

Data Analysis

```
In [1]: 1 # Modules
        2 print(help("modules"))
```

Please wait a moment while I gather a list of all available modules...

```
C:\Users\Alekhya\Anaconda3\lib\site-packages\IPython\kernel\__init__.py:13: ShimWarning: The `IPython.kernel` package has been deprecated since IPython 4.0. You should import from ipykernel or jupyter_client instead.
```

```
"You should import from ipykernel or jupyter_client instead.", ShimWarning)
WARNING: AstropyDeprecationWarning: astropy.utils.compat.futures is now deprecated - use concurrent.futures instead [astropy.utils.compat.futures]
```

```
C:\Users\Alekhya\Anaconda3\lib\site-packages\dask\config.py:168: YAMLLoadWarning: calling yaml.load() without Loader=... is deprecated, as the default Loader is unsafe. Please read https://msg.pyyaml.org/load (https://msg.pyyaml.org/load) for full details.
```

```
data = yaml.load(f.read()) or {}
```

```
C:\Users\Alekhya\Anaconda3\lib\site-packages\distributed\config.py:20: YAMLLoadWarning: calling yaml.load() without Loader=... is deprecated, as the default Loader is unsafe. Please read https://msg.pyyaml.org/load (https://msg.pyyaml.org/load) for full details.
```

```
defaults = yaml.load(f)
```

```
C:\Users\Alekhya\Anaconda3\lib\site-packages\nltk\twitter\__init__.py:22: UserWarning: The twython library has not been installed. Some functionality from the twitter package will not be available.
```

```
"The twython library has not been installed. "
```

```
DEBUG:pip._internal.vcs.versioncontrol:Registered VCS backend: bzt
```

```
DEBUG:pip._internal.vcs.versioncontrol:Registered VCS backend: git
```

```
DEBUG:pip._internal.vcs.versioncontrol:Registered VCS backend: hg
```

```
DEBUG:pip._internal.vcs.versioncontrol:Registered VCS backend: svn
```

```
C:\Users\Alekhya\Anaconda3\lib\site-packages\skimage\novice\__init__.py:103: UserWarning: The `skimage.novice` module was deprecated in version 0.14. It will be removed in 0.16.
```

```
warnings.warn("The `skimage.novice` module was deprecated in version 0.14. "
```

```
C:\Users\Alekhya\Anaconda3\lib\site-packages\skimage\viewer\utils\core.py:10: UserWarning: Recommended matplotlib backend is `Agg` for full skimage.viewer functionality.
```

```
warn("Recommended matplotlib backend is `Agg` for full "
```

```
C:\Users\Alekhya\Anaconda3\lib\site-packages\sphinx\websupport\__init__.py:25: RemovedInSphinx20Warning: sphinx.websupport module is now provided as sphinxcontrib-websupport. sphinx.websupport will be removed at Sphinx-2.0. Please use the package instead.
```

```
RemovedInSphinx20Warning)
```

```
C:\Users\Alekhya\Anaconda3\lib\site-packages\qtawesome\iconic_font.py:276: UserWarning: You need to have a running QApplication to use QtAwesome!
```

```
warnings.warn("You need to have a running "
```

```
C:\Users\Alekhya\Anaconda3\lib\pkgutil.py:107: VisibleDeprecationWarning: zmq.eventloop.minitornado is deprecated in pyzmq 14.0 and will be removed.
```

```
Install tornado itself to use zmq with the tornado IOloop.
```

```
yield from walk_packages(path, info.name+'.', onerror)
```

Crypto

bz2

menuinst

sortedcollections

Cython

cProfile

mimetypes

sortedcontainers

IPython	calendar	mistune	soupsieve
OpenSSL	certifi	mk1	sphinx
PIL	cffi	mk1_fft	sphinxcontrib
PyQt5	cgi	mk1_random	spyder
__future__	cgitb	mmap	spyder_breakpoints
_abc	chardet	mmapfile	spyder_io_dcm
_ast	chunk	mmsystem	spyder_io_hdf5
_asyncio	click	modulefinder	spyder_kernels
_bisect	cloudpickle	more_itertools	spyder_profiler
_blake2	clyent	mpmath	spyder_pylint
_bootlocale	cmath	msgpack	sqlalchemy
_bz2	cmd	msilib	sqlite3
_cffi_backend	code	msvcrt	sre_compile
_codecs	codecs	multipledispatch	sre_constants
_codecs_cn	codeop	multiprocessing	sre_parse
_codecs_hk	collections	navigator_updater	ssl
_codecs_iso2022	colorama	nbconvert	sspi
_codecs_jp	colorsys	nbformat	sspicon
_codecs_kr	commctrl	netbios	stat
_codecs_tw	compileall	netrc	statistics
_collections	comtypes	networkx	statsmodels
_collections_abc	concurrent	nlk	storemagic
_compat_pickle	conda	nntplib	string
_compression	conda_build	nose	stringprep
_contextvars	conda_env	notebook	struct
_csv	conda_package_handling	nt	subprocess
_ctypes	conda_verify	ntpath	sunau
_ctypes_test	configparser	ntsecuritycon	symbol
_datetime	constantly	nturl2path	sympy
_decimal	contextlib	numba	sympyprinting
_dummy_thread	contextlib2	numbers	symtable
_elementtree	contextvars	numexpr	sys
_functools	copy	numpy	sysconfig
_hashlib	copyreg	numpydoc	tables
_heapq	crypt	odbc	tabnanny
_imp	cryptography	odo	tarfile
_io	csv	olefile	tblib
_json	ctypes	opcode	telnetlib
_locale	curl	openpyxl	tempfile
_lsprof	curses	operator	terminado
_lzma	cwp	optparse	test
_markupbase	cycler	os	test_data
_md5	cython	packaging	test_path
_msi	cythonmagic	pandas	test_pycosat
_multibytecodec	cytoolz	pandocfilters	testpath
_multiprocessing	dask	parser	tests
_nsis	dataclasses	parso	textwrap
_opcode	datashape	partd	this
_operator	datetime	past	threading
_osx_support	dateutil	path	time
_overlapped	dbi	pathlib	timeit
_pickle	dbm	pathlib2	timer
_py_abc	dde	patsy	tkinter
_pydecimal	decimal	pdb	tlz
_pyio	decorator	pep8	token
_pylief	defusedxml	perfmon	tokenize
_pysistent_version	difflib	pickle	toolz

_pytest	dis	pickleshare	tornado
_queue	distributed	pickletools	tqdm
_random	distutils	pip	trace
_sha1	doctest	pipes	traceback
_sha256	docutils	pkg_resources	tracemalloc
_sha3	dummy_threading	pkginfo	traitlets
_sha512	easy_install	pkgutil	tty
_signal	email	platform	turtle
_sitebuiltins	encodings	plistlib	turtledemo
_socket	ensurepip	pluggy	twisted
_sqlite3	entrypoints	ply	types
_sre	enum	poplib	typing
_ssl	errno	posixpath	unicodcsv
_stat	et_xmlfile	pprint	unicodedata
_string	fastcache	profile	unittest
_strptime	faulthandler	prometheus_client	urllib
_struct	filecmp	prompt_toolkit	urllib3
_symtable	fileinput	pstats	uu
_system_path	filelock	psutil	uuid
_testbuffer	flask	pty	venv
_testcapi	flask_cors	pvectorc	warnings
_testconsole	fnmatch	py	wave
_testimportmultiple	formatter	py_compile	wcwidth
_testmultiphase	fractions	pyasn1	weakref
_thread	ftplib	pyasn1_modules	webbrowser
_threading_local	functools	pycbr	webencodings
_tkinter	future	pycodestyle	werkzeug
_tracemalloc	gc	pycosat	wheel
_warnings	genericpath	pycparser	widgetsnextension
_weakref	getopt	pycurl	win2kras
_weakrefset	getpass	pydoc	win32api
_win32sysloader	gettext	pydoc_data	win32clipboard
_winapi	gevent	pydotplus	win32com
_winxptheme	glob	pyexpat	win32con
_yaml	glob2	pyflakes	win32console
abc	graphviz	pygments	win32cred
adodbapi	greenlet	pylab	win32crypt
afxres	gzip	pylint	win32cryptcon
aifc	h5py	pyodbc	win32event
alabaster	hamcrest	pyparsing	win32evtlog
anaconda_navigator	hashlib	pyreadline	win32evtlogutil
anaconda_project	heapdict	pysistent	win32file
antigravity	heapq	pytest	win32gui
appdirs	hmac	pytest_arraydiff	win32gui_struct
argparse	html	pytest_doctestplus	win32help
array	html5lib	pytest_openfiles	win32inet
asn1crypto	http	pytest_remotedata	win32inetcon
ast	hyperlink	pythoncom	win32job
astroid	idlelib	pytz	win32lz
astropy	idna	pywin	win32net
asynchat	imageio	pywin32_testutil	win32netcon
asyncio	imagesize	pywintypes	win32pdh
asyncore	imaplib	pywt	win32pdhquery
atexit	imghdr	pyximport	win32pdhutil
atomicwrites	imp	qtawesome	win32pipe
attr	importlib	qtconsole	win32print

audioop	importlib_metadata	qtpy	win32process
automat	incremental	queue	win32profile
autoreload	inspect	quopri	win32ras
babel	io	random	win32rcparser
backcall	ipaddress	rasutil	win32security
backports	ipykernel	re	win32service
base64	ipykernel_launcher	readline	win32serviceutil
bcrypt	ipython_genutils	regcheck	win32timezone
bdb	ipywidgets	regutil	win32trace
binascii	isapi	reprlib	win32traceutil
binhex	isort	requests	win32transaction
binstar_client	isympy	rlcompleter	win32ts
bisect	itertools	rmagic	win32ui
bitarray	itsdangerous	rope	win32uiOLE
bkcharts	jdcal	ruamel_yaml	win32verstamp
blaze	jedi	run	win32wnet
bleach	jinja2	runpy	win_inet_pton
bokeh	json	sched	win_unicode_conso
le			
boto	jsonschema	scipy	wincertstore
bottleneck	jupyter	scripts	winerror
brain_argparse	jupyter_client	seaborn	winioctlcon
brain_attrs	jupyter_console	secrets	winnt
brain_builtin_inference	jupyter_core	select	winperf
brain_collections	jupyterlab	selectors	winpty
brain_curses	jupyterlab_launcher	send2trash	winreg
brain_dateutil	jupyterlab_server	service_identity	winsound
brain_fstrings	keyring	servicemanager	winxpgui
brain_func tools	keyword	setuptools	winxptheme
brain_gi	kiwisolver	shelve	wrapt
brain_hashlib	lazy_object_proxy	shlex	wsgiref
brain_http	lib2to3	shutil	xdr lib
brain_io	libarchive	signal	xldr
brain_mechanize	libfuturize	simplegeneric	xlswriter
brain_multiprocessing	libpasteurize	singledispatch	xlwings
brain_namedtuple_enum	lief	singledispatch_helpers	xlwt
brain_nose	linecache	sip	xml
brain_numpy	llvmlite	sipconfig	xmlrpc
brain_pkg_resources	locale	sipdistutils	xxsubtype
brain_pytest	locket	site	yaml
brain_qt	logging	six	zict
brain_random	lxml	skimage	zipapp
brain_re	lzma	sklearn	zipfile
brain_six	macpath	smtpd	zipimport
brain_ssl	mailbox	smtplib	zipp
brain_subprocess	mailcap	sndhdr	zlib
brain_threading	markupsafe	snowballstemmer	zmq
brain_typing	marshal	socket	zope
brain_uuid	math	socketserver	
bs4	matplotlib	socks	
builtins	mccabe	sockshandler	

Enter any module name to get more help. Or, type "modules spam" to search for modules whose name or summary contain the string "spam".

None

```
DEBUG:matplotlib.pyplot:Loaded backend module://ipykernel.pylab.backend_inlin  
e version unknown.
```



In [2]:

```
1 # packages
2 pip list
```

Package	Version
-----	-----
alabaster	0.7.12
anaconda-client	1.7.2
anaconda-navigator	1.9.7
anaconda-project	0.8.2
appdirs	1.4.4
asn1crypto	0.24.0
astroid	2.2.5
astropy	3.1.2
atomicwrites	1.3.0
attrs	19.1.0
Automat	20.2.0
Babel	2.6.0
backcall	0.1.0
backports.os	0.1.1
backports.shutil-get-terminal-size	1.0.0
bcrypt	3.1.6
beautifulsoup4	4.7.1
bitarray	0.8.3
bkcharts	0.2
blaze	0.11.3
bleach	3.1.0
bokeh	1.0.4
boto	2.49.0
Bottleneck	1.2.1
certifi	2020.12.5
cffi	1.12.2
chardet	3.0.4
Click	7.0
cloudpickle	0.8.0
clyent	1.2.2
colorama	0.4.1
comtypes	1.1.7
conda	4.10.1
conda-build	3.17.8
conda-package-handling	1.3.11
conda-verify	3.1.1
constantly	15.1.0
contextlib2	0.5.5
cryptography	2.6.1
cycler	0.10.0
Cython	0.29.6
cytoolz	0.9.0.1
dask	1.1.4
datashape	0.5.4
decorator	4.4.0
defusedxml	0.5.0
distributed	1.26.0
docutils	0.14
entrypoints	0.3
et-xmlfile	1.0.1
fastcache	1.0.2

filelock	3.0.10
Flask	1.0.2
Flask-Cors	3.0.10
future	0.17.1
gevent	1.4.0
glob2	0.6
graphviz	0.14.1
greenlet	0.4.15
h5py	2.9.0
heapdict	1.0.0
html5lib	1.0.1
hyperlink	21.0.0
idna	2.8
imageio	2.5.0
imagesize	1.1.0
importlib-metadata	0.0.0
incremental	17.5.0
ipykernel	5.1.0
ipython	7.4.0
ipython-genutils	0.2.0
ipywidgets	7.4.2
isort	4.3.16
itsdangerous	1.1.0
jdcal	1.4
jedi	0.13.3
Jinja2	2.10
jsonschema	3.0.1
jupyter	1.0.0
jupyter-client	5.2.4
jupyter-console	6.0.0
jupyter-core	4.4.0
jupyterlab	0.35.4
jupyterlab-launcher	0.13.1
jupyterlab-server	0.2.0
keyring	18.0.0
kiwisolver	1.0.1
lazy-object-proxy	1.3.1
libarchive-c	2.8
llvmlite	0.28.0
loket	0.2.0
lxml	4.3.2
MarkupSafe	1.1.1
matplotlib	3.0.3
mccabe	0.6.1
menuinst	1.4.16
mistune	0.8.4
mkl-fft	1.0.10
mkl-random	1.0.2
more-itertools	6.0.0
mpmath	1.1.0
msgpack	0.6.1
multipledispatch	0.6.0
navigator-updater	0.2.1
nbconvert	5.4.1
nbformat	4.4.0
networkx	2.2
nlTK	3.4

nose	1.3.7
notebook	5.7.8
numba	0.43.1
numexpr	2.6.9
numpy	1.16.2
numpydoc	0.8.0
odo	0.5.1
olefile	0.46
openpyxl	2.6.1
packaging	19.0
pandas	0.24.2
pandocfilters	1.4.2
parso	0.3.4
partd	0.3.10
path.py	11.5.0
pathlib2	2.3.3
patsy	0.5.1
pep8	1.7.1
pickleshare	0.7.5
Pillow	5.4.1
pip	20.2.3
pkginfo	1.5.0.1
pluggy	0.9.0
ply	3.11
prometheus-client	0.6.0
prompt-toolkit	2.0.9
psutil	5.6.1
py	1.8.0
pyasn1	0.4.8
pyasn1-modules	0.2.8
pycodestyle	2.5.0
pycosat	0.6.3
pycparser	2.19
pycrypto	2.6.1
pycurl	7.43.0.2
pydotplus	2.0.2
pyflakes	2.1.1
Pygments	2.3.1
PyHamcrest	2.0.2
pylint	2.3.1
pyodbc	4.0.26
pyOpenSSL	19.0.0
pyparsing	2.3.1
pyreadline	2.1
pyrsistent	0.14.11
PySocks	1.6.8
pytest	4.3.1
pytest-arraydiff	0.3
pytest-astropy	0.5.0
pytest-doctestplus	0.3.0
pytest-openfiles	0.3.2
pytest-remotedata	0.3.1
python-dateutil	2.8.0
pytz	2018.9
PyWavelets	1.0.2
pywin32	223
pywinpty	0.5.5

PyYAML	5.1
pyzmq	18.0.0
QtAwesome	0.5.7
qtconsole	4.4.3
QtPy	1.7.0
requests	2.21.0
rope	0.12.0
ruamel-yaml	0.15.46
scikit-image	0.14.2
scikit-learn	0.20.3
scipy	1.2.1
seaborn	0.9.0
Send2Trash	1.5.0
service-identity	18.1.0
setuptools	40.8.0
simplegeneric	0.8.1
singledispatch	3.4.0.3
six	1.12.0
sklearn	0.0
snowballstemmer	1.2.1
sortedcollections	1.1.2
sortedcontainers	2.1.0
soupsieve	1.8
Sphinx	1.8.5
sphinxcontrib-websupport	1.1.0
spyder	3.3.3
spyder-kernels	0.4.2
SQLAlchemy	1.3.1
statsmodels	0.9.0
sympy	1.3
tables	3.5.1
tblib	1.3.2
terminado	0.8.1
testpath	0.4.2
toolz	0.9.0
tornado	6.0.2
tqdm	4.31.1
traitlets	4.3.2
Twisted	19.2.0
unicodcsv	0.14.1
urllib3	1.24.1
wcwidth	0.1.7
webencodings	0.5.1
Werkzeug	0.14.1
wheel	0.33.1
widgetsnbextension	3.4.2
win-inet-pton	1.1.0
win-unicode-console	0.5
wincertstore	0.2
wrapt	1.11.1
xlrd	1.2.0
XlsxWriter	1.1.5
xlwings	0.15.4
xlwt	1.3.0
zict	0.1.4

```
zinn
```

```
a 2 2
```

WARNING: You are using pip version 20.2.3; however, version 21.1.2 is available.

You should consider upgrading via the 'C:\Users\Alekhya\Anaconda3\python.exe -m pip install --upgrade pip' command.

Note: you may need to restart the kernel to use updated packages.

Numpy- Numerical Python

In [3]:

```
1 import numpy as np
```

In [4]: 1 print(dir(np))

```
['ALLOW_THREADS', 'AxisError', 'BUFSIZE', 'CLIP', 'ComplexWarning', 'DataSource', 'ERR_CALL', 'ERR_DEFAULT', 'ERR_IGNORE', 'ERR_LOG', 'ERR_PRINT', 'ERR_RAISE', 'ERR_WARN', 'FLOATING_POINT_SUPPORT', 'FPE_DIVIDEBYZERO', 'FPE_INVALID', 'FPE_OVERFLOW', 'FPE_UNDERFLOW', 'False_', 'Inf', 'Infinity', 'MAXDIMS', 'MAY_SHARE_BOUNDS', 'MAY_SHARE_EXACT', 'MachAr', 'ModuleDeprecationWarning', 'NAN', 'NINF', 'NZERO', 'NaN', 'PINF', 'PZERO', 'RAISE', 'RankWarning', 'SHIFT_DIVIDEBYZERO', 'SHIFT_INVALID', 'SHIFT_OVERFLOW', 'SHIFT_UNDERFLOW', 'ScalarType', 'Tester', 'TooHardError', 'True_', 'UFUNC_BUFSIZE_DEFAULT', 'UFUNC_PYVALS_NAME', 'VisibleDeprecationWarning', 'WRAP', '_NoValue', '_UFUNC_API', '_NUMPY_SETUP_', '__all__', '__builtins__', '__cached__', '__config__', '__doc__', '__file__', '__git_revision__', '__loader__', '__mkkl_version__', '__name__', '__package__', '__path__', '__spec__', '__version__', '_add_newdoc_ufunc', '_arg', '_distributor_init', '_globals', '_mat', '_mklinit', '_pytesttester', 'abs', 'absolute', 'absolute_import', 'add', 'add_docstring', 'add_newdoc', 'add_newdoc_ufunc', 'alen', 'all', 'allclose', 'alltrue', 'amax', 'amin', 'angle', 'any', 'append', 'apply_along_axis', 'apply_over_axes', 'arange', 'arccos', 'arccosh', 'arcsin', 'arcsinh', 'arctan', 'arctan2', 'arctanh', 'argmax', 'argmin', 'argpartition', 'argsort', 'argwhere', 'around', 'array', 'array2string', 'array_equal', 'array_equiv', 'array_repr', 'array_split', 'array_str', 'asanyarray', 'asarray', 'asarray_chkfinite', 'ascontiguousarray', 'asfarray', 'asfortranarray', 'asmatrix', 'asscalar', 'atleast_1d', 'atleast_2d', 'atleast_3d', 'average', 'bartlett', 'base_repr', 'binary_repr', 'bincount', 'bitwise_and', 'bitwise_not', 'bitwise_or', 'bitwise_xor', 'blackman', 'block', 'bmat', 'bool', 'bool8', 'bool_', 'broadcast', 'broadcast_arrays', 'broadcast_to', 'busday_count', 'busday_offset', 'busdaycalendar', 'byte', 'byte_bounds', 'bytes0', 'bytes_', 'c_', 'can_cast', 'cast', 'cbart', 'cdouble', 'ceil', 'cfloat', 'char', 'character', 'chararray', 'choose', 'clip', 'clongdouble', 'clongfloat', 'column_stack', 'common_type', 'compare_chararrays', 'compat', 'complex', 'complex128', 'complex64', 'complex_', 'complexfloating', 'compress', 'concatenate', 'conj', 'conjugate', 'convolve', 'copy', 'copysign', 'copyto', 'core', 'corrcoef', 'correlate', 'cos', 'cosh', 'count_nonzero', 'cov', 'cross', 'csingle', 'ctypeslib', 'cumprod', 'cumproduct', 'cumsum', 'datetime64', 'datetime_as_string', 'datetime_data', 'deg2rad', 'degrees', 'delete', 'deprecate', 'deprecate_with_doc', 'diag', 'diag_indices', 'diag_indices_from', 'diagflat', 'diagonal', 'diff', 'digitize', 'disp', 'distutils', 'divide', 'division', 'divmod', 'doc', 'dot', 'double', 'dsplit', 'dstack', 'dtype', 'dual', 'e', 'ediff1d', 'einsum', 'einsum_path', 'emath', 'empty', 'empty_like', 'equal', 'errstate', 'euler_gamma', 'exp', 'exp2', 'expand_dims', 'expm1', 'extract', 'eye', 'f2py', 'fabs', 'fastCopyAndTranspose', 'fft', 'fill_diagonal', 'find_common_type', 'finfo', 'fix', 'flatiter', 'flatnonzero', 'flexible', 'flip', 'fliplr', 'flipud', 'float', 'float16', 'float32', 'float64', 'float_', 'float_power', 'floating', 'floor', 'floor_divide', 'fmax', 'fmin', 'fmod', 'format_float_positional', 'format_float_scientific', 'format_parser', 'frexp', 'frombuffer', 'fromfile', 'fromfunction', 'fromiter', 'frompyfunc', 'fromregex', 'fromstring', 'full', 'full_like', 'fv', 'gcd', 'generic', 'genfromtxt', 'geomspace', 'get_array_wrap', 'get_include', 'get_printoptions', 'getbufsize', 'geterr', 'geterrcall', 'geterrobj', 'gradient', 'greater', 'greater_equal', 'half', 'hamming', 'hanning', 'heaviside', 'histogram', 'histogram2d', 'histogram_bin_edges', 'histogramdd', 'hsplit', 'hstack', 'hypot', 'i0', 'identity', 'iinfo', 'imag', 'in1d', 'index_exp', 'indices', 'indexact', 'inf', 'info', 'infty', 'inner', 'insert', 'int', 'int0', 'int16', 'int32', 'int64', 'int8', 'int_', 'int_asbuffer', 'intc', 'integer', 'interp', 'intersect1d', 'intp', 'invert', 'ipmt', 'irr', 'is_busday', 'isclose', 'iscomplex', 'iscomplexobj', 'isfinite', 'isfortran', 'isin', 'isinf', 'isnan', 'isn
```

```
at', 'isneginf', 'isposinf', 'isreal', 'isrealobj', 'isscalar', 'issctype',
'issubclass_', 'issubdtype', 'issubstctype', 'iterable', 'ix_', 'kaiser', 'kro
n', 'lcm', 'ldexp', 'left_shift', 'less', 'less_equal', 'lexsort', 'lib', 'li
nalg', 'linspace', 'little_endian', 'load', 'loads', 'loadtxt', 'log', 'log1
0', 'log1p', 'log2', 'logaddexp', 'logaddexp2', 'logical_and', 'logical_not',
'logical_or', 'logical_xor', 'logspace', 'long', 'longcomplex', 'longdouble',
'longfloat', 'longlong', 'lookfor', 'ma', 'mafromtxt', 'mask_indices', 'mat',
'math', 'matmul', 'matrix', 'matrixlib', 'max', 'maximum', 'maximum_sctype',
'may_share_memory', 'mean', 'median', 'memmap', 'meshgrid', 'mgrid', 'min',
'min_scalar_type', 'minimum', 'mintypecode', 'mirr', 'mod', 'modf', 'moveaxi
s', 'msort', 'multiply', 'nan', 'nan_to_num', 'nanargmax', 'nanargmin', 'nanc
umprod', 'nancumsum', 'nanmax', 'nanmean', 'nanmedian', 'nanmin', 'nanpercent
ile', 'nanprod', 'nanquantile', 'nanstd', 'nansum', 'nanvar', 'nbytes', 'ndar
ray', 'ndenumerate', 'ndfromtxt', 'ndim', 'ndindex', 'nditer', 'negative', 'n
ested_iters', 'newaxis', 'nextafter', 'nonzero', 'not_equal', 'nper', 'npv',
'numarray', 'number', 'obj2sctype', 'object', 'object0', 'object_', 'ogrid',
'oldnumeric', 'ones', 'ones_like', 'outer', 'packbits', 'pad', 'partition',
'percentile', 'pi', 'piecewise', 'place', 'pmt', 'poly', 'poly1d', 'polyadd',
'polyder', 'polydiv', 'polyfit', 'polyint', 'polymul', 'polynomial', 'polysu
b', 'polyval', 'positive', 'power', 'ppmt', 'print_function', 'printoptions',
'prod', 'product', 'promote_types', 'ptp', 'put', 'put_along_axis', 'putmas
k', 'pv', 'quantile', 'r_', 'rad2deg', 'radians', 'random', 'random_intel',
'rank', 'rate', 'ravel', 'ravel_multi_index', 'real', 'real_if_close', 'rec',
'recarray', 'recfromcsv', 'recfromtxt', 'reciprocal', 'record', 'remainder',
'repeat', 'require', 'reshape', 'resize', 'result_type', 'right_shift', 'rin
t', 'roll', 'rollaxis', 'roots', 'rot90', 'round', 'round_', 'row_stack', 's
_', 'safe_eval', 'save', 'savetxt', 'savez', 'savez_compressed', 'sctype2cha
r', 'sctypeDict', 'sctypeNA', 'sctypes', 'searchsorted', 'select', 'set_numer
ic_ops', 'set_printoptions', 'set_string_function', 'setbufsize', 'setdiff1
d', 'seterr', 'seterrcall', 'seterrobj', 'setxor1d', 'shape', 'shares_memor
y', 'short', 'show_config', 'sign', 'signbit', 'signedinteger', 'sin', 'sin
c', 'single', 'singlecomplex', 'sinh', 'size', 'sometrue', 'sort', 'sort_comp
lex', 'source', 'spacing', 'split', 'sqrt', 'square', 'squeeze', 'stack', 'st
d', 'str', 'str0', 'str_', 'string_', 'subtract', 'sum', 'swapaxes', 'sys',
'take', 'take_along_axis', 'tan', 'tanh', 'tensordot', 'test', 'testing', 'te
sts', 'tile', 'timedelta64', 'trace', 'tracemalloc_domain', 'transpose', 'tra
pz', 'tri', 'tril', 'tril_indices', 'tril_indices_from', 'trim_zeros', 'tri
u', 'triu_indices', 'triu_indices_from', 'true_divide', 'trunc', 'typeDict',
'typeNA', 'typecodes', 'typename', 'ubyte', 'ufunc', 'uint', 'uint0', 'uint1
6', 'uint32', 'uint64', 'uint8', 'uintc', 'uintp', 'ulonglong', 'unicode', 'u
nicode_', 'union1d', 'unique', 'unpackbits', 'unravel_index', 'unsignedintege
r', 'unwrap', 'ushort', 'vander', 'var', 'vdot', 'vectorize', 'version', 'voi
d', 'void0', 'vsplit', 'vstack', 'warnings', 'where', 'who', 'zeros', 'zeros_
like']
```

```
In [6]: 1 help(np.vstack)
```

Help on function vstack in module numpy:

vstack(tup)

Stack arrays in sequence vertically (row wise).

This is equivalent to concatenation along the first axis after 1-D arrays of shape `(N,)` have been reshaped to `(1,N)`. Rebuilds arrays divided by `vsplit`.

This function makes most sense for arrays with up to 3 dimensions. For instance, for pixel-data with a height (first axis), width (second axis), and r/g/b channels (third axis). The functions `concatenate`, `stack` and `block` provide more general stacking and concatenation operations.

Parameters

tup : sequence of ndarrays

The arrays must have the same shape along all but the first axis.
1-D arrays must have the same length.

Returns

stacked : ndarray

The array formed by stacking the given arrays, will be at least 2-D.

See Also

stack : Join a sequence of arrays along a new axis.

hstack : Stack arrays in sequence horizontally (column wise).

dstack : Stack arrays in sequence depth wise (along third dimension).

concatenate : Join a sequence of arrays along an existing axis.

vsplit : Split array into a list of multiple sub-arrays vertically.

block : Assemble arrays from blocks.

Examples

```
>>> a = np.array([1, 2, 3])
```

```
>>> b = np.array([2, 3, 4])
```

```
>>> np.vstack((a,b))
```

```
array([[1, 2, 3],
       [2, 3, 4]])
```

```
>>> a = np.array([[1], [2], [3]])
```

```
>>> b = np.array([[2], [3], [4]])
```

```
>>> np.vstack((a,b))
```

```
array([[1],
       [2],
       [3],
       [2],
       [3],
       [4]])
```

```
In [7]: 1 a = np.array([1,2,3,4])
```

```
In [8]: 1 a
```

```
Out[8]: array([1, 2, 3, 4])
```

```
In [9]: 1 a.ndim # it will find out which dimension
```

```
Out[9]: 1
```

```
In [10]: 1 a.dtype # data type
```

```
Out[10]: dtype('int32')
```

```
In [11]: 1 b = np.array(["apssdc"])
2 b.dtype
```

```
Out[11]: dtype('<U6')
```

```
In [12]: 1 c = np.array([1,"apssdc",5.6])
2 c
```

```
Out[12]: array(['1', 'apssdc', '5.6'], dtype='<U11')
```

```
In [15]: 1 a = np.append(a,[1,2,3])
```

```
In [16]: 1 a
```

```
Out[16]: array([1, 2, 3, 4, 1, 2, 3])
```

```
In [19]: 1 range(1,10,0.5)
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-19-27d58624578b> in <module>
----> 1 range(1,10,0.5)
```

```
TypeError: 'float' object cannot be interpreted as an integer
```

```
In [23]: 1 np.arange(1,23.5,0.5) # start,end(exclusive),step
```

```
Out[23]: array([ 1. ,  1.5,  2. ,  2.5,  3. ,  3.5,  4. ,  4.5,  5. ,  5.5,  6. ,
        6.5,  7. ,  7.5,  8. ,  8.5,  9. ,  9.5, 10. , 10.5, 11. , 11.5,
        12. , 12.5, 13. , 13.5, 14. , 14.5, 15. , 15.5, 16. , 16.5, 17. ,
        17.5, 18. , 18.5, 19. , 19.5, 20. , 20.5, 21. , 21.5, 22. , 22.5,
        23. ])
```

```
In [25]: 1 x = np.linspace(1,100,34) # start,ending(inclusive),number of divisions/part
```

```
In [26]: 1 x
```

```
Out[26]: array([ 1.,  4.,  7., 10., 13., 16., 19., 22., 25., 28., 31.,
                34., 37., 40., 43., 46., 49., 52., 55., 58., 61., 64.,
                67., 70., 73., 76., 79., 82., 85., 88., 91., 94., 97.,
                100.])
```

```
In [27]: 1 x.ndim
```

```
Out[27]: 1
```

```
In [28]: 1 len(x)
```

```
Out[28]: 34
```

```
In [29]: 1 x.itemsize
```

```
Out[29]: 8
```

```
In [30]: 1 x.nbytes
```

```
Out[30]: 272
```

```
In [32]: 1 np.full(10,23)
```

```
Out[32]: array([23, 23, 23, 23, 23, 23, 23, 23, 23, 23])
```

```
In [33]: 1 np.full(10,"a")
```

```
Out[33]: array(['a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a'], dtype='<U1')
```

```
In [34]: 1 ## 2d- array
2
3 b = np.array([[1,2,3],[3,4,5]])
4 b
```

```
Out[34]: array([[1, 2, 3],
                [3, 4, 5]])
```

```
In [35]: 1 b.ndim
```

```
Out[35]: 2
```



```
In [38]: 1 ### 3d -array
          2 c = np.array([[[1,2,3],[3,4,5]],[[8,9,10],[7,8,9]]])
          3 c
```

```
Out[38]: array([[[ 1,  2,  3],
                  [ 3,  4,  5]],

                [[ 8,  9, 10],
                  [ 7,  8,  9]]])
```

```
In [39]: 1 c.ndim
```

```
Out[39]: 3
```

```
In [40]: 1 a
```

```
Out[40]: array([1, 2, 3, 4, 1, 2, 3])
```

```
In [41]: 1 b #(2,3)
```

```
Out[41]: array([[1, 2, 3],
                 [3, 4, 5]])
```

```
In [42]: 1 c #(2,2,3) positions,rows,columns
```

```
Out[42]: array([[[ 1,  2,  3],
                  [ 3,  4,  5]],

                [[ 8,  9, 10],
                  [ 7,  8,  9]]])
```

```
In [43]: 1 d = np.array([[8,9,10],[4,5,6]])
          2 d
```

```
Out[43]: array([[ 8,  9, 10],
                 [ 4,  5,  6]])
```

```
In [44]: 1 x = np.array([b,d])
```

```
In [45]: 1 x
```

```
Out[45]: array([[[ 1,  2,  3],
                  [ 3,  4,  5]],

                [[ 8,  9, 10],
                  [ 4,  5,  6]]])
```

```
In [46]: 1 a
```

```
Out[46]: array([1, 2, 3, 4, 1, 2, 3])
```

```
In [47]: 1 a[3]
```

```
Out[47]: 4
```

```
In [48]: 1 a[-1]
```

```
Out[48]: 3
```

```
In [50]: 1 a[2:4]
```

```
Out[50]: array([3, 4])
```

```
In [49]: 1 b
```

```
Out[49]: array([[1, 2, 3],  
               [3, 4, 5]])
```

```
In [52]: 1 b[1][2]
```

```
Out[52]: 5
```

```
In [53]: 1 b[1,2] # row index and column index
```

```
Out[53]: 5
```

```
In [56]: 1 b[:,2:]
```

```
Out[56]: array([[3],  
               [5]])
```

```
In [57]: 1 b[1:,1:]
```

```
Out[57]: array([[4, 5]])
```

```
In [58]: 1 c
```

```
Out[58]: array([[[ 1,  2,  3],  
                [ 3,  4,  5]],  
               [[ 8,  9, 10],  
                [ 7,  8,  9]])
```

```
In [59]: 1 c[1,0,1]
```

```
Out[59]: 9
```

```
In [60]: 1 c[0,1,1]
```

```
Out[60]: 4
```

```
In [61]: 1 c[0]
```

```
Out[61]: array([[1, 2, 3],  
               [3, 4, 5]])
```

```
In [62]: 1 c[1]
```

```
Out[62]: array([[ 8,  9, 10],  
               [ 7,  8,  9]])
```

```
In [63]: 1 c
```

```
Out[63]: array([[[ 1,  2,  3],  
                 [ 3,  4,  5]],  
               [[ 8,  9, 10],  
               [ 7,  8,  9]])
```

```
In [64]: 1 c[:,1,1]
```

```
Out[64]: array([4, 8])
```

```
In [65]: 1 c[:, :, 2:]
```

```
Out[65]: array([[[ 3],  
                 [ 5]],  
               [[10],  
               [ 9]])
```

```
In [66]: 1 c[:, :, -1:]
```

```
Out[66]: array([[[ 3],  
                 [ 5]],  
               [[10],  
               [ 9]])
```

```
In [67]: 1 # reshaping an array
```

```
In [68]: 1 a
```

```
Out[68]: array([1, 2, 3, 4, 1, 2, 3])
```

```
In [69]: 1 a.shape
```

```
Out[69]: (7,)
```

```
In [71]: 1 b.shape
```

```
Out[71]: (2, 3)
```

```
In [72]: 1 c.shape
```

```
Out[72]: (2, 2, 3)
```

```
In [73]: 1 a.reshape(2,4)
```

```
-----  
ValueError                                Traceback (most recent call last)  
<ipython-input-73-5fcd0a049a15> in <module>  
----> 1 a.reshape(2,4)
```

```
ValueError: cannot reshape array of size 7 into shape (2,4)
```

```
In [74]: 1 a.reshape(2,3)
```

```
-----  
ValueError                                Traceback (most recent call last)  
<ipython-input-74-d3d8e5b27339> in <module>  
----> 1 a.reshape(2,3)
```

```
ValueError: cannot reshape array of size 7 into shape (2,3)
```

```
In [76]: 1 x = a.reshape(1,7)
```

```
In [77]: 1 x
```

```
Out[77]: array([[1, 2, 3, 4, 1, 2, 3]])
```

```
In [84]: 1 x[0,3]
```

```
Out[84]: 4
```

```
In [80]: 1 a
```

```
Out[80]: array([1, 2, 3, 4, 1, 2, 3])
```

```
In [81]: 1 a[3]
```

```
Out[81]: 4
```

```
In [78]: 1 x.ndim
```

```
Out[78]: 2
```

```
In [79]: 1 a.reshape(7,1)
```

```
Out[79]: array([[1],
                [2],
                [3],
                [4],
                [1],
                [2],
                [3]])
```

```
In [85]: 1 a.reshape(1,-1)
```

```
Out[85]: array([[1, 2, 3, 4, 1, 2, 3]])
```

```
In [86]: 1 a.reshape(-1,1)
```

```
Out[86]: array([[1],
                [2],
                [3],
                [4],
                [1],
                [2],
                [3]])
```

```
In [87]: 1 a.reshape(-1,-1)
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-87-aa43799da6cd> in <module>
----> 1 a.reshape(-1,-1)
```

ValueError: can only specify one unknown dimension

```
In [88]: 1 a.reshape(2,-1)
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-88-4787a072710a> in <module>
----> 1 a.reshape(2,-1)
```

ValueError: cannot reshape array of size 7 into shape (2,newaxis)

```
In [89]: 1 f = np.array([1,2,3,4,5,6])
          2 f
```

```
Out[89]: array([1, 2, 3, 4, 5, 6])
```

```
In [90]: 1 f.reshape(3,2)
```

```
Out[90]: array([[1, 2],
               [3, 4],
               [5, 6]])
```

```
In [91]: 1 f.reshape(2,3)
```

```
Out[91]: array([[1, 2, 3],
               [4, 5, 6]])
```

```
In [92]: 1 f.reshape(1,6)
```

```
Out[92]: array([[1, 2, 3, 4, 5, 6]])
```

```
In [93]: 1 f.reshape(6,1)
```

```
Out[93]: array([[1],
               [2],
               [3],
               [4],
               [5],
               [6]])
```

```
In [94]: 1 f.reshape(2,-1)
```

```
Out[94]: array([[1, 2, 3],
               [4, 5, 6]])
```

```
In [95]: 1 f.reshape(-1,2)
```

```
Out[95]: array([[1, 2],
               [3, 4],
               [5, 6]])
```

```
In [97]: 1 f.reshape(4,-1)
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-97-b4e9a98b6d82> in <module>
----> 1 f.reshape(4,-1)
```

```
ValueError: cannot reshape array of size 6 into shape (4,newaxis)
```

```
In [98]: 1 s = np.array([1,2,3,4,5,6,7,8,9,10])
        2 s
```

```
Out[98]: array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

```
In [99]: 1 s.reshape(10,1,1)
```

```
Out[99]: array([[[ 1]],  
               [[ 2]],  
               [[ 3]],  
               [[ 4]],  
               [[ 5]],  
               [[ 6]],  
               [[ 7]],  
               [[ 8]],  
               [[ 9]],  
               [[10]])
```

```
In [100]: 1 s.reshape(5,2,1)
```

```
Out[100]: array([[[ 1],  
                  [ 2]],  
                [[ 3],  
                  [ 4]],  
                [[ 5],  
                  [ 6]],  
                [[ 7],  
                  [ 8]],  
                [[ 9],  
                  [10]])
```

```
In [ ]: 1
```