

Preparation Steps

In order for PRoNTo to work properly, other than having PRoNTo itself, you also need 2 other softwares in your computer, MATLAB and SPM12.

- A recent version of MATLAB (R2018a and later) is required. **Make sure to include the Statistics and Signal Processing toolboxes when installing MATLAB** as they are also required. Note: Windows and Mac users may encounter compiler issues when using R2016b and R2017b. Files may need to be compiled again. Users can install the MATLAB-required compiler from the MATLAB Add-Ons, or by manually downloading MinGW.
- You also need to have SPM12 installed on your computer. Users are recommended to have the latest update installed. So if you already have a version of SPM12 on your computer, please update it. Various bugs, especially in terms of weight visualization, arise from out-of-date SPM versions. You can download SPM from [this link](#). Fill in all the info required and make sure to choose SPM12 in the “SPM Version” section.

After you finished the steps above then you are ready to download PRoNTo. The latest public version can be downloaded, after registration, from the following [link](#). After downloading the zipped file containing PRoNTo, the installation proceeds as follow:

- Uncompress the zipped file in your favourite directory, for example C:\PRoNTo\.
- Launch MATLAB.
- Go to the HOME menu and click “Set path”.
- Click on the “Add with Subfolders” button and select the folder you added PRoNTo, i.e. C:\PRoNTo\ in our example.
- Click the “Save” button.

Some routines (such as external toolboxes used within PRoNTo) are written in C++ (.cpp files) for increased efficiency. We are trying to provide these compiled routines for the usual Operating Systems (OS's) such as Windows, Mac and Linux. If your OS is not listed or for any reason routines do not work properly then you should compile the routines for your specific OS.

Once installed, there are three ways to call up PRoNTo functionalities. To launch the toolbox GUI, just type “prt” or “pronto” at the MATLAB prompt and the main GUI figure will pop up. From there on simply click on the processing step needed. Most functions of PRoNTo have been integrated into the matlabbatch batching system (like SPM12) and the batching GUI is launched from the main GUI by clicking on the Batch button. Of course most tools can also be called individually by calling them directly from the MATLAB prompt, or for scripting in a .m file.

Troubleshooting compiled routines

All the external libraries required by PRoNTo are provided. All the required MEX files are also provided for 64 bit Windows, MacOS and Linux (Ubuntu/Debian) systems. However, if your system specifications do not align with those above, making new MEX files is necessary. In such cases you will most likely hit an error. Please follow the instructions below in case this happens.

Compiling LIBSVM: Some problems when using SVMs might also arise due to LIBSVM, in which case, you might need to compile it on your own. The first thing that needs to be done is to download the desired LIBSVM version (usually the latest one) from their [website](#). Then, the process will depend on your operating system. If the steps described below do not work, please refer to the README file that comes with LIBSVM.

- **Microsoft Windows:**

- Make sure you have a C++ compiler install. If not, you can install Microsoft Visual C/C++.
- Copy the libsvm folder to the “machines” directory of your PRoNTo installation. (e.g. C:\PRoNTo\machines\).
- Open a DOS command window and change to the libsvm folder in the previous step (`cd C:\PRoNTo\machines\libsvm-3.17\`). If the environment variables of VC++ have not been set, run the following command: `C:\Program Files\Microsoft Visual Studio 10.0\VC\bin\vcvars32.bat`. This command might be different, depending on the path of your Visual Studio installation.
- In the libsvm folder run the command: `nmake -f Makefile.win clean all`
- If no errors appear, open MATLAB.
- Change to the “matlab” folder inside the libsvm folder (e.g. C:\PRoNTo\machines\libsvm-3.17\matlab\).
- Run make in the MATLAB Command Window. If there are no errors, you have just successfully compiled LIBSVM to be used with MATLAB.

Remember, if you want to use the version that you have just compiled, you have to add the libsvm folder to your path in MATLAB. If you have more than one libsvm folder inside the “machines” folder, please remove one of them from the MATLAB path. You should only have one libsvm folder in your path.

- **Unix (Mac OS or Linux)**

- Make sure you have a C++ compiler installed. If you are using Mac OS, please install “Xcode”. On Linux systems, you should already have “gcc” installed.
- Copy the libsvm folder to the “machines” directory of your PRoNTo installation (e.g. /home/<username>/PRoNTo/machines/).
- Open a terminal window and change to the “machines” directory: `cd PRoNTo/machines/`
- Compile libsvm by running the following command: `make`
- If no errors appear, open MATLAB.

- Change to the “matlab” folder inside the libsvm folder (e.g. PRoNTo/machines/libsvm-3.17/matlab/).
- Run make in the MATLAB Command Window. If there are no errors, you have just successfully compiled LIBSVM to be used with MATLAB.

Remember, if you want to use the version that you have just compiled, you have to add the libsvm folder to your path in MATLAB. If you have more than one libsvm folder inside the «machines» folder, please remove one of them from the MATLAB path. You should only have one libsvm folder in your path.

Compiling GPML: In case there are compiler problems when using GPs due to GPML, you will need to compile it on your own. The instructions to do so are approximately the same as with LIBSVM.

- Inside MATLAB, change to the “util” folder inside the gpml folder (e.g. PRoNTo/machines/gpml/gpmlmatlab- v3.5-2014-12-08/util/).
- Run make in the MATLAB Command Window. If there are no errors, you have just successfully compiled GPML to be used with MATLAB.