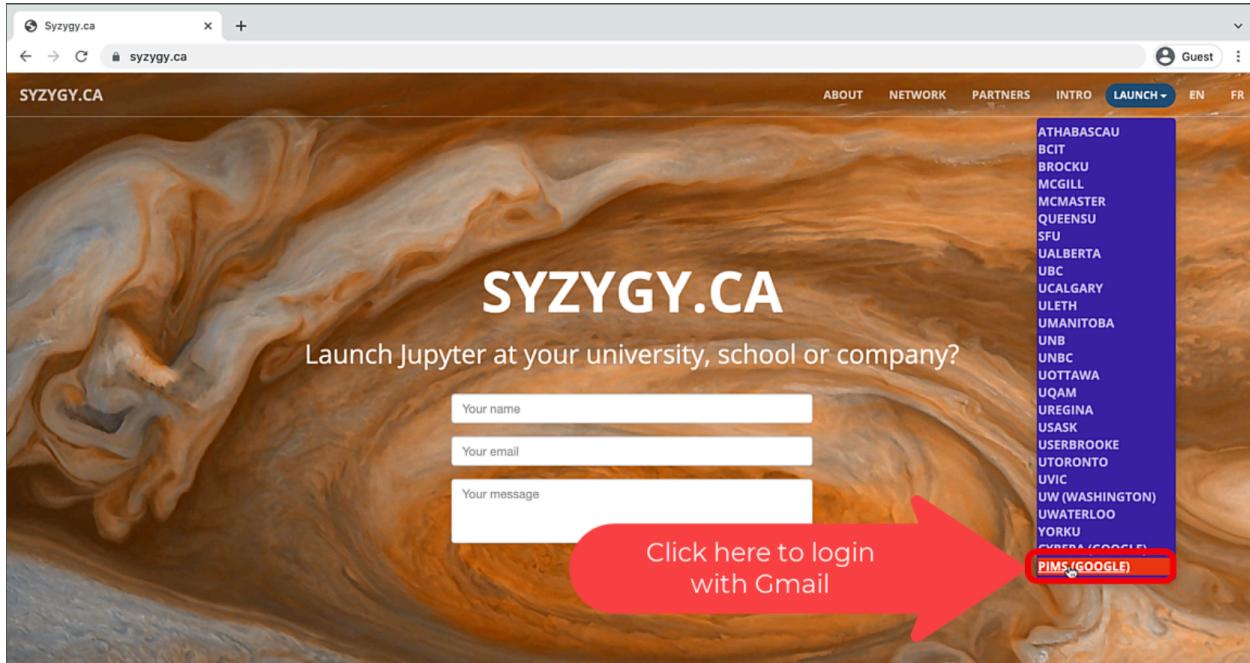
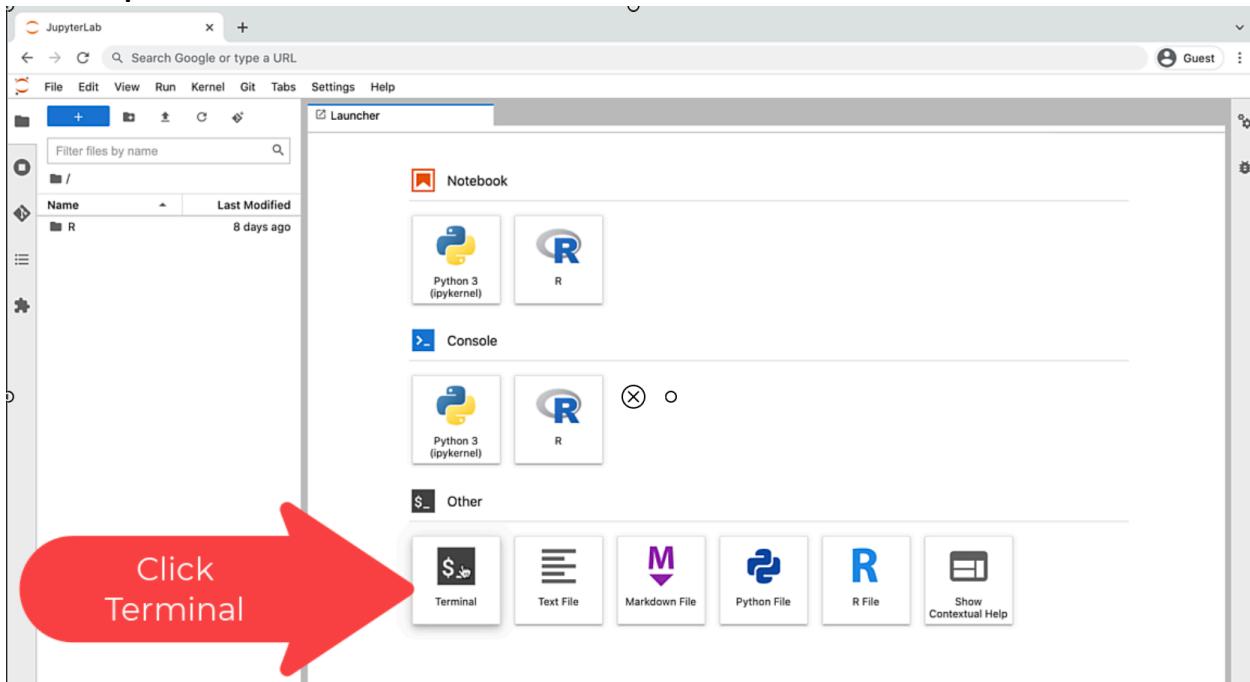


Scholar Metrics Scraper Instructions

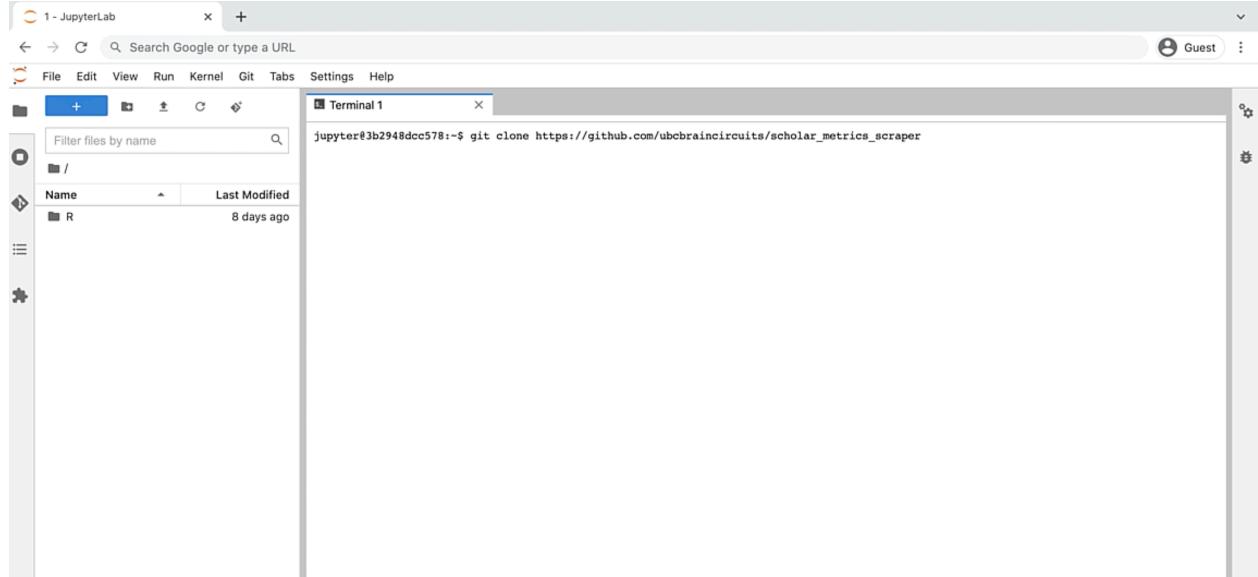
1. Go to syzygy.ca, click Launch, and either sign in with your Gmail or institutional account.



2. Open Terminal

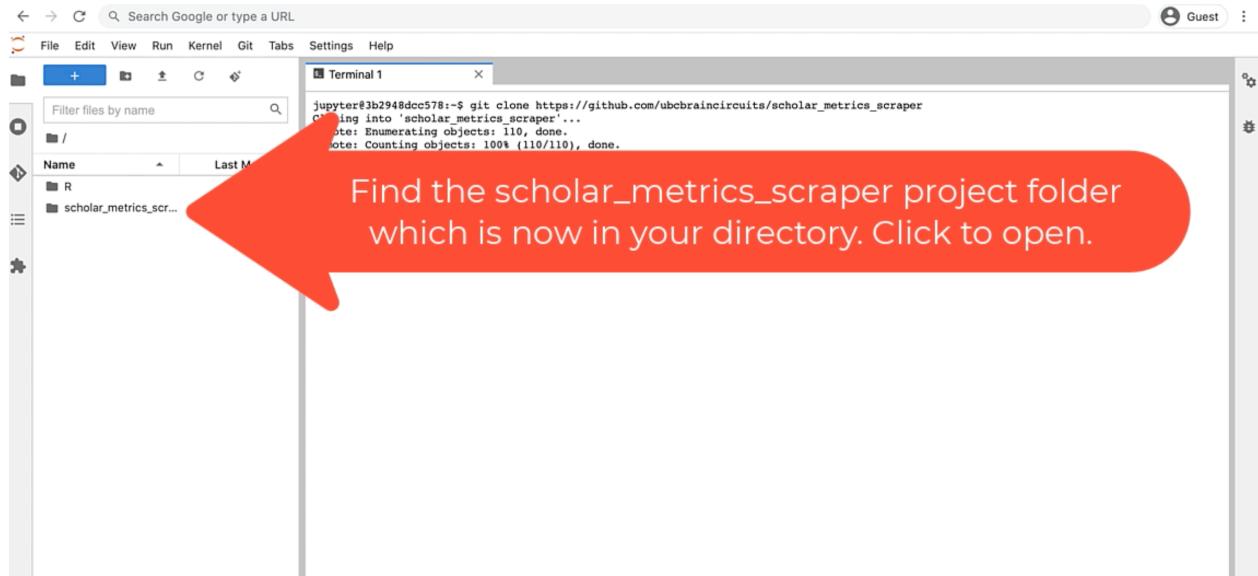


3. Clone the project from Git. Type "git clone https://github.com/ubcbraincircuits/scholar_metrics_scraper" and press enter.



A screenshot of the JupyterLab interface. On the left is a file browser showing a directory structure with an 'R' folder. On the right is a terminal window titled 'Terminal 1' containing the command: `jupyter@3b2948dcc578:~$ git clone https://github.com/ubcbraincircuits/scholar_metrics_scraper`. The terminal output shows the cloning process starting.

The project folder should now be in your directory



A screenshot of the JupyterLab interface. On the left is a file browser showing a directory structure with an 'R' folder and a newly cloned 'scholar_metrics_scraper' folder. A large red arrow points from the terminal window in the previous screenshot to this 'scholar_metrics_scraper' folder in the file browser. A callout bubble contains the text: 'Find the scholar_metrics_scraper project folder which is now in your directory. Click to open.'

4. Install scholarly. Type “pip install scholarly” and press enter.

```
jupyter@3b2948dcc578:~$ git clone https://github.com/ubcbraincircuits/scholar_metrics_scraper
Cloning into 'scholar_metrics_scraper'...
remote: Enumerating objects: 110, done.
remote: Total 110 (delta 45), reused 0 (delta 0), pack-reused 0
remote: Compressing objects: 100% (106/106), done.
remote: Total 110 (delta 45), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (110/110), 2.43 MiB | 5.97 MiB/s, done.
Resolving deltas: 100% (45/45), done.
jupyter@3b2948dcc578:~$ pip install scholarly
```

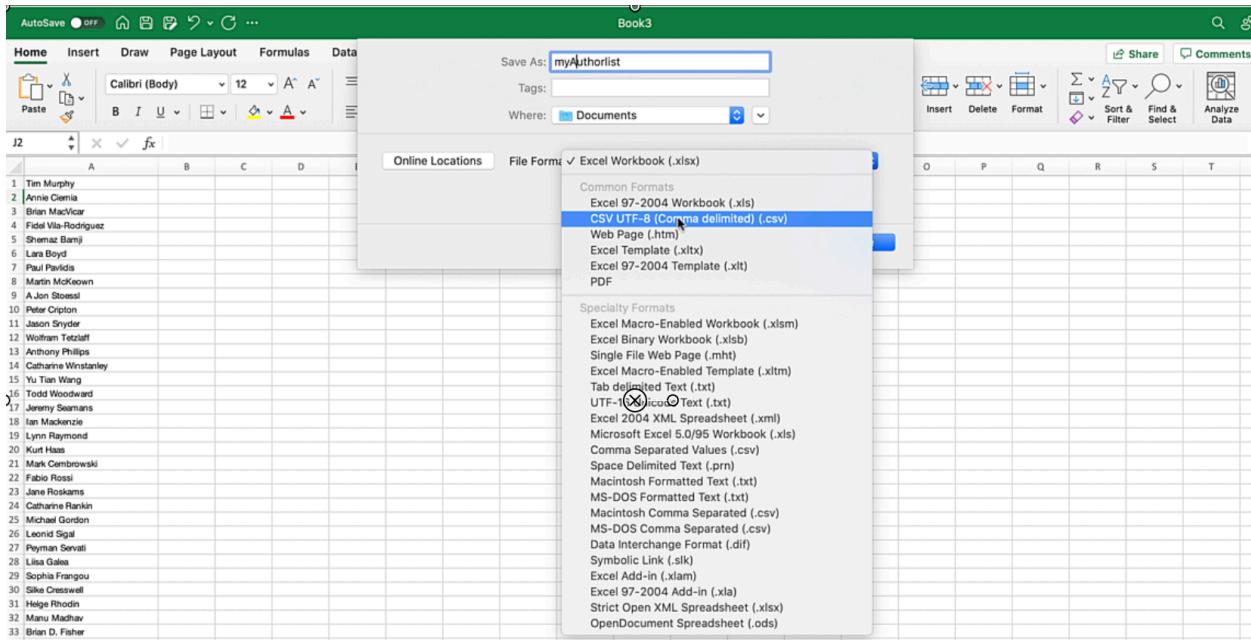
- If you receive an error, type “pip install --user scholarly” and press enter.

```
Requirement already satisfied: sniffio in /opt/conda/lib/python3.9/site-packages (from trio->0.17->selenium->scholarly) (1.2.0)
Requirement already satisfied: sortedcontainers in /opt/conda/lib/python3.9/site-packages (from trio->0.17->selenium->scholarly) (2.4.0)
Collecting wproto==0.14
  Downloading wproto-0.1.0-py3-none-any.whl (24 kB)
Requirement already satisfied: cryptography<=1.3.4 in /opt/conda/lib/python3.9/site-packages (from urllib3[secure]=>1.26->selenium->scholarly) (35.0.1)
Requirement already satisfied: pyOpenSSL>=0.14 in /opt/conda/lib/python3.9/site-packages (from urllib3[secure]=>1.26->selenium->scholarly) (21.0.0)
Requirement already satisfied: pytz>=2015.7 in /opt/conda/lib/python3.9/site-packages (from babel=>1.3->sphinx>=1.6->sphinx-rtd-theme->scholarly) (2021.3)
Requirement already satisfied: cffi>=1.12 in /opt/conda/lib/python3.9/site-packages (from cryptography=1.3.4->urllib3[secure]=>1.26->selenium->scholarly) (1.14.6)
Requirement already satisfied: MarkupSafe>=2.0 in /opt/conda/lib/python3.9/site-packages (from Jinja2>=2.3->sphinx>=1.6->sphinx-rtd-theme->scholarly) (2.0.1)
Collecting h1<1,>=0.9.0
  Downloading h1-0.12.0-py3-none-any.whl (54 kB)
    54 kB / 404 kB/s
Requirement already satisfied: pycares in /opt/conda/lib/python3.9/site-packages (from ffi>=1.12->cryptography>=1.3.4->urllib3[secure]=>1.26->selenium->scholarly) (2.20)
Building wheels for collected packages: bibtexparser, fake-useragent, free-proxy, future
  Building wheel for bibtexparser (setup.py) ... done
    Created wheel for bibtexparser: filename=bibtexparser-1.2.0-py3-none-any.whl size=36711 sha256=4597068be094480a9ca0dal4f7088583e750a8d5b5842f75a8ed9bcce733710
    Stored in directory: /home/jupyter/.cache/pip/wheels/8d/7a/34/56413b9ab410bb24cf1403d91657a97d4c0e938773b8ee86d
  Building wheel for fake-useragent (setup.py) ... done
    Created wheel for fake-useragent: filename=fake_useragent-0.1.11-py3-none-any.whl size=13502 sha256=7074b59bf9850d483c67a9c9d70c809c43865832cd45da9ed8a13ecb307521d
    Stored in directory: /home/jupyter/.cache/pip/wheels/e7/76/7dd4644ad06526ab0e1b4fa2e802fadbb015771b7dcfc9d2
  Building wheel for free-proxy (setup.py) ... done
    Created wheel for free-proxy: filename=free_proxy-1.0.4-py3-none-any.whl size=4284 sha256=0fc8f0c98427c16ccf79d4822139a3733d5b434f0af3937f40d7b65bd5472c8
    Stored in directory: /home/jupyter/.cache/pip/wheels/d8/f8/60/7429382e082238a641ff51924a8d077ccc82778115124d41bd
  Building wheel for future (setup.py) ... done
    Created wheel for future: filename=future-0.18.2-py3-none-any.whl size=491070 sha256=ccecc512cf8d8d617c79f2dd731164ee89c84826701f6f8ad7450d975f1e3b630
    Stored in directory: /home/jupyter/.cache/pip/wheels/2f/a0/d3/4030d9f80e6b3be787f19fc911b8e7aa462986a40able4bb94
Successfully built bibtexparser fake-useragent free-proxy future
Installing collected packages: urllib3, outcome, h1, wproto, trio, sphinxcontrib-serializinghtml, sphinxcontrib-qthelp, sphinxcontrib-jsmath, sphinxcontrib-htmlhelp, sphinxcontrib-devhelp, sphinxcontrib-applehelp, imagesize, alabaster, trio-websocket, sphinx, future, sphinx-rtd-theme, selenium, free-proxy, fake-useragent, deprecated, bibtexparser, arrow, scholarly
  Attempting uninstall: urllib3
    Found existing installation: urllib3 1.25.11
      Uninstalling urllib3-1.25.11:
ERROR: Could not install packages due to an OSError: [Errno 13] Permission denied: 'INSTALLER'
Consider using the ``--user`` option or check the permissions.

jupyter@3b2948dcc578:~$ pip install --user scholarly
```

**5. Create a CSV file with a list of author names and/or Scholar IDs in a single column.
Upload this to your project directory.**

- Save the author list as a CSV file



- Upload the author list CSV to the project directory

The screenshot shows a JupyterLab interface with a sidebar on the left displaying a file tree. A red arrow points to the 'Upload Files' button in the sidebar. A red box highlights the text 'Click here to upload files' above the upload button. The main content area shows the 'ScholarScraper' project documentation, which includes the UBC logo, the text 'THE UNIVERSITY OF BRITISH COLUMBIA Dynamic Brain Circuits in Health & Disease Research Excellence Cluster', and sections for 'Introduction' and 'Installation and Setup'.

6. Make some modifications in the ScholarScraper notebook.

- Modify author_list_csv to match the author list CSV filename which you just uploaded.

The screenshot shows a Jupyter Notebook interface with the following details:

- Title Bar:** scholar_metrics - JupyterLab
- URL:** pims.syzygy.ca/jupyter/user/nicoleacheung/lab/tree/scholar_metrics_scrapers/ScholarScraper.ipynb
- File Menu:** File, Edit, View, Run, Kernel, Git, Tabs, Settings, Help
- Launcher:** Shows a list of files in the current directory:
 - Name
 - Last Modified
 - Images (7 minutes ago)
 - dbc_faculty_groups.... (7 minutes ago)
 - GroupedCollabs.ipynb... (7 minutes ago)
 - myAuthorlist.csv (seconds ago) **(highlighted with a red box)**
 - README.md (7 minutes ago)
 - ScholarCollabs.ipynb (7 minutes ago)
 - ScholarScraper.ipynb (7 minutes ago)
- Code Cell [12]:**

```
#If you receive an error running this cell for the first time, try running it again.  
from scholarly import scholarly  
import csv  
import warnings  
import pandas as pd  
import numpy as np  
import matplotlib  
from matplotlib import pyplot as plt  
from matplotlib import cm as CM
```
- Text Cell [13]:**

2. Modify the names of the input and output files. The name of the input file should match the name of the author list CSV file. If you followed the setup instructions, the CSV file should now be in the same directory as this notebook file. The output file does not have to exist yet (it will be created).

```
author_list_csv = 'myAuthorlist.csv'  
output_data_csv = 'ss_output_data.csv'
```
- Text Cell [6]:**

3. Load in the author names from the CSV file.

```
author_names = []  
with open(author_list_csv, encoding ="utf-8-sig") as csv_file:  
    csv_reader = csv.reader(csv_file, delimiter=',')  
    for row in csv_reader:  
        if (len(row) == 1):
```
- Text Cell [7]:**

4. Modify the affiliations list with institutions which the researchers are affiliated with.

```
affiliations = ['University of British Columbia', 'UBC', 'Djavad Mowafaghian']
```

7. Run the ScholarScraper notebook code cells 1-7.

- You can run a cell by clicking anywhere inside the cell and either clicking the play button or pressing shift + enter.

The screenshot shows a Jupyter Notebook interface with a sidebar on the left displaying a file tree. The main area shows a code cell numbered [12] containing Python code for initializing the scholarly package and importing CSV. A red arrow points to the play button in the toolbar above the code cell, and a red callout bubble says 'Click the play button to run the cell'.

```
[12]: #If you receive an error running this cell for the first time, try running it again.
from scholarly import scholarly
import csv
import warnings
import pandas as pd
import numpy as np
import matplotlib
from matplotlib import pyplot as plt
from matplotlib import cm as CM
```

- Cell number 5 (which retrieves the data for each author from Google Scholar) may take several minutes.
- After running cell number 7, you should find a CSV saved to your directory with the data for each author. You can click it to view it or right-click to download.

The screenshot shows a Jupyter Notebook interface with a sidebar on the left displaying a file tree. The main area shows a code cell numbered [8] containing Python code for creating a dictionary of coauthor counts. A large red arrow points to the 'ss_output_data.csv' file in the file tree on the left, and a red callout bubble says 'Here it is!'. The cell output shows the command '7. Write rows to output CSV file'.

```
[8]: # Create dictionary with author names as keys and dictionary (coauthor name as key, number of collaborations as value)
# as value
collabs_dict={}
for key in pub_authors:
    for author in pub_authors[key]:
        for coauthor in pub_authors[key]:
            if coauthor is not author:
                if author not in collabs_dict.keys() or coauthor not in collabs_dict[author].keys():
                    collabs_dict.setdefault(author,{})[coauthor]=1
                else:
                    collabs_dict[author][coauthor]+=1
```

8. Check the last column of the table for warnings and if necessary, modify your author list CSV.

- Check the last column for warnings - the row for Dr. Tim Murphy has been flagged with a warning. The author data retrieved contains an affiliation that does not match one from the affiliations list.

The screenshot shows two Jupyter Notebook environments. The top environment displays a table with columns: h-index 5 years, i10-index, i10-index 5 years, Publications, Coauthors, Affiliation, and Warning. One row in the 'Warning' column contains the text 'Affiliation does not match!'. The bottom environment shows a similar table where the 'Name' column for the first row is highlighted in blue, indicating it's selected. A large red arrow points from the text 'This data is for Timothy F. Murphy.' to this highlighted name.

	h-index 5 years	i10-index	i10-index 5 years	Publications	Coauthors	Affiliation	Warning
1	34	215	113	['Infection in the patho...		SUNY Disti...	
2	17	22	21	['Examining object loc...		University o...	
3	42	133	82	['Glial and neuronal c...	{'Anthony Phillips': 1, '...	Professor, ...	
4	25	62	58	['Effectiveness of the...	{'Sophia Frangou': 4, '...	Assistant Pr...	
5	25	37	36	['The p75 neurotrophi...	{'Brian MacVicar': 1, '...	Professor, ...	
6	47	143	133	['Observation of amou...	{'Todd S. Woodward': 1,...	Professor, ...	
7	47	136	119	['Coexpression analys...	{'Shernaz Bamji': 2, 'K...	Professor of...	
8	32	123	81	['Removing electron...	{'Lara Boyd': 2, 'A Jon...	Professor of...	
9	45	225	147	['Mutations in LRRK2 ...	{'Lara Boyd': 2, 'Marti...	University o...	
10	30	105	87	['The impact of trans...	{'Wolfram Tetzlaff': 13}	Professor of...	

	Name						
1	Timothy F. Murphy						
2	Annie Vogel Ciernia						
3	Brian MacVicar						
4	Fidel Vila-Rodríguez	VymFpE...	225	3575	2914	27	
5	Shernaz Bamji	C832lNEAAAAJ	55	4534	1646	30	
6	Lara Boyd	yutn3mwAAAAJ	320	11385	7860	58	
7	Paul Pavlidis	ataejQAAAAJ	244	15463	6537	67	
8	Martin J. McKeown	nTYWydBAAAJ	346	14100	5530	42	
9	A Jon Stoezl OR jon ...	KnhWlpwAAAAJ	452	26874	9672	73	
10	Peter Cripton	hm6JTkAAAAJ	287	6316	3554	41	

- Find the author's Google Scholar profile and make sure you have their correct information. If you don't, you will need to modify your author list. You can either correct their name or replace their name with their Google Scholar ID which can be found in their profile URL (see the red box below).
 - In our case, we attempted to retrieve data for "Tim Murphy", but the retrieval resulted in data from another author named "Timothy F. Murphy". We either need to change the name in the author list to "Tim H. Murphy" or replace it with the Scholar ID (qJjM8hkAAAAJ).

The screenshot shows a Google Scholar search results page. A red box highlights the URL 'scholar.google.ca/citation?user=qJjM8hkAAAAJ&hl=en'. A large red arrow points to the profile page for 'Tim H. Murphy' at the University of British Columbia. The profile page includes a photo, the name 'Tim H. Murphy', the affiliation 'University of British Columbia', and a list of publications. A red callout bubble states: 'His profile name is "Tim H. Murphy"'.

- Modify the author list. In this case, we changed “Tim Murphy” to “Tim H. Murphy”

A screenshot of Microsoft Excel showing a list of authors in column A. The first row is selected, and the formula bar displays "Tim H. Murphy". The list includes names like Tim H. Murphy, Anna Clemia, Brian MacVicar, Fidel Vila-Rodriguez, Shemraz Bamji, Lara Boyd, Paul Pavlidis, Martin McKeown, and others. A note at the top of the sheet says "Possible Data Loss: Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format."

- Right click to delete the old author list

A screenshot of a Jupyter Notebook interface. On the left, there is a file tree showing various files and folders. A context menu is open over the 'myAuthorlist.csv' file, with the 'Delete' option highlighted. On the right, there is a table titled 'ss_output_data.csv' with columns: Name, Scholar ID, Document Count, Cited by, Cited by 5 years, h-index, and h-index 5 years. The table contains several rows of data.

- Upload the corrected author list

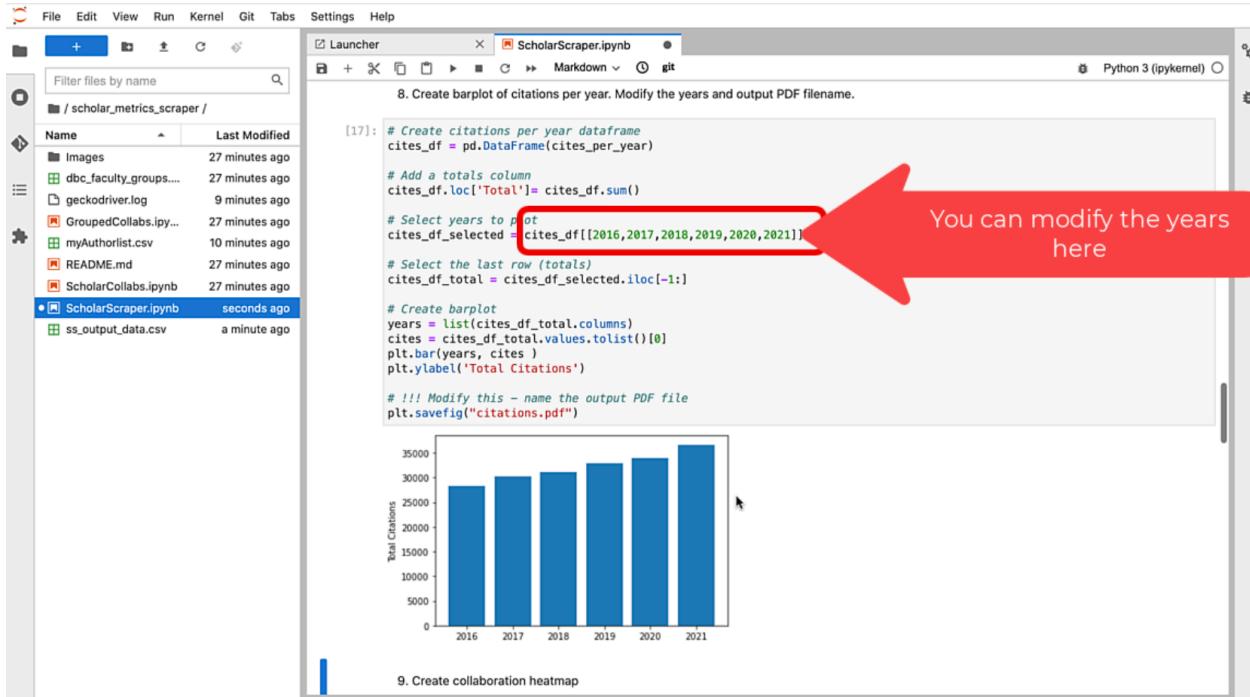
A screenshot of a JupyterLab interface on a UBC server. A red arrow points to the 'Upload Files' button in the toolbar. The main area shows a list of files in the file tree on the left and the UBC logo and name on the right. A large red callout bubble says "Click here to upload files".

- Run code cells 1-7 again
- Check the warnings column again to ensure the author's data has been corrected.

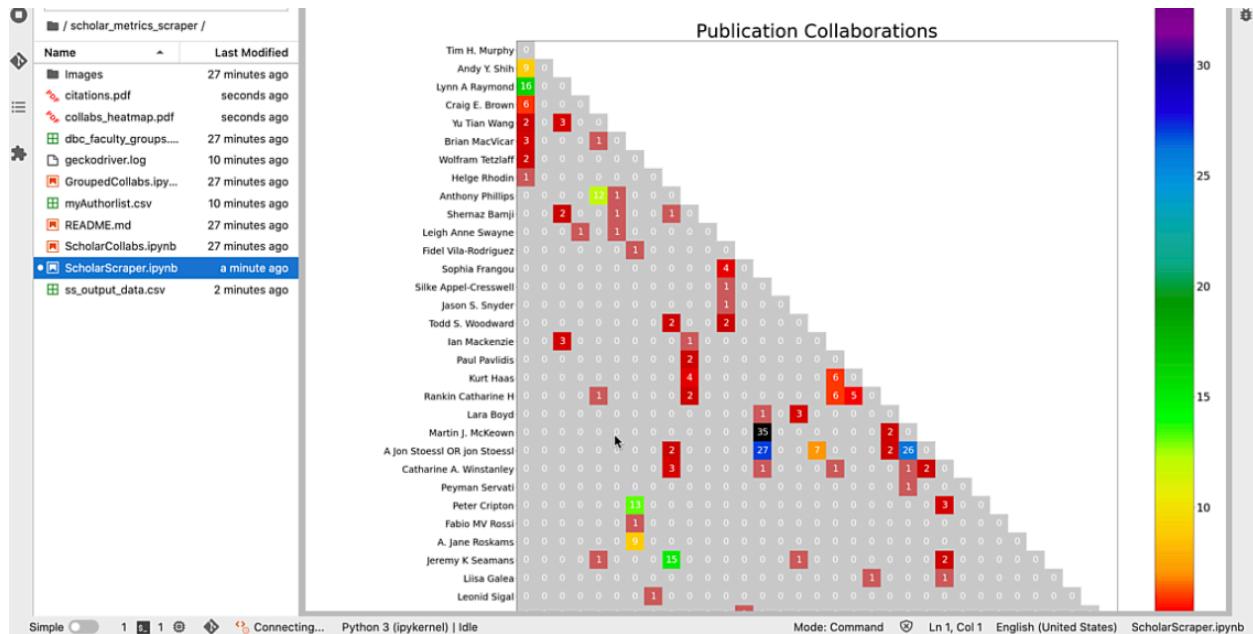
A screenshot of a Jupyter Notebook interface showing a table titled 'ss_output_data.csv'. The table has columns: h-index 5 years, i10-index, and Warning. The 'Warning' column contains several entries, including one that was previously present but is now gone. A red arrow points to the 'Warning' column, and a red callout bubble says "The warning is gone!".

9. Run the ScholarScraper notebook code cells 8 and 9.

- Code cell 8 creates a bar chart of total citations/year for the group. The range of years can be modified as shown below.



- Code cell 9 creates a collaboration heatmap, visualizing the number of collaborations between two coauthors



- Both diagrams are saved to the project directory as PDFs for viewing and downloading.

10. Open the ScholarCollabs notebook. Make some minor modifications to the notebook before running.

- Make sure the output data name matches the CSV filename.

```

library(circlize)
library(RColorBrewer)

Skipping install of 'patchwork' from a github remote, the SHA1 (79223d30) has not changed since last install.
Use 'force = TRUE' to force installation

Skipping install of 'circlize' from a github remote, the SHA1 (14116da5) has not changed since last install.
Use 'force = TRUE' to force installation

Installing package into '/home/jupyter/R/x86_64-pc-linux-gnu-library/4.0'
(as 'lib' is unspecified)

2. Define the name of the data file (CSV created by Scholar Scraper)

[2]: # !!! Modify this - Set the name of the CSV file that was created as output form the ScholarScraper notebook
      ss_output_file = "ss_output_data.csv"

```

3. Define the title, colors, and whether you want to create weighted or non-weighted diagram.

[View color palettes here](#)

```

[3]: # !!! Modify the diagram title
      title = "Dynamic Brain Circuits Collaborations"

# !!! Modify the colour palette.
c_palette <- brewer.pal(12,"Paired")

```

- Modify the title, colour palette, and set weighted to TRUE if you want a weighted diagram (which weights links by the number of collaborations) or FALSE if you want an unweighted diagram.

```

myAuthorlist.csv           13 minutes ago
README.md                  30 minutes ago
* ScholarCollabs.ipynb    30 minutes ago
ScholarScraper.ipynb       2 minutes ago
ss_output_data.csv         4 minutes ago

3. Define the title, colors, and whether you want to create weighted or non-weighted diagram.

View color palettes here

[3]: # !!! Modify the diagram title
      title = "Publication Coauthors"

# !!! Modify the colour palette.
c_palette <- brewer.pal(12,"Paired")

# !!! Modify this - Set to TRUE if you want a weighted diagram or FALSE if you want a non-weighted diagram.
      weighted = TRUE

```

- You can change the PDF filename (where the image will be saved) here.

```

dbc_faculty_groups....    32 minutes ago
geckodriver.log            15 minutes ago
GroupedCollabs.ipynb       32 minutes ago
myAuthorlist.csv            15 minutes ago
README.md                  32 minutes ago
* ScholarCollabs.ipynb    a minute ago
ScholarScraper.ipynb        4 minutes ago
ss_output_data.csv          7 minutes ago

8. Create the chord diagram. Modify the name of the output PDF file. You can make additional optional modifications as well (read the comments below).

[11]: # Modify this!!!
      pdf("chord_diagram_weighted.pdf")

# set up the parameters
circos.clear()
circos.par(start.degree = 90,gap.degree = 1,
          track.margin = c(-0.1, 0.1),
          points.overflow.warning = FALSE, canvas.xlim = c(-1.3,1.3),
          canvas.ylim = c(-1.3,1.3))
par(mar = c(0,0,2,0),xpd = TRUE, cex.main = 1.5)

# create the chord diagram
chordDiagram(links,
             grid.col = color.ind,
             transparency = 0.25,
             diffHeight = -0.04,
             annotationTrack = "arid".

```

11. Run the ScholarCollabs notebook code cells.

- After running the last code cell (8), the diagram will be saved in your project directory.

The screenshot shows a Jupyter Notebook interface with two tabs: 'Launcher' and 'ScholarCollabs.ipynb'. The 'ScholarCollabs.ipynb' tab is active, displaying the following code:

```
# Add graduation on axis
cos.axis(
    h = "top",
)

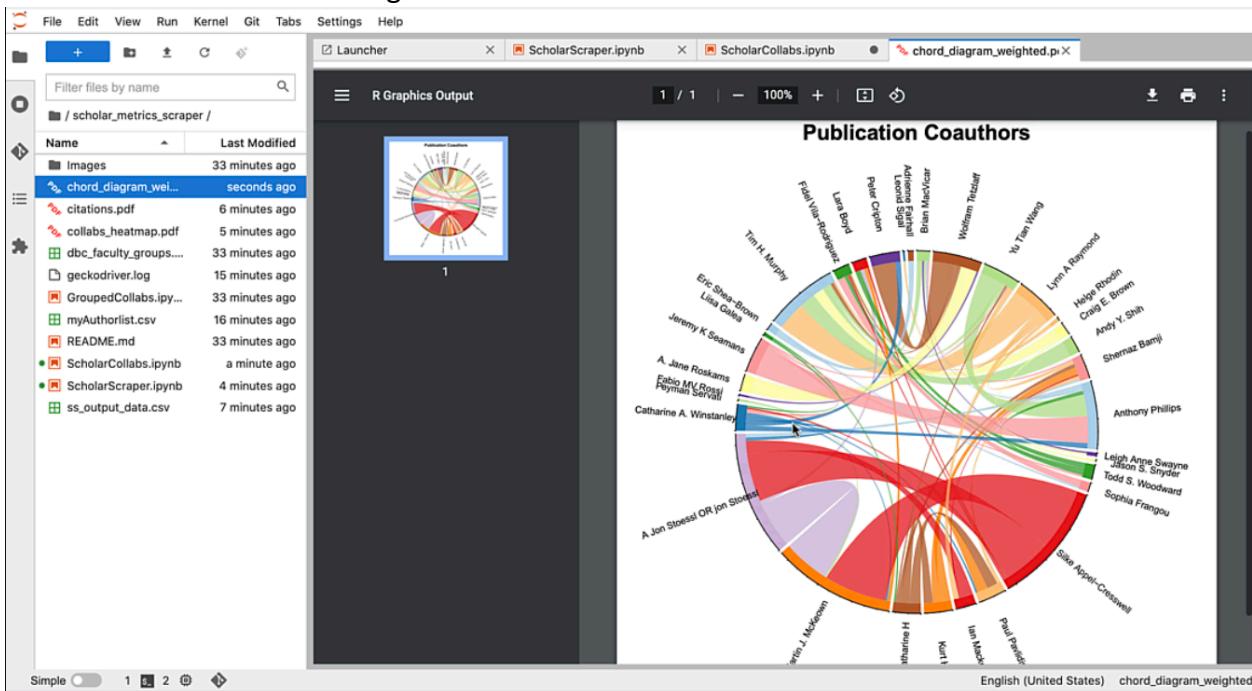
# Add the title (user can modify the title in step 3)
title(title,outer=FALSE)

dev.off()

png: 2
```

A red arrow points from the text 'The diagram is saved as a PDF here' to the 'png: 2' line in the code.

- Click it to view the diagram.



12. Open the GroupedCollabs notebook. You will need an additional CSV file with each author's group to run this notebook.

- Create a CSV file with author names in the first column (red box), group names in the first row (blue box), and author names in their respective rows and under their group's column (green box). Author names will need to match the names in the output data CSV file (and therefore, match their Google Scholar profile names).

	Faculty of Medicine	Faculty of Applied Science	Faculty of Science	Faculty of Arts	Not UBC
2	Tim H. Murphy		Annie Vogel Ciernia		
3	Annie Vogel Ciernia				
4	Brian MacVicar				
5	Fidel Vila-Rodriguez				
6	Shernaz Bamji				
7	Lara Boyd				
8	Paul Pavlidis				
9	Martin J. McKeown				
10	A Jon Stoessl OR jon Stoessl				
11	Peter Cripton				
12	Jason S. Snyder				
13	Vesna Sossi				
14	Wolfram Tetzlaff				
15	Anthony Phillips				
16	Catharine A. Winstanley				
17	Yu Tian Wang				
18	Todd S. Woodward				
19	Jeremy K Seamans				
20	Ian Mackenzie				
21	Lynn A Raymond				
22	Kurt Haas				
23	Mark S. Cembrowski				
24	Fabio MV Rossi				
25	A. Jane Roskams				
26	Rankin Catharine H				
27	Michael Gordon				
28	Leonid Sigal				
29	Peyman Servati				
30	Liisa Galea				
31	Sophia Frangou				
32	Silke Appel-Cresswell				
33	Helge Rhodin				

- Save as CSV and upload to the project directory.

The screenshot shows a JupyterLab interface with a file upload dialog. A red arrow points to the 'Upload Files' button in the dialog. A red callout bubble with the text 'Click here to upload files' is positioned over the dialog. The sidebar on the left shows a file tree with the following contents:

- /scholar_metrics_scrapers/
 - Name
 - Last Modified
 - Images
 - dbc_faculty_groups....
 - GroupedCollabs.ipynb
 - README.md
 - ScholarCollabs.ipynb
 - ScholarScrapers.ipynb

The main area displays the University of British Columbia logo and the text 'THE UNIVERSITY OF BRITISH COLUMBIA' and 'Dynamic Brain Circuits in Health & Disease Research Excellence Cluster'.

13. Make some modifications to the GroupedCollabs notebook.

- Modify the group_file name to match the CSV filename.

```

2. Define the name of the data file (CSV created by Scholar Scraper), investigator names CSV file, and group CSV file

• [52]: # !!! Modify this to match the name of the CSV file that was created as output from the ScholarScraper notebook
ss_output_file = "ss_output_data.csv"
    ... modify this to match the name of your groupings CSV file. See instructions above.
group_file = "myGroups.csv"

```

- Again, make sure the ss_output_data name matches the CSV file.

```

2. Define the name of the data file (CSV created by Scholar Scraper), investigator names CSV file, and group CSV file

• [52]: # !!! Modify this to match the name of the CSV file that was created as output from the ScholarScraper notebook
ss_output_file = "ss_output_data.csv"
    ... modify this to match the name of your groupings CSV file. See instructions above.
group_file = "myGroups.csv"

3. Define the title, colors, and whether you want to create weighted or non-weighted diagram.

View color options here.

```

- Modify the title, colours (one for each group), and group names. Again, set weighted to TRUE or FALSE depending on whether you want a weighted diagram.

```

• [53]: # !!! Modify the diagram title
title = "Coauthors Grouped by Faculty"

# !!! Modify the colour palette. Make sure there are the same number of colours as groups.
c_palette = c("red","orange","green","cyan","magenta")

# !!! Modify the group names. These groups will be paired with the colours in the c_palette, in the same order.
group_names = c("Medicine",
               "Applied Science",
               "Science",
               "Arts",
               "Cascadia")

# !!! Modify this - Set to TRUE if you want a weighted diagram or FALSE if you want a non-weighted diagram.
weighted = TRUE

```

- Name the PDF where the diagram will be saved to.

```

8. Create the chord diagram. Modify the name of the output PDF file. You can make additional optional modifications as well (read the comments below).

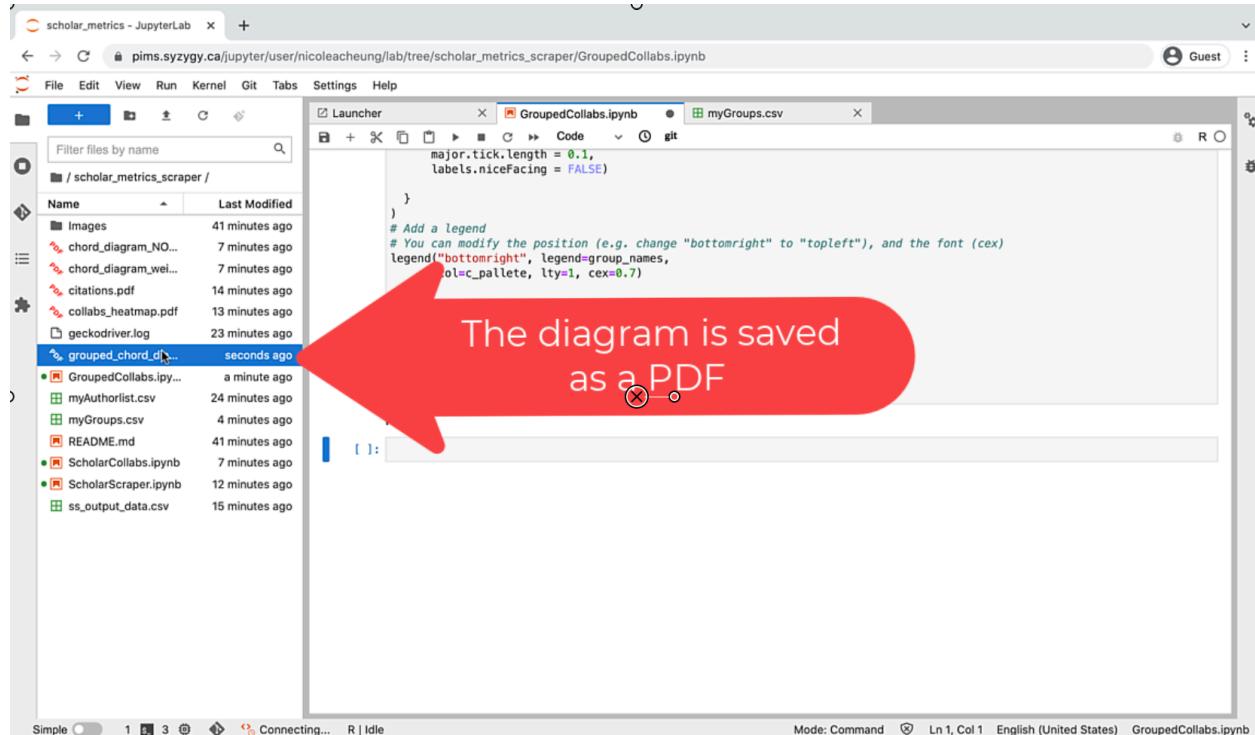
• [58]: # !!! Modify this
pdf("grouped_chord_diagram_weighted.pdf")

# set up the parameters
circos.clear()
circos.par(start.degree = 90,gap.degree = 1,
          track.margin = c(-0.1, 0.1))

```

14. Run the GroupedCollabs notebook code cells.

- After running the last code cell (8), the diagram will be saved to your project directory.



The diagram is saved as a PDF

```
major.tick.length = 0.1,  
labels.niceFacing = FALSE)  
}  
# Add a legend  
# You can modify the position (e.g. change "bottomright" to "topleft"), and the font (cex)  
legend("bottomright", legend=group_names,  
col=c_pallete, lty=1, cex=0.7)
```

- Click it to view the diagram.

