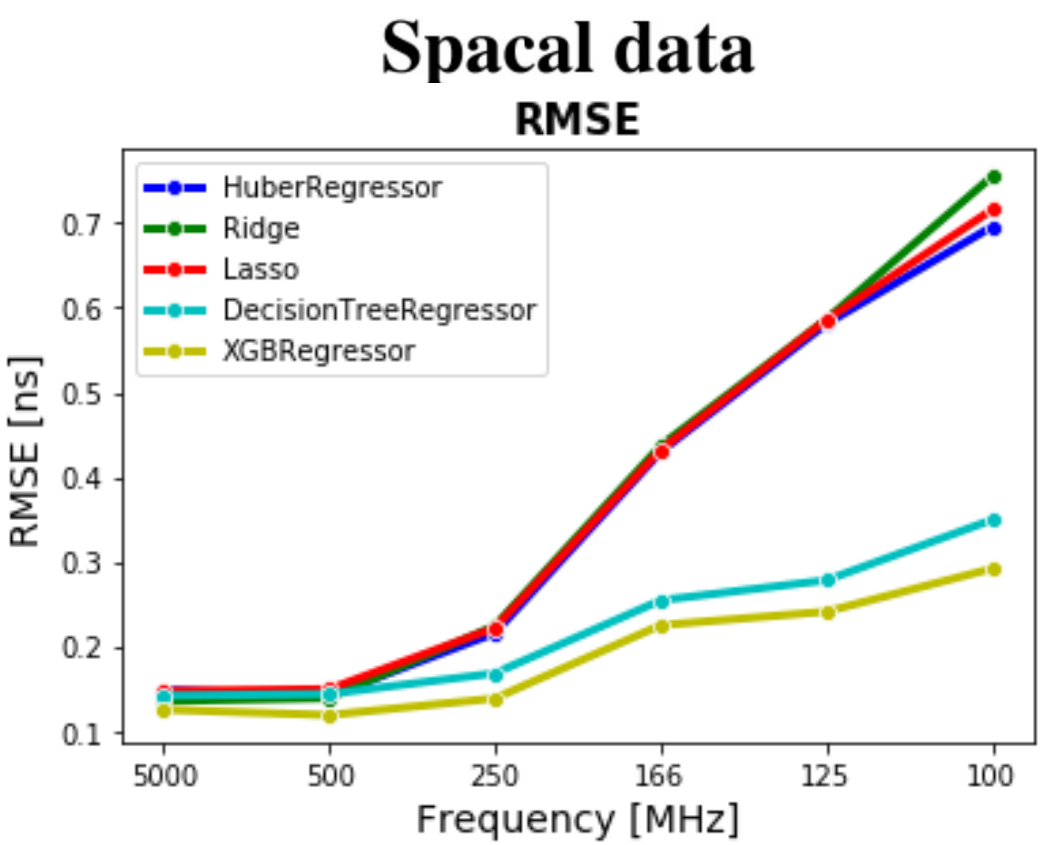
Model tuning

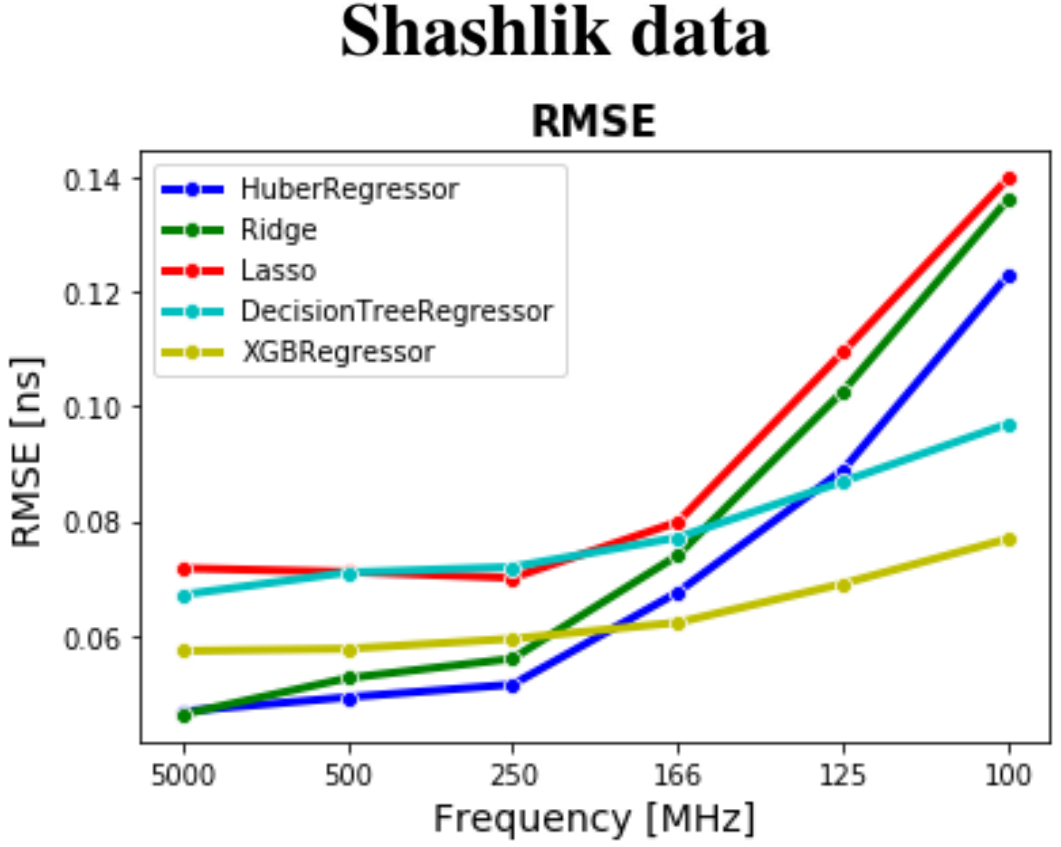
```
space = {
    'booster': hp.choice('booster', ['gbtree', 'gblinear', 'dart']),
    'eta': hp.loguniform('eta', low=np.log(0.001), high=np.log(1)),
    'gamma': hp.loguniform('gamma', low=np.log(0.001), high=np.log(100)),
    'max_depth': ho_scope.int(hp.quniform('max_depth', low=5, high=50, q=2)),
    'lambda': hp.loguniform('lambda', low=np.log(0.001), high=np.log(10)),
    'alpha':  hp.loguniform('alpha', low=np.log(0.001), high=np.log(10)),
}
```

XGB Regressor:
95% confidence interval:
RMSE[ns]: 0.12482 (+/- 0.03096)

Baseline metric:
RMSE[ns]: 0.09752

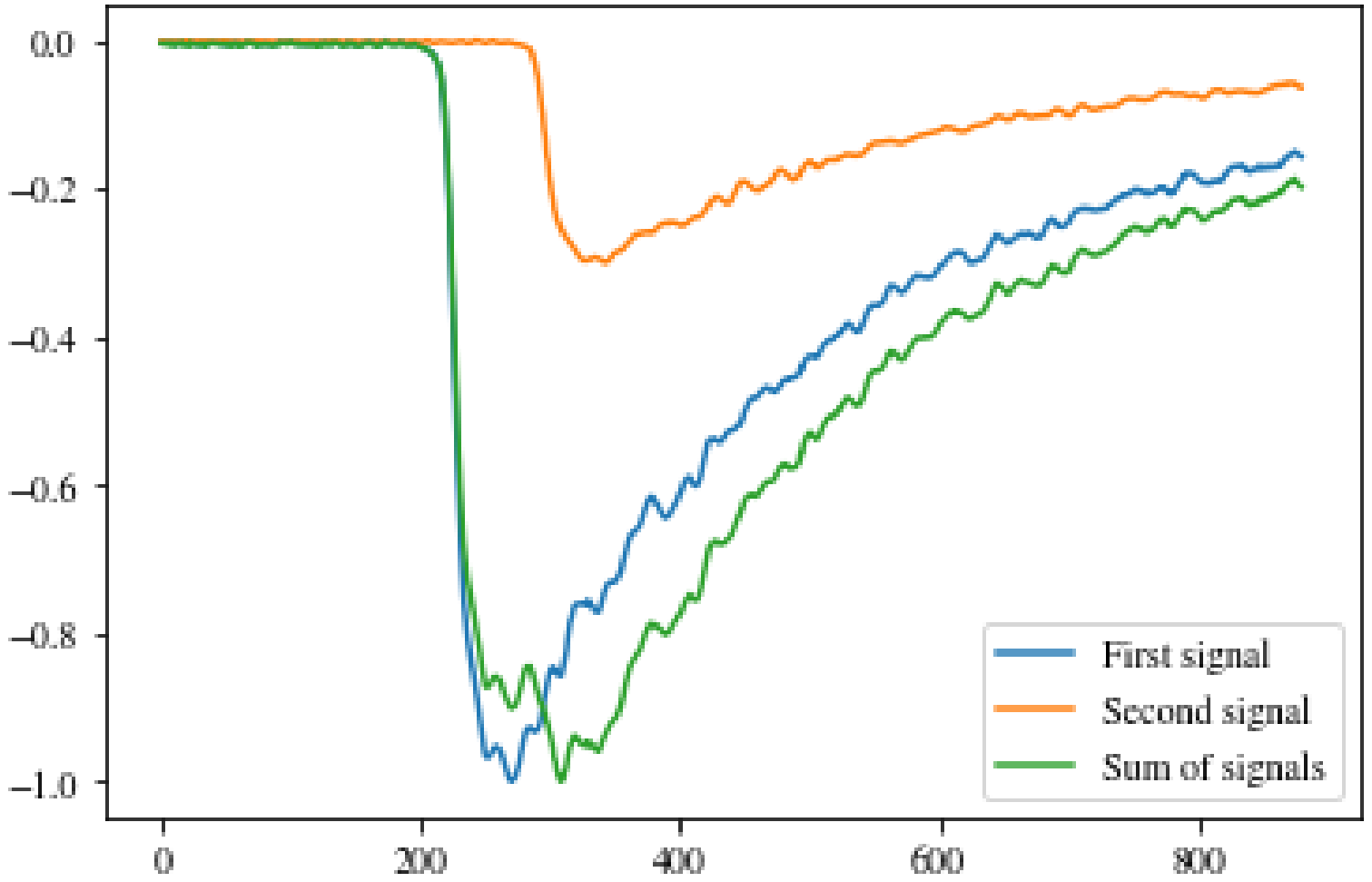
Models comparison



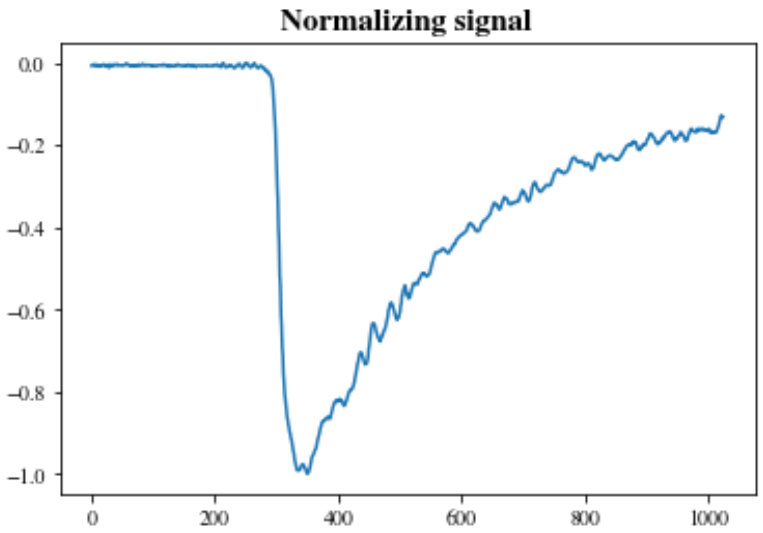


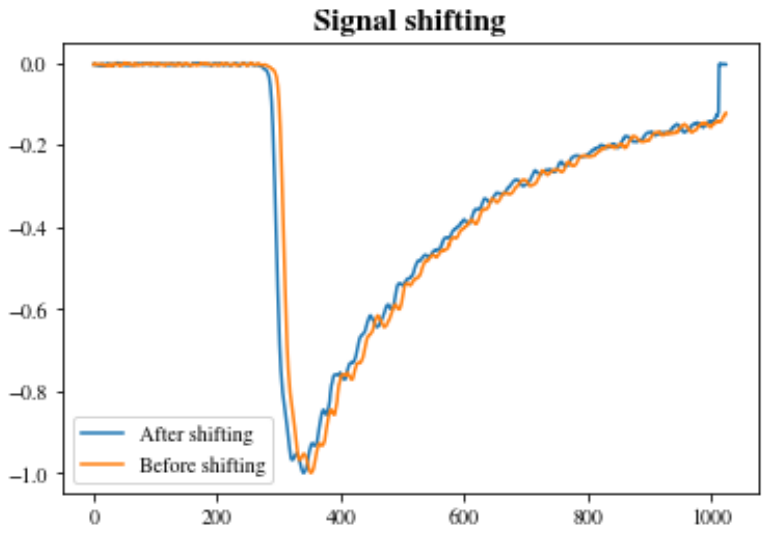
It illustrates that we can reduce the sampling frequency at least by 10 times!

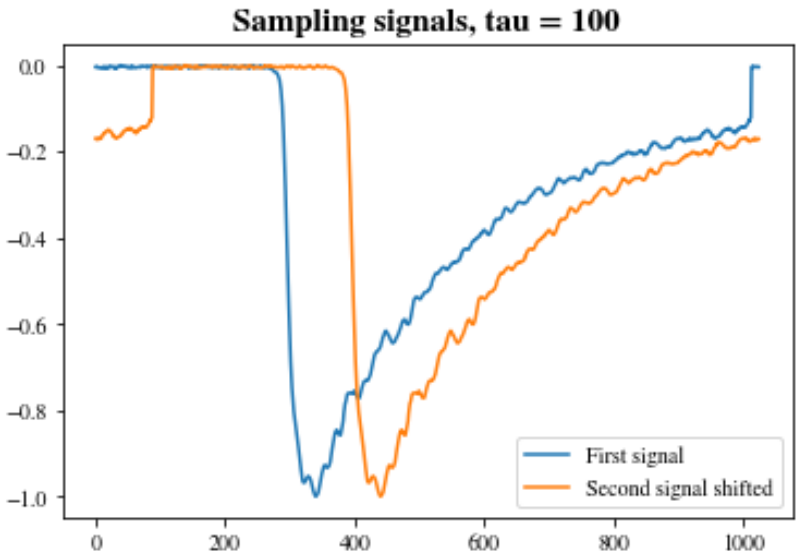
What if there are two signals?

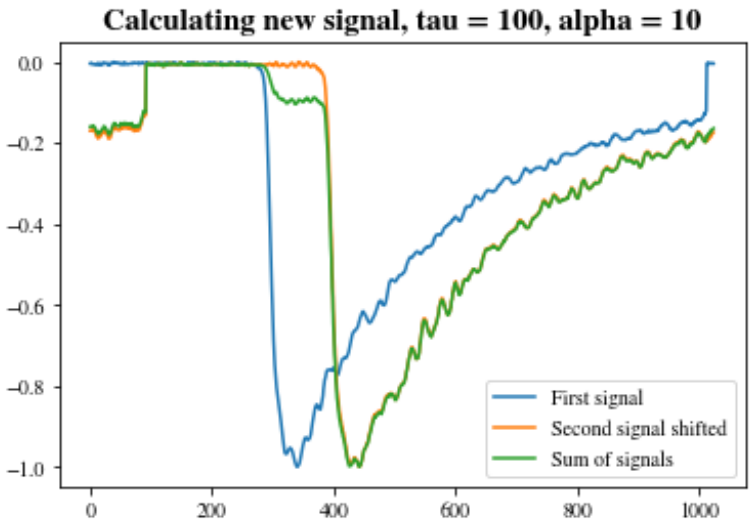


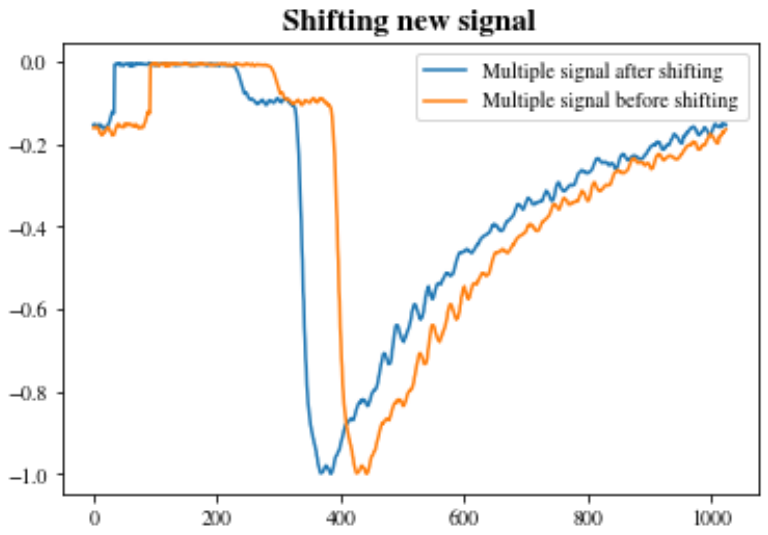
A few words about data preparation

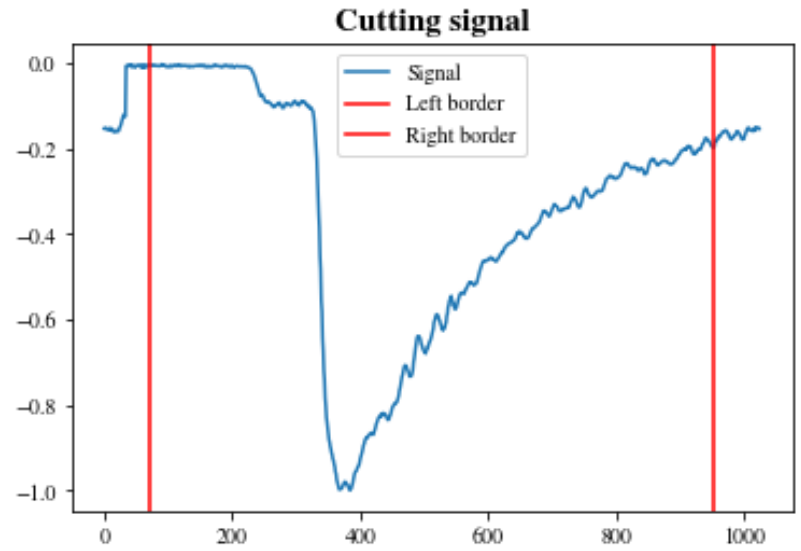




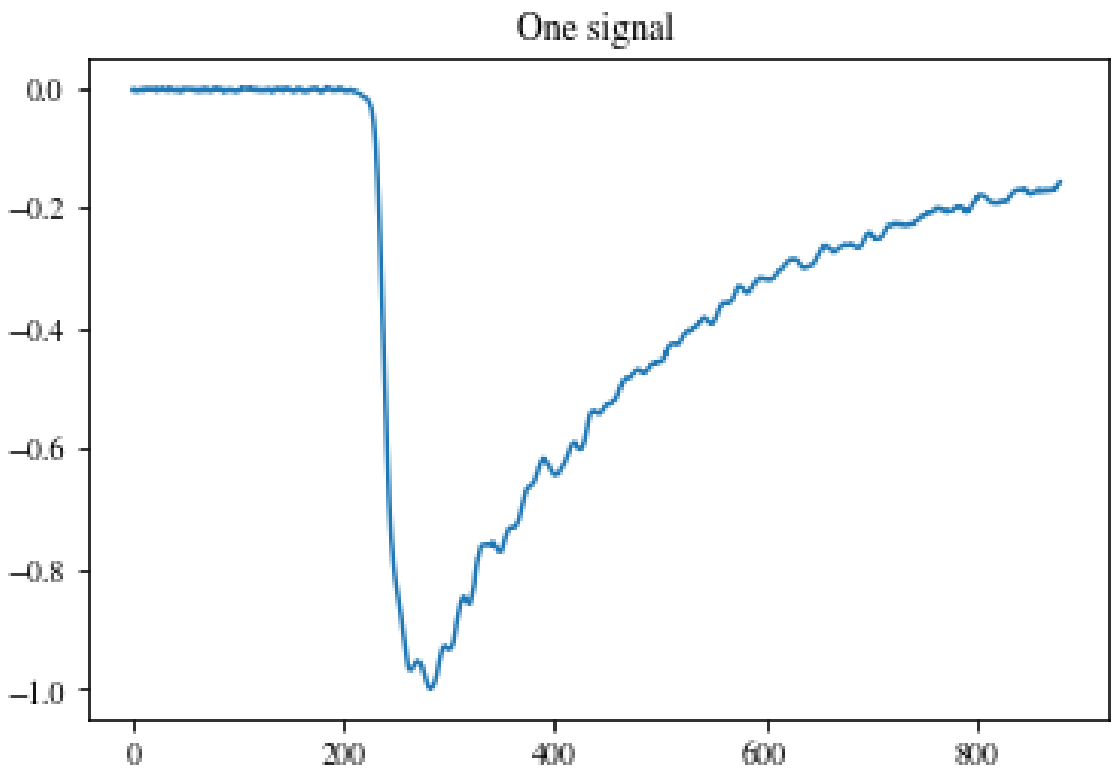




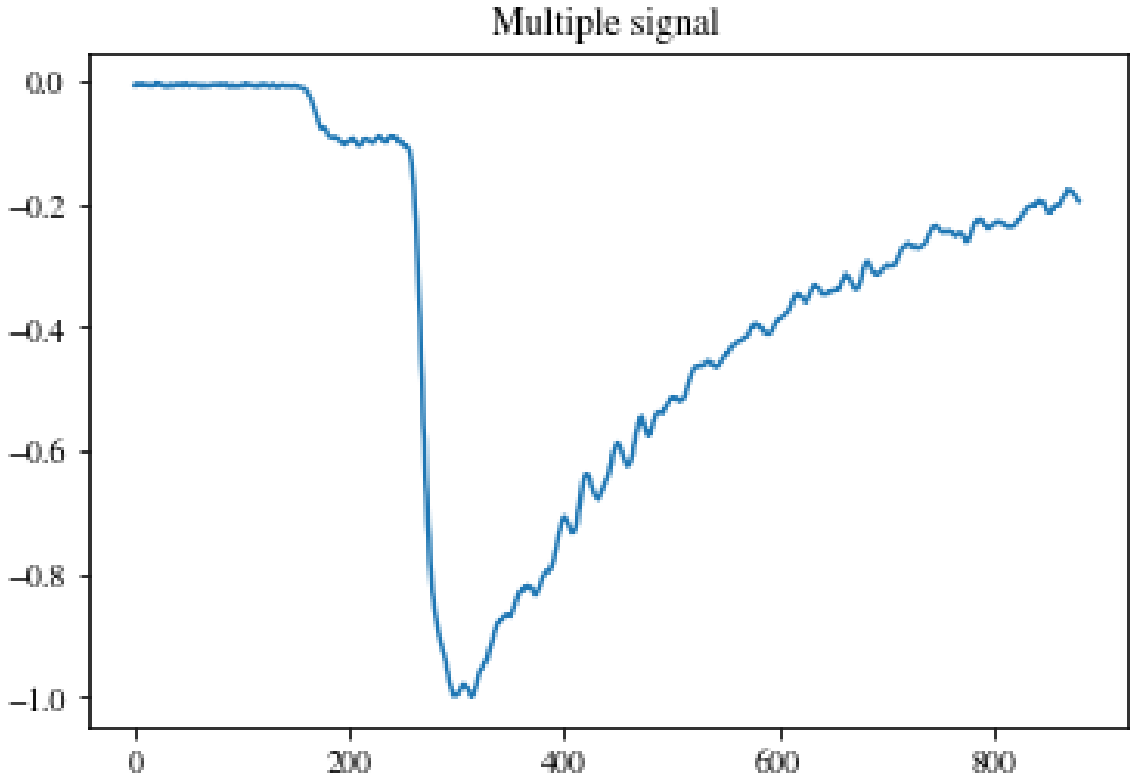




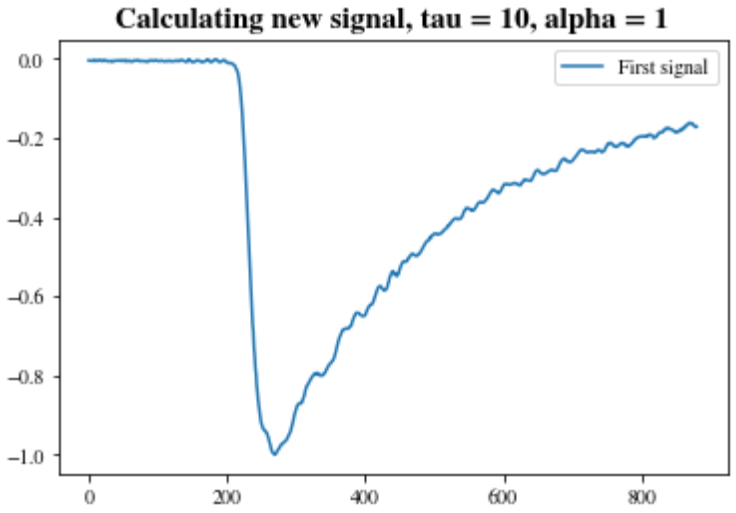
Problem 2.
Identify the presence of two signals

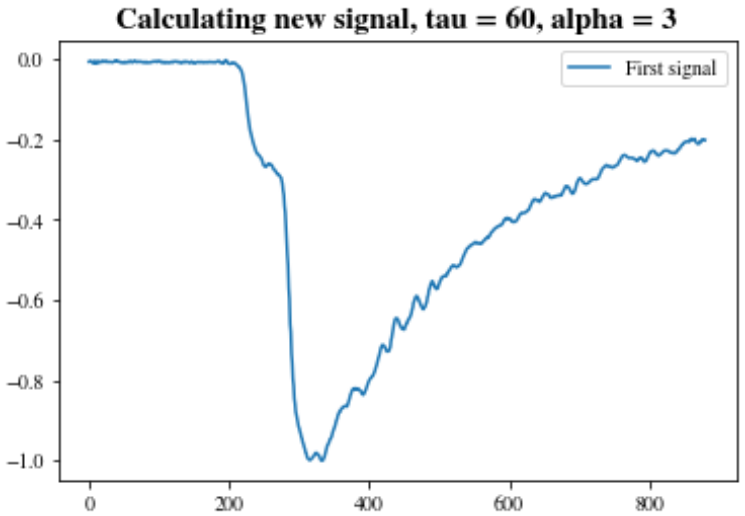


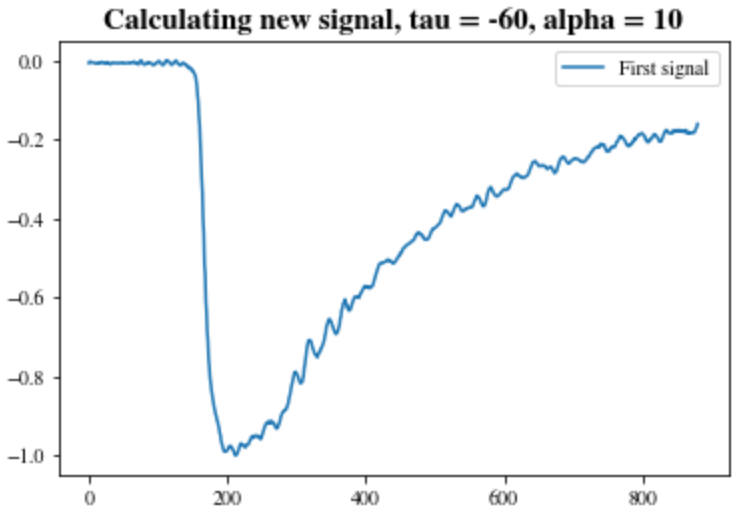
vs

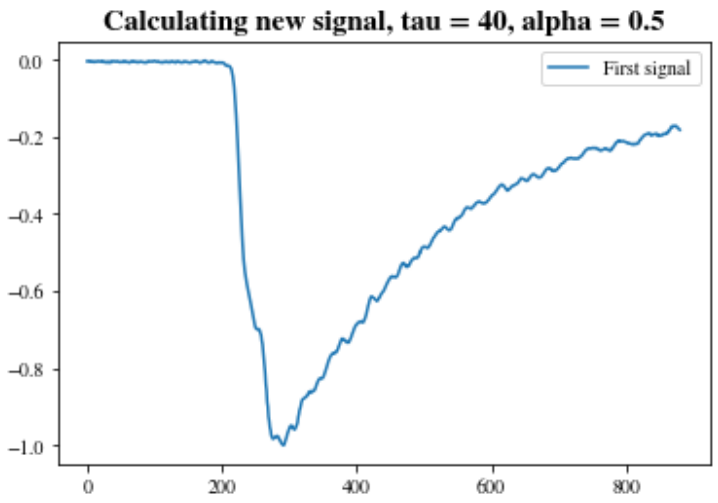


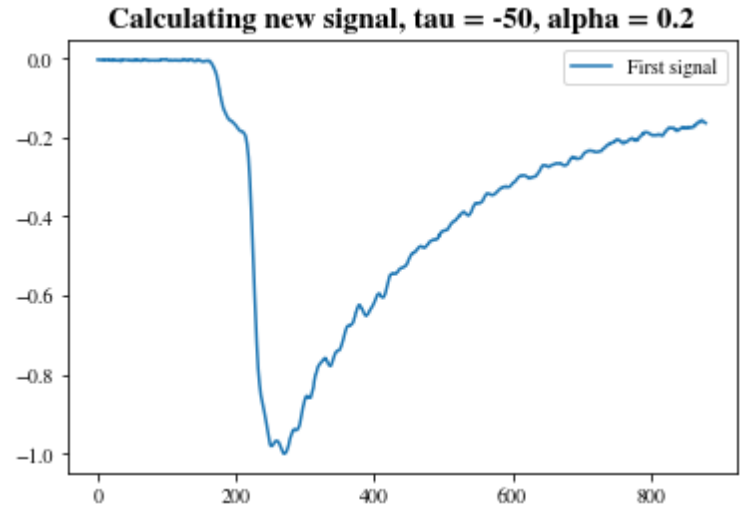
Problem 2. Examples of two signals

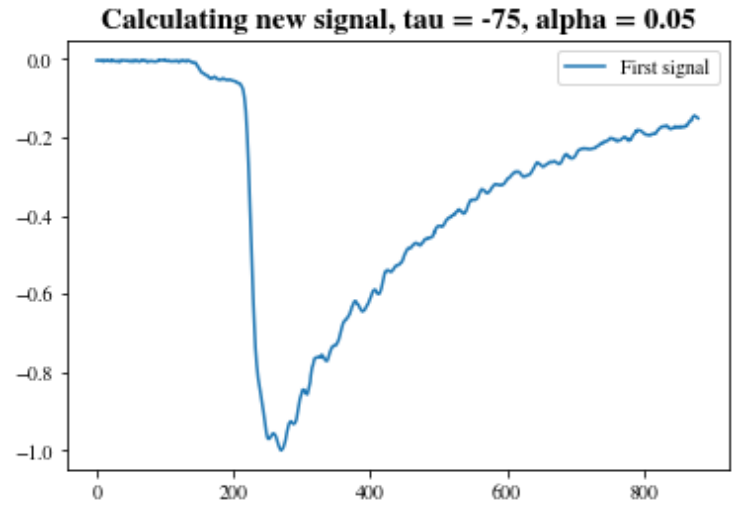




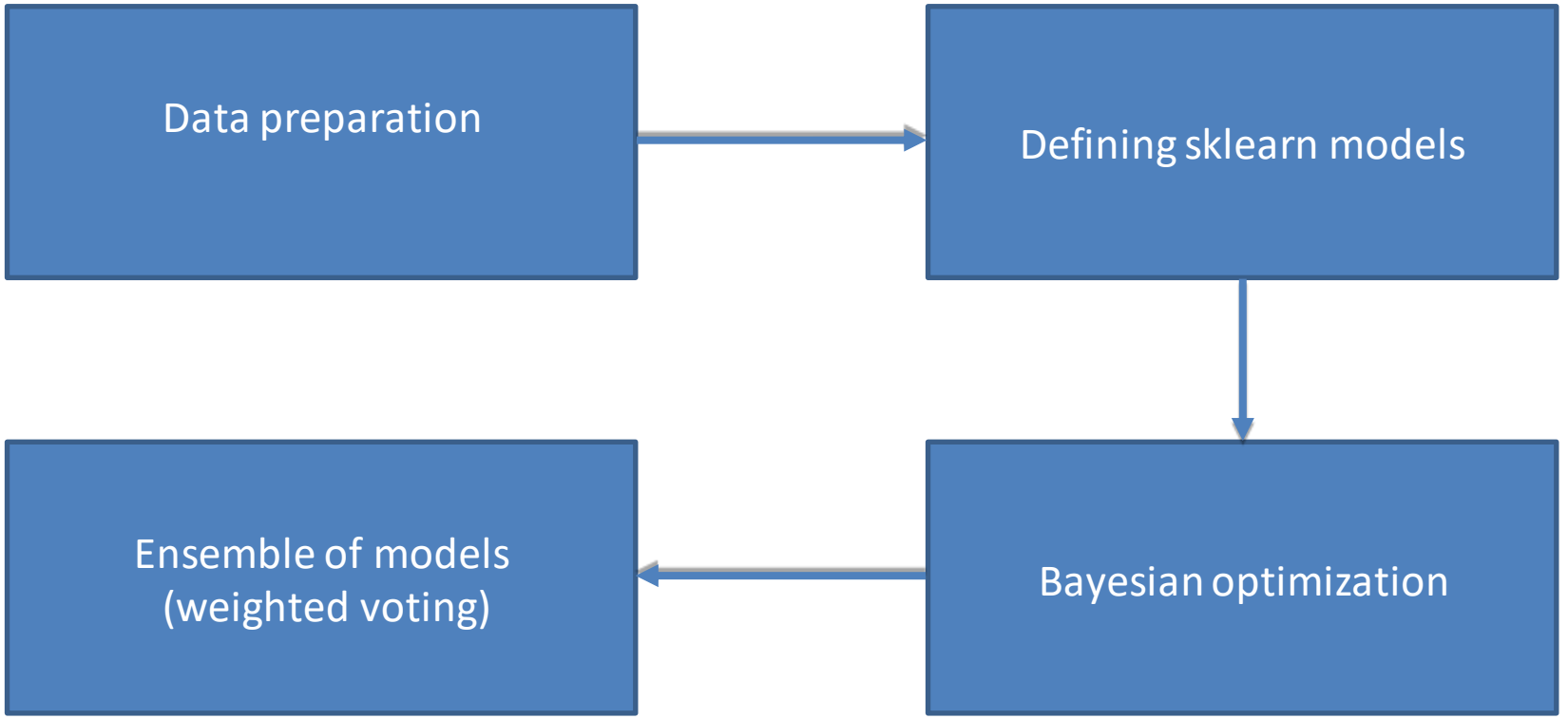








Problem 2. Pipeline



We tune our models separately for each sampling rate

How to evaluate our results?

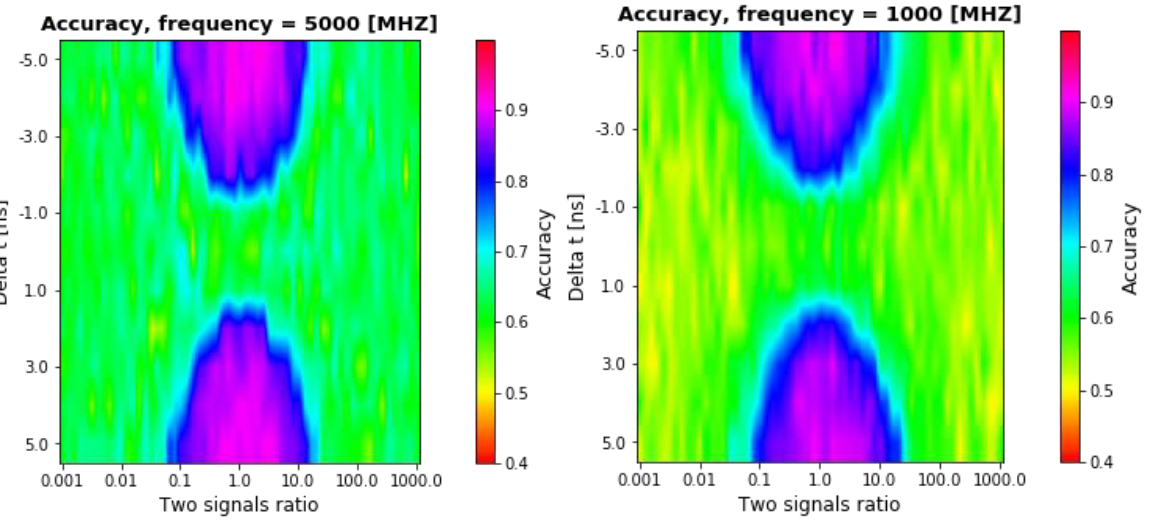
Confusion Matrix

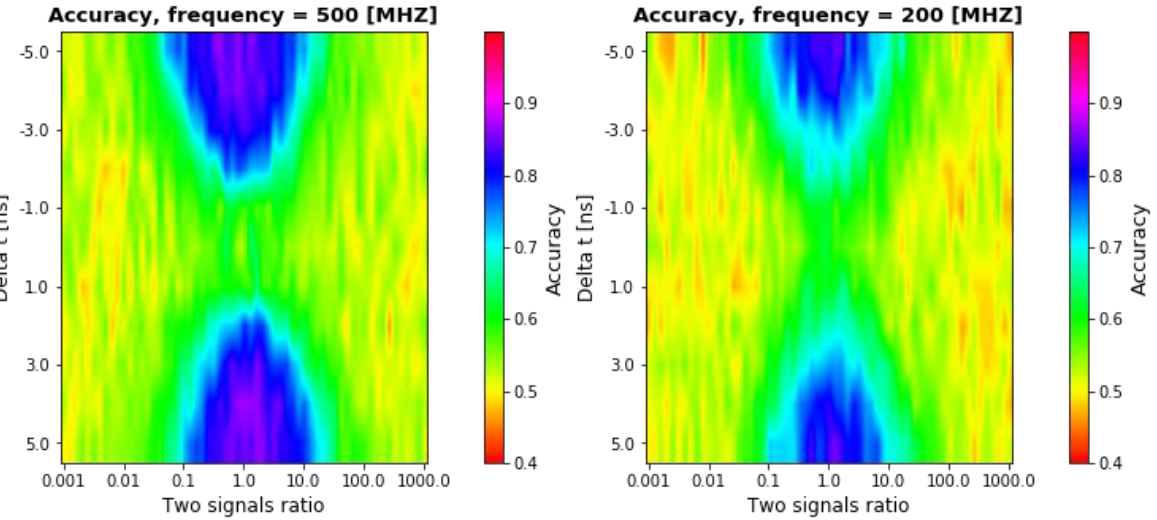
	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

$$\textbf{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

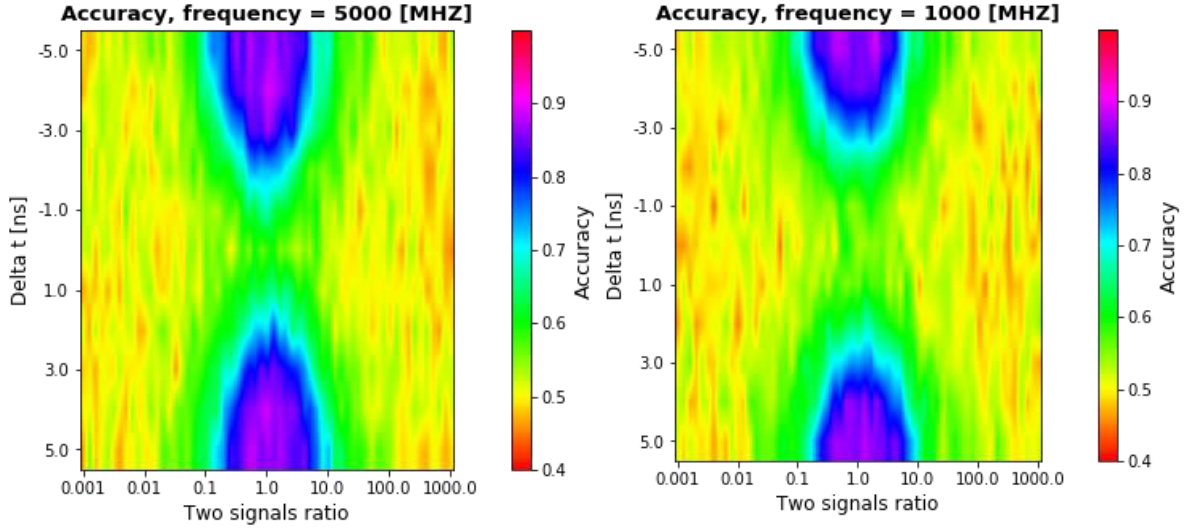
Problem 2. Results comparison

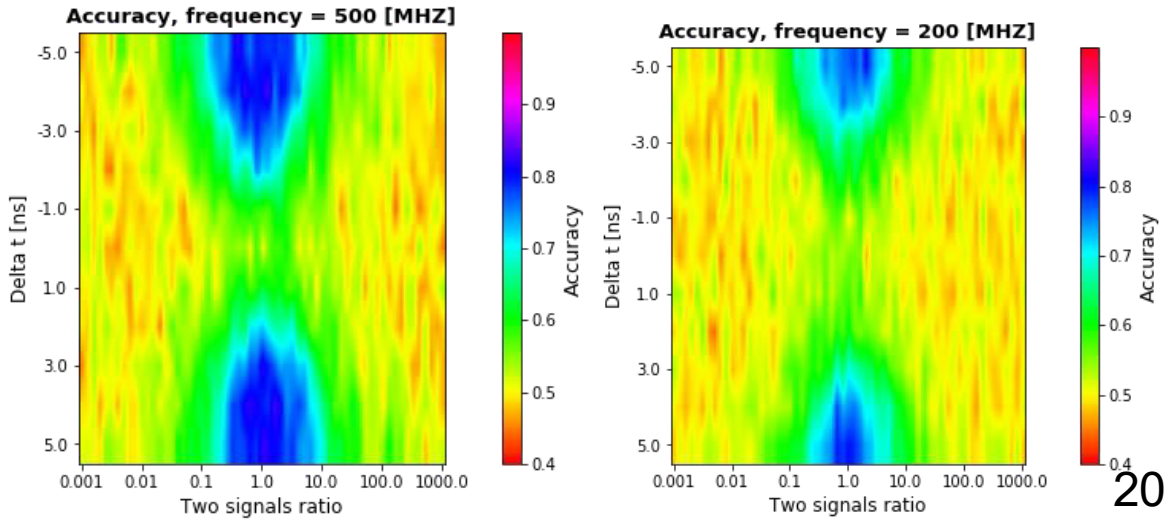
Shashlik data



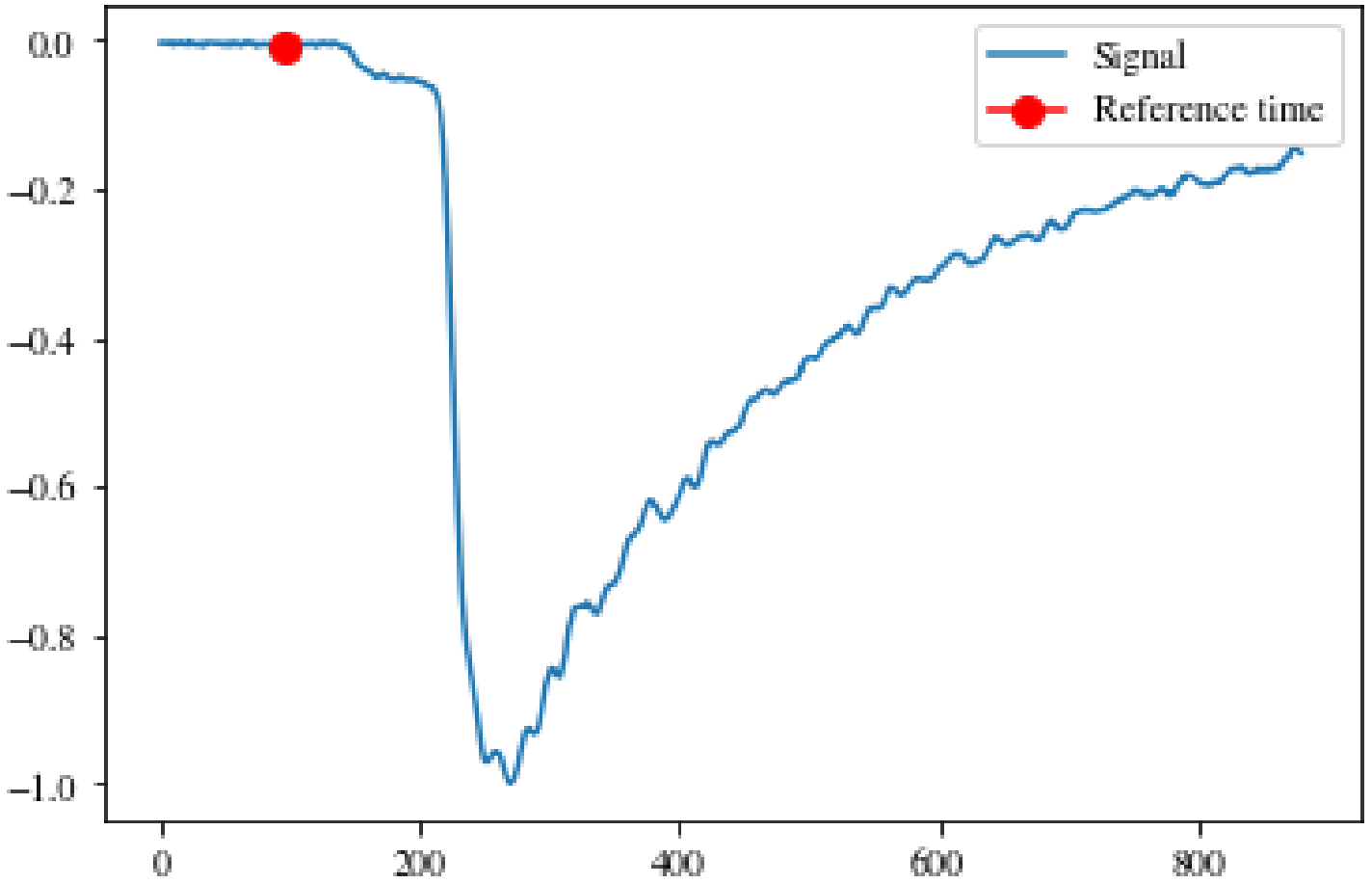


Spacal data





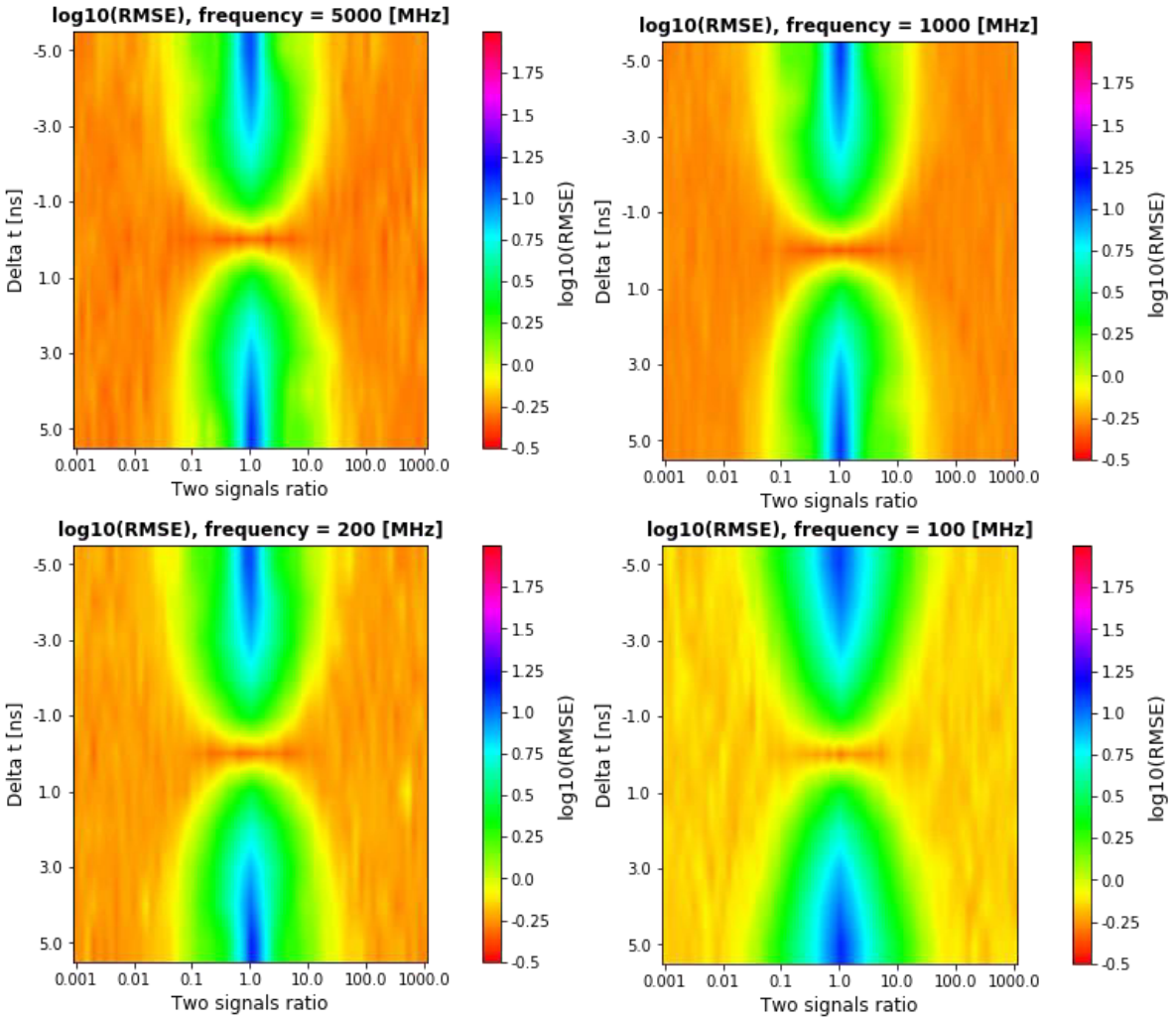
Problem 3.
Predict the reference time of a multiple signal



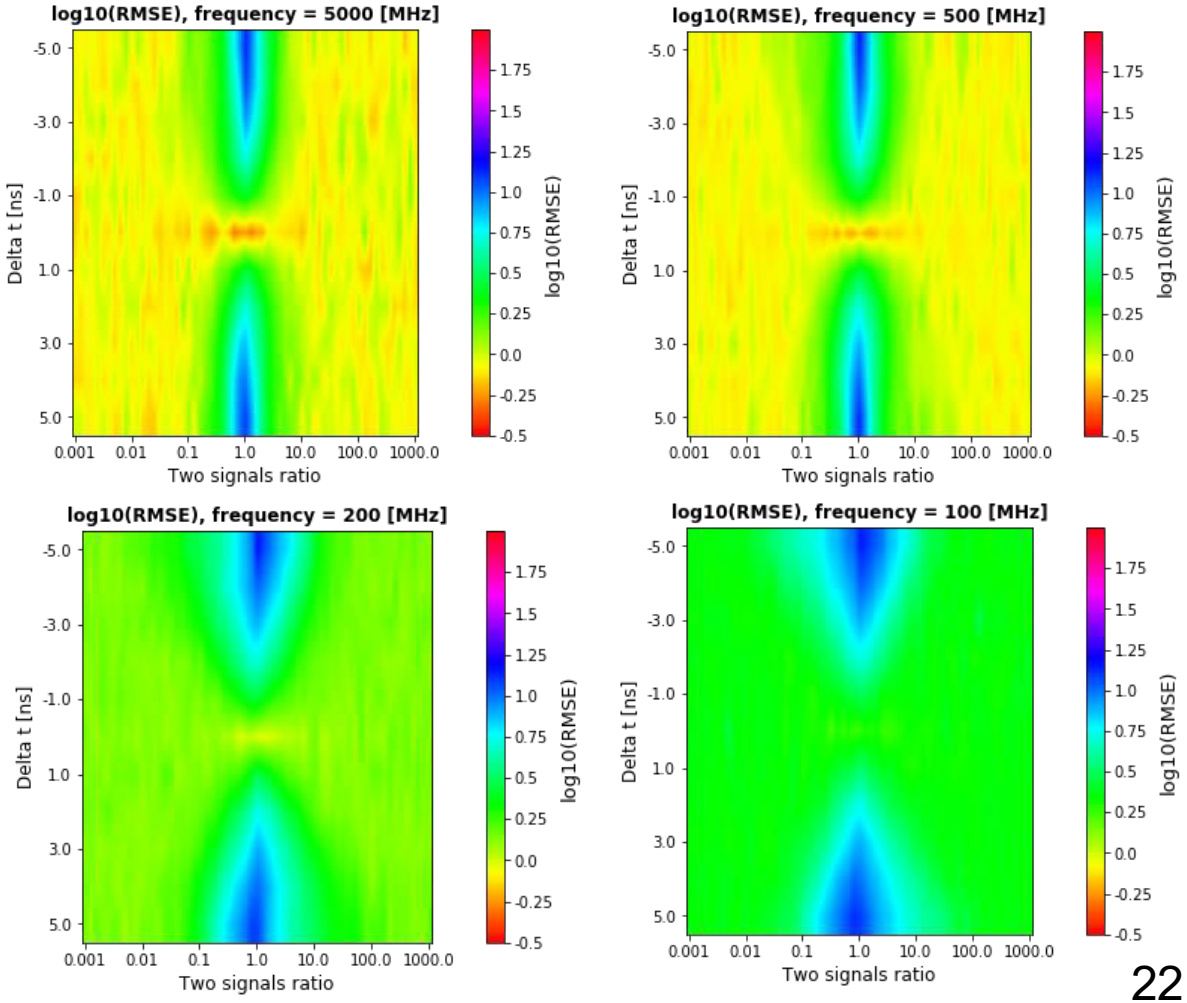
We will predict the reference time of a more significant signal

Problem 3. Results comparison

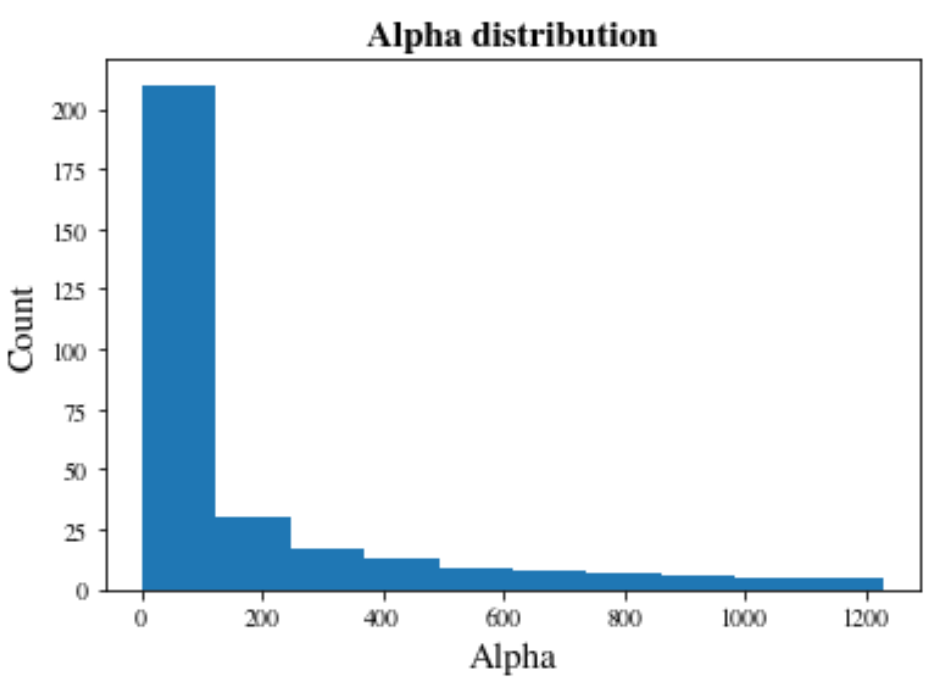
Shashlik data

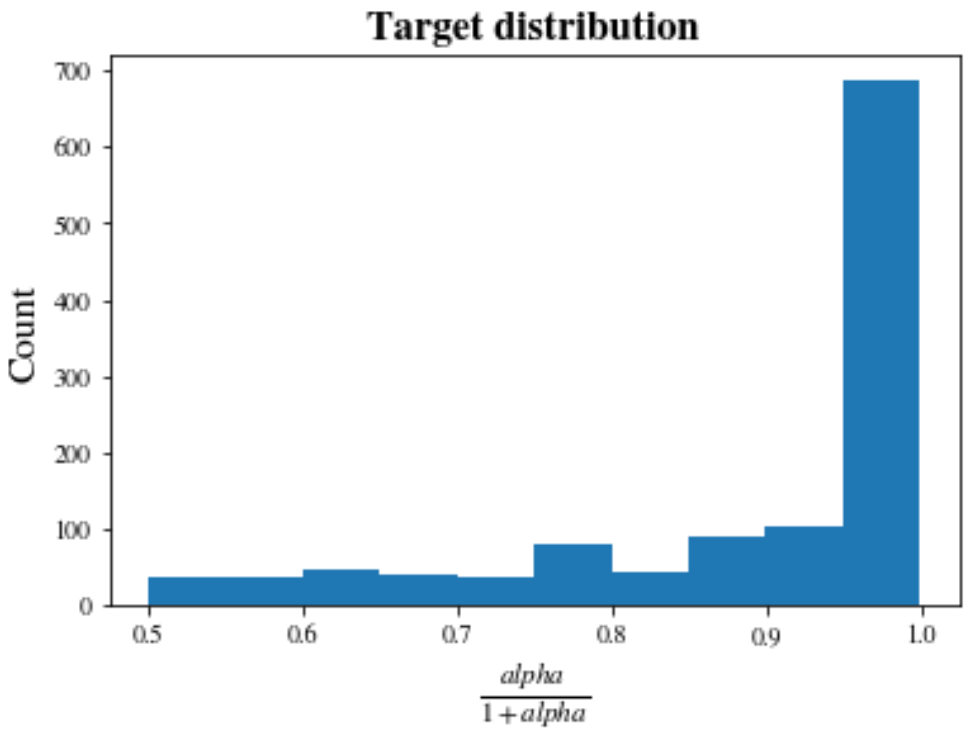


Spacal data

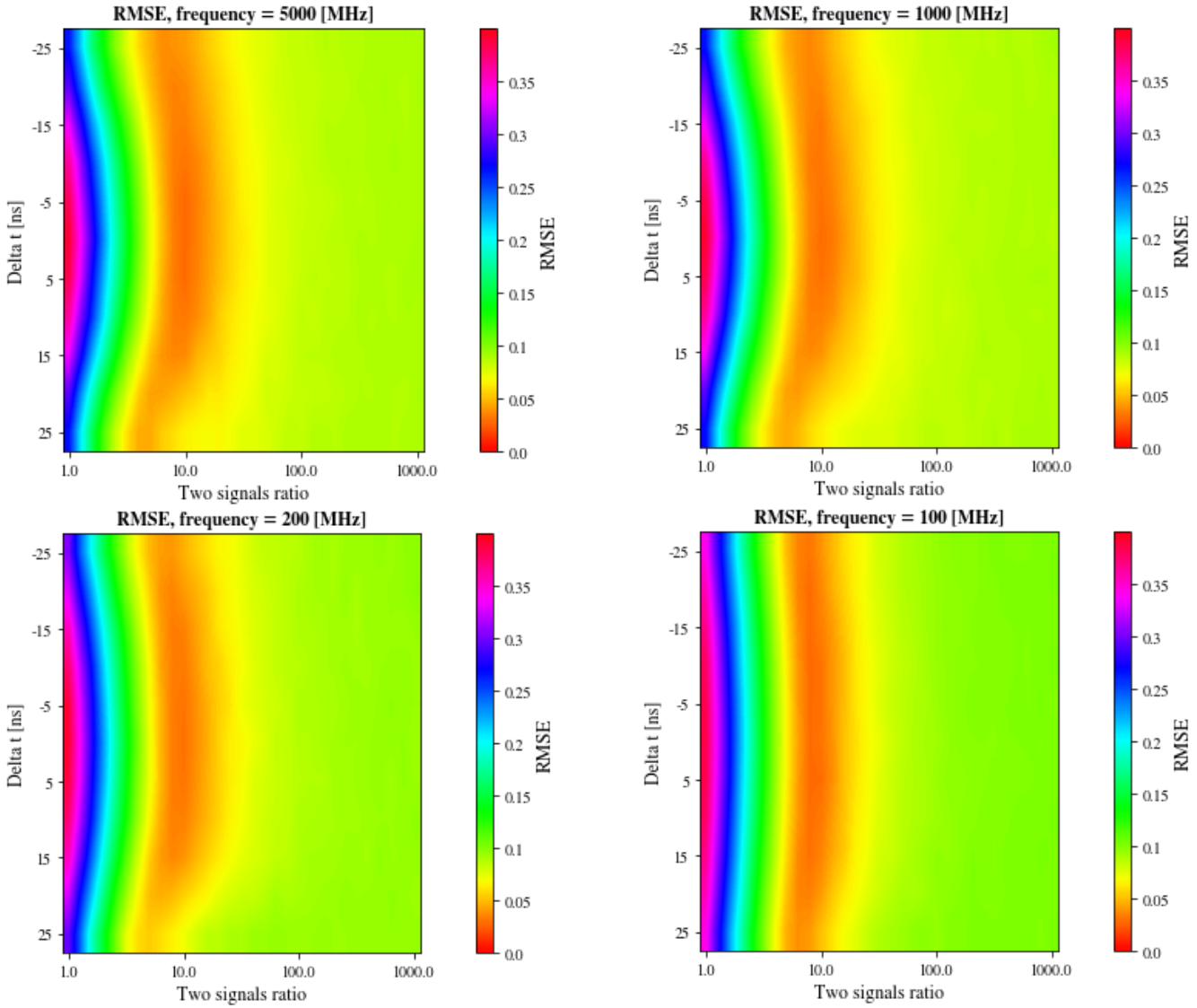


Problem 4. Predict the amplitudes ratio



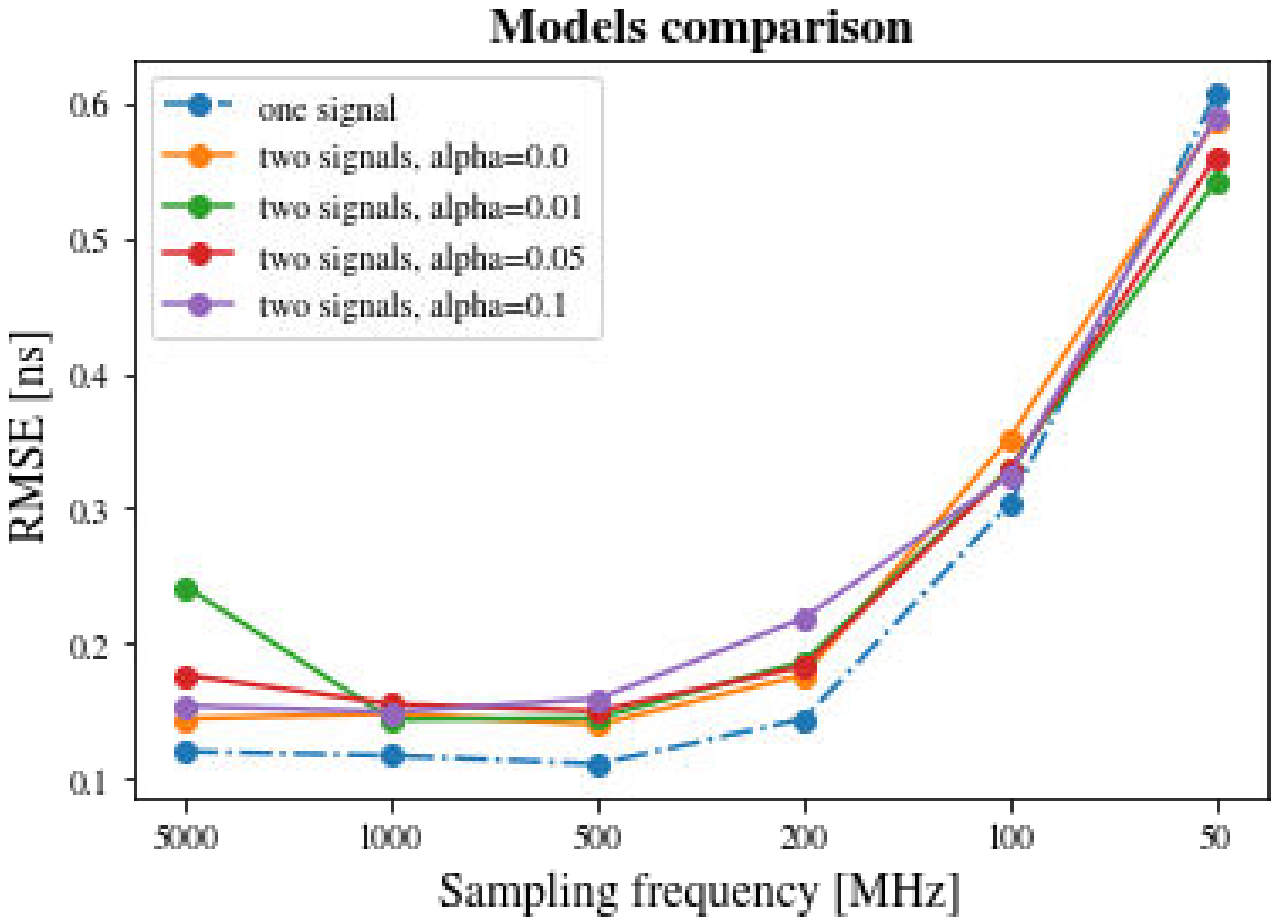


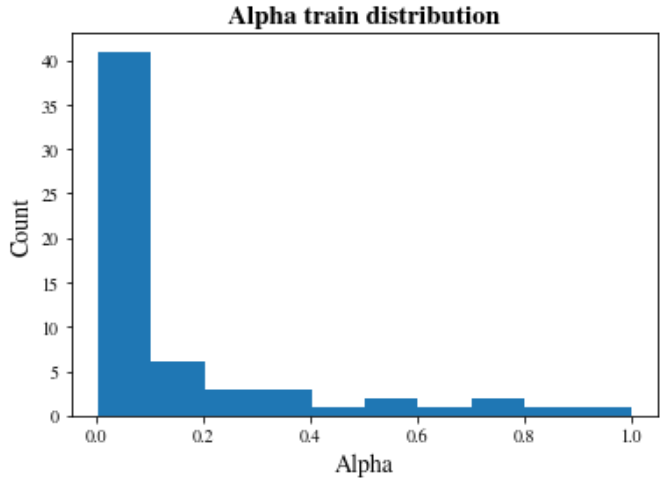
Problem 4. Results

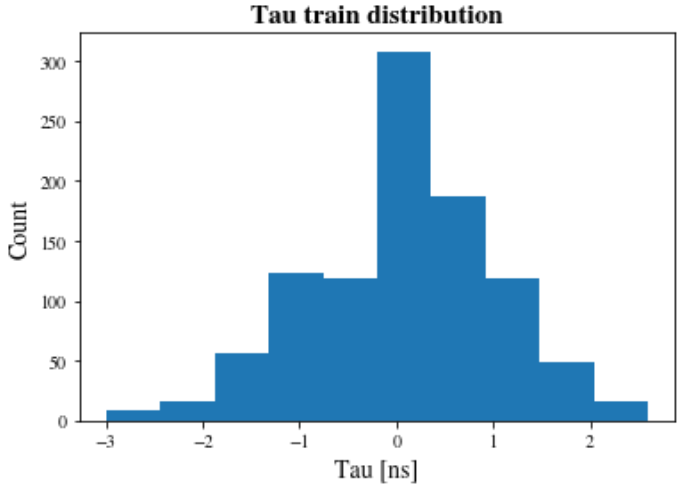


In terms of a few ns we may see that timing property doesn't really affect the quality

Comparing the scores of a model for one signal and a model for two signals







Conclusions

- Spacal timing properties are much noisier comparing to shashlik data
- Using signals obtained in test beam measurements we estimated
  - Effect of signal sampling rate on the timing resolution
  - Efficiency to identify presence of another particle in the signal
  - Disturbing of time measurements due to another particle contribution in the signal
- Obtained results may be plugged into the physics simulation to evaluate effects of higher occupancies on ECAL reconstruction

Any questions?

Mail: ***pavel.fakanov@gmail.com***

Github: ***https://github.com/whiteRa2bit/cern_summer_project***