

FUNKCIONÁLIS SPECIFIKÁCIÓ

NTI Wavelet Preprocessing Tool

Nyáry Anna
2015

Tartalomjegyzék

A program célja	1
Technikai részletek	1
Python-IDL bridge	1
Felhasználói felület	2
Adatstruktúra és adatkezelés	2
Ábrázolás	3
Időtolás és átmintavételezés	4
Kimenet	5
Szkriptelhetőség	5
Fejlesztési lehetőségek	5

A program célja

A Python nyelven írt program mérési adatok előkészítését végzi. Az adatbeolvasás különböző formátumú és eredetű fájlok kezelésére képes. A rendelkezésre álló adatsorból tetszőlegesen sok pont kiválasztható az elemzésre, adatpontok száma vagy adott időtartam alapján. A feldolgozott adatok végül átadhatók az NTI Wavelet Tools programcsomagba további kiértékelésre.

Technikai részletek

A programkód a JetBrains PyCharm Educational Edition 1.0.1 fejlesztői környezetben Python nyelven íródott, és Anaconda3 2.1.0 64bit interpretert használ.

A modul C++ alapú Qt keretrendszer felhasználásával készült, ugyanis a grafikus felhasználói felület (GUI) tervezése Qt Designer programmal történt. A GUI tervező programmal Qt objektumokból egyszerűen megalkotható az elképzelt ablak vagy párbeszédpanel. A létrehozott felületet ui kiterjesztéssel lehet menteni. A szkriptben felhasznált PyQt4 csomag ugyancsak a Qt keretrendszeren alapul, annak Python nyelvű felhasználhatóságát realizálja.

A Python szkript létrehozásakor olyan osztályt hozunk létre, amely öröklí a megfelelő PyQt-beli osztály tulajdonságait, QWindow(ablak osztály) vagy QDialog (párbeszédpanel osztály). A Qt Designerben létrehozott ui fájl betöltésével rendelkezésre áll a megtervezett GUI minden részletével.

A programban felhasznált csomagok:

- **QtGui, QtCore:** PyQt4-ből importált csomagok a GUI-val kapcsolatos osztályokat tartalmazzák
- **uic:** PyQt4-ből importált csomag az ui fájlok kezelésére
- **easygui:**
- **scipy.io**
- **numpy**

A modulhoz tartozó py fájlok:

Python-IDL bridge

Az IDL 8.5-ös verziójával megjelent a Python és az IDL közötti kommunikáció lehetősége.

nti_wavelet_preprocessing_data_class.py
<i>Data</i> nevű osztály kezeli adatsort. Jellemzői az adatsor és metaadatai. Metódusai: átmintavételezés, ...
nti_wavelet_preprocessing_data_logic.py
<i>DataLogic</i> nevű osztály kezeli az adatsorlistát. Jellemzője az adatsorok listája. Metódusai: beolvasás, NTI Wavelet Tools meghívása.
nti_wavelet_preprocessing_gui_logic.py
A <i>DialogWithLogic</i> osztály a felhasználói felület fő ablakát kezeli.
nti_wavelet_preprocessing_readtext_logic.py
A <i>ReadTextDialog</i> osztály a szövegfájl beolvasásakor felugró ablakot hozza létre.

Felhasználói felület

A modul rendelkezik felhasználói felülettel, ugyanakkor szkriptelhető is, azaz a felhasználó számára a parancssorból is használható. Ennek megfelelően a grafikus felület elkülönített és független az adatkezeléstől.

Adatstruktúra és adatkezelés

Az adatstruktúrát a *Data* osztállyal hozzuk létre. Egy beolvasott adatsor a *Data* egyik példánya, így a metaadatok az osztály attribútumaiként (jellemzőiként) kerülnek tárolásra. Az adatsorral végzett időtolási és átmintavételezési eljárásokat ?osztálymetódusokként (osztályfüggvényekként)? definiáljunk.

A *Data* példányainak listáját a *DataLogic* osztály kezeli. Ez azt jelenti, hogy a *DataLogic* egy példánya a beolvasott adatsorok listáját hozza létre. A *DataLogic* osztály metódusa az adatbeolvasás, és az adatok az NTI Wavelet Tools programnak való továbbítása.

A modul hasznos funkciója, hogy az eddigiekben is gyakran használt *get_rawsignal.pro* IDL nyelven írt ?szkripttel? kommunikálni képes. A *get_rawsignal.pro* az NTI Wavelet Tools adatbeolvasó modulja, mellyel különböző berendezések jelei elérhetőek. A különböző nyelven írt programok közötti átjárást a korábbiakban bemutatott Python-IDL bridge szolgáltatja.

Az adatbeolvasás másik lehetséges módja a mat és txt kiterjesztéssel rendelkező fájlok betöltése. Egy fájl egy adat- és idősort tartalmazhat, több

adatsor párhuzamos beolvasására nincs lehetőség. A bináris és szöveg fájlok beolvasása különbözően történik.

A szövegfájlok beolvasásakor az adatelválasztó (delimiter) és a fejléc mérete (headersize) megadható. A felhasználói felületben egy felugró ablakban adhatjuk meg az értékeket. Az átláthatóság érdekében a felugró ablakban a fájl tartalma közvetlenül megjelenik.

Bináris fájlban lehetőség van a metaadatok előzetes megadására. Amennyiben a metaadatok a megfelelő elnevezéssel rendelkeznek, a program automatikusan be tudja olvasni őket. A megadható metaadatok:

experiment_name	
shot_number	
channel_name	
coordinates_history	
data_history	
sampling_time	
sampling_frequency	
nyquist_frequency	
n_data_points	

A felhasználói felületen a beolvasott adatsorok megjelennek a Data panelben, ahol adatsor névre kattintva lehetőség van az adatsor nevének és a metaadatoknak a megadására, illetve szerkesztésére. A metaadatok mellett a program által számított mintavételezési frekvencia is megjelenik. Az adatsorhoz rendelt metaadatok a későbbi kezelhetőség céljából, több beolvasott fájl egyértelmű elkülönítésére szolgálnak. Az adott adatsorhoz tartozó ?x? gombbal kitörölhetjük a listából, a checkbox bejelölésével pedig kiválaszthatjuk ábrázolásra.

Ábrázolás

A modul egyik hasznos funkciója a felhasználói felületen beolvasott adatok vizuális megjelenítése. Az ábrázolás az adatsorok áttekinthetőségét szolgálja. A Data panelben az ábrázolandó adatsorok a megfelelő checkboxra való kattintással kijelölhetők. A grafikonon az adatsorok többféleképpen ábrázolhatók, az opciók: egy grafikonon (*'Plot together'*), egymás alatt (*'*"), mátrixban (*'Plot in matrix'*). Az opció kiválasztását követően az ábrák a Plot gomb lenyomása után jelennek meg.

Időtolás és átmintavételezés

Az adatsorokból elemzésre szánt pontok elsősorban a hozzájuk tartozó időintervallum meghatározásával választhatóak ki. Az intervallumkiválasztás módjai:

- kezdő- és végidőpont meghatározása,
- kezdőidőpont és adatok számának megadása,
- grafikus kijelölés.

Az intervallumkiválasztás során csak létező időpont adható meg, és csak olyan időintervallum adható meg, ami mindegyik elemzésre szánt jelben jelen van.

Több idősor esetén felmerülő probléma a közös időalap megválasztása, melyet átmintavételezéssel oldunk meg. Az intervallumot egy adatsorban határozzuk meg, ezért szükség lehet a többi adatsor időbeli eltolásához a kezdőidőpontok egyezése céljából. Az időtolás és az átmintavételezés egy rutinon belül fut. Egy kívánt idősor meghatározása után a program a nem megfelelő időalapú adatsorokból újat készít, így végül az idősorok azonos időalappal jeleníthetők meg. A t kezdőidőpontot és az f mintavételezési frekvenciát vagy az n adatpontok számát a felhasználó adja meg. Az egyező kezdőidőpont beállításához minden idősorban megkeressük a t -hez legközelebb eső pontot $t + \Delta t$ időpontban. Az adott idősorban N adatpont található, fs a mintavételezési frekvenciája. Szükségünk van a **phase** vektorra, ami megadja, hogy adott frekvencián a Δt időeltolás mekkora fázist jelent.

$$\mathbf{phase} = \Delta t \cdot \frac{\boldsymbol{\nu}}{2} \cdot \pi = \mathit{fraction} \cdot \frac{\boldsymbol{\nu}}{fs} \cdot \pi,$$

ahol $\boldsymbol{\nu}$ egy $N/2$ hosszúságú vektor, melynek első eleme 0, utolsó fs , köztük az elemek egyenletes eloszlással helyezkednek el; a $\mathit{fraction}$ pedig a $\Delta t/ts = \Delta t \cdot fs$ arány. A Fourier-transzformáltat $\exp(-i \cdot \mathbf{phase})$ -vel kell beszorozni.

$$\begin{aligned} \text{FFT}(x) &= xfft \\ xfft &= xfft \cdot \exp(-i \cdot \mathbf{phase}). \end{aligned}$$

Az átmintavételezéshez az adatsor frekvenciáját kell megvizsgálni a további eljárások kivitelezéséhez. Amennyiben az fs mintavételezési frekvencia nagyobb, mint a kívánt f frekvencia, a Fourier-transzformált jelben az $f/2$ frekvenciánál nagyobb frekvenciakomponenseket levágjuk. Ha fs kisebb,

mint f , akkor a Fourier-transzformáltban az $[fs/2, f/2]$ intervallumot kinulázzuk. Így mindkét esetben a Fourier-transzformált legmagasabb frekvenciája $f/2$ lesz. Az inverz Fourier-transzformáció után a jelünk mintavételezési frekvenciája valóban f lesz.

$$xx = \text{FFT}^{-1}(xxfft)$$

Kimenet

A beolvasott adatsorok az átmintavételezés után új fájlba mentésére vagy az NTI Wavelet Tools programcsomagba való importálásra is lehetőség van. A feldolgozott adatok sav, mat, txt, wav kiterjesztéssel menthetők új fájlba.

Szkriptelhetőség

Az alábbi rész a modul felhasználói felület nélküli használatát mutatja be.

Fejlesztési lehetőségek