

PyNanoLab软件操作手册

1. 简介

PyNanoLab是一个使用PySide6框架结合Python的科学计算生态构建的用户界面(GUI)软件。它旨在提供一个跨平台、现代化和易于操作的界面，用于数据的分析处理和数据的可视化。PyNanoLab从最初的用于纳米孔道离子电流信号的分析工具，逐渐发展为科学数据分析的综合性软件。

PyNanoLab 主要有以下几个主要特性:

- **跨平台支持:** PyNanoLab可以在多个操作系统上运行，包括Windows、Mac和Linux。
- **现代化界面:** 采用现代化的用户界面元素和交互方式，以提供良好的用户体验。
- **数据管理:** PyNanoLab提供了统一的数据管理功能，支持常见的数据格式，导入的数据，分析的结果和绘制的图表可以保存为统一的项目文件中，用以重复使用。
- **数据分析:** 软件内置了数据分析工具，包括数据可视化、统计分析和等功能，帮助用户探索和理解实验数据。
- **拓展工具箱:** : PyNanoLab提供了多种数据分析工具箱，添加扩展功能和插件，以满足特定的分析要求。

2. 安装

请参照在线教程：<https://pynanolab.com/html/install>

2.1 系统要求

为了在计算机上安装和运行PyNanoLab软件，您的系统需要满足以下最低要求

操作系统：任意带有GUI界面的64位 Windows、macOS和Linux发行版。

内存：建议至少8GB RAM。

PyNanoLab本身对系统的性能没有太大要求，但是当处理大量数据，特别是对高采样率的大量数据进行可视化的过程中需要消耗大量的内存和CPU资源。

PyNanoLab 使用了众多科学计算库和GUI库，因此可能会和当前系统python的模块冲突或者出现兼容性问题，因此建议在一个独立的虚拟环境中安装。首先，您应该已经在系统中安装了Python或Conda虚拟环境，建议使用Miniconda，并将Conda添加到系统环境变量中。在Windows系统中，您还需要安装一个终端程序。建议使用git-windows或Windows Terminal。

2.2 软件安装

- 首先，为PyNanoLab 创建独立的虚拟环境，在terminal中执行以下命令：

```
conda create -n pnl python=3.11.3
source activate
conda activate pnl # activate the pnl environment.
conda install numpy
```

在上述命令中，我们使用conda 创建了一个名为 *pnl* 的python虚拟环境，其中指定python版本为 3.11.3。接着激活 *pnl* 环境，并使用conda在此环境中安装mkl版本的numpy。

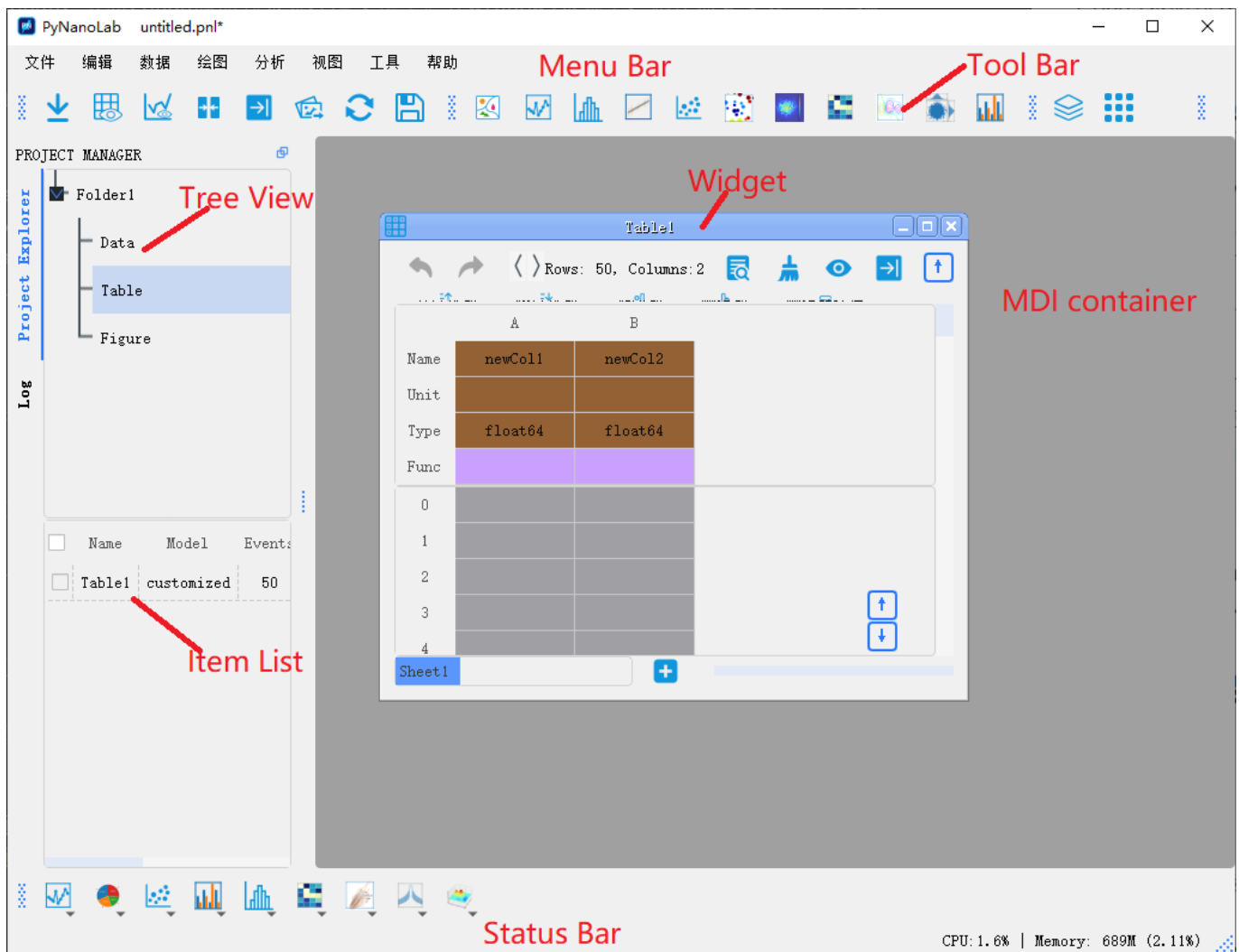
- 然后，使用pip 工具直接安装pynanolab和相关依赖

```
pip install --upgrade pynanolab # 更新或安装最新版本
```

至此，pynanolab 已安装在 *pnl* 虚拟环境中，在激活pnl的终端中直接执行

3. 基础操作

3.1 界面导览



PyNanoLab 的界面主要包括菜单栏、工具栏、项目管理和多文档管理器。

3.1.1 项目管理器

项目管理器用于管理软件的所有文件和对象，包含一个树浏览器和列表浏览器。PyNanoLab的项目文件以文件夹形式的字典进行管理，一个项目可以创建多个文件夹进行管理。其中每一个文件夹分别包含固定的Data、Table和Figure三个文件夹。

- **Data文件夹**: 管理采样数据, 数据形式为多通道多组的采样数据, 目前支持 Axon的.abf文件、LabVIEW的 .tdms 文件、和 LightField 的.spe光谱文件。
- **Table文件夹**: 管理所有生成的和外部导入的表格文件, 包括 文本文件(.txt)、.csv、.xlsx、.mat、.npy文件等。
- **Figure文件夹**: 管理所有的绘图对象。

在选中树浏览器的子文件夹后, 列表浏览器将会列出存储在该文件夹下的所有条目, 鼠标双击条目即可打开对应的窗口。

3.1.2 多窗口管理器

PyNanoLab 所有打开的窗口都显示在多文档管理器管理器中, 因此可以支持同时打开多个窗口。数据的操作和处理也在对应的窗口中进行。菜单栏的 **级联显示**、**平铺显示** 和 **最小化显示** 可以用于整理多文档管理器管理器窗口。右键菜单中列出了常用的管理命令, 主要包括文件的导入、导出和删除。双击条目的名称可以自定义修改当前项的名字。

在PyNanoLab中, **Data文件夹**和**Table文件夹**中所有的条目打开的窗口仅仅是**数据的一个视图**, 因此关闭窗口并不会删除任何数据, 因此在关闭此类窗口是没有关闭警告。**Figure文件夹**中的窗口即包含数据本身, 因此关闭图形窗口将会删除对应的绘图对象。

3.2 数据文件的管理

PyNanoLab构建了一个独立的文件用于存储整个项目的内容, 从而可以实现数据的重新加载和图形的重新绘制。*.pnl 是PyNanoLab的项目文件格式, 其实质是一个 ZIP 格式的压缩包, 在压缩包里面以文件夹的形式整理数据, 对应树浏览器的主文件夹的所有内容, 每一个主文件夹也以ZIP形式存储, 其后缀名为 *.pnlfd。对于子文件夹, **Data文件夹**和 **Figure文件夹** 使用PyTables的存储为 HDF5 文件, 可以实现大量采样数据的存储, 其后缀名分别为 *.pnldat和 *.pnlfigz。 **Table文件夹** 使用Pandas 的 HDFStore 接口存储为 HDF5文件, 其后缀名为 *.pnltbz。上述三个文件也可以从压缩包中解压出来单独的导入到PyNanoLab中, 软件也支持将每个子文件夹独立的导出。

```
# 一个PyNanoLab项目文件的基本结构
model.pnl
├── Folder1.pnlfd
│   ├── Data.pnldat
│   ├── Table.pnltbz
│   └── Figure.pnlfigz
├── Folder2.pnlfd
│   ├── Data.pnldat
│   ├── Table.pnltbz
│   └── Figure.pnlfigz
```

虽然, PyNanoLab的项目文件存储没有上限, 但仍然不建议在一个项目文件中存储大量的数据, 特别是包含特别多的原始数据, 从而增加存储和打开的时间以及出现错误的概率。建议单个文件不超过4GB的大小, 以免在某些文件系统中遇到位置的问题。

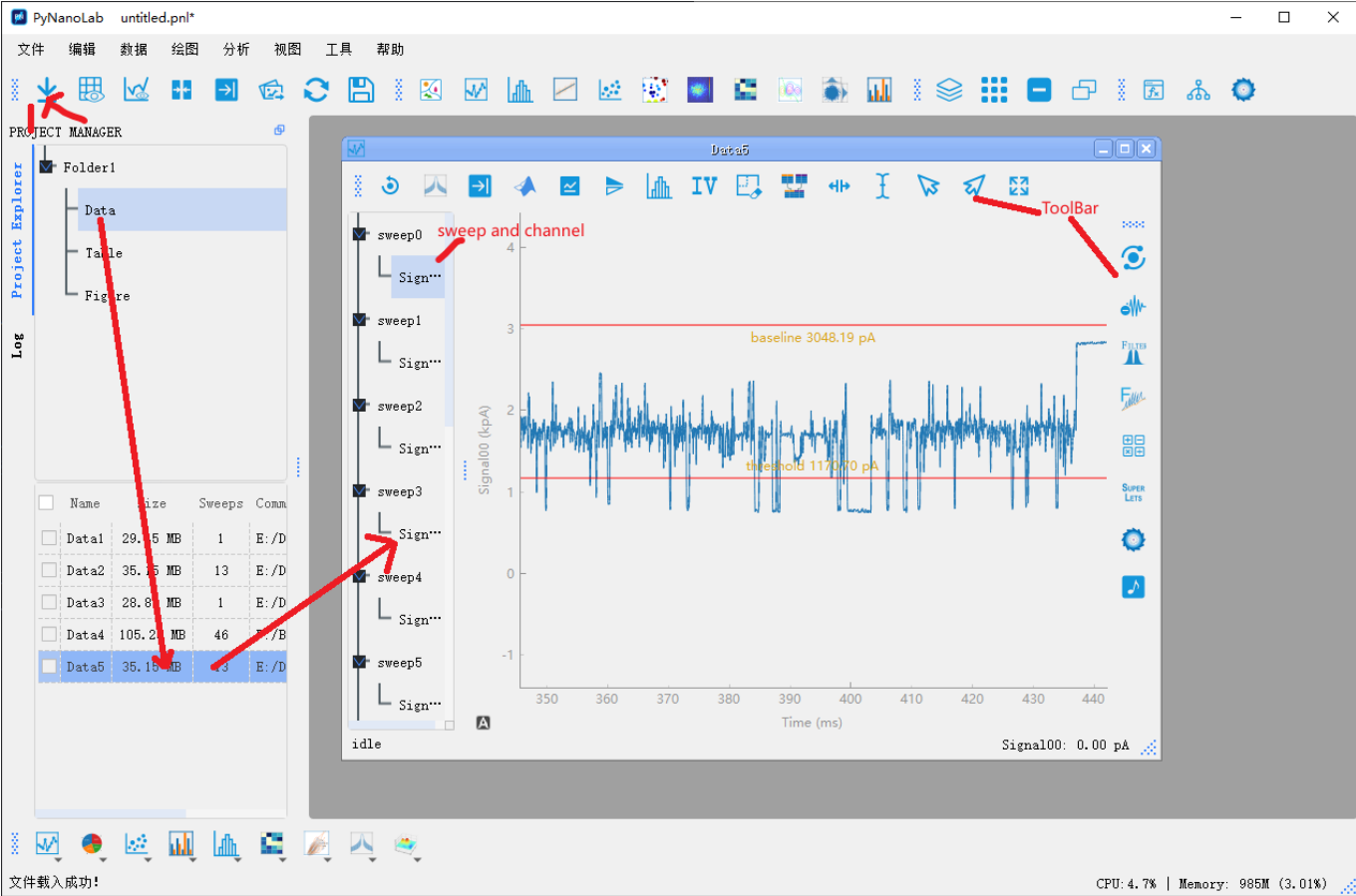
当打开项目文件出错的时候, 可是直接使用解压缩软件 (例如:7z) 将 *.pnl内的文件解压出来, 单独的导入到软件中, 或者直接编写程序查看和导出数据。

4. Data模块

PyNanoLab根据导入的数据类型价格数据放入不同的文件夹进行处理，Data文件夹主要处理采集的各种非表格结构的数据。通过工具栏的 导入 按钮或者直接拖曳文件到树浏览器即可打开文件。

导入数据文件后，双击该文件的条目即可创建数据的视图窗口，在PyNanoLab中，采样数据的可视化使用 Pyqtgraph 框架构建,可以实现数据的高速显示。

4.1 采样数据的处理



在PyNanoLab中，软件的文件IO接口会自动从文件读取所需的采样率、滤波、单位、名称等信息，ABF等电生理的采样数据由sampledatawidget进行处理，如上图所示，主要包括：

- **数据浏览器** 在左侧的树浏览器中，列出了该数据文件所有的sweeps和在每一个sweep下的通道，鼠标选中对应通道，即可在绘图区域显示该通道的数据。
- **数据可视化** 数据显示区域使用Pyqtgraph的lineplot绘制，图形区域支持3点鼠标操作。在图形上点击右键保持，拖曳即可实现图形的缩放和位移。在绘图区，分别添加了两个纵向的cursor，用于辅助选择图形中的数据。
- **工具栏** 包含所有的用于数据处理的工具：
 - : 重置数据，窗口保留了一份原始数据的备份。可以点击此按钮将数据重置为初始状态。
 - : 计算并绘制光标区间内数据的功率谱密度。
 -

5. Table模块

- 自定义界面布局
- 扩展功能和插件

- 脚本编写和执行

6. 可视化模块

- 常见错误和解决方法
- 常见功能使用问题解答

7. 工具箱

- 联系我们
- 提交反馈
- 获取帮助

8. 高级功能

- 版本更新内容

9. 常见问题解答