**Journal Report 19**

**01/19/2024**

In class, we had presentations all the way till 1/31/24. As a result, I spent at least 1 hour on days that we usually have class on the project. I researched different preprocessing techniques as more advanced models forecast variables with a significant amount of noise. The two processing techniques I found to be promising are the Fourier transforms and Gaussian smoothing.

**01/20/2024**

I trained a RandomForest Model in order to compare my results with the XGBModel. These are the hyperparameter values.

model = RandomForest(

lags=40,

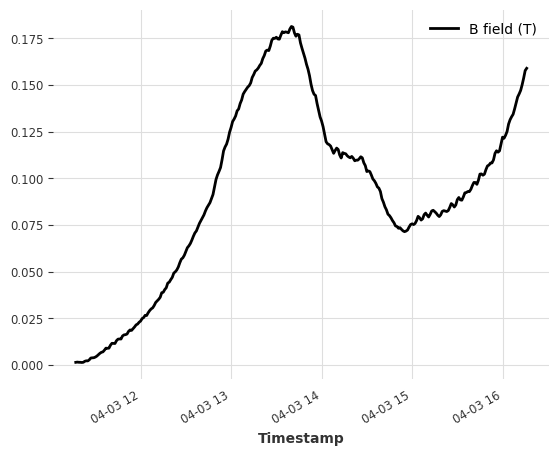
lags\_past\_covariates=40,

lags\_future\_covariates=[0,1,2,3,4,5],

output\_chunk\_length=300,

)

These are arbitrary values that I randomly set and am looking for simple hyperparameter optimization algorithms. The result for this model on a forecast of 300 timestamps is shown below. There is significantly less noise compared to the XKBModel and seems to be following the general pattern of the actual data.



**01/22/2024**

I tried training a TemporalFusionTransform (TFT) Model in order to compare my results with the XGBModel and RandomForest. These are the hyperparameter values.

model = TFTModel(

input\_chunk\_length=100,

output\_chunk\_length=300,

n\_epochs=2,

)

I set the hyperparameters arbitrarily again and am still searching for hyperparameter optimization algorithms. However, I ran into an error:

**NotImplementedError: The operator 'aten::upsample\_linear1d.out' is not currently implemented for the MPS device. If you want this op to be added in priority during the prototype phase of this feature, please comment on** [**https://github.com/pytorch/pytorch/issues/77764**](https://github.com/pytorch/pytorch/issues/77764)**. As a temporary fix, you can set the environment variable `PYTORCH\_ENABLE\_MPS\_FALLBACK=1` to use the CPU as a fallback for this op. WARNING: this will be slower than running natively on MPS.**

I believe that this problem is caused by the hyperparameter values and am researching how to troubleshoot this.

**01/24/2024**

I spent time further researching more models. Specifically, I went back and analyzed the RNN model that I first trained. Upon closer inspection, I noticed that the B-field and other attributes do oscillate. The voltage was the most noticeable and occurs on the largest scale, but it's important to note that the other attributes do too. I don’t think it’s important that my model preserves these oscillations as I can set a specific minimum threshold for my anomaly detection model to consider specific time frames as anomalies. I’m not sure how this would work for the voltage attribute. I spent the class researching different anomaly aggregators and ways to pinpoint anomalies based on a forecast model.

**01/29/2024**

I found that the Darts Library holds a method called gridsearch() that enables optimization of hyperparameters. I believe that this will allow me to get better performances on the more complex models that I trained on 1/20 and 1/24. I spend the class time reading the documentation for the method.

**classmethod gridsearch(parameters, series, past\_covariates=None, future\_covariates=None, forecast\_horizon=None, stride=1, start=0.5, start\_format='value', last\_points\_only=False, show\_warnings=True, val\_series=None, use\_fitted\_values=False, metric=<function mape>, reduction=<function mean>, verbose=False, n\_jobs=1, n\_random\_samples=None, fit\_kwargs=None, predict\_kwargs=None)**

**01/31/2024**

I spent class time implementing the grid search method that I researched on 1/29. I am trying to understand the parameters of the method as I am not sure what data values should go in. This problem seems to be from the way my data is formatted so I spent 1 ⁄ 3 of the class researching the parameters of the gridsearch method. I hope to fix this error by Feb 2.