

An Integrated Data-Driven Framework for Predictive Safety and Sustainability Assessment

Harry Sarimveis

National Technical University of Athens

**ANTHOS'26 Solutions Session 4:
Projects solutions to the needs of (scientific) Implementors**

11 March 2026



THE PINK PROJECT HAS RECEIVED FUNDING FROM THE EUROPEAN UNION'S HORIZON EUROPE RESEARCH AND INNOVATION PROGRAMME UNDER GRANT AGREEMENT NO. 101137809.

ASSOCIATED PARTNERS (I.E. (A) SWISS PARTNERS AND (B) UK PARTNERS) HAVE RECEIVED NATIONAL FUNDING FROM (A) THE SWISS STATE SECRETARIAT FOR EDUCATION, RESEARCH AND INNOVATION (SERI), AND (B) INNOVATE UK.

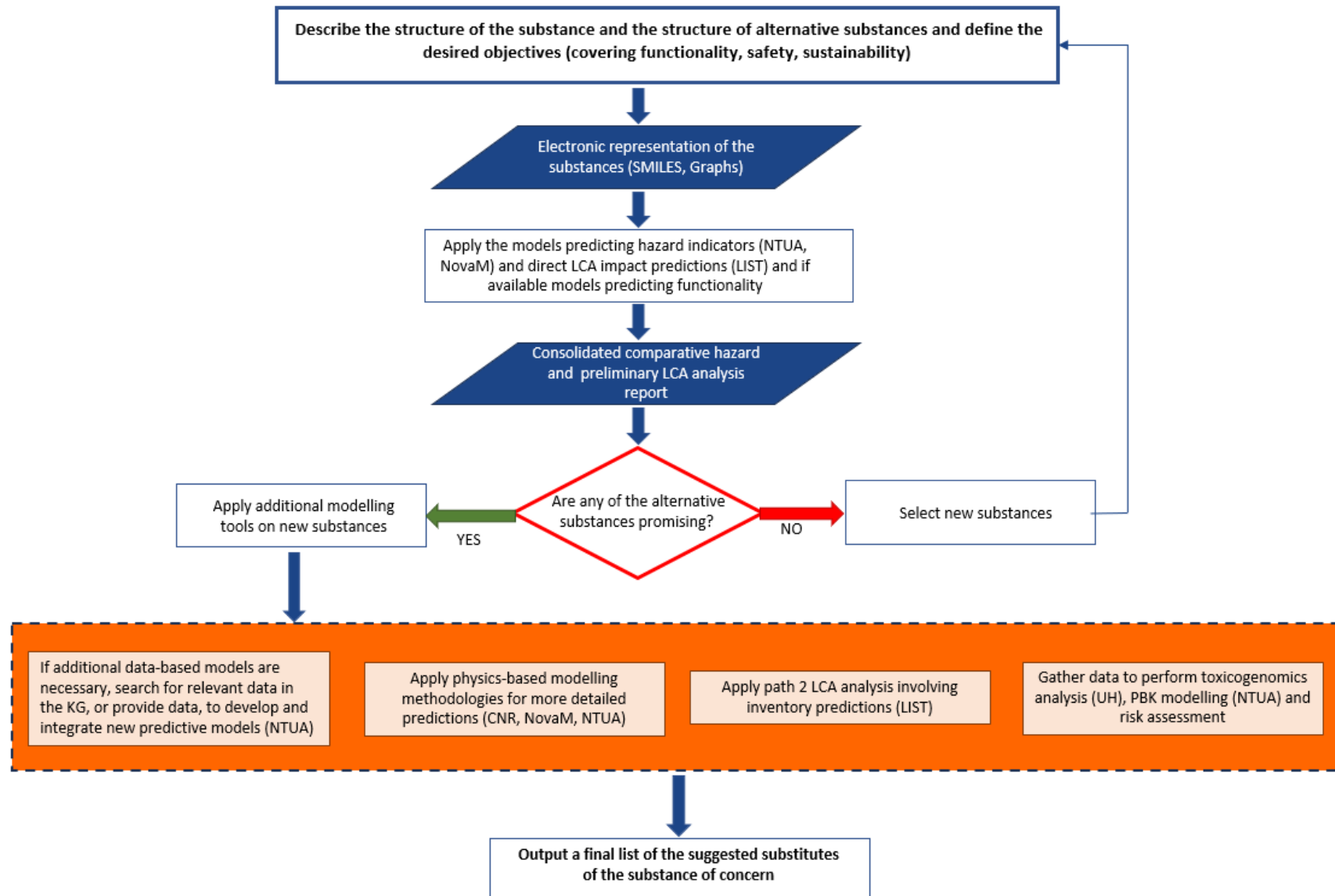
PINK's high-level objectives

- Develop innovative modelling and simulation approaches addressing industrial SSbD needs
- Make the software accessible to SMEs and industry through an open innovation platform
- Validate the platform on PINK Developmental Case Studies and Industrial Demonstrators provided by SMEs and large industries
- Fully implement open science and FAIR principles contributing to establishing a European chemicals and materials data, modelling and software ecosystem
- Strengthen knowledge transfer through collaboration and exploiting synergies
- Boost the innovative capacity of SMEs and industry and make them more agile to respond to external and internal influences

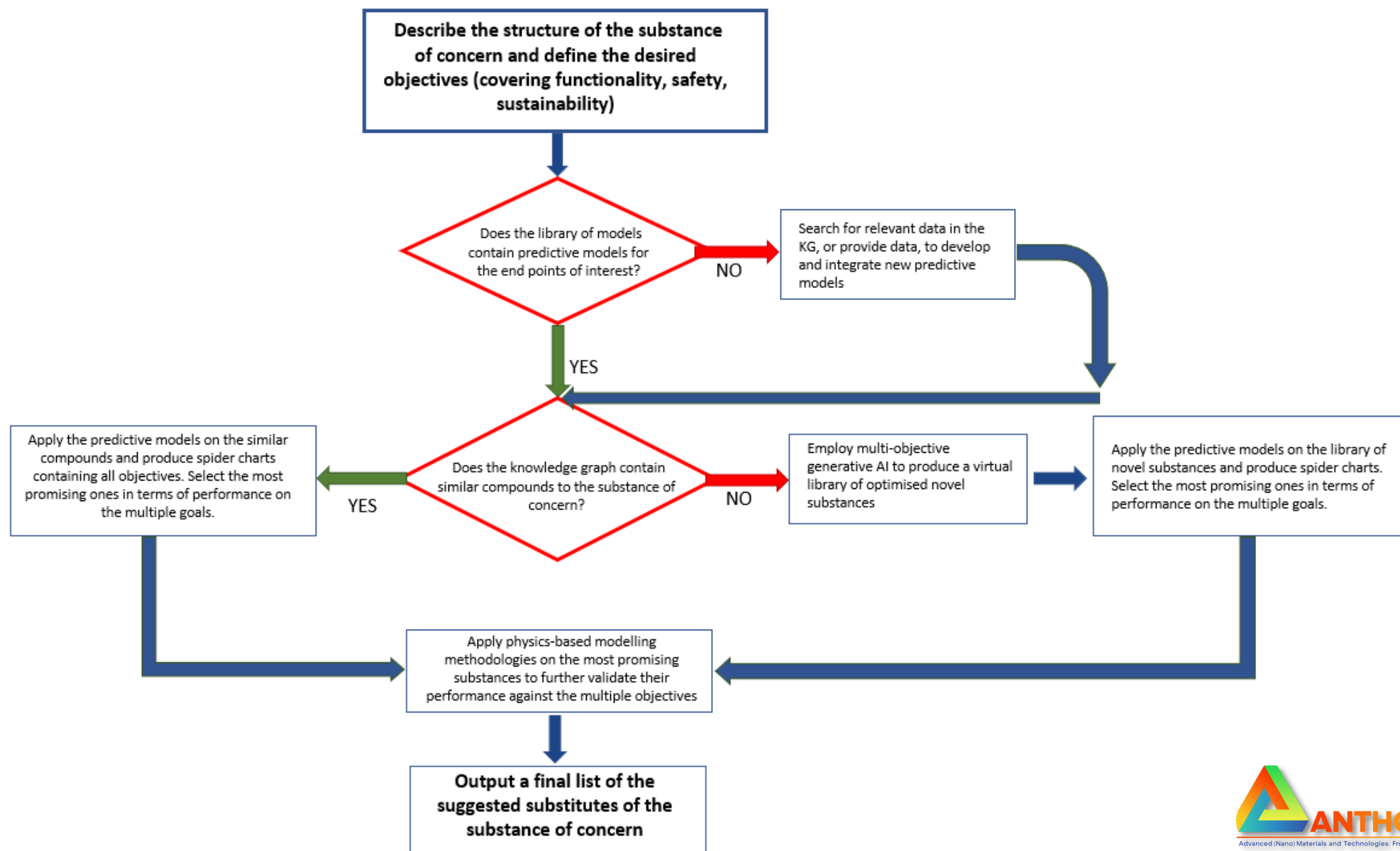
Modelling objectives

The goal is to develop a **comprehensive toolset that will generate the input for the PINK Decision-Support System (DSS)**, merging both data-driven and physics-based modelling approaches with specialised cheminformatics, bioinformatics, and Life-Cycle Assessment (LCA) tools and biokinetics simulations and integrated with data search and AI technologies.

Development of Decision Support Systems

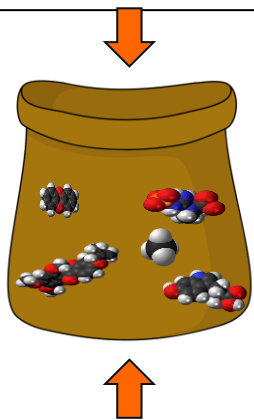


Development of Decision Support Systems



Stage-gated SSbD assessment

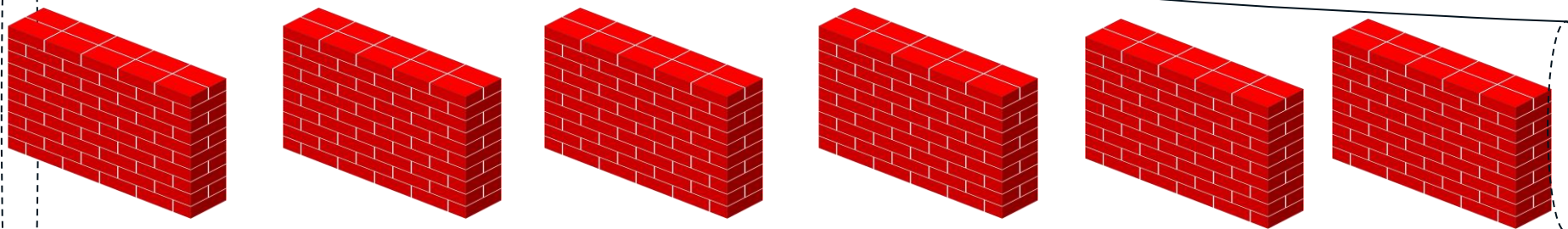
Similarity search,
Literature search,
Mol. generation
(Generative AI)



Expert-guided
compound
recommendations

Cost-effective assessment strategy

- prioritization of assessment stages by computational cost
- searching over wide space but filtering out early



Shortlist
for expert
evaluation

Jaqpot cloud platform

Welcome to Jaqpot!

Jaqpot is your all-in-one, open-source platform for creating, exploring, and deploying machine learning models. Whether you're a data scientist, researcher, or developer, we've got the tools you need to bring your models to life.

[Getting started](#)[Learn more →](#)

Developed by the Unit of Process Control and Informatics at the National Technical University of Athens

Create a model

Learn how to deploy your ML models using our SDKs



Explore models

Discover a wide range of pre-built models across various categories



My Models

Access and manage your uploaded models or models that others shared with you



My Results

View and analyze your past experiments and results



jaqpot

Register



Register



Jaqpot cloud platform

app.jaqpot.org/dashboard/explore

Jaqpot
Explore models

NAVIGATION

- Your models
- Your results
- Settings
- Sign In

LLMS
Large language models for text generation and analysis

GENERATIVE
Models designed to generate novel data, such as molecules,...

IMAGE-ANALYSIS
This category includes AI-driven models or ImageJ-algorithms...

READ-ACROSS
Read-across is a computational approach used for property...

PFAS
Predictive models for per- and polyfluoroalkyl substances

NANOMATERIALS
Predictive models for nanomaterials

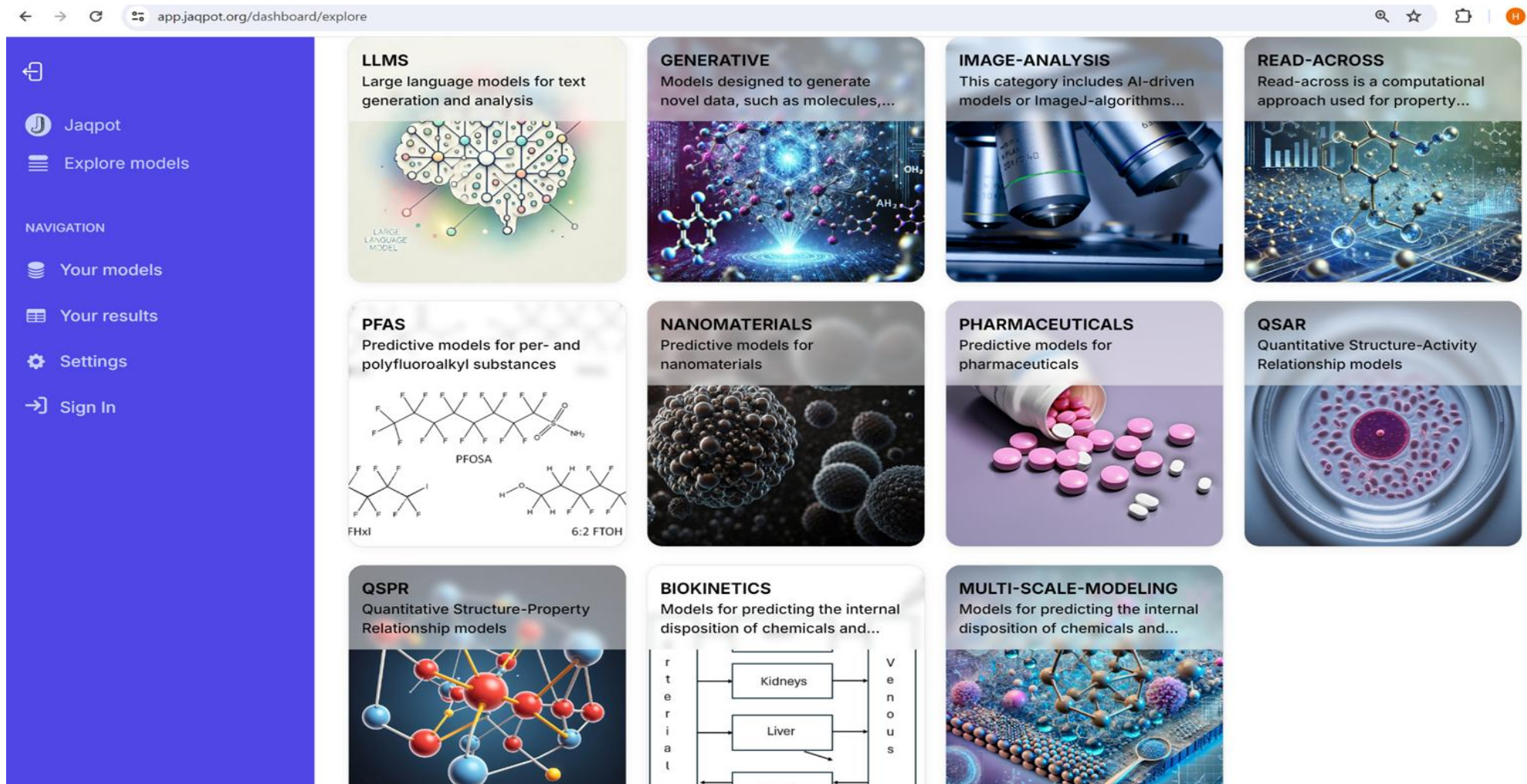
PHARMACEUTICALS
Predictive models for pharmaceuticals

QSAR
Quantitative Structure-Activity Relationship models

QSPR
Quantitative Structure-Property Relationship models

BIOKINETICS
Models for predicting the internal disposition of chemicals and...

MULTI-SCALE-MODELING
Models for predicting the internal disposition of chemicals and...



PINK Organisation on the Jaqpot cloud platform

PINK

 Harry Sarimveis
@hsarimv@mail.ntua.gr

[Description](#) [People](#) [Shared](#) [Edit](#)

The PINK EU project aims at providing integrated computational approaches for the introduction of safe-and-sustainable-by-design chemicals and materials into the market. PINK will develop a Tiered Approach, where each development cycle will start with mining and creating data on the functionality, safety, and sustainability, as well as cost/economic feasibility, which will be provided simultaneously as input for the decision support workflow. The PINK Tiered Approach will categorise all computational methods into three tiers that parallel the material development TRLs, with respect to their data needs, throughput, and confidence levels. At earlier development stages (ideation/design phase of material development), where less data is available, high-throughput computational approaches of low tiers will be applied to avoid limiting the exploration and missing promising design routes due to time constraints (human and computer) and costs.

For more information visit <https://pink-project.eu/>

Title	Model Description
QSAR toolbox Calculator integration	This service provides access to the calculators included in the QSAR toolbox
QSAR Toolbox Model Integration	This service provides access to the QSAR models included in the QSAR toolbox
QSAR Toolbox Profiler integration	This service provides access to the profilers included in the QSAR toolbox
DILI Prediction Model	This model predicts if a substance induces liver injury
IndusChemFate PBTK	IndusChemFate is a PBTK model designed to estimate blood and urine concentrations of multiple chemicals and their metabolites given specific exposure scenarios.
IndusChemFate Lite PBTK	A lite version of IndusChemFate PBTK model, limiting metabolism to liver-only and single metabolites, reducing complexity and parameter requirements compared to the original model

<https://app.jaqpot.org/dashboard/organizations/PINK/description>

Application Programming Interfaces

The screenshot displays the Swagger UI for the Jaqpot API. The browser address bar shows the URL `api.jaqpot.org/swagger-ui/index.html`. The Swagger logo and version `1.0.0` are visible in the top left, while the endpoint `/v3/api-docs` and an `Explore` button are in the top right. The main content area describes the Jaqpot API as a modern RESTful API for model management and prediction services, built using Spring Boot and Kotlin. It includes links to the Jaqpot website, email contact, and the Apache 2.0 license. Below this, a 'Servers' section shows a dropdown menu with the selected server `https://api.jaqpot.org - Generated server url` and an `Authorize` button. The API endpoints are organized into three sections: **dataset**, **api-keys**, and **organization**. Each section lists endpoints with their HTTP methods, paths, and descriptions. The **dataset** section includes endpoints for getting datasets by model ID, creating a dataset, and getting datasets by user ID. The **api-keys** section includes endpoints for getting all API keys, creating an API key, deleting an API key, and updating an API key. The **organization** section includes an endpoint for getting all organizations for a specific user.

Method	Path	Description
GET	<code>/v1/user/models/{modelId}/datasets</code>	Get Datasets by Model ID
POST	<code>/v1/user/models/{modelId}/datasets</code>	Create a dataset
GET	<code>/v1/user/datasets</code>	Get Datasets by User ID
GET	<code>/v1/user/api-keys</code>	Get All API Keys for the User
POST	<code>/v1/user/api-keys</code>	Create an API Key for the User
DELETE	<code>/v1/user/api-keys/{key}</code>	Delete an API Key
PATCH	<code>/v1/user/api-keys/{key}</code>	Update API Key
GET	<code>/v1/organizations</code>	Get all organizations for a specific user

Example: Ready Biodegradability Classification Model

Graph Neural Network (GAT) for ReadyBiodegradability Classification

giannis savvas
@john_savvas about 2 months ago TORCHSCRIPT

Description Features Predict Metrics Discussion

Choose Your Prediction Input Method

☒ Fill out the form or ☐ Upload a CSV file (max 100 rows)

SMILES *

N#C/C(C(=O)OCC(CC)CCCC)=C(/c1ccccc1)c2ccccc1

Submit

Result

ID 10584

Success

less than a minute ago less than 5 seconds

Export CSV

Total 1 result

SMILES	ReadyBiodegradability	Probabilities
<chem>N#C/C(C(=O)OCC(CC)CCCC)=C(/c1ccccc1)c2ccccc1</chem>	0	0: 0.867 1: 0.133

Dashboard > Models > Graph Neural Network (GAT) for ReadyBiodegradability Classification

Graph Neural Network (GAT) for ReadyBiodegradability Classification

giannis savvas
@john_savvas over 1 year ago TORCHSCRIPT

Description Features Predict Metrics Discussion

Data Sources: <https://zenodo.org/records/3540701>, <https://zenodo.org/records/8255910>.

The total data (6,849) were split into train-validation-test with a 80-10-10 while keeping the same class balance.

A Graph Neural Network (GAT) using both node and edge attributes was used to train the model and evaluate it on the validation set. Hyperparameter tuning was implemented to obtain the best possible validation loss score.

The final model has a Balanced Accuracy score of 0.86 and ROC-AUC score of 0.92 in the test set.

Input SMILES to get the prediction

```
[7]: input_data = [{"SMILES": "N#C/C(C(=O)OCC(CC)CCCC)=C(/c1ccccc1)c2ccccc1"}]
```

Get prediction though SDK

```
[8]: prediction = jaqpot.predict_sync(model_id=1928, dataset=input_data)
print(prediction)

[{'jaqpotMetadata': {'probabilities': [0.867, 0.133], 'jaqpotRowId': '0'}, 'ReadyBiodegradability': 0.0}]
```

AI agents for SSBD assessment

Verification Required: Always verify predictions through experimental validation or consultation with domain experts before practical application.

Supported by: This work is supported by the [PINK Project](#) - a Horizon Europe initiative for Safe-and-Sustainable-by-Design computational approaches for chemicals and materials.

QSARion Chemical Property Prediction Agent

Chat Message (Disclaimer: Messages are being monitored for debugging purposes. Do not enter any personal information.)

what is the solubility of ethane?

Step 1 – user input



Step 1

Thought: To find the solubility of Ethane, I will first retrieve the SMILES string for Ethane using the `retrieve_smiles_tool`. Then, I will use the `qsartoolbox_get_property_value_tool` with the appropriate model to predict the solubility.

Code:

```
# Retrieve the SMILES string for Ethane
ethane_smiles = retrieve_smiles_tool(compound_name="Ethane")
print("SMILES for Ethane:", ethane_smiles)
```

⌵ Used tool python_interpreter

```
# Retrieve the SMILES string for Ethane
ethane_smiles = retrieve_smiles_tool(compound_name="Ethane")
print("SMILES for Ethane:", ethane_smiles)
```

⌵ Execution Logs

SMILES for Ethane: CC

Last output from code snippet:

None

Step 1 | Input-tokens:8,272 | Output-tokens:107 | Duration: 13.61

Step 2 – trigger appropriate models

Step 2

Thought: I have the SMILES string for Ethane, which is "CC". Now, I will use the `qsartoolbox_get_property_value_tool` to predict the solubility of Ethane using the "Water Solubility (fragments) (EPISUITE)" model.

Code:

```
# Predict the solubility of Ethane using the Water Solubility (fragments) (EPISUITE) model
solubility_result = qsartoolbox_get_property_value_tool(smiles="CC", qsartoolbox_qsar_model="Water Solubility (fragments) (EPISUITE)")
print("Solubility result:", solubility_result)
```

Calling tools:

```
[{'id': 'call_2', 'type': 'function', 'function': {'name': 'python_interpreter', 'arguments': '# Predict the solubility of Ethane using the Water Solubility (fragments) (EPISUITE) model\nsolubility_result = qsartoolbox_get_property_value_tool(smiles="CC", qsartoolbox_qsar_model="Water Solubility (fragments) (EPISUITE)")\nprint("Solubility result:", solubility_result)'}}
```

✖ Used tool python_interpreter

```
# Predict the solubility of Ethane using the Water Solubility (fragments) (EPISUITE) model
solubility_result = qsartoolbox_get_property_value_tool(smiles="CC", qsartoolbox_qsar_model="Water Solubility (fragments) (EPISUITE)")
print("Solubility result:", solubility_result)
```

📄 Execution Logs

Solubility result: {'DomainResult': 'Undefined', 'DataType': 'Qsar prediction.', 'RigidPath': 'Physical Chemical Properties#Water solubility', 'Endpoint': 'Water Solubility', 'MetaData': ['Prediction approach=External SAR/QSAR', 'Domain=Undefined (Water Solubility (fragments) (EPISUITE))', 'Endpoint=Water Solubility', 'SAR/(Q)SAR=Water Solubility (fragments) (EPISUITE) (v1.0)', 'Domain status=Undefined', 'Primary group=N/A', 'Primary group with metabolism=N/A', '(Q)SAR options='], 'Value': **383.29**, 'Qualifier': '', 'MinQualifier': '', 'MaxQualifier': '', 'Unit': 'mg/L', 'Family': 'Mass concentration', 'Name': 'Ethane', 'ChemId': '3c6c8350-508d-1765-d8b2-f0c9e8f71c12', 'jaqpotMetadata': {'jaqpotRowId': '0'}}
Last output from code snippet:
None

Step 2 | Input-tokens:8,489 | Output-tokens:263 | Duration: 23.99

Step 3 - results

Step 3

Thought: The solubility of Ethane has been predicted to be 383. 29 mg/L using the "Water Solubility (fragments) (EPISUITE)" model from the QSAR Toolbox.

Code:

```
final_answer(answer="383. 29 mg/L")
```



⌵ ⚙ Used tool python_interpreter

Step 3 | Input-tokens:9,140 | Output-tokens:72 | Duration: 12.52

Final answer:

383. 29 mg/L

QSAR Toolbox and EPA CCTE

Experimental and calculated data points are provided from the QSAR toolbox API and the EPA's CCTE API.

They follow a different rationale and their output is formatted to a different schema.

A common schema was structured to join information while retaining information from both sources.



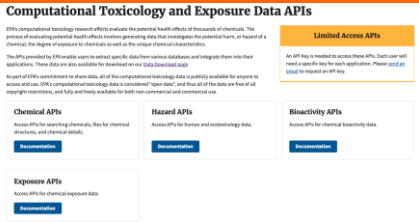
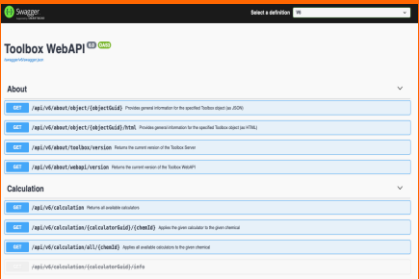
OECD QSAR Toolbox
NTUA-hosted API server

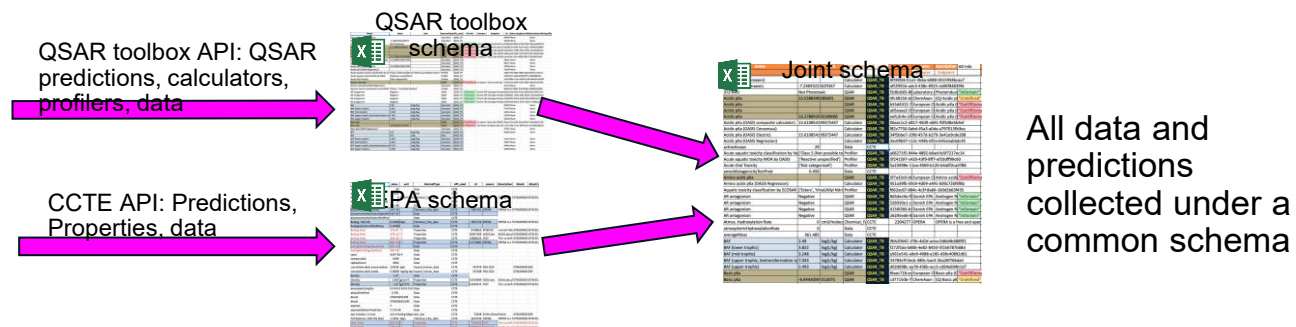
<http://54.155.131.85/swagger/index.html>



Computational Toxicology
and Exposure Data APIs

<https://api-ccte.epa.gov/docs/index.html>





Example of generating a QSAR Toolbox prediction through Jaqpot

Dashboard > Models > QSAR Toolbox Model Integration

QSAR Toolbox Model Integration

 Jaqpot Team
@jaqpot-team 3 months ago  QSAR_TOOLBOX_QSAR_MODEL

[Description](#) [Features](#) [Predict](#) [Metrics](#) [Discussion](#)

Choose Your Prediction Input Method

☒ Fill out the form or ☐ Upload a CSV file (max 100 rows)

A Smiles representation *

N#C/C(C(=O)OCC(CC)CCCC)=C(/c1ccccc1)c2ccccc1

QSAR Model *

VEGA - Fish Acute (LC50) Toxicity model (

Submit

Result

ID 10586 [🔗](#)

Success

 less than a minute ago  less than 20 seconds

[📄](#) Export CSV

Total 1 result

Rows per page
25

A Smiles representation	QSAR Model	Value	Unit	Endpoint	Donator	DomainResult	Name
<chem>N#C/C(C(=O)OCC(CC)CCCC)=C(/c1ccccc1)c2ccccc2</chem>	934036e4-c346-4aca-aec3-dc3ae1c85980	0.0894259385321039	mg/L	LC50		OutOfDomain	

< 1 >

QSAR: performance, applicability, uncertainty and explainability

1

Performance

Model-level (fit & predictivity)

Regression: R^2 , RMSE
Classification: AUC, MCC
CV + external test

Output: $R^2=0.72$, RMSE=0.35

2

Applicability

Compound-level (in vs out)

kNN similarity/coverage
Leverage check
In/out badge + reason

Output: In-domain (sim=0.81)

3

Uncertainty

Prediction-level (how sure?)

Prediction interval (e.g.,
90%)
Conformal (optional)
Used in ranking

Output: Pred=3.2 (PI: 2.6–3.9)

4

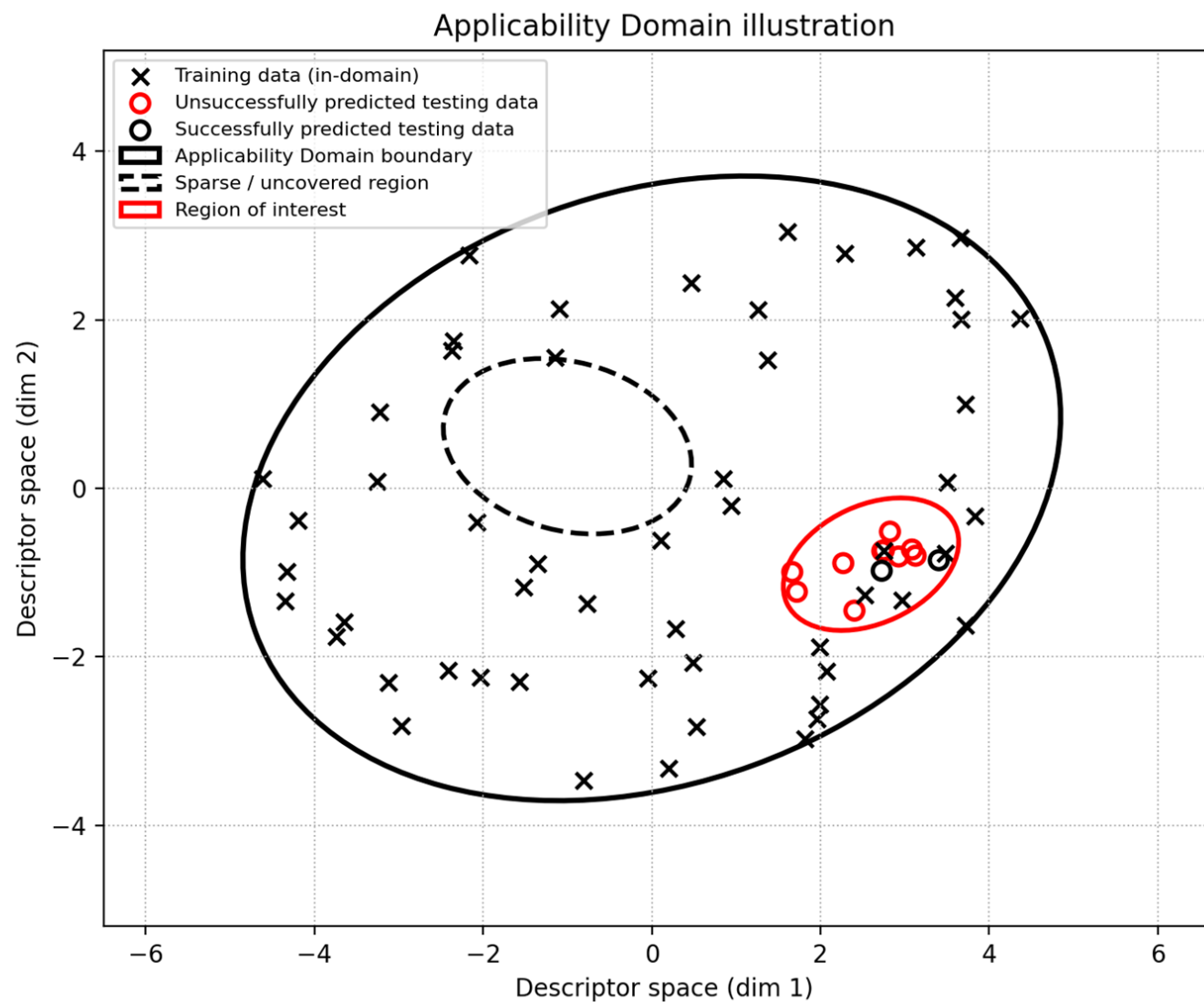
Explainability

Why did the model say this?

Top drivers/features
Simple 'reason' sentence

Output: Key drivers

Applicability domain illustration



The PINK matrix

SSbD Framework - Indicator	Type of Hazard	Type of model	Expressed via	More info about readout	Inputs to the model	Reference	Modeller	Ready for Modelling Data File	Source of data	Model URL	Report (QMRF, MODA etc.)	Additional information	Id
Carcinogenicity													
Carcinogenicity (genotox and nongenotox) alerts by ISS	Human Health	QSAR - Categorical	Categories include: ['Aliphatic halogens (Genotox); Alkyl		SMILES	https://qsartoolbox.org/	ISS, Istituto Superiore di			https://qsartoolbox.org/d/1A-PrNyn-XGgJyLs_c9xalITGxbC5qHoB			2256b12d-5ef8-49a2-af9a-
FDA RCA Cancer Female Mouse - Danish QSAR DB CASE Ultra	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative', 'Equivocal']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1UfVikwcN9Nii3T1g9D463njkgq5c5Gr/view?us			e2078536-1427-4d94-88dc-
FDA RCA Cancer Female Mouse - Danish QSAR DB Leadscape model	Human Health	QSAR - Categorical	No units, 1 for positives and 0 for negatives.		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1exYfEJEfMego3NmXu-TdkUF0V-			12b85d79-c468-4da6-8e2e-
FDA RCA Cancer Female Rat - Danish QSAR DB CASE Ultra	Human Health	QSAR - Categorical	Categories include: ['Equivocal', 'Positive',		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1fsXTKkRz2ITCoac5MfUou84IbOpnZcN/view?us			eb6861f5-178d-4109-8bc3-
FDA RCA Cancer Female Rat - Danish QSAR DB Leadscape model	Human Health	QSAR - Categorical	No units, 1 for positives and 0 for negatives.		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1zVafqI89v0Fj3r8YO-nbYk-			892add19-9efa-4011-b8dc-
FDA RCA Cancer Male Mouse - Danish QSAR DB CASE Ultra	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative', 'Equivocal']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1BUJe35m02P9pJUXxYOVDfrGpXOtRpw5G/view			adc64575-36e0-4eb5-86df-
FDA RCA Cancer Male Mouse - Danish QSAR DB Leadscape model	Human Health	QSAR - Categorical	No units, 1 for positives and 0 for negatives.		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1mQwCqkix8Ch0xAgpnGMCU-			41b3e7be-562a-44a2-82f6-
FDA RCA Cancer Male Rat - Danish QSAR DB CASE Ultra	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/14wdyHTIDBRAETU0hAMPKSTcmUKym0L-			f7395d96-ddfb-4b68-b8cd-
FDA RCA Cancer Mouse - Danish QSAR DB CASE Ultra model	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative', 'Equivocal']		SMILES	https://qsartoolbox.org/	Danish QSAR Group at DTU			https://qsartoolbox.org/d/1ufxfvQXqR5YxxOutfhqalc3-			9bb94df1-4788-4f17-bbd9-
FDA RCA Cancer Mouse - Danish QSAR DB Leadscape model	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1WgdOMBe8MaL78HMyppaw6vMz7pov6wYR-			c11ced21-23d1-4574-aa17-
FDA RCA Cancer Rat - Danish QSAR DB CASE Ultra model	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative', 'Equivocal']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1k8uI3cGds6z0Kk3prODpHavzk6zHcSg/view?			3fa56226-315a-4e18-9021-
FDA RCA Cancer Rat - Danish QSAR DB Leadscape model	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1KY4PhiceQvPvKHPDhfM63DLMJlqHa_nE/view			7ef3e1a1-d10a-4278-bd90-
FDA RCA Cancer Rodent - Danish QSAR DB CASE Ultra model	Human Health	QSAR - Categorical	Carcinogenicity in rodent liver (exclusively), positive or		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1-2-9IRc5lyfml-			1e7d360f-85da-4957-84cf-
FDA RCA Cancer Rodent - Danish QSAR DB Leadscape model	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1-2bnx-k7N6hXl2TPpYGNlaxfTz-			1d743a96-53b9-428f-bae0-
Liver Specific Cancer in Rat or Mouse - Danish QSAR DB battery	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1evay6urEoY-dqnTl3556PdINa8SGEp5B/			c40d6a42-80bd-4917-8878-
Liver Specific Cancer in Rat or Mouse - Danish QSAR DB CASE	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1Z-4OmG9cccnIIPhVWtch5g			0a590899-12a7-466c-a8dc-
Liver Specific Cancer in Rat or Mouse - Danish QSAR DB	Human Health	QSAR - Categorical	Categories include: ['Positive', 'Negative']		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1Xlfu3C3t0QUH8Q1ngklx2hknRlVvPfz/view?			a329e3be-5b40-4fd0-b5ad-
Liver Specific Cancer in Rat or Mouse - Danish QSAR DB SciQSAR	Human Health	QSAR - Categorical	Positive or Negative		SMILES	https://qsartoolbox.org/	Danish EPA			https://qsartoolbox.org/d/1BWTPS_5GONAtIOjdAlK32ht1RhVcNFH/view?			a86720fe-34f5-4bbc-a3f8-
VEGA - Carcinogenicity inhalation classification model (IRFMN)	Human Health	QSAR - Categorical	0 (non-carcinogen), 1 (carcinogen)		SMILES	https://qsartoolbox.org/	Istituto di Ricerche			https://qsartoolbox.org/d/1Tth-j9K0xe9UPx_QDQ4Vqp6V	Appears in Acute Toxicity: Inhalation and Carcinogenicity		494572f1-eaca-49ba-95d4-
VEGA - Carcinogenicity model (CAESAR)	Human Health	QSAR - Categorical	0 (non-carcinogen), 1 (carcinogen)		SMILES	https://qsartoolbox.org/	Istituto di Ricerche			https://qsartoolbox.org/d/1ByfOIHJbOtppdnZxDFI8ldOTY01VbyTz/view?			edaafece-0b6d-49a3-9801-
VEGA - Carcinogenicity model (IRFMN-Antares)	Human Health	QSAR - Categorical	0 (non-carcinogen), 1 (carcinogen)		SMILES	https://qsartoolbox.org/	Istituto di Ricerche			https://qsartoolbox.org/d/1e00Zj8pDSj4Ur8On_skdmcI-			0665aa46-d259-4194-9f4b-
VEGA - Carcinogenicity model (IRFMN-ISSCAN-CGX)	Human Health	QSAR - Categorical	0 (non-carcinogen), 1 (carcinogen)	TOX 7.7. Carcinogenicity (in	SMILES	https://qsartoolbox.org/	Istituto di Ricerche			https://qsartoolbox.org/d/1bdAr03gfzUFh4yuf34ah4-			1e68bcdf-b2b8-4d84-b45e-
VEGA - Carcinogenicity model	Human Health	QSAR - Categorical	0 (non-carcinogen), 1		SMILES	https://qsartoolbox.org/	Istituto di			https://qsartoolbox.org/d/1ZVQuvP37ivW1h			e9b1d082-4358-

1. Hazard Assessment

2. Production & processing phase

3. Final Application Phase

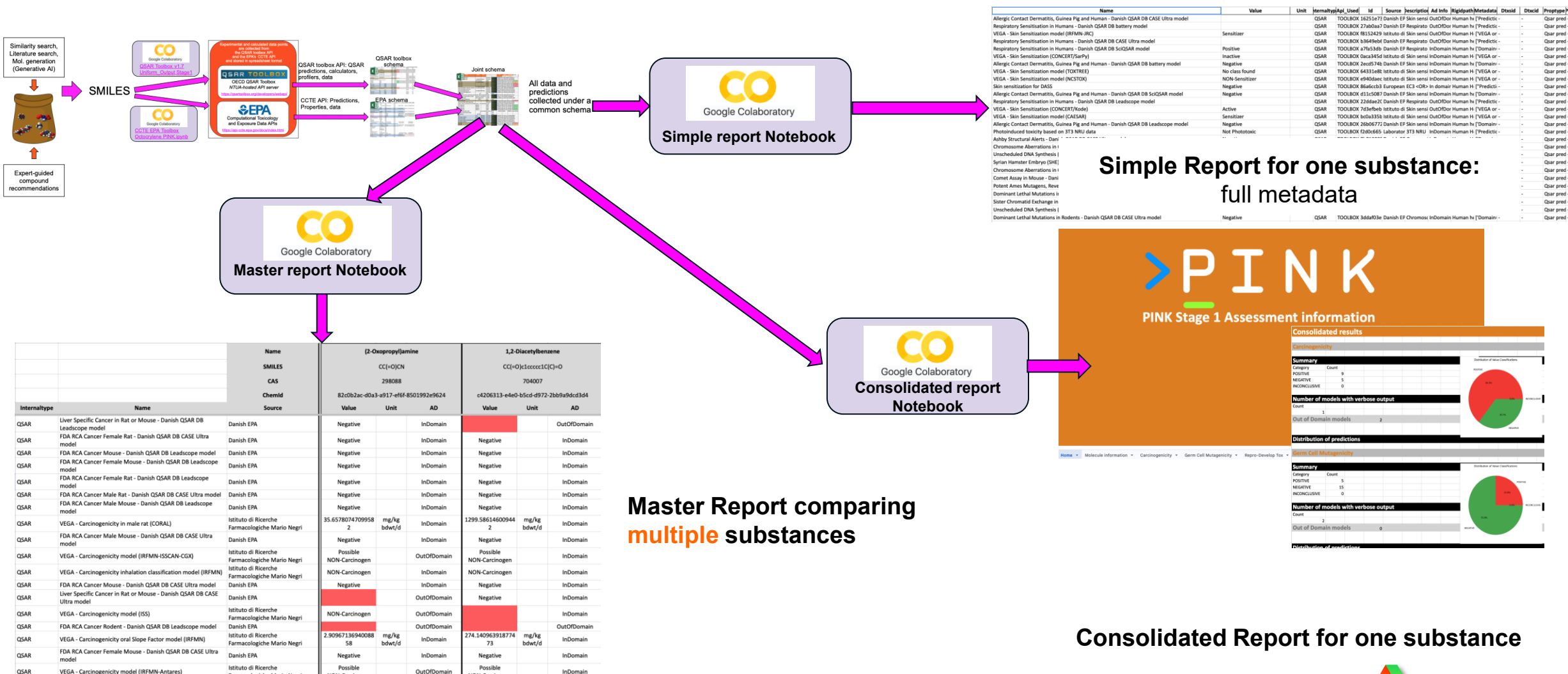
4. Environmental sustainability

5. Social-Economic Sustainability

Functionality

+

Workflow for SSbD assessment



Consolidated report

Consolidated results

Carcinogenicity

Summary

Category	Count
POSITIVE	9
NEGATIVE	5
INCONCL	0

Number of models with verbose output

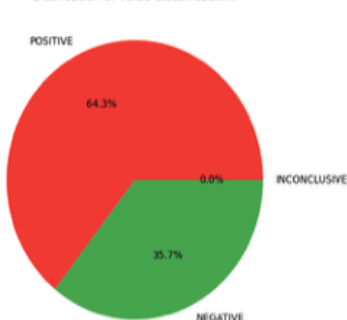
Count

1

Out of Domain model:

2

Distribution of Value Classifications



Distribution of predictions

Germ Cell Mutagenicity

Summary

Category	Count
POSITIVE	5
NEGATIVE	15
INCONCL	0

Number of models with verbose output

Count

2

Out of Domain model:

0

Distribution of Value Classifications



Distribution of predictions

Reproductive Developmental Toxicity

Summary

Category	Count
POSITIVE	0
NEGATIVE	7
INCONCL	0

Number of models with verbose output

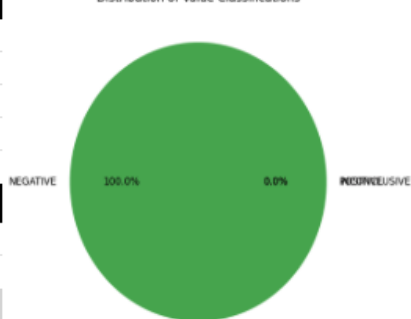
Count

1

Out of Domain model:

2

Distribution of Value Classifications



Distribution of predictions

Endocrine Disruption

Summary

Category	Count
POSITIVE	1
NEGATIVE	15
INCONCL	0

Number of models with verbose output

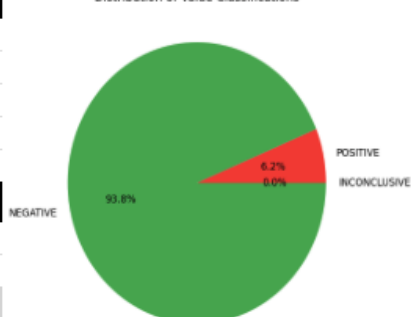
Count

1

Out of Domain model:

2

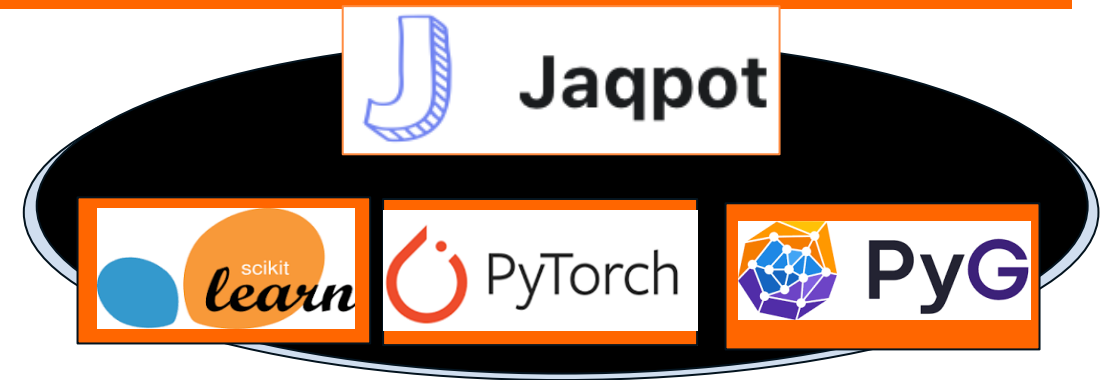
Distribution of Value Classifications



Distribution of predictions

The Jaqpotpy library for the development and integration of models

jaqpotpy is a Python library that helps you build models and use the Jaqpot platform. You can use it to create, train and deploy machine learning models based on **Scikit-learn**, **PyTorch** and **PyTorch Geometric**.



README MIT license

Build and test **passing** Publish to PyPI **passing**

Jaqpotpy

The jaqpotpy library enables you to upload and deploy machine learning models to the Jaqpot platform. Once uploaded, you can manage, document, and share your models via the Jaqpot user interface at <https://app.jaqpot.org>. You can also make predictions online or programmatically using the [Jaqpot API](#).

Getting Started

Prerequisites

- Python 3.10
- An account on <https://app.jaqpot.org>

Installation

Install jaqpotpy using pip:

```
pip install jaqpotpy
```

Model Training and Deployment

Source code here:
<https://github.com/ntua-unit-of-control-and-informatics/jaqpotpy>

Jaqpot docs

- Intro
- Getting started
- Jaqpotpy**
- ONNX Integration
- Scikit-learn models
- PyTorch models
- PyTorch Geometric models
- Full API Reference
- Examples
- Offline Models
- Jaqpot API
- Docker models

Jaqpotpy

jaqpotpy is a Python library that helps you build models and use the Jaqpot platform. You can use it to create, train and deploy machine learning models.

Key Features:

- **Build Models:** Create and train models with preprocessing and evaluation tools
- **Upload Models:** Upload your trained models to Jaqpot
- **Custom Tools:** Add your own preprocessing steps, featurization and metrics
- **Use Jaqpot:** Connect to Jaqpot API to deploy models and get predictions
- **Deploy:** Get your models ready for production use
- **Offline Predictions:** Download models and run predictions locally without internet

Jaqpotpy helps you take your models from your computer to the Jaqpot platform, where others can use them. With the new offline capabilities, you can also bring Jaqpot models back to your local environment for private, fast predictions.

ONNX Integration

In this section, we explain Jaqpot's use of ONNX to ensure co...

Scikit-learn models

6 items

PyTorch models

5 items

PyTorch Geometric models

4 items

Full API Reference

24 items

Examples

GitHub Examples

Offline Models

This is a new feature that allows you to download models from ...

Tutorials and documentation is available here: <https://jaqpot.org/docs/jaqpotpy/>

Automatic calculation of descriptors

Mordred Web UI

Descriptor calculator

Upload

[smi](#) or [sdf](#) file

☐ Generate 3D conformer ☒ Desalt

<https://mordred-descriptor.github.io/documentation/master/descriptors.html>

The Mordred library includes more than 1800 descriptors which are grouped into 50 categories.

Summary

#	NAME	
1819	WPol	113.0
1820	Zagreb1	320.0
1821	Zagreb2	399.0
1822	mZagreb1	19.46
1823	mZagreb2	11.37

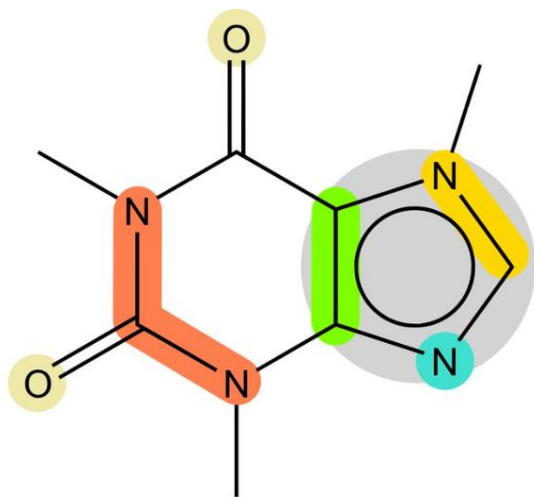
Moriwaki, H., Tian, YS., Kawashita, N. *et al.* Mordred: a molecular descriptor calculator. *J Cheminform* **10**, 4 (2018).

Molecular descriptor guide <https://www.epa.gov/sites/default/files/2015-05/documents/moleculardescriptorsguide-v102.pdf>

Automatic calculation of fingerprints

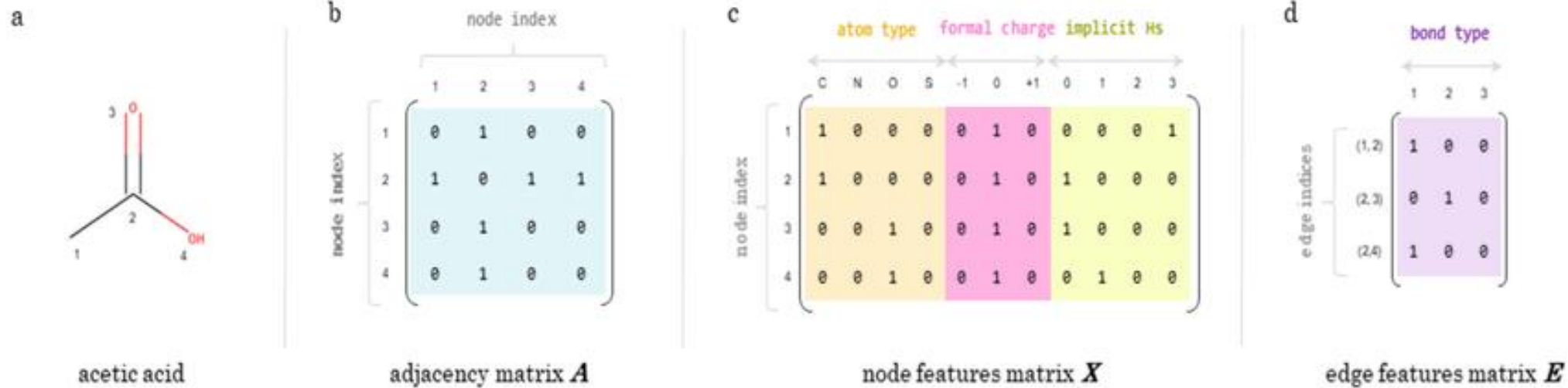
The MACCS key (Molecular ACCess System) is a binary fingerprint (a string of 166 positions where each position contains 0 or 1). Each bit position represents the presence (=1) or absence (=0) of a predetermined structural feature. The feature definitions for MACCS keys are available at the following address:

<https://github.com/rdkit/rdkit/blob/master/rdkit/Chem/MACCSkeys.py>



Key	Description
65	N in aromatic bonds with C
77	N separated by 2 bonds
83	heteroatoms in 5 ring
89	O separated by 4 bonds
99	C in C=C
162	aromatics

Graph Representations of chemical substances



The Jaqpotpy library for the development and integration of models

jaqpotpy assists users during models' lifecycle in feature preprocessing, model evaluation and DoA calculations.

```
import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestRegressor
from jaqpotpy import Jaqpot
from jaqpotpy.datasets import JaqpotTabularDataset
from jaqpotpy.models import SklearnModel

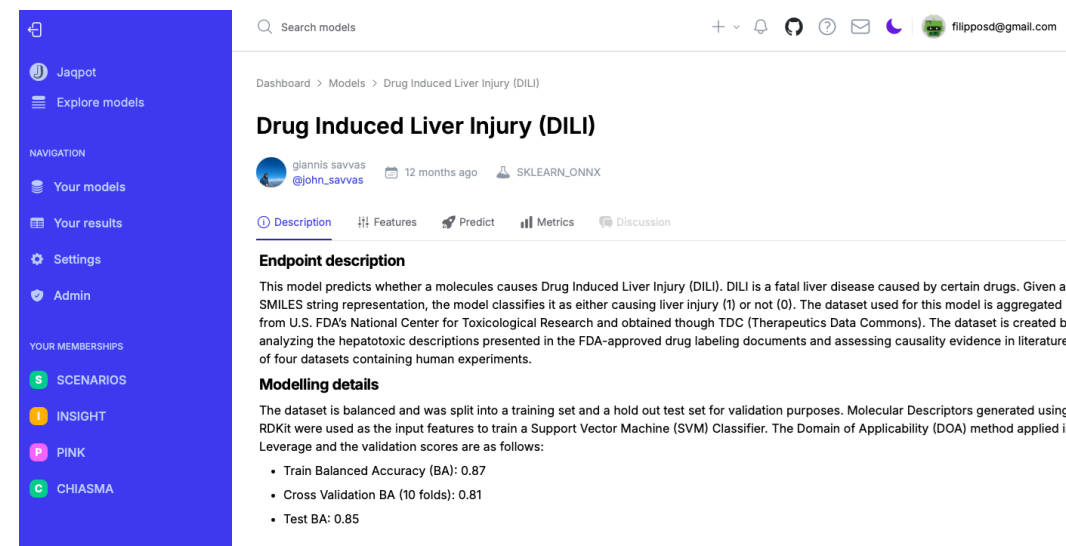
# Creating a Simulated Dataset for Model Training
np.random.seed(42)
X1 = np.random.rand(100)
X2 = np.random.rand(100)
ACTIVITY = 2 * X1 + 3 * X2 + np.random.randn(100) * 0.1
df = pd.DataFrame({"X1": X1, "X2": X2, "ACTIVITY": ACTIVITY})
y_cols = ["ACTIVITY"]
x_cols = ["X1", "X2"]

# Step 1: Create a Jaqpotpy dataset
dataset = JaqpotTabularDataset(df=df, y_cols=y_cols, x_cols=x_cols, task=ModelTask)

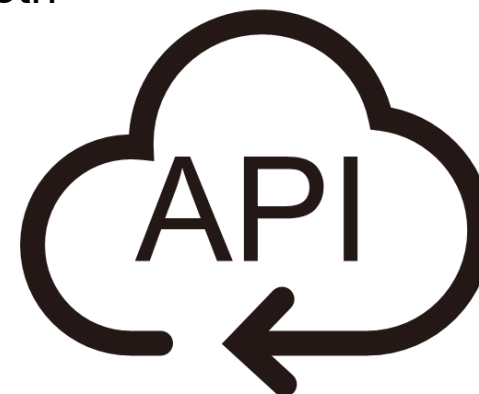
# Step 2: Build a model
rf = RandomForestRegressor(random_state=42)
myModel = SklearnModel(dataset=dataset, model=rf)
myModel.fit()

# Step 3: Upload the model on Jaqpot
jaqpot = Jaqpot()
jaqpot.login() # log in to Jaqpot
myModel.deploy_on_jaqpot(
    jaqpot=jaqpot,
    name="Demo: Regression",
    description="This is a description",
    visibility=ModelVisibility.PRIVATE
)
```

Deployment leads to a model accessible both
over its dedicated webpage and over API

The screenshot shows the Jaqpot web interface. On the left is a blue sidebar with navigation links: Jaqpot, Explore models, Your models, Your results, Settings, and Admin. Below these are 'YOUR MEMBERSHIPS' with icons for SCENARIOS, INSIGHT, PINK, and CHIASMA. The main content area shows the 'Drug Induced Liver Injury (DILI)' model page. It includes a description of the model's purpose, its endpoint description, and modelling details such as Train Balanced Accuracy (BA) of 0.87, Cross Validation BA of 0.81, and Test BA of 0.85. The page also has tabs for Description, Features, Predict, Metrics, and Discussion.

<https://app.jaqpot.org/dashboard/models/2019/description>



Generating and integrating new PBK models

The model creator can provide information that will accompany the model as metadata from its creation. Fields can be customised to the PINK annotation schema, for example the type of impact covered.

```
{
  "name": "IndusChemFate PBTK",
  "type": "R_PBPk",
  "libraries": [
    {
      "name": "deSolve",
      "version": "1.40",
      "id": 62462,
      "createdAt": "2025-05-13T11:18:48.819025Z",
      "updatedAt": "2025-05-13T11:18:48.819025Z"
    }
  ],
  "dependentFeatures": [
    {
      "key": "M_Adip_0",
      "name": "M_Adip_0",
      "featureType": "FLOAT",
      "id": 41782,
      "units": "µmol",
      "range": null,
      "description": "Amount of substance 0 in adipose",
      "featureDependency": "DEPENDENT",
      "visible": null,
      "possibleValues": null,

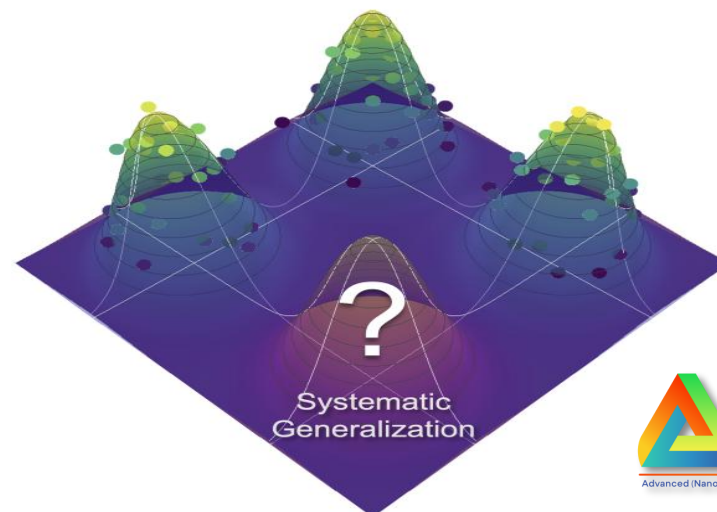
```

```
    "username": "vassilis_minadakis",
    "firstName": "Vassilis",
    "lastName": "Minadakis",
    "email": "vassilisminadakis@gmail.com",
    "emailVerified": true,
    "avatarUrl": "https://d2zoqz4gyxc03g.cloudfront.net/avatars/8ba56047-97db-4954-affb-b4f78bab0c05.jpg",
    "canEdit": null,
    "createdAt": null
  },
  "canEdit": false,
  "isAdmin": false,
  "selectedFeatures": null,
  "tags": "IndusChemFate, PBK",
  "legacyPredictionService": null,
  "scores": {
    "train": ☐,
    "test": ☐,
    "crossValidation": ☐
  },
  "rPbpkConfig": null,
  "dockerConfig": null,
  "createdAt": "2025-05-13T11:18:48.819025Z",
  "updatedAt": "2025-05-13T11:18:48.819025Z"
}
```

Generative modelling approach

- The goal is to generate molecules with **optimized properties**.
- Molecules should be **valid** and **diverse**
- **Generative Flow Network (GFlowNet)**¹ is used .
- This particular generative framework is chosen due to it's **explorative** nature to generate a diverse set of high-quality solutions with desired properties.

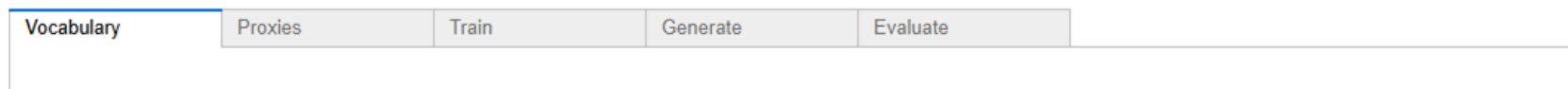
1. <https://arxiv.org/abs/2106.04399>



GenAI workflow

3. Train a GFlowNet (molecule generator using fragments, reward guided by proxy models):

- Open [gflownet_trainer.ipynb](#) notebook inside the repo and Run All Cells.
- Navigate through different tabs:



(1) Vocabulary:

Build or load the fragment set that defines the 'building blocks' for your molecules.

Fragment vocabulary

Use in Train/Generate: this selection defines the fragment set used by the environment.

Use in run: JSON path:

[Fragment builder](#)

Active fragment viewer

[View active fragments](#)

Viewer height:

Active fragments: 72

```
Br
C
C#N
C1=CCCCC1
C1=CNC=CC1
C1CC1
C1CCCC1
C1CCCCC1
```

(2) Proxies:

Load your pre-trained proxy models and configure objectives for each property (minimization, maximization, targeted range, scaling bounds).

Proxy config: ?

Proxy weights: ?

[Load proxies → build fields](#) ?

Per-proxy settings (auto-generated)

Proxy 1: $\lambda_{\max}$

Goal: Weight:

mode: ?

Range constraint (absolute units)

low: high:

margin: ?

CSV: ?

Proxy 2: f

Goal: Weight:

Scaling bounds (absolute units)

min: max:

CSV: ?

Example for Multi-Objective Property Optimization:

- Target specific range for maximum absorbance wavelength ($\lambda_{\max}$): target = 300-360 nm
- Maximize absorbance factor (f)

(3) Train:

Adjust hyperparameters and train the model.

The model is automatically saved in the indicated folder.

The GFlowNet learns to sample molecules proportional to the rewards defined in the Proxies tab.

Training Hyperparameters

3. GFlowNet Training

Adjust hyperparameters and train the model. The GFlowNet learns to sample molecules proportional to the rewards defined in the Proxies tab.

Random seed: ?

Learning rate: ?

Temperature: ?

Base log dir:

Training steps: ?

Batch (samples/step): ?

Max fragments: ?

Embedding size: ?

Transformer layers: ?

Attention heads: ?

MPL layers: ?

[Train GFlowNet](#) Progress: Elapsed: 00:05

>PINK

GenAI workflow

(4) Generate:

- Use the model you just trained or load a pre-trained model in Run log_dir.
- Sample molecules with Generate+Score.
- Download designs and visualize them.

Molecule cards collect all info
about a specific design
(including AD data)

Example for Multi-Objective Property Optimization:

- Target specific range for maximum absorbance wavelength (λ_{max}): target = 300-360 nm
- Maximize absorbance factor (f)

Gather statistics

Top-100 UNIQUE summary (Total reward (weighted))

Summary (All selected)

	count	mean	std	min	max
Total reward (weighted)	100.0	0.259995	0.016565	0.233797	0.301421
λ_{max} prediction	100.0	331.718048	10.569038	309.345154	357.026520
λ_{max} reward	100.0	1.000000	0.000000	1.000000	1.000000
f prediction	100.0	1.920690	0.122370	1.727154	2.226714
f reward	100.0	0.519991	0.033129	0.467595	0.602841

Summary (Inside AD for all proxies) — 100/100

	count	mean	std	min	max
Total reward (weighted)	100.0	0.259995	0.016565	0.233797	0.301421
λ_{max} prediction	100.0	331.718048	10.569038	309.345154	357.026520
λ_{max} reward	100.0	1.000000	0.000000	1.000000	1.000000
f prediction	100.0	1.920690	0.122370	1.727154	2.226714
f reward	100.0	0.519991	0.033129	0.467595	0.602841

Summary (Outside AD for ≥ 1 proxy) — 0/100

No rows.

4. Inference & Generation

Load a trained model to sample and score new molecules.

Instructions:

1. Ensure your run folder contains config.yaml and model_state.pt.
2. **Important:** Proxy objectives in the 'Proxies' tab must match the training configuration.
3. Use the **Sync Proxies to Model** button to automatically align settings with the selected run.

Run log_dir: logs/gflownet_ui/run_001_1772462336

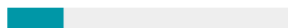
Mols/round: 50

Rounds: 5

 Sync Proxies to Model

 Generate + Score

 Download CSV

Progress: 

Round: 2 / 5

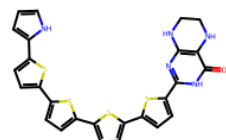
View: Top-K by metric

Top-K: 100

Metric: Total reward (weighted)

Sort: Maximize

Top 4 Designs (Sorted by Total Reward)

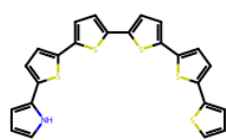


SMI: O=C1[nH]C(-c2ccc(-c3ccc(-c4ccc...

Total R: 0.301

λ_{max} : 344.04 AD: Inside

f: 2.23 AD: Inside

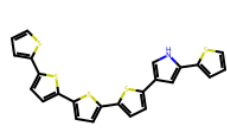


SMI: C1C[nH]C(-c2ccc(-c3ccc(-c4ccc...

Total R: 0.301

λ_{max} : 354.88 AD: Inside

f: 2.22 AD: Inside

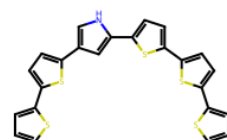


SMI: C1CSC(-c2ccc(-c3ccc(-c4ccc(-c5...

Total R: 0.289

λ_{max} : 355.06 AD: Inside

f: 2.14 AD: Inside



SMI: C1CSC(-c2ccc(-c3[nH]C(-c4ccc...

Total R: 0.289

λ_{max} : 354.29 AD: Inside

f: 2.13 AD: Inside

Visualize top
candidates

Illustrative example: Generating molecules with high logP values

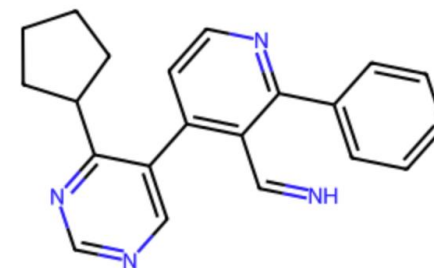
Model-Framework is available on jaqpot: <https://app.jaqpot.org/dashboard/models/2032/description>

QSAR ToolBox LogP predictions

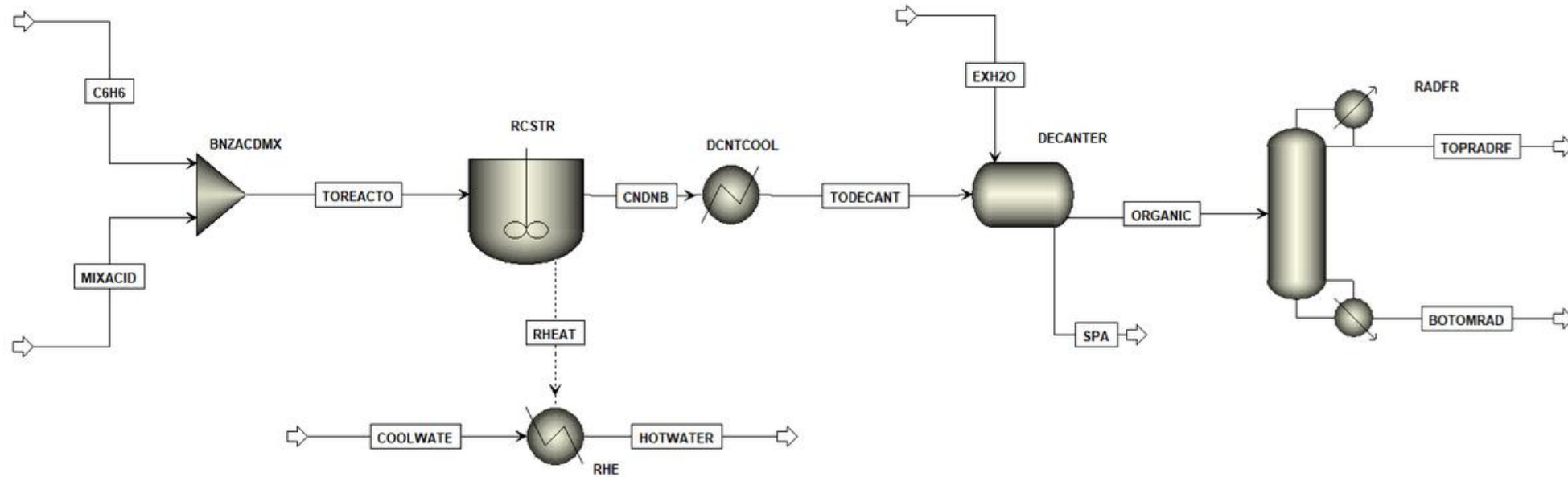
Proxy GNN model used in GFlowNet to predict LogP

Example of a generated molecule

	Prediction	SMILES
9.6566	8.277	<chem>Oc1cc(C2=CC(C3C=CCCC3)CCC2)cc(-c2ccc3ccccc3c2)c1</chem>
10.7758	9.521	<chem>CC(C)(C)C1C(C2=CCCC2)CCCC1c1cccc(-c2cccs2)c1</chem>
8.7837	8.299	<chem>c1ccc(-c2ccnc(-c3ccc(C4CCC(c5ccccc5)CC4)cc3)c2)cc1</chem>
7.3031	6.107	<chem>c1cc(C2CC2)c(-c2ccc3ccccc3c2)c(C2COCC2C2CCCO2)c1</chem>
11.1945	9.373	<chem>C1CCC(C2CC(C3CCC(C4CCCC4)CC3)CC(C3CCOC3)C2)CC1</chem>
7.7512	6.581	<chem>CCc1ccc(C2OCCC2C2CCC(c3ccccc3)CC2)s1</chem>
6.8858	5.097	<chem>c1n[nH]c(C2CCCCC2)c1C1CCCCC1C1CCCCC1N1CCNCC1</chem>
4.9659	2.868	<chem>N=Cc1c(-c2cncnc2C2CCCC2)ccnc1-c1ccccc1</chem>



MOO Example – nitration of aromatic (benzene)



Input variables:

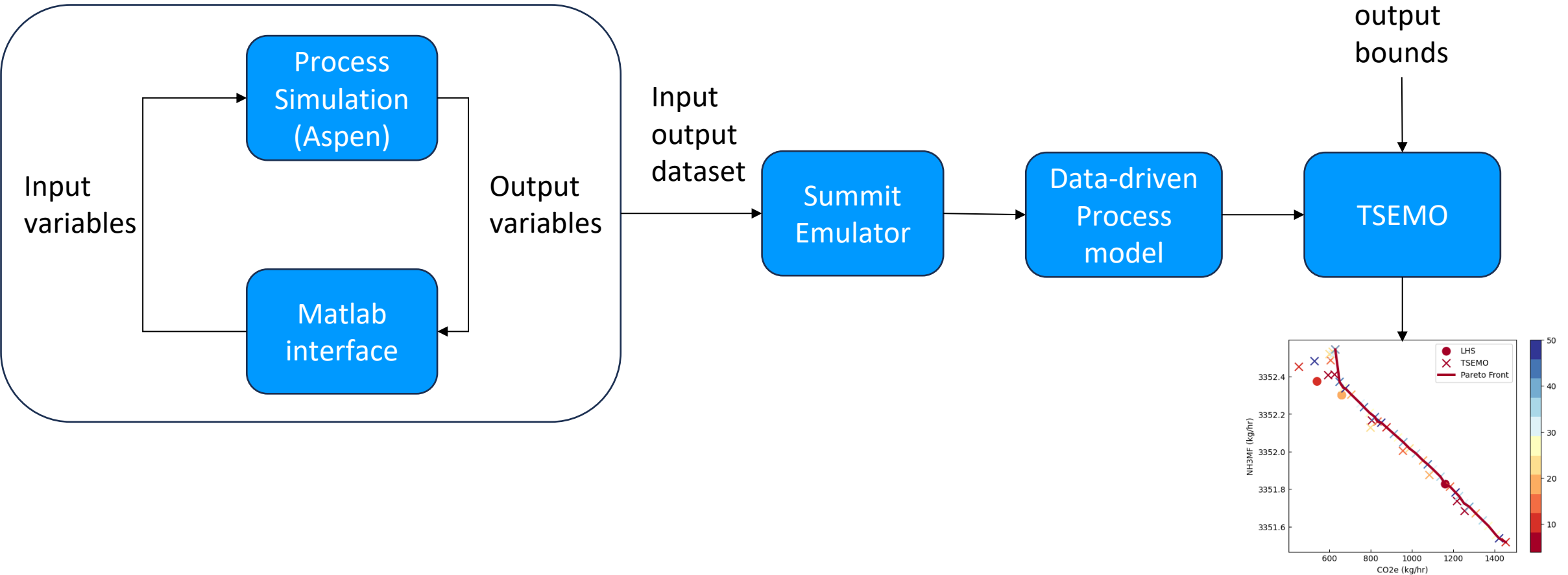
Variables	(units)	Limits
Reactor Temperature (C)		[65.0,95.0]
Residence time (hr)		[6.0,13.0]
Benzene to acids ratio (w/w)		[0.0,1.0]

4 Objectives:

1. NB_Mol_Percentage_Yield (% mol NB/MOI C6H6) $\Rightarrow$ Process efficiency (production capacity)
2. DNB_Mol_Yield (mol DNB/hr) $\Rightarrow$ Safety related (toxic byprod.)
3. CO2e per kg NitroC6h6 (kg CO2/hr) $\Rightarrow$ LCA related (emissions)
4. TOTALCOST (\$) $\Rightarrow$ LCC related (production costs)

Production process MOO - Workflow 3

Sensitivity analysis



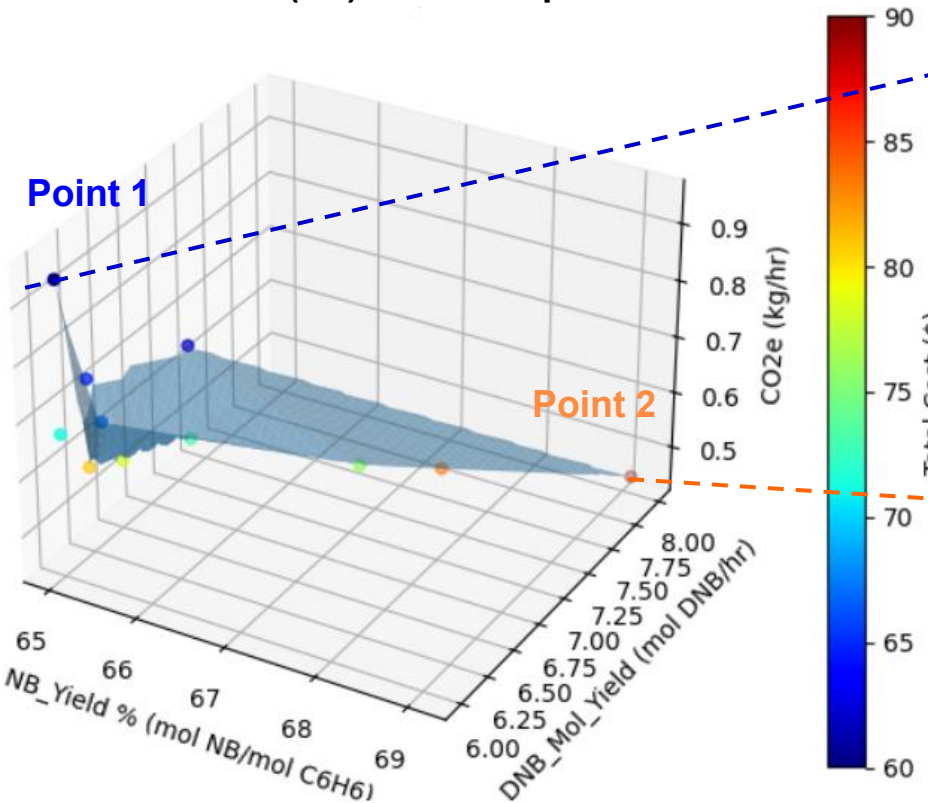
Pareto front

MOO Example – nitration of aromatic (benzene)

Reading the results

Values of the input parameters
are decided by the designer

Pareto surface (2D) + heat map for Total cost



R_T	ResTime	Benz2AcidRatio	NByield	DNByield	CO2e	TOTALCOST
DATA	DATA	DATA	DATA	DATA	DATA	DATA
65.0	6.0	0.940553	60.0	0.0	11.0	13811.0
65.724664	6.563721	0.770414	61.987709	0.000173	11.413369	14276.293945
65.030944	6.206928	0.738339	64.898796	0.000194	12.32363	14536.286133
65.480091	6.001707	0.717673	66.898422	0.000226	12.935511	14658.143555
65.0	6.0	0.673955	71.166206	0.000314	14.402148	15174.744141
65.988574	6.371139	0.650601	73.401672	0.000393	15.111883	15502.807617
67.528135	6.660242	0.627405	75.91481	0.000495	15.888021	15788.65918
65.393521	6.229343	0.607747	78.379135	0.000528	16.9161	16062.123047
65.054719	6.189286	0.586421	80.922569	0.000603	17.849514	16366.34668
67.931696	7.095091	0.566099	83.320786	0.000746	18.505066	16815.646484
69.108916	7.980441	0.455444	90.0	0.001396	21.0	18933.447266

Example design options:

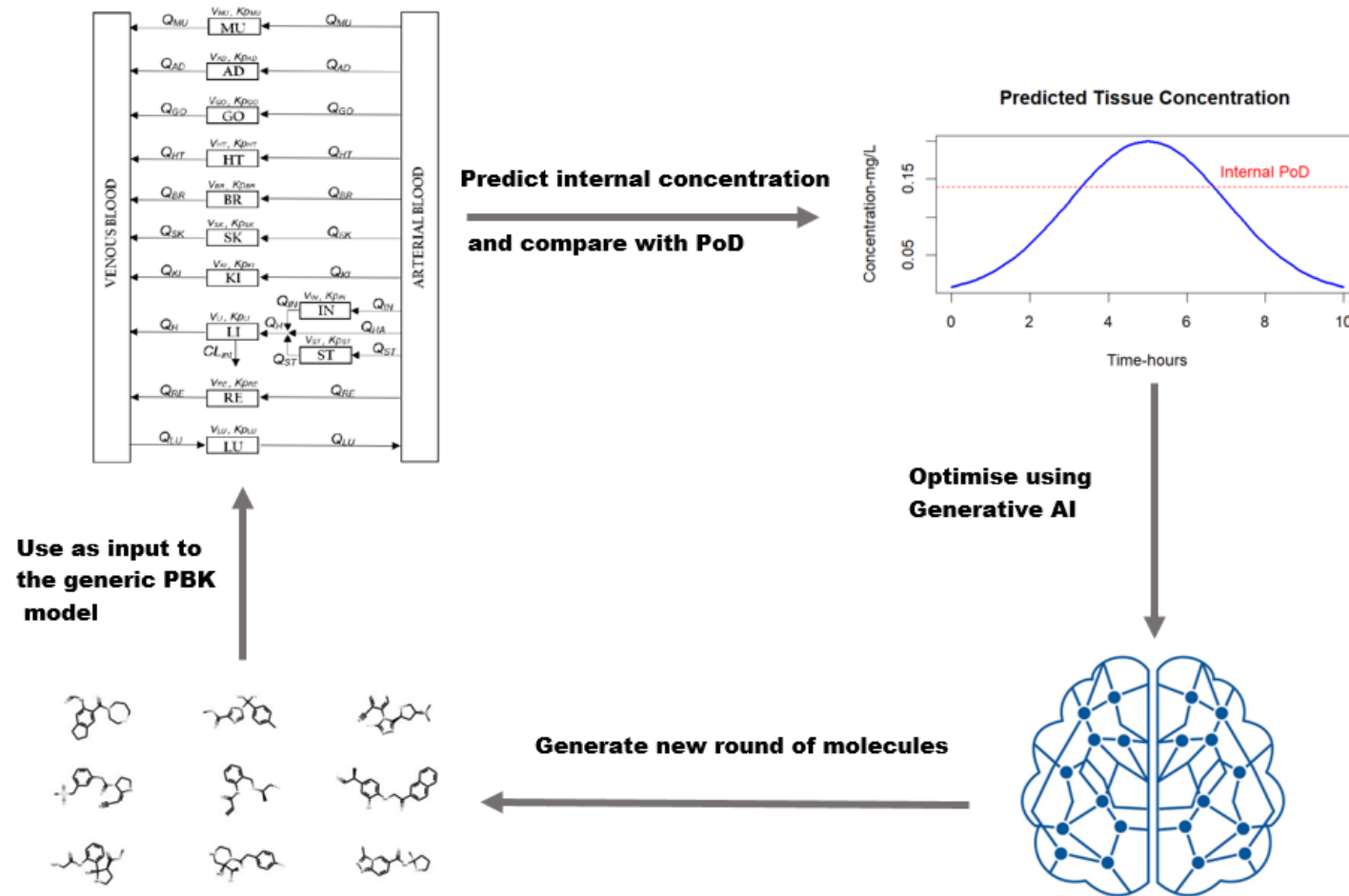
Point 1: minimum total cost, CO₂e, DNB **but** also minimum NB

Point 2: maximum total cost, CO₂e, NB **but** also maximum DNB

>PINK

Integrating biokinetics in the PINK DSS

- Develop and integrate PBK models for small molecules with an embedded QSAR for predicting partition coefficients from chemical structures
- Utilise the PBK model **for forward dosimetry and estimate the internal burden at the target organ** (linked to the KE) following a toxicologically-relevant exposure
- Compute risk by comparing internal concentrations to **internal Points of Departure (PoD)** connected to specific Key Events (KE) in AOPs
- Include this analysis in an iterative learning process, penalise molecules that accumulate in organs and the resulting concentration exceeds the defined internal PoD



IndusChemFate PBTK model - A Generic PBTK model

- Originally developed in VisualBasic (Available through Excel Application)
- Structured by 11 compartments
- Exposure paths: Inhalation and Skin
- Elimination paths: Urinary, Exhalation, Metabolism
- Metabolism for up to 4 metabolites can be simulated
- Parameterization through QSARs

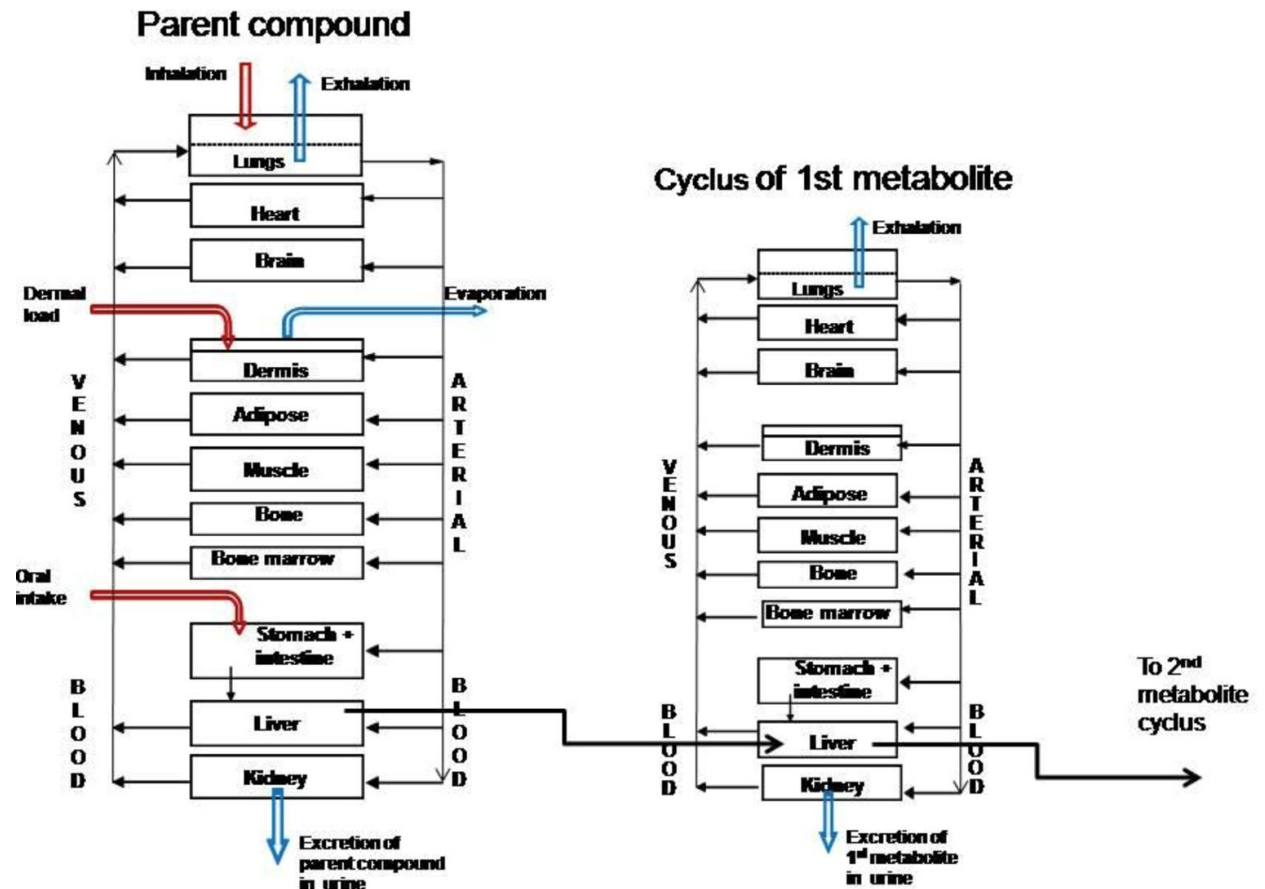
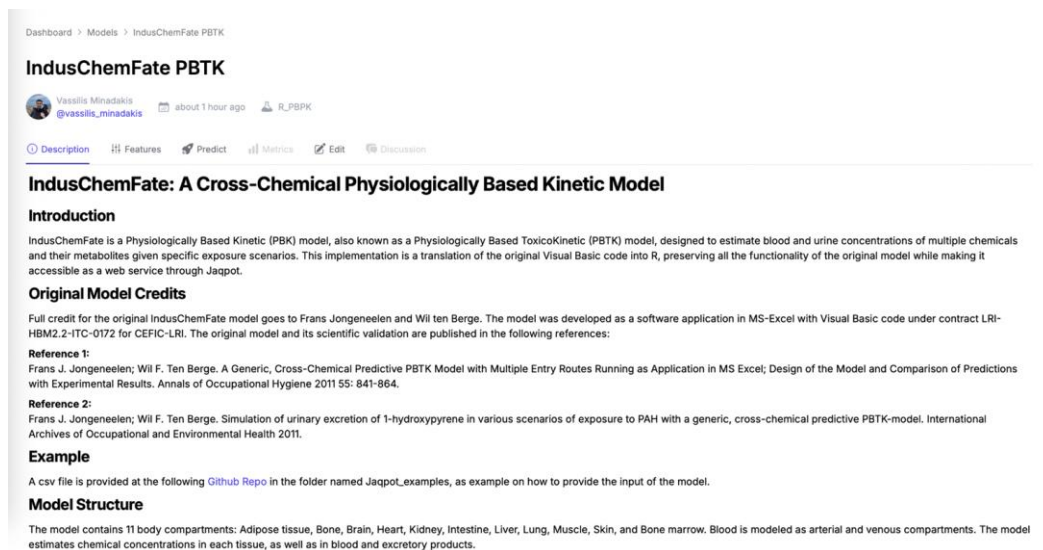


Figure PBTK_1: Structural representation of IndusChemFate (sourced by Jongeneelen and ten Berge (2011)).

Integration of IndusChemate on Jaqpot

Full Version of IndusChemFate Model

- Simulate up to 4 metabolites
- Applies metabolism processes to any compartment
- Large number of input parameters required as input
- Available User Interface and API



Dashboard > Models > IndusChemFate PBTK

IndusChemFate PBTK

Vassilis Minadakis @vassilis_minadakis about 1 hour ago R_PBPK

Description Features Predict Metrics Edit Discussion

IndusChemFate: A Cross-Chemical Physiologically Based Kinetic Model

Introduction

IndusChemFate is a Physiologically Based Kinetic (PBK) model, also known as a Physiologically Based Toxicokinetic (PBTK) model, designed to estimate blood and urine concentrations of multiple chemicals and their metabolites given specific exposure scenarios. This implementation is a translation of the original Visual Basic code into R, preserving all the functionality of the original model while making it accessible as a web service through Jaqpot.

Original Model Credits

Full credit for the original IndusChemFate model goes to Frans Jongeneelen and Wil ten Berge. The model was developed as a software application in MS-Excel with Visual Basic code under contract LRI-HBM2.2-ITC-0172 for CEFIC-LRI. The original model and its scientific validation are published in the following references:

Reference 1:
Frans J. Jongeneelen; Wil F. Ten Berge. A Generic, Cross-Chemical Predictive PBTK Model with Multiple Entry Routes Running as Application in MS Excel; Design of the Model and Comparison of Predictions with Experimental Results. *Annals of Occupational Hygiene* 2011 55: 841-864.

Reference 2:
Frans J. Jongeneelen; Wil F. Ten Berge. Simulation of urinary excretion of 1-hydroxypyrene in various scenarios of exposure to PAH with a generic, cross-chemical predictive PBTK-model. *International Archives of Occupational and Environmental Health* 2011.

Example

A csv file is provided at the following [Github Repo](#) in the folder named Jaqpot_examples, as example on how to provide the input of the model.

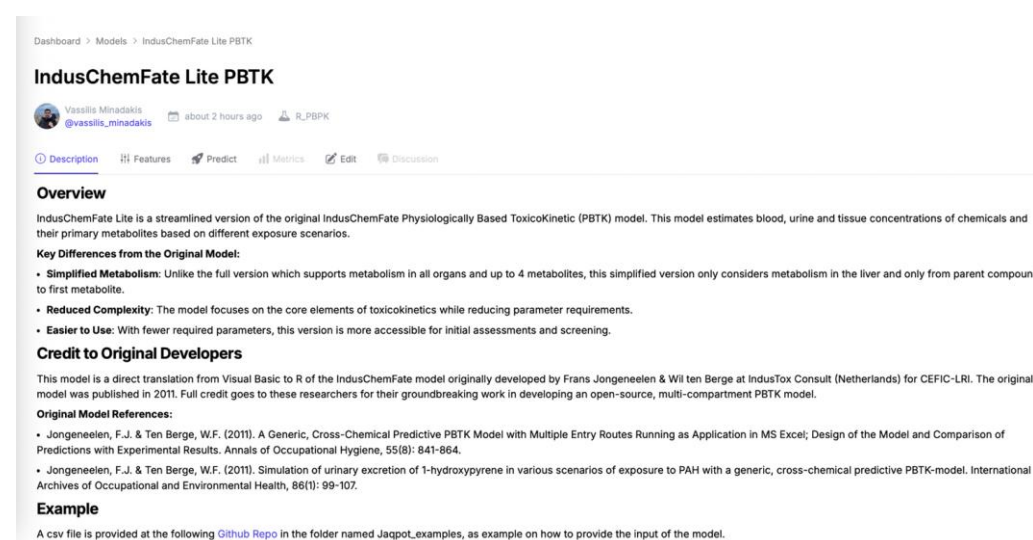
Model Structure

The model contains 11 body compartments: Adipose tissue, Bone, Brain, Heart, Kidney, Intestine, Liver, Lung, Muscle, Skin, and Bone marrow. Blood is modeled as arterial and venous compartments. The model estimates chemical concentrations in each tissue, as well as in blood and excretory products.

<https://app.jaqpot.org/dashboard/models/2124/description>

Lite Version of IndusChemFate Model

- Simulate up to 1 metabolite
- Applies metabolism processes only in Liver compartment
- Much less input parameters are required (more user-friendly).
- Available User Interface and API



Dashboard > Models > IndusChemFate Lite PBTK

IndusChemFate Lite PBTK

Vassilis Minadakis @vassilis_minadakis about 2 hours ago R_PBPK

Description Features Predict Metrics Edit Discussion

Overview

IndusChemFate Lite is a streamlined version of the original IndusChemFate Physiologically Based Toxicokinetic (PBTK) model. This model estimates blood, urine and tissue concentrations of chemicals and their primary metabolites based on different exposure scenarios.

Key Differences from the Original Model:

- **Simplified Metabolism:** Unlike the full version which supports metabolism in all organs and up to 4 metabolites, this simplified version only considers metabolism in the liver and only from parent compound to first metabolite.
- **Reduced Complexity:** The model focuses on the core elements of toxicokinetics while reducing parameter requirements.
- **Easier to Use:** With fewer required parameters, this version is more accessible for initial assessments and screening.

Credit to Original Developers

This model is a direct translation from Visual Basic to R of the IndusChemFate model originally developed by Frans Jongeneelen & Wil ten Berge at IndusTox Consult (Netherlands) for CEFIC-LRI. The original model was published in 2011. Full credit goes to these researchers for their groundbreaking work in developing an open-source, multi-compartment PBTK model.

Original Model References:

- Jongeneelen, F.J. & Ten Berge, W.F. (2011). A Generic, Cross-Chemical Predictive PBTK Model with Multiple Entry Routes Running as Application in MS Excel; Design of the Model and Comparison of Predictions with Experimental Results. *Annals of Occupational Hygiene*, 55(8): 841-864.
- Jongeneelen, F.J. & Ten Berge, W.F. (2011). Simulation of urinary excretion of 1-hydroxypyrene in various scenarios of exposure to PAH with a generic, cross-chemical predictive PBTK-model. *International Archives of Occupational and Environmental Health*, 86(1): 99-107.

Example

A csv file is provided at the following [Github Repo](#) in the folder named Jaqpot_examples, as example on how to provide the input of the model.

<https://app.jaqpot.org/dashboard/models/2123/description>

Integration of IndusChemate on Jaqpot

Fully documented: Descriptions and Units are provided for all parameters of the model through GUI.

IndusChemFate Lite PBTK



Vassilis Minadakis

@vassilis_minadakis

 about 2 hours ago

 R_PBPk

 Description

 Features

 Predict

 Metrics

 Edit

 Discussion

Independent Features

Name	Units	Range	Description	Type	Actions
dens_0	mg/cm ³ or g/L		Density of substance 0	FLOAT	
mw_0	g/mol		Molecular Weight of substance 0	FLOAT	
vpPa_0	Pa		Vapor Pressure of substance 0	FLOAT	
ikow_0			Log(Kow) of substance 0 in blood (pH=7.4)	FLOAT	
DermLKow			Log(Kow) of substance 0 in skin (pH=5.5)	FLOAT	
wsol_0	mg/L		Water solubility of substance 0	FLOAT	

Dependent Features

Name	Units	Description	Type	Actions
M_Adip_0	μmol	Amount of substance 0 in adipose	FLOAT	
M_Adip_1	μmol	Amount of substance 1 in adipose	FLOAT	
M_Bone_0	μmol	Amount of substance 0 in bone	FLOAT	
M_Bone_1	μmol	Amount of substance 1 in bone	FLOAT	
M_Brain_0	μmol	Amount of substance 0 in brain	FLOAT	
M_Brain_1	μmol	Amount of substance 1 in brain	FLOAT	

The R code of both models is provided in Github repo, along with CSV files to run demo simulations.

IndusChemFate_PBTK_R_Implementation Public

Edit Pins Unwatch 5 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file Add file Code

vassilismin Octocrylene_example_notebook eaba858 · 5 days ago 13 Commits

- Jaqpote_examples Octocrylene_example_notebook 5 days ago
- Jaqpote_service.R fix bug with initializing odes last week
- Lite_Jaqpote_service.R fix bug with initializing odes last week
- README.md Update README.md 2 months ago

README

IndusChemFate PBTK Model - R Implementation

This repository contains two R implementations of the IndusChemFate Physiologically Based Toxicokinetic (PBTK) model, which was originally developed in Visual Basic for MS Excel by Frans Jongeneelen and Wil ten Berge.

Overview

IndusChemFate is a multi-chemical PBTK model designed to estimate blood and urine concentrations of chemicals and their metabolites under various exposure scenarios. The model is capable of simulating:

About: R implementation of the IndusChemFate PBTK model for chemical fate and exposure assessment.

- Readme
- Activity
- Custom properties
- 0 stars
- 5 watching
- 0 forks

Report repository

Releases: No releases published. [Create a new release](#)

Packages: No packages published. [Publish your first package](#)

Languages

Available at: https://github.com/ntua-unit-of-control-and-informatics/IndusChemFate_PBTK_R_Implementation/tree/main

Octocrylene Example

Predictions using Jaqpot GUI

IndusChemFate Lite PBTK

Yasir M. Minadakis @yasir_minadakis 34 minutes ago

Description Features Predict Metrics Edit Discussion

Choose Your Prediction Input Method

Fit out the form or Upload a CSV file (max 100 rows)

dens_0	mw_0	vpPa_0	Row_0
1051	361.5	0	6.8815

DermKow	wsol_0	Resorpt_0	FrachpRt_0
6.8815	0.003808	-100	0

dens_1	mw_1	vpPa_1	Row_1
0	0	0	0

wsol_1	Resorpt_1	FrachpRt_1	scenario
0	-100	0	1

body_weight	Temp	SkinArea	RespProt
70	25	20000	1

DermProt	DcrBctusRt	HLsub	Cmp
1	0	1	0

DeposRate	DeposRate_times	DeposRate	DeposRate_times
0.00036	0	0	0

Exp.times	DeposRate	DeposRate_times	OralDose
0	0	3	0

OralDose.times	Vmax_Liver_0	Vmax_p_Liver_0	Km_Liver_0
0	0	0	1

sim.start	sim.end	sim.step
0	240	1

Submit

Result Print

ID 18813 Success

17 minutes ago less than 10 seconds

Export CSV

MassSurf	StratCorn	UnInVol	Sum_Bolus	C_Adip_0	C_Adip_1	C_Bone_0	C_Bone_1	C_Brain_0	C_Brain_1	C_Heart_0	C_Heart_1	C_Kidney_0	C_Kidn
0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0	0.000e+0
-3.434e-16	1.904e-04	8.242e-02	0.000e+0	3.486e-02	0.000e+0	4.501e-02	0.000e+0	2.882e-01	0.000e+0	1.757e-01	0.000e+0	3.024e-01	0.000e+0
-3.289e-15	2.353e-04	1.248e-01	0.000e+0	2.042e-01	0.000e+0	2.173e-01	0.000e+0	1.234e-01	0.000e+0	4.758e-01	0.000e+0	8.180e-01	0.000e+0

Predictions using Python SDK of Jaqpot

```
IndusChemFate_Octocrylene_Simulation.ipynb
File Edit View Insert Runtime Tools Help

[39] # Install required modules
      pip install jaqpot-python-sdk
      pip install jaqpot-api-client
      pip install python-dotenv
      import pandas as pd

[40] # Login to Jaqpot with API keys
      from dotenv import load_dotenv
      load_dotenv("keys.env") # Returns True if successful

[41] # Initialize a JaqpotApiClient
      from jaqpot_python_sdk.jaqpot_api_client import JaqpotApiClient
      jaqpot_api_client = JaqpotApiClient()

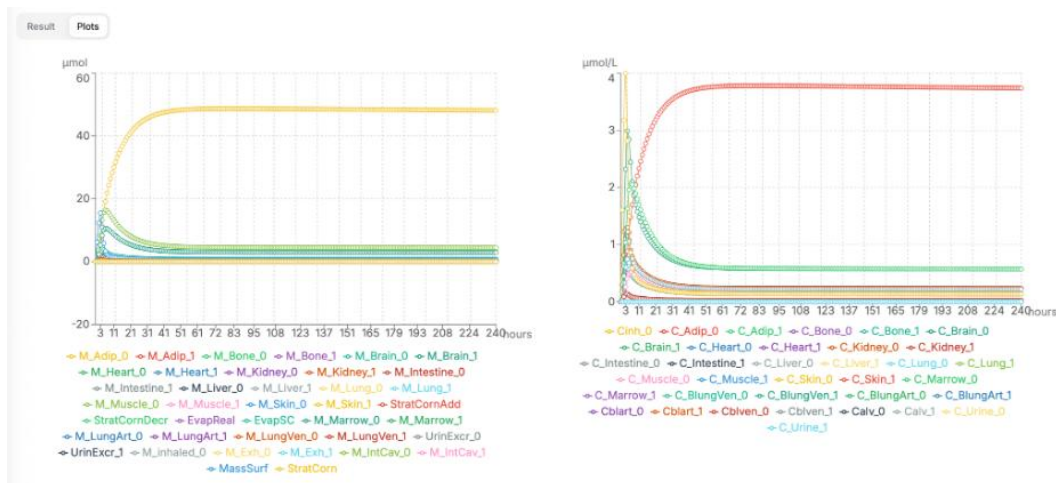
[42] input_data_dict = {
      'dens_0': 1051, 'mw_0': 361.5, 'vpPa_0': 0, 'lkow_0': 6.8815, 'DermKow': 6.8815, 'wsol_0': 0.003808, 'Resorpt_0': -100, 'FrachpRt_0': 0, 'dens_1'
    }

[43] # Make predictions using IndusChemFate Lite on Jaqpot
      prediction = jaqpot_api_client.predict_sync(model_id=2115, dataset=input_data_dict)
      print(prediction)

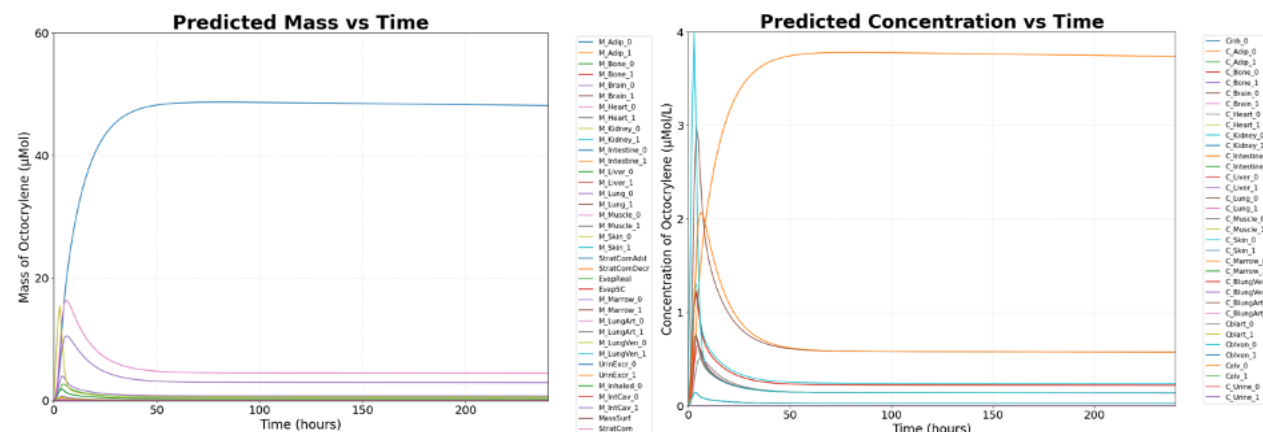
[{'M_Adip_0': '0.000e+00', 'M_Adip_1': '0.000e+00', 'M_Bone_0': '0.000e+00', 'M_Bone_1': '0.000e+00', 'M_Brain_0': '0.000e+00', 'M_Brain_1': '0.000e+00'}
```

Biokinetics simulations, Octocrylene Example

Automatically created plots by Jaqpot GUI.



Plots created with python code through Google Colab notebook.



The PINK team



1 Harry Sarimveis

National Technical University of Athens
(NTUA)

>PINK

>_THANK YOU!

WWW.PINK-PROJECT.EU



THE PINK PROJECT HAS RECEIVED FUNDING FROM THE EUROPEAN UNION'S HORIZON EUROPE RESEARCH AND INNOVATION PROGRAMME UNDER GRANT AGREEMENT NO. 101137809.

ASSOCIATED PARTNERS (I.E. (A) SWISS PARTNERS AND (B) UK PARTNERS) HAVE RECEIVED NATIONAL FUNDING FROM (A) THE SWISS STATE SECRETARIAT FOR EDUCATION, RESEARCH AND INNOVATION (SERI), AND (B) INNOVATE UK.