Reproducibility Vignette for "Counterfactual Forecasting for Panel Data"

FOCUS Code Supplements

Python installation

Although the majority of the codes are implemented in `R`, user needs to have Python in the system. For installing any necessary Python packages, user can use `pip` in terminal. Following are some examples:

```
pip install numpy
pip install scipy
pip install matplotlib
...
```

For installing the codes for our benchmark algorithms, follow the instructions and dependencies of the following repositories:

– mSSA: `https://github.com/AbdullahO/mSSA`.

– SyNBEATS: `https://github.com/Crabtain959/SyNBEATS`.

`R` package dependencies

The necessary functions can be installed by running `install.packages("<package_name>")` in `R` console. User needs the following library function.

```
source("library_causalTS.R")
```

```
##   |                                                                      |
```

Experimental details

We conduct the experiments for FOCUS and mSSA with R functions in `library_causalTS.R`.

Data generation

The data sets used in the manuscript are generated with `gen_data_DGP<dgp_index>.R` function. `<dgp_ind>` = 0, 1, 2 corresponds to DGP-1, 2, 3 respectively in the manuscript

```
source("gen_data_DGP0.R") # change the index from 0 to 1, 2
```

The example line generates data sets coming from DGP-1 with $N = 64$ and $T = 32, 64, 128, 256$. For $T = 32$ and `iteration_number`$ = 1,\ldots, 30$, the example outputs are the following triplets–

- `DGP0_C0_N64_T32_iter<iteration_number>.csv`: the future mean potential outcome targets up to three horizons with dimension $N \times 3$.
- `DGP0_Y_N64_T32_iter<iteration_number>.csv`: The outcome panel with dimension $N \times 1.5T$ (T training time and $T/2$ maximum post observation time points for SyNBEATS)
- `DGP0_mcarW_N64_T32_iter<iteration_number>.csv`: The observation matrix of dimension $N \times T$ from MCAR(0.7).

The user can set customized options for the DGP parameters inside the in the function `generate_data` inside `gen_data_DGP0.R`.

Forecast outputs

```
source("forecast_DGP0.R") # change the index from 0 to 1, 2
```

The example line gives the outputs of FOCUS for DGP-1. The output file name is `DGP0_out.RData` and the output list `out_dgp0` contains the following array objects (the objects are accessible by `$` operation, e.g. `out_dgp0$<object_name>`):

- `rep_<method>_h<horizon>`: Entry-by-entry forecast MSFE across 30 trials. `method` has two options– `focus` or `ms`; `horizon` takes values 1, 2, 3.

SyNBEATS outputs

For extracting the SyNBEATS outputs, user needs to run the Jupyter notebook `SyNBEATS_forecast.ipynb`. The outputs have the following example format–

`DGP0/synout_files/N64_T32/synout_DGP0_Y_N64_T32_iter1.csv`.

- **Cleaning the SyNBEATS outputs:**

Open the file `syn_error_preprocess.R`. In line 2, change the `dgp_ind` accordingly, save the file and run the whole file with the following

```
source("syn_error_preprocess.R") # change dgp_ind in line 2 from 0 to 1, 2
```

This command creates and saves a list object called `syn_errors_final` that has attributes `rep_syn_h<horizon>`, a 4×30 matrix with `horizon` = 1,2,3. The rows correspond to $T = 32, 64, 128, 256$ and the columns represent the replications from 1 to 30. Each $(t, r)^{\text{th}}$ entry records the MSFE for time t and replication r.

Reproducing Figure 1(a)-(d) in the manuscript

Figure 1(a)-(d) capture the average MSFE (with one standard deviation error bars) across 30 trials for FOCUS, mSSA and SyNBEATS.

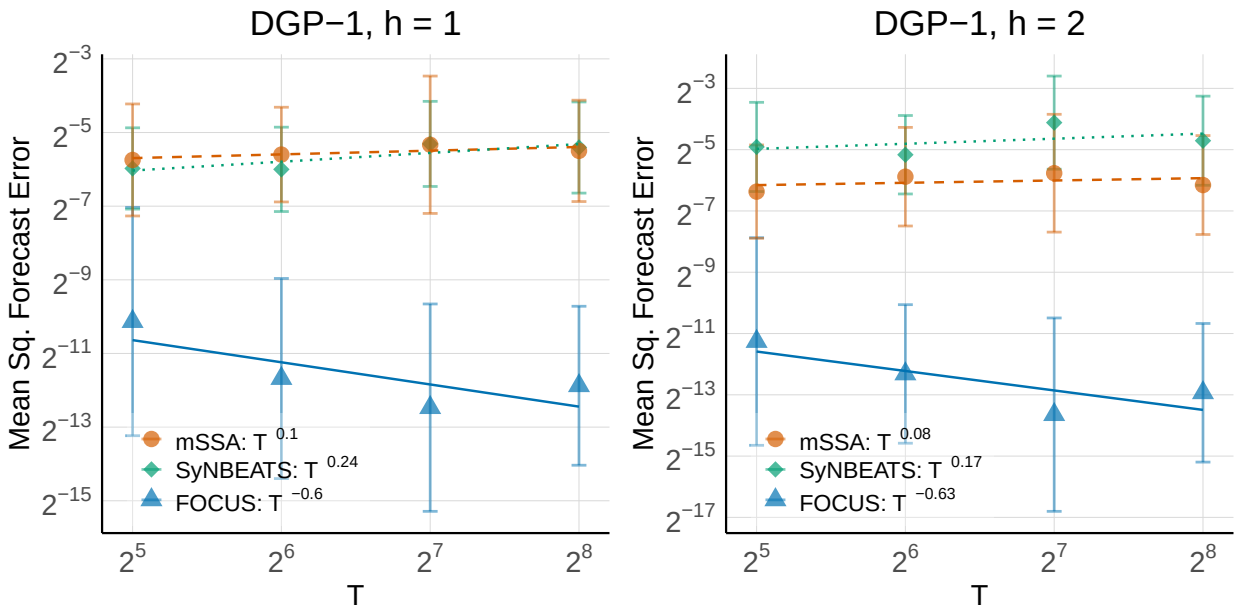
```
source("err_vs_T_plt.R")
```

```
##   |                                                                      |
```

```
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
```

```
gridExtra::grid.arrange(plt1, plt2, ncol = 2)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
```



This line creates the plots of MSFE across competing methods with T in the file

`aistats_out/DGP<dgp_ind>_h<horizon>.pdf`

with `dgp_ind` = 0, 1, 2 and `horizon` = 1, 2, 3. Similarly we recover the Figure 3 (a), (b) and Figure 4 (a), (b), (c) in the Appendix.

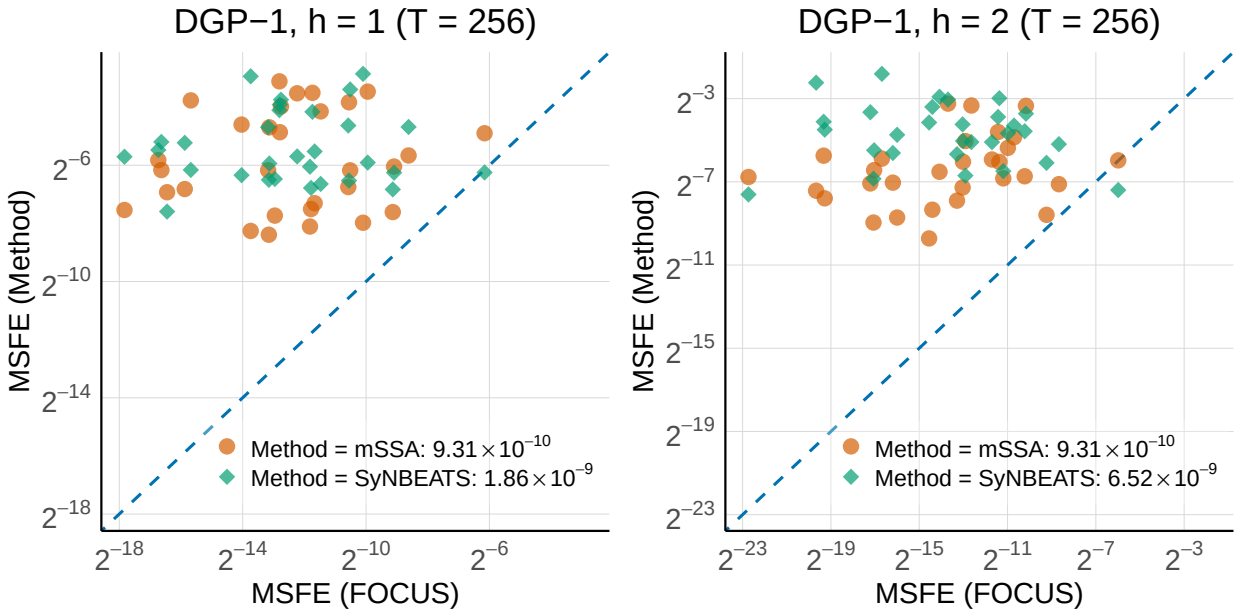
Reproducing Figure 1(e)-(h) in the manuscript

Figure 1(e)-(h) present the scatterplots of MSFE values for FOCUS and the competing method (mSSA and SyNBEATS) across 30 trials.

```
source("scatter_plot_fixed_N_T.R")
```

```
##   |                                                                      |
```

```
gridExtra::grid.arrange(plt1, plt2, ncol = 2)
```



This line creates the scatter plots of MSFE across 30 trials in the file

`aistats_out/scatter_plots/scatter_plot_DGP<dgp_ind>_T<value_of_T>_h<horizon>.pdf`

for `dgp_ind` = 0, 1, 2 `T` = 32, 64, 128, 256, and `horizon` = 1, 2, 3. This line also produces Figure 3(c), (d), and Figure 4(d)-(f) in the Appendix.

Reproducing HeartSteps results

Below we provide the details for running experiments with the HeartSteps data set.

Data cleaning

The data pre-processing is performed by running the following line (details can be found in the manuscript, Appendix section H.1):

```
source("HeartSteps_Experiments/suggestions_clean.R")
```

This line creates three main CSV files inside the folder

`./HeartSteps_Experiments/HeartStepsV1-main/data_files/cleaned_data`:

*`suggestions_select.csv`: The subset of HeartSteps with `avail == TRUE` and no `NA` observations.

*`W_full.csv`: Full panel of the observation indicator/ indicators for `avail & send == TRUE`.

*`Y_full.csv`: Full panel of log(1+ `jbsteps30`) indicator/ indicators for `avail & send == TRUE`.

Forecasting with Focus and mSSA

For obtaining the forecast result at time point $T + 5$ for the observed panel until time T (Section 5.2 of the manuscript), we run the following:

```
source("HeartSteps_Experiments/HeartSteps_forecast.R")
```

```
##   |                                                                      |
```

The output `hs_out` is saved in the following file

`HeartSteps_Experiments/HeartStepsV1-main/hs_out.RDatax`

as a list with the following important attributes:

– `z` (`nz`): The users at time $T+5$ with zero (non-zero) steps.

– `Y_nz`, `Y_nz_focus`, `Y_nz_ms`: The outcome variables at time $T+5$ of users indexed by `z` for the test data, predicted steps by FOCUS and mSSA respectively.

– `SS_<index_type>_<method>` and `MS_<index_type>_<method>`: The sum of square and mean squared forecast errors for `method` = `focus` (FOCUS) and `ms` (mSSA).

For obtaining the Figure 2(a) and 6(a)-(c), uncomment and run lines 117-149 and change the factor components for obtaining the correlation plots. The following line yields the Figure 2(c) and 2(d) in the manuscript. Figure 2(b), 6(d) and the other similar plots for different values of T are also obtained as outputs.

```
source("HeartSteps_Experiments/HeartSteps_forecast_plot.R")
```

```
##   |                                                                      |
```

```
gridExtra::grid.arrange(plt1, plt2, ncol = 2)
```