

Data and Python code for "Næss, Marius W., Fisktjønmo, G. L. H. and Bårdsen, B.-J. Social networks and ecological collapse: A case study of the Tana salmon fishery in Norway."

Table of Content

- **General information**
- **Python version used**
- **Known issues and solutions**
- **Step-by-step guide for creating a virtual environment**
- **Install**
 1. **Create virtual environment**
 2. **Activate virtual environment**
 3. **Install general packages**
 4. **Run scripts**
- **Folder overview**
 - Data
 - Figures
 - Code
 - Supplement
- **Running without graphviz**

General information

Note, these instructions are for installing and running the analyses on Windows. It is not using Anaconda but rather Python as stand-alone. Has been tested by a clean install of Python.

Everything posted is posted as stand-alone Jupyter Notebook (<https://jupyter.org/>).

Please note that development has been done using VS Code (<https://code.visualstudio.com/>) with Jupyter notebook extensions installed.

Please note that Python needs access to Path for it to work (see e.g., <https://www.computerhope.com/issues/ch000549.htm>).

Keep in mind that the notebook runs a test to check whether you have the correct version of python (3.11.9) and networkx (2.8.8) installed. If not, you will get an error message.

Python version used:

Python 3.11.9

Known issues and solutions

Installing Python might be hampered by install privileges and organisational policies. Additionally, running Jupyter Notebook might be hampered by organisational firewall policies since it is a server based environment for running Python code.

Since this paper is a social network analysis, `graphviz` and `pygraphviz` is used for layout purposes. On window, installing `graphviz` and `pygraphviz` is non-trivial.

For installing `graphviz`, we recommend installing chocolatey (<https://chocolatey.org/>) and running `choco install graphviz` from command prompt. The version used here is `graphviz 6.0.2`. Alternatively, you can use `winget` (but remember to add `Graphviz` to path which `winget` seems not to do).

For `pygraphviz` see <https://pygraphviz.github.io/documentation/stable/install.html> for details. However, `pygraphviz` will be installed using the `requirements.txt` file according to the instructions below as long as you have Visual C/C++ (<https://visualstudio.microsoft.com/visual-cpp-build-tools/>).

Now, following this approach installs `graphviz` and `pygraphviz` globally on your computer. For some this might not be ideal and kind of defeats the purpose of using `venv` for creating virtual environments. If you do not wish to use `graphviz` and `pygraphviz` a possible workaround is provided in the section **Running without graphviz** at the end of this file.

If having another python version installed you can install the required version using `winget` (<https://learn.microsoft.com/en-us/windows/package-manager/winget/>) from CMD: `winget install python.pyhton.3.11`. If you have problems (and being windows, you probably have), open PowerShell with admin privileges and type `winget install -e --id Python.Python.3.11 --scope machine`.

Install instructions below is based on this approach.

Step-by-step guide for creating a virtual environment

Suggest creating a new virtual environment and install all from there (see below).

This environment consist of all packages used for

1. Cleaning and selecting data
2. Data wrangling
3. Plotting and visualisation
4. Analyses

Due to potential issues with personal data, scripts and datasets used for the first two steps are not publicly available.

Install

All of this is done from the Command Prompt or Powershell.

1. Create virtual environment

`cd` to folder where you keep virtual environments and type (requires python 3.11 to be installed):

```
py -3.11 -m venv tana-networks-env.
```

2. Activate virtual environment

Type `tana-networks-env\Scripts\activate`

3. Install general packages

`cd` to root dir of project (where you have all the files and code located) and type:

```
pip install -r requirements.txt
```

To see all packages, package versions and dependencies, you can look at `requirements_full.txt` (generated by `python -m pip freeze > requirements_full.txt` in cmd). This can also be used as install file, but we do not recommend it.

4. Run scripts

`cd` to folder with `.ipynb` files and type:

```
jupyter notebook
```

OR open individual notebooks in VS Code.

Folder Overview

- **Data\Final Data**

Folder containing anonymised datasets and networks used in manuscript.

Networks:

- `allocation_network.gml`
- `help_network.gml`
- `information_network.gml`
- `management_network.gml`

Datasets:

- `Reason_for_gift.xlsx`
- `Reason_for_keeping.xlsx`

See `metadata.md` or `metadata.pdf` file in folder for details.

- **Figures**

Folder containing all the figures in the manuscript. Keep in mind that all figures (1-6) will be overwritten when running the notebook. Additionally, the network figures (Figure 3-6) might look different between

each run due to graphviz positioning: layout engines like `neato` or `fdp` use randomised initial placements. However, the overall network structure is the same.

- **Code**

Folder containing all the code: You can just open notebook to see the code without installing, but then you need an editor that can show notebooks, like VS Code (<https://code.visualstudio.com/>).

Files:

- `Analyses_and_plots_FINAL.ipynb`

This notebooks creates all plots and results.

See `metadata.md` or `metadata.pdf` file in `Final data` folder for details about the datasets and variables.

- **Config**

Folder containing configuration python script for setting the root directory and color scheme for network figures. File: `definitions.py`

- **Support**

Folder containing supporting functions. Files:

- `helpers.py`.
- `supplement.py`.

- **Supplement**

Folder containing all supplementary figures used in S1: Supplementary material.

Running without `graphviz`

To run without `graphviz` and `pygraphviz` you need to change the `requirements.txt` file by removing the line:

```
--config-settings="--global-option=build_ext" --config-settings="--global-option=-
IC:\Program Files\Graphviz\include" --config-settings="--global-option=-LC:\Program
Files\Graphviz\lib" pygraphviz
```

Then, for each notebook depending on `graphviz` you have to delete/comment out the following import code else you get an error:

- `from networkx.drawing.nx_pydot import graphviz_layout`

Additionally, you need to change any network figures using `pygraphviz` for layout purposes. So basically, you have to look through all the network plots with the code `pos = graphviz_layout(G, prog='fdp')`, where `G` is a shorthand for the network (keep in mind that `prog` varies) and replace it with a call to general `networkx` positioning. For example, replace it with `pos = nx.spring_layout(G, gravity=0.4, seed=42)`, which uses a spring layout. `gravity` and `seed` can be changed, suggest reading the documentation

https://networkx.org/documentation/stable/reference/generated/networkx.drawing.layout.spring_layout.html#networkx.drawing.layout.spring_layout.