ISLDScd24学习材料

1. Where to learn new things?

1.1 Wikipedia

- wikipedia.org
  - *Hands-On:*
    - Try search Wiki of Chatgpt
  - *Notes:*
    - use English for more info and accuracy.

1.2 Youtube

- youtube.com

1.2 MOOC

- Massive open online course (MOOC): https://en.wikipedia.org/wiki/Massive_open_online_course
- List of MOOC providers: https://en.wikipedia.org/wiki/List_of_MOOC_providers
  - Hands-On:
    - Try to follow and complete a free MOOC course on Coursera, Khan Academy, Stanford Online, MIT OCW orUdacity

1.3 BiliBili

- bilibili.com
- Notes: uploads are mostly in Chinese or have Chinese subtitles

1.4 YouTube

- youtube.com
- Notes: uploads of most latest developments in Ai world are found here
- Tip: due to the info gap between in-land and out-world, you might make money by moving the latest uploads from YT to Bili

1.5 Cheatsheets

- Search on google.com or find on cheat-sheets.org
  - *Hands-On:*
    - Try searching "cheatsheet docker pdf":
      - https://docs.docker.com/get-started/docker_cheatsheet.pdf
    - Find a Ubuntu cheatsheet on cheat-sheets.org:
      - https://www.cheat-sheets.org/saved-copy/ubunturef.pdf
  - *Notes:*
    - Use the English language of google search for more relevant results.
    - Some are available on official sites, such as the docker cheatsheet https://docs.docker.com/get-started/docker_cheatsheet.pdf.
    - Pdf/Html formatted cheatsheet. Pdf is good for printing and quick reference, html is good for cross-referencing.
    - There are novice and advanced versions of cheatsheets, with varied info density. Use accordingly as you progress.

2. Scientific Literature Search Tools

2.1 LetPub

- letpub.com.cn (free)
  - *Hands-On:*
    - Try searching for 期刊名： *Nature*
    - Try searching for 中科院分区：1区，影响因子：>10，结果排序：影响因子，查看期刊
  - *Notes:*
    a. Develop a sense of journal quality by the above excersise

2.2 GoogleScholar

- scholar.google.com (free) - search by researcher or keyword
  - *Hands-On:*
    - Search by Author Name
  - *Notes:*

a. Search and summa

The Chicago Manual of Style, 17th Edition

2.3 ChatGPT

- chatgpt.com (free) - Q&A session
  - *Hands-On:*
    - Search for literature

      Search: give me a list of latest high-quality papers published in 2024 on predicting disease status from genomic and phenomic profiles
    - Summarize a paper
  - Notes:
    - The current free version is GPT-4o.
    - Hallucination: making up citations, be sure to check them manually.
    - Timing: is not timely updated, use Copilot for more current search results.
  - Try other ChatGPT functions: write code, help debug and edit papers.

2.4 Copilot/Bing

- bing.com (free) - search + Q&A session

3. Data Science Development Tools

3.1 Windows (EN)

- Notes
  - **See labx信息环境 for more windows tips**
  - **Also see Windows11/10 统一设置 for installing clean multi-language Windows**

3.2 Docker

- **Install Docker in windows**
  - *Hands-On:*
    - Tutorial Video: https://www.bilibili.com/video/BV11L411g7U1

https://www.bilibili.com/video/BV11L411g7U1/

Docker 1小时快速上手教程，无废话纯干货_哔哩哔哩_bilibili

Docker 1小时快速上手教程，无废话纯干货共计8条视频，包括:Docker 简介和安装、用 Docker 快速安装软件、构建自己的 Docker 镜像等，UP主更多精彩视频，请关注UP账号。

- Tutorial: https://docker.easydoc.net/doc/81170005/cCewZWoN/lTKfePfP

- *Notes:*

- **Use an Ubuntu container**
  - *Hands-On:*
    - Tutorial Video: https://www.bilibili.com/video/BV1y34y197mP

https://www.bilibili.com/video/BV1y34y197mP

02docker安装ubuntu与基本操作介绍_哔哩哔哩_bilibili

-, 视频播放量 8572、弹幕量 2、点赞数 72、投硬币枚数 17、收藏人数 80、转发人数 8, 视频作者 ITKEY, 作者简介 爱技术，爱分享，折腾技术使我快乐!，相关视频:实体机安装Ubuntu，我再也不用win了，vmware安装…

    - Official ubuntu docker images: https://hub.docker.com/_/ubuntu/
    - Tutorial: https://docs.docker.com/engine/install/ubuntu

```
h> docker pull ubuntu                # pull the ubuntu image to local
h> docker image ls                   # check the image is here
h> docker run -d -v /mnt/e/Temp:/mnt/Temp --name mu ubuntu sleep infinity   # run the ubuntu container, mount the E:\Temp dir and and name it mu
h> docker ps -a | grep mu            # view your container is running
h> docker exec -it mu /bin/bash      # interactively exec bash and get a terminal
c# cat /etc/os-release               # check ubuntu version
c# apt-get update && apt-get install python3 # install python3
c# touch /mnt/Temp/test.txt          # make a new file in E:\Temp
h> docker commit mu upy3             # save the container change to a new image upy3
h> docker stop mu                    # stop the container process, can be reused
h> docker rm mu                      # remove the container process
h> docker run -d --name mp3 upy3 sleep infinity
h> docker exec -it mp3 /bin/bash     # interactively exec bash and get a terminal
```

  - *Notes:*

- Install WSL2 before docker is recommended
- Powershell and WSL2 shells share the same docker space
- -v /mnt/e/Temp:/mnt/Temp requires WSL2 and automount on
- grep only works in the WSL2 shell
- Use xxx --help or man xxx to see how to use xxx command

3.3 Linux (Ubuntu)

- **Learn to use Ubuntu**
  - *Hands-On:*
    - Tutorial Video (needs a better one):

      https://www.bilibili.com/video/BV1W4411A7yf

      黑马—Ubuntu教程_哔哩哔哩_bilibili

      黑马—Ubuntu教程共计100条视频，包括:01-课程安排、02-励志公式和python体验、03-什么是操作系统等，UP主更多精彩视频，请关注UP账号。

    - Ubuntu Docs: https://help.ubuntu.com/
    - Ubuntu Cheatsheet: https://www.cheat-sheets.org/saved-copy/ubunturef.pdf
    - Linux Cheatsheet: https://web.archive.org/web/20240119115021/https://www.cheat-sheets.org/project/tldr/command/special-most-used-linux-commands/
    - Advanced Commands, Commandline Tools and Enviroment Variables:

> apt # manage additional tools and commands, https://en.wikipedia.org/wiki/APT_(software)
>
> vim # a neat yet powerful text editor, https://en.wikipedia.org/wiki/Vim_(text_editor)
>
> grep # you can almost find anything, https://en.wikipedia.org/wiki/Grep
>
> awk # efficiently manipulate complex tables, https://en.wikipedia.org/wiki/AWK
>
> sed # efficiently manipulate large texts, https://en.wikipedia.org/wiki/Sed
>
> | # efficiently connects command outputs, https://en.wikipedia.org/wiki/Pipeline_(Unix)
>
> ssh # manage your server from anywhere (scp, sftp, rsync, ...), https://en.wikipedia.org/wiki/SSH
>
> nohup # manage the job process (disown, bg, &, ...), https://en.wikipedia.org/wiki/Job_control_(Unix)

> /usr/bin/python # the python interpreter to run .py scripts
>
> /usr/bin/Rscript # the R interpreter to run .R scripts
>
> /bin/bash # the Bash interpreter to run .sh scripts
>
> history # don't lose anything you've typed, can search and edit with vi
>
> env # environmental variables, all you can set https://en.wikipedia.org/wiki/Environment_variable
>
> wget # elink, curl, call you can view and get from internet https://en.wikipedia.org/wiki/wget

- *Notes:*
  - Powershell and WSL shell share the same docker space
  - Shell commands and commandline tools are also available within vim
  - Stderr and stdout are savable, searchable and manipulatable with &1 and &2

3.4 Bourne-Again SHell (Bash)

- **Learn to use Shell (Bash)**
  - *Hands-On:*
    - Tutorial Video (paid content, needs a better one):

https://www.bilibili.com/video/BV1AT411Y7bq

【伯乐大典】最实用的Bash脚本知识_哔哩哔哩_bilibili

【伯乐大典】最实用的Bash脚本知识共计8条视频，包括:1. 什么是Bash脚本、2. 变量、3. 用户输入等，UP主更多精彩视频，请关注UP账号。

https://www.bilibili.com/cheese/play/ss15269

跟我一起学bash脚本编程_哔哩哔哩_bilibili

首页 番剧 直播 游戏中心 会员购 漫画 赛事 去巴黎 去巴黎 下载客户端 登录 登录后你可以: 免费看高清视频 多端同步播放记录 发表弹幕/评论 热门番剧影视看不停 立即登录 首次使用? 点我注册 大会员 消息 动态 收藏 历史 …

    - Bash Docs: https://www.gnu.org/software/bash/manual/bash.html
    - Bash Cheatsheet: https://www.cheat-sheets.org/saved-copy/bash_ref.pdf
    - Advanced Bash Programming:

> bash # how to talk to Linux https://en.wikipedia.org/wiki/Bash_(Unix_shell)

  - *Notes:*

3.5 Application: convert py2 Python package to py3

- **Learn to convert the py2 elsa package to py3**

```
apt install -y 2to3
cd /mnt/tmp/lec3 && git clone git@github.com:labxscut/elsa.git
2to3 -w setup.py
cd elsa
for f in `ls *.py`; do 2to3 -w $f; done;
python setup.py build
cd build/lib.linux-x86_64-3.10
 (for f in `ls lsa/*.py`; do echo "### $f ###";
PYTHONPATH=$(pwd):$PYTHONPATH python $f; done;) >err.txt 2>&1
 #by now you shall see the errors in err.txt which needs to be fixed
for v3
```

3.6 Application: batch download PDB files

- **Tutorial (todo)**
  - *Hands-On:*
  - *Notes:*

4. Python Coding Tools

4.1 Docker, PyNotebook and Conda

- **Install Python + Conda + PyNotebook in Dockerized Ubuntu**
  - *Hands-On:*
    - Review of basic docker operations

> h> docker run -v /mnt/e/tmp:/mnt/tmp -d --name islds1 islds sleep infinity # create a working container from islds image with a mounted working directory
>
> h> docker run -v E:\tmp:/mnt/tmp -d --name islds1 islds # create a working container from islds image with a mounted working directory in powershell
>
> h> docker exec -it islds1 /bin/bash # open a bash on islds1
>
> h> docker commit islds1 islds # commit modifications in islds-new to islds image
>
> h> docker stop islds1 && docker rm islds1 # kill and remove islds1 (&& is a cmd connector only available with a Linux shell, exec the cmds one-by-one in powershell)
>
> h> docker start islds1 # if interrupted and the container stopped in middle
>
> h> docker attach islds1 # reconnect to the containter

- Add Python 3, Anaconda and Jupyter Notebook to islds interactively; Tutorial: https://docs.anaconda.com/anaconda/install/linux/

```
c# cd /mnt/tmp/lec3                    # use this working direcotry
c# apt-get update --fix-missing        # fix missing packages
c# apt-get -y install python3 python3-dev    # install python3 and headers
c# apt-get -y install jupyter-notebook       # install jupyter notebook
c# curl -L -o anaconda.sh https://repo.anaconda.com/archive/Anaconda3-2024.06-1-Linux-x86_64.sh && /bin/bash anaconda.sh  # download anaconda and install, need manual inputs in middle
```

- Instead, we can install them using a Dockerfile; First, create a Dockerfile.pnb:

```
# Use your existing Docker image as the base
FROM ubuntu:latest
# Set the working directory (optional, depending on your existing setup)
WORKDIR /usr/src/app
# Prepare apt
RUN apt-get update --fix-missing
# Install Jupyter Notebook
RUN apt-get install -y jupyter-notebook
# Install python3
RUN apt-get install -y python3 python3-dev python3-pip
# Expose port 8888 for Jupyter Notebook
EXPOSE 8888
# Define the default command to run when starting the container
CMD ["jupyter", "notebook", "--ip='0.0.0.0'", "--port=8888", "--no-browser", "--allow-root"]
```

- Build the islds-new container image

```
h> docker build -t islds -f Dockerfile.pnb .
```

- A Test Run:

```
h> docker run -p 8888:8888 -v /mnt/e/tmp:/mnt/tmp islds
```

- *Notes:*
  - Conda Cheatsheet: https://docs.conda.io/projects/conda/en/latest/user-guide/cheatsheet.html

- Can build the Dockerfile in a linux station and use it for new build in windows

- **Install Conda and PyNotebook in windows**
  - *Hands-On:*
    - Official Anaconda for windows : https://www.anaconda.com/download
    - Tutorial Video: https://www.bilibili.com/video/BV1jf4y1j7Vi

      https://www.bilibili.com/video/BV1jf4y1j7Vi
      数据科学之铲-5分钟搞定Anaconda和Jupyter notebook的配置_哔哩哔哩_bilibili
      官网:https://www.anaconda.com/products/individual清华镜像:https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/?C=M&O=A, 视频播放量 7541、弹幕量 …
  - *Notes:*
    - We will continue the course with a dockerized pynotebook.

4.2 Python Packaging and Deliverables

- **Learn how to package and deliver a python program**
  - *Hands-On:*
    - Python Packaging Tutorial
      - https://www.bilibili.com/video/BV194411r7a8

        https://www.bilibili.com/video/BV194411r7a8
        Python——包和模块_哔哩哔哩_bilibili
        Python——包和模块共计100条视频，包括:1-Python包和模块-基本概念解释、2-Python包和模块-作用、3-Python包和模块-分类等，UP主更多精彩视频，请关注UP账号。
    - Python Packaging Basics

```
samplepackage/
|
├── samplepackage/
|   ├── __init__.py
|   ├── module1.py
|   └── module2.py
|
├── tests/
|   ├── __init__.py
```

```
    ├── test_module1.py
    └── test_module2.py
    
├── README.md
├── LICENSE
├── setup.py
└── requirements.txt
```

- `samplepackage/` : The directory containing your package.
- `init.py` : An empty file that tells Python that this directory should be treated as a package.
- `module1.py` , `module2.py` : Python modules with your package's s
- `tests/` : Directory containing tests for your package.
- `README.md` : A markdown file describing the project.
- `LICENSE` : The license file.
- `setup.py` : The build script for setuptools.
- `requirements.txt` : A file listing the package dependencies.

- An Example Python Package:
  - https://github.com/labxscut/elsa
  - Study this setup.py
- The elsa (py2) Dockerfile with Deployable Python Package:

```
FROM ubuntu:latest

# Set the working directory (will return to WORKDIR after each RUN)

WORKDIR /setup

# Install prerequisites

RUN apt-get update --fix-missing

RUN apt-get -y install curl git git-lfs build-essential             # install curl, git and build tools

RUN apt-get -y install python2 python2-dev                  # install python2

RUN apt-get -y install python-is-python2 python-dev-is-python2 # set python2 to system python

RUN apt-get -y install python-setuptools                    # install python2 and necessaries
```

```
RUN curl https://bootstrap.pypa.io/pip/2.7/get-pip.py | python2  # install pip
RUN pip install numpy scipy                                      # install numpy and scipy
# RUN pip install conda                                          # install conda
# Install elsa
RUN git lfs clone --verbose https://bitbucket.org/charade/elsa.git
RUN cd elsa && python2 setup.py install --force
# Run elsa
RUN lsa_compute --help
```

- Build and use the islds-elsa container image

```
h> docker build -t islds-elsa -f Dockerfile.elsa .
h> docker run -v /mnt/e/tmp:/mnt/tmp -d --name u1 islds-elsa sleep infinity
h> docker exec -it u1 lsa_compute --help      # run dockerized elsa app lsa_compute
h> docker exec -it u1 lsa_compute /mnt/tmp/lec4/ARISA20.csv /mnt/tmp/lec4/ARISA20.theo.elsa -r 1 -s 127 -d 3 -p theo                    # use the lsa_compute for scientific calculation
```

- *Notes:*
  - The elsa package was build for python 2.
  - A tool inside a docker container is dockerized, and can be used by docker exec container tool
  - To reach dockerhub (blocked by GFW) configure DockerDesktop/Settings/Resources/Proxies accordingly.
  - To avoid disk space configure DockerDesktop/Settings/Resources/Advanced/"Disk Image Location" to a folder with abundant free space.
  - For interactive debug, note will not return to WORKDIR after each RUN cmd, needs manual cd
  - The input file can be provided in /mnt/e/tmp and the output file can be found in /mnt/e/tmp

5. Code Management Tools

5.1 Git and Github

- **Git and Github**
  - *Hands-On (dockerized git):*
    - Github: https://en.wikipedia.org/wiki/GitHub
    - Install and use a dockerized git

```
h> docker run -v /mnt/e/tmp:/mnt/tmp -d --name u1 islds-elsa # use dockerized git
h> docker exec u1 /bin/bash
c# apt-get install -y git git-lfs
h> docker commit u1 islds-elsa
```

    - Git Tutorial: https://www.bilibili.com/video/BV1WW411Q7EW

> https://www.bilibili.com/video/BV1WW411Q7EW
> **Git基本命令行用法讲解_哔哩哔哩_bilibili**
> Git基本命令行用法讲解共计34条视频，包括:01-git-help、02-git-config、03-git-init等，UP主更多精彩视频，请关注UP账号。

  - *Hands-On (SignIn/SingUp to Github):*
    - SignIn/SingUp to Github: https://github.com
    - Tutorial Video: https://www.bilibili.com/video/BV1614y1k7CS

> https://www.bilibili.com/video/BV1614y1k7CS
> **使用教育邮箱申请github学生包以及免费copilot_哔哩哔哩_bilibili**
> 这个网址包含我的文字说明，不过是为暨南大学的学生量身定制的，当然，只要你有edu.cn邮箱都能够适用，https://qm0.website/myhtml/github-copilot-tutorial.html, 视频播放量 30264、弹幕量 2、点赞数 285、…

  - *Hands-On (configure git to access GitHub via ssh):*
    - Configure dockerized git to use GitHub via ssh

```
c# mkdir ~/.ssh             #create .ssh folder
c# ssh-keygen -t id_rsa_work -C "your_email@example.com"
c# cp id_rsa_work* ~/.ssh/    #if you already have keys
c# vi ~/.ssh/config
###  ~/.ssh/config ###
Host github
 Hostname github.com
 User chaelir
```

> IdentityFile ~/.ssh/id_rsa_work
>
> c# ssh -T -v git@github.com
>
> #Hi chaelir! You've successfully authenticated, but GitHub does not provide shell access.

- *Hands-On (configure git to use GitHub):*
  - Configure git to use github

> c# git config --global user.name "Charlie Xia"
>
> c# git config --global user.email "lcx.scut@outlook.com"
>
> c# git clone git@github.com:chaelir/islds.git
>
> c# git commit && git push

- *Notes:*
  - Use an edu.cn email account for free copilot? Not verified

5.2 Application: manage Python code with Git

- **CCNMF: An exmaple R Package presented and maintained on GitHub**
  - *Hands-On:*
    - Github: https://github.com/labxscut/ccnmf

> c# cd /mnt/tmp/lec4 && git clone https://github.com/labxscut/CCNMF.git

  - *Notes:*
    -

5.3 Application: manage R code with Git

- **ELSA: An example Python Package presented and maintained on GitHub**
  - *Hands-On:*
    - Github: https://github.com/chaelir

> c# ssh-keygen # save the key in ~/.ssh/id_github, add the pub key to the account
>
> c# chmod 400 ~/.ssh/id_github # if you copied somewhere else, change the file readable only2you
>
> c# vi ~/.ssh/config

    - The contents of ~/.ssh/config

> Host github

> Hostname github.com
>
> User git
>
> IdentityFile ~/.ssh/id_github

- Github: https://github.com/labxscut/elsa

> c# ssh -T git@github.com # this should work
>
> c# cd /mnt/tmp/lec4 && git clone git@github.com:labxscut/elsa.git #or fork and clone your own repo
>
> c# cd elsa && cp ../Dockerfile.elsa .
>
> c# git add Dockerfile.elsa
>
> c# git commit -m "added a new working Dockerfile"
>
> c# git push
>
> h> docker commit u1 islds # commit this to the new image file

- Take a look at the Github, it should reflect the change you've just made

- *Notes:*
  - All labx repos are owned by the labxcode (LABX-GitHub-Admin) account
  - To be allowed to write to the repos, you need to request "Write" permission and granted by the admin.
  - The new islds new image file shall have github configured next time you spin it up.
  - If you are a collaborator who pushes edits to repo of others, please invite them to write and accept first.

5.4 Application: misc git operations

- **Learn commonly used git operations**

```
### git create a new repo on github
# github create a repo -> clone the repo to local -> add files -> commit and push

### git rename a branch and delete the old branch.
git branch -m v3 py3
git push origin --delete v3
git push origin py3
git push --set-upstream origin py3

### git change remote origin and push to new origin
```

```
git remote set-url origin git@github.com:labxscut/islds.git
git pull

```

6. R Coding Tools

6.1 R and Rstudio

- **Install R + Rstudio in Windows**
  - *Hands-On:*
    - Tutorial: https://www.bilibili.com/video/BV18w4m127L5

      https://www.bilibili.com/video/BV18w4m127L5

      科研技能 | R和RStudio的安装和使用简介_哔哩哔哩_bilibili

      正好本UP新买的台式电脑里没装R和RStudio，我想着借此机会给大家录一个安装软件的视频教程。由于R的很多教程都可以在网上搜到经验贴，所以本视频更想给大家演示的是如何使用百度来完成自主学习。,…
- **Use an official R + Rstudio docker image**
  - *Hands-On:*
    - Use it with docker desktop on windows
      - h> docker pull rocker/rstudio
      - h> docker run -d -p 8787:8787 -e PASSWORD=your_password --name rstudio rocker/rstudio
    - Open localhost:8787 and input username and password to use
  - *Notes:*
    - You will still have to commit your changes to the container as an image to save and reuse
- **DIY install R + Rstudio in Dockerized Ubuntu**
  - Self-review basic docker operations
  - *Hands-On:*

```
h> docker image tag islds-elsa islds            # give islds-elsa the new name islds

h> docker run -v /mnt/e/tmp:/mnt/tmp -d --name u1 islds sleep infinity # create a working container

h> docker run -v E:\tmp:/mnt/tmp -d --name u1 islds # if in powershell

h> docker exec -it u1 /bin/bash   # open a bash
```

```
h> docker commit u1 islds # commit interactive modifications
h> docker stop u1 && docker rm u1 # kill and remove container
```

- **Add R and Rstudio to container interactively**
  - Tutorial: https://posit.co/download/rstudio-server/

```
c# cd /mnt/tmp/lec5                    # use this working direcotry
c# apt-get update --fix-missing        # fix missing packages
c# apt-get -y install r-base r-base-dev    # install r-base and r-base-dev for headers
c# apt-get -y install gdebi-core       # install jupyter notebook
c# curl -L -o rs.deb https://download2.rstudio.org/server/focal/amd64/rstudio-server-2024.04.2-764-amd64.deb    # download the rstudio install file
c# gdebi rs.deb                        # install rstudio server
c# useradd -m -s /bin/bash rstudio && echo "rstudio:rstudio" | chpasswd #usr:pwd
```

- **Use dockerized Rstudio-Server:**
  - A Test Run on port 8787:

```
h> docker run -p 8787:8787 -v /mnt/e/tmp:/mnt/tmp d --name u1 islds
c# rstudio-server verify-installation   # fix if any errors
c# rstudio-server stop
c# rstudio-server start
```

- Misc **Resources**
  - R Cheatsheet: https://www.datacamp.com/cheat-sheet/getting-started-r
  - Rstudio Cheatsheet: https://rstudio.github.io/cheatsheets/html/rstudio-ide.html
  - How to expose host folder to Rstudio server?
  - Learn R based on new packages, some R cheatsheet is way too old, not recommended!, bad E.g.
    - https://cran.r-project.org/doc/contrib/Short-refcard.pdf

6.2 Application: Dockerfiles for R + Py devel

- **Use the image from islds repo**
  - code available at: https://github.com/labxscut/islds

```
h> docker image tag islds-elsa islds              # give islds-elsa the new name islds
h> docker run -v /mnt/e/tmp:/mnt/tmp -d --name u1 islds sleep infinity # create a working container
h> docker run -v E:\tmp:/mnt/tmp -d --name u1 islds # if in powershell
h> docker exec -it u1 /bin/bash   # open a bash
h> docker commit u1 islds # commit interactive modifications
h> docker stop u1 && docker rm u1 # kill and remove container
```

- **Self-build: build isl-cpp use Ubuntu:latest and Dockerfile.cpp**

```
### Dockerfile.cpp ###
FROM ubuntu:latest
# All Necessary configs
ENV DEBIAN_FRONTEND=noninteractive
SHELL ["/bin/bash", "-c"]
WORKDIR /setup

# Install C prerequisites
RUN apt-get update --fix-missing && \
   apt-get -y install curl git git-lfs build-essential && \
   rm -rf /var/lib/apt/lists/*
```

- **Self-build: build isl-conda use isl-cpp and Dockerfile.conda**

```
### Dockerfile.conda ###
FROM isl-cpp
# All Necessary configs
ENV DEBIAN_FRONTEND=noninteractive
SHELL ["/bin/bash", "-c"]
WORKDIR /setup

# Install Python and Conda
RUN apt-get update --fix-missing && \
   apt-get -y install python3 python3-dev python3-pip && \
```

```
    curl -sLo miniconda.sh https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh && \
    bash miniconda.sh -b -p ~/miniconda && \
    echo 'export PATH=~/miniconda/bin:$PATH' >> ~/.bashrc && \
    rm miniconda.sh && \
    rm -rf /var/lib/apt/lists/*
```

- **Self-build: build isl-rpy use isl-conda and Dockerfile.rpy**

```
### Dockerfile.rpy ###
FROM isl-conda
# All Necessary configs
ENV DEBIAN_FRONTEND=noninteractive
SHELL ["/bin/bash", "-c"]
WORKDIR /setup

# Install RStudio server, port 8787
# docker run -p 8787:8787 -v /mnt/e/tmp/:/home/rstudio/data --name rs -d isl-rpy rstudio-server start
RUN apt-get update --fix-missing && \
    apt-get -y install r-base r-base-dev && \
    apt-get -y install gdebi-core && \
    curl -sLo rstudio-server.deb https://download2.rstudio.org/server/jammy/amd64/rstudio-server-2024.04.2-764-amd64.deb && \
    gdebi -n rstudio-server.deb && \
    rm rstudio-server.deb && \
    useradd -m -s /bin/bash rstudio && \
    echo "rstudio:rstudio" | chpasswd && \
    rm -rf /var/lib/apt/lists/*

# Install Jupyter Notebook server, port 8888
# docker run -p 8888:8888 -v /mnt/e/tmp:/mnt/tmp --name pnb -d isl-rpy jupyter notebook
```

```
ENV PATH=~/miniconda/bin:$PATH
RUN conda create -n islds -y python=3.8 && \
  echo "source activate islds" >> ~/.bashrc && \
  . ~/miniconda/etc/profile.d/conda.sh && \
  conda activate islds && \
  pip install -y numpy scipy matplotlib && \
  conda install -y -c conda-forge notebook


# docker run -p 8787:8787 -p 8888:8888 -v /mnt/e/tmp/:/home/rstudio/data -v /mnt/e/tmp:/mnt/tmp --name rpy -d isl-rpy sleep infinity
# rstudio-server start
# jupyter notebook --allow-root --notebook-dir=/mnt/tmp --ip='0.0.0.0' --port=8888 --no-browser
```

- **Self-build: commands to build Dockerfiles and run ervers**

```
docker build . -f Dockerfile.cpp -t isl-cpp
docker build . -f Dockerfile.cpp -t isl-conda
docker build . -f Dockerfile.cpp -t isl-rpy
docker run -p 8787:8787 -p 8888:8888 -v /mnt/e/tmp/:/home/rstudio/data -v /mnt/e/tmp:/mnt/tmp --name rpy -d isl-rpy sleep infinity
rstudio-server start #use rstudio on localhost:8787
jupyter notebook --allow-root --notebook-dir=/mnt/tmp --ip='0.0.0.0' --port=8888 --no-browser # use notebook on localhost:8888
```

6.3 Data Visualization with R

- **ggplot2**
  - *Hands-On:*
    - Tutorial Video: https://www.bilibili.com/video/BV1pP411x7Tv

https://www.bilibili.com/video/BV1pP411x7Tv

1.新版ggplot2-3.4.2精讲——基础语法_哔哩哔哩_bilibili

零基础学习ggplot2图形语法, 视频播放量 7466、弹幕量 2、点赞数 235、投硬币枚数 112、收藏人数 805、转发人数 31, 视频作者 五柳冰冰, 作者简介 R包最全分享者! 跟踪R官网上最新R包的发布信息，精…

  - *Notes:*

    - Plotting Gallery: https://r-graph-gallery.com/
    - ColorBrewer (use to have a RColorBrewer site): https://colorbrewer2.org/
    - Identify the color set in ColorBrewer and use it with the RColorBrewer package: https://r-graph-gallery.com/38-rcolorbrewers-palettes.html
- **tidyverse**
  - *Hands-On:*
    - Tutorial Video: https://www.bilibili.com/video/BV1Nj41167Bv

    https://www.bilibili.com/video/BV1Nj41167Bv
    4.新版ggplot2精讲——从整洁的数据开始_哔哩哔哩_bilibili
    4.新版ggplot2精讲——从整洁的数据开始, 视频播放量 928、弹幕量 2、点赞数 23、投硬币枚数 8、收藏人数 31、转发人数 1, 视频作者 五柳冰冰, 作者简介 R包最全分享者! 跟踪R官网上最新R包的发布信息，精炼讲解其…

  - *Notes:*
    - Tidyverse is a new way to organize and manage data in R, an alternative to the old dataframe.
    - Many SQL operations are supported in tidyverse, so you can treat data.table as a database.
    - Tidyverse cheatsheet: https://www.datacamp.com/cheat-sheet/tidyverse-cheat-sheet-for-beginners
- **shiny**
  - *Hands-On:*
    - Beginner Tutorial: https://www.youtube.com/watch?v=jxsKUxkiaLI
    - Tutorial Video: https://www.youtube.com/watch?v=9uFQECk30kA
  - *Notes:*
    - Tidyverse and Shiny tutorials are scarce on Bili, use youtube instead.

7. Document Writing Tools

7.1 Follow the author's instruction

- **Where to find the instruction**
  - *Hands-On (Authoring a paper)*
    - Find the **Bioinformatics** Journal Author's Instruction
    - Go https://academic.oup.com/bioinformatics/ ; > Submit ; > Author Guidelines

> Scope
>
> Types of Manuscript
>
> Pre-submission enquiries
>
> Format-free submissions
>
> General Policies
>
> Open Access
>
> Manuscript Preparation
>
> Detailed Scope Guidelines
>
> Manuscript Transfer

    - Pay attention to manuscript types; pages; words, figures, table number limits
    - Pay attention to software and data release requirements (github?)
    - Pay attention to figure requirements: format, resolution
    - Pay attention to citation and reference requirements
    - Pay attention to sections by manuscript type
    - Find and use the .doc or .tex template (if available)
        - Tex: https://www.overleaf.com/latex/templates/oup-general-template/ybpypwncdxyb
        - Doc: https://static.primary.prod.gcms.the-infra.com/static/site/bioinformatics/document/cabios-word-temp.zip?node=9b946b36faff1c196dba&version=541801:1bc4a5f5521ada3b11e7
    - Ex. Find a suitable journal/conference for your paper submission, edit your paper according to their instruction, include pleace holder.
- *Hands-On (Oral presentation a paper)*
    - Find the ICML 2024 conference speaker's instruction.
    - Official conference site: https://icml.cc/Conferences/2024/PresenterInstructions
    - Pay attention to: presentation time, question time, connector available, screen ratio/size, ...
- *Hands-On (Poster presenting a paper)*
    - Find the ICML 2024 conference poster's instruction
    - Official conference site: https://icml.cc/Conferences/2024/PosterInstructions
    - Pay attention to: virtual or real, video, poster ratio/size, posting/presenting/retracting time, ...

- **Submit to biorxiv and medrxiv to use preprints transfer service**
  - biorxiv.org
  - medrxiv.org

7.2 Writing word document

- **Write a paper using journal .doc template**
  - *Hands-On (Writing a doc manuscript)*
    - Bioinformatics doc template (awful)

      MS Word Template Bioinformatics.dotx
    - NAR template (cool)

      nar-word-template.docx
    - So NAR can be used for writing and change to Bioinformatics before submission
    - But sections, figures, citations, word limits still follow Bioinformatics instruction
    - You can save the NAR template for other purposes, e.g. writing the paper for this course

      elsa-draft.zip
  - *Notes*
    - Must use Microsoft Word, no WPS!

7.3 Writing latex document

- **Write a paper using journal .tex template**
  - *Hands-On (Writing a tex manuscript)*
    - Both Bioinformatics and NAR use the same latex template (awful)
      - https://www.overleaf.com/latex/templates/oup-general-template/ybpypwncdxyb

      MS Word Template Bioinformatics.tex
    - Biostatistics has a good looking latex template (cool)
      - https://www.overleaf.com/latex/templates/template-for-biostatistics-journal/fggrvggtkxmg

      biostatistics-template.tex
    - So Biostistics can be used for writing and change to Bioinformatics before submission

    - Sections, figures, citations, word limits still follow Bioinformatics instruction
    - You can save the Biostatistics template for other purposes, e.g. writing the paper for this course

  CCNMF_NAR_latex_manuscript.zip

- **Write an oral presentation with beamer .tex template**
  - *Hands-On (Writing a tex slide)*

  elsa-talk.zip
- **Write a poster with .tex template**
  - *Hands-On (Writing a tex poster)*

  elsa-poster.zip
- **Use Texlive + vim is highly recommended**
  - *Hands-On (Install and use texlive and vim)*
    - Install texlive in ubuntu based docker

    > apt-get install texlive vim

7.4 Writing PowerPoint document

- **Draw graphical art with PowerPoint**
  - *Hands-On*
    - Draw an excellent concept illustration / graphical abstract (**Fig. 1**) with PowerPoint

    CCNMF Fig1 concept illustration.pptx
    - Draw a nice workflow / dataflow chart (**Fig. 2**) with PowerPoint.
- **Write an oral presentation with PowerPoint**
  - *Hands-On*

    CCNMF-ppt-presentation.pptx
- **Write a poster with PowerPoint**
  - *Hands-On*

    CCNMF-ppt-poster.pptx

8. Document Editing Tools

8.1 Painless Internet Connection

- Use wegetcloud.ltd + clash_for _windows proxy (see instructions for purchase and how to therein).
- Download and install an **Edge** browser.
- Sign in it with a dedicated microsoft MS account (say usr.scut@outlook.com if your are in labx)
- Use the Edge and the MS account for all your academic accounts/searches/favorites in the future.

8.2 Copilot

- (Free tier)
- Connect to the internet with a USA proxy.
- Open Edge and visit www.bing.com, find copilot in search, tab and right corner
- Sign-in bing with the MS account to access your copilot q&a history anywhere
- **Application**: expand a paper's outline with Copilot

```
#Prompts Examples#
Can you expand to a paragraph based on this top sentence, "THE TOPIC SENTENCE"?
```

8.3 ChatGPT

- (Free tier, Pay for more functions USD 20/Month)
- Open the Edge and visit chatgpt.com, register user with openai's chatgpt
- **Application:** advanced writing and organization of large text body with Chatgpt

```
#Prompts Examples#
This is an introduction a non-native speaker student wrote in English. Can you reorganize the flow of text, break and exchange paragraphs if necessary, and also address these considerations you raised.
```

8.4 Grammarly

- (Free tier, Pay for more functions, USD 144/Year)
- Install and register user with grammarly: https://www.grammarly.com/desktop
- **Application**: near-finish editing of a paper using Grammarly

8.5 Zotero

- (Free, Pay for Space 2GB USD20/Year)
- Install and register user with zotero: https://www.zotero.org/
- Tutorial: https://www.zotero.org/user/validate/
- **Application**: manage a paper's references with Zotero
  - E.g. Find a new reference online, add it to Zotero and cite it in paper

9. Team Collaborating Tools

9.1 OneDrive

- OneDrive: https://en.wikipedia.org/wiki/OneDrive
- See labx信息环境 for more OneDrive tips.
- **Application**: OneDrive and ChatGPT integration， upload and analyze a docx file.
- 华工微软正版化平台 - 似乎不能用了！

9.2 Outlook

- Outlook: https://en.wikipedia.org/wiki/Outlook
- See labx信息环境 for more Outlook tips.

9.3 Feishu

- OneDrive: https://en.wikipedia.org/wiki/Feishu
- See labx信息环境 for more Feishu tips.
- Application: programmatic export of Feishu Wiki Doc
  - Go https://open.feishu.cn/app/ > Create Custom App > name it wiki_access > Click wiki_access > Credentials and Basic Info > Note the App Id and Secret
  - Go https://open.feishu.cn/api-explorer/ > Click wiki_access > Generate tenant_access_token and user_access_token ; will expire in 6900 secs. see https://open.feishu.cn/document/server-docs/api-call-guide/calling-process/get-access-token
  - Go https://open.feishu.cn/app/ > Click wiki_access > Permissions & Scopes > Enable scopes wiki:wiki:readonly, docs:document:readonly, docs:document::export; drive:export:readonly for access a Doc on Wiki
  - With the above configuration, using the commands as follows:

```
### use | jq . to view the output nicely
### access a Doc: https://open.feishu.cn/document/server-docs/docs/docs-overview
### every Doc on Feishu has a document_id, or wiki Doc has obj_token



### need to get login_preauthorization_code > user_access_token
### need to use user_access_token to see what the user can see
### go API scopes and add wiki:wiki scope to wiki_access
### go publish the API as wiki_access 1.0.0
curl -X POST https://open.feishu.cn/open-apis/auth/v3/app_access_token/internal/ -H "Content-Type: application/json" -d '{"app_id": "ur_app_id", "app_secret": "ur_app_secret"}'
# E.g. curl -X POST https://open.feishu.cn/open-apis/auth/v3/app_access_token/internal/ -H "Content-Type: application/json" -d '{"app_id": "cli_a642980d2cbe500e", "app_secret": "7jciRfpGq2zQ4KgJ7iVFqhOEo3g8Atsb"}'
# app/tenant_access_token: t-g1048ngcMO6OZPSPOMDJCECXLPDEFHETXE5V4BN4
### get a list of wiki spaces
curl -X GET 'https://open.feishu.cn/open-apis/wiki/v2/spaces' -H 'Authorization: Bearer u-eJSY_lmnR2vEgbn0Tv0Bpa5l0AAw45fHpMw0g1c8wIsu'
curl -i -X POST 'https://open.feishu.cn/open-apis/authen/v1/access_token' -H 'Content-Type: application/json' -H 'Authorization: Bearer ur_app_access_token' -d '{ "code": "xMSldislSkdK", "grant_type": "authorization_code" }'
# E.g. curl -X POST 'https://open.feishu.cn/open-apis/authen/v1/access_token' -H 'Content-Type: application/json' -H 'Authorization: Bearer t-g1048ndT2YRYYCYM2TY7HLTIALS27UZGU6F67G6D' -d '{ "code": "xMSldislSkdK", "grant_type": "authorization_code" }'
curl -X POST 'https://open.feishu.cn/open-apis/authen/v1/oidc/access_token' \
-H 'Authorization: Bearer t-g1048ndT2YRYYCYM2TY7HLTIALS27UZGU6F67G6D' \
-H 'Content-Type: application/json' \
-d '{
  "code": "xMSldislSkdK",
  "grant_type": "authorization_code"
}'
curl -X GET 'https://open.feishu.cn/open-apis/wiki/v2/spaces' -H 'Authorization: Bearer u-fdw0PMShJeM9EbpKtQOQ1ikh4Ix5g5d9OG00l1oywBKA'
# E.g. curl -X GET 'https://open.feishu.cn/open-apis/wiki/v2/spaces' -H 'Authorization: Bearer u-fdw0PMShJeM9EbpKtQOQ1ikh4Ix5g5d9OG00l1oywBKA'
curl -X GET https://open.feishu.cn/open-apis/wiki/v2/pages/:page_token -H "Authorization:
```

```
ur_tenant_access_token"
# E.g. curl -X GET https://open.feishu.cn/open-apis/wiki/v2/pages -H "Authorization: t-g1048ndT2YRYYCYM2TY7HLTIALS27UZGU6F67G6D"

curl -X GET 'https://open.feishu.cn/open-apis/wiki/v2/spaces/:space_token/pages' -H 'Authorization: Bearer u-fdw0PMShJeM9EbpKtQOQ1ikh4Ix5g5d9OG00l1oywBKA'
#e.g. curl -s -X GET 'https://open.feishu.cn/open-apis/wiki/v2/spaces/7120803795587596289/pages' \ -H 'Authorization: Bearer u-dSuhK5i7x8gGqbuLAyUitl5l0A4045LHNMw00hs8wIpq'
#
```