LiMo – An Intelligent Library Model

LiMo is a tool for creating Intelligent Library Models for digital applications. It enables the generation of datasets used to train the models, which in turn helps perform a more accurate timing analysis and build a robust signoff framework.

I. Content

LiMo is released with the following top-level directories:

1. bin
2. scripts
3. gates
4. output

The content of each directory is described below:

1. **bin:** It contains:

   - **limo**: This is the top level bash script to launch the tool

2. **scripts:** It contains:

   - **limo.tcl:** This is the top-level TCL script responsible for executing LiMo. It provides a TCL interface to the user. You should have TCL installed on your system to run the limo.

3. **gates directory**

   The gates directory contains multiple subdirectories. Each subdirectory represents different digital logic gates (example, NAND2X1, NOR2X1, NAND3X1, NOR3X1, etc.), and a simulation.py file. Each subdirectory further contains:

   - **Simulation/schematic:** Contains input files for SPICE simulation (netlist) and the simulation results (psf).

     ✔ **netlist:** contains the spectre netlist and stimuli . Note that the netlistHeader and netlistFooter files are also needed in the same directory.

     ✔ **psf:** created by Cadence Spectre simulator. Stores output data in a single or double precision binary or human-readable ASCII format. Widely supported by many other simulators. Virtuoso design framework extensively uses PSF format.

- **scripts:** It contains various scripts for simulating and gathering data for characterizing, training and creating library models for a gate. Since, the way simulation is done and data is collected can vary for each gate, we have scripts that are gate-specific.

    - ✔ **user_input:** A script with user-defined input parameters for dataset generation.
    - ✔ **script/data_gen_none.py:** Generates data without optimization, including skew input points.
    - ✔ **script/data_gen_skew_opt.py:** Retains data points with variable delays and removes those causing constant delays during dataset generation.
    - ✔ **script/data_gen_skew_slew_opt.py:** Optimizes skew and slew, treating slew values as a binary search tree to achieve the desired delay consistency.
    - ✔ **loadData.py:** Loads a dataset from a specified CSV input file
    - ✔ **infoData.py:** Generates dataset information and saves it to a text file.
    - ✔ **plotData.py:** Creates plots and graphs to visualize the dataset.
    - ✔ **splitData.py:** Splits the dataset into training and testing.
    - ✔ **preProcessData.py:** Prepares and preprocesses the dataset for training or testing.
    - ✔ **trainModel.py:** Trains a machine learning model and generates the trained model for future use.
    - ✔ **testModel.py:** Tests a machine learning model on a dataset and generates a report.

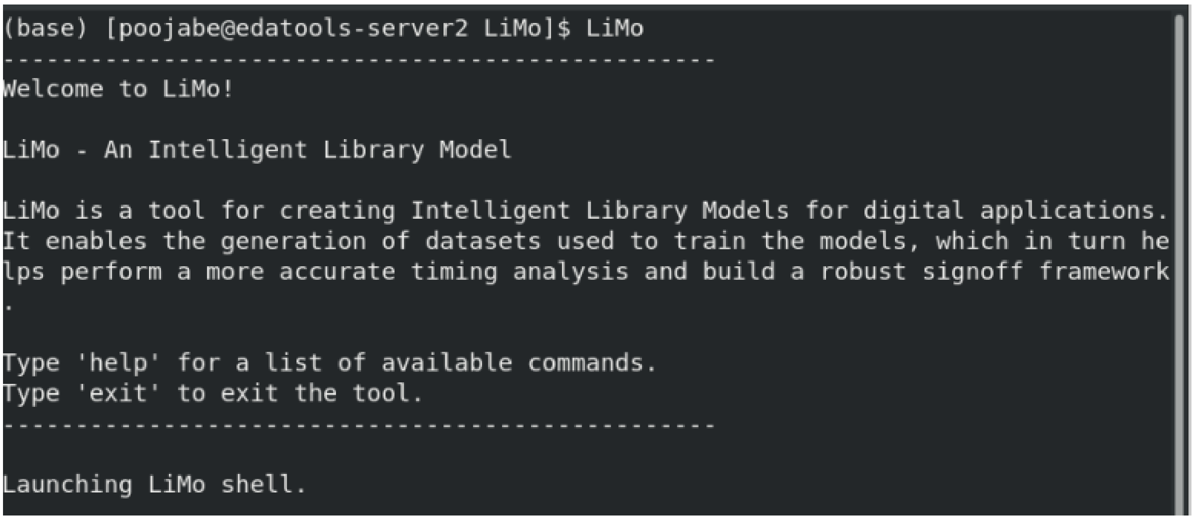
    Note: You can run the data_gen script for a specific gate using "python data_gen_none.py", "python data_gen_skew.py", and "python data_gen_skew_slew.py" from the corresponding gate directory (changes in input is done through input.json).

- **Simulation.py:** Used for multiprocessing.

4. **Output:** The output directory contains the output files generated by the tool.

II. How to Install the tool?

1. Clone LiMo from github to your local machine
   - Let the installation directory be: ***INSTALL_DIR***
2. Update the ***PATH*** to point to the ***bin*** directory of the in:
   - Assume bash shell, the following command will prepend the *bin* directory of the installation to the existing PATH ($PATH).
     export PATH="$INSTALL_DIR/bin:$PATH"
   - You can make the above change in ***~/.bashrc*** also.
3. Run LiMo from any directory using the following command
   - LiMo

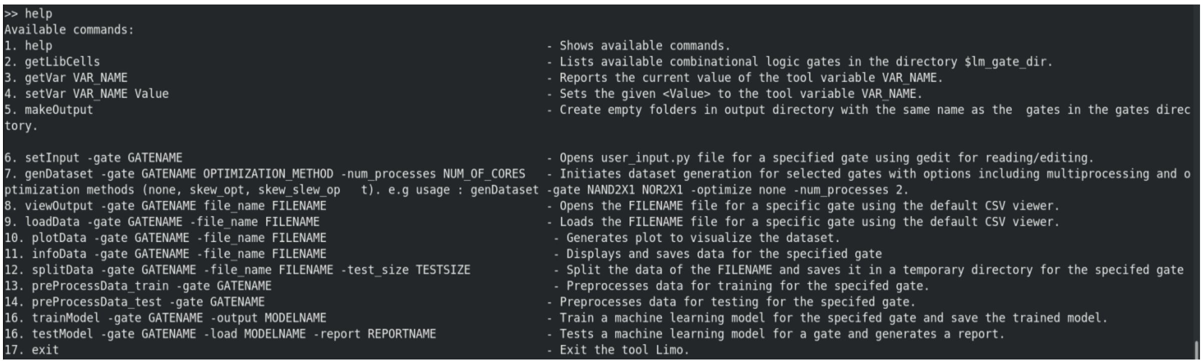


```
(base) [poojabe@edatools-server2 LiMo]$ LiMo
------------------------------------------------
Welcome to LiMo!

LiMo - An Intelligent Library Model

LiMo is a tool for creating Intelligent Library Models for digital applications.
It enables the generation of datasets used to train the models, which in turn he
lps perform a more accurate timing analysis and build a robust signoff framework
.

Type 'help' for a list of available commands.
Type 'exit' to exit the tool.
------------------------------------------------

Launching LiMo shell.
```

III. Commands and Options

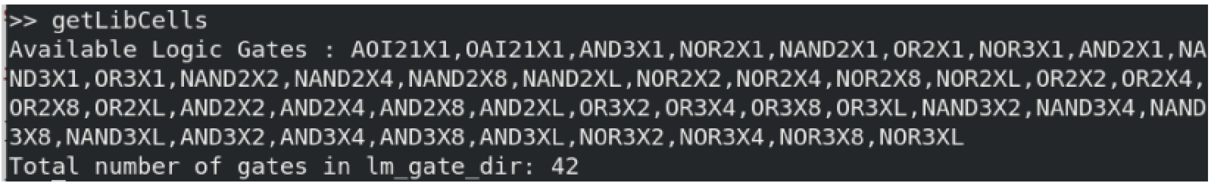
LiMo provides an interactive shell where you can run various LiMo commands. Currently, the following commands are supported:

☐ **help:** Provides a list of available commands and their descriptions.



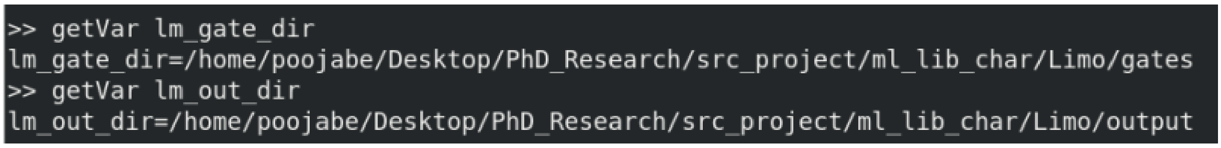
```
>> help
Available commands:
1. help                                                                        - Shows available commands.
2. getLibCells                                                                 - Lists available combinational logic gates in the directory $lm_gate_dir.
3. getVar VAR_NAME                                                             - Reports the current value of the tool variable VAR_NAME.
4. setVar VAR_NAME Value                                                       - Sets the given <Value> to the tool variable VAR_NAME.
5. makeOutput                                                                  - Create empty folders in output directory with the same name as the  gates in the gates direc
tory.

6. setInput -gate GATENAME                                                     - Opens user_input.py file for a specified gate using gedit for reading/editing.
7. genDataset -gate GATENAME OPTIMIZATION_METHOD -num_processes NUM_OF_CORES   - Initiates dataset generation for selected gates with options including multiprocessing and o
ptimization methods (none, skew_opt, skew_slew_op   t). e.g usage : genDataset -gate NAND2X1 NOR2X1 -optimize none -num_processes 2.
8. viewOutput -gate GATENAME file_name FILENAME                                - Opens the FILENAME file for a specific gate using the default CSV viewer.
9. loadData -gate GATENAME -file_name FILENAME                                 - Loads the FILENAME file for a specific gate using the default CSV viewer.
10. plotData -gate GATENAME -file_name FILENAME                                 - Generates plot to visualize the dataset.
11. infoData -gate GATENAME -file_name FILENAME                                 - Displays and saves data for the specified gate
12. splitData -gate GATENAME -file_name FILENAME -test_size TESTSIZE            - Split the data of the FILENAME and saves it in a temporary directory for the specifed gate
13. preProcessData_train -gate GATENAME                                         - Preprocesses data for training for the specifed gate.
14. preProcessData_test -gate GATENAME                                         - Preprocesses data for testing for the specifed gate.
16. trainModel -gate GATENAME -output MODELNAME                                - Train a machine learning model for the specifed gate and save the trained model.
16. testModel -gate GATENAME -load MODELNAME -report REPORTNAME                - Tests a machine learning model for a gate and generates a report.
17. exit                                                                       - Exit the tool Limo.
```

- ☐ **getLibCells:** Lists all available digital logic gates for dataset generation. It will list all the cells present in the *gates* directory. (For the *gates* directory in the release there are 42 cells)



```
>> getLibCells
Available Logic Gates : AOI21X1,OAI21X1,AND3X1,NOR2X1,NAND2X1,OR2X1,NOR3X1,AND2X1,NA
ND3X1,OR3X1,NAND2X2,NAND2X4,NAND2X8,NAND2XL,NOR2X2,NOR2X4,NOR2X8,NOR2XL,OR2X2,OR2X4,
OR2X8,OR2XL,AND2X2,AND2X4,AND2X8,AND2XL,OR3X2,OR3X4,OR3X8,OR3XL,NAND3X2,NAND3X4,NAND
3X8,NAND3XL,AND3X2,AND3X4,AND3X8,AND3XL,NOR3X2,NOR3X4,NOR3X8,NOR3XL
Total number of gates in lm_gate_dir: 42
```

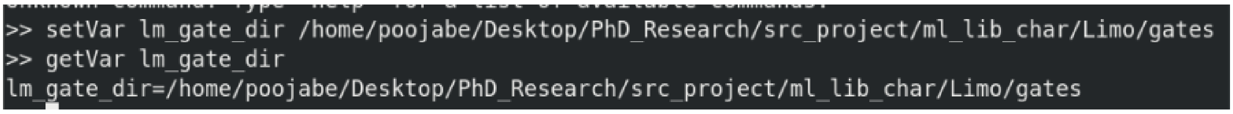
- ☐ **getVar VAR_NAME:** Reports the current value of the tool variable VAR_NAME. Currently, the following variables are supported: *lm_gate_dir* and *lm_out_dir*.



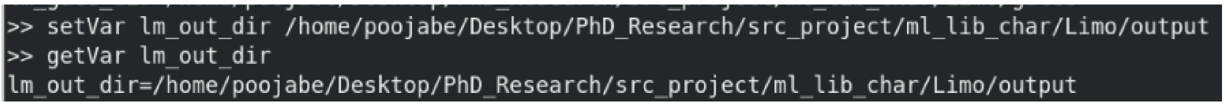
```
>> getVar lm_gate_dir
lm_gate_dir=/home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/gates
>> getVar lm_out_dir
lm_out_dir=/home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output
```

- ☐ **setVar VAR_NAME Value:** Sets the given <Value> to the tool variable VAR_NAME.

  We can define *gates* directory path and *output directory path* using this command:

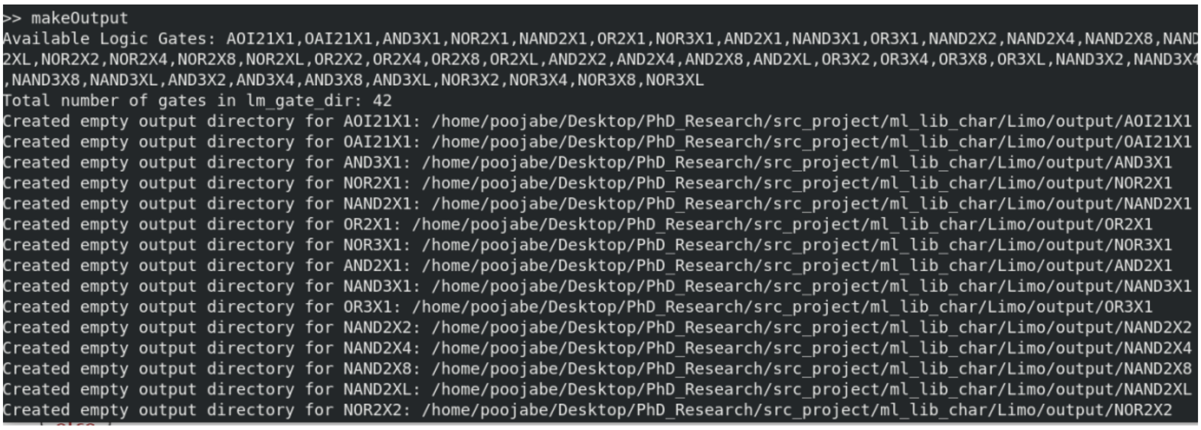


```
Unknown command. Type 'help' for a list of available commands.
>> setVar lm_gate_dir /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/gates
>> getVar lm_gate_dir
lm_gate_dir=/home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/gates
```



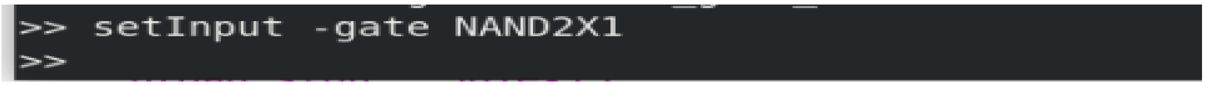
```
>> setVar lm_out_dir /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output
>> getVar lm_out_dir
lm_out_dir=/home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output
```

- ☐ **makeOutput**: Create empty directories in the *output* directory with the same names as the cells defined in the *gates* directory.



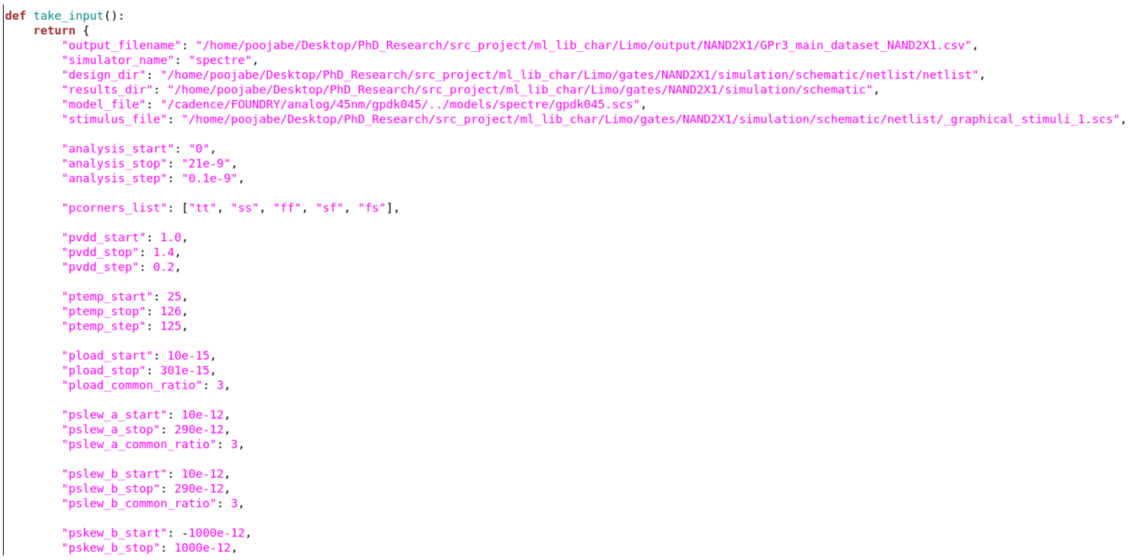
```
>> makeOutput
Available Logic Gates: AOI21X1,OAI21X1,AND3X1,NOR2X1,NAND2X1,OR2X1,NOR3X1,AND2X1,NAND3X1,OR3X1,NAND2X2,NAND2X4,NAND2X8,NAND
2XL,NOR2X2,NOR2X4,NOR2X8,NOR2XL,OR2X2,OR2X4,OR2X8,OR2XL,AND2X2,AND2X4,AND2X8,AND2XL,OR3X2,OR3X4,OR3X8,OR3XL,NAND3X2,NAND3X4
,NAND3X8,NAND3XL,AND3X2,AND3X4,AND3X8,AND3XL,NOR3X2,NOR3X4,NOR3X8,NOR3XL
Total number of gates in lm_gate_dir: 42
Created empty output directory for AOI21X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/AOI21X1
Created empty output directory for OAI21X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/OAI21X1
Created empty output directory for AND3X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/AND3X1
Created empty output directory for NOR2X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NOR2X1
Created empty output directory for NAND2X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND2X1
Created empty output directory for OR2X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/OR2X1
Created empty output directory for NOR3X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NOR3X1
Created empty output directory for AND2X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/AND2X1
Created empty output directory for NAND3X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND3X1
Created empty output directory for OR3X1: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/OR3X1
Created empty output directory for NAND2X2: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND2X2
Created empty output directory for NAND2X4: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND2X4
Created empty output directory for NAND2X8: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND2X8
Created empty output directory for NAND2XL: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND2XL
Created empty output directory for NOR2X2: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NOR2X2
```

□ **setInput -gate GATENAME:** Opens the *user_input.tcl* file for a specific gate using the default text editor. For example, **setInput -gate NAND2X1** opens the *user_input.tcl* file for the NAND2X1 gate.



```
>> setInput -gate NAND2X1
>>
```

user_input.py will be opened for NAND2X1



```
def take_input():
    return {
        "output_filename": "/home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND2X1/GPr3_main_dataset_NAND2X1.csv",
        "simulator_name": "spectre",
        "design_dir": "/home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/gates/NAND2X1/simulation/schematic/netlist/netlist",
        "results_dir": "/home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/gates/NAND2X1/simulation/schematic",
        "model_file": "/cadence/FOUNDRY/analog/45nm/gpdk045/../models/spectre/gpdk045.scs",
        "stimulus_file": "/home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/gates/NAND2X1/simulation/schematic/netlist/_graphical_stimuli_1.scs",

        "analysis_start": "0",
        "analysis_stop": "21e-9",
        "analysis_step": "0.1e-9",

        "pcorners_list": ["tt", "ss", "ff", "sf", "fs"],

        "pvdd_start": 1.0,
        "pvdd_stop": 1.4,
        "pvdd_step": 0.2,

        "ptemp_start": 25,
        "ptemp_stop": 126,
        "ptemp_step": 125,

        "pload_start": 10e-15,
        "pload_stop": 301e-15,
        "pload_common_ratio": 3,

        "pslew_a_start": 10e-12,
        "pslew_a_stop": 290e-12,
        "pslew_a_common_ratio": 3,

        "pslew_b_start": 10e-12,
        "pslew_b_stop": 290e-12,
        "pslew_b_common_ratio": 3,

        "pskew_b_start": -1000e-12,
        "pskew_b_stop": 1000e-12,
```

□ **genDataset:**Initiates dataset generation for selected gates with options including multiprocessing and optimization methods (e.g., Constant delay, binary search tree).

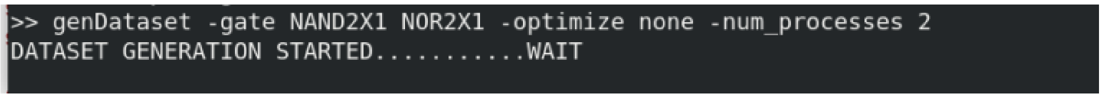
Parameters:

- **-gate**: Specify the gate's name for which you want to generate datasets.
- **-num_processes**: Specify the number of CPU cores to use for multiprocessing.

- **-optimize**: Specify the optimization method (none, skew_opt, skew_slew_opt) when generating datasets for selected gates.

  Example:

  genDataset –gate NOR2X1 ORX1 -optimize none -num_processes 2

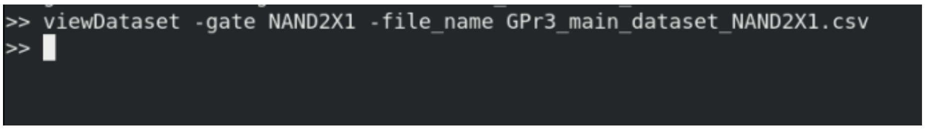


```
>> genDataset -gate NAND2X1 NOR2X1 -optimize none -num_processes 2
DATASET GENERATION STARTED..........WAIT
```

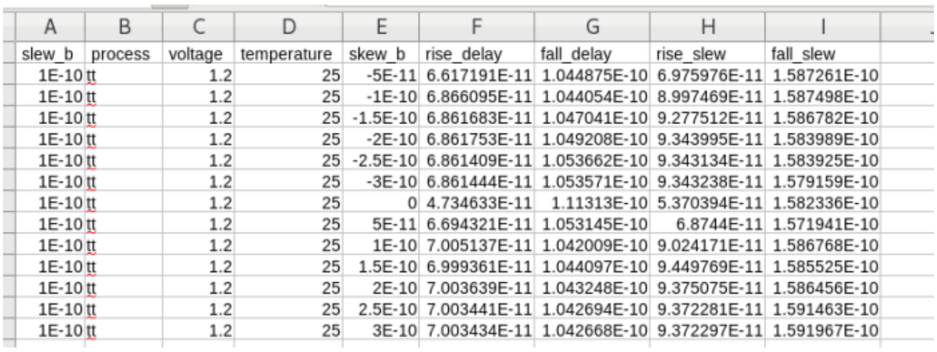
  Dataset generation will start and the generated dataset will get stored in the output/NAND2X1

□ **viewDataset -gate GATENAME –file_name FILENAME:** Opens the FILENAME file for a specific gate using the default CSV viewer. For example, **viewDataset -gate NAND2X1 –file_name output.csv** opens the output.csv file for the NAND2X1 gate. (file should be stored in the output directory of specific gate)

Dataset generated will get open

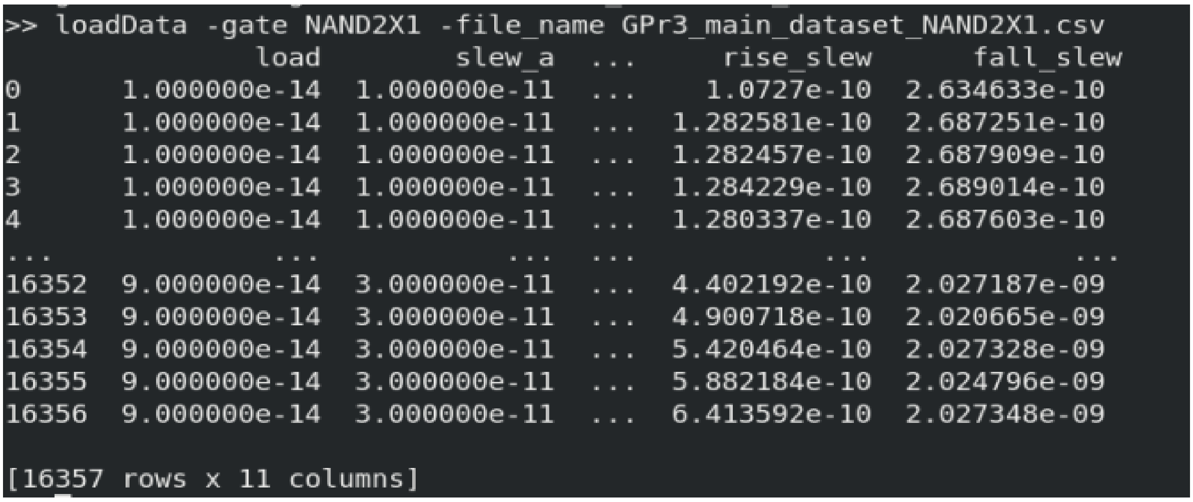


```
>> viewDataset -gate NAND2X1 -file_name GPr3_main_dataset_NAND2X1.csv
>> █
```



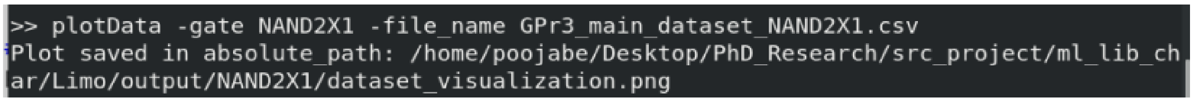
A	B	C	D	E	F	G	H	I
slew_b	process	voltage	temperature	skew_b	rise_delay	fall_delay	rise_slew	fall_slew
1E-10	tt	1.2	25	-5E-11	6.617191E-11	1.044875E-10	6.975976E-11	1.587261E-10
1E-10	tt	1.2	25	-1E-10	6.866095E-11	1.044054E-10	8.997469E-11	1.587498E-10
1E-10	tt	1.2	25	-1.5E-10	6.861683E-11	1.047041E-10	9.277512E-11	1.586782E-10
1E-10	tt	1.2	25	-2E-10	6.861753E-11	1.049208E-10	9.343995E-11	1.583989E-10
1E-10	tt	1.2	25	-2.5E-10	6.861409E-11	1.053662E-10	9.343134E-11	1.583925E-10
1E-10	tt	1.2	25	-3E-10	6.861444E-11	1.053571E-10	9.343238E-11	1.579159E-10
1E-10	tt	1.2	25	0	4.734633E-11	1.11313E-10	5.370394E-11	1.582336E-10
1E-10	tt	1.2	25	5E-11	6.694321E-11	1.053145E-10	6.8744E-11	1.571941E-10
1E-10	tt	1.2	25	1E-10	7.005137E-11	1.042009E-10	9.024171E-11	1.586768E-10
1E-10	tt	1.2	25	1.5E-10	6.999361E-11	1.044097E-10	9.449769E-11	1.585525E-10
1E-10	tt	1.2	25	2E-10	7.003639E-11	1.043248E-10	9.375075E-11	1.586456E-10
1E-10	tt	1.2	25	2.5E-10	7.003441E-11	1.042694E-10	9.372281E-11	1.591463E-10
1E-10	tt	1.2	25	3E-10	7.003434E-11	1.042668E-10	9.372297E-11	1.591967E-10

□ **loadData -gate GATENAME –file_name FILENAME:** Load data for a specific gate GATENAME from the output directory of that gate. (file should be stored in the output directory of specific gate)

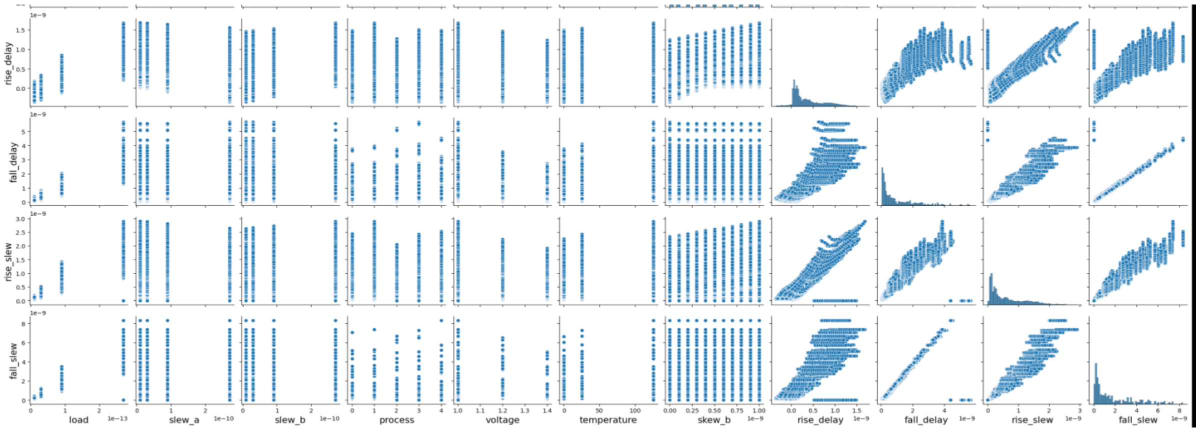


```
>> loadData -gate NAND2X1 -file_name GPr3_main_dataset_NAND2X1.csv
              load        slew_a  ...     rise_slew     fall_slew
0     1.000000e-14  1.000000e-11  ...    1.0727e-10  2.634633e-10
1     1.000000e-14  1.000000e-11  ...  1.282581e-10  2.687251e-10
2     1.000000e-14  1.000000e-11  ...  1.282457e-10  2.687909e-10
3     1.000000e-14  1.000000e-11  ...  1.284229e-10  2.689014e-10
4     1.000000e-14  1.000000e-11  ...  1.280337e-10  2.687603e-10
...            ...           ...  ...           ...           ...
16352 9.000000e-14  3.000000e-11  ...  4.402192e-10  2.027187e-09
16353 9.000000e-14  3.000000e-11  ...  4.900718e-10  2.020665e-09
16354 9.000000e-14  3.000000e-11  ...  5.420464e-10  2.027328e-09
16355 9.000000e-14  3.000000e-11  ...  5.882184e-10  2.024796e-09
16356 9.000000e-14  3.000000e-11  ...  6.413592e-10  2.027348e-09

[16357 rows x 11 columns]
```

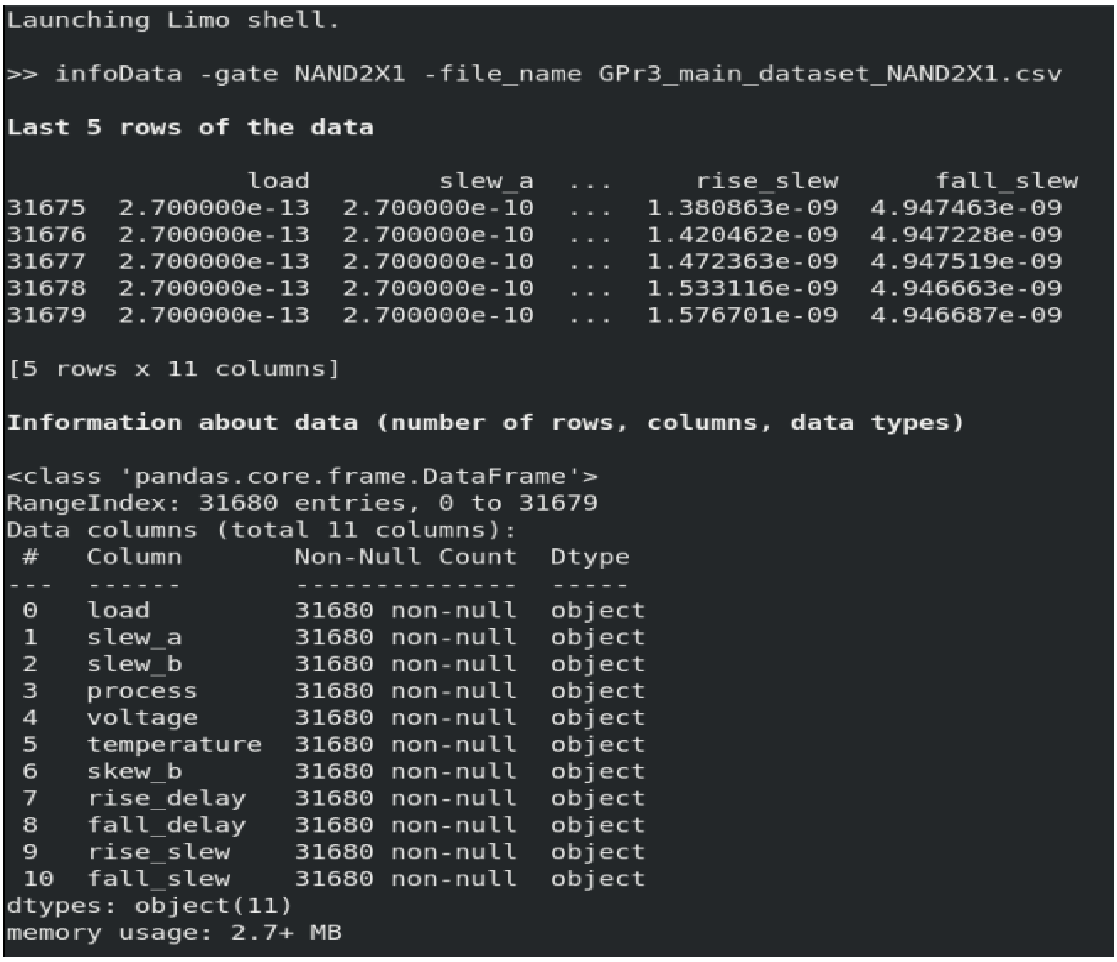
- □ **plotData -gate GATENAME –file_name FILENAME:** Generates plots to visualize the dataset for the specific gate GATENAME. Save the plot in the output directory of the specific gate. (file should be stored in the output directory of specific gate)



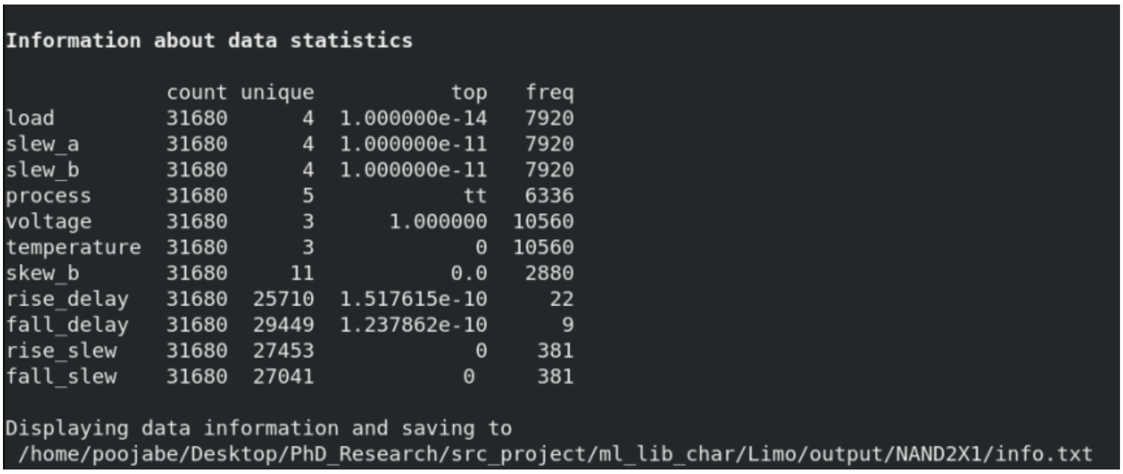
```
>> plotData -gate NAND2X1 -file_name GPr3_main_dataset_NAND2X1.csv
Plot saved in absolute_path: /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_ch
ar/Limo/output/NAND2X1/dataset_visualization.png
```



- □ **infoData -gate GATENAME –file_name FILENAME:** Generates information for the mentioned FILENAME file stored in the output directory of the specific gate.



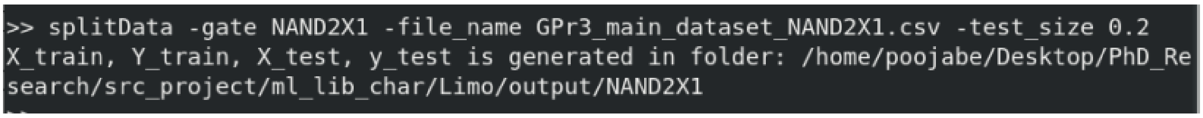
```
Launching Limo shell.

>> infoData -gate NAND2X1 -file_name GPr3_main_dataset_NAND2X1.csv

Last 5 rows of the data

                load          slew_a   ...     rise_slew      fall_slew
31675  2.700000e-13  2.700000e-10  ...  1.380863e-09  4.947463e-09
31676  2.700000e-13  2.700000e-10  ...  1.420462e-09  4.947228e-09
31677  2.700000e-13  2.700000e-10  ...  1.472363e-09  4.947519e-09
31678  2.700000e-13  2.700000e-10  ...  1.533116e-09  4.946663e-09
31679  2.700000e-13  2.700000e-10  ...  1.576701e-09  4.946687e-09

[5 rows x 11 columns]

Information about data (number of rows, columns, data types)

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 31680 entries, 0 to 31679
Data columns (total 11 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   load         31680 non-null  object
 1   slew_a       31680 non-null  object
 2   slew_b       31680 non-null  object
 3   process      31680 non-null  object
 4   voltage      31680 non-null  object
 5   temperature  31680 non-null  object
 6   skew_b       31680 non-null  object
 7   rise_delay   31680 non-null  object
 8   fall_delay   31680 non-null  object
 9   rise_slew    31680 non-null  object
 10  fall_slew    31680 non-null  object
dtypes: object(11)
memory usage: 2.7+ MB
```



```
Information about data statistics

             count unique           top   freq
load         31680      4  1.000000e-14   7920
slew_a       31680      4  1.000000e-11   7920
slew_b       31680      4  1.000000e-11   7920
process      31680      5            tt   6336
voltage      31680      3      1.000000  10560
temperature  31680      3             0  10560
skew_b       31680     11           0.0   2880
rise_delay   31680  25710  1.517615e-10     22
fall_delay   31680  29449  1.237862e-10      9
rise_slew    31680  27453             0    381
fall_slew    31680  27041            0     381

Displaying data information and saving to
 /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND2X1/info.txt
```

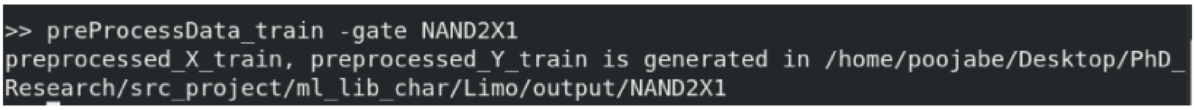
Run Below commands in sequence

☐ **splitData -gate GATENAME –file_name FILENAME -test_size TESTSIZE:** This will split the dataset and save it in the output directory of the specified gate as X_train.csv, y_train.csv, X_test.csv, y_test.csv. (file should be stored in the output directory of specific gate)



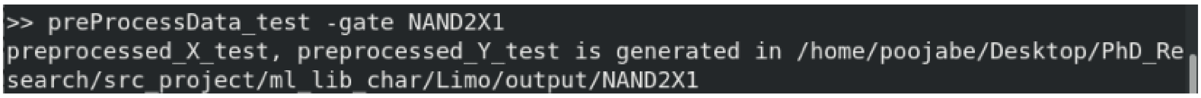
```
>> splitData -gate NAND2X1 -file_name GPr3_main_dataset_NAND2X1.csv -test_size 0.2
X_train, Y_train, X_test, y_test is generated in folder: /home/poojabe/Desktop/PhD_Re
search/src_project/ml_lib_char/Limo/output/NAND2X1
>>
```

□ **preProcessData_train -gate GATENAME:** will take X_train and y_train for specified gate from the output directory and pre-process data and save the pre-processed files for specified gate in the output directory as preprocessed_X_train.csv, preprocessed_y_train.



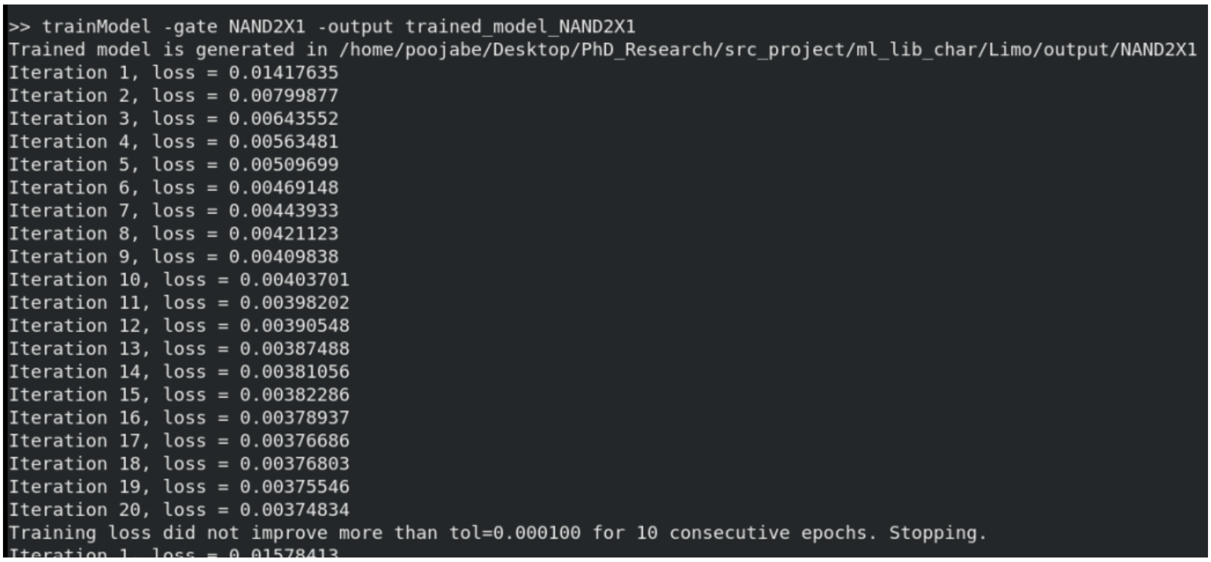
```
>> preProcessData_train -gate NAND2X1
preprocessed_X_train, preprocessed_Y_train is generated in /home/poojabe/Desktop/PhD_
Research/src_project/ml_lib_char/Limo/output/NAND2X1
```

□ **preProcessData_test -gate GATENAME:** will take X_test and y_test for specified gate from the output directory and pre-process data and save the pre-processed files for specified gate in the output directory as preprocessed_X_test.csv, preprocessed_y_test.

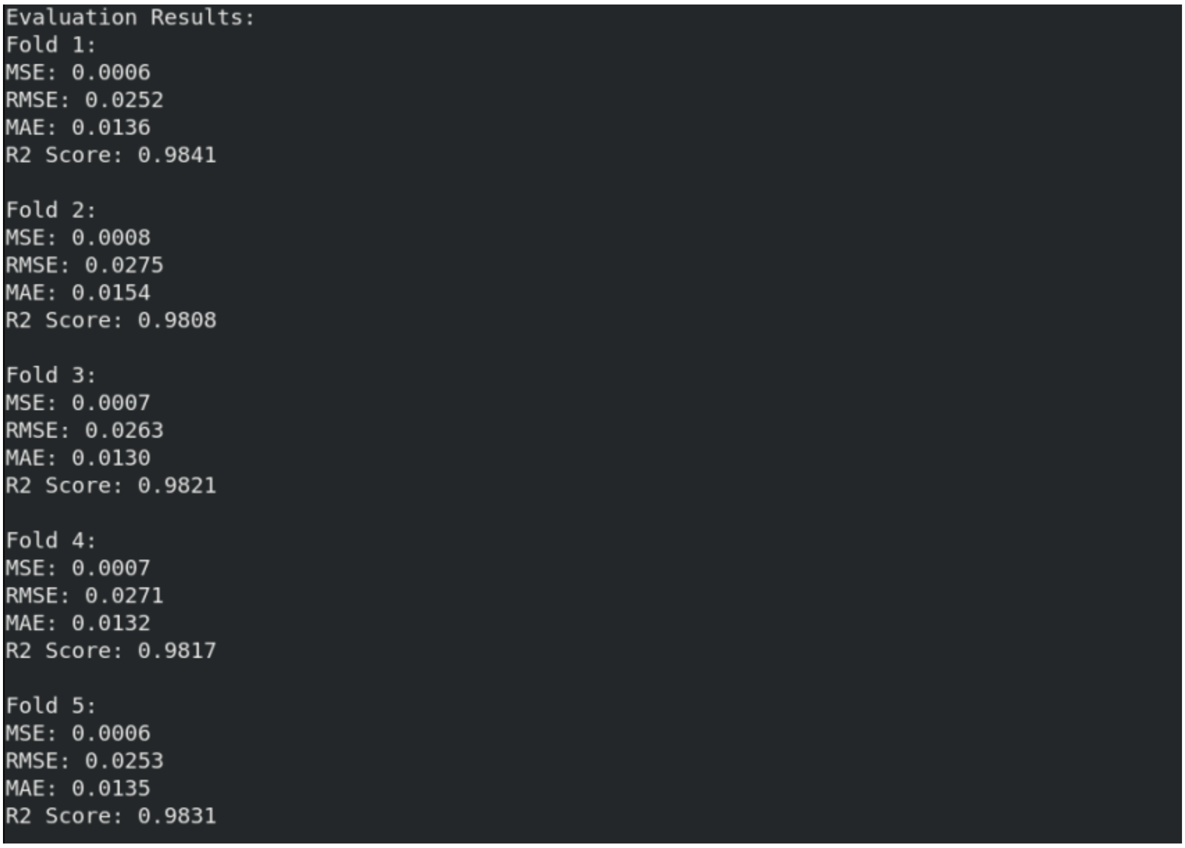


```
>> preProcessData_test -gate NAND2X1
preprocessed_X_test, preprocessed_Y_test is generated in /home/poojabe/Desktop/PhD_Re
search/src_project/ml_lib_char/Limo/output/NAND2X1
```

□ **trainModel -gate GATENAME -output MODELNAME:** This will take preprocessed_X_train.csv, preprocessed_y_train from the specified gate in output directory and train the model over it and generate the trained model with MODELNAME.

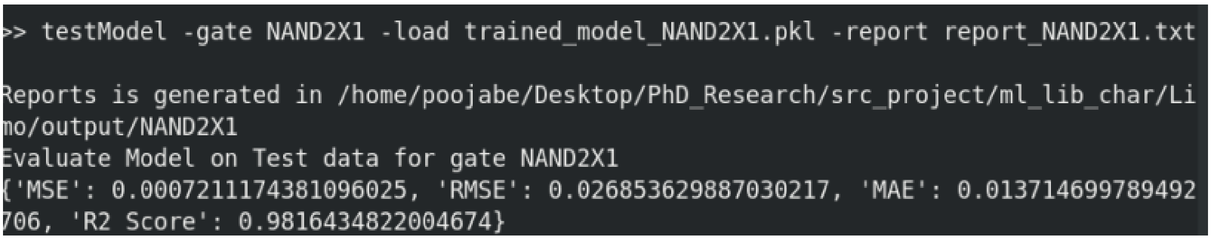


```
>> trainModel -gate NAND2X1 -output trained_model_NAND2X1
Trained model is generated in /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Limo/output/NAND2X1
Iteration 1, loss = 0.01417635
Iteration 2, loss = 0.00799877
Iteration 3, loss = 0.00643552
Iteration 4, loss = 0.00563481
Iteration 5, loss = 0.00509699
Iteration 6, loss = 0.00469148
Iteration 7, loss = 0.00443933
Iteration 8, loss = 0.00421123
Iteration 9, loss = 0.00409838
Iteration 10, loss = 0.00403701
Iteration 11, loss = 0.00398202
Iteration 12, loss = 0.00390548
Iteration 13, loss = 0.00387488
Iteration 14, loss = 0.00381056
Iteration 15, loss = 0.00382286
Iteration 16, loss = 0.00378937
Iteration 17, loss = 0.00376686
Iteration 18, loss = 0.00376803
Iteration 19, loss = 0.00375546
Iteration 20, loss = 0.00374834
Training loss did not improve more than tol=0.000100 for 10 consecutive epochs. Stopping.
Iteration 1, loss = 0.01578413
```



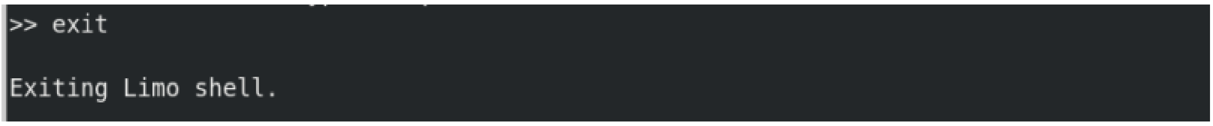
```
Evaluation Results:
Fold 1:
MSE: 0.0006
RMSE: 0.0252
MAE: 0.0136
R2 Score: 0.9841

Fold 2:
MSE: 0.0008
RMSE: 0.0275
MAE: 0.0154
R2 Score: 0.9808

Fold 3:
MSE: 0.0007
RMSE: 0.0263
MAE: 0.0130
R2 Score: 0.9821

Fold 4:
MSE: 0.0007
RMSE: 0.0271
MAE: 0.0132
R2 Score: 0.9817

Fold 5:
MSE: 0.0006
RMSE: 0.0253
MAE: 0.0135
R2 Score: 0.9831
```

☐ **testModel -gate GATENAME –load MODELNAME –report REPORTNAME**: This will load the trained model and evaluate model over preprocessed_X_test, preprocessed_y_test from the output directory and generate report with the REPORTNAME.



```
>> testModel -gate NAND2X1 -load trained_model_NAND2X1.pkl -report report_NAND2X1.txt

Reports is generated in /home/poojabe/Desktop/PhD_Research/src_project/ml_lib_char/Li
mo/output/NAND2X1
Evaluate Model on Test data for gate NAND2X1
{'MSE': 0.0007211174381096025, 'RMSE': 0.026853629887030217, 'MAE': 0.013714699789492
706, 'R2 Score': 0.9816434822004674}
```

☐ **exit:** Exits the LiMo tool.



```
>> exit

Exiting Limo shell.
```

IV. How to create your own setup for Limo?

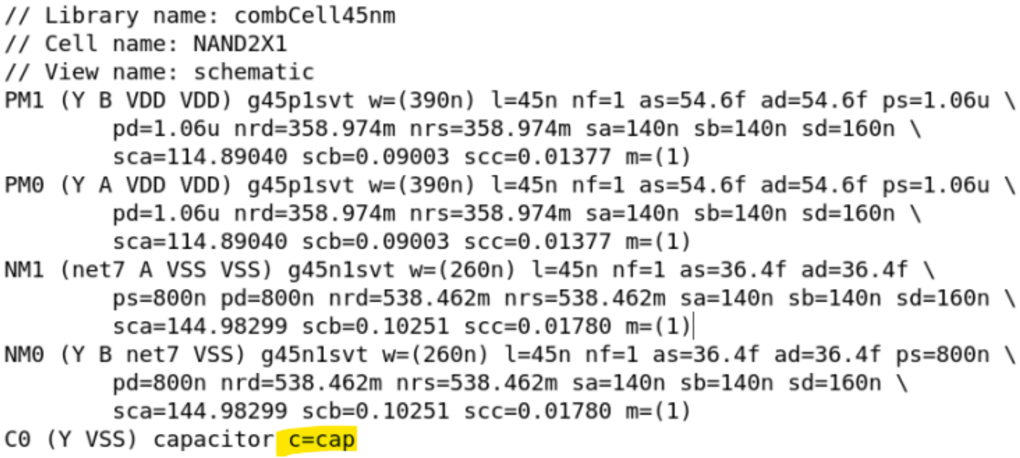
This release contains a directory named *gates*. The *gates* directory contains subdirectories for each logic gate for which an intelligent library model needs to be created. If you want to create a setup for your own library cells, you need to create a directory similar to *gates*. Then, using the *setVar* command, set the tool variable *lm_gate_dir* to the created directory.

The created *gates* library should contain the following information:

1. **subdirectories in *gates*:** Create subdirectories for each logic gate in your library. For example: NAND2X1, NOR2X1, NAND3X1, NOR3X1, etc.

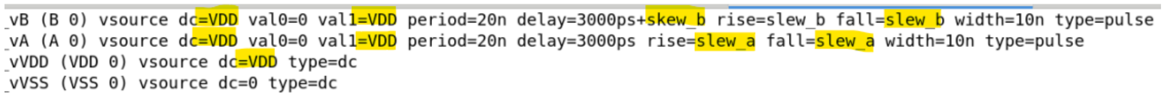
2. **gates/simulation/schematic:** Contains input files for simulation (netlist) and simulation results (psf) for each gate.

   - **netlist:** contains the spectre netlist and stimuli. Change the netlist, netlistHeader and netlistFooter with your netlist files. In netlist define the value as variable by using place holders like this:



```
// Library name: combCell45nm
// Cell name: NAND2X1
// View name: schematic
PM1 (Y B VDD VDD) g45p1svt w=(390n) l=45n nf=1 as=54.6f ad=54.6f ps=1.06u \
        pd=1.06u nrd=358.974m nrs=358.974m sa=140n sb=140n sd=160n \
        sca=114.89040 scb=0.09003 scc=0.01377 m=(1)
PM0 (Y A VDD VDD) g45p1svt w=(390n) l=45n nf=1 as=54.6f ad=54.6f ps=1.06u \
        pd=1.06u nrd=358.974m nrs=358.974m sa=140n sb=140n sd=160n \
        sca=114.89040 scb=0.09003 scc=0.01377 m=(1)
NM1 (net7 A VSS VSS) g45n1svt w=(260n) l=45n nf=1 as=36.4f ad=36.4f \
        ps=800n pd=800n nrd=538.462m nrs=538.462m sa=140n sb=140n sd=160n \
        sca=144.98299 scb=0.10251 scc=0.01780 m=(1)
NM0 (Y B net7 VSS) g45n1svt w=(260n) l=45n nf=1 as=36.4f ad=36.4f ps=800n \
        pd=800n nrd=538.462m nrs=538.462m sa=140n sb=140n sd=160n \
        sca=144.98299 scb=0.10251 scc=0.01780 m=(1)
C0 (Y VSS) capacitor c=cap
```

   Also in stimuli define the value as variable by using place holders like this:



```
_vB (B 0) vsource dc=VDD val0=0 val1=VDD period=20n delay=3000ps+skew_b rise=slew_b fall=slew_b width=10n type=pulse
_vA (A 0) vsource dc=VDD val0=0 val1=VDD period=20n delay=3000ps rise=slew_a fall=slew_a width=10n type=pulse
_vVDD (VDD 0) vsource dc=VDD type=dc
_vVSS (VSS 0) vsource dc=0 type=dc
```