



The EBRAINS Knowledge Graph

Introductory Tutorial

Laura Morel, Alix Bonard, Peyman Najafi

NeuroPSI Paris-Saclay Institute of Neuroscience, CNRS, Université Paris-Saclay, France.

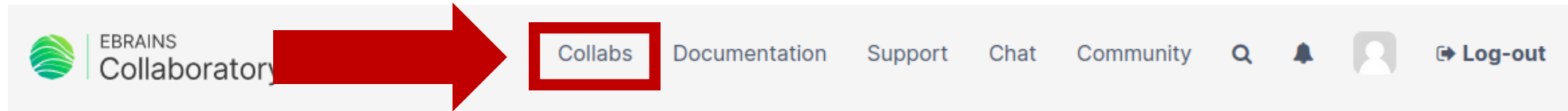
Introduction

Setup: creating a collab in the Collaboratory

- Available here: <https://wiki.ebrains.eu>
- The EBRAINS Collaboratory allows users to coordinate and work together in an online environment to publish work, document methodology, share data
- Collabs are workspaces in the Collaboratory

Setup: creating a collab in the Collaboratory

- <https://wiki.ebrains.eu>



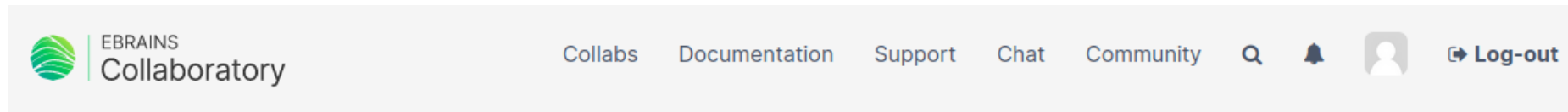
Collaborate.
Create reproducible science.
Discover EBRAINS services at
work.
From anywhere.

Getting started



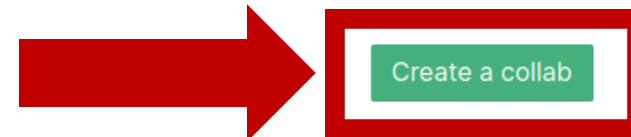
Setup: creating a collab in the Collaboratory

- <https://wiki.ebrains.eu>



Collab Search

Find available collabs searching for words in title and description.
You can refine the results using the filters.



Search word in collab titles & descriptions. You can use AND, OR and - operators.

Search

Setup: creating a collab in the Collaboratory

- <https://wiki.ebrains.eu>

Create a Collab

This page is used to create a new collab in the Collaboratory Wiki. Please plan your collab names in advance to avoid having many collabs with few pages and files. Thank you.

Collab Title (required)

The perfect title concisely states the goals of this collaboration workspace.

EBRAINS users day 2025 - myusername

✓ Min. 5 characters.

Collab Name (required)

Also known as Collab Id. This name (also called slug) will appear in the URL to your collab. It has to be unique. The field is autofilled based on the collab title. You can edit it now using the text box. Once the collab has been created, the title can still be edited but not the name.

ebrains-users-day-2025-myusername

✓ 33 of 40 characters. Min. 5 characters. No spaces.

Description

Describe what are the objectives of this collaboration workspace and the general context. This field is searched by the collab search tool. Use keywords that will help find this collab.

This is an example of a collab description

✓ 42 of 500 characters.

Visibility

Private



Private collabs are only accessible to people added to the collab's Team. The collab contents are not viewable by other users.

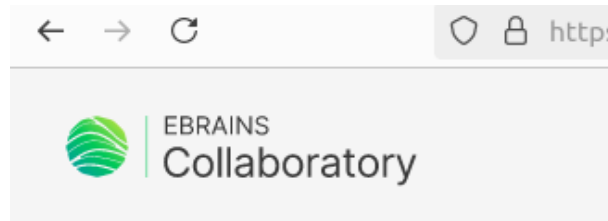
Public



Public collabs are accessible to everybody, even without an EBRAINS account. However, only members of the collab's Team with edit rights can modify anything in the collab.

Setup: creating a collab in the Collaboratory

- <https://wiki.ebrains.eu>



Private Member

EBRAINS users day 2025 - myusername

Bucket

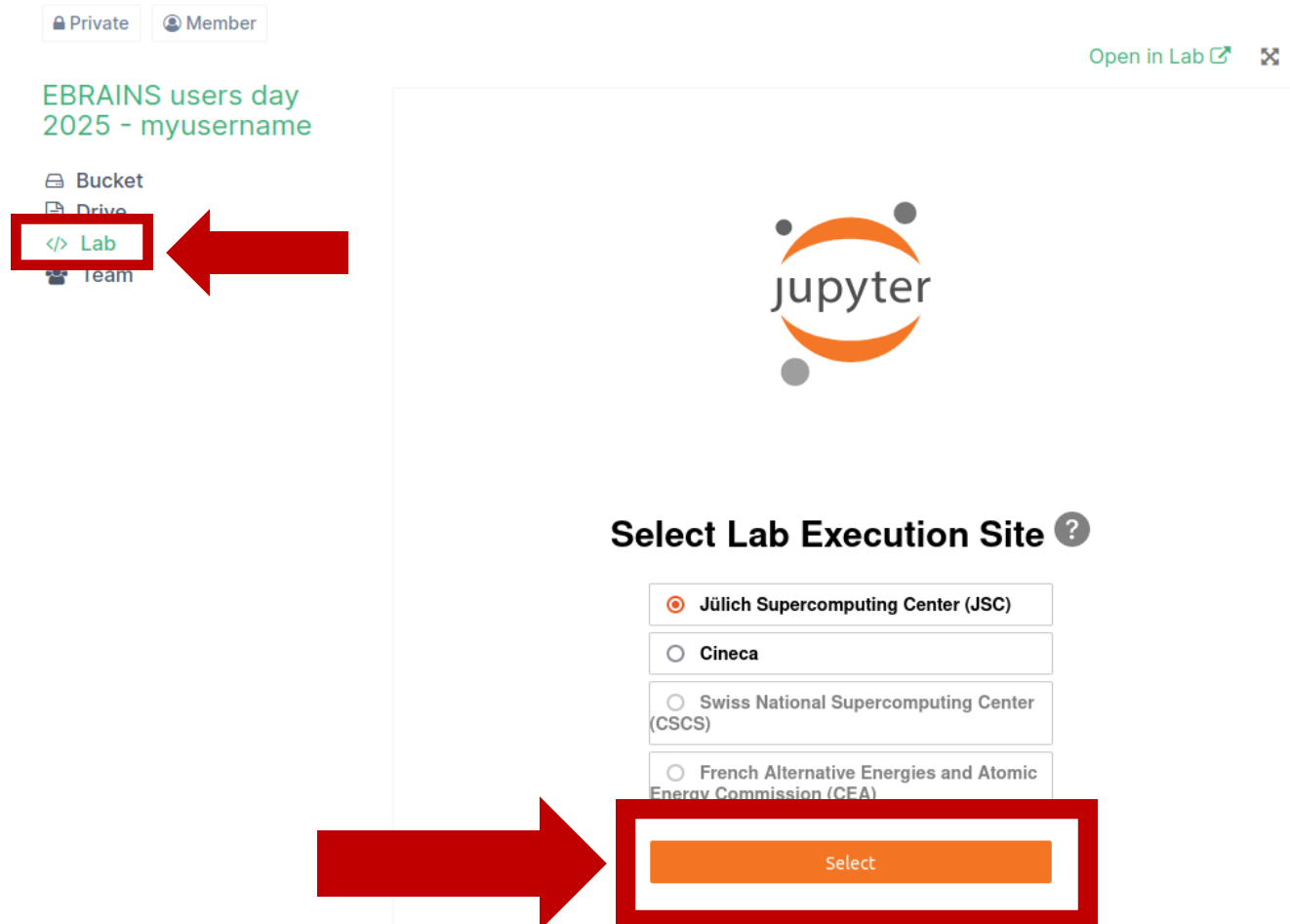
Drive

Lab

Team

- **Bucket:** large storage space but cannot be edited, ideal for data
- **Drive:** files and documents that may need to be edited or moved
- **Lab:** cloud-based development environment for developing and running programs without needing to install anything
- **Team:** manage team members and their permissions
- Check out [the documentation](#) for more information

Use the Lab in a collab



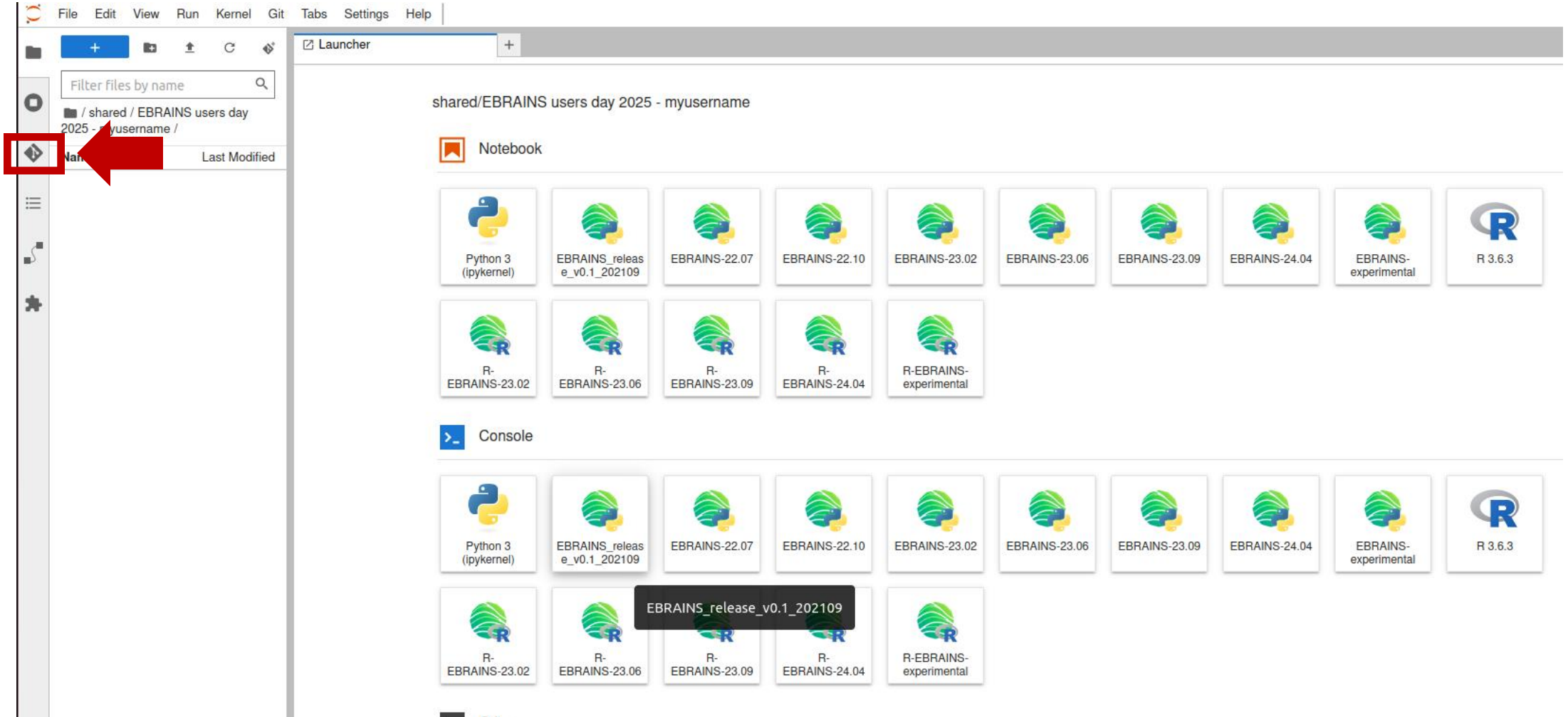
Use the Lab in a collab

The screenshot shows the EBRAINS Lab interface. On the left, a sidebar contains a file explorer and a search bar. The main area displays a grid of notebooks and consoles. Three blue arrows with white text point to specific elements:

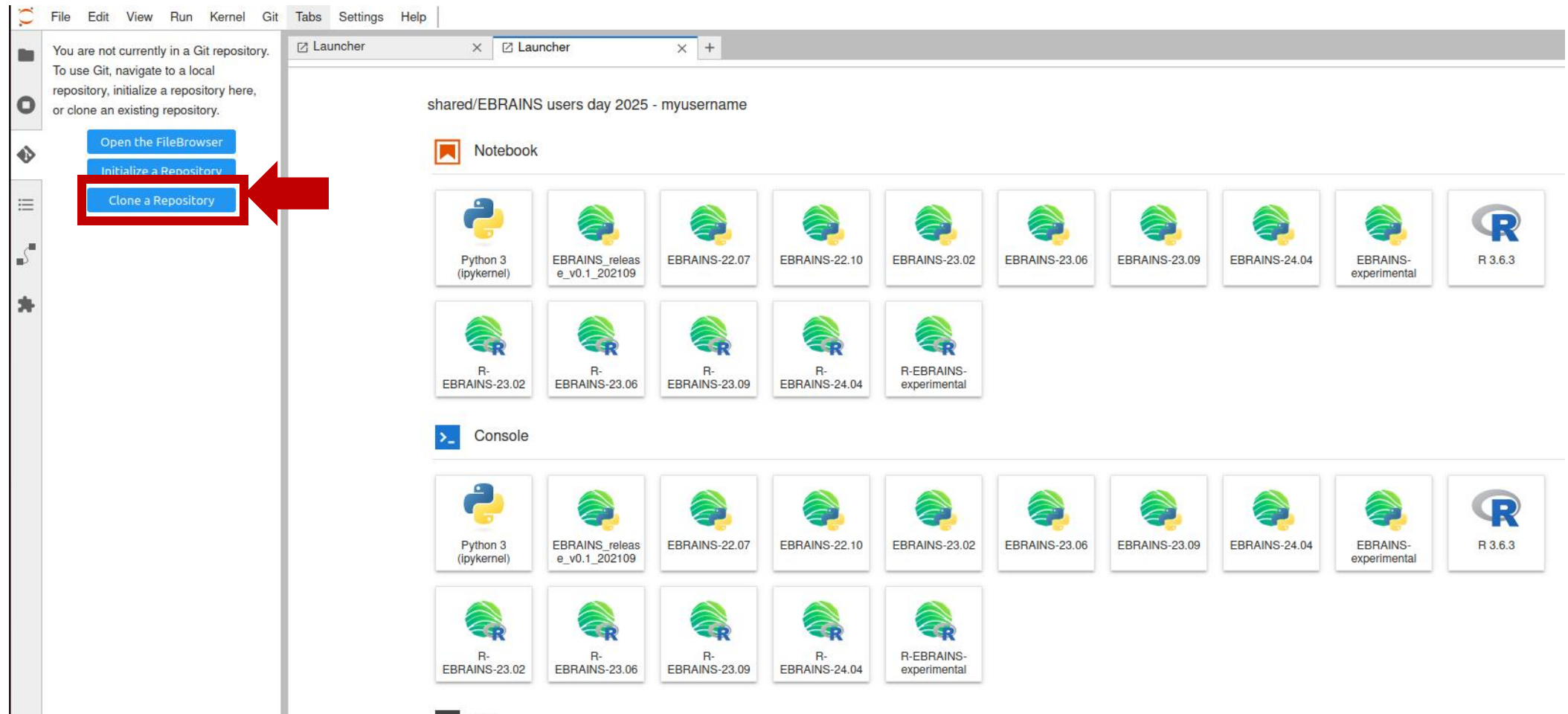
- Open the launcher**: Points to the '+' button in the top-left sidebar.
- notebook in a specific**: Points to the 'EBRAINS_release_v0.1_202109' notebook icon in the top row of the Notebook section.
- Open a console**: Points to the 'EBRAINS_release_v0.1_202109' console icon in the bottom row of the Console section.

The interface shows a grid of notebooks and consoles. The top row of notebooks includes Python 3 (ipykernel), EBRAINS_release_v0.1_202109, EBRAINS-22.07, EBRAINS-22.10, EBRAINS-23.02, EBRAINS-23.06, EBRAINS-23.09, EBRAINS-24.04, EBRAINS-experimental, and R 3.6.3. The bottom row of notebooks includes R-EBRAINS-23.02, R-EBRAINS-23.06, R-EBRAINS-23.09, R-EBRAINS-24.04, and R-EBRAINS-experimental. The console section also displays a grid of similar icons.

Retrieve the tutorial notebooks via Github

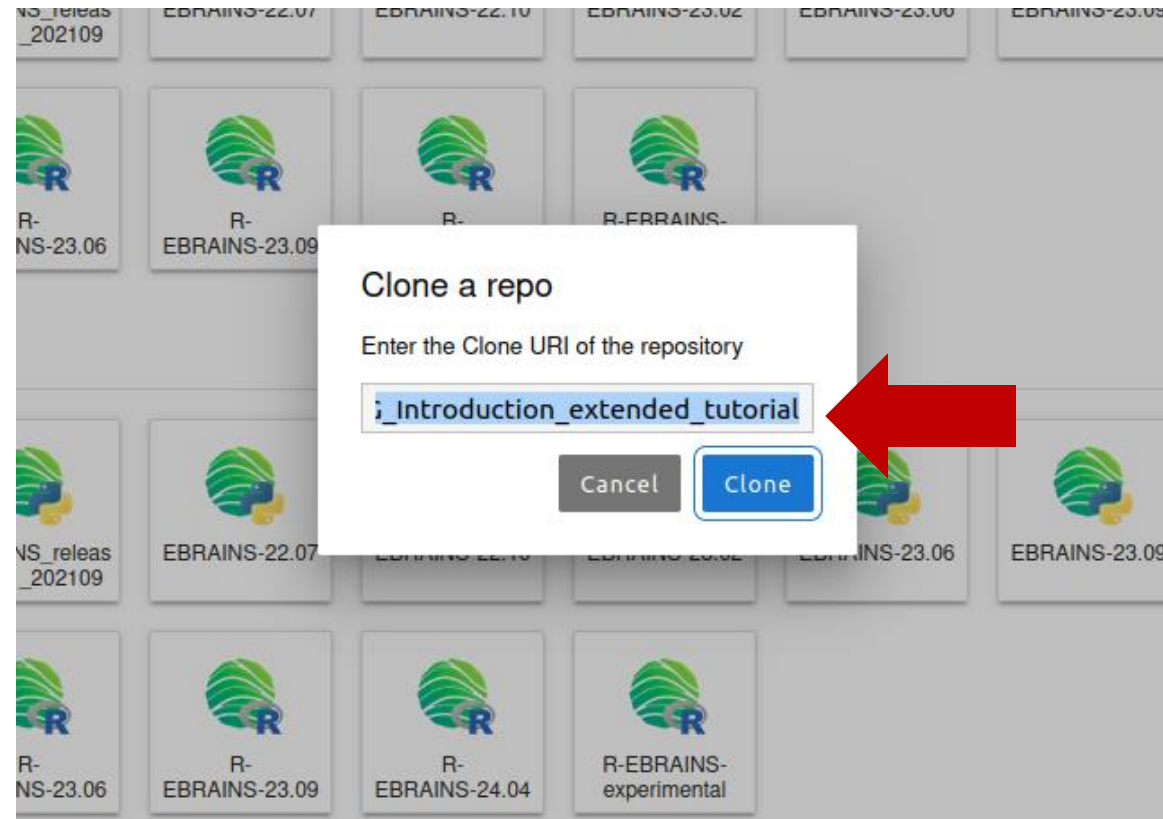


Retrieve the tutorial notebooks via Github



Retrieve the tutorial notebooks via Github

- https://github.com/Alixbonard/EBRAINS_users_day_2025_KG_Introduction_extended_tutorial



2. Searching a dataset with EBRAINS Knowledge Graph Search (KG search)

2. Search a dataset with the Knowledge Graph

Try to search a dataset related to hippocampal CA1 pyramidal neurons.

Requirements:



- free access
- 6 months old C57BL/6J mice
- Alzheimer's disease
- Electrophysiology experimental approach

Resources:

<https://search.kg.ebrains.eu/>

2. Search a dataset with the Knowledge Graph

Results:



Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Released : 2020-04-21

Accessibility : free access

Custodians : Marie, H.


In this study we analyzed the intrinsic electrophysiological properties of **CA1** excitatory hippocampal...

Alzheimer's disease **amyloidopathy** **current clamp** **whole cell patch clamp**

Resources:

<https://search.kg.ebrains.eu/>

Search a dataset with the Knowledge Graph

 DATASET

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Salgueiro-Pereira, A. R.; Marie, H.

Overview

Data descriptor

How to cite


Get data


Publications

Specimens

Related resources

How to use


DOI:  10.25493/YJFW-HPY

Released:  2020-04-21

Accessibility: free access

License: [Creative Commons Attribution 4.0 International](#)

Ethics assessment: EU-compliant

Custodians:  [Marie, H.](#)

In this study we analyzed the intrinsic electrophysiological properties of CA1 excitatory hippocampal neurons in a mouse model of Alzheimer's Disease (AD) at two age points: a presymptomatic age (3-4 months) and a symptomatic age: (9-10 months). At this latter age, this APPPS1 model harbors amyloid plaques and hippocampus-dependent cognitive alterations. Little is known about the excitability alterations in the hippocampus that correlate to these cognitive deficits. Using patch clamp electrophysiology we recorded CA1 pyramidal neurons from control littermates (Wild types) and APPPS1 slices of hippocampus at these two age points.

Study targets: [Alzheimer's disease](#)


Preparation: [in vitro](#)

Experimental approach: [electrophysiology](#)


Technique:

- [whole cell patch clamp](#)
- [current clamp](#)

Keywords:

-  [amyloidopathy](#)

Search a dataset with the Knowledge Graph

DATASET 

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Salgueiro-Pereira, A. R.; Marie, H.

Overview

Data descriptor

How to cite


Get data


Publications

Specimens

Related resources

How to use


DOI:  10.25493/YJFW-HPY

Released:  2020-04-21

Accessibility: free access

License: [Creative Commons Attribution 4.0 International](#)

Ethics assessment: EU-compliant

Custodians:  [Marie, H.](#)

In this study we analyzed the intrinsic electrophysiological properties of CA1 excitatory hippocampal neurons in a mouse model of Alzheimer's Disease (AD) at two age points: a presymptomatic age (3-4 months) and a symptomatic age: (9-10 months). At this latter age, this APPPS1 model harbors amyloid plaques and hippocampus-dependent cognitive alterations. Little is known about the excitability alterations in the hippocampus that correlate to these cognitive deficits. Using patch clamp electrophysiology we recorded CA1 pyramidal neurons from control littermates (Wild types) and APPPS1 slices of hippocampus at these two age points.

Study targets: [Alzheimer's disease](#)


Preparation: [in vitro](#)

Experimental approach: [electrophysiology](#)


Technique:





- [whole cell patch clamp](#)
- [current clamp](#)

Keywords:

-  [amyloidopathy](#)

Search a dataset with the Knowledge Graph


 DATASET



ESC

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Salgueiro-Pereira, A. R.; Marie, H.

[Overview](#)
[Data descriptor](#)
[How to cite](#)
[Get data](#)
[Publications](#)
[Specimens](#)
[Related resources](#)
[How to use](#)


DOI:  10.25493/YJFW-HPY

Released:  2020-04-21

Accessibility: free access

License: [Creative Commons Attribution 4.0 International](#)

Ethics assessment: EU-compliant

Custodians:  [Marie, H.](#)

In this study we analyzed the intrinsic electrophysiological properties of CA1 excitatory hippocampal neurons in a mouse model of Alzheimer's Disease (AD) at two age points: a presymptomatic age (3-4 months) and a symptomatic age: (9-10 months). At this latter age, this APPPS1 model harbors amyloid plaques and hippocampus-dependent cognitive alterations. Little is known about the excitability alterations in the hippocampus that correlate to these cognitive deficits. Using patch clamp electrophysiology we recorded CA1 pyramidal neurons from control littermates (Wild types) and APPPS1 slices of hippocampus at these two age points.

Study targets: [Alzheimer's disease](#)


Preparation: [in vitro](#)

Experimental approach: [electrophysiology](#)

Technique:

- [whole cell patch clamp](#)
- [current clamp](#)

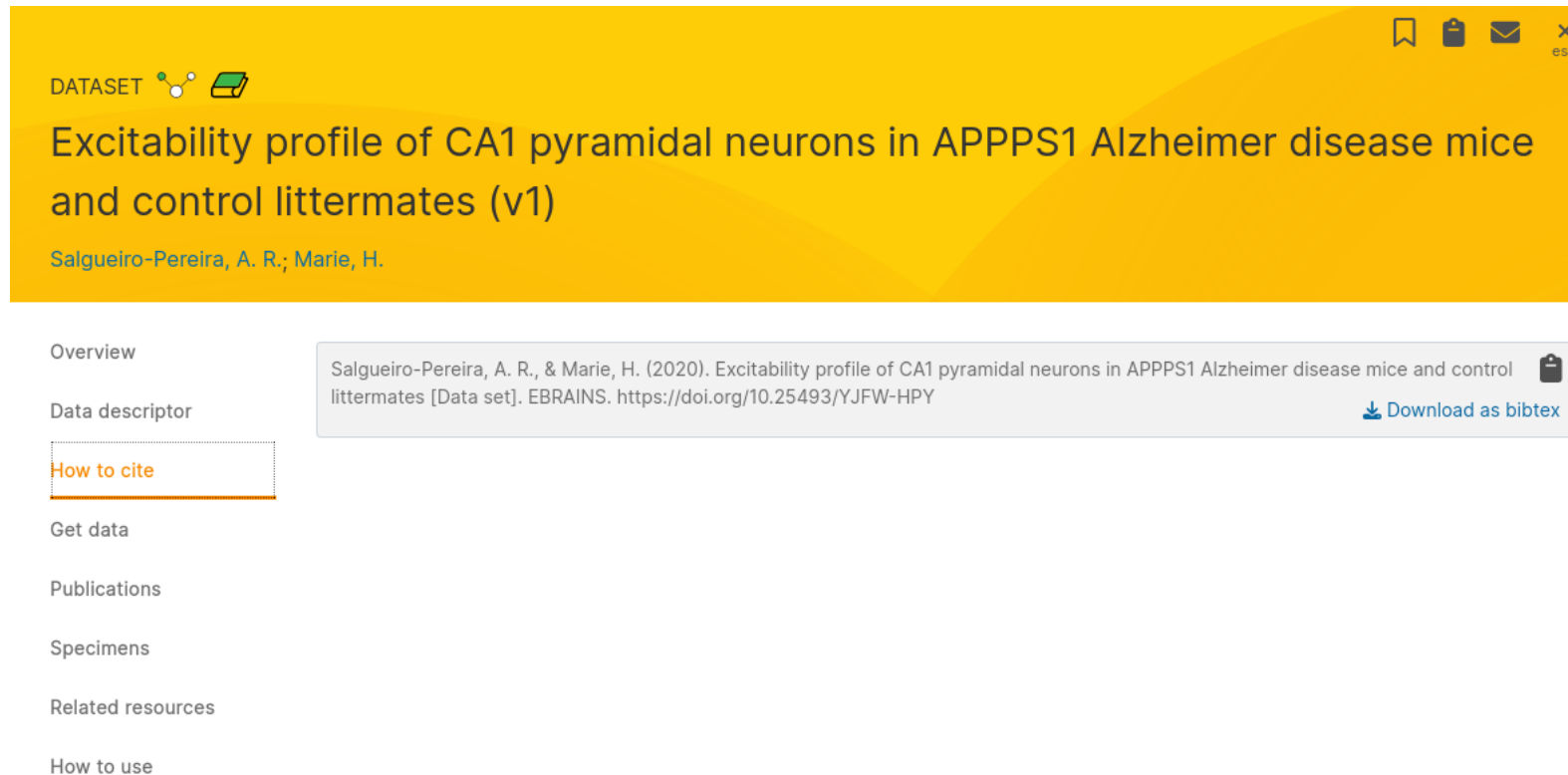
Keywords:

-  [amyloidopathy](#)



Search a dataset with the Knowledge Graph

The screenshot displays the EBRAINS dataset interface. At the top, a yellow header bar contains the word 'DATASET' with a small icon, and the dataset title 'Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)' in bold. Below the title, the authors 'Salgueiro-Pereira, A. R.; Marie, H.' are listed. On the right side of the header, there are icons for bookmark, download, email, and a close button labeled 'ESC'. Below the header, a sidebar on the left lists navigation options: 'Overview' (selected), 'Data descriptor', 'How to cite', 'Get data', 'Publications', 'Specimens', 'Related resources', and 'How to use'. The main content area is titled 'HBP Data Descriptor' and contains the following information:
TITLE
Excitability profile of CA1 pyramidal neurons in APP-PS1 Alzheimer disease mice and control littermates.
AUTHORS
Ana Rita Salgueiro-Pereira¹, Hélène Marie¹
AFFILIATIONS
1. Université Côte d'Azur, CNRS UMR 7275, IPMC, Valbonne, France
corresponding author(s): Hélène MARIE (marie@ipmc.cnrs.fr)
ABSTRACT
In this study we analyzed the intrinsic electrophysiological properties of CA1 excitatory hippocampal neurons in a mouse model of Alzheimer's Disease (AD) at two age points: a presymptomatic age (3-4 months) and a symptomatic age: (9-10 months). At this latter age, this APPPS1 model harbors amyloid plaques and hippocampus-dependent cognitive alterations. Little is known about the excitability alterations in the hippocampus that correlate to these cognitive deficits. Using patch clamp electrophysiology we recorded CA1 pyramidal neurons from control littermates (Wild types) and APPPS1 slices of hippocampus at these two age points. We provide these data for HBP users.

Search a dataset with the Knowledge Graph



The screenshot shows the EBRAINS dataset interface. At the top, a yellow banner displays the dataset title "Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)" and the authors "Salgueiro-Pereira, A. R.; Marie, H.". Below the banner, a sidebar on the left lists navigation options: Overview, Data descriptor, How to cite (highlighted with an orange border), Get data, Publications, Specimens, Related resources, and How to use. The main content area shows the citation: "Salgueiro-Pereira, A. R., & Marie, H. (2020). Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates [Data set]. EBRAINS. https://doi.org/10.25493/YJFW-HPY" and a "Download as bibtex" link.

DATASET  

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Salgueiro-Pereira, A. R.; Marie, H.

Overview

Data descriptor

How to cite


Get data

Publications

Specimens

Related resources



How to use

Salgueiro-Pereira, A. R., & Marie, H. (2020). Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates [Data set]. EBRAINS. <https://doi.org/10.25493/YJFW-HPY> 

[Download as bibtex](#)

Search a dataset with the Knowledge Graph

The screenshot shows the EBRAINS dataset interface. At the top, a yellow banner displays the dataset title: "Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)" by "Saigueliro-Pereira, A. R.; Marie, H.". Below the banner, a sidebar on the left contains navigation links: Overview, Data descriptor, How to cite, Get data (highlighted with an orange bar), Publications, Specimens, Related resources, and How to use. The main content area shows a "Filter by" dropdown set to "none" and a file list of 75 files. The first file, "ext-d000001_ADNeuronModel_pub", is expanded, revealing sub-items: "APPPS1_mouse_model_3-4_months", "APPPS1_mouse_model_9-10_months", "control_3-4_months", "control_9-10_months", "HBP-DataDescriptor_ADNeuronModel_v2.pdf", and "Licence (CCBY).pdf". To the right of the file list, a folder icon is shown with the name "ext-d000001_ADNeuronModel_pub" and a "Download dataset" button. Below the button, a note states: "By downloading the Dataset you agree to the [Terms of use](#)".

DATASET  

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Saigueliro-Pereira, A. R.; Marie, H.

Overview

Data descriptor

How to cite

Get data

Publications

Specimens

Related resources

How to use

Filter by none

75 files

Search the files...

- ext-d000001_ADNeuronModel_pub
 - APPPS1_mouse_model_3-4_months
 - APPPS1_mouse_model_9-10_months
 - control_3-4_months
 - control_9-10_months
 - HBP-DataDescriptor_ADNeuronModel_v2.pdf
 - Licence (CCBY).pdf



Name: ext-d000001_ADNeuronModel_pub

Download dataset

By downloading the Dataset you agree to the [Terms of use](#)

Please alert us at curation-support@ebrains.eu for errors or quality concerns regarding the dataset, so we can forward this information to the Data Custodian responsible.

Search a dataset with the Knowledge Graph

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Salgueiro-Pereira, A. R.; Marie, H.

Overview

Data descriptor

How to cite

Get data

Publications

Specimens

Related resources



How to use

Publications :
Vitale, P., Salgueiro-Pereira, A. R., Lupascu, C. A., Willem, M., Migliore, R., Migliore, M., & Marie, H. (2021). Analysis of Age-Dependent Alterations in Excitability Properties of CA1 Pyramidal Neurons in an APPPS1 Model of Alzheimer's Disease. *Frontiers in Aging Neuroscience*, 13. <https://doi.org/10.3389/fnagi.2021.668948>
DOI: [10.3389/fnagi.2021.668948](https://doi.org/10.3389/fnagi.2021.668948)

Please alert us at curation-support@ebrains.eu for errors or quality concerns regarding the dataset, so we can forward this information to the Data Custodian responsible.

Search a dataset with the Knowledge Graph

The screenshot displays the EBRAINS Knowledge Graph interface. At the top, a yellow header bar contains the word 'DATASET' with a small icon, followed by the dataset title 'Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)' and the authors 'Salgueiro-Pereira, A. R.; Marie, H.'. Below the header, a sidebar on the left lists navigation options: Overview, Data descriptor, How to cite, Get data, Publications, Specimens (highlighted with an orange border), Related resources, and How to use. The main content area is titled 'Select the items of the tree to get more details about the individual elements.' and shows a tree structure under the 'Specimen' category. The tree lists 13 subjects (S22 to S77) and their corresponding tissue sample collections. A legend at the bottom indicates that yellow circles represent 'Subject' and teal circles represent 'Tissue sample collection'. On the right side, a summary panel provides statistics: 13 subjects, 13 tissue sample collections, species 'Mus musculus' (13 subjects), sex 'male' (13 subjects), and pathology 'Alzheimer's disease' (6 subjects, 1 tissue sample collection).

DATASET  

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Salgueiro-Pereira, A. R.; Marie, H.

Overview

Data descriptor

How to cite

Get data

Publications

Specimens

Related resources

How to use

Select the items of the tree to get more details about the individual elements.

Specimen

- Subject S22
 - Tissue sample collection S22
- Subject S23
 - Tissue sample collection S23
- Subject S24
 - Tissue sample collection S24
- Subject S26
 - Tissue sample collection S26
- Subject S27
 - Tissue sample collection S27
- Subject S31
 - Tissue sample collection S31
- Subject S75
 - Tissue sample collection S75
- Subject S76
 - Tissue sample collection S76
- Subject S77
 - Tissue sample collection S77

● Subject ● Tissue sample collection

Specimen

Subjects: 13

Tissue sample collections: 13

Species: [Mus musculus](#)

- 13 subjects

Sex: [male](#)

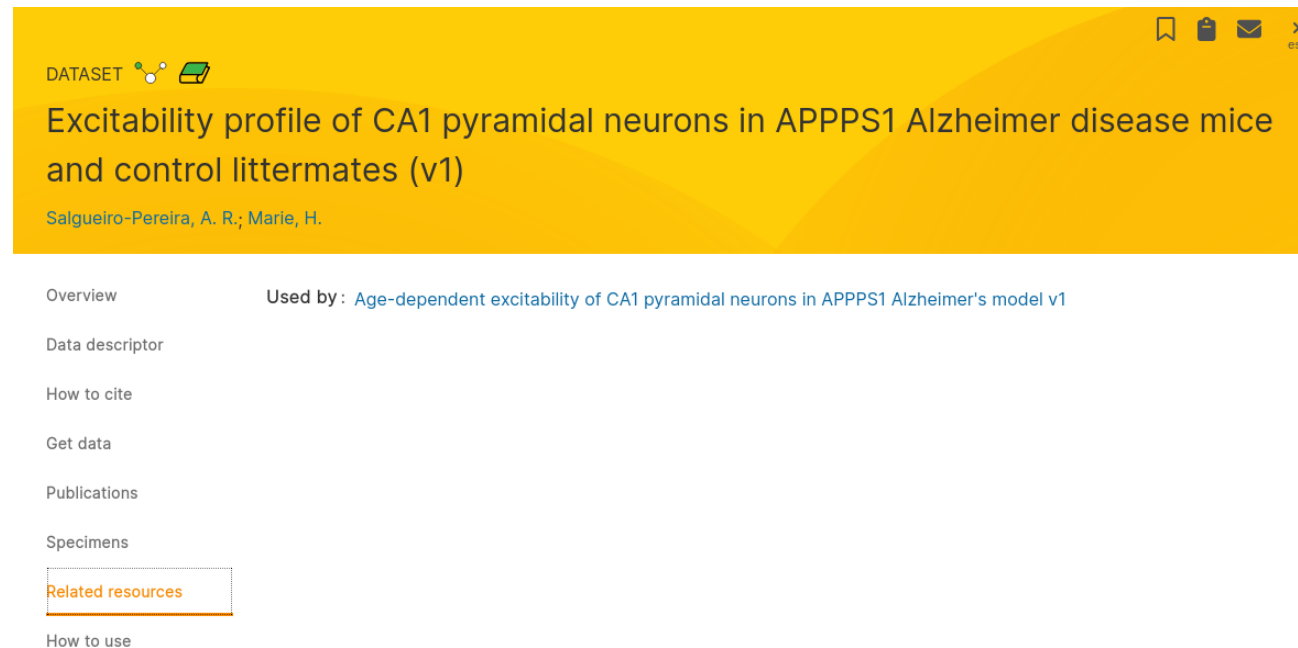
- 13 subjects

Pathology: [Alzheimer's disease](#)

- 6 subjects
- 1 tissue sample collection

Please alert us at curation-support@ebrains.eu for errors or quality concerns regarding the dataset, so we can forward this information to the Data Custodian responsible.

Search a dataset with the Knowledge Graph



The screenshot shows a dataset page with a yellow header. The header contains the word 'DATASET' with a small icon, and the title 'Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)'. Below the title is the author 'Saqueiro-Pereira, A. R.; Marie, H.'. In the top right corner of the header are icons for bookmark, folder, email, and a close button labeled 'esc'. Below the header is a sidebar with links: Overview, Data descriptor, How to cite, Get data, Publications, Specimens, Related resources (highlighted with an orange border), and How to use. To the right of the sidebar, under 'Used by:', is a link to 'Age-dependent excitability of CA1 pyramidal neurons in APPPS1 Alzheimer's model v1'.

DATASET

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Saqueiro-Pereira, A. R.; Marie, H.

Overview Used by: [Age-dependent excitability of CA1 pyramidal neurons in APPPS1 Alzheimer's model v1](#)

Data descriptor

How to cite

Get data

Publications

Specimens


Related resources

How to use

Please alert us at curation-support@ebrains.eu for errors or quality concerns regarding the dataset, so we can forward this information to the Data Custodian responsible.

Search a dataset with the Knowledge Graph

The screenshot shows the EBRAINS dataset interface. At the top, a yellow header bar contains the word 'DATASET' with a molecular icon, the dataset title 'Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)', and the authors 'Salgueiro-Pereira, A. R.; Marie, H.'. Below the header, a sidebar on the left lists navigation options: Overview, Data descriptor, How to cite, Get data, Publications, Specimens, Related resources, and 'How to use' (which is highlighted with an orange border). The main content area on the right is titled 'Learning resources :' and contains a link to a 'Case study of data reuse'. Below this, it has a section for 'Programmatic access to metadata :' explaining how to use the EBRAINS Knowledge Graph (KG) and providing a 'KG Query Builder' button. Further down, it describes the 'KG REST-API' and 'KG Core SDKs'. At the bottom, a footer note asks users to report errors to curation-support@ebrains.eu.

DATASET 

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Salgueiro-Pereira, A. R.; Marie, H.

- Overview
- Data descriptor
- How to cite
- Get data
- Publications
- Specimens
- Related resources
- How to use**

Learning resources :
[Case study of data reuse: Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates](#)

Programmatic access to metadata :
To make programmatic use of the (meta-)data of EBRAINS, you have different options to interact with the [EBRAINS Knowledge Graph \(KG\)](#):

KG Query Builder
With the [KG Query Builder](#), you can design your own query to retrieve metadata for this instance and those of the same type conveniently via UI without the requirement of learning a graph query language (see the [tutorial](#)). You can also save the query and use it with the REST-API and the KG Core SDKs (see below).

KG REST-API
You can use the [KG REST-API](#) to access and/or manipulate metadata on the EBRAINS KG as well as to run queries saved previously in the KG Query Builder.

KG Core SDKs
The [KG Core SDKs](#) provide convenient ways to authenticate and make use of the functionality of the KG REST-API with your favorite programming language (currently available for Python and JavaScript/TypeScript).

Please alert us at curation-support@ebrains.eu for errors or quality concerns regarding the dataset, so we can forward this information to the Data Custodian responsible.

Search a dataset with the Knowledge Graph

The screenshot shows the EBRAINS dataset interface. At the top, a yellow header bar contains the word 'DATASET' with a small icon, and the dataset title 'Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)' in large black text. Below the title, the authors 'Salgueiro-Pereira, A. R.; Marie, H.' are listed. A sidebar on the left contains a list of navigation links: Overview, Data descriptor, How to cite, Get data, Publications, Specimens, Related resources, and How to use (which is highlighted with an orange border). The main content area on the right is divided into sections: 'Learning resources' with a link to a case study, 'Programmatic access to metadata' with instructions on using the EBRAINS Knowledge Graph (KG), 'KG Query Builder' with a detailed explanation of its functionality and a 'Build a query for this instance' button, 'KG REST-API' with instructions on using the REST-API, and 'KG Core SDKs' with information about the SDKs. At the bottom, a footer note asks users to report errors or quality concerns to curation-support@ebrains.eu.

DATASET

Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates (v1)

Salgueiro-Pereira, A. R.; Marie, H.

Overview

Data descriptor

How to cite

Get data

Publications

Specimens

Related resources

How to use

Learning resources :

[Case study of data reuse: Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and control littermates](#)

Programmatic access to metadata :

To make programmatic use of the (meta-)data of EBRAINS, you have different options to interact with the [EBRAINS Knowledge Graph \(KG\)](#):

KG Query Builder

With the [KG Query Builder](#), you can design your own query to retrieve metadata for this instance and those of the same type conveniently via UI without the requirement of learning a graph query language (see the [tutorial](#)). You can also save the query and use it with the REST-API and the KG Core SDKs (see below).

Build a query for this instance

KG REST-API

You can use the [KG REST-API](#) to access and/or manipulate metadata on the EBRAINS KG as well as to run queries saved previously in the KG Query Builder.

KG Core SDKs

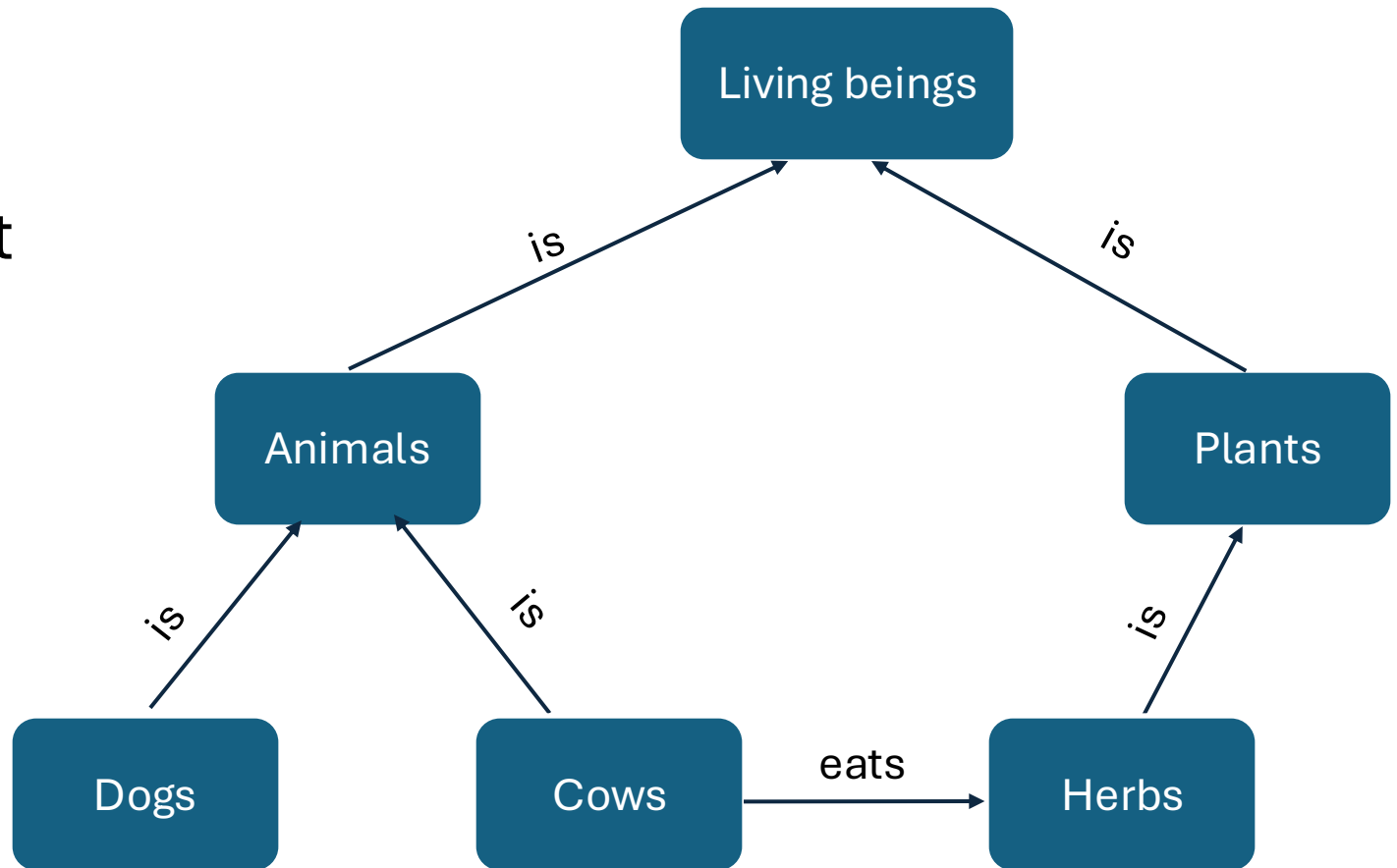
The [KG Core SDKs](#) provide convenient ways to authenticate and make use of the functionality of the KG REST-API with your favorite programming language (currently available for Python and JavaScript/TypeScript).

Please alert us at curation-support@ebrains.eu for errors or quality concerns regarding the dataset, so we can forward this information to the Data Custodian responsible.

3. Retrieving a dataset in a Jupyter notebook with the Fairgraph library

What is a Knowledge Graph?

- A way to represent entities and their relationships



Retrieve datasets and their metadata with Python : the fairgraph library

- Client and authentication token

1. Client with: JupyterLab

```
1: from fairgraph import KGClient
kg_client = KGClient(host="core.kg.ebrains.eu") # if you want to have access to
```

2. Client with: Local system

You can have access to your token in the [KG Editor](#) in your profile button (copy token to clipboard)

```
1: # in a local terminal:
# cd </path/to/tutorial/folder> # change the directory where the tutorial file .
# export KG_AUTH_TOKEN=<paste the token> (e.g. 'eyJhbGciOi..nPq')
#
! pip install fairgraph # to install fairgraph in your python environment
# in your jupyter notebook:
from fairgraph import KGClient
import os

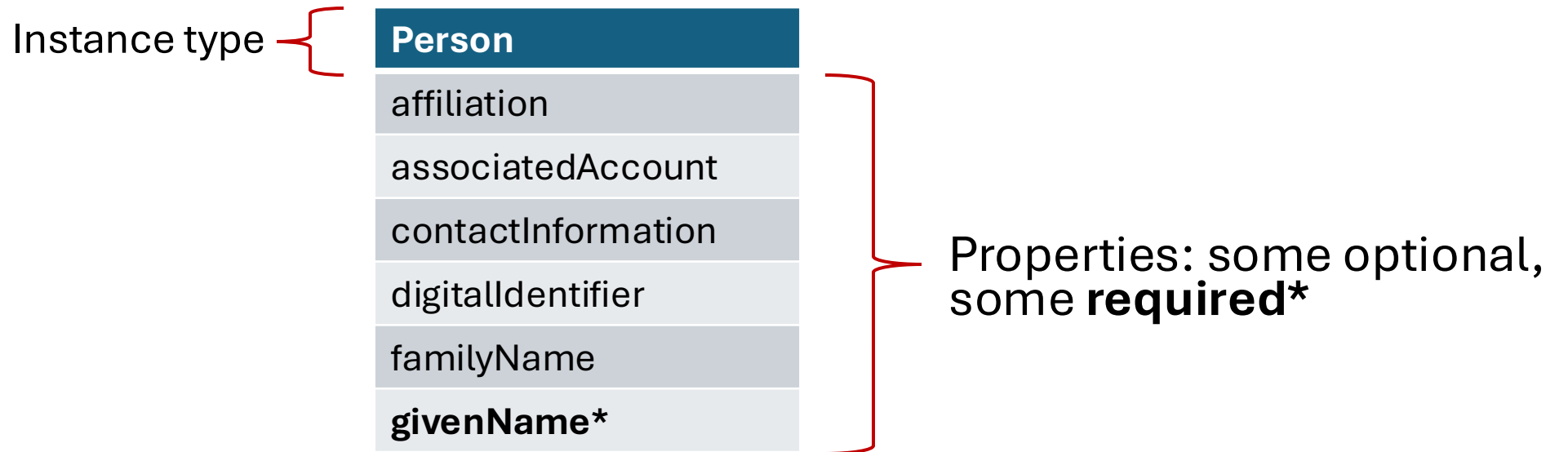
#kg_token = os.environ['KG_AUTH_TOKEN']
#kg_client = KGClient(host="core.kg.ebrains.eu", token=kg_token)

# alternative approach 1:
# run in an another cell - get the token everytime you need
# kg_token = KGClient(host='core.kg.ebrains.eu').user_info()

# alternative approach 2:
# copy your token from the KG Editor directly in the jupyter notebook as below
kg_token = "yourtoken"
kg_client = KGClient(host="core.kg.ebrains.eu", token=kg_token)
```

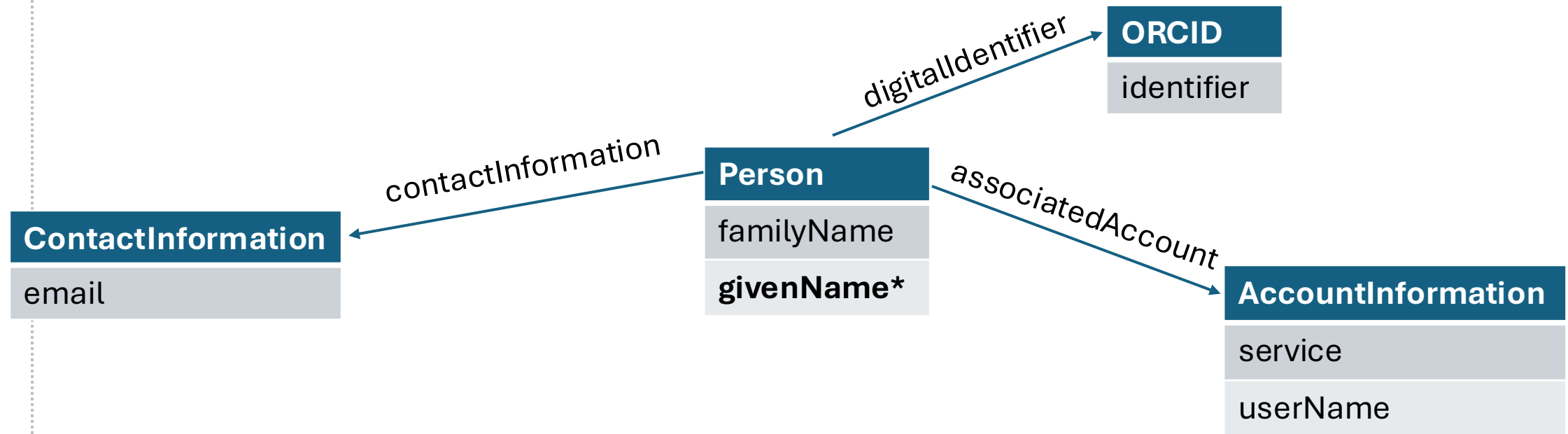
OpenMINDS

- Need to be able to infer information (e.g. a Person must have a name)
- Metadata framework for linked data: openMINDS
- openMINDS contains schema specifications for multiple metadata models



OpenMINDS

- Properties may expect links to other instances



Find schema specifications and more detailed explanations here:

<https://openminds-documentation.readthedocs.io/en/latest>

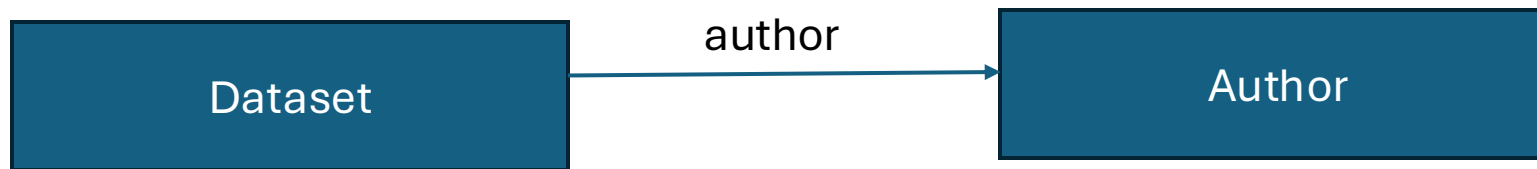
The fairgraph Python library

- Available here: <https://github.com/HumanBrainProject/fairgraph/>
- Uses a client to access the KG: needs your credentials
 - Retrieve a token from the KG editor or use the EBRAINS Jupyter lab (with a collab, for example)
- Important classes:
 - KGClient
 - KGObject: parent class for every object on the KG
 - KGProxy: representation of a KGObject: type and identifier are known, but not its metadata
 - KGQuery: Representation of one or more KGObjects identified by a range of possible types and by some of their metadata, but whose specific identifier(s) is/are not known.

The fairgraph Python library

- Requires some understanding of openMINDS
- Does not follow links by default

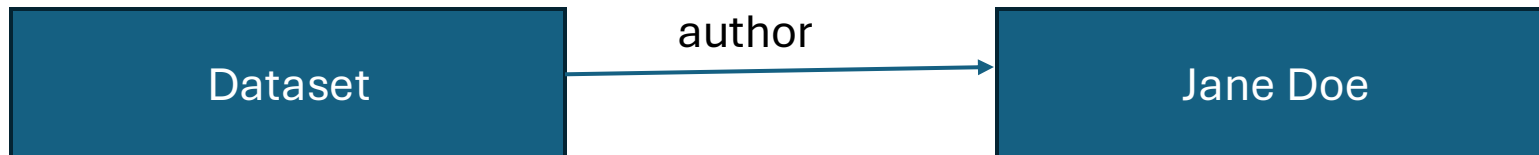
```
Dataset.by_name('Dataset name', kg_client)
```



The fairgraph Python library

- Use the follow_links parameter to follow links

```
Dataset.by_name('Dataset name', kg_client, follow_links = {"authors": {}})
```



Retrieve datasets and their metadata with Python : the fairgraph library

- Retrieving the dataset:
 - Using the dataset name
 - Using the ID
 - Using the DOI

Retrieve datasets and their metadata with Python : the fairgraph library

- Downloading the dataset

Downloading the dataset

Once the dataset retrieved in a fairgraph object, you can download the all dataset into your local system (your working directory for example) you can use the property `download` . To do so, you have to define your `local_path` . Then, you must use download property with the `local_path`, the `kg_client` as arguments. You must accept the terms of use as shown below.

```
[ ]: # Get your local path
import os

local_path = os.path.abspath('path/to/download/directory')
# or
local_path = os.getcwd() # get the working directory

[ ]: dataset_version.download(local_path='...', client=kg_client, accept_terms_of_use=True)
```

Exploring dataset metadata

- Authors

Authors

In the previous part, you learned how to retrieve a dataset with `Dataset` or `DatasetVersion` presenting explicit and non-explicit metadata. Here you will learn how to explicit them. To explicitly retrieve authors you have to use the argument `follow_links`. When you fetch a dataset with `Dataset` or `DatasetVersion`, you fetch default nodes (i.e. explicit metadata), with `follow_link` you add more nodes. Thereby, you can access the target metadata. Let's try with authors:

You want to retrieve the names of the dataset's authors. By consulting the [Dataset schema specifications](#) in the openMINDS documentation, you can see which link you need to follow to obtain that piece of information.

```
[ ]: # Retrieve the dataset object:
dataset = Dataset.by_name('Excitability profile of CA1 pyramidal neurons in APPPS1 Alzheimer disease mice and cor
print(f'This dataset has {len(dataset.authors)} authors. {dataset.authors[0].family_name} {dataset.authors[0].giv
```

Exploring dataset metadata

- **Exercise:** Find the following metadata:
 - Datafiles
 - Publication
 - DOI
 - Experimental Approach
 - Techniques
 - Studies Specimens

Finding datasets with fairgraph queries: A brief introduction

4. Finding datasets with fairgraph queries: a brief introduction

This part is a brief introduction to queries. Please see the tutorial [EBRAINS Knowledge Graph Advanced](#) for more information. One other use of Fairgraph library is the Queries. Queries are useful if you need to search a dataset using filters. As presented above, you can use `DatasetVersion` to obtain a list of potential datasets of interest. For more information you can consult this [link](#)

Let's see an example.

Run this cell to obtain datasets having "neurons" in their name:

```
In [ ]: datasets = DatasetVersion.list(kg_client, name="neurons")
        for dataset in datasets:
            print(dataset.name)
```

Let's try "patch-clamp" technique:

```
In [ ]: datasets = DatasetVersion.list(kg_client, name="patch-clamp")
        for dataset in datasets:
            print(dataset.name)
```

As you see, you can obtain datasets corresponding to your filter. Fairgraph library is still in development, so queries may not run properly. You can use the [EBRAINS query builder](#) to directly build queries and run them on the KG. For more information regarding this tool, click [here](#).

A space just for your notes:

Exercise in autonomy

5. Exercise in autonomy

For this last part, you will have to reproduce what you have learned in the previous parts. Remember that you can look up schema specifications on the [openMINDS documentation](#).

List of dataset:

- 10.25493/YJFW-HPY
- 10.25493/VAV5-BXU
- 10.25493/CHJG-7QC
- 10.25493/M1AQ-3AC
- 10.25493/M1V0-WE3
- 10.25493/3NTS-Q0B
- 10.25493/JNFA-HDP
- 10.25493/5GE0-6MF

1. Exercise: Use fairgraph query to find a dataset of interest

2. Exercise: Explore the dataset metadata. Can you retrieve the authors? See if there is more datasetversion? Retrieve the date of publication? The publication DOI? You can start with these questions and explore more relevant metadata

3. Exercise: Retrieve dataset files. Can you obtain the name of each file?

4. Exercise: Download the dataset



The EBRAINS Knowledge Graph

Extended Tutorial: Practice case studies for data reuse

Laura Morel, Alix Bonard, Peyman Najafi

NeuroPSI Paris-Saclay Institute of Neuroscience, CNRS, Université Paris-Saclay, France.

Introduction

Table of contents:

Hands-on case study: Figure reproduction with and without metadata

- Explore the datafile
- Reproduce the figure 4C from [The microcircuits of striatum in silico. Proceedings of the National Academy of Sciences](https://doi.org/10.1073/pnas.2000671117) (DOI: [10.1073/pnas.2000671117](https://doi.org/10.1073/pnas.2000671117)) without metadata
- Reproduce the same figure 4C with metadata

Hands-on a learning resource: Salgueiro-Peirera-Marie-2018 (optional)

- Explore the KG Search to find a dataset with fairgraph queries find the Salgueiro Pererra Marie case study
- Follow the case study

Finding a dataset with fairgraph queries

- Authentication / authorization

1. Client with: JupyterLab

```
1: from fairgraph import KGClient
kg_client = KGClient(host="core.kg.ebrains.eu") # if you want to have access to
```

2. Client with: Local system

You can have access to your token in the [KG Editor](#) in your profile button (copy token to clipboard)

```
1: # in a local terminal:
# cd </path/to/tutorial/folder> # change the directory where the tutorial file .
# export KG_AUTH_TOKEN=<paste the token> (e.g. 'eyJhbGciOi..nPq')
#
! pip install fairgraph # to install fairgraph in your python environment
# in your jupyter notebook:
from fairgraph import KGClient
import os
#kg_token = os.environ['KG_AUTH_TOKEN']
#kg_client = KGClient(host="core.kg.ebrains.eu", token=kg_token)
# alternative approach 1:
# run in an another cell - get the token everytime you need
# kg_token = KGClient(host='core.kg.ebrains.eu').user_info()
# alternative approach 2:
# copy your token from the KG Editor directly in the jupyter notebook as below
kg_token = "yourtoken"
kg_client = KGClient(host="core.kg.ebrains.eu", token=kg_token)
```

Finding a dataset with fairgraph queries

Fairgraph Query

You need to retrieve a dataset related to striatal interneurons with low spiking threshold with `DatasetVersion.list`

```
In [ ]: "Paste your code here"
```

Download the dataset

To be able to use the dataset, you need to download it. You can use the property `download` of `DatasetVersion`. Don't forget to use `follow_links` to retrieve the dataset files.

```
In [ ]: "Paste your code here"

with zipfile.ZipFile(dataset_path, "r") as z:
    z.extractall("downloads")
```

Checking the dataset - Solution

Run the following cell to check the dataset and to download it.

```
In [ ]: import os
import sys
sys.path.insert(0, os.path.join(os.getcwd(), ".local"))
```

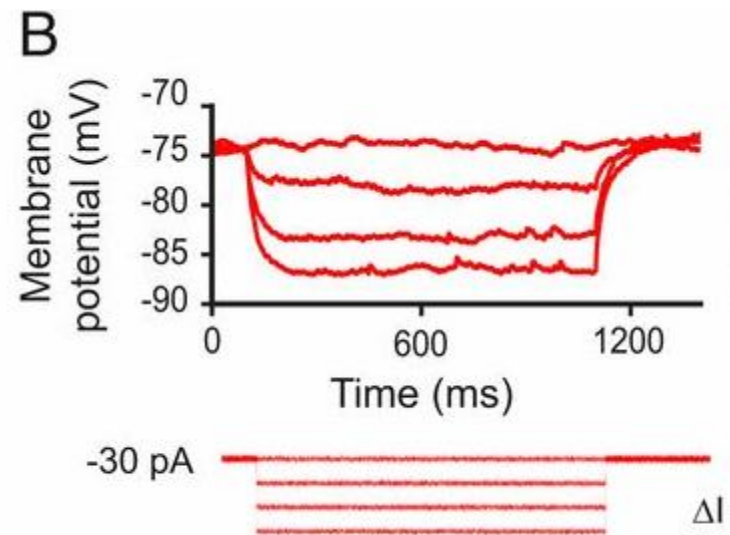
```
In [ ]: import fairgraph.openminds.core as omcore
import logging
import warnings
import zipfile
```

2. Exploring and visualizing the data

- **Exploring the retrieve dataset**
 - **Exercise:** Explore the data descriptor to understand the dataset structure

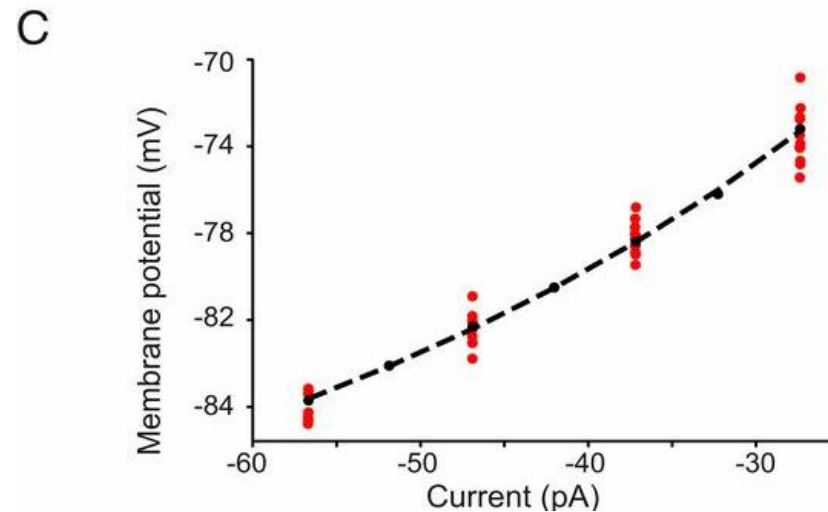
2. Exploring and visualizing the data

- **Exploring the retrieve dataset**
 - **Exercise:** Explore the data descriptor to understand the dataset structure
 - Figure 4B:



3. Reproducing figure without metadata

- **Exploring the retrieve dataset**
 - **Exercise:** Modify the code below in order to reproduce the Figure 4B.



4. Reproducing figure with metadata

Retrieving in-depth metadata

4. Reproducing figure with metadata

With metadata some step presented above can be skip. Let's have a look. The first step was to find current amplitudes values. This step could be very challenging if you don't know how to start and retrieve information as currents. With metadata, you can have a beginning of information that guides you toward figure reproduction. This dataset is quite challenging, therefore, you still need to go through all the membrane potential retrieving steps. However, it is simplifying the process.

Retrieving in-depth metadata

The aim of this part, is to show you why metadata are important for data reuse and how we can use them.

This part might be scary for some of you. In the future, the in-depth curation team will provide more tutorials to retrieve in-depth metadata. This part is an example of in-depth metadata retrieving. If you want to play around, do. If you don't want, just run the code.

The objective is to retrieve the current amplitudes.

```
1 [ ]: import fairgraph.openminds.core as omcore
import fairgraph.openminds.stimulation as omstim
import fairgraph.openminds.ephys as ephys
import json

follow_links = {"has_parts" : {}} # create a short cut with dataset_version and stimulation activity

dataset_version = omcore.DatasetVersion.list(kg_client, digital_identifier__identifier = '10.25493/5GE0-6MF', fol

# Part 1: in-depth metadata retrieving

stimulation_activities = [part for part in dataset_version.has_parts if isinstance(part, omstim.StimulationActivi
stimuli = []

for stimulation_activity in stimulation_activities:
    # We find every unique EphysStimulus associated with the dataset
    stimulus = stimulation_activity.stimuli.resolve(kg_client)
    if stimulus not in stimuli:
        stimuli.append(stimulus)
    # print(stimuli)

specification_json = stimuli[0].specifications.resolve(kg_client).configuration # We retrieve the configuration i
specification = json.loads(specification_json) # We convert the JSON string into a python dictionary
l1_current_value = specification["amplitudes"]
print(f'current amplitudes: {l1_current_value}')
```

Conclusion