

## Obrada informacija: Prva laboratorijska vježba

Rijeke imaju veliki gospodarski značaj, od termoelektrana i nuklearnih elektrana kojima su rijeke nužne za rad, javne vodoopskrbe do poljoprivrede i transporta. Premda ljudi danas uvelike upravljaju vodotocima, uslijed vremenskih prilika ipak može doći do poplava, erozija tla i sličnih prirodnih nepogoda. Kako bismo se mogli na vrijeme pripremiti za potencijalne probleme, potrebno je istražiti ponašanje rijeka na temelju višegodišnjih praćenja vodostaja. U ovoj laboratorijskoj vježbi ćete analizirati vodostaje rijeke Save kroz period od 25 godina.

Unutar komprimiranog direktorija u kojem se nalazi ova bilježnica, nalazi se i .mat datoteka s podacima o vodostaju rijeke Save u periodu od 1.1.1982. do 31.12.2007. Svaki podatak označava jedan dan mjerenja vodostaja.

Vježba se izvodi u Pythonu/Google Colabu, a sve naredbe potrebne za provedbu vježbe te njihova objašnjenja dana su u predavanju. Ova laboratorijska vježba nosi 3 boda.

Kad ste gotovi s vježbom, na *Moodle* postavite .pdf izvješće s vježbe (.pdf izvješće možete generirati s naredbom File-Print-Location: PDF).

### 1. Učitajte biblioteke: NumPy, matplotlib.pyplot, scipy.fft, signal iz scipy te pywt.

```
# Ovo je mjesto na kojem možete izvoditi svoj kod.  
import numpy as np  
import matplotlib.pyplot as plt  
from scipy import io, signal  
from scipy.fft import fft, ifft  
import pywt
```

### 2. Učitajte podatke o vodostaju iz .mat datoteke koja se nalazi unutar istog komprimiranog direktorija. Učitani podaci su spremljeni u rječnik. Izdvojite ključ i pripadne podatke o vodostajima te po potrebi smanjite dimenziju.

Indented block

Savjet: .mat datoteku učitajte sa svog Google Drivea (unutar mape Colab Notebooks na svom Google Driveu postavite .mat datoteku). Ostali savjeti nalaze se u sljedećem programskom odsječku.

```
# Ovo je mjesto na kojem možete izvoditi svoj kod.
```

```
from google.colab import drive  
drive.mount('/content/drive')
```

```
# Nakon što pokrenete ovaj odsječak koda, pojavit će se link u prozoru ispod. Klikn  
# dobiti kod koji trebate kopirati u izlaz ispod ovog prozora (u prozor "Enter your  
# pojavit će se poruka "Mounted at /content/drive".
```

```
# Ako unutar takvog Colab Notebook direktorija imate spremljenu datoteku "OIkobas.m
from scipy import io
kobas_iz_matlaba=io.loadmat("drive/My Drive/Colab Notebooks/OIkobas.mat")
print(kobas_iz_matlaba)
kobas = kobas_iz_matlaba['kobas']

Drive already mounted at /content/drive; to attempt to forcibly remount, call
{'__header__': b'MATLAB 5.0 MAT-file, Platform: MACI64, Created on: Sat Oct 10
[674],
[685],
...,
[242],
[230],
[228]], dtype=uint16)}
```

Double-click (or enter) to edit

Double-click (or enter) to edit

**3. Ispišite:** broj dana za koje postoje podaci o vodostajima, srednju vrijednost vodostaja, standardnu devijaciju, minimalnu vrijednost, maksimalnu vrijednost te median.

```
# Ovo je mjesto na kojem možete izvoditi svoj kod.
kobas = np.squeeze(kobas)
#broj dana za koje postoje podaci
print("broj dana za koje postoje podatci o vodostajima: ", len(kobas))
#srednja vrijednost vodostaja
print("srednja vrijednost vodostaja: ", np.mean(kobas))
#standardna devijacija
print("standardna devijacija: ", np.std(kobas))
#minimalna vrijednost
print("minimalna vrijednost: ", np.min(kobas))
#maksimalna vrijednost
print("maksimalna vrijednost: ", np.max(kobas))
#medijan
print("medijan: ", np.median(kobas))
```

```
broj dana za koje postoje podatci o vodostajima: 9496
srednja vrijednost vodostaja: 294.25758213984835
standardna devijacija: 183.09167165945507
minimalna vrijednost: 17
maksimalna vrijednost: 878
medijan: 255.0
```

```
# This is formatted as code
```

**4. Nacrtajte vodostaj. Obilježite x i y os te naslov slike.**

```
# Ovo je mjesto na kojem možete izvoditi svoj kod.
```

```
plt.figure(figsize=(15,5))
plt.plot(kobas)

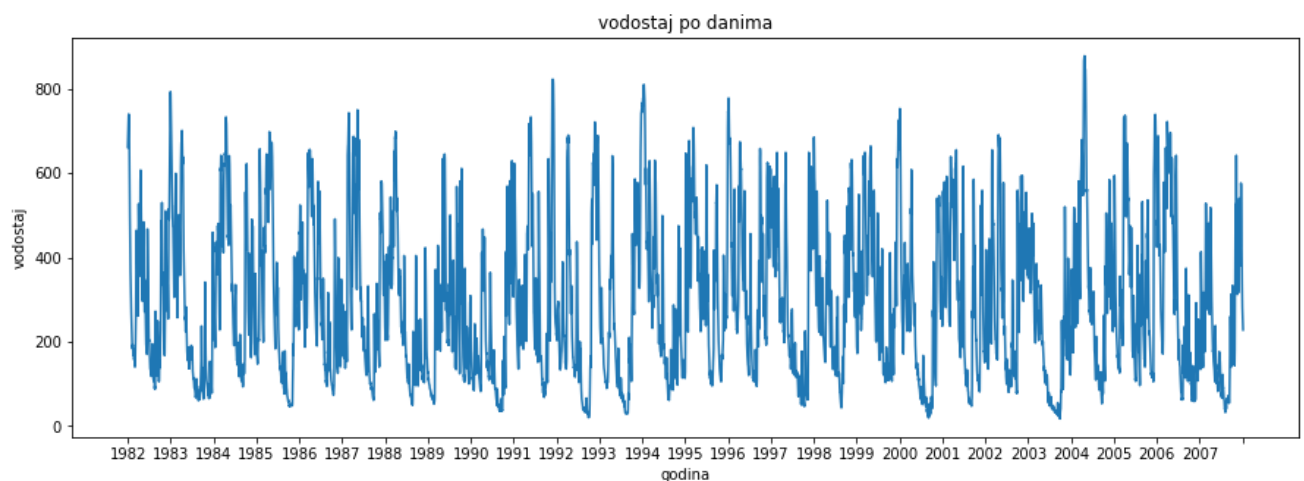
#plt.xticks(1982, 2007)

#postavljanje imena x i y osi
plt.xlabel('godina')
plt.ylabel('vodostaj')

pocetnaGod = 1982
plt.xticks(np.arange(0, kobas.size, 365), np.arange(pocetnaGod, int(pocetnaGod + ko

#postavljanje naslova
plt.title('vodostaj po danima')

plt.show()
```



5. Izračunajte diskretnu Fourierovu transformaciju zadanog vodostaja. Ispišite dobivene vrijednosti. Nacrtajte apsolutnu vrijednost dobivenih koeficijenata.

[link text](#)

```
# Ovo je mjesto na kojem možete izvoditi svoj kod.
```

```
#fft - diskretna fourierova transf
#stft - f. transformacija na vremenskom otvoru
x = np.array(kobas)
kobasDFT = fft(x)
print(kobasDFT)

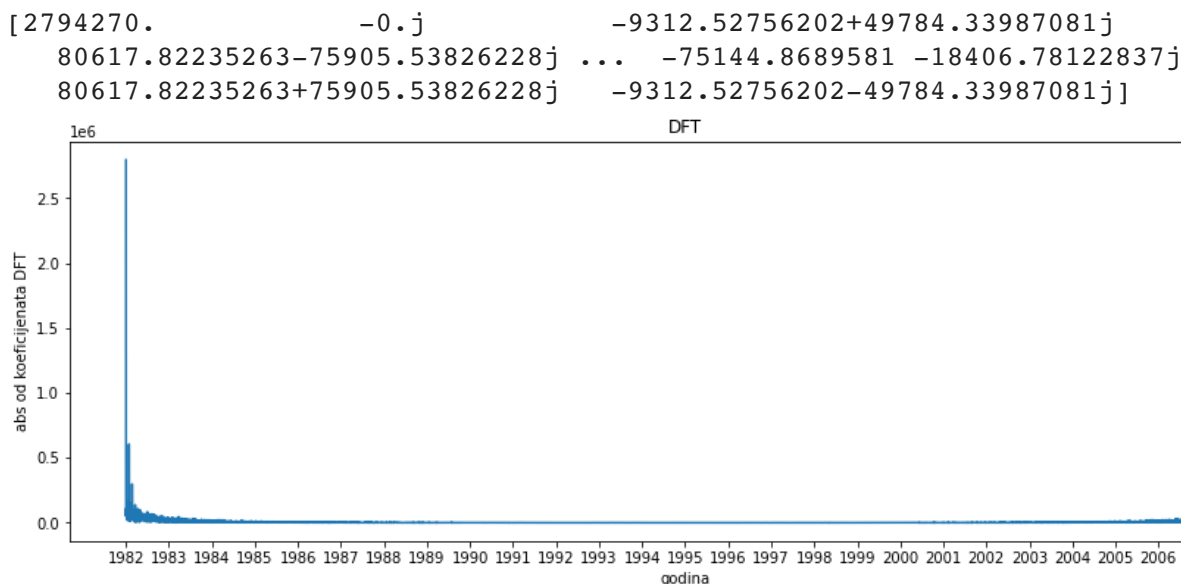
plt.figure(figsize=(15,5)) #za crtanje

#nacrtaj apsolutnu vrijednost od ft
plt.plot(abs(kobasDFT)) #plot -> KONTINUIRANI graf
```

```
#navedi imena osi i naslov
plt.title('DFT')
plt.xlabel('godina')
plt.ylabel('abs od koeficijenata DFT')

pocetnaGod = 1982
plt.xticks(np.arange(0, kobas.size, 365), np.arange(pocetnaGod, int(pocetnaGod + ko

#za prikaz uvijek
plt.show()
```



**6.** Izračunajte Fourierovu transformaciju na vremenskom otvoru zadanog vodostaja koristeći pravokutni otvor širine 2 godine. Prikažite rezultat pomoću pcolormesh. Odgovorite: koje frekvencije su vidljive u vodostaju? Koje godine se javljaju?

```
# Ovo je mjesto na kojem možete izvoditi svoj kod.

#fft - diskretna fourierova transf
#stft - f. transformacija na vremenskom otvoru

#postavis T na jednu godinu - broj uzoraka signala
T = 365

#postavis frekveniciju ocitavanja signala na 365 - svaki dan
fs = 365

#postavi pravokutni otvor
window = 'boxcar'

#f - niz ocitanih frekvenicija
```

```

#t - niz vremena
# Zxx - STFT od x
f,t,Zxx = signal.stft(kobas, fs, window, nperseg=2 * T)
#nperseg - duljina otvora -> 2T jer su 2 godine u pitanju

plt.figure(1)
plt.subplot(2, 1, 1)

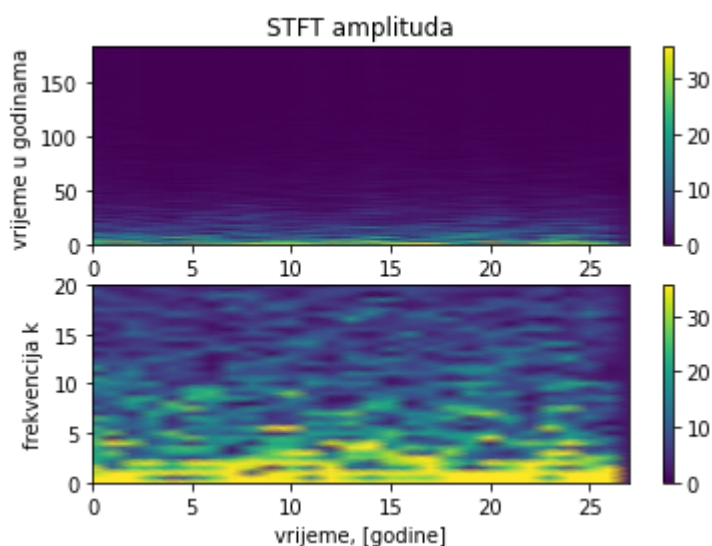
#bojanje...
plt.pcolormesh(t, f, np.abs(Zxx), vmin = 0, vmax = np.abs(Zxx.max()) / 10, shading='

plt.colorbar()
plt.title('STFT amplituda')
plt.xlabel('frekvencija k')
plt.ylabel('vrijeme u godinama')

plt.subplot(2, 1, 2)
plt.pcolormesh(t, f, np.abs(Zxx), vmin = 0, vmax = np.abs(Zxx.max()) / 10, shading
plt.colorbar()
plt.xlabel('vrijeme, [godine]')
plt.ylabel('frekvencija k')
plt.ylim(ymax = 20, ymin = 0)

plt.show()

```



ODGOVOR: Ovo je mjesto za odgovor na pitanja iz 6. zadatka.

1. U ovom primjeru frekvencije možemo očitati precizno. U vodostaju su vidljive frekvencije 1 (svako godinu dana), 2 (svako pola godine), te se za 4 (svako godisnje doba) vidi također. To sve možemo vidjeti po žutoj boji koja označava gdje je frekvencija "jača".
2. Mi imamo od 1982. do 2007., a prema grafu možemo vidjeti da je najveći vodostaj bio prve dvije godine mjerenja, te 1990, pa onda 1998. do 2006. sa jednom malo sušnijom godinom a npr oko 1986. (+2 godine) je bilo sušno razdoblje.

7. Upotrijebite drugu širinu otvora (po izboru) i ponovite prethodni zadatak. Ispišite koju širinu ste upotrijebili. Objasnite razlike u dobivenim slikama.

```
# Ovo je mjesto na kojem možete izvoditi svoj kod.

#isto kao 6, samo smo ovdje koristili okvir od pola godine
f, t, Zxx = signal.stft(kobas, fs, window, nperseg = 0.5 * T)
plt.figure(1)

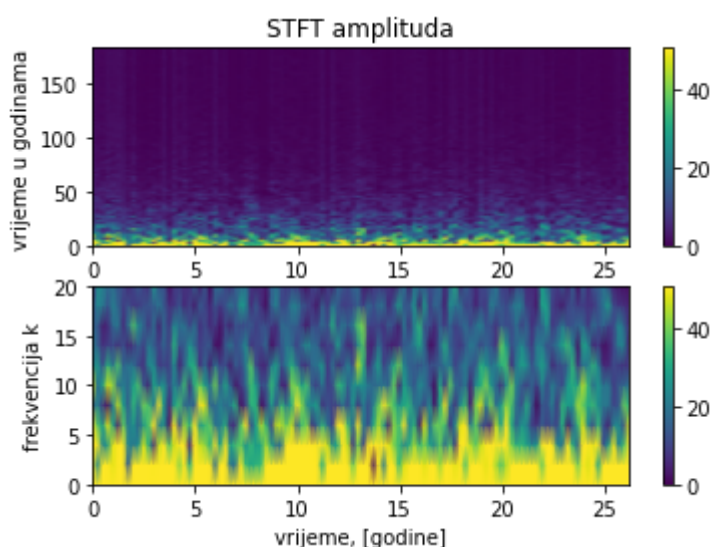
plt.subplot(2, 1, 1)

#bojanje...
plt.pcolormesh(t, f, np.abs(Zxx), vmin = 0, vmax = np.abs(Zxx.max()) / 10, shading='

plt.colorbar()
plt.title('STFT amplituda')
plt.xlabel('frekvencija k')
plt.ylabel('vrijeme u godinama')

plt.subplot(2, 1, 2)
plt.pcolormesh(t, f, np.abs(Zxx), vmin = 0, vmax = np.abs(Zxx.max()) / 10, shading
plt.colorbar()
plt.xlabel('vrijeme, [godine]')
plt.ylabel('frekvencija k')
plt.ylim(ymax = 20, ymin = 0)

plt.show()
```



ODGOVOR: Ovo je mjesto za odgovor na pitanje iz 7. zadatka.

U ovom smo zadatku stavili  $0.5 * T$  širinu otvora (4 puta manju) i možemo primjetiti da se preciznije mogu očitati trenutci u vremenu, ali manje precizno frekvencije. Što se tiče rezultata,

zamućeniji je i nejasniji prikaz od prethodnog, ali možemo bolje vidjeti sušnija, tj. vlažnija razdoblja (godine)... od 1983. do 1988. je bilo vlažno, pa je recimo oko 1989. bilo jkako suho, onda sljedećih par godina opet vlažno...

**8.** Ispišite sve obitelji kontinuiranih valića koje se nalaze u PyWavelets biblioteci. Na istoj slici, koristeći subplot naredbu, nacrtajte dvije valićne funkcije po izboru. U naslovu svake slike napišite o kojim valićima se radi.

```
# Ovo je mjesto na kojem možete izvoditi svoj kod.
#prvo ispisujemo sve obitelji "kontinuiranih" valića

lista = pywt.wavelist(kind='continuous')
print('Kontinuirani valići: ')
for i in lista:
    print('    -', i)

#1. izabrat ćemo cgau1
valic1 = 'cgau1'
#2. izabrat ćemo shan
valic2 = 'shan'

#graf za prvi
w = pywt.ContinuousWavelet(valic1)
#print('INFO O CGAU1')
#print(w)
psi,t = w.wavefun(level=10)

plt.figure(1)
plt.subplot(1,2,1)
plt.plot(t, psi)
plt.title(valic1)

#graf za drugi valić
w = pywt.ContinuousWavelet(valic2)
#print('INFO O SHAN')
#print(w)
psi,t = w.wavefun(level=10)

plt.subplot(1,2,2)
plt.plot(t, psi)
plt.title(valic2)

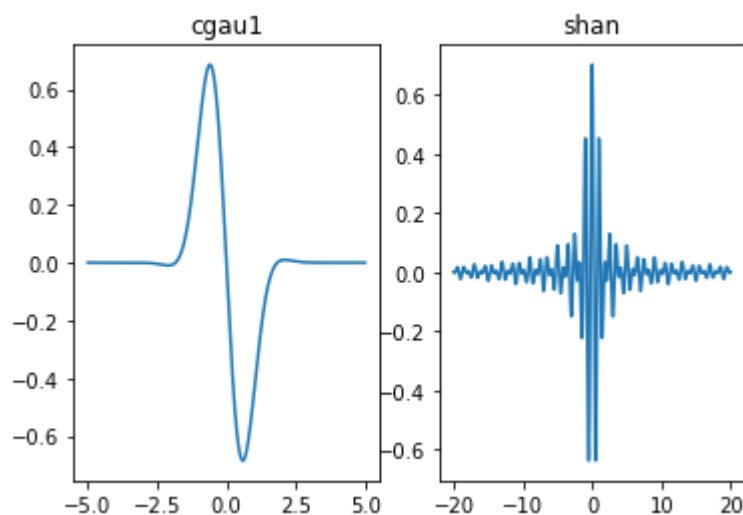
plt.show()

print()
print()
```

☞ Kontinuirani valići:

- cgau1
- cgau2
- cgau3
- cgau4
- cgau5
- cgau6
- cgau7
- cgau8
- cmor
- fbsp
- gaus1
- gaus2
- gaus3
- gaus4
- gaus5
- gaus6
- gaus7
- gaus8
- mexh
- morl
- shan

```
/usr/local/lib/python3.7/dist-packages/numpy/core/_asarray.py:83: ComplexWarning:
  return array(a, dtype, copy=False, order=order)
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:26: FutureWarning:
/usr/local/lib/python3.7/dist-packages/numpy/core/_asarray.py:83: ComplexWarning:
  return array(a, dtype, copy=False, order=order)
```



9. Koristeći Morlet valić, odredite kontinuiranu valićnu transformaciju zadanog vodostaja.

Nacrtajte apsolutnu vrijednost dobivenih koeficijenata. Odgovorite: koje frekvencije su vidljive u vodostaju? Koje godine se javljaju?

# Ovo je mjesto na kojem možete izvoditi svoj kod.

```
valic = 'morl'
```

```
w = pywt.ContinuousWavelet(valic)
psi,t = w.wavefun(level=10)
```

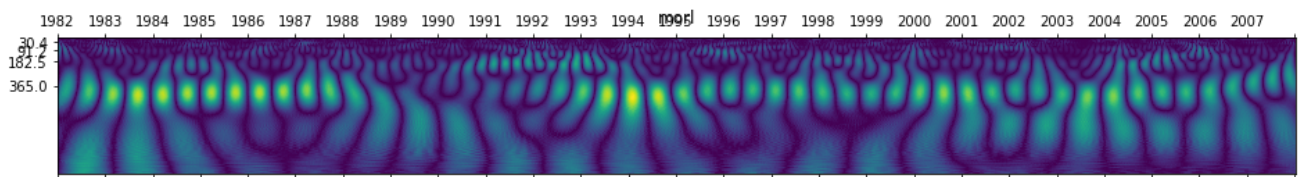


```
#kod s predavanja
step = 0.7 #korak
skala = np.arange(0.1, 2*T, step) #određuje kolko ce bit razmaknuto kao

#definiranje varijabli za valičnu transformaciju
coef,freqs = pywt.cwt(x,skala, valic)

#Nacrtajte apsolutnu vrijednost dobivenih koeficijenata.
plt.matshow(abs(coef))
pocetnaGod = 1982
plt.xticks(np.arange(0, kobas.size, 365), np.arange(pocetnaGod, int(pocetnaGod + ko
plt.yticks([T/12, T/4, T/2, T]) #gdje ces oznaciti na y osi
plt.title(valic)

plt.show()
```



ODGOVOR: Ovo je mjesto za odgovor na pitanja iz 9. zadatka.

- 1.) Koje frekvencije su vidljive u vodostaju? najvidljivije frekvencije na grafu možemo uočiti na broju 365, tj na 1 T (1 period od godinu dana), što nam govori da je ponavljanje najizraženije svako godinu dana. Vide se i ostala ponavljanja svako pola godine ili 4 mjeseca, ali puno slabije
- 2.) Koje godine se javljaju? Na ovoj slici puno jasnije možemo vidjeti sušne/vlažne godine, te sam tp gore i krivo procjenjivao ali neću mijenjati... veći vodostaji su bili u razdobljima od 1983. do 1988. te od 1993. do 1995. Razdoblje s najmanjim vodostajima (sušno razdoblje) je od 1989. do 1992.

Double-click (or enter) to edit

## 10. Odgovorite: objasnite razliku u dobivenim rezultatima STFT i CWT.

STFT je Fourierova transformacija na vremenskom otvoru gdje možemo sami birati veličinu okvira. Ovisno o izabranoj veličini, bolje možemo uočiti frekvencije / vremenski pregled. CWT je kontinuirana valična transformacija.

STFT koristi sin i cos za prikaz signala ("igra se" s frekvencijama i amplitudama), dok CWT koristi valiće po kojima je dobila ime (neke unaprijed određene funkcije s točno određenim brojem minimuma i maksimuma). Tu možemo sami prilagođavati izgled valića pomoću skale (sužavanje / širenje)

U ovom primjeru možemo kroz odgovore na pitanja vidjeti da je CWT bolje opisao traženi problem... na STFT se manje jasno vide frekvencije i vremena koja tražimo..na CWT se odlično vide ponavljanja za  $k=1$ , tj. ponavljanja svake godine.

ODGOVOR: *Ovo je mjesto za odgovor na pitanje iz 10. zadatka.*

Double-click (or enter) to edit

---

✓ 13s completed at 10:02 PM

