

Adatfeldolgozó programcsomag fejlesztése



EGYENLETES ÉS NEM EGYENLETES JELEK KÖZTI
ÁTTÉRÉS

Egyetemi konzulens: Dr. Orosz György
Külső konzulens: Tóth Katalin

Készítette: Witzl Ádám
witzladam@edu.bme.hu

Összefoglalás

- Feladat leírása
- Python
- Jelek reprezentációja
- Osztálydiagram
- Nem egyenletes mintavételezés
- Jelek közötti áttérés
 - Interpoláció
- Előrelépési lehetőség

Feladat leírása

```
167 def get_derivate_signal(self):
168     raise NotImplementedError("Implement in subclass")
169
170 class AnalysisEQ(AnalysisMETA):
171     def get_average(self, t1=np.NINF, t2=np.PINF):
172         """This function calculates the average between t1 and t2 times"""
173         if _np.isfinite(t2) and t2 >= self._signal_object.parameters:
174             if _np.isfinite(t1) and t1 <= self._signal_object.parameters:
175                 windowed_signal = self.get_window(t1, t2, 'interp')
176                 if isinstance(windowed_signal, self._signal_object.__class__):
177                     return self._cython_package.calculate_avg_EQ(windowed_signal)
178                 else:
179                     return windowed_signal.analysis.get_average()
180             raise NotImplementedError("Implement in subclass")
181
182 def get_rms(self, t1=np.NINF, t2=np.PINF):
183     """This function calculates the rms between t1 and t2 times"""
184     if _np.isfinite(t2) and t2 >= self._signal_object.parameters:
185         if _np.isfinite(t1) and t1 <= self._signal_object.parameters:
```

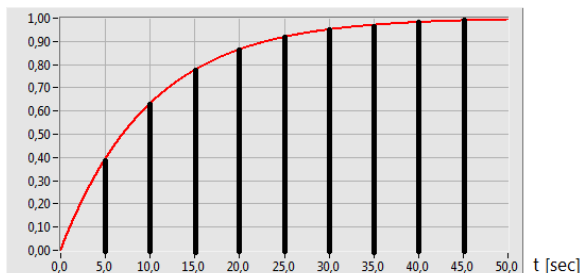


adatfeldolgozó programcsomag fejlesztése

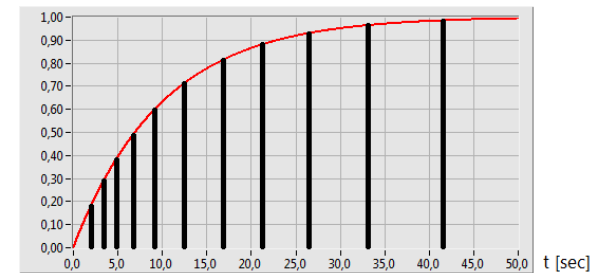
funkciók:

- kijelzés
- szűrés
- éldetektálás
- digitalizálás
- RMS, AVG számítás
- stb.

Egyenletesen mintavételezett



Nem egyenletesen mintavételezett





Python

- szkript nyelv
- jelenlegi verzió: 2.7

felhasznált library-k:



- tömbök, mátrixokkal végezhető műveletek
- NumPy tömb objektuma az ndarray
- sokkal gyorsabb, mint a hagyományos Python listákkal végezhető műveletek



- használja a NumPy könyvtárat
- univerzális megoldásokat nyújt optimalizációs, statisztikai és jelfeldolgozási problémákra

unittest

- tesztesetek előállítására

Jelek reprezentációja

	időreprezentáció	jelreprezentáció
egyenletesen mintavételezett	• offset • Δt • pontok száma	• tömb
nem egyenletesen mintavételezett	• tömb	• tömb

Osztálydiagram

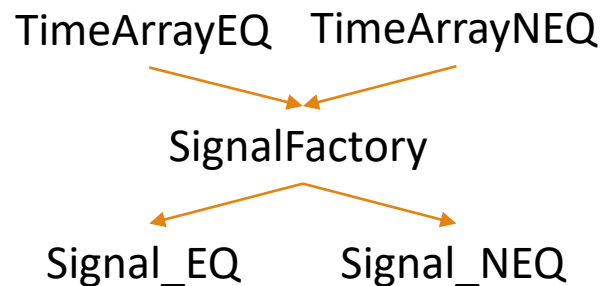
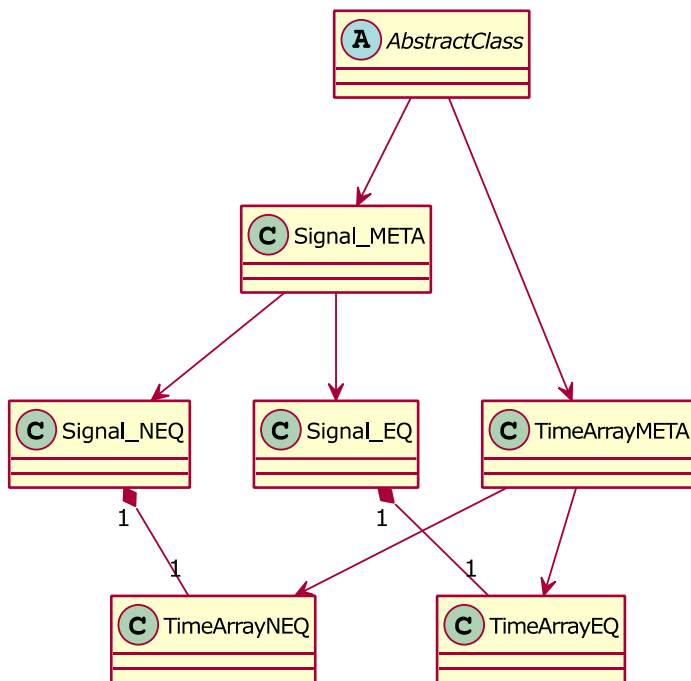
Absztrakt osztály

közös interfészt biztosít az eltérően mintavételezett jelekhez

Tartalmazás

egy jelhez pontosan egy időreprezentáció tartozik

Factory method



Nem egyenletes mintavételezés

Előfordulása

- eseményvezérelt mintavételezés
- beágyazott rendszerekben
- ahol nincs szükség egyenletesen mintavételezett jelre

Előnye

- kevesebb memóriaigény
- nincs felesleges adat, ha jó időpontokban történik a mintavételezés

Jelek közötti áttérés

➤ Nem egyenletes → egyenletes

- interpoláció(lineáris, polinom)

➤ Egyenletes → nem egyenletes

- nem szignifikáns adatok eldobása
- minél nagyobb különbség a megengedett annál jobban lehet tömöríteni

Interpoláció

Lineáris interpoláció

- mért értékek összekötése, egy-egy szakasszal
- legegyszerűbb

Polinom interpoláció

- leggyakrabban klasszikus polinommal: $\Psi_k(x) = x^{k-1}$
- Lagrange-interpoláció konstruktív eljárás
- Neville-féle iterált interpoláció

Python: `scipy.interpolate` ➡

- 1-D interpolation(`interp1d`)
- Spline interpolation(`interpolate.splXXX`)

További feladatok

Python 2.7-ről áttérés Python 3.9-re

- algoritmusok fejlesztése
- további funkciók hozzáadása
- műveletek definiálása egyenletesen és nem egyenletesen mintavételezett jelek esetére