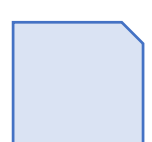


Files and Flow of Data in the Model for Inferring Cyclic Chemical Graphs

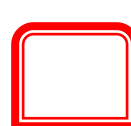
Used symbols and conventions



Files provided by the User



Files produced by the programs



Executable program

Terminal command

Module 1 (Calculating Feature Vectors)

```
python eliminate.py INPUT.sdf
```

The extension “_eli.sdf” is automatically added to the input file name.

```
./FV_ec INPUT_eli.sdf OUTPUT_desc.csv
```

The name of the output file is decided by the user and part of the command arguments.

Module 2 (Training an ANN)

```
python mol-infer_ANN.py  
INPUT_FV.csv  
INPUT_values.csv  
OUTPUT 20 10
```

- The suffixes “_biases.txt” and “_weights.txt” are added automatically.
- The filename “OUTPUT” is decided by the user.
- The constructed ANN has 2 hidden layers with 20 and 10 nodes each.

Module 3 (MILP)

```
python  
infer_cyclic_graphs_ec_id.py  
OUTPUT 42 SPEC.txt RES 1
```

- “42” is a target value given by the user
- “1” is a selection of CPLEX as MILP solver

Module 4 (Generating Graph Structures)

```
./generate_isomers  
SPEC.txt 10 100000 5 100  
OUTPUT.sdf RES_partition.txt
```

- “10” specifies 10 seconds as a computation time limit for each stage
- “100000” is an upper bound on feature vectors stored in memory
- “5” is the number of sample graphs stored for each feature vector
- “100” is an upper bound on the number of resulting graphs

INPUT.sdf

eliminate.py

INPUT_eli.sdf

FV_ec

OUTPUT_desc.csv

INPUT_values.txt

mol-infer_ANN.py

OUTPUT_biases.txt

OUTPUT_weights.txt

Infer_cyclic_graphs_ec_id.py

RES.sdf

RES_partition.txt

SPEC.txt

generate_isomers

OUTPUT.sdf

“OUTPUT.sdf” is the final result of this model.
In this example, it has up to 100 cyclic chemical graphs with structure specification given in the file “SPEC.txt”, that have target value “42”.