* Code must be in a .pyx file
* Must consist of functions and imports: def, cdef, cpdef
* No errors in .pyx file as it will result in a failure of all functions
* If updating an existing cython module, make sure it’s not in use. It will fail otherwise
* Must have Visual Studio installed to compile the code.

**Setting up (ex: SoccerHycomHF)**

Make a separate file named setup.py. You only need to make a file like this once per module.

try:

from setuptools import setup

from setuptools import Extension

except ImportError:

from distutils.core import setup

from distutils.extension import Extension

from Cython.Distutils import build\_ext

import numpy as np

ext\_modules = [Extension("SoccerHycomHF",["SoccerHycomHF.pyx"])] *# import name and .pyx file that you want to cythonize*

setup(

name= 'Generic model class',

cmdclass = {'Soccer\_ext': build\_ext}, *#the cmd name to update the file.*

include\_dirs = [np.get\_include()],

ext\_modules = ext\_modules)

Open up command prompt and change to the directory that has both the setup.py and the .pyx file. Type this into the command prompt:

python setup.py Soccer\_ext --inplace

The build will be created after and all you have to do to use the functions is write import (and the import name from the setup. To update an existing module, change the content in the .pyx and rebuild it in the command prompt using the command line above.