**This week:**

The training, validation and test data has been determined and concatenated in different folders. I will use the years 1979-2008 as training, which is %80 of the total data. 2008-2009 as validation and 2010-2018 for testing. Training data consists of the 7 parameters given. Constants have not been used yet.

The training data has been resampled with a step size of 12 hours down to 21916 time points. I have trained this data on my UNet model. Here is the description of the model:

The UNet model has 2 paths, encoding and decoding. Encoding path has 3 Convolution layers. They start with 64 filters and double in the following layers. Each convolution has kernel size 3, and a padding of size 1 is applied to retain the original resolutions. After the convolution the data is downscaled by half. Then a bridge step is applied. It again does a 2D convolution but reduces the 256 channels into a more compact representation of the features. The implementation of the bridge part is not yet fully determined.

In the decoding path we apply 3 Upscaling blocks, each of which consists of a DeConv layer followed by a Conv layer. DeConv layers have respectively 256, 128 and 64 filters. They all upscale the resolution by a factor of two. After each DeConv Layer, the feature results from the encoding path with 64-128-256 channels is concatenated with the DeConv results to minimize information loss. This concatenated result is put into a 2D convolution which reduces its channels by half again.

At the end we apply another Upscaling block, but 64 channels are reduced into 7 channels that correspond to our input parameters. After each Conv and DeConv layer a ReLU activation is applied, except the last Conv layer which has a Tanh activation.

In conclusion the 7-parameter input is first downscaled in the encoding path and their features are extracted in the bridge. Then these features are upscaled and input values are reconstructed by using the extracted features.

My first experiments with SGD optimization with a learning rate of 0.1 and 0.2, batch size of 10, L1 Loss and 50 epochs seem to quickly reach 0.26 loss and then slow down to finally reach a loss between 0.22 and 0.20. Weights are initialized with Xavier initialization. Mind that this experiment neither applies feature dropout nor uses constant parameters in the input. This is only to observe the learning behavior of the UNet model on our data and how accurate the reconstruction is.

**What is planned for the next week:**

Complete the bridge part, add feature dropout, Use the constants to make the learning better.

Look at the validation loss and training loss to compare and see when overfitting might happen.

Observe the extracted features, Start experimenting with greedy feature selection method applied on the probe predictor as described in the paper by X. Zhang et. al.