AUTOMATED INTEGRATION OF STRUCTURAL, BIOLOGICAL AND METABOLIC SIMILARITIES TO SUSTAIN READ-ACROSS

An automated procedure for the selection of analogues for data gap-filling is made available here (https://github.com/DGadaleta88/RAX_tool). Given a target compound, analogues are identified with a decision algorithm that integrates three orthogonal similarity metrics, considering toxicologically relevant aspects:

1. **Structural similarity** based on structural fingerprints
2. **Biological similarity** based on biological fingerprints codifying the outcomes of high throughtput screening (HTS) assays from PubChem
3. **Metabolic similarity** based on the presence of common metabolic pathways between the target and the analogue as simulated by SyGMa library (https://github.com/3D-e-Chem/sygma).

   Structural filters based on the presence of maximum common substructures (MCS) and common functional groups can be also applied to narrow the chemical space for the analogue(s) search. Read-across prediction is made for the target based on the activity of analogue(s) identified with multiple similarity methods.

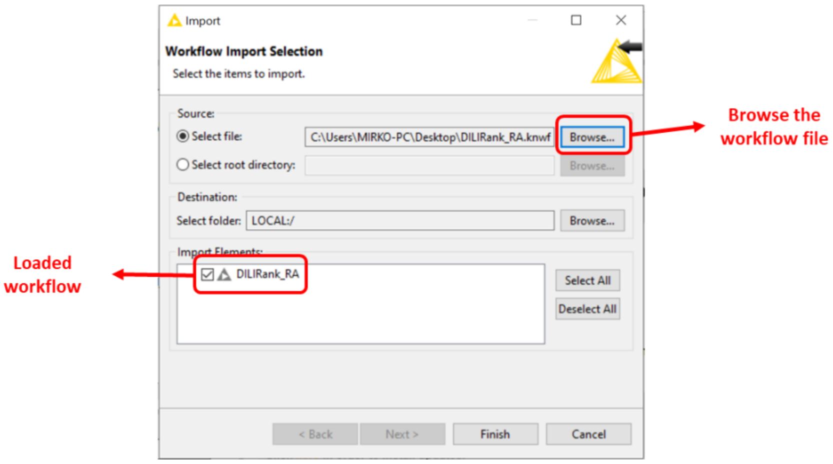
CONTENT OF THE REPOSITORY

- **DILIRank_RA.knwf**: the workflow to import in KNIME, using the DILIRAnk dataset as source for DILI classification.
- **ToxRef_RA.knwf:** the workflow to import, using ToxRef dataset as source for hepatotoxicity classification.
- **User's Guide.pdf:** the file contains instructions for the installation and use of the workflow.
- **README.md:** Markdown file to compile into html containing instructions for the installation and use of the workflow.
- **Amlodipine_DILI_RAX.png:** output of example that shows the list of analogue(s) as a pie chart.
- **Amlodipine_DILI_RAX.xlsx:** output of example that contains results of the workflow.

INSTALLATION

HOW TO INSTALL THE WORKFLOW

1. Install the last version of KNIME. It can be downloaded at https://www.knime.com/knime-analytics-platform (Windows, Linux/Unix, Mac OS X, 64bits).
2. Open KNIME.
3. Go to *"File -> Import KNIME Workflow"*.
4. Tick *"Select File:"* and go to *"Browse…"*. Select the *.knwf file of the workflow (DILIRAnk_RA_knwf or ToxRef_RA.knwf).
5. Click to *"Finish"*.



Workflow Import Selection

Select the items to import.

Source:

● Select file: C:\Users\MIRKO-PC\Desktop\DILIRank_RA.knwf

○ Select root directory:

Destination:

Select folder: LOCAL:/

Import Elements:

☑ DILIRank_RA

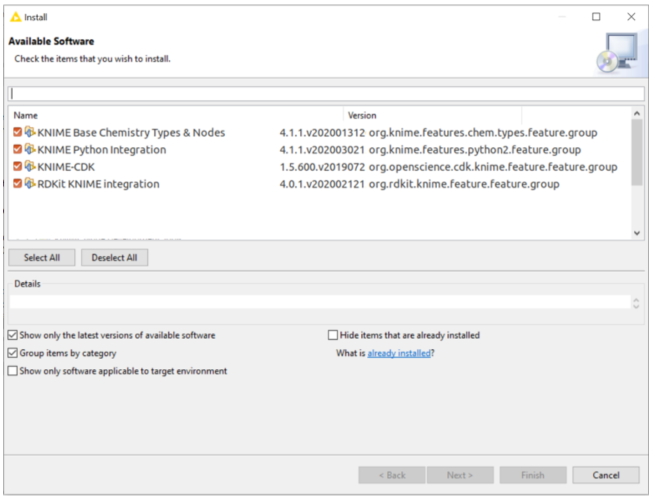
Browse the workflow file

Loaded workflow

6. The workflow now is in your "KNIME Explorer" menu on the left of the screen. Double click on the workflow to open it.
7. If some of the plugins used for the workflow are missing, a message will appear asking you to install the missing extensions. Click on *"Ok"*. The procedure will guide you in the installation of the missing extensions. Restart KNIME to make the new plugins working. Plugins that are necessary for running the workflow are:

- KNIME Base Chemistry Types & Nodes
- KNIME Python Integration
- KNIME-CDK
- RDKit KNIME integration

In case some of the plugins are not automatically identified, go to *File* → *Install KNIME Extensions*. The KNIME Integrations can be found manually by entering *their name* into the search box.



Available Software

Check the items that you wish to install.

Name	Version	
☑ KNIME Base Chemistry Types & Nodes	4.1.1.v202001312	org.knime.features.chem.types.feature.group
☑ KNIME Python Integration	4.1.1.v202003021	org.knime.features.python2.feature.group
☑ KNIME-CDK	1.5.600.v2019072	org.openscience.cdk.knime.feature.feature.group
☑ RDKit KNIME integration	4.0.1.v202002121	org.rdkit.knime.feature.feature.group

☑ Show only the latest versions of available software

☑ Group items by category

☐ Show only software applicable to target environment

☐ Hide items that are already installed

What is already installed?

HOW TO INSTALL PYTHON FOR KNIME

Additional indications to install Python for KNIME can be found at the following link: https://docs.knime.com/latest/python_installation_guide/index.html

1. Download and install Anaconda from https://www.anaconda.com/distribution/. Choose Anaconda with Python 3
2. Open the Anaconda Prompt (anaconda3).
3. Create and activate a new RDKit Python environment (my-rdkit-env) with the following commands:

```
conda create -c https://conda.anaconda.org/rdkit -n my-rdkit-env rdkit
conda activate my-rdkit-env
```

4. Use the following command to install python libraries required for the RAX workflow. Librares required are *sygma, bokeh, selenium, geckodriver, firefox, pandas, numpy*.

```
conda install sygma bokeh selenium geckodriver firefox -c conda-forge
```

Windows

5.1. Create a new *.txt file and copy/paste the following texts

```
@REM Adapt the folder in the PATH to your system
@SET PATH="<Anaconda Path>\Scripts";%PATH%
@CALL activate my-rdkit-env || ECHO Activating python environment failed
@python %*
```

Replace <*Anaconda Path*> with your Anaconda installation path (e.g. C:\ProgramData\Anaconda3).

6.1. Save the **.txt* file and change its extension to **.bat*. To change the extension, you need to change your PC settings to show hidden extensions of files, then manually rename the file from **.txt* to **.bat*. Place the **.bat* file in a directory of your choose.

Linux / Mac

5.2. Create a new *.sh file and copy/paste the following texts:

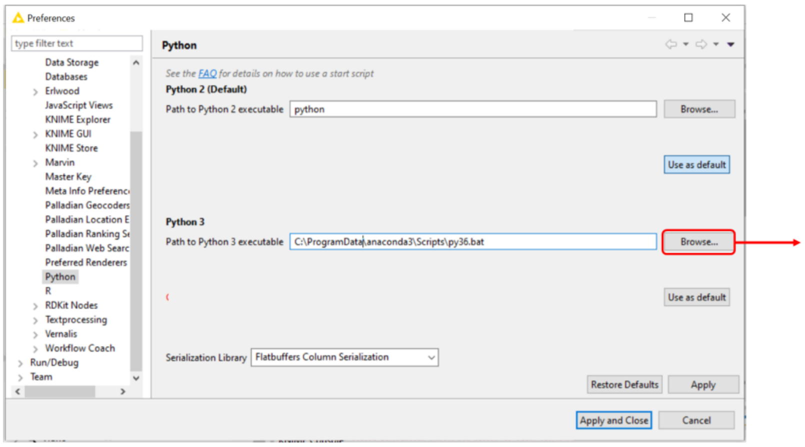
```
#! /bin/bash
# Start by making sure that the anaconda folder is on the PATH
# so that the source activate command works.
# This isn't necessary if you already know that
# the anaconda bin dir is on the PATH
export PATH="<Anaconda Path>/bin:$PATH"

conda activate my-rdkit-env
python "$@" 1>&1 2>&2
```

Replace <*Anaconda Path*> with your Anaconda installation path (e.g. /home/lubuntu/anaconda3).

6.2 Place the *.sh file in a directory of your choose. You additionally need to make the file executable (i.e. chmod gou+x <Name of your file>.sh)

7. From KNIME, go to *"File -> Preferences -> KNIME -> Python"*. Click on *"Browse…"* next to the *"Path to Python 3 executable"* search bar and browse the **.bat/*sh.* file.
8. Click to the "Apply and Close" button.



Python

See the FAQ for details on how to use a start script

Python 2 (Default)

Path to Python 2 executable: python

Python 3

Path to Python 3 executable: C:\ProgramData\anaconda3\Scripts\py36.bat

Serialization Library: Flatbuffers Column Serialization

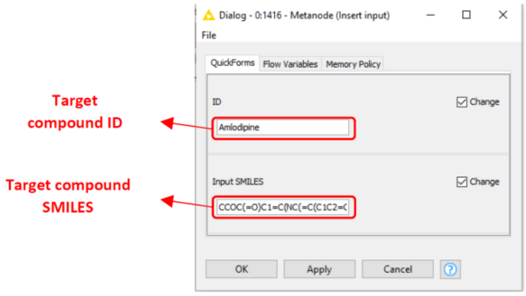
Browse the *.bat / *.sh file for Python 3 execution

USE THE WORKFLOW

HOW TO USE THE WORKFLOW

1. Load the input in the *"Insert input compound"* wrapped metanode:

- Double click on the metanode.
- Type the ID and the related input SMILES.
- Click on *"Ok"*.



ID ☑ Change
Amlodipine

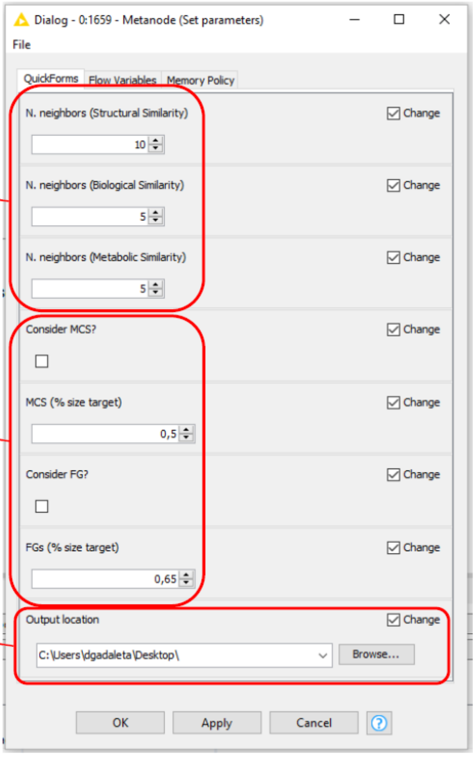
Input SMILES ☑ Change
CCOC(=O)C1=C(NC(=C(C1C2=C

2. Execute the workflow by clicking on the "*Execute all the executable nodes*" button on the top of the window or press Shift+F7.
3. Modify the settings in the second wrapped metanode, "*Set parameters*".

- Double click on the metanode.
- Select the number of analogue(s) to consider for each of the three similarity. By default, ten analogue(s) are retrieved based on structural similarity and five based on metabolic and biological similarity.
- A preliminary selection of source chemicals is also possible based on the presence of Maximum Common Substructures (MCS) and common functional group(s) (FGs) with the target. If appropriate, check the "Consider MCS?" and/or "Consider FG?" options to narrow the chemical space to search analogue(s). If the options are checked, it is possible to set the thresholds ("% size target") to include a chemicals in the analogue(s) selection.
- Select the path of the output file. By default, a new directory named *"RAX_results"* is created on the Desktop, that includes output files.
- Click on *"Ok"*.

4. Execute the workflow by clicking on the "*Execute all the executable nodes*" button on the top of the window or press Shift+F7. The "Set parameters" node will change from the *"Paused"* state to the *"Running"* state. If the first part of the workflow is executed successfully, the node will change to the "Executed" state. If the node return to the "Paused" state, some problems occurred during the procedure. These may be related, e.g. to an incorrect Python 3 configuration.

5. Press "*Save*" on the top-left of the window to save the workflow.



Dialog - 0:1659 - Metanode (Set parameters)

File

QuickForms | Flow Variables | Memory Policy

N. neighbors (Structural Similarity): 10 ☑ Change

N. neighbors (Biological Similarity): 5 ☑ Change

N. neighbors (Metabolic Similarity): 5 ☑ Change

Consider MCS? □ ☑ Change

MCS (% size target): 0,5 ☑ Change

Consider FG? □ ☑ Change

FGs (% size target): 0,65 ☑ Change

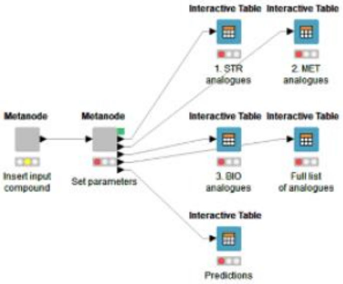
Output location: C:\Users\dgadaleta\Desktop\ Browse... ☑ Change

OK | Apply | Cancel

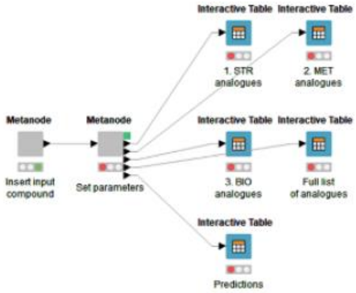
Set number of analogue(s) for each similarity

Check options to narrow the chemical space for analogue(s) search

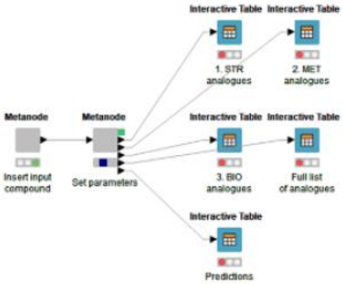
Set the location of output files



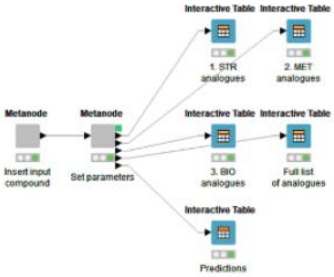
1. Select the input compound



2. Set preferences for the analogue(s) search procedure



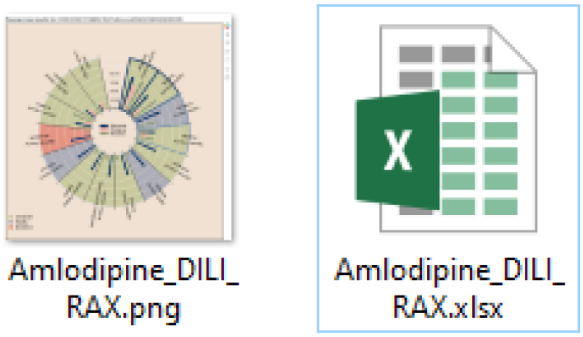
3. The workflow is running...



4. The search is succesfully ccompleted. Check the selected compounds from the Tables or from the output files (.xlsx and .png)

HOW TO READ THE OUTPUT

1. To inspect the list of selected analogue(s) within KNIME, select the *"Interactive Table"* nodes, then click on the "*Open the first view of the selected node*" button on the top of the page or press F10. To inspect the RAX plot, right click on the "Set parameters" metanode, then select the *"0:Port 1"* option from the drop-down menu.
2. To inspect the results reported in the output files, go to the *"RAX_results"* directory that is present in the output location specified before. By default, the *"RAX_directory"* is created on the Desktop. The directory include two files:



.png file that shows the list of analogue(s) as a pie chart. Each slice of the chart represents a different analogue, with the background colour indicative of the analogue's activity and histogram bars that give estimation of the four similarities with respect to the target (i.e., blue for structural, pink for biological and green for metabolic similarity). Outlined slices indicate analogues included in multiple similarity lists (i.e., dark blue for three lists and light blue for two lists).

.xlsx file including four different sheets:

- **Structural analogues:** list of closest neighbour compounds based on structural similarity. For each analogue, the structural similarity with respect of the target is reported. By default, ten analogues are selected. If checked, the presence of MCS and common FGs are verified here (yes/no), and a list of functional groups identified for each analogue is reported.
- **Biological analogues:** list of closest neighbour compounds based on biological similarity. For each analogue the following information are reported: the number of biological assays that are negative (*Cn*) or positive (*Cp*) for both the target and the analogue; the number of biological assays that are positive for the target but negative for the analogue (*Tp*); the number of biological assays that are negative for the target but positive for the analogue (*Sp*); the biological similarity (*BioSim*); the reliability associated to the BioSim (i.e., based on the number of assays in common between the target and the analogue); the final BioSim value weighted on the reliability.
- **Metabolic analogues:** list of closest neighbour compounds based on metabolic similarity. For each analogue the following information are reported: the metabolic

similarity value; the number and the list of metabolic pathways that are present only in the target, only in the analogue or in both compounds; the number and the list of common metabolites between the target and the analogue.

- **Full_list_analogues:** complete list of all the previous analogues that includes all the information above. For each analogue, the list(s) in which it appears and the relative ranks are indicated (e.g. STR1, BIO3 is the top-ranked structural analogue and the third-ranked biological analogue). Analogue(s) included in multiple similarity lists are considered more suitable with respect of those included in a single list.
- **Predictions:** reads-across predictions. Prediction of activity for the target is made by averaging activities of analogue(s) included in at least one, two or three similarity lists (*"group_threshold"*). For each prediction, the number of analogues used to sustain the prediction and their activity is indicated. If structural filters based on MCS and common functional groups are used, this is specified in the *"Source_filter"* column ("*none*" is displayed when no filters are used).

REFERENCE

Further details on the algorithms used to calculate similarities can be found in the reference publication:

CONTACT

Domenico Gadaleta, PhD

Computational Toxicology Unit

Laboratory of Environmental Chemistry and Toxicology

Department of Environmental Health Sciences

Istituto di Ricerche Farmacologiche Mario Negri IRCCS

Via Mario Negri 2, 20156 Milano, Italy

Tel. +39 02 3901 4396 e-mail: domenico.gadaleta@marionegri.it