Part 1. Installing essential programs and setting python environment for scRNAseq analysis

Using Python for single cell RNA seq data analysis

Seoul National University, Department of Biological Science

Laboratory of Development and Disease Modeling

Jong Hwi Kim

Step 1. Installing essential programs for setting basic python environment

Python, Miniconda, Jupyter notebook, Biopython, Github, Unpacker

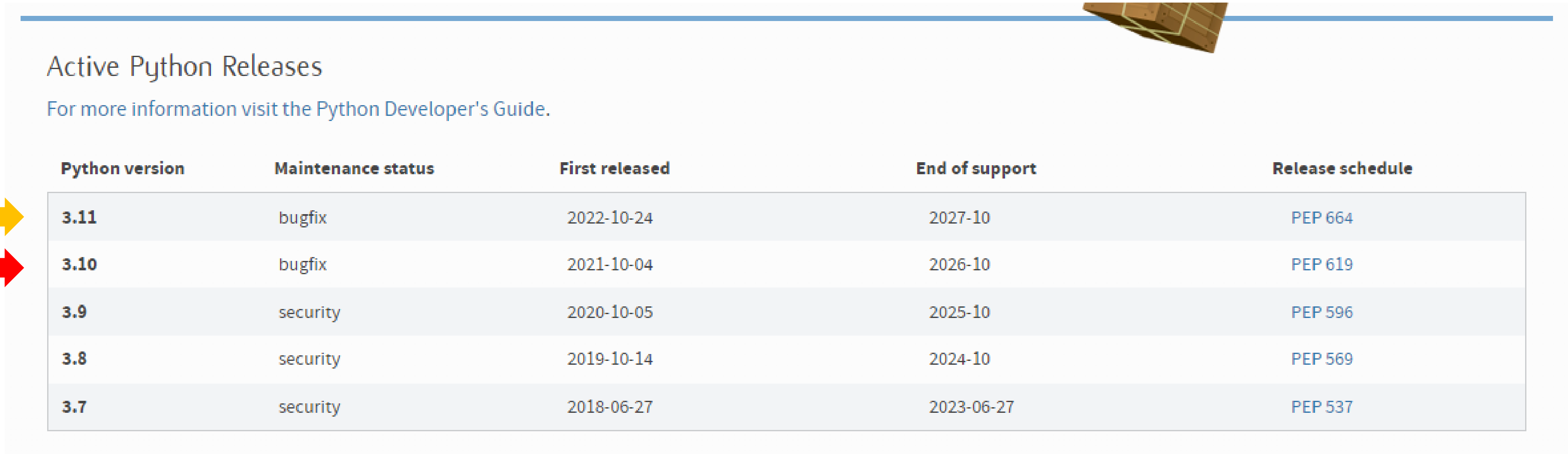
1. Installation

1.1. Python

- First step of setting Python environment on your personal computer, is obviously to install Python.
- You can download various versions of Python, including the latest version, on its official website
- Latest version of Python (02/13/2023) is 3.11.02, but installing the prior version is strongly recommended
- Various packages (Pre-developed tools that could be used in Python) often does not get updated to be compatible with the latest version of Python
- https://www.python.org/

Latest version

Installation recommended version

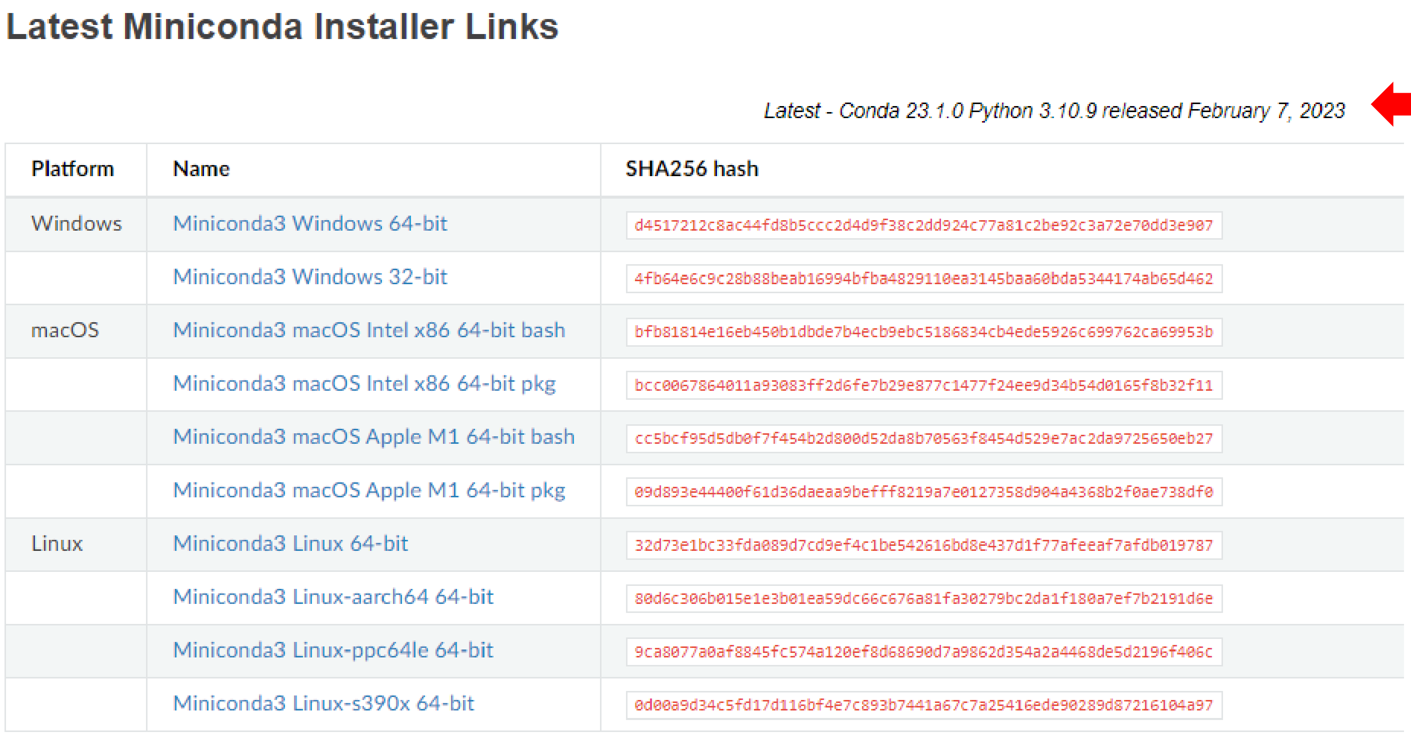


Active Python Releases

For more information visit the Python Developer's Guide.

Python version	Maintenance status	First released	End of support	Release schedule
3.11	bugfix	2022-10-24	2027-10	PEP 664
3.10	bugfix	2021-10-04	2026-10	PEP 619
3.9	security	2020-10-05	2025-10	PEP 596
3.8	security	2019-10-14	2024-10	PEP 569
3.7	security	2018-06-27	2023-06-27	PEP 537

- Miniconda is a starter package for Anaconda, a Linux based program that enable you to interact with Python
- It also contains basic packages that are frequently used in Python network
- You can use the Miniconda as operating command prompt to launch an application or install packages

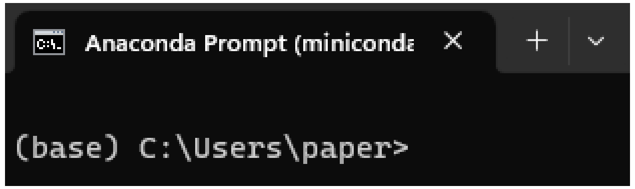


Latest Miniconda Installer Links

Latest - Conda 23.1.0 Python 3.10.9 released February 7, 2023

Platform	Name	SHA256 hash
Windows	Miniconda3 Windows 64-bit	d4517212c8ac44fd8b5ccc2d4d9f38c2dd924c77a81c2be92c3a72e70dd3e907
	Miniconda3 Windows 32-bit	4fb64e6c9c28b88beab16994bfba4829110ea3145baa60bda5344174ab65d462
macOS	Miniconda3 macOS Intel x86 64-bit bash	bfb81814e16eb450b1dbde7b4ecb9ebc5186834cb4ede5926c699762ca69953b
	Miniconda3 macOS Intel x86 64-bit pkg	bcc0067864011a93083ff2d6fe7b29e877c1477f24ee9d34b54d0165f8b32f11
	Miniconda3 macOS Apple M1 64-bit bash	cc5bcf95d5db0f7f454b2d800d52da8b70563f8454d529e7ac2da9725650eb27
	Miniconda3 macOS Apple M1 64-bit pkg	09d893e44400f61d36daeaa9befff8219a7e0127358d904a4368b2f0ae738df0
Linux	Miniconda3 Linux 64-bit	32d73e1bc33fda089d7cd9ef4c1be542616bd8e437d1f77afeeaf7afdb019787
	Miniconda3 Linux-aarch64 64-bit	80d6c306b015e1e3b01ea59dc66c676a81fa30279bc2da1f180a7ef7b2191d6e
	Miniconda3 Linux-ppc64le 64-bit	9ca8077a0af8845fc574a120ef8d68690d7a9862d354a2a4468de5d2196f406c
	Miniconda3 Linux-s390x 64-bit	0d00a9d34c5fd17d116bf4e7c893b7441a67c7a25416ede90289d87216104a97

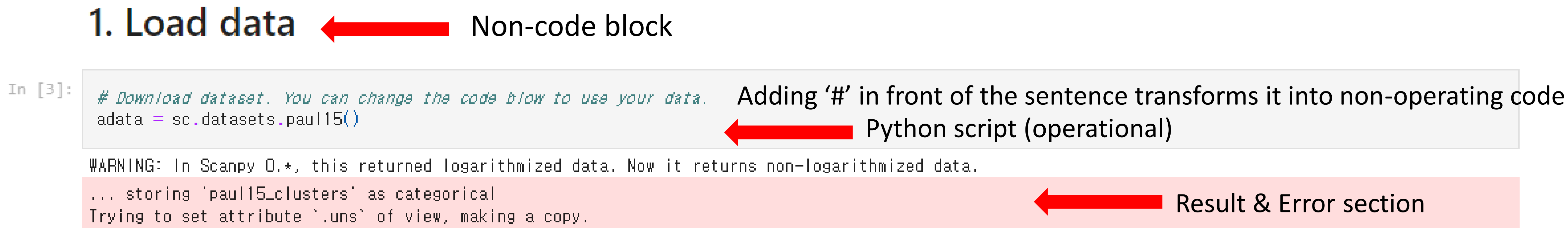
As you can see, the latest version of Python is not always your best option



```
(base) C:\Users\paper>
```

**User's tip: Changing the user name into English, with no 'U' in the first letter is highly recommended. Python cannot read \U nor other language and will report error every time.

- Jupyter notebook is an browser-based programming tool that enables separation of codes
- We can also alter the kernel to contain non-code blocks in the middle of the notebook to provide additional information
- You can add any other programming language to start writing a new notebook (ex. R, C++)
- For python, .ipynb files can be saved, which contains every code and results that could be shared in single document



1. Load data

Non-code block

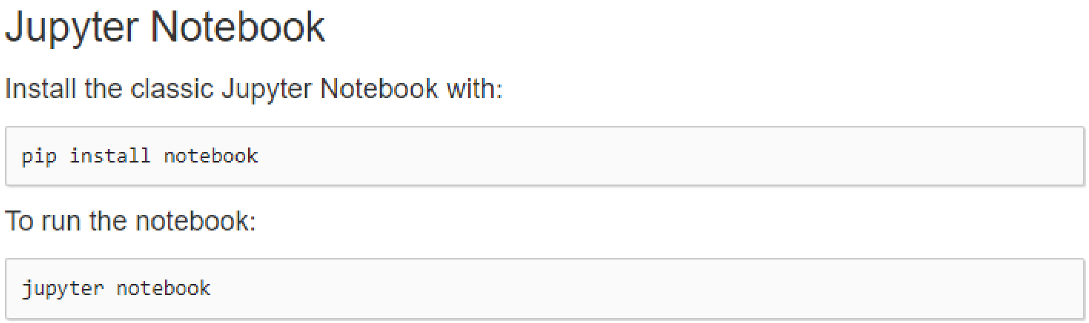
```
In [3]:  # Download dataset. You can change the code blow to use your data.
         adata = sc.datasets.paul15()
```

Adding ‘#’ in front of the sentence transforms it into non-operating code

Python script (operational)

```
WARNING: In Scanpy 0.*, this returned logarithmized data. Now it returns non-logarithmized data.
... storing 'paul15_clusters' as categorical
Trying to set attribute `.uns` of view, making a copy.
```

Result & Error section



Jupyter Notebook

Install the classic Jupyter Notebook with:

```
pip install notebook
```

To run the notebook:

```
jupyter notebook
```

1. Installation

1.4. BioPython

- BioPython is single package that contains various Python packages that are frequently used
- AnnData, Matplot.lib and Pandas are always used in Scanpy
- Basic packages listed on the previous sentence are important to be mastered before learning Scanpy
- Package could be installed with using ‘pip install’ command in the python kernel
- After installing every package, the kernel should be restarted and package should be imported in order to use the package

**User’s tip: Biopython has various packages for analyzing sequence data itself.

Check the URL below to get deeper understanding about the package

http://biopython.org/DIST/docs/tutorial/Tutorial.html

1.5. GitHub

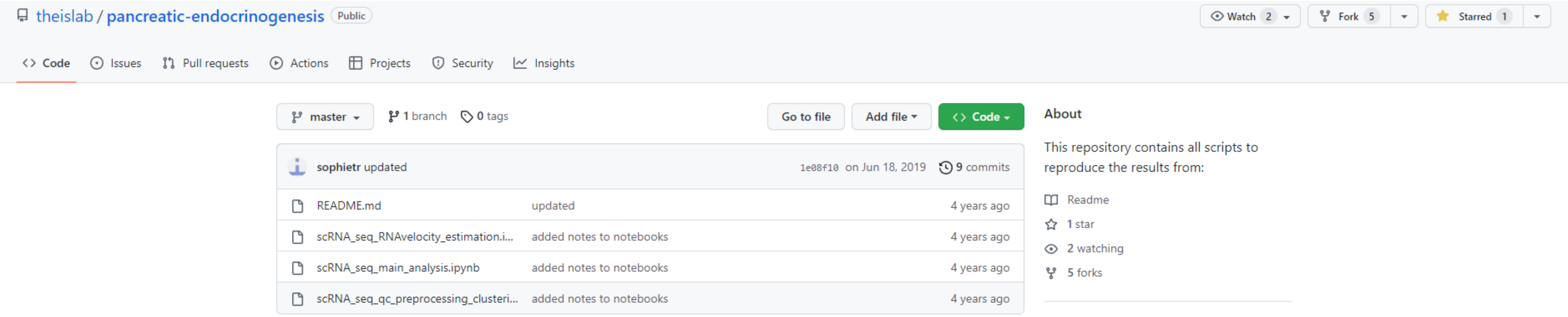
- GitHub is a website for depositing codes, making query to the users, and searching for coders with the same interest.
- GitHub provides pypi API to directly import python material.
- Also, GitHub has a lot of users majoring in bioinformatics, especially the ones who actually develop and update the packages
- You may also find jupyter notebook files that was used in the published article

paperhwi / Single_Cell_Python

- You can create your own directory on your own GitHub page

- You can follow certain individuals like Facebook. Sometimes they upload their raw analysis files, too!

Add to follow



theislab / pancreatic-endocrinogenesis Public

sophietr updated 1e08f10 on Jun 18, 2019 9 commits

README.md	updated	4 years ago
scRNA_seq_RNAvelocity_estimation.i...	added notes to notebooks	4 years ago
scRNA_seq_main_analysis.ipynb	added notes to notebooks	4 years ago
scRNA_seq_qc_preprocessing_clusteri...	added notes to notebooks	4 years ago

About

This repository contains all scripts to reproduce the results from:

Readme

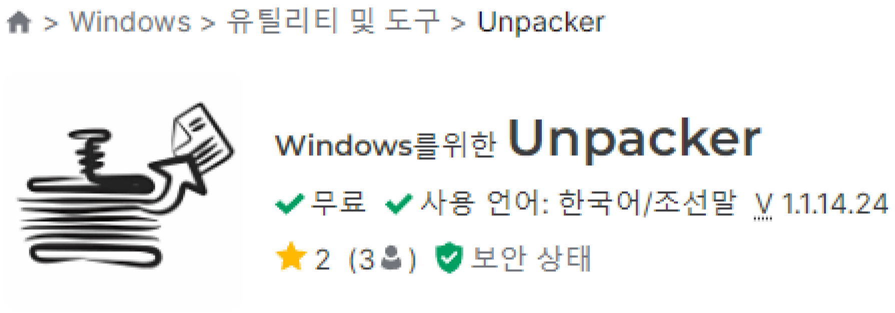
1 star

2 watching

5 forks

1.6. Unpacker

- While using Python as the tool for bioinformatic analysis, you will encounter tons of .gz files
- Unpacker is a program that can unzip any type of files downloaded from GEO database
- You can also use 'tar –xzf' command in prompt by pasting path of the file directory, but it is hard to tackle
- Downloading the installation file in https://unpacker.softonic.kr/ is an option, or you can get in windows app store

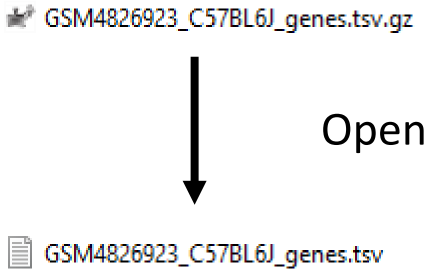


> Windows > 유틸리티 및 도구 > Unpacker

Windows를위한 Unpacker

✓무료 ✓사용 언어: 한국어/조선말 V 1.1.14.24

★ 2 (3) 보안 상태



GSM4826923_C57BL6J_genes.tsv.gz

GSM4826923_C57BL6J_genes.tsv

Open .gz file with Unpacker to unzip

**User's tip

- Txt, tsv file: read with scanpy.read_text(path_for_the_file)
- Csv file: read with scanpy.read_csv(path_for_the_file)
- Loom file: read with scanpy.read_loom(path_for_the_file) – loom files are hard to read as the loompy has ceased to sync with latest version of Scanpy. Try installing via 'pip install loom == *older version*'
- Affy file: Microarray file should be processed into matrix via BioAffy. And than import the matrix/h5ad file

Step 2. Installing Scanpy and basic attachable packages

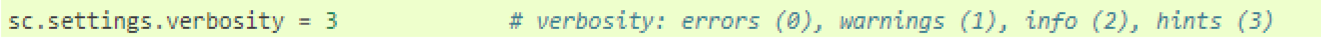
Scanpy, AnnData, Pandas, SCVI, Matplot.lib

**User's tip: If the version does not fit the currently installed Scanpy, search the compatible/older version of the package and reinstall with 'pip install *package_name==version*'

Scanpy – Single-Cell Analysis in Python

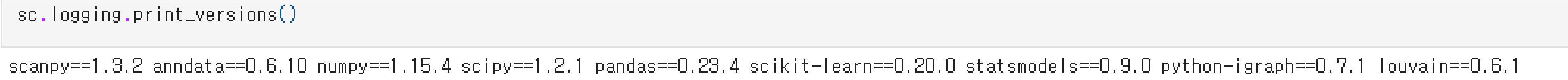
Scanpy is a scalable toolkit for analyzing single-cell gene expression data built jointly with anndata. It includes preprocessing, visualization, clustering, trajectory inference and differential expression testing. The Python-based implementation efficiently deals with datasets of more than one million cells.

- Scanpy provides tutorials for the users (https://scanpy.readthedocs.io/en/stable/tutorials.html)
- Scanpy also provides detailed descriptions about each command that you can use during analysis
- If you set the verbosity, you can also earn feedbacks for writing valid codes



```
sc.settings.verbosity = 3             # verbosity: errors (0), warnings (1), info (2), hints (3)
```

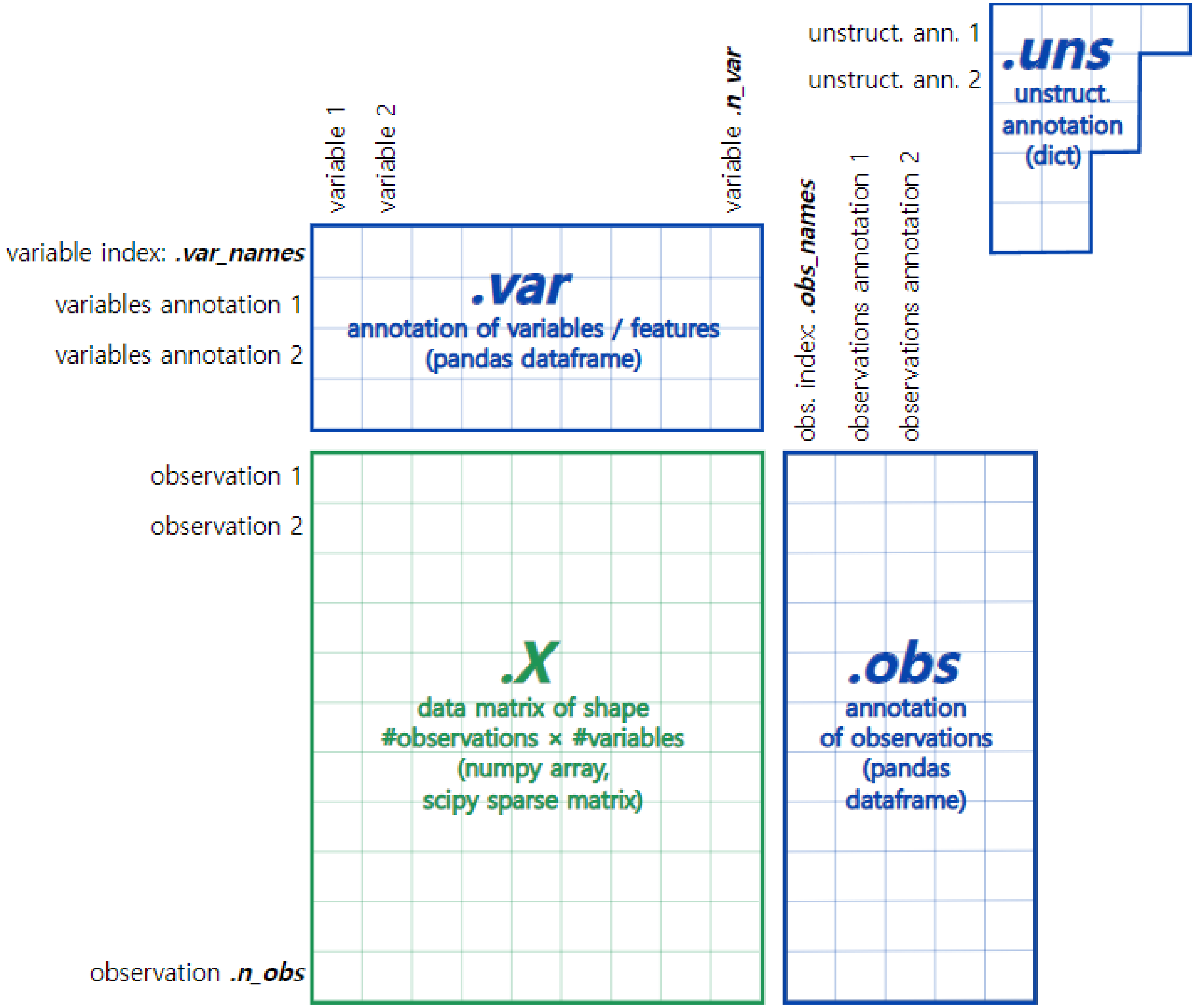
- Set the verbosity to 3 in prior, and dumb it down if you get used to analysis
- Checking all the versions of the package before importing dataset is highly recommended



```
sc.logging.print_versions()
```

```
scanpy==1.3.2 anndata==0.6.10 numpy==1.15.4 scipy==1.2.1 pandas==0.23.4 scikit-learn==0.20.0 statsmodels==0.9.0 python-igraph==0.7.1 louvain==0.6.1
```

2.2. Anndata

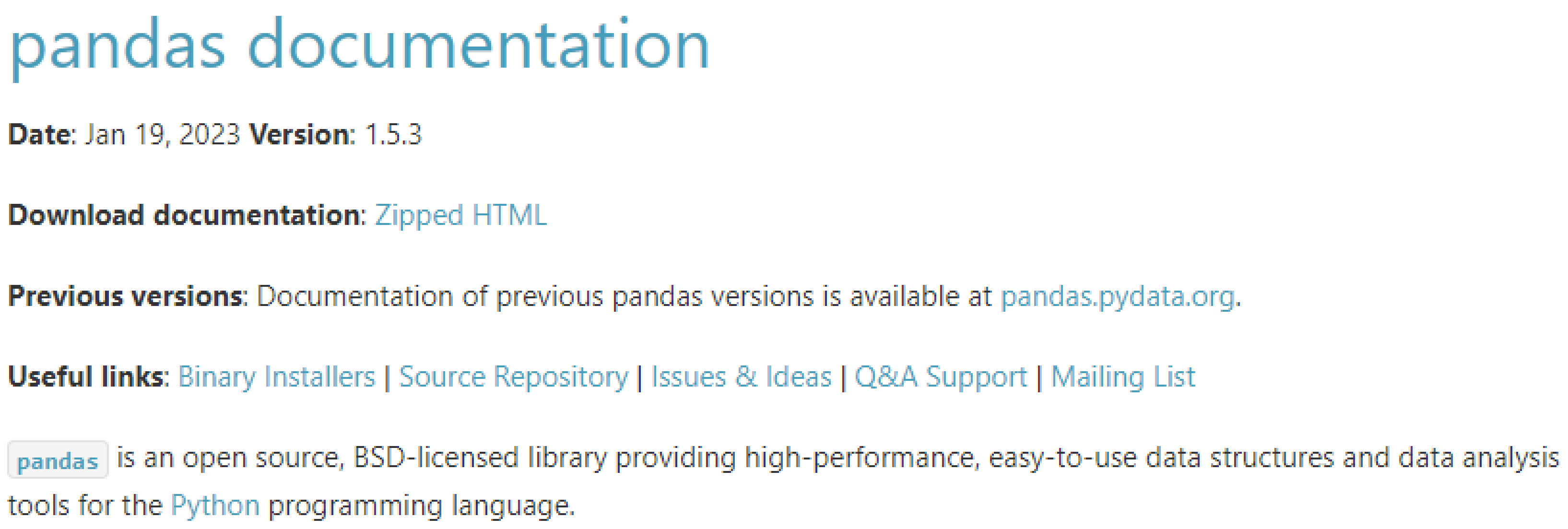


- Scanpy is based on Anndata
- Anndata enables importing and reading matrix files composed of variables and observations
- For scRNAseq, .var should be gene ID and .obs should be cell ID
- Adding variables and observations is available, and as the analysis goes on, the objects are added

**User tip. You can add '.T' at the end of importing command to change var and obs

- Pandas enables to tackle dataframe objects
- Understanding the mechanism behind pd objects are not necessary for the beginners, but if you try to add or delete .var or .obs in dataframe or make new dataframe from the original (c.f. sub-clustering)
- Check the official website for further information

https://pandas.pydata.org/docs/user_guide/10min.html



pandas documentation

Date: Jan 19, 2023 **Version**: 1.5.3

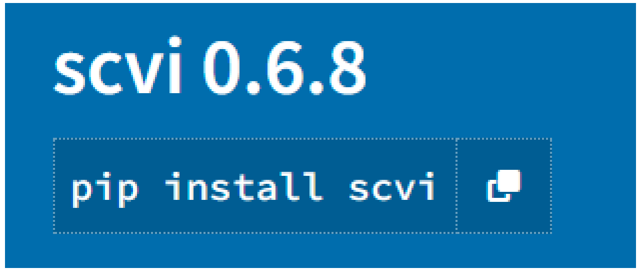
Download documentation: Zipped HTML

Previous versions: Documentation of previous pandas versions is available at pandas.pydata.org.

Useful links: Binary Installers | Source Repository | Issues & Ideas | Q&A Support | Mailing List

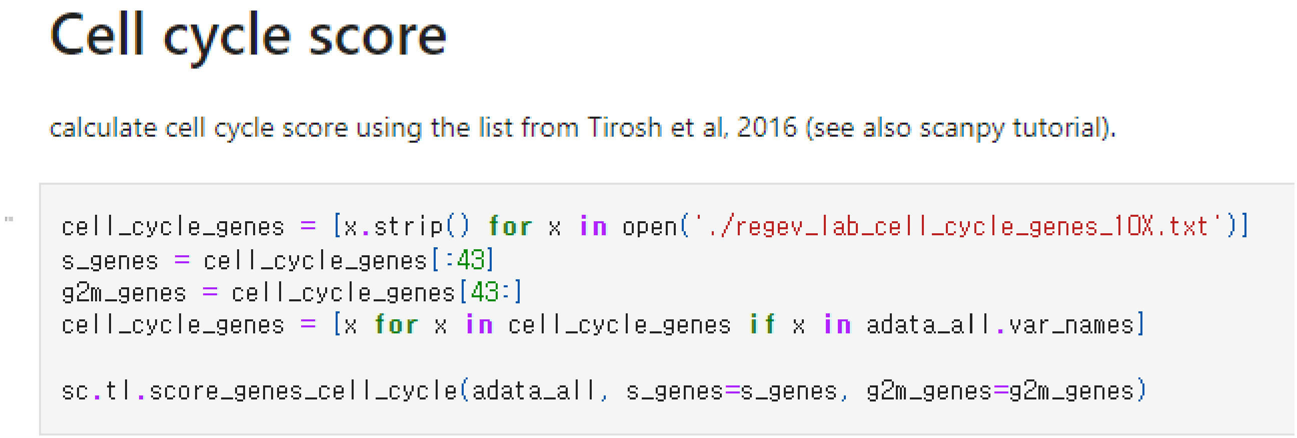
`pandas` is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language.

- SCVI is a package used for doublet removal during data pre-processing step.
- Using SCVI module to remove doublet is very laborious step as the algorithm should be calculated for every dataframe
- Consider using modified version or skipping the doublet removal step if the imported data is from well-known source
- https://github.com/scverse/scvi-tools



scvi 0.6.8

```
pip install scvi
```



Cell cycle score

calculate cell cycle score using the list from Tirosh et al, 2016 (see also scanpy tutorial).

```
cell_cycle_genes = [x.strip() for x in open('./regev_lab_cell_cycle_genes_10X.txt')]
s_genes = cell_cycle_genes[:43]
g2m_genes = cell_cycle_genes[43:]
cell_cycle_genes = [x for x in cell_cycle_genes if x in adata_all.var_names]

sc.tl.score_genes_cell_cycle(adata_all, s_genes=s_genes, g2m_genes=g2m_genes)
```

matplotlib

- Matplotlib provides all kinds of visualization methods in Python
- By using matplotlib objects, you can visualize any kind of plots and control annotations and designs of them
- Learning how to use this fluently should be considered after mastering processing and clustering itself
- Packages like seaborn is based on this package, so installing this in prior will help you to tackle other visualization API

seaborn: statistical data visualization #

https://matplotlib.org/stable/gallery/index.html

https://seaborn.pydata.org/index.html

Part 2. Performing analysis and advanced visualizing & analyzing methods

Using Python for single cell RNA seq data analysis

Seoul National University, Department of Biological Science

Laboratory of Development and Disease Modeling

Jong Hwi Kim

RELEASE DATE: 2/21/23

Step 3. Understanding the work flow of pre-processing

Parsing the data, Reading and trimming the dataset, Cell cycle regression, Doublet removal, Removing mitochondrial and ribosomal gene, Basic PCA and neighbor analysis

Step 4. Calculating PCA/t-SNE/UMAP and clustering

Parsing the data, Reading and trimming the dataset, Removing mitochondrial and ribosomal gene, Basic PCA and neighbor analysis

Step 5. Advanced analysis

Violin plot, Dot plot, Pseudotime analysis, RNA velocity, Heatmap, Subclustering

Part 3. Integrating various dataframe for cross-checking and batch effect removal

Using Python for single cell RNA seq data analysis

Seoul National University, Department of Biological Science

Laboratory of Development and Disease Modeling

Jong Hwi Kim

RELEASE DATE: 2/28/23

Step 6. Integrating datasets

AnnData.concat, BBKNN, SCALEX

Step 7. Batch effect removal for un-biased analysis

Step 8. Applying newly obtained datasets on currently existing plots