I followed the instructions in the team's repository as reproduced the following results that are presented in their report:

1. Plotted predictions for models: DeepAR, LSTM, TFT. As a complete training of all models takes a significant amount of time I trained them only the amount of time required to obtain preliminary results. I didn't find the notebook for training and evaluating the NeuralProphet model.
2. I ran the sample small test that is presented in example_run.py to make sure all models work. This time the NeuralProphet model was there.
3. Finally, I recreated LIBRA dataset evaluation using evaluation_results.ipybn to avoid running the whole benchmark.

All results presented in the report are reproducible. The obtained evaluation looks comprehensive and reasonable.

The team didn't use any project management framework (such as Kedro) as well as there are no scripts that support extensive configuration options for training models - only Jupyter notebooks. However, this is greatly outweighed by a very clean code structure, an exhaustive set of evaluation tools, the usage of PyTorch Lighting that makes training code more readable, the ease of the reproducibility of the results, and the presence of unit tests that makes the subsequent development of the project much easier.

I believe that with the aforementioned cons being fixed the project can be classified as suitable for real-world problems. In particular, time series forecasting can be helpful in such areas as population growth forecasting.