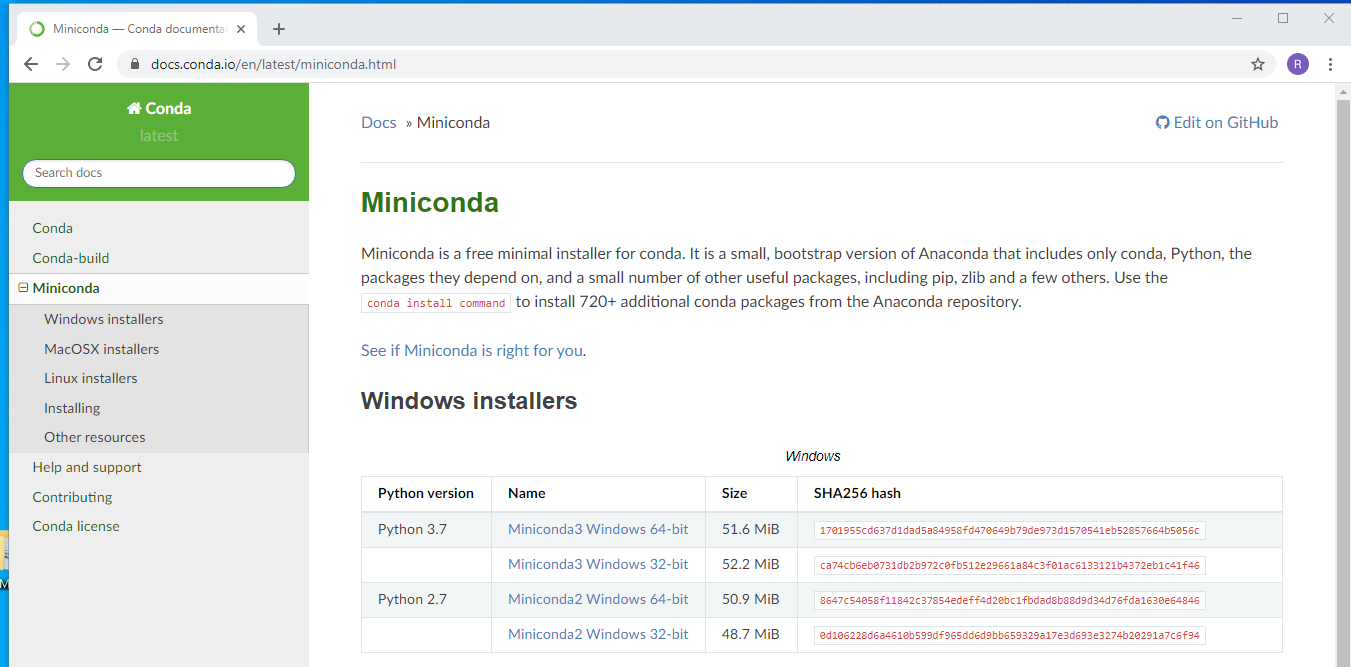
# How work with “katcorr”

* **If you have experience with Python, please directly proceed section 4**

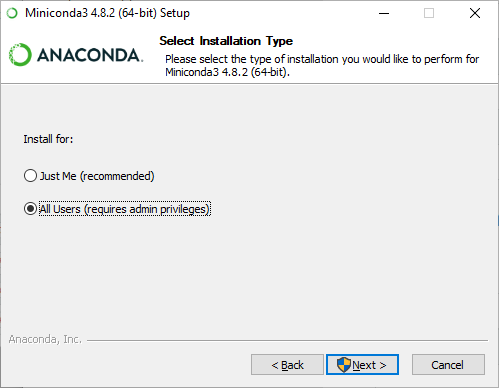
## Install Miniconda (Python3.7, usually 64-bit)

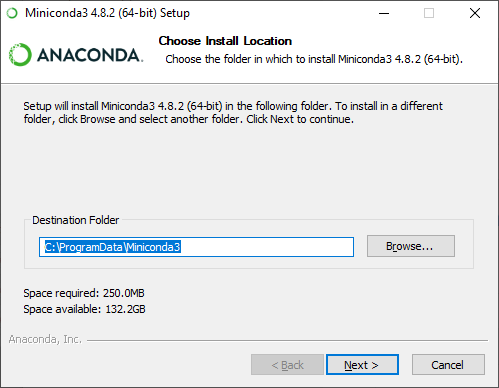
If you haven’t done so, please install Miniconda first.

<https://docs.conda.io/en/latest/miniconda.html>

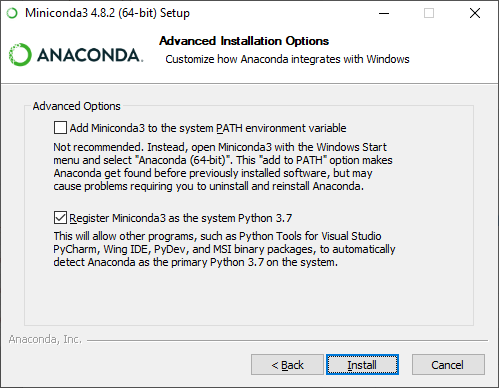


Install for all users:





Leave second option selected:



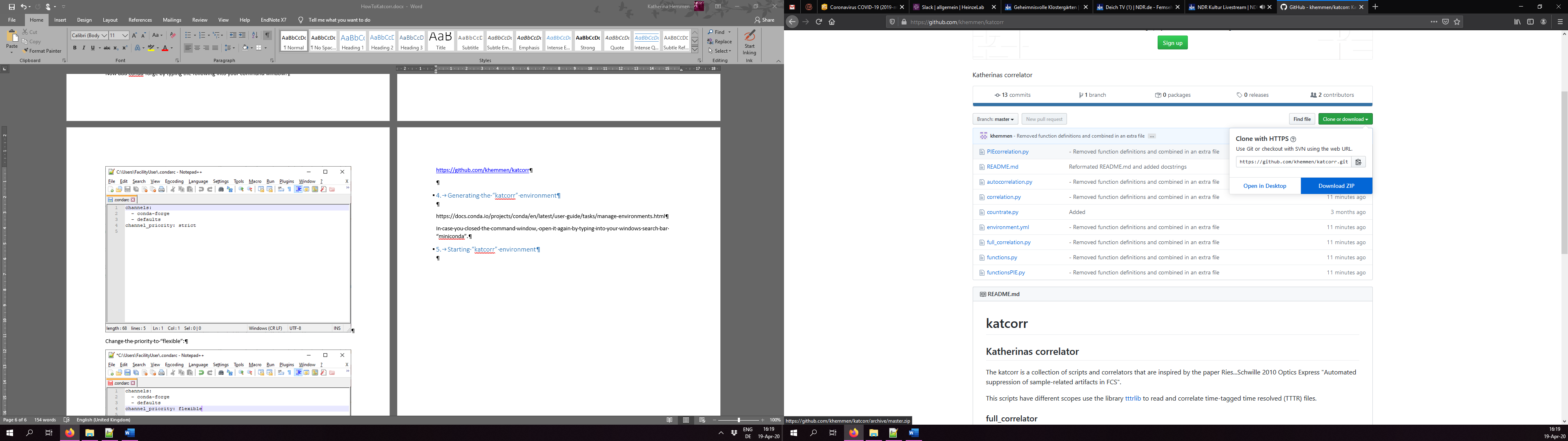
## Getting “katcorr” from Github

Go to the following website and download the newest version:

<https://github.com/khemmen/katcorr>

You will need to create an account at Github to be able to download the files.

**As this is a private repository, please send me your account name and I will send you an invite!**



Best is to place the folder on “C:\Users\username”

## Generating the “katcorr” environment

<https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html>

Option: “Creating an environment from an environment.yml file”

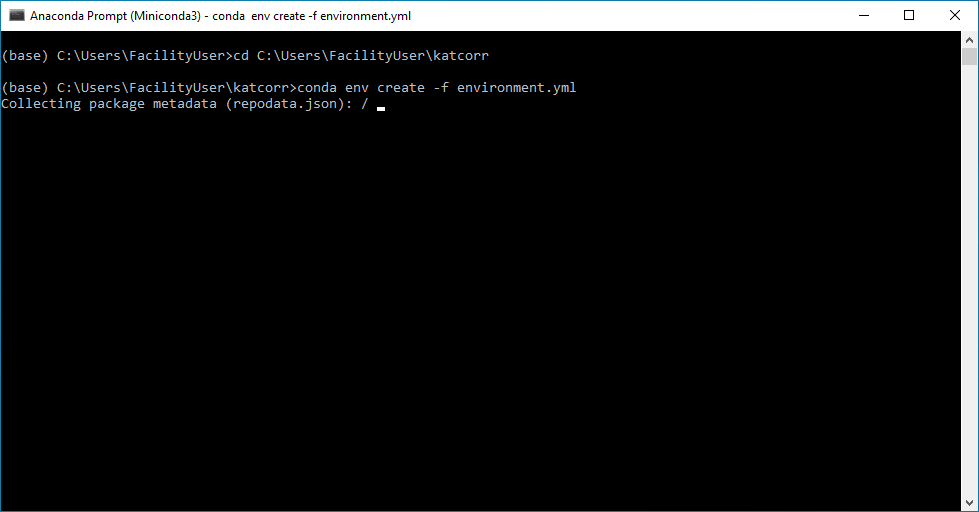
In case you closed the command window, open it again by typing into your windows search bar “miniconda”.

Navigate to your folder by typing into the console:

(base) C:\Users\User > cd C:\Users\User\katcorr

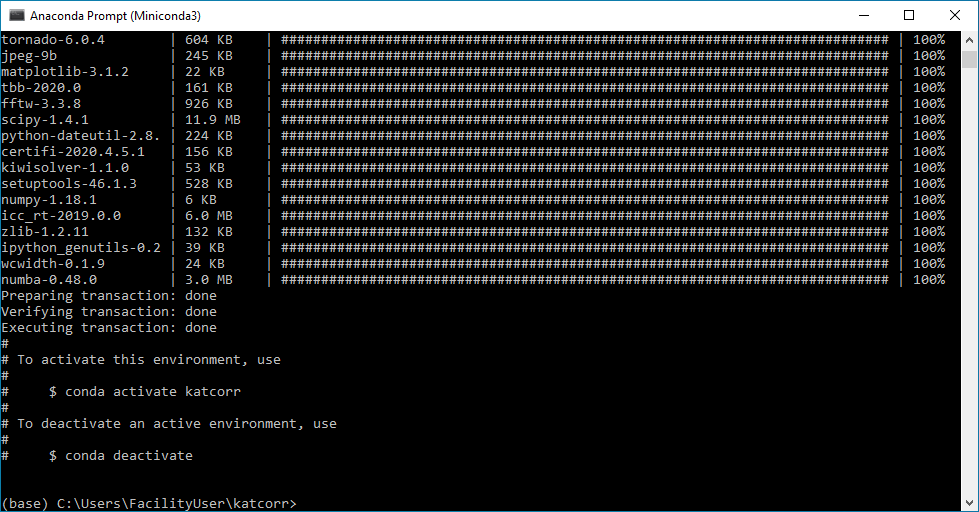
Generate the new environment named „katcorr”:

(base) C:\Users\User\katcorr > conda env create -f environment.yml



It will download quite some packages and dependencies.

When it is done it will show the following:



## Starting “katcorr” environment

Activate the environment by typing:

conda activate katcorr

When you want to use “katcorr” next time, you can directly start from section 4 using the environment activation.

Please make sure you navigate to the katcorr-folder before trying to run any command.

(base) C:\Users\User > conda activate katcorr

(katcorr) C:\Users\User > cd C:\Users\username\katcorr

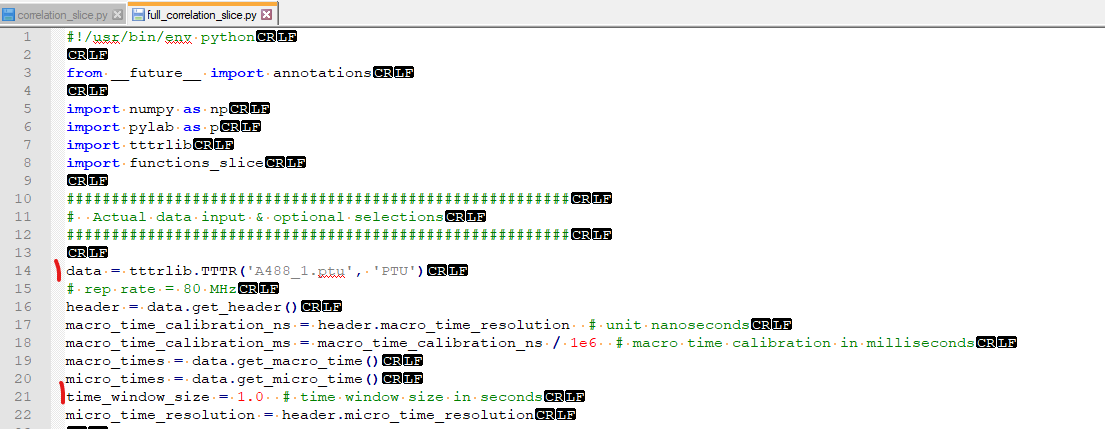
(katcorr) C:\Users\username\katcorr >

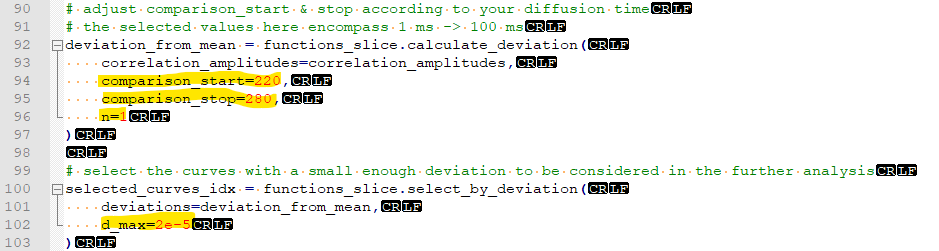
## Running your first correlation

As stated in the README-file and accompanying literature, this script slices long measurements in shorter pieces, correlates these pieces, calculates from each of these pieces a mean squared deviation (MSD) and finally selects all slices below a certain threshold value of MSD to generate an average correlation curve.

Prior to starting your first correlation you need to provide the following information:

* Filename (just name or complete path as “E:\path-to-data\filename.ptu “)
* The slice size (in seconds), which you would like your data to be cut
* Your comparison range, i.e. from which part of the curve you would your MSD to be calculated, usually the rise term-are is compared
  + Please be aware the comparison range is given in indices not time
  + Here, the given values for slowly diffusing membrane protein encompass the range from 220-280
* Number of curves to compare to, here: n = 1 -> all curves are compared to the first one
* Threshold value d\_max, all curves/slices with MSD below this values are accepted

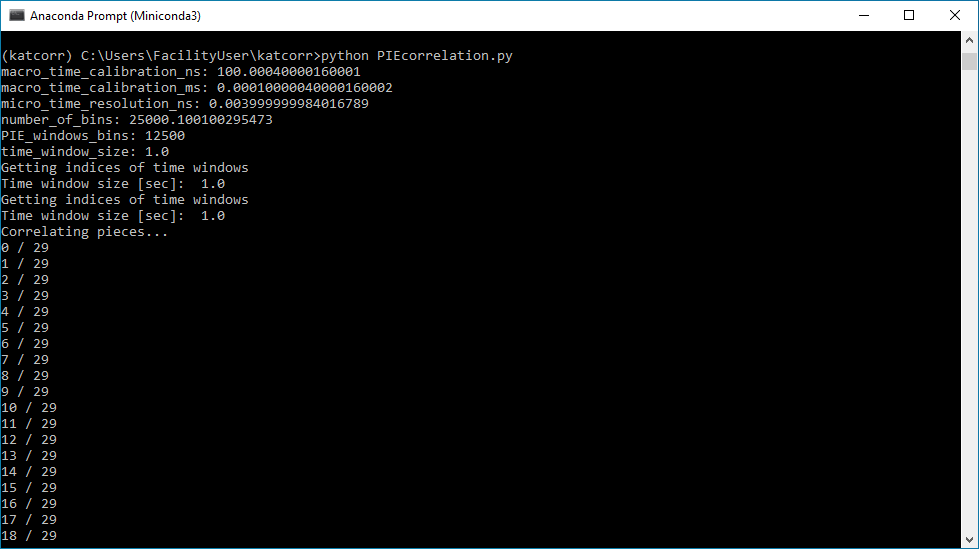




After you have made all your changes, save the script and go back to the console.

Run the “full\_correlation\_slice” script by typing:

(katcorr) C:\Users\username\katcorr > python full\_correlation\_slice.py



The console gives you an overview about the progress of how many slices of the total have already correlated.

Please be aware that if both auto- and crosscorrelation need to be computed this slicing process is repeated thrice: cross correlation as 1st step, followed by autocorrelation of channel 1 and autoocrrelation of channel 2. Please note, the selection of which curves to keep and average is based solely on the cross-correlation! I.e. for autocorrelations the same time-slices are selected as in the cross-correlation case.

After the correlations are done, an image is shown of

* The correlations
* The deviations
* The fluorescence intensity (count rate)
* (fluorescence anisotropy -> is polarization-resolved data is correlated)

The following data is saved:

* Deviations as txt-file
* Countrates: total - ch1 –ch2 – steady-state anisotropy (as txt file)
* Cross- and autocorrelations: time – correlation amplitude – suren\_column – standard deviation (as cor-file, ASCI file can be opened with Notepad)
  + Suren\_column: filled with 0’s for compliance with fitting with Kristine & ChiSurf