기본 생물정보학 분석법 공부방법

향후 추가 예정 (v240214)

이 문서의 학습 목표:

`README` 만 읽으면 돌릴 수 있어요~ 를 알려주는 것

- CLI (Command-Line Interface)를 통해 어떤 생물정보학 프로그램이던, `README`만 읽으면 돌릴 수 있게 되는 것
- `README`란?
  - 프로그램의 개발자가 사용자에게 어떻게 설치하는지, 프로그램의 목적이 무엇인지, 어떻게 수행하는지, 수행 결과를 어떻게 해석하는지 알려주는 문서
  - 텍스트로 작성하며, github에 올릴 (reposit)할 때는 특별한 경우가 아닌 한 영어로 작성하는 것이 원칙.
- 이 문서에서는 `README`만 읽으면 분석을 수행할 수 있는 기초 체력을 함양하는 것을 목적으로 한다.

```
"아니, 그러면 README만 읽으면 무슨 프로그램이든 돌릴 수 있는 거 아냐?"
맞는 말씀.


그러나 README를 작성하는 사람들이 과연 읽는 사람을 100% 배려해서 완전 초보자도 프로그램을 쓸 수 있도록 고려할까?
아니다.


따라서 이번 문서에서는 README를 읽는 방법과 읽은 이후 어떻게 프로그램을 실제로 돌리는지 알아볼 것이다.
```

- 이 문서는 문제집에 가깝다.
  - 모음집에 있는 프로그램들을 직접 돌리다 보면, 실력이 자동으로 어느 샌가 성장해 있다.

대부분 분석의 워크플로우

1. 원하는 분석을 구상한다. (예: Genome assembly를 하고 싶어!)

2. 원하는 분석을 구현한 논문을 찾거나 프로그램을 찾는다. (예: Genome assembler를 구글에 검색)
    a. 이 때, 논문은 bioRxiv에 키워드로 검색하거나 최고 권위지인 Nature Biotech부터 Nature genetics, Nucleic Acides Research, Bioinformatics, Breifings in bioinformaics, BMC bioinformatics 대충 이런 순서로 키워드 검색하여 찾는다. 높은 저널일수록 `README`가 자세할 확률이 높다. (리뷰어들이 깐깐하므로)
    b. 다 찾았는데 없다면 아마 본인이 개발을 직접 해야 할 확률이 높지만, 지푸라기잡는 심정으로 github에 키워드로 검색하면 나올 때도 있다.
3. 해당 분석을 제시한 논문의 초록과 메소드를 읽고 대강 감을 잡는다. (예: Flye의 README와 INSTALL)
4. github, gitlab이 대부분일 테지만 가끔가다 오래된 분이거나 진성 컴덕이 교신저자라면 sourceforge나 냅다 ftp 같은 데다가 sourcecorde와 매뉴얼을 공개하는 경우가 있다.
    a. 매뉴얼이 없는 프로그램은 없다. 만약 프로그램의 매뉴얼이 없다면 쓰지 말자. 99%의 확률로 쓰레기 프로그램이다.
    b. 가끔 너무 좋은 프로그램인데 매뉴얼이나 `README`가 불친절해서 콜릭들이나 동종업계 종사자들이 보다못해 자기 나름대로 `README`나 매뉴얼을 배포하는 경우가 있다.
    c. 엄청나게 좋은 프로그램들은 대부분 `-h`와 같은 `--help`옵션을 치면 자세히 설명해 준다. 마틴 교수님의 `MMSeqs`가 진짜 사용자 배려 (UI; User Interface) 부분의 최강자다. 너무 프로그램 잘 만드셔서 화가 나고 억울할 정도.
5. 좋은 프로그램들이라면 install 방법, input과 output, algorithm은 설명해 준다.
    a. 이 네가지가 없다면 안 좋은 프로그램일 확률이 높다. 대안을 찾아보자.
6. Install 하는 것, input을 만드는 것, output을 해석하고 algorithm을 이해하여 다른 사람에게 설명하는 것이 생물정보 분석의 거의 다다.

컴퓨터와 친해지기

아묻따 아래 링크를 공부하자.
https://missing-semester-kr.github.io/
꼭 위 링크에서 “Version control with git”까지 읽은 후 다음 단계로 넘어가자.

생물정보학 파일 형식들

https://hhj6212.github.io/biology/tech/2020/08/26/Bioinformatics-fileformats.html
https://bioinfoblog.tistory.com/149

파일 형식들을 알아야 프로세싱할 기본 체력이 생긴다.
형식들을 외워서 종이에 그릴 수 있을 때 다음으로 넘어간다.

기초 체력 함양

- 자 install (설치) 는, 대부분의 경우 분석자가 직접 하지는 않는다.
  - 높은 확률로 분석 고인물이 서버 관리를 하고 있고, 설치는 서버 자원을 직접적으로 건드리는 부분이기 때문에 민감한 사안이라 관리자보고 해달라 하면 흔쾌히 해주기 때문이다.
- 그런데, 랩 첫째 informatician이라거나 자기만 informatics를 한다거나 그런 경우라면, 직접 해야 한다.
- 어떻게 하는 것이 가장 좋을지, 본인이 경험한 바를 토대로 설명한다.

냅다 분석용 프로그램을 설치하지 말고
무조건, anaconda, miniconda, mamba, micromamba를 사용하기를 권장한다.

- 당신이 어느 서버를 가던, 서버에 당신의 계정이 생성된다면 당장은 아무 폴더도 없을 것이다.
- 그럴 때, 무조건 위의 네 개 중 하나를 설치하자.
- 필자는 micromamba를 설치하기를 권장한다.
  - anaconda는 python으로 개발된 (느림) 자동으로 추천 채널과 패키지들이 설치되어 (무거움) 있다.
  - miniconda는 python으로 개발된 (느림) 추천 패키지가 없는 가벼운 프로그램이다.
  - mamba는 C++로 개발된 (빠름) 자동으로 추천 채널과 패키지들이 설치된 프로그램이다.
  - micromamba는 C++로 개발된 (빠른) 추천 패키지가 없는 가벼운 프로그램이다.

```
"${SHELL}" <(curl -L micro.mamba.pm/install.sh)
```

해당 명령어를 통해 micromamba를 설치할 수 있다. (linux 검은창 (bash) 에다가 바로 복붙하고 엔터하면 된다.

자, 이 프로그램을 왜 쓰느냐 하면
"한 폴더 안에 두 개 이상의 이름이 같은 파일" 이 있을 수 없기 때문이다.
모든 충돌의 근원은 이거다.
그러면 무슨 문제가 생기느냐, A라는 사람이 `samtools`를 먼저 깔고, B 라는 사람이 같은 경로에 `samtools`를 또 깔 수는 없다는 말이다. A라는 사람은 먼저 깔았기 때문에 해당 프로그램과 dependencyu에 대한 permission이 있지만 B라는 사람은 없기 때문에 분석을 수행할 수 없다.

micromamba와 같은 환경 관리 툴을 쓰면 해당 문제에서 자유로워진다. 왜냐하면 global하게 설치하는 것이 아닌 개인 계정 하에 설치하기 때문에 같은 경로에 설치되지 않기 때문이다.
또한, 자신의 계정의 home 하에 있기 때문에 일일히 관리자에게 요청하지 않아도 자신의 프로그램을 management할 수 있다.

그럼 이제 설치는, `README`에 있는 "install with conda"와 같은 section을 참고하여 간단하게

```
micromamba install -c bioconda samtools # conda install -c bioconda samtools라고 묘사된 README도 있다.
```

와 같은 명령어로 설치하면 된다.

안 깔린다면, 80%의 확률로 존재할 서버 관리자에게 부탁하여 깔아달라고 하자. 서버 관리자는 당신에게 무조건 도움을 줄 수 있다.

- 요약
  - micromamba를 깔고, 이를 이용해 프로그램을 설치한다.
  - 안 그러면 서버 자원이 꼬이는 대재앙이 일어난다. (애초에 관리자가 그렇게 되도록 냅두지 않는다...)
  - 컴파일러 버전이나 C 라이브러리라도 꼬이면 서버를 리셋해야 한다 ㅠㅠ

Input 만들기

- Input형식을 지켜주지 않으면, 당연한 말이지만 프로그램 자체가 작동하지를 않는다.
- 필자는 주로 Input은 python, perl, awk+bash 세 개 중 각이 나오는(가장 쉬워보이는) 스크립트 언어를 이용해 프로세싱한다. perl은 거의 안쓴다.
  - 파이썬의 기본적인 문법은 해당 자료를 참고하자.
    - https://www.opentutorials.org/course/4769
      - 이거 오래걸리니까 천천히 공부하기.
  - 기초 코딩은 배워야 한다. Python의 Biopython 패키지는 가장 쉽게 접근해 볼 수 있는 모듈이다.
    - 파이썬도 micromamba를 이용해 설치해야겠쥬?
  - 한주현씨의 바이오파이썬 블로그: https://korbillgates.tistory.com/72
    - 해당 블로그에 중요한 태스크들에 대한 방법들이 잘 나와 있다.

        - 해당 도서도 추천한다.
        - https://github.com/bjpublic/biopython
        - 해당 튜토리얼도 도움이 된다.
        - https://www.tutorialspoint.com/biopython/index.htm
    - 파이썬 실력을 테스트해 볼 수 있는 문제풀이 사이트도 있다. (장혜식 교수님 같은 분들이 랭커로 있다...!)
        - https://rosalind.info/problems/locations/
    - 그리고 챗지피티와 구글 제미니를 잘 이용하자.
        - 웬만한 태스크들은 설명만 잘 하면 대부분 잘 코딩해 준다.
        - 그래도 초반에는 실력 향상을 위해 직접 코딩을 추천함.

- awk은 필자에게 마틴 교수님의 수업 내용과 자료가 있으니, 필요하면 jaylim0518@gmail.com으로 연락 달라.

이게 자신이 생긴다면, 어디 한 번 이 프로그램을 돌려 봐라.
https://github.com/wyp1125/MCScanX
과연 당신은 정제에 성공할까?
프로그램이 무사히 돌아간다면, input은 성공!

Practical

- 다음은 특정 태스크들에 대한 링크이다.
- 직접 실행하고 output을 `README`에 따라 해석해 보면서 실력을 함양하자.
- 요정도만 다 돌릴 수 있도록 훈련하면 웬만한 건 다 된다.

초보자

- Homology search
    - BLAST (오래된 프로그램이라 github이 없어 다른 링크로 대체)
        - https://open.oregonstate.education/computationalbiology/chapter/command-line-blast/
    - MMSeqs2의 easy-**모듈
        - https://github.com/soedinglab/MMseqs2
        - https://mmseqs.com/latest/userguide.pdf
    - DIAMOND
        - https://github.com/bbuchfink/diamond
- Raw 데이터 다운로드

    - 이걸 할 줄 알아야 short read 들에 대한 데이터를 얻고 뒤엣 문제들을 풀 수 있겠죠?
    - https://github.com/ncbi/sra-tools

중급자

- Genome assembly
    - Flye
        - https://github.com/fenderglass/Flye
    - Verkko
        - https://github.com/marbl/verkko
    - Nextdenovo
        - https://github.com/Nextomics/NextDenovo
- Synteny scan
    - JCVI-mcscan python version
        - https://github.com/tanghaibao/jcvi
    - i-adhore
        - https://github.com/VIB-PSB/i-ADHoRe
- Read Mapping
    - Hisat2
        - http://daehwankimlab.github.io/hisat2/
    - Bowtie2
        - https://bowtie-bio.sourceforge.net/bowtie2/index.shtml

고급자

- RNA-seq downstream analysis 전반
    - https://github.com/trinityrnaseq/trinityrnaseq/wiki
- GWAS
    - GEMMA
        - https://github.com/genetics-statistics/GEMMA
    - plink
        - https://zzz.bwh.harvard.edu/plink/

- Computational molecular phylogenetics
    - PAML - MCMCTree
        - http://abacus.gene.ucl.ac.uk/software/paml.html
    - BEAST
        - https://beast.community/

README를 읽고, 돌려보자.
모르는 게 있다면 직접 물어보거나 jaylim0518@gmail.com으로 연락 주길 바란다.

Drawing

- 자, 분석을 다 했으면 이제 그림을 그릴 차례다.
- 그림으로 예쁘게 포장해야 논문이 된다.
- 필자는 주로 R이나 python으로 그림을 대강 그린 다음, ppt로 예쁘게 다듬는다.
- Python - matplotlib + seaborn
  - matplotlib
    - https://wikidocs.net/92071
  - seaborn
    - https://wikidocs.net/86290
- R - ggplot2
  - https://ggplot2.tidyverse.org/
  - https://wikidocs.net/73370

P.S. 나에게 이 간단한 모음집을 작성하게 만든 반려 S에게 아이스크림 300개 분량의 감사 인사를 전한다.