

Literate programming

What is literate programming ?

-
1. Donald E. Knuth, Literate Programming, 1984
. https://en.wikipedia.org/wiki/Literate_programming#Workflow

Literate programming

What is literate programming ?

*"Let us change our traditional attitude to the construction of programs :
Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather
on explaining to humans what we want the computer to do."*¹

-
1. Donald E. Knuth, Literate Programming, 1984
. https://en.wikipedia.org/wiki/Literate_programming#Workflow

Literate programming

What is literate programming ?

*"Let us change our traditional attitude to the construction of programs :
Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather
on explaining to humans what we want the computer to do."*¹

*"Literate programming is a programming paradigm introduced by Donald Knuth in which a computer
program is given an explanation of its logic in a natural language, such as English, interspersed with
snippets of macros and traditional source code, from which compilable source code can be generated."*

-
1. Donald E. Knuth, Literate Programming, 1984
. https://en.wikipedia.org/wiki/Literate_programming#Workflow

Literate programming

What does it look like ?

Literate programming

hello.ipynb x Untitled1.ipynb x hello.ipynb x Untitled.ipynb x

My Python 3.8

This is an example title

I can add more infos about my code to explain to other users and aslo to remember what my code is doing

- > - Cell with text to give informations
- > - Cell with code to view what's happen
- > - plot are also present in the output
- > - You can export the raw file in jupyter format but also in pdf, word, tex...

Subtitle for configuration import

```
[1]: import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Subtitle for my first plot

```
[2]: def gaussian(x, a, b, c):
    return a * np.exp(-b * (x-c)**2)

def noisy_gaussian():
    # gaussian array y in interval -5 <= x <= 5
    nx = 100
    x = np.linspace(-5.0, 5.0, nx)
    y = gaussian(x, a=2.0, b=0.5, c=1.5)
    noise = np.random.normal(0.0, 0.2, nx)
    y += noise
    return x, y

def fit(x, y, n):
    pfit = np.polyfit(x, y, n)
    yfit = np.polyval(pfit, x)
    return yfit

def plot(x, y, yfit):
    plt.plot(x, y, "r", label="Data")
```

This is an example title

I can add more infos about my code to explain to other users and aslo to remember what my code is doing

- Cell with text to give informations
- Cell with code to view what's happen
- plot are also present in the output
- You can export the raw file in jupyter format but also in pdf, word, tex...

Subtitle for configuration import

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Subtitle for my first plot

```
In [2]: def gaussian(x, a, b, c):
    return a * np.exp(-b * (x-c)**2)

def noisy_gaussian():
    # gaussian array y in interval -5 <= x <= 5
    nx = 100
    x = np.linspace(-5.0, 5.0, nx)
    y = gaussian(x, a=2.0, b=0.5, c=1.5)
    noise = np.random.normal(0.0, 0.2, nx)
    y += noise
    return x, y

def fit(x, y, n):
    pfit = np.polyfit(x, y, n)
    yfit = np.polyval(pfit, x)
    return yfit
```

Literate programming

What does it look like ?

Interactive programming interface allowing to combine both natural and computer languages

Literate programming

What does it look like ?

Interactive programming interface allowing to combine both natural and computer languages

In one file

- Explanation
- Code
- Results
- Graphs and plots

Literate programming

Why using literate programming frameworks ?

- Labbook

Literate programming

Why using literate programming frameworks ?

- Labbook
- Day-to-day analysis

Literate programming

Why using literate programming frameworks ?

- Labbook
- Day-to-day analysis
- Make automatic reports

Literate programming

Why using literate programming frameworks ?

- Labbook
- Day-to-day analysis
- Make automatic reports
- Write scientific article

Literate programming

example

Example of an article written using a notebook

. <https://www.frontiersin.org/articles/10.3389/fphys.2018.00787/full>

Literate programming

example

The screenshot shows the GitHub interface for the repository 'colomoto / colomoto-docker'. The repository is public and has 6 watchers, 6 forks, and 15 stars. The main content area displays a Jupyter Notebook file named 'Usecase - Mutations enabling tumour invasion.ipynb' under the path 'colomoto-docker / usecases'. The notebook was updated by 'pauleve' with the commit '14905c2' on April 24, 2020. The notebook has 2513 lines (2513 sloc) and is 422 KB in size. The title of the notebook is 'Prediction of Mutations to Control Pathways Enabling Tumour Cell Invasion with the CoLoMoTo Interactive Notebook (Tutorial)'. The authors listed are Nicolas Levy^{1,2}, Aurélien Naldi³, Céline Hernandez³, Gautier Stoll⁴⁻⁶, Denis Thieffry³, Andrei Zinoviyev⁹⁻¹¹, Laurence Calzone⁹⁻¹¹, and Loïc Paulevé^{1,*}. The footnotes provide affiliations for each author: ¹ LRI UMR 8623, Université Paris-Sud, CNRS, Université Paris-Saclay, Orsay, France; ² École Normale Supérieure de Lyon, France; ³ Computational Systems Biology team, Institut de Biologie de l'École Normale Supérieure, CNRS UMR8197, INSERM U1024, École Normale Supérieure, PSL Université, Paris, France; ⁴ Université Paris Descartes/Paris V, Sorbonne Paris Cité, Paris, France; ⁵ Équipe 11 labellisée Ligue Nationale contre le Cancer, Centre de Recherche des Cordeliers, Paris, France; ⁶ Institut National de la Santé et de la Recherche Médicale, U1138, Paris, France; ⁷ Université Pierre et Marie Curie, Paris, France.

Literate programming

example

The screenshot shows a Jupyter Notebook environment. On the left is a file explorer with a search bar and a table of files. The main area displays a notebook titled 'hello.ipynb' with the following content:

Prediction of Mutations to Control Pathways Enabling Tumour Cell Invasion with the CoLoMoTo Interactive Notebook (Tutorial)

Authors: Nicolas Levy^{1,2}, Aurélien Naldi³, Céline Hernandez³, Gautier Stoll⁴⁻⁸, Denis Thieffry³, Andrei Zinovyev⁹⁻¹¹, Laurence Calzone⁹⁻¹¹, Loïc Paulevé^{1,*}

¹ LRI UMR 8623, Université Paris-Sud, CNRS, Université Paris-Saclay, Orsay, France; ² École Normale Supérieure de Lyon, France; ³ Computational Systems Biology team, Institut de Biologie de l'Ecole Normale Supérieure, CNRS UMR8197, INSERM U1024, École Normale Supérieure, PSL Université, Paris, France; ⁴ Université Paris Descartes/Paris V, Sorbonne Paris Cité, Paris, France; ⁵ Équipe 11 labellisée Ligue Nationale contre le Cancer, Centre de Recherche des Cordeliers, Paris, France; ⁶ Institut National de la Santé et de la Recherche Médicale, U1138, Paris, France; ⁷ Université Pierre et Marie Curie, Paris, France; ⁸ Metabolomics and Cell Biology Platforms, Gustave Roussy Cancer Campus; Villejuif, France; ⁹ Institut Curie, PSL Research University, Paris, France; ¹⁰ INSERM, U900, Paris, France; ¹¹ MINES ParisTech, PSL Research University, CBIO-Centre for Computational Biology, Paris, France

Abstract

Boolean and multi-valued logical formalisms are increasingly used to model complex cellular networks. To ease the development and analysis of logical models, a series of software tools have been proposed, often with specific assets. However, combining these tools typically implies a series of cumbersome software installation and model conversion steps. In this respect, the *CoLoMoTo Interactive Notebook* provides a joint distribution of several logical modelling software tools, along with an interactive web Python interface easing the chaining of complementary analyses.

Our computational workflow combines (1) the importation of a GiNsim model and its display, (2) its format conversion using the Java library BioLQM, (3) the formal prediction of mutations using the OCaml software Pint, (4) the model checking using the C++ software NuSMV, (5) quantitative stochastic simulations using the C++ software MaBoSS, and (6) the visualisation of results using the Python library matplotlib. We rely on the CoLoMoTo Interactive Notebook to provide a thoroughly documented and executable notebook of our computational analysis.

To illustrate our approach, we use a recent Boolean model of the signalling network controlling tumour cell invasion and migration. Our model analysis culminates with the prediction of sets of mutations presumably involved in a metastatic phenotype. Beyond the specific application considered here, the resulting Jupyter notebook constitutes a tutorial for the use and combination of several logical modelling tools, which can be easily adapted to analyse other models. Furthermore, the notebook web interface and the companion Docker distribution of logical modelling tools greatly facilitate the production of accessible and reproducible model analysis workflows.

Literate programming

example

The screenshot shows the Frontiers website interface. At the top, there's a navigation bar with the Frontiers logo, 'About us', 'All journals', 'All articles', and a 'Submit your research' button. Below this is a secondary navigation bar for 'Frontiers in Physiology' with links to 'Sections', 'Articles', 'Research Topics', 'Editorial board', and 'About journal'. The main content area features a 'PROTOCOLS article' section with the journal name 'Front. Physiol.', date '06 July 2018', section 'Sec. Systems Biology Archive', and a DOI link. It also mentions the article is part of a 'Research Topic' on 'Logical Modeling of Cellular Processes'. The title of the article is 'Prediction of Mutations to Control Pathways Enabling Tumor Cell Invasion with the CoLoMoTo Interactive Notebook (Tutorial)'. The authors listed are Nicolas Levy, Aurélien Naldi, Céline Hernandez, Gautier Stoll, Denis Thieffry, Andrei Zinovyev, Laurence Calzone, and Loïc Paulevé. At the bottom, there are footnotes for each author's affiliation, ranging from LRI UMR 8623 to Institut National de la Santé et de la Recherche Médicale.

frontiers About us ▾ All journals All articles Submit your research

Frontiers in Physiology Sections ▾ Articles Research Topics Editorial board About journal ▾

PROTOCOLS article

Front. Physiol., 06 July 2018
Sec. Systems Biology Archive
<https://doi.org/10.3389/fphys.2018.00787>

This article is part of the Research Topic
Logical Modeling of Cellular Processes: From Software Development to Network Dynamics
[View all 23 Articles >](#)

Prediction of Mutations to Control Pathways Enabling Tumor Cell Invasion with the CoLoMoTo Interactive Notebook (Tutorial)

Nicolas Levy^{1,2}, Aurélien Naldi³, Céline Hernandez³, Gautier Stoll^{4,5,6,7,8}, Denis Thieffry³,
 Andrei Zinovyev^{9,10,11,12}, Laurence Calzone^{9,10,11} and Loïc Paulevé^{4,1*}

¹ LRI UMR 8623, Centre National de la Recherche Scientifique, Université Paris-Sud, Université Paris-Saclay, Orsay, France
² Ecole Normale Supérieure de Lyon, Lyon, France
³ Computational Systems Biology Team, Institut de Biologie de l'École Normale Supérieure, Centre National de la Recherche Scientifique UMR8197, INSERM U1024, École Normale Supérieure, PSL Université, Paris, France
⁴ Université Paris Descartes, Sorbonne Paris Cité, Paris, France
⁵ Equipe 11 Labellisée Ligue Nationale contre le Cancer, Centre de Recherche des Cordeliers, Paris, France
⁶ Institut National de la Santé et de la Recherche Médicale, Paris, France
⁷ Université Pierre et Marie Curie, Paris, France

Markup

A markup language uses tags to define elements within a document.
Three different types and usage :

Markup

A markup language uses tags to define elements within a document.
Three different types and usage :

- Presentational (used by traditional word-processing systems)

Markup

A markup language uses tags to define elements within a document.

Three different types and usage :

- Presentational (used by traditional word-processing systems)
- Procedural, provides instructions to process the text (e.g. TeX, PostScript)

Markup

A markup language uses tags to define elements within a document.

Three different types and usage :

- Presentational (used by traditional word-processing systems)
- Procedural, provides instructions to process the text (e.g. TeX, PostScript)
- Descriptive, to label documents parts (e.g. LaTeX, HTML, XML...)

Markdown

Markdown is a Lightweight markup language
Designed to be :



Markdown

Markdown is a Lightweight markup language

Designed to be :

- easy to write using any generic text editor (plain-text-formatting syntax)



Markdown

Markdown is a Lightweight markup language

Designed to be :

- easy to write using any generic text editor (plain-text-formatting syntax)
- easy to read in the raw format



Markdown

Markdown is a Lightweight markup language

Designed to be :

- easy to write using any generic text editor (plain-text-formatting syntax)
- easy to read in the raw format



```
# Heading level 1
## Heading level 2
### Heading level 3

**Bold text**

*Italic*

~~strikethrough text~~

A [link](http://www.example.com)

Item list
- one point
  - a subpoint
- another point
- last point
```

Markdown

Markdown is a Lightweight markup language

Designed to be :

- easy to write using any generic text editor (plain-text-formatting syntax)
- easy to read in the raw format



Used on Github to make the README.md

But how is this useful for literate programming ?

When you want to weave both code (to be interpreted) and formatting information, you precisely need a lightweight language for the formatting part.

R notebooks vs Jupyter(Lab) notebook



VS



www.educba.com

R notebook

1 Sweave in 2002

Leisch, Friedrich (2002). "Sweave, Part I : Mixing R and LaTeX : A short introduction to the Sweave file format and corresponding R functions"

R notebook

1 Sweave in 2002

Leisch, Friedrich (2002). "Sweave, Part I : Mixing R and LaTeX : A short introduction to the Sweave file format and corresponding R functions"

2 knitr in 2011

"The knitr package was designed to be a transparent engine for dynamic report generation with R, solve some long-standing problems in Sweave, and combine features in other add-on packages into one package"

RMarkdown



2012 Rmarkdown was born !

. <https://rmarkdown.rstudio.com>

RMarkdown

2012 Rmarkdown was born !



. <https://rmarkdown.rstudio.com>

RMarkdown

2012 Rmarkdown was born !



"When you run render, R Markdown feeds the .Rmd file to knitr, which executes all of the code chunks and creates a new markdown (.md) document which includes the code and its output. The markdown file generated by knitr is then processed by pandoc which is responsible for creating the finished format."

RMarkdown

```

1 ---
2 title: |
3   <center> Rapport d'analyse séquençage NGS </center>
4   <center> Détection de variants SARS-CoV-2 </center>
5 author: |
6   <center> Service de Virologie </center>
7   <center> Laboratoire de Biologie du CHU de Clermont-Ferrand </center>
8
9 date: "r format(Sys.time(), '%d %B %Y')""
10 output:
11   rmdformats::readthedown|
12 editor_options:
13   chunk_output_type: console
14 ---
15
16 # Configuration de l'analyse
17 ```{r config, engine.opts='-l',echo=F}
18 # chemin absolu vers les fichiers, scripts de l'application
19 lineage_analysis_dir <- "/home/pierre/Seafle/Seafle/lineage_analysis/"
20 #Sys.setenv(params = lineage_analysis_dir)
21 source(paste0(lineage_analysis_dir, "conf_R/initialize_lineage.txt"))
22 confR <- list.files(path = lineage_analysis_dir, pattern = conf_lineage, full.names = T, recursive = T)
23 # chargement du code R de configuration
24 source(confR)
25 # export for bash chunk the path of ANALYSE
26 Sys.setenv(ANALYSE = ANALYSE)
27 ```
28
29 ```
30 `r kable(config_files_input,col.names = "",caption = "Liste des fichiers de configurations")`
  
```

Configuration de l'analyse
 Paramètres d'entrée de Covid-Seq depuis Basespace
 Import des données de la séquence S
 Création du jeu de données global
 Liste des mutations de référence des clades
 Analyse des patients et des contrôles
 Analyse des mutations dans S
 Rendu des résultats pour *GLIMS*
 Rendu des résultats pour *GISAID*

RMarkdown

Rapport d'analyse séquençage NGS Détection de variants SARS- CoV-2

Configuration de l'analyse

Paramètres d'entrée de Covid-Seq
depuis Basespace

Import des données de la séquence S

Création du jeu de données global

Liste des mutations de référence des
clades

Analyse des patients et des contrôles

Analyse des mutations dans S

Rendu des résultats pour GLIMS

Rendu des résultats pour GISAIID



Service de Virologie
Laboratoire de Biologie du CHU de
Clermont-Ferrand

Import des données de la séquence S

Nombre de séquences trouvées dans le fichier fasta depuis basespace : 93

Nombre de séquences trouvées dans le fichier fasta aligné via Nextclade : 93

Création du jeu de données global

Le fichier de fusion des données contient 96 entrées

Liste des mutations de référence des clades

Copy

CSV

Excel

PDF

Print

Show

10

entries

Search:

Tableau des mutations de références utilisées pour l'analyse des variants

scorpio_clade	lineage.ref	synonymous_mutation_ref
All	A	All
20A/EU2	B.1.160	C4543T,G5629T,C11497T,T26876C
20A/S:126A	B.1.620	C241T,C3037T,C15324T,T20049C,A23416T,T27795-,T27796-,A27797-,A28271T
20A/S:439K	B.1.258	C8047T
20A/S:98F	B.1.221	C28651T
20B/S:1122L	B.1.1.302	
20B/S:626S	B.1.1.277	
20B/S:732A	B.1.1.519	C203T,C222T,C241T,C3037T,C10954T,T19839C,C21306T,C29197T
20C/S:80Y	B.1.367	G4960T,C6070T,C7303T,C7564T,C10279T,C10525T,C10582T,C27804T

Jupyter

- 1 2011 : IPython (interactive Python shell) with notebook functionalities

. <https://jupyter.org/>

Jupyter

- 1 2011 : IPython (interactive Python shell) with notebook functionalities
- 2 2014 : Spin-off project called Project Jupyter a non-profit, open-source project maintained by a strong community

. <https://jupyter.org/>

Jupyter

- 1 2011 : IPython (interactive Python shell) with notebook functionalities
- 2 2014 : Spin-off project called Project Jupyter a non-profit, open-source project maintained by a strong community
 - "Jupyter will always be 100% open-source software, free for all to use and released under the liberal terms of the modified BSD license"

. <https://jupyter.org/>

Jupyter

- 1 2011 : IPython (interactive Python shell) with notebook functionalities
- 2 2014 : Spin-off project called Project Jupyter a non-profit, open-source project maintained by a strong community
 - "Jupyter will always be 100% open-source software, free for all to use and released under the liberal terms of the modified BSD license"
 - A reference to the three core programming languages supported by Jupyter (Julia, Python and R)

. <https://jupyter.org/>

Jupyter

What is it exactly ?

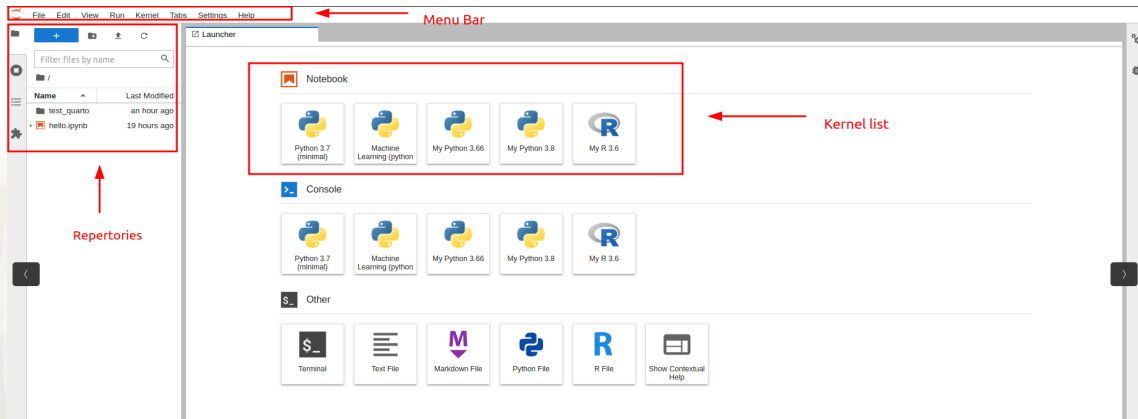
Web-based interactive computational environment

Jupyter

What is it exactly ?

Web-based interactive computational environment

- Web-based : client/server

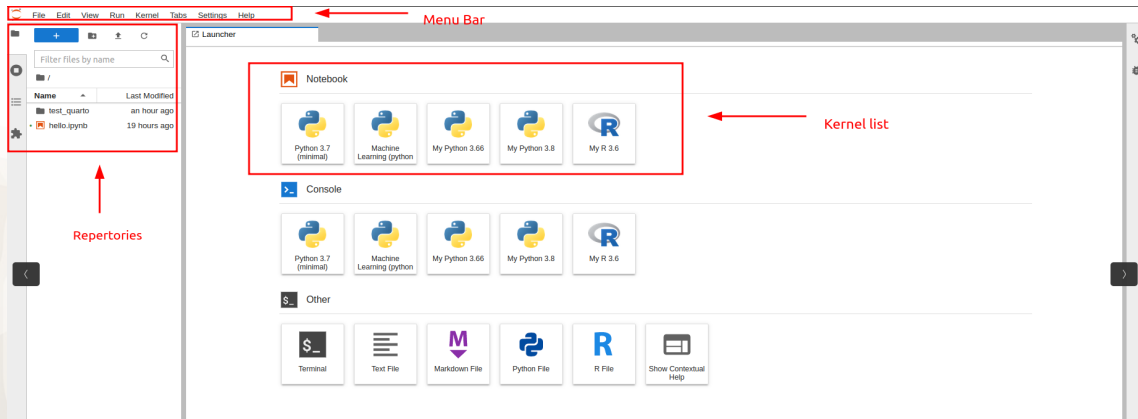


Jupyter

What is it exactly ?

Web-based interactive computational environment

- Web-based : client/server
- Interactive : notebook system

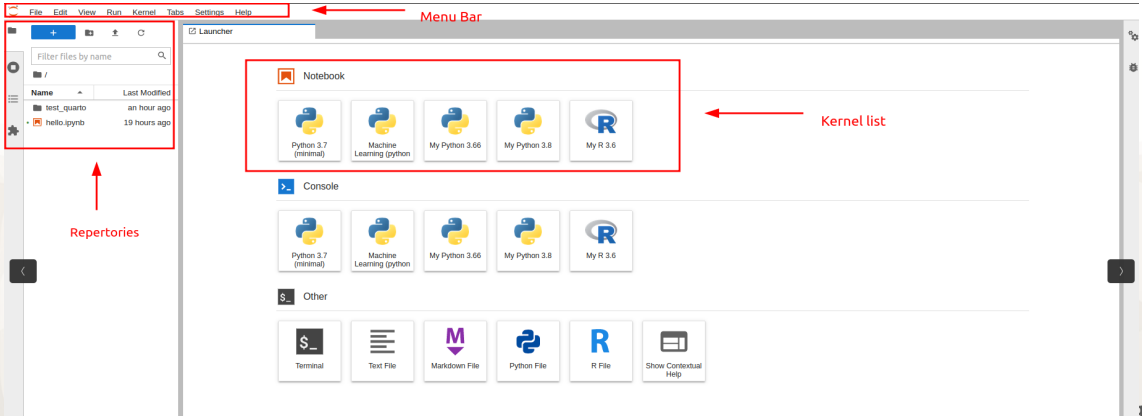


Jupyter

What is it exactly ?

Web-based interactive computational environment

- Web-based : client/server
- Interactive : notebook system
- Computational environment : console, many kernels available...



Jupyter

What is it exactly ?

Web-based interactive computational environment

The screenshot displays the Jupyter Notebook web interface. On the left is a file browser with a search bar and a table of files:

Name	Last Modified
test_quarto	an hour ago
hello.ipynb	19 hours ago

The main area shows a notebook cell with the following content:

In this graph, the vertices shown in white or orange denote components that are OFF (value 0) or ON (value 1) respectively.

Assessing the probabilities to reach alternative attractors using MaBoSS

MaBoSS [Stoll2017](https://doi.org/10.1093/bioinformatics/btx123) is a C++ software enabling the stochastic simulation of Boolean networks by translating them into continuous time Markov processes. Each node activation and inactivation is associated with an "up" and a "down" rate, which specify the propensity of the corresponding transitions. From a given state, the simulation integrates all the possible node updates and derives a probability and a duration for each transition. By default, all transitions are assigned the same rate. For a given set of initial conditions, MaBoSS produces time trajectories and estimates probabilities of model states over the whole simulation time. Steady state distributions can thus be approximated, provided that a sufficient number of sufficiently long simulations have been performed.

The aim of this section is to reproduce part of the results obtained by [Cohen2015](https://doi.org/10.1371/journal.pcbi.1004571) which show that a Notch (NICD) gain-of-function together with a p53 loss-of-function prevent reaching a stable apoptotic phenotype.

First, we convert the bioQM model to MaBoSS:

```
[ ]: wt_sim = biolqm.to_maboss(lqm)
```

The variable `wt_sim` is a Python object that gathers both the Boolean network rules and the settings for the simulations, including the transition rates.

Simulation setup

The stochastic simulation of Boolean networks with MaBoSS requires the specification of several parameters.

Initial states

First, a distribution of initial states must be specified: each simulation then starts from a state sampled from this distribution. The distribution is determined by assigning a probability to start in state 0 or in state 1 to each node. By default, a node has a probability 1 to start in state 0.

The `maboss` Python library provides widgets to ease the assignment of this initial distribution. The following code enables the definition of a distribution of initial states with all nodes at 0, but `DNAdamage` and `ECMicroenv` with equiprobable 0 and 1 values. After pressing "OK", the notebook cell will be replaced by the actual Python call resulting in equal probabilities for these two nodes to start in active or inactive states.

```
[ ]: maboss.wg_set_istate(wt_sim)

[ ]: #maboss.wg_set_istate(wt_sim)
maboss.set_nodes_istate(wt_sim, ["DNAdamage", "ECMicroenv"],
                        [0.5, 0.5])
```

Output nodes

Using MaBoSS, we can focus on the output nodes and ignore the other nodes, which enable us to identify the corresponding phenotypes. This can be done using the following code:

PRACTICE

- Jupyter training link
- Rmarkdown training link