

# 12. Spektralna analiza in filtriranje

Domača naloga pri predmetu Modelska analiza I

Avtor: Matic Noč

10.1.2017

## 1 Uvod

Ponovno bomo obravnavali populacijske modele, le v nekoliko pogledu. V prejšnji nalogi smo preprosto reševali diferencialne enačbe in tako uvideli, da rešitve res opisujejo npr. periodičnost populacije ali pa razmah/izumrtje. Tokrat pa bomo problem prevedli na diskretno obliko in tako s žrebanjem poiskali kaj se dogaja s populacijami.

## 2 Frekvenčni spekter

Pri prvi nalogi smo signaloma s 512 točkami na datotekah *val2.dat* in *val3.dat* določili frekvenčni spekter. Ker ni bilo podano koliko časa smo vzorčili, smo vzeli arbitrani čas 1s, kar pomeni da je čas med vzorčenjem  $dt = \frac{1s}{512}$  in frekvenčna skala pri FT sega le do Nyquistove frekvence  $\mu = \frac{1}{2dt} = 256Hz$ , seveda pa ostali 256 točk sega tudi simetrično za negativne frekvence. Preizkušali smo različne okenske funkcije, ki nam skušajo odstraniti neveznost na robovih in opazovali sprememjanje spektra, če analiziramo krajše intervale, npr. 64 ali 128.

Diskretno Fourierovo transformacijo lahko definiramo z enačbo

$$F_k = \sum_{n=0}^{N-1} f_n \cdot e^{-i2\pi kn/N}, \quad (1)$$

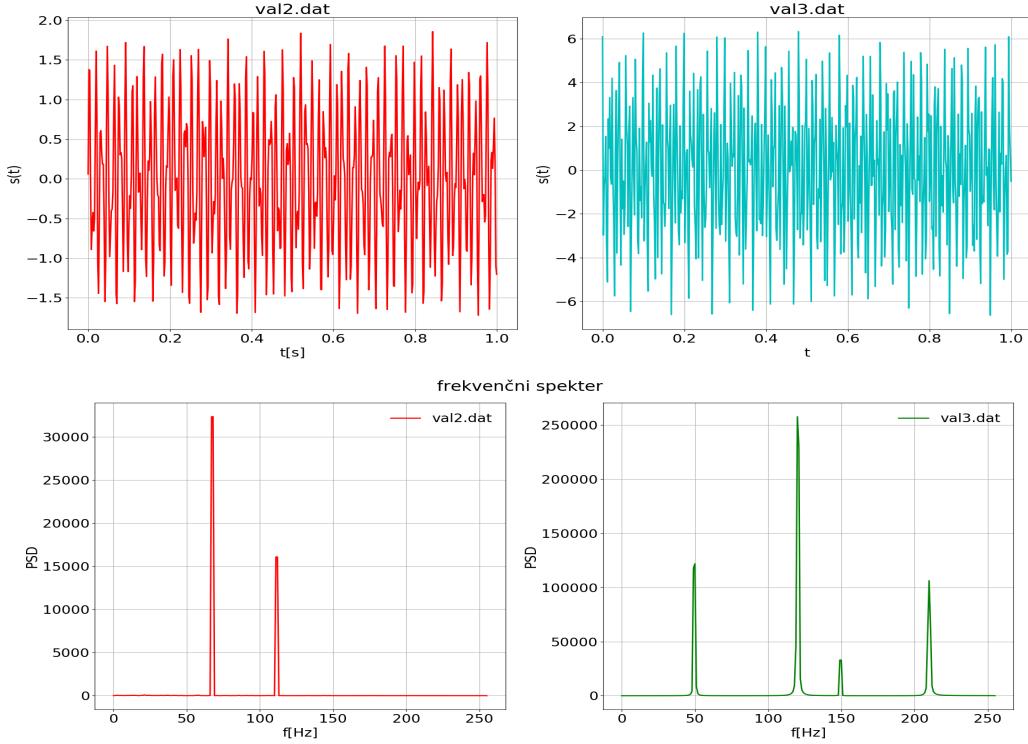
kjer so  $f_n$  vhodni podatki,  $F_k$  pa Fourierova transformiranka, ustrezno pa lahko definiramo inverz kot

$$f_n = \frac{1}{N} \sum_{k=0}^{N-1} F_k \cdot e^{i2\pi kn/N}. \quad (2)$$

Slednja transofrmacija zahteva zanko n za vsako točko k, kar pomeni, da imamo  $\sim n^2$  operacij, kar pri 512 točkah ni problematično vendar pa pri frekvenci vzorčenja 44 kHz postane počasno, na srečo pa so s pomočjo dinamičnega programiranja ustvarili FFT, ki izkoristi vse optimizacijske možnosti in deluje v logaritemskem času.

Najprej sem oba signala narisal s 512-imi točkami v času  $t = [0, 1]$ , kot je prikazano na levem grafu nato pa s Fourierovo transformacijo (FFT) dodatka *scipy.fftpack* izračunal frekvenčni spekter, iz katerega sem izračunal spektralno moč, ki je enaka

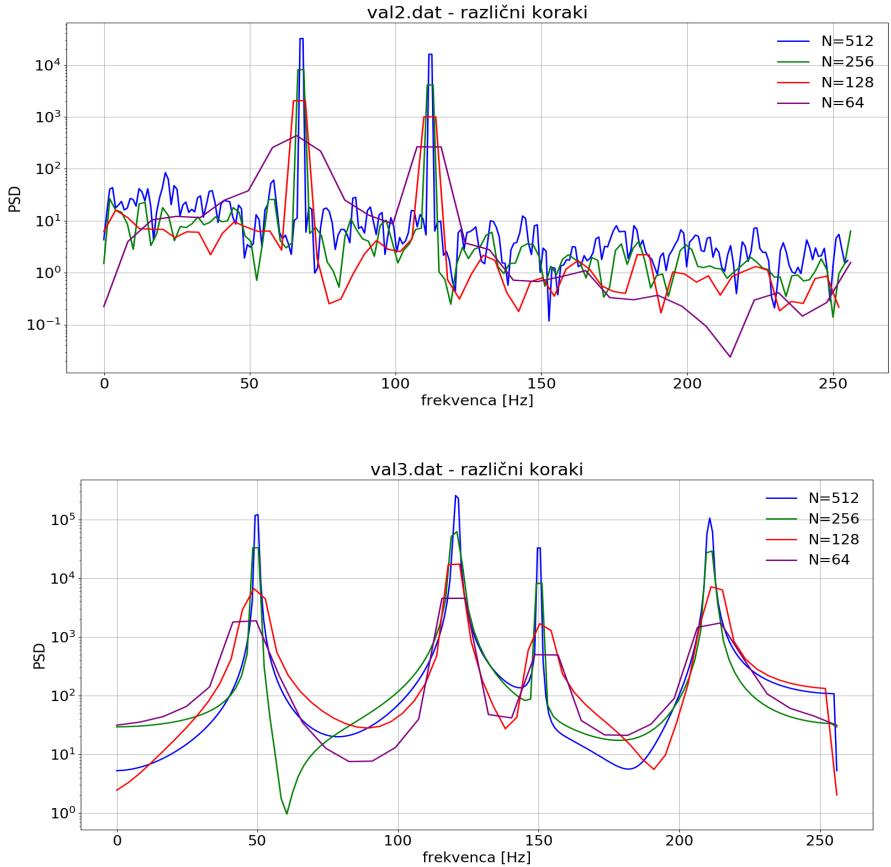
$$\begin{aligned} PSD &= F_k F_k, k \in robovi \\ PSD &= \frac{1}{2}(F_i F_i + F_j F_j), i, j - soseda, k \notin robovi \end{aligned} \quad (3)$$



Slika 1: Prikaz signalov (zgoraj) in spektralne gosotote (spodaj)

## 2.1 Vpliv izbora točk na spekter

Poglejmo si kaj se zgodi, če vzamoe le prvih 64,128,256 točk spektra in naredimo FFT. Seveda se frekvenca samplanja ne spremeni, tako da naš spekter še vedno kaže do mejne Nyquistove frekvence. Ker bomo vzeli manjši del signala, bi morali še vedno najti vrhove, ki bodo izstopali iz spektra, vendar pa bodo zaradi tega ker je skala večja od njihove dejanske skale, raztegnjeni in nižji ker smo prejeli manj informacije, kar si lahko ogledamo na spodnjih grafih, kjer smo risali  $\log(PSD)$ , da bomo lažje opazili razlike



Slika 2: Preiskus manjšega dela signala, nam da pričakovane rezultate. Spektralna gostota je širša in vrhovi so manjši. Zanimivo pa se mi zdi, da tudi pri 64 točkah lahko dobimo informacijo o vrhu, čeprav pa je ta sporna, saj pri spektralni analizi želimo čim večjo ločljivost frekvenc!

## 2.2 Okenske funkcije

V nadaljevanju sem preizkusil različne okenske funkcije, s katerimi namerno deformiramo signal, da popravimo neperiodičnost našega signala. Značilnosti oken so, da so majhna na robu in čim večja na sredini, v bližini centralne transformiranke. Poznamo več vrst oken npr.

*Welchevo okno*, definirano kot

$$w(n) = 1 - \left( \frac{n - \frac{N-1}{2}}{\frac{N-1}{2}} \right)^2 \quad (4)$$

*Hannovo okno*

$$w(n) = 0.5 \left[ 1 - \cos \left( \frac{2\pi n}{N-1} \right) \right] = \sin^2 \left( \frac{\pi n}{N-1} \right), \quad (5)$$

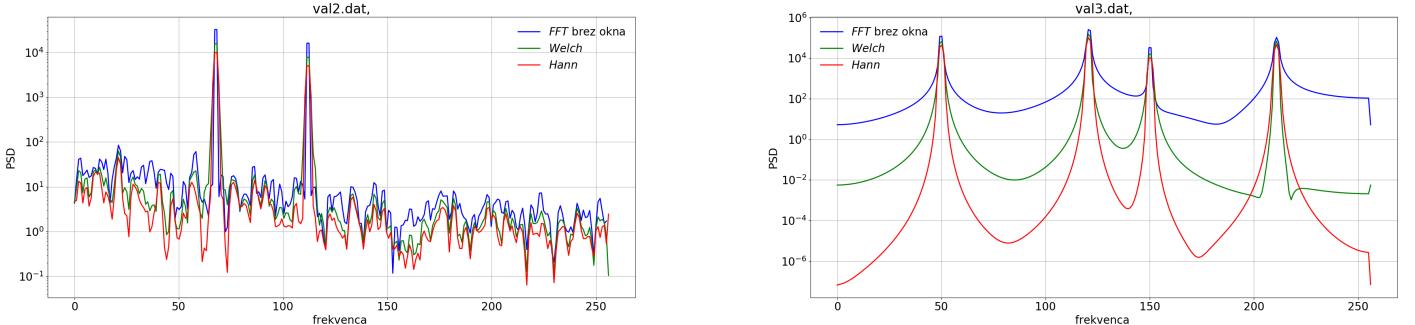
*Bartlettovo okno*

$$w(n) = 1 - \left| \frac{n - \frac{N-1}{2}}{\frac{L}{2}} \right|, \quad (6)$$

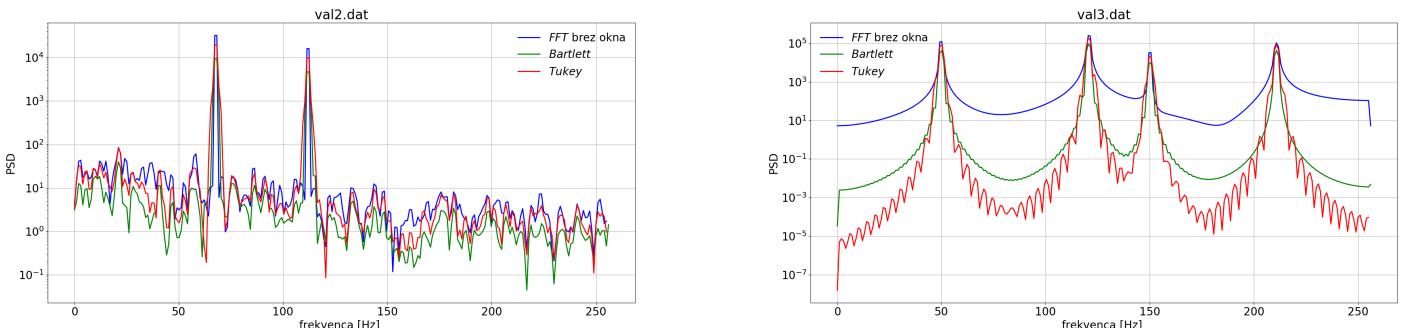
kjer gre trikotno okno, za katerega vzamemo  $L = N - 1$ . Poznamo še okno *Tukey*, definirano kot

$$w(n) = \begin{cases} \frac{1}{2} \left[ 1 + \cos \left( \pi \left( \frac{2n}{\alpha(N-1)} - 1 \right) \right) \right] & 0 \leq n < \frac{\alpha(N-1)}{2} \\ 1 & \frac{\alpha(N-1)}{2} \leq n \leq (N-1)(1 - \frac{\alpha}{2}) \\ \frac{1}{2} \left[ 1 + \cos \left( \pi \left( \frac{2n}{\alpha(N-1)} - \frac{2}{\alpha} + 1 \right) \right) \right] & (N-1)(1 - \frac{\alpha}{2}) < n \leq (N-1) \end{cases} \quad (7)$$

Metode je precej preprosta, saj potrebujemo  $F_k = FFT(w_n s_n)$



Slika 3: Prikaz okenske funkcije Welch in Hahn. Vidimo da se šum zmanjša za celo dekado, kar pomeni, da so naši vrhovi veliko bolj ostri in je šuma manj.



Slika 4: Prikaz okenske funkcije Bartlett, Turkey. Vidimo, da vsa okna popravijo naš signal in izostrijo vrhove ter zmanjšajo šum. To je seveda odvisno od našega signala, vendar pa vidimo, da na signalu *val3.dat* na desni šum zmanjšamo kar za 4 dekade oziroma 10 000 krat. Turkey zanimivo zaoscilira našo moč, zato se mi zdi manj priročen. Najboljše okno za naš primer je tisto ki najbolj zmanjša šum, torej *Hann*, sinusno okno in *Bartlett* - trikotno okno, seveda pa je to odvisno od primera do primera in je verjetno treba poskusiti in opazovati katere vrhove imamo in kateri so šum.

### 3 Dekonvolucija signalov

Recimo, da želimo meriti signal  $u(t)$ . Vedno je ta signal vključen v vezje, ki ga lahko zapišemo s prenosno funkcijo  $r(t)$ . Izhodni signal pa je konovljenja  $r(t)$  in  $u(t)$  ter dodatni šum zaradi vezja

$$c(t) = u(t) * r(t) + n(t) = s(t) + n(t) \quad (8)$$

Izmerimo  $c(t)$ . Zanima nas kako dobiti prvotni signal  $u(t)$ . Uporabimo fourierovo transformacijo, za katero velja, da konvolucija postane množenje ustreznih transformirank

$$C(f) = U(f)R(f) + N(f) \quad (9)$$

Če je  $N(f) = 0$  lahko  $u(t)$  izračunamo kot

$$u(t) = FFT^{-1}\left[\frac{C(f)}{R(f)}\right] \quad (10)$$

Če pa imamo šum, potem pa je ta metoda odpade in moramo najti drugačen način za približek  $u(t)$ . To naredimo s pomočjo minimizacije

$$\int |\widetilde{u(t)} - u(t)|^2 = \int |\widetilde{U(f)} - U(f)|^2 \quad (11)$$

kjer je

$$\widetilde{U(f)} = \frac{C(f)}{R(f)}\Phi(f) \quad (12)$$

razpišemo in odvajamo po  $\Phi$  in dobimo

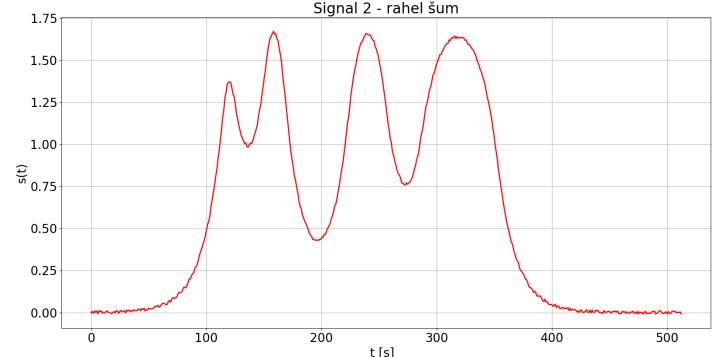
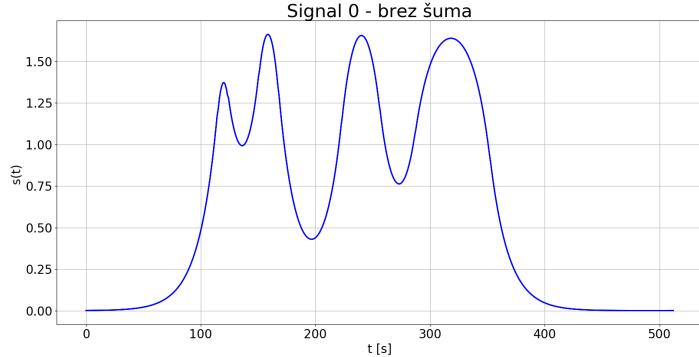
$$\Phi(f) = \frac{|S(f)|^2}{|S(f)|^2 + |N(f)|^2}. \quad (13)$$

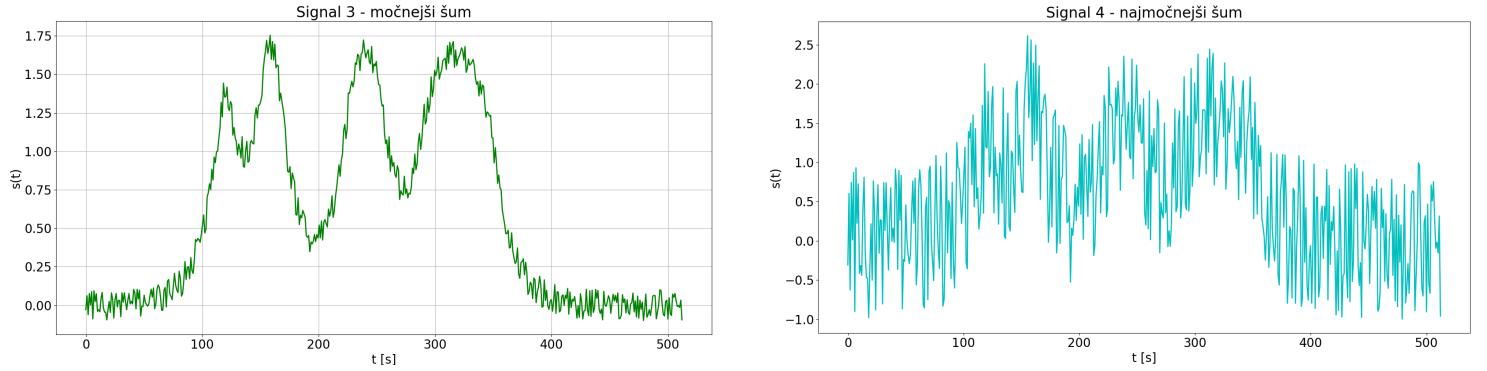
Šuma  $n(t)$  in posledično  $N(f)$  ne poznamo eksplisitno, zato ga moramo oceniti. Tako dobimo našo oceno za signal kot

$$F^{-1}(\widetilde{U(f)}) = F^{-1}\left(\frac{C(f)}{R(f)}\Phi(f)\right) \quad (14)$$

### 3.1 Signali

Na voljo imamo 4 signale, ki naraščajo po zašumljenosti.





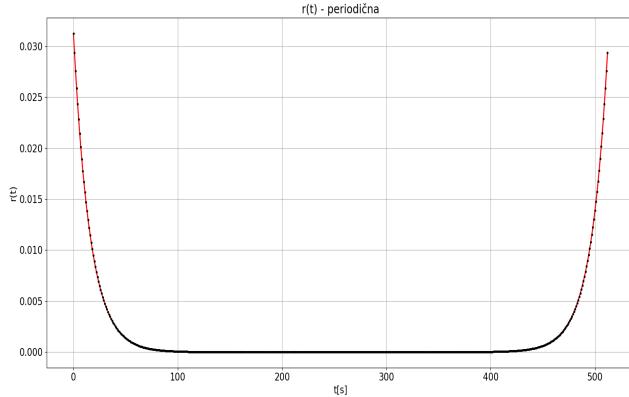
Slika 5: Prikaz Signalov

### 3.2 Dekonvolucija signalov

Izmerimo, da je prenosna funkcija vezja sestavljena iz dveh eksponentnih funkcij, pri čemer je druga zrcaljena čez ordinatno os

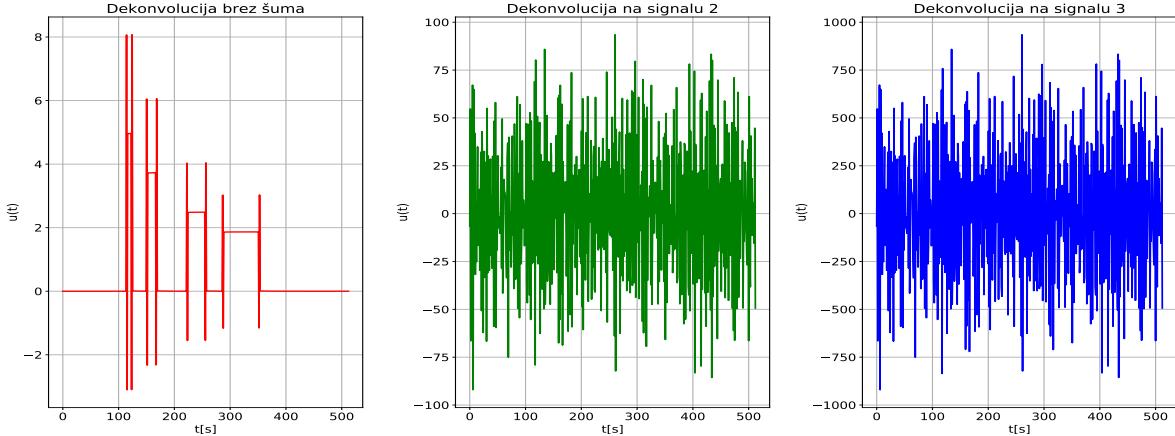
$$r(t) = \frac{1}{2\tau} e^{\frac{-|t|}{\tau}}. \quad (15)$$

Vzemimo da je  $\tau = 16$  in čas samplanja  $512s$  tako da je Nyquistova frekvenca  $\nu_{nyquist} = 0.5Hz$ .



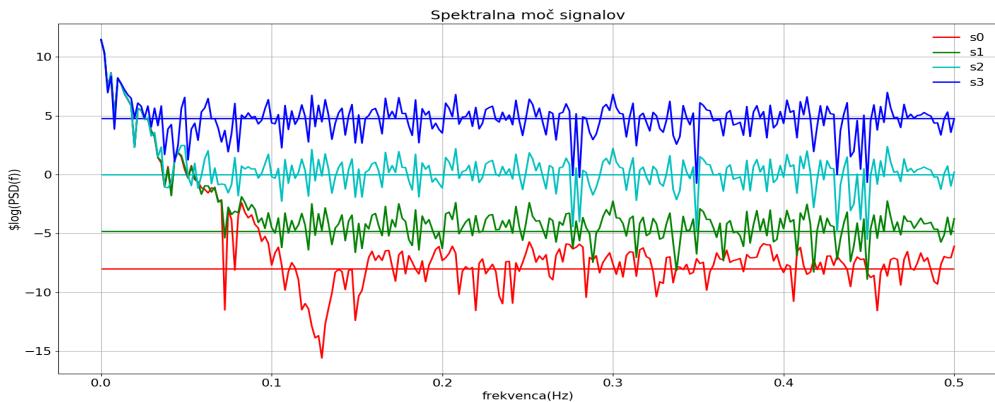
Slika 6: Pazimo, da je  $r(t)$  periodična, tako da zadnja točka ne ponovi prve vrednosti.

Na signalu brez šuma lahko napravimo dekonvolucijo brez Wienerjevega filtra



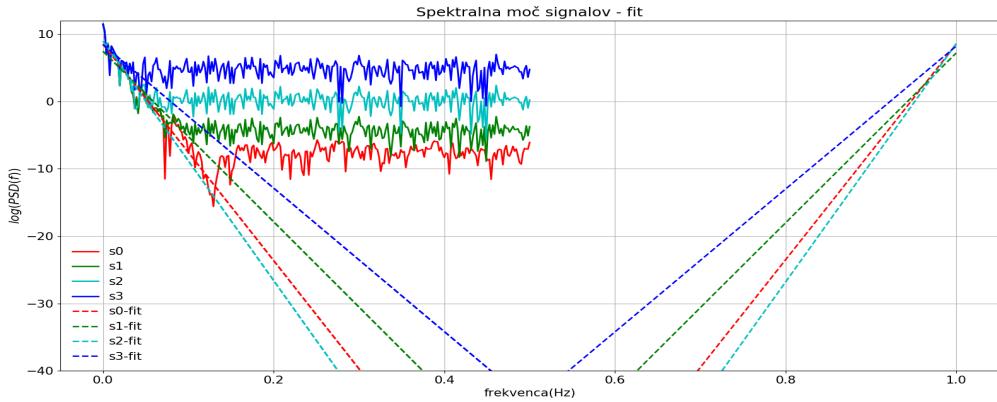
Slika 7: Poskus dekonvolucije brez Winerjevega filtra, vidimo da na signalih ki so zašumljeni ne deluje več.

Za izračun Winerjevega filtra  $\Phi$  moramo oceniti  $|N(f)|^2$  in  $|S(f)|^2$ .  $|N(f)|^2$  ocenimo tako da narišemo spektralno moč in pogledamo kakšno je moč šuma, torej tisti del spektra, kjer se vrhovi končajo in je v povprečju konstanten.



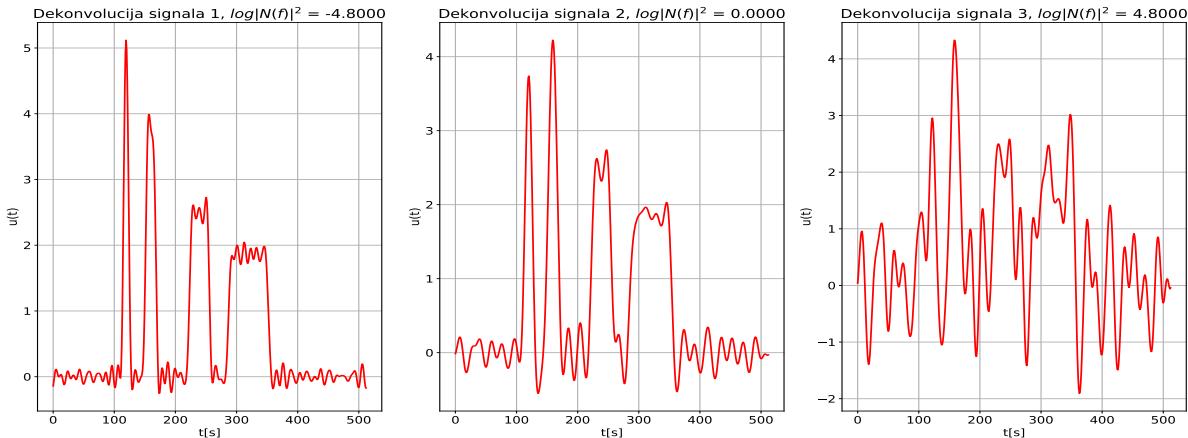
Slika 8: Spektralna moč signalov nam da oceno za velikost šum, ki pravilno narašča z zašumljenostjo signala. Ideja je v tem, da verjetno v našem sistemu nimamo konstatne moči tako širokega spektra frekvenc in ta mora ustrezati šumu.

Za  $|S(f)|^2$ , pa imamo problem, saj je to Fourierova transformiranka signala  $s(t)$  brez  $n(t)$ , ki pa ga mi poznamo le v začetnem delu, kjer vidimo da ima obliko premice, tako da nanj interpoliramo premico v logaritemskem prostoru oziroma  $S(f) = Ae^{-Bf}$  v normalnem prostoru. Paziti moramo tudi, da za fourierov obrat potrebujemo tudi negativni del spektra in tako tudi na negativni strani interpoliramo premico, ki pa jo za boljšo predstavo prezrcalimo čez ordinatno os.



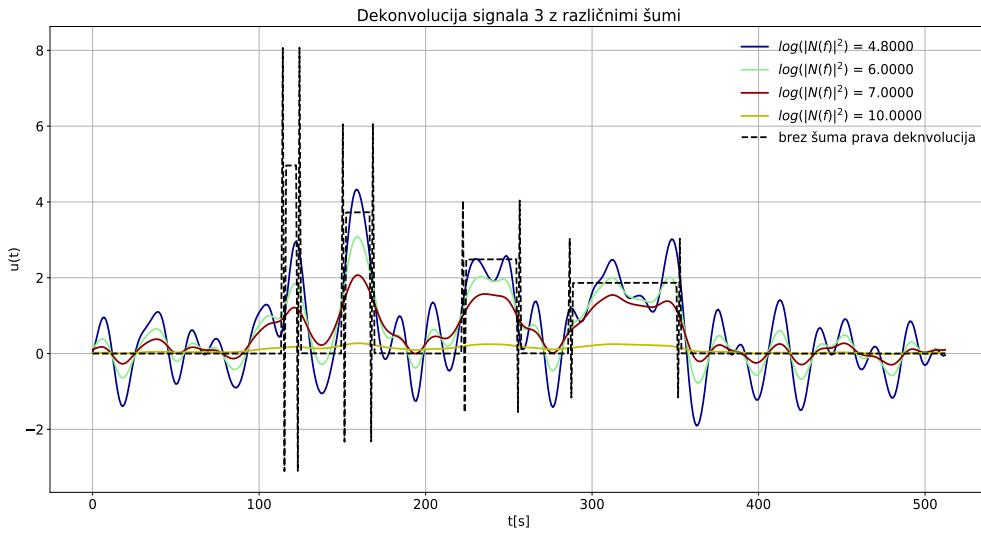
Slika 9: Do neke mejne frekvence  $\nu_{mejni}$ , ki jo ocenimo - ko je naš spekter enak le šumu; interpoliramo premico, ki se konča na polovici drugega dela pa prezrcalimo čez ordinatno os. Paziti moramo, da zadnja točka prezrcaljene premice ni enaka začetni, da ohranimo periodičnost. Vidimo tudi da je najbolj zašumljen signal najmanj strm.

Sedaj poznamo  $|S(f)|^2$  in  $|N(f)|^2$ , ki ju moramo še pretvoriti iz logaritemsko skale s pomočjo inverzne funkcije in nato lahko izračunamo  $\Phi(f)$ , in poskusimo s dekonvolucijo s pomočjo Wienerjevega filtra.



Slika 10: Dekonvolucija zašumljenih signalov - vidimo, da smo precej dobro ocenili vrednost signala in šuma, saj nam dekonvolucija da dobre rezultate, vendar pa vidimo, da smo za najbolj zašumljen signal ocenili premalo šuma.

Poskusimo sedaj s več različnimi vrednosti šuma na signalu 3, ki je najbolj zašumljen



Slika 11: Vpliv šuma na dekonvolucijski signal  $u(t)$ : Vidimo, da nam prevelik šum preveč zaduši vrednosti signala (siva), med tem ko za premajhen šum (modra, 4.8) v  $u(t)$  prejmemmo preveč šuma in tako signal ni razpoznaven, najboljša rešitev je tako svetlo modri šum, ki nam ojači le vrhove in zaduši stranski šum

### 3.3 Dekonvolucija slike

Recimo da z vrstičnim tunelskim mikroskopom zajemom sliko, ki jo predstavimo s  $N$  stolpcji in  $M$  vrsticami. To naredimo tako da odberemo vsak stolpec posebaj in si zapišemo intenzitete električnega toka/svetlobe. Ko merimo se signal žal razmaže zaradi dveh pojavov; šuma v merilnem sistemu in vpliva sosednjih izmerjenih mest na sosedje. Tak vpliv vpišemo s prenosno funkcijo, npr. eksponentno

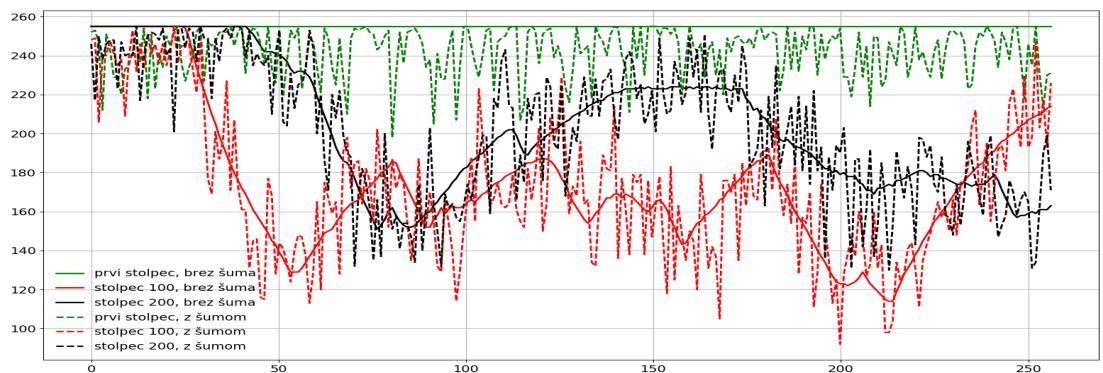
$$r(t) = \frac{1}{\tau} e^{-\frac{t}{\tau}} \quad (16)$$

Oglejmo si primer dobljenih slik na testnem primeru Lincoln.



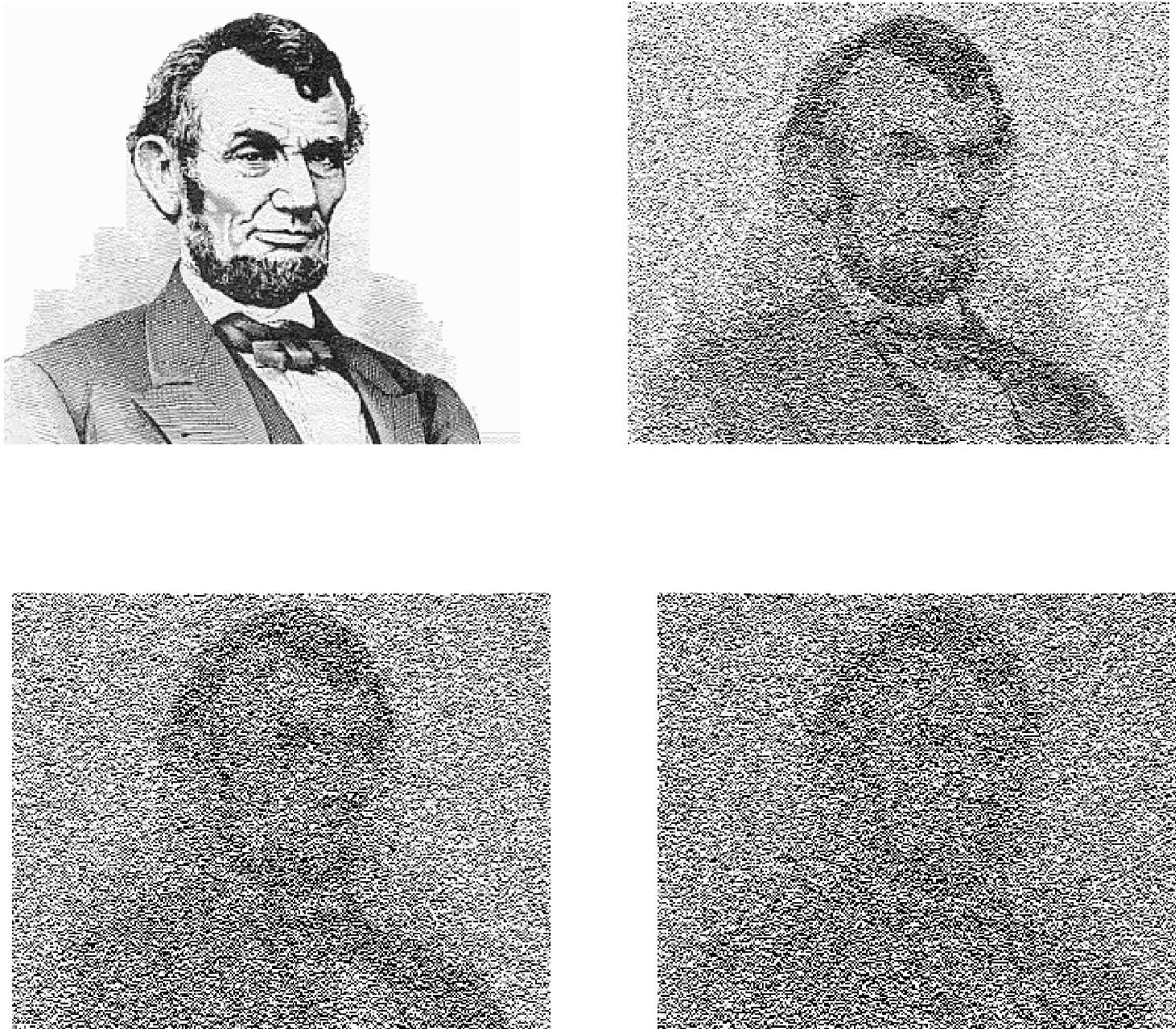
Slika 12: Lincolnove podobe od 1 do 5 so različno zašumljene. Prva slika je brez šuma, slike od 2-5 pa so vse zašumljene. Vse so malo izmaknjene v navpični smeri kar je vpliv konvolucije s signalom  $r(t)$ .

Vsak stolpec je tako 1D signal, ki ga moramo dekonvoluirati.



Slika 13: Primeri signalov. Vidimo da nam šum namesto ravne črte popači signal

S pomočjo enačbe (12) dekonvoluiramo nezašumljeno sliko in poskusimo na zašumljenih slikah



Slika 14: Dekonvolucijska slika Lincoln ni več popačena. Na zašumljenih slikah pa brez Wienerjevga filtra samo še povečamo šum, čeprav smo sliko načeloma dekonvoluirali.

### 3.4 Wienerjev filter

Oceniti moramo  $\Phi$ . Lahko bi to storili avtomatizirano za vsak stolpec posebaj a se raje odločimo za izračun spekralne gostote povprečnega signala po stolpcih. Iz tam lahko ocenimo šume za vsako sliko in poskusimo s linearnim fitom, ki pa nam ne opiše dobro signal zato raje za  $|S(f)|^2$  uporabimo kar  $|C(f)|^2$ , ki pa mu porežemo visoke frekvence, kjer je spekralna moč le še konstantna.



Slika 15: Linearni poskus se izjalovi.

Če preprosto odrežemo del signalov in ga uporabimo v Wienrjev filter dobimo že dosti boljše rezultate.



Slika 16: Prikaz slik po zašumljenosti od leve proti desni. Iz grafa spektralnih moči vidimo, da je cut off za bolj zašumljene slike večji. Čeprav so slike dekonvoluirane pa imajo naslednje probleme. Niso ostre, in ostajajo zašumljene. Poskusimo se znebiti šuma tako da na graf fitamo eksponent in nato odrežemo signal, stem pa spektralna moč postane bolj gladka.

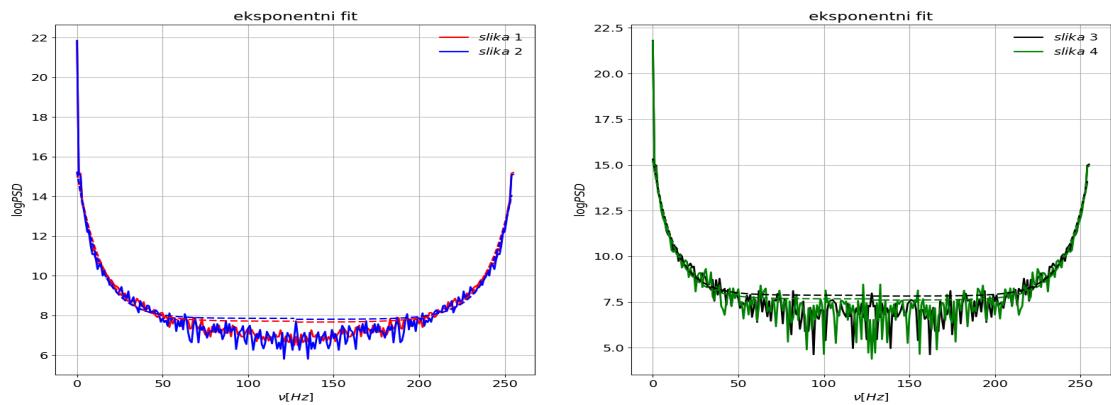
Čeprav so slike dekonvoluirane kljub šumu, pa imajo naslednje probleme. Niso ostre, in ostajajo zašumljene. Poskusimo se znebiti šuma tako da na graf fitamo eksponent in nato odrežemo signal, stem pa spektralna moč postane bolj gladka. Najprej si poglejmo vpliv velikosti cut off frekvence na sliki 4.

Poskusimo s eksponetnim fitom, da bi zmajšali šum

### Poskus eksponentnega fita



Slika 17: Eksponentni fit s istimi parametri kot na sliki 16. Vidimo, da se tako znebimo zašumljenosti v primerjavi s sliko 16.



Slika 18: Eksponentni fit.

### 3.5 Vpliv cut off na sliko



Slika 19: Cut off 60,45,35 Hz na sliki 2 (rahlo zašumljena ) nam poreže vedno višje frekvence, zato je slika manj ostra od leve proti desni,

Za izboljšanje slike bi morali avtomatizirati iskanje Winerjevega filtra na vsakem signalu in nato dekonvoluirati vsak stolpec s ustreznim  $\Phi$ . S slikami ki smo jih dobili iz povprečenja smo vseeno zadovoljni saj pri dovolj močnem šumu težko zares očistimo sliko.

## 4 Zaključek

Naučili smo se kako dekonvoluirati signale in vpliv okenskih funkcij za izboljšavo fourierove transformacije. Pogledali smo si tudi praktičen primer dekonvolucije in filtriranja slik.