□□□□

 RC bootcamp (Reservoir computing; RC)RC□□

Jupyter Notebook

□□□□□□□□□□□□□□□□□□□□□□□Python□□□□ □□□□□□□□□Jupyter Notebook□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□ □□□□□□

□□1. Google Colaboratory + Google Drive (□□□: □□)

Google Colaboratory (Colab) Jupyter Notebookを□□□□□□□□□□□□□□□□□□□ □□□□□□□□ □□□□□□□□□□□□□ (`numpy` や `matplotlib` □) □□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□ □□□□□□□□□□□ □□□□□ □□ https://rc-bootcamp.github.io/ □Colab□□□□□□□□□□□□□□□□□□□□ **Google Drive**□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□ □□Reservoir computing□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

0. Google□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
1. RC bootcampの□□ (zip□□□□) □□□□□□□□□□□□□□□□□□□□□□□
2. Google drive□□
3. `.ipynb` □□□□□□□□□□□□□notebook□□□□□□□□□□□□□□□
4. □□□□□□□□□ `Shift+Enter` □□□□(□□□□□□□□□□□□□) □□□□□□□□□□
5. `drive.mount("/content/gdrive")` □□□□□□□□□□□□□□□ `if False:` □□□□ `if True:` □□□□□□□□□□□□□□□□□□Google Drive□□□□□□□□□□□□□ (`%cd /content/gdrive/My Drive/... □□□`)□
6. □□□Google Drive□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□5.□□□□□□Google Drive□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□2. uv + VSCode (□□□: □□)

□□□□□□□□□□□□ `uv` □□□□□□□□□□□□□VSCode□□□ □□□□□□VSCode□□□ (□□□□□□□□□□□□□□□□□□□)□

Step 1. uv□□□□□□□

`uv` □python□□□□□□□□□□□□□□□□ □□□ (conda□□□□□□□□uv□□□□□□□□□□□□□□)□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□

Linux□MacOS□□□

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
curl -LsSf https://astral.sh/uv/install.sh | sh
```
□□□□□□□□□□□□□□□□□

```
uv help
```
□□□□□□□□□□□□□□□□□□□□□□□□□□□□

Windows□□□

PowerShell□□□□ (□□□□□□□□□□□□□□Windows PowerShell□□□□□□Windows PowerShell □□□□□□□□□□□□□□□□□□) □□□□□□□□□□□□□□□□□□□□□□□

```
powershell -ExecutionPolicy ByPass -c "irm
https://astral.sh/uv/install.ps1 | iex"
```
□□□□□□□□□□□□□□□

```
uv help
```
□□□□□□□□□□□□□□□□□□□□□□□□□□□

Step 2. VSCode□□□

[Visual Studio Code (VSCode)](…)□Microsoft□□□□□□□□□□□□□□□□□□□□□□IDE□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□ `code` □□ (Windows□□□□□PowerShell□□□□□□□□□□)□

```
code PATH_TO_DIR
```
VSCode□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□ `Extensions` □□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□ `.vscode/extensions.json` □□□□□□□□□□□□□□ VSCode□□□

Step 3. □□□□□□□□

□□□□□□VSCode□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ `Ctrl+Shift+@` □□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
uv sync
```
`uv.lock` □□□□□□□□□□□□□□□□□□□□□□□□□□□ `.venv` □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□ □□□□□□□□□□Python□□□□□□Python□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□

 Ctrl+Shft+p Python: Select Interpreter (.venv) .ipynb Select Kernel Python Kernel

VSCode Linux、MacOS

```
source .venv/bin/activate
```
Windows

```
.venv\Scripts\activate
```

Step 4. VSCode ()

 Ruff 的Python的Linting VSCode的 .vscode/settings.json

```json
{
    "notebook.output.wordWrap": true,
    "[python]": {
        "editor.formatOnSave": true,
        "editor.codeActionsOnSave": {
            "source.fixAll": "explicit",
            "source.organizeImports": "explicit"
        },
        "editor.defaultFormatter": "charliermarsh.ruff"
    },
    "notebook.formatOnSave.enabled": true,
    "notebook.codeActionsOnSave": {
        "notebook.source.fixAll": "explicit",
        "notebook.source.organizeImports": "explicit",
    },
    "jupyter.notebookFileRoot": "${fileDirname}",
}
```