# Experiment reproduction.

## LSTM_extrapolation

### Training

In the folder *experiments*, the files *lstm_extrapolation_seed#.yml* contain the configuration that will be used to run the different models.  There are certain keywords that use to access the required information. You must modify these links depending on where such files are stored. Below, we present a list of the keywords that must be modified depending on your directory structure.

- train_basin_file: ../data/basin_id/basins_camels_us_531.txt
- validation_basin_file: ../data/basin_id/basins_camels_us_531.txt
- test_basin_file: ../data/basin_id/basins_camels_us_531.txt
- per_basin_train_periods_file: train_split_file_new.p
- per_basin_test_periods_file: test_split_file_new.p
- data_dir: ../data/CAMELS_US

After the files are updated, modify ***line 10*** of the *NH_run_experimenty.py* file and run the program.

```
# Train model ---------
start_run(config_file=Path("lstm_extrapolation_seed1.yml", gpu=0))
```

### Testing

When the simulation is finished a folder with all training information will be created.  You should reference this folder on **line 13** of the *NH_run_experimenty.py* file and run the evaluation part (comment line 10 and uncomment line 13 and 14).

```
# Test model ---------
run_dir = Path("runs/ lstm_extrapolation_seed_100_0807_151158")
eval_run(run_dir=run_dir, period="test")
```

In the same simulation folder, a subfolder named *test* will be created, along with a dictionary with all the information of the testing period.

## Hybrid_extrapolation

### Training

Same as the LSTM, the first step is to link the correct paths into the *hybrid_extrapolation .yml* file. After the files are updated, modify ***line 10*** of the *NH_run_experimenty.py* file and run the program.

```
start_run(config_file=Path("hybrid_extrapolation.yml", gpu=0))
```

### Testing

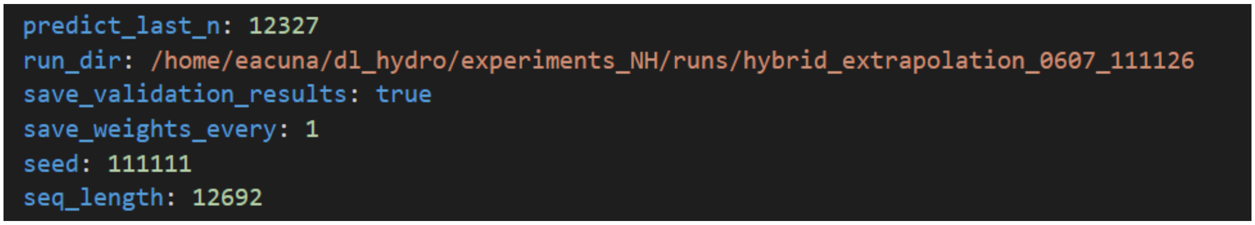
Our hybrid models were trained using a sequence length of 730 days and a warmup period of 365 days. This means that the first 365 days were only used to warmup the internal states of the model, while the predictions of the last 730-365=365 were actually used for training.

In testing, we ran the hybrid models in a rollout for the whole time period (1980-2014), and then, in a postprocess step, extract the specific dates we need for testing. Therefore, we still used 365 days to warmup or model, but then we retrieve the predictions from day 366 until the 30.09.2014.

Given that the CAMELS-US dataset starts on the 01.01.1980, and we need a warmup period of 365 days, the first actual day we can get a prediction is 31.12.1980 (366 position, note: 1980 was a leap year). If we need information from 31.01.1980 to 30.09.2014, we can make the following calculations:

*warmup_start_data = (31.12.1980) – 365d = 01.01.1980*
*new_sequence_length = (30.09.2014) – (01.01.1980) + 1d = 12692*
*new_predit_last_n = 12692-3654 = 12327*

The last two values need to be modified in the files before running the evaluation, however we need to modify the correct file. **Inside** the folder that was produce during training there is a *config.yml* file. This is a based on the arguments that we indicated in *hybrid_extrapolation.yml* but is stored inside the simulation folder. In this *config.yml* file is where one should modify the sequence_length parameter and the predict_last_n.



```
predict_last_n: 12327
run_dir: /home/eacuna/dl_hydro/experiments_NH/runs/hybrid_extrapolation_0607_111126
save_validation_results: true
save_weights_every: 1
seed: 111111
seq_length: 12692
```

After we do this modification, we can run the *eval_run* function in the *NH_run_experiement.py* file

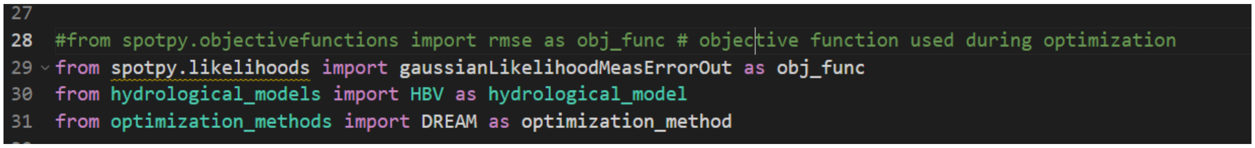
```
# Test model ---------
run_dir = Path("runs/hybrid_extrapolation_0607_111126")
eval_run(run_dir=run_dir, period="test")
```

In the same simulation folder, a *test* subfolder will be created, along with a dictionary with all the information of the testing period.

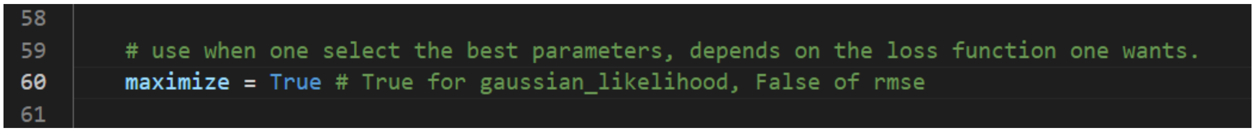
**<u>HBV model.</u>**

In the *Hy2DL/conceptual_model* directory, open the *experiment_camels_us.py* file. Modify the paths between lines 42-57 to be able to correctly access the information. When all the information is correctly referenced, the model can be run.

If you want to run the calibration using the DREAM algorithm, do the following changes in line 28, 29, 31 and 60.



```
27
28  #from spotpy.objectivefunctions import rmse as obj_func # objective function used during optimization
29  from spotpy.likelihoods import gaussianLikelihoodMeasErrorOut as obj_func
30  from hydrological_models import HBV as hydrological_model
31  from optimization_methods import DREAM as optimization_method
```



```

    # use when one select the best parameters, depends on the loss function one wants.
    maximize = True # True for gaussian_likelihood, False of rmse

```

After running the model calibrations with both SCE and DREAM, run the *conceptualmodel_postprocess.ipynb* notebook. This will create a dictionary with all the information of the testing period.

**Analysis.**

After all the simulations have been run, you can run the *ResultAnalysis.ipynb* notebook located in the *results* folder. With this you can reproduce all the figures and analysis presented in the publication.