*This document was written in approx. 08.11.2018.*

[Python dependent tools 1](#_Toc15907495)

[Troubleshoot: 4](#_Toc15907496)

[Force update 4](#_Toc15907497)

[SSL relatederror 5](#_Toc15907498)

# Python dependent tools

Note:

1. If python2 or pip2 is not recognized and you don’t have python3 installed use python and pip instead. If you have both Python 2 and Python 3 installed it is better to define soft-link (mklink python2.exe python.exe).
2. I assume that you have Windows 64 bit. If not, change the suffix of the file appropriately.
3. This tutorial doesn’t use dockers.
4. From <https://repo.anaconda.com/archive/> download & Install Anaconda2-4.0.0-Windows-x86\_64.exe to c:\programs\Anaconda
5. On cmd type  
      
   python2 -m pip install --upgrade pip  
     
   Note: If you have **SSL related** error, see troubleshoot section below.
6. From <https://www.microsoft.com/EN-US/DOWNLOAD/DETAILS.ASPX?ID=44266> download & Install VCForPython27.msi
7. From <https://www.lfd.uci.edu/~gohlke/pythonlibs/> download    
   sasl-0.2.1-cp27-cp27m-win\_amd64.whl to some directory.
8. Go to that directory (run cmd, cd to it) and run

python2 -m pip install sasl-0.2.1-cp27-cp27m-win\_amd64.whl  
  
Note: If you have **SSL related** error, see troubleshoot section below.  
  
(On Linux run   
yum install python-virtualenv cyrus-sasl-devel  
python2 -m pip2 install sasl  
)

1. Optionally, you can install numpy+mkl  
     
   Numpy+MKL is linked to the [Intel® Math Kernel Library](https://software.intel.com/en-us/intel-mkl/) and includes required DLLs in the numpy.DLLs directory.  
   Numpy+Vanilla is a minimal distribution, which does not include any optimized BLAS libray or C runtime DLLs.  
     
   From <https://www.silx.org/pub/wheelhouse/old/> download

numpy-1.9.2+mkl-cp27-none-win\_amd64.whl

1. Go to that directory (run cmd, cd to it) and run

python2 -m pip install numpy-1.9.2+mkl-cp27-none-win\_amd64.whl

1. Optionally, you can install scipy   
     
   From <https://www.silx.org/pub/wheelhouse/old/> download   
   scipy-0.18.0-cp27-cp27m-win\_amd64.whl
2. Go to that directory (run cmd, cd to it) and run

python2 -m pip install scipy-0.18.0-cp27-cp27m-win\_amd64.whl

1. Save attached file to the working directory (from step 2)



1. In this directory run

python2 -m pip2 install -r req-hive.txt

1. Install **Virtualenv** and VirtualEnvWrapper-win. <https://www.pythonforbeginners.com/basics/how-to-use-python-virtualenv>

# Troubleshoot:

## Force update

--no-deps  
python2 -m pip install --upgrade --no-deps --ignore-installed --no-cache futures==3.1.1

## **SSL related** error

When you execute python2 –m pip install <package>, and you get the following **SSL related** error:

pip is configured with locations that require TLS/SSL, however the ssl module in Python is not available.

Collecting <package>

Could not fetch URL https://pypi.python.org/simple/<package>/: There was a problem confirming the ssl certificate: Can't connect to HTTPS URL because the SSL module is not available. - skipping

Could not find a version that satisfies the requirement <package> (from versions: )

No matching distribution found for <package>

See <https://stackoverflow.com/questions/41328451/ssl-module-in-python-is-not-available-when-installing-package-with-pip3>

The problem can be caused by DLLs in the Windows\System32 folder (e.g. libcrypto-1\_1-x64.dll or libssl-1\_1-x64.dll or others) placed there by other software.

The fix was installing Win64OpenSSL-1\_1\_1b.exe from <https://slproweb.com/products/Win32OpenSSL.html> which replaces the dlls by more recent versions.