

Anaconda-2021.05 Installation

Author: Bart Gerritsen

E-mail: B.H.M.Gerritsen@TUDelft.nl

Intended audience

ICT Staff -- Werkplekbeheer, Digital Exam environment

Overview

This procedure describes how to install the TU Delft software portal Anaconda version;

- the Anaconda base versions is the [Anaconda Individual Edition 2021.05](#). See menu option *Products*
- the installation assumes **MULTIPLE users** (referred to as: "*system installation*")
- as per recommendation by Anaconda, Anaconda users sharing a system installation are grouped and granted read and write permissions for the root folder and folders below
- a single `base` environment (a.k.a. environment `anaconda3`), as created and configured in the official installation procedure by Anaconda, residing in `C:\ProgramData\Anaconda3`
- the installation has to be **multi platform**;
 - for Windows (X86_64)
 - for MacOS
 - for Linux
- the TU Delft stylesheet `tudelft.css` will not be installed in the system install, but distributed separately
- this installation includes, among others, `Spyder`, `VS Code`, `Jupyter Notebook`, `JupyterLab`, and `RStudio`
- this installation includes `scikit-learn`, `TensorFlow`, and `PyTorch`, for Machine Learning
- the Individual Edition is an **all-open-source** installation (no licenses required). This includes the installation of [PyCharm Professional with Anaconda Plugin](#) offered and promoted as part of the official installation, and **free-for-academic** use, a free [Community Edition of the JetBrains Datalore](#) for online editing and execution of Jupyter notebooks, and a trial version of [IBM Watson Studio for Anaconda users](#), that lets you run IBM Notebooks on IBM's Watson Studio for AI and data science, in the cloud.

After this installation, and after adding a few extra packages as per demand by various users in the Faculties, the packaging can commence and the final packaged version can be transferred to the [TU Delft Software Portal](#).

Remark

Students or Staff requiring an installation under *their own user account* can download and installer from the [Anaconda site](#), install, and activate or install the additional packages themselves. If for a minimum installation or just for a Python interpreter, they may want to install `miniconda`.

Resources

1. [Anaconda Individual Edition 2021.05](#)
2. [Introduction Anaconda Individual Edition](#)
3. [Installation procedure for multiple users -- system installation](#)
4. [PyCharm Professional with Anaconda plugin](#)
5. [Packages](#)
6. [Miniconda resources](#)
7. [Hashes](#)

Installation -- what steps need to be undertaken?

- Stage 1: install [Anaconda Individual Edition](#) (details are given below)
- Stage 2: post-install the extra packages as specified by Faculties during the Consultation Round (see below)

Stage 1: Install Anaconda Individual Edition

1. download the **installer** from the [download site](#), for Windows, MacOS, or Linux (see below)
2. install the software as it comes; select **multiple** users (all users) for a system install
3. make a user group `Anaconda-Users` and adjust the file access rights for this group as specified in [the installation procedure](#)
4. verify that the installation has been successful (see below)

Documentation Windows installation procedure

1. [single user](#)
2. [multiple user](#)

MacOS installation procedure

1. [single user](#)
2. [command line install](#)

Linux installation procedure

1. [single user](#)
2. [multiple users](#)

Verification the Installation

See [here](#) how to verify the installation. Also, see [here](#). [This page](#) also gives some pointers.

Troubleshooting

[Problems and troubleshooting](#)

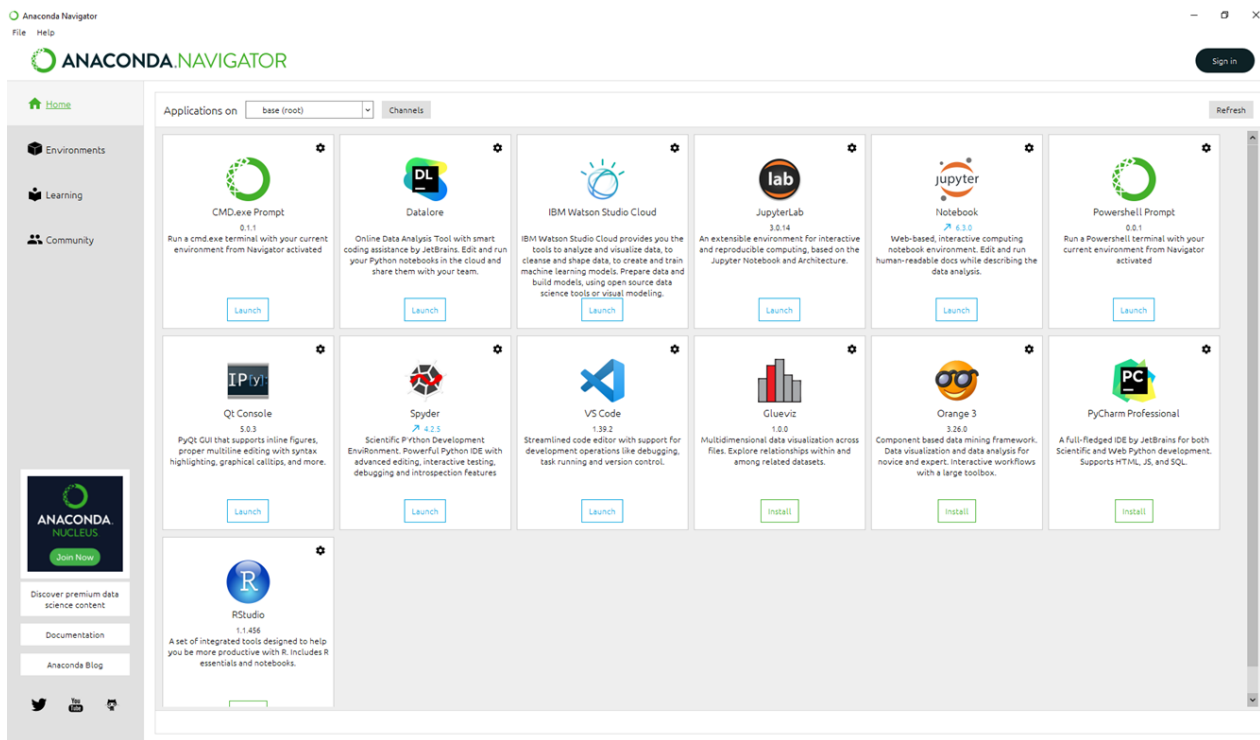
FAQ

[Frequently asked questions](#)

Stage 2:

1. add (post-install) the extra packages to complete the installation as desired by the participants in the Consultation round (see specifications below)

Start by opening **Anaconda-Navigator** (from the Windows Start Menu) and from that environment launch PowerShell-Prompt (see image below).



In the PowerShell thus provided, check if the *base environment* `anaconda3` is selected (as reflected in the prompt; here: `anaconda3`, which is good) and check if `python` and `conda` are in the path, by prompting their versions. The output should show something like:

```
(anaconda3) PS C:\Users\bhmgerritsen> python --version
Python 3.8.8
(anaconda3) PS C:\Users\bhmgerritsen> conda --version
conda 4.10.1
(anaconda3) PS C:\Users\bhmgerritsen>
```

We are now good to go installing the extra packages using `conda` commands on the command line. To install, issue the command in the rightmost column, in the table below, one-by-one:

package name	channel	command
slycot	conda-forge	<code>conda install -c conda-forge slycot</code>
control	conda-forge	<code>conda install -c conda-forge control</code>
nidaqmx-python	conda-forge	<code>conda install -c conda-forge nidaqmx-python</code>
pulp	conda-forge	<code>conda install -c conda-forge pulp</code>

Finally, still in the PowerShell like above, use `pip` to install packages that cannot be installed by `conda`. Usually, `conda` figures out for each of the packages to be installed, which *version* is needed to keep the entire environment sane. Generally, `pip` does not do that. To that end, Anaconda has its own `pip`, to overcome this lack of *version control*:

```
conda list pip
pip 21.1.2 pypi_0 pypi
```

Use it to install the remaining packages that insist on `pip`, like so:

```
python -m pip install salabim
python -m pip install coolprop
```

and:

```
python -m pip install opencv-python
```

and:

```
python -m pip install tsp
```

Verify. Done.

Appendix I

Looking for additional packages (for Stage 2) proceeds as follows. Open a PowerShell Prompt from the Anaconda-Navigator, as shown above. Next:

1. to check if a package is installed, use `conda list <package name>`
2. to search for info on a package on the web, in a specific *channel*, use `conda search -c <channel-name> --info <package name>`

Example: see if `numpy` has already been installed on our system installation:

```
conda list numpy
```

yields the output:

```
(Anaconda3) PS C:\Users\bhmgerriksen> conda list numpy
# packages in environment at C:\Users\bhmgerriksen\anaconda3:
#
# Name                          Version          Build      Channel
numpy                          1.20.1          py38h34a8a5c_0
numpy-base                     1.20.1          py38haf7ebc8_0
numpydoc                       1.1.0           pyhd3eb1b0_1
```

informing us that `numpy 1.20.1` has already been installed in environment `C:\ProgramData\Anaconda3` (i.e. the `base` environment for all users), along with its dependencies.

If not installed, search the web for a package can be done as follows.

Example: search the web (channel Conda-Forge) for an additional package, by the name `slycot` :

```
conda search -c conda-forge --info slycot
```

yielding a long list of output:

backports.shutil_get_terminal_size	1.0.0	pyhd3eb1b0_3	defaults
backports.tempfile	1.0	pyhd3eb1b0_1	defaults
backports.weakref	1.0.post1	py_1	defaults
bcrypt	3.2.0	py38he774522_0	defaults
beautifulsoup4	4.9.3	pyha847dfd_0	defaults
bitarray	1.9.2	py38h2bbff1b_1	defaults
bkcharts	0.2	py38_0	defaults
black	19.10b0	py_0	defaults
blas	1.0	mk1	defaults
bleach	3.3.0	pyhd3eb1b0_0	defaults
blosc	1.21.0	h19a0ad4_0	defaults
bokeh	2.3.2	py38haa95532_0	defaults
boto	2.49.0	py38_0	defaults
bottleneck	1.3.2	py38h2a96729_1	defaults
brotli	1.0.9	ha925a31_2	defaults
brotlipy	0.7.0	py38h2bbff1b_1003	defaults
bzip2	1.0.8	he774522_0	defaults
ca-certificates	2021.4.13	haa95532_1	defaults
cairo	1.16.0	h63a05c6_1001	conda-forge
certifi	2020.12.5	py38haa95532_0	defaults
cffi	1.14.5	py38hcd4344a_0	defaults
chardet	4.0.0	py38haa95532_1003	defaults
charls	2.2.0	h6c2663c_0	defaults
click	7.1.2	pyhd3eb1b0_0	defaults
cloudpickle	1.6.0	py_0	defaults
clyent	1.2.2	py38_1	defaults
colorama	0.4.4	pyhd3eb1b0_0	defaults
comtypes	1.1.9	py38haa95532_1002	defaults
conda	4.10.1	py38haa244fe_0	conda-forge
conda-build	3.21.4	py38haa95532_0	defaults
conda-content-trust	0.1.1	pyhd3eb1b0_0	defaults
conda-env	2.6.0	1	defaults
conda-package-handling	1.7.3	py38h8cc25b3_1	defaults
conda-repo-cli	1.0.4	pyhd3eb1b0_0	defaults
conda-token	0.3.0	pyhd3eb1b0_0	defaults
conda-verify	3.4.2	py_1	defaults
console_shortcut	0.1.1	4	defaults
contextlib2	0.6.0.post1	py_0	defaults
cryptography	3.4.7	py38h71e12ea_0	defaults
curl	7.71.1	h2a8f88b_1	defaults
cycler	0.10.0	py38_0	defaults
cython	0.29.23	py38hd77b12b_0	defaults
cytoolz	0.11.0	py38he774522_0	defaults
dask	2021.4.0	pyhd3eb1b0_0	defaults
dask-core	2021.4.0	pyhd3eb1b0_0	defaults
decorator	5.0.6	pyhd3eb1b0_0	defaults
defusedxml	0.7.1	pyhd3eb1b0_0	defaults
diff-match-patch	20200713	py_0	defaults
distributed	2021.4.0	py38haa95532_0	defaults
docutils	0.17	py38haa95532_1	defaults
entrypoints	0.3	py38_0	defaults
et_xmlfile	1.0.1	py_1001	defaults
fastcache	1.1.0	py38he774522_0	defaults
filelock	3.0.12	pyhd3eb1b0_1	defaults
flake8	3.9.0	pyhd3eb1b0_0	defaults
flask	1.1.2	pyhd3eb1b0_0	defaults
freetype	2.10.4	hd328e21_0	defaults
fsspec	0.9.0	pyhd3eb1b0_0	defaults
future	0.18.2	py38_1	defaults
fuzzywuzzy	0.18.0	pyhd8ed1ab_0	conda-forge
get_terminal_size	1.0.0	h38e98db_0	defaults
gevent	21.1.2	py38h2bbff1b_1	defaults
giflib	5.2.1	h62dcd97_0	defaults
glob2	0.7	pyhd3eb1b0_0	defaults
glpk	4.65	h8ffe710_1004	conda-forge
greenlet	1.0.0	py38hd77b12b_2	defaults
h5py	2.10.0	py38h5e291fa_0	defaults
hdf5	1.10.4	h7ebc959_0	defaults
heapdict	1.0.1	py_0	defaults
html5lib	1.1	py_0	defaults
icc_rt	2019.0.0	h0cc432a_1	defaults
icu	58.2	ha925a31_3	defaults
idna	2.10	pyhd3eb1b0_0	defaults
igraph	0.9.2	h29cbd77_0	conda-forge

imagecodecs	2021.3.31	py38h5da4933_0	defaults
imageio	2.9.0	pyhd3eb1b0_0	defaults
imagesize	1.2.0	pyhd3eb1b0_0	defaults
importlib-metadata	3.10.0	py38haa95532_0	defaults
importlib_metadata	3.10.0	hd3eb1b0_0	defaults
iniconfig	1.1.1	pyhd3eb1b0_0	defaults
intel-openmp	2021.2.0	haa95532_616	defaults
intervaltree	3.1.0	py_0	defaults
ipykernel	5.3.4	py38h5ca1d4c_0	defaults
ipython	7.22.0	py38hd4e2768_0	defaults
ipython_genutils	0.2.0	pyhd3eb1b0_1	defaults
ipywidgets	7.6.3	pyhd3eb1b0_1	defaults
isort	5.8.0	pyhd3eb1b0_0	defaults
itsdangerous	1.1.0	pyhd3eb1b0_0	defaults
jdcalf	1.4.1	py_0	defaults
jedi	0.17.2	py38haa95532_1	defaults
jinja2	2.11.3	pyhd3eb1b0_0	defaults
joblib	1.0.1	pyhd3eb1b0_0	defaults
jpeg	9b	hb83a4c4_2	defaults
json5	0.9.5	py_0	defaults
jsonschema	3.2.0	py_2	defaults
jupyter	1.0.0	py38_7	defaults
jupyter-packaging	0.7.12	pyhd3eb1b0_0	defaults
jupyter_client	6.1.12	pyhd3eb1b0_0	defaults
jupyter_console	6.4.0	pyhd3eb1b0_0	defaults
jupyter_contrib_core	0.3.3	py_2	conda-forge
jupyter_contrib_nbextensions	0.5.1	pyhd8ed1ab_2	conda-forge
jupyter_core	4.7.1	py38haa95532_0	defaults
jupyter_highlight_selected_word	0.2.0	py38haa244fe_1002	conda-forge
jupyter_latex_envs	1.4.6	pyhd8ed1ab_1002	conda-forge
jupyter_nbextensions_configurator	0.4.1	py38haa244fe_2	conda-forge
jupyter_server	1.4.1	py38haa95532_0	defaults
jupyterlab	3.0.14	pyhd3eb1b0_1	defaults
jupyterlab_pygments	0.1.2	py_0	defaults
jupyterlab_server	2.4.0	pyhd3eb1b0_0	defaults
jupyterlab_widgets	1.0.0	pyhd3eb1b0_1	defaults
keyring	22.3.0	py38haa95532_0	defaults
kiwisolver	1.3.1	py38hd77b12b_0	defaults
krb5	1.18.2	hc04afaa_0	defaults
lazy-object-proxy	1.6.0	py38h2bbff1b_0	defaults
lcms2	2.12	h83e58a3_0	defaults
lerc	2.2.1	hd77b12b_0	defaults
libaec	1.0.4	h33f27b4_1	defaults
libarchive	3.4.2	h5e25573_0	defaults
libblas	3.9.0	1_h8933c1f_netlib	conda-forge
libcbblas	3.9.0	5_hd5c7e75_netlib	conda-forge
libcurl	7.71.1	h2a8f88b_1	defaults
libdeflate	1.7	h2bbff1b_5	defaults
libiconv	1.15	h1df5818_7	defaults
liblapack	3.9.0	5_hd5c7e75_netlib	conda-forge
liblapacke	3.9.0	5_hd5c7e75_netlib	conda-forge
liblief	0.10.1	ha925a31_0	defaults
libpng	1.6.37	h2a8f88b_0	defaults
libsodium	1.0.18	h62dcd97_0	defaults
libspatialindex	1.9.3	h6c2663c_0	defaults
libssh2	1.9.0	h7a1dbc1_1	defaults
libtiff	4.2.0	hd0e1b90_0	defaults
libxml2	2.9.10	hb89e7f3_3	defaults
libxslt	1.1.34	he774522_0	defaults
libzopfli	1.0.3	ha925a31_0	defaults
llvmlite	0.36.0	py38h34b8924_4	defaults
loket	0.2.1	py38haa95532_1	defaults
lxml	4.6.3	py38h9b66d53_0	defaults
lz4-c	1.9.3	h2bbff1b_0	defaults
lzo	2.10	he774522_2	defaults
m2w64-gcc-libgfortran	5.3.0	6	defaults
m2w64-gcc-libs	5.3.0	7	defaults
m2w64-gcc-libs-core	5.3.0	7	defaults
m2w64-gmp	6.1.0	2	defaults
m2w64-libwinpthread-git	5.0.0.4634.697f757	2	defaults
mako	1.1.4	pyh44b312d_0	conda-forge
markupsafe	1.1.1	py38he774522_0	defaults
matplotlib	3.3.4	py38haa95532_0	defaults
matplotlib-base	3.3.4	py38h49ac443_0	defaults

mccabe	0.6.1	py38_1	defaults
menuinst	1.4.16	py38he774522_1	defaults
mistune	0.8.4	py38he774522_1000	defaults
mkl	2021.2.0	haa95532_296	defaults
mkl-service	2.3.0	py38h2bbff1b_1	defaults
mkl_fft	1.3.0	py38h277e83a_2	defaults
mkl_random	1.2.1	py38hf11a4ad_2	defaults
mock	4.0.3	pyhd3eb1b0_0	defaults
more-itertools	8.7.0	pyhd3eb1b0_0	defaults
mpir	3.0.0	he025d50_1002	conda-forge
mpmath	1.2.1	py38haa95532_0	defaults
msgpack-python	1.0.2	py38h59b6b97_1	defaults
msys2-conda-epoch	20160418	1	defaults
multipledispatch	0.6.0	py38_0	defaults
mypy_extensions	0.4.3	py38_0	defaults
navigator-updater	0.2.1	py38_0	defaults
nbclassic	0.2.6	pyhd3eb1b0_0	defaults
nbclient	0.5.3	pyhd3eb1b0_0	defaults
nbconvert	6.0.7	py38_0	defaults
nbformat	5.1.3	pyhd3eb1b0_0	defaults
nbgrader	0.6.1	py38haa244fe_1	conda-forge
nest-asyncio	1.5.1	pyhd3eb1b0_0	defaults
networkx	2.5	py_0	defaults
nlTK	3.6.1	pyhd3eb1b0_0	defaults
nose	1.3.7	pyhd3eb1b0_1006	defaults
notebook	6.3.0	py38haa95532_0	defaults
numba	0.53.1	py38hf11a4ad_0	defaults
numexpr	2.7.3	py38hb80d3ca_1	defaults
numpy	1.20.1	py38h34a8a5c_0	defaults
numpy-base	1.20.1	py38haf7ebc8_0	defaults
numpydoc	1.1.0	pyhd3eb1b0_1	defaults
olefile	0.46	py_0	defaults
openjpeg	2.3.0	h5ec785f_1	defaults
openpyxl	3.0.7	pyhd3eb1b0_0	defaults
openssl	1.1.1k	h2bbff1b_0	defaults
orderedset	2.0.3	py38h294d835_4	conda-forge
packaging	20.9	pyhd3eb1b0_0	defaults
pandas	1.2.4	py38hd77b12b_0	defaults
pandoc	2.12	haa95532_0	defaults
pandocfilters	1.4.3	py38haa95532_1	defaults
paramiko	2.7.2	py_0	defaults
parso	0.7.0	py_0	defaults
partd	1.2.0	pyhd3eb1b0_0	defaults
path	15.1.2	py38haa95532_0	defaults
path.py	12.5.0	0	defaults
pathlib2	2.3.5	py38haa95532_2	defaults
pathspec	0.7.0	py_0	defaults
patsy	0.5.1	py38_0	defaults
pep8	1.7.1	py38_0	defaults
pexpect	4.8.0	pyhd3eb1b0_3	defaults
pickleshare	0.7.5	pyhd3eb1b0_1003	defaults
pillow	8.2.0	py38h4fa10fc_0	defaults
pip	21.1.2	pypi_0	pypi
pixman	0.38.0	hfa6e2cd_1003	conda-forge
pkginfo	1.7.0	py38haa95532_0	defaults
pluggy	0.13.1	py38haa95532_0	defaults
ply	3.11	py38_0	defaults
powershell_shortcut	0.0.1	3	defaults
prometheus_client	0.10.1	pyhd3eb1b0_0	defaults
prompt-toolkit	3.0.17	pyh06a4308_0	defaults
prompt_toolkit	3.0.17	hd3eb1b0_0	defaults
psutil	5.8.0	py38h2bbff1b_1	defaults
ptyprocess	0.7.0	pyhd3eb1b0_2	defaults
py	1.10.0	pyhd3eb1b0_0	defaults
py-lief	0.10.1	py38ha925a31_0	defaults
pycairo	1.20.0	py38h979ce04_1	conda-forge
pycodestyle	2.6.0	pyhd3eb1b0_0	defaults
pycosat	0.6.3	py38h2bbff1b_0	defaults
pycparser	2.20	py_2	defaults
pycurl	7.43.0.6	py38h7a1dbc1_0	defaults
pydocstyle	6.0.0	pyhd3eb1b0_0	defaults
pyerfa	1.7.3	py38h2bbff1b_0	defaults
pyflakes	2.2.0	pyhd3eb1b0_0	defaults
pygments	2.8.1	pyhd3eb1b0_0	defaults

pylint	2.7.4	py38haa95532_1	defaults
pyls-black	0.4.6	hd3eb1b0_0	defaults
pyls-spyder	0.3.2	pyhd3eb1b0_0	defaults
pynacl	1.4.0	py38h62dcd97_1	defaults
pyodbc	4.0.30	py38ha925a31_0	defaults
pyopenssl	20.0.1	pyhd3eb1b0_1	defaults
pyarsing	2.4.7	pyhd3eb1b0_0	defaults
pyqt	5.9.2	py38ha925a31_4	defaults
pyreadline	2.1	py38_1	defaults
pyrsistent	0.17.3	py38he774522_0	defaults
pysocks	1.7.1	py38haa95532_0	defaults
pytables	3.6.1	py38ha5be198_0	defaults
pytest	6.2.3	py38haa95532_2	defaults
python	3.8.8	hdbf39b2_5	defaults
python-dateutil	2.8.1	pyhd3eb1b0_0	defaults
python-editor	1.0.4	py_0	conda-forge
python-igraph	0.9.1	py38hc48d4f8_0	conda-forge
python-jsonrpc-server	0.4.0	py_0	defaults
python-language-server	0.36.2	pyhd3eb1b0_0	defaults
python-levenshtein	0.12.2	py38h294d835_0	conda-forge
python-libarchive-c	2.9	pyhd3eb1b0_1	defaults
python_abi	3.8	1_cp38	conda-forge
pytz	2021.1	pyhd3eb1b0_0	defaults
pywavelets	1.1.1	py38he774522_2	defaults
pywin32	227	py38he774522_1	defaults
pywin32-ctypes	0.2.0	py38_1000	defaults
pywinpty	0.5.7	py38_0	defaults
pyyaml	5.4.1	py38h2bbff1b_1	defaults
pymzq	20.0.0	py38hd77b12b_1	defaults
qdarkstyle	2.8.1	py_0	defaults
qt	5.9.7	vc14h73c81de_0	defaults
qtawesome	1.0.2	pyhd3eb1b0_0	defaults
qtconsole	5.0.3	pyhd3eb1b0_0	defaults
qtpy	1.9.0	py_0	defaults
qutip	4.6.1	py38h2f20550_0	conda-forge
regex	2021.4.4	py38h2bbff1b_0	defaults
requests	2.25.1	pyhd3eb1b0_0	defaults
rope	0.18.0	py_0	defaults
rtree	0.9.7	py38h2eaa2aa_1	defaults
ruamel_yaml	0.15.100	py38h2bbff1b_0	defaults
scikit-image	0.18.1	py38hf11a4ad_0	defaults
scikit-learn	0.24.1	py38hf11a4ad_0	defaults
scipy	1.6.2	py38h66253e8_1	defaults
seaborn	0.11.1	pyhd3eb1b0_0	defaults
send2trash	1.5.0	pyhd3eb1b0_1	defaults
setuptools	52.0.0	py38haa95532_0	defaults
simplegeneric	0.8.1	py38_2	defaults
singledispatch	3.6.1	pyhd3eb1b0_1001	defaults
sip	4.19.13	py38ha925a31_0	defaults
six	1.15.0	py38haa95532_0	defaults
snappy	1.1.8	h33f27b4_0	defaults
sniffio	1.2.0	py38haa95532_1	defaults
snowballstemmer	2.1.0	pyhd3eb1b0_0	defaults
sortedcollections	2.1.0	pyhd3eb1b0_0	defaults
sortedcontainers	2.3.0	pyhd3eb1b0_0	defaults
soupsieve	2.2.1	pyhd3eb1b0_0	defaults
sphinx	4.0.1	pyhd3eb1b0_0	defaults
sphinxcontrib	1.0	py38_1	defaults
sphinxcontrib-applehelp	1.0.2	pyhd3eb1b0_0	defaults
sphinxcontrib-devhelp	1.0.2	pyhd3eb1b0_0	defaults
sphinxcontrib-htmlhelp	1.0.3	pyhd3eb1b0_0	defaults
sphinxcontrib-jsmath	1.0.1	pyhd3eb1b0_0	defaults
sphinxcontrib-qthelp	1.0.3	pyhd3eb1b0_0	defaults
sphinxcontrib-serializinghtml	1.1.4	pyhd3eb1b0_0	defaults
sphinxcontrib-websupport	1.2.4	py_0	defaults
spyder	4.2.5	py38haa95532_0	defaults
spyder-kernels	1.10.2	py38haa95532_0	defaults
sqlalchemy	1.4.7	py38h2bbff1b_0	defaults
sqlite	3.35.4	h2bbff1b_0	defaults
statsmodels	0.12.2	py38h2bbff1b_0	defaults
suitesparse	5.4.0	h5d0cbe0_1	conda-forge
sympy	1.8	py38haa95532_0	defaults
tbb	2020.3	h74a9793_0	defaults
tblib	1.7.0	py_0	defaults

terminado	0.9.4	py38haa95532_0	defaults
testpath	0.4.4	pyhd3eb1b0_0	defaults
textdistance	4.2.1	pyhd3eb1b0_0	defaults
texttable	1.6.3	pyh9f0ad1d_0	conda-forge
threadpoolctl	2.1.0	pyh5ca1d4c_0	defaults
three-merge	0.1.1	pyhd3eb1b0_0	defaults
tiffiffle	2021.4.8	pyhd3eb1b0_2	defaults
tk	8.6.10	he774522_0	defaults
toml	0.10.2	pyhd3eb1b0_0	defaults
toolz	0.11.1	pyhd3eb1b0_0	defaults
tornado	6.1	py38h2bbff1b_0	defaults
tqdm	4.59.0	pyhd3eb1b0_1	defaults
traitlets	5.0.5	pyhd3eb1b0_0	defaults
typed-ast	1.4.2	py38h2bbff1b_1	defaults
typing_extensions	3.7.4.3	pyha847dfd_0	defaults
ujson	4.0.2	py38hd77b12b_0	defaults
unicodcsv	0.14.1	py38_0	defaults
urllib3	1.26.4	pyhd3eb1b0_0	defaults
vc	14.2	h21ff451_1	defaults
vs2015_runtime	14.27.29016	h5e58377_2	defaults
watchdog	1.0.2	py38haa95532_1	defaults
wcwidth	0.2.5	py_0	defaults
webencodings	0.5.1	py38_1	defaults
werkzeug	1.0.1	pyhd3eb1b0_0	defaults
wheel	0.36.2	pyhd3eb1b0_0	defaults
widgetsnextension	3.5.1	py38_0	defaults
win_inet_pton	1.1.0	py38haa95532_0	defaults
win_unicode_console	0.5	py38_0	defaults
wincertstore	0.2	py38_0	defaults
winpty	0.4.3	4	defaults
wrapt	1.12.1	py38he774522_1	defaults
xlrd	2.0.1	pyhd3eb1b0_0	defaults
xlswriter	1.3.8	pyhd3eb1b0_0	defaults
xlwings	0.23.0	py38haa95532_0	defaults
xlwt	1.3.0	py38_0	defaults
xmltodict	0.12.0	py_0	defaults
xz	5.2.5	h62dcd97_0	defaults
yaml	0.2.5	he774522_0	defaults
yapf	0.31.0	pyhd3eb1b0_0	defaults
zeromq	4.3.3	ha925a31_3	defaults
zfp	0.5.5	hd77b12b_6	defaults
zict	2.0.0	pyhd3eb1b0_0	defaults
zipp	3.4.1	pyhd3eb1b0_0	defaults
zlib	1.2.11	h62dcd97_4	defaults
zope	1.0	py38_1	defaults
zope.event	4.5.0	py38_0	defaults
zope.interface	5.3.0	py38h2bbff1b_0	defaults
zstd	1.4.5	h04227a9_0	defaults