In this folder we discover dimensionless numbers, dimensionless relations, and associated figures.

Two key Jupyter Notebooks (.ipynb) files. One has the cells to operate on the energy balance, one on the mass balance. The only difference is data preparation.

Mass balance:  
pydimension\_mass\_bal\_torch\_lambda\_tuning\_v4\_looping\_train\_val\_test

Energy balance:

Pydimension\_energy\_bal\_torch\_lambda\_tuning\_v2\_looping\_c

**Output of Dimensionless Numbers:**  
Both of these codes will take in information held in the “data” folder (go up on level in the file tree).

Both will output excel sheets containing information on the discovered dimensionless number(s) in the location of your choice.

*Mass balance terms or energy balance terms?*

In this case, we have chosen to output discovered dimensionless relationships to i) dim\_num\_eb\_val\_test for predicting and ii) dim\_num\_mb\_val\_test for predicting . Consider the mass balance case.

*Pcked or unpacked?*

We further divide them into packed/unpacked. Consider the packed case.

*Number of rates*

These, in turn, are further divided by how many different process variables did we consider including. For example, did we draw process variables from the top 2 rates? Three rates? Six? Consider six rates.

*Which penalty on the gamma vector?*

For the case of six rates, we divide this further into each value of lambda\_gamma used to generate the outputs. Each folder is named for the lambda\_gamma value used.

*Outputs:*

Bad\_terms: where the dimensionless number had poor performance.

Good\_terms: where the dimensionless number had good performance.

Summary\_recheck: this summary provides the performance of each dimensionless number. This includes the r^2 performance, the gamma vector, the beta vector, the para\_threshold (also called “clamp”, the cutoff for the gamma value). Also shown are what exponents these correspond to for each variable used.   
  
For the extrapolation performance, go to tab “ext dimension number”.

**Plotting:**

For plotting the results, we have three plots:

1. Parity plot: plots the predictions vs the reality.
2. Rate tuning: charts how dimensionless number performance (test set) improves as process variables from more rates are included.
3. Lambda tuning: this shows the change in dimensionless number (test set and extrapolation set) performance and size as the penalty on dimensionless number size is tuned.

For this example, we go to the packed case of the mass balance. We plot for six rates.

Folder: dim\_num\_mb\_val\_test -> packed

Parity Plot File: “pydimension\_mass\_bal\_terms\_parity.ipynb”

Rate/lambda tuning: “plotting\_\_v3.ipynb”