# Appendix C.2: Overview of Python Data Science Libraries used in XCast

**NumPy** **(https://numpy.org/)** NumPy is a core python library used for manipulating unlabeled multidimensional arrays (Harris et. al., 2020). It is a cornerstone of the Python language and implements the basic numerical and scientific programming utilities like array manipulations. It is highly-performant and allows Python users access to highly optimized low-level functionality. Many NumPy functions are implemented with compiled languages, to reduce the overhead introduced by the Python language. The most commonly used NumPy functionalities are array indexing, array slicing, array arithmetic, and basic linear algebra.

**SciPy** **(https://scipy.org/)** SciPy is a widely-used scientific computing library for Python (Virtanen et. al., 2020). It is a companion library of NumPy, and is designed to apply more complex scientific programming functionality to NumPy arrays. It makes the highly-optimized low-level linear algebra libraries, Basic Linear Algebra Subprograms (BLAS) and Linear Algebra Package (LAPACK) directly accessible in Python. It also implements more complex statistical methods like basic regression analysis, kernel smoothing, interpolation and much more.

**Statsmodels (https://www.statsmodels.org/stable/)** Statsmodels is a Python library built specifically for statistical analysis. (Seabold, Skipper, & Perktold, 2010) It implements a diverse set of regression analyses, including linear regression, logistic regression, and generalized linear models.

**Scikit-Learn** **(https://scikit-learn.org/stable/)** Scikit-Learn is a popular open-source machine learning library in Python (Pedregosa et. al., 2011). It is supported by a large community of contributors and users, which can be found at https://blog.scikit-learn.org/. Its intuitive interface makes it an ideal entryway to Python machine learning; numerous undergraduate and graduate-level courses around the world use it to teach statistics and machine learning. It implements a broad array of machine learning techniques including regression methods, decision tree-based methods, clustering methods, neural network-based methods, preprocessing methods, validation methods, and skill assessment methods. Its implementations are also high-enough performance to be used at scale in industry. Scikit-Learn

**Xarray** **(https://docs.xarray.dev/en/stable/)** Xarray is a powerful Python library for manipulating labeled multidimensional arrays (Hoyer, S., Hamman, J, 2017). It adds functionality to NumPy and Dask by tracking the names, sizes, and indexing orders of underlying unlabelled arrays, and by labeling them with coordinates which can then be used for indexing and selecting data. It is commonly used to manipulate gridded geospatial data in NetCDF or HDF5 format. A number of geospatial data libraries built on top of Xarray have been developed in recent years.

**Dask** **(https://www.dask.org/)** Dask is a high-performance computing library built to extend the functionality of standard libraries like NumPy and SciPy (Rocklin, M. 2015). It allows data to be manipulated out-of-core, i.e., without the entire dataset being loaded to memory at once. This allows the user to work with datasets too large for memory. It also implements efficient parallelism by adopting a pure-python task scheduling approach. Dask’s APIs and lazily executed functions create “graphs”, which are then interpreted and optimized dynamically by Dask’s task schedulers. Much has been written about Dask’s architecture and task scheduling (https://docs.dask.org/en/stable/scheduling.html), but in short, Dask provides a broad array of task schedulers, for use in many different contexts. Multithreading and multiprocessing are both possible with Dask. It is possible to specify a dask task scheduler to suit a multitude of computing environments, large and small. Even more detail about Dask’s internals is available at https://docs.dask.org/en/stable/internals.html.