

# Mafija praktikum: Hitra Fourierova transformacija in korelacijske funkcije

Andrej Kolar-Požun

November 8, 2016

## 1 Uvod

Pri naslednjih nalogah bom uporabljal hitro Fourierovo transformacijo(FFT), ki je v Pythonu že implementirana v scipy knjižnici. Ta funkcija ima tudi že vgrajen "zero-padding", tako da doda potrebno število ničel, da bodo signali, katere bom obdeloval iste dolžine, ter da bo ničel dovolj, da ne bo prišlo do odtujitve. Oknenja tokrat ne bom uporabil, ker bomo pri korelaciji itak naredili še inverzno transformacijo. Zanimala nas bo predvsem korelacijska funkcija, ki nekako pove, kako podobna sta si dva signala. Računal jo bom na ta način:

$$C(g, h) = \mathcal{F}^{-1}[\mathcal{F}g \cdot (\mathcal{F}h)^*]$$

Kjer  $*$  predstavlja kompleksno konjugacijo in  $\mathcal{F}$  predstavlja Fourierovo transformacijo. Zanimala nas bo tudi avtokorelacijska funkcija oziroma korelacijska funkcija same s sabo, ki je seveda definirana kot:

$$C(h, h) = \mathcal{F}^{-1}[|\mathcal{F}h|^2]$$

Da se bo lepše videlo, bom rajši računal reskalirano avtokorelacijsko funkcijo:

$$\tilde{C}(h, h)_n = \frac{C(h, h)_n - \langle h \rangle^2}{C(h, h)_0 - \langle h \rangle^2}$$
$$\langle h \rangle = \frac{1}{N} \sum_{k=0}^{N-1} h_k$$

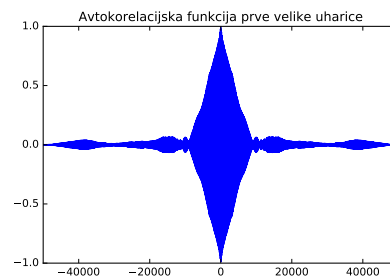
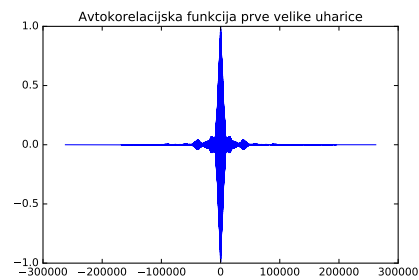
Tudi pri korelaciji dveh signalov bom uvedel neko vrsto reskalacije:

$$\tilde{C}(g, h) = \frac{C(g, h)}{|f||g|}$$
$$|f|^2 = |f_0|^2 + |f_1|^2 + \dots + |f_{n-1}|^2$$

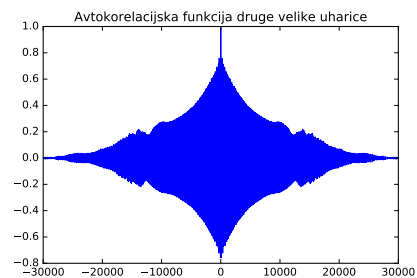
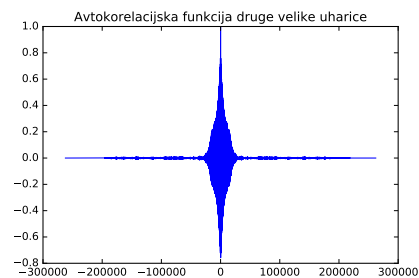
## 2 Oglašanje velike uharice

Podane imamo posnetke oglašanja dveh sov velikih uharic, in posnetke istih sov skupaj z motečim ozadjem. S korelacijo bom poskusil ugotoviti za katero sovo gre v slednjih posnetkih. Najprej si oglejmo avtokorelacijsko funkcijo obeh sov z in brez ozadja.

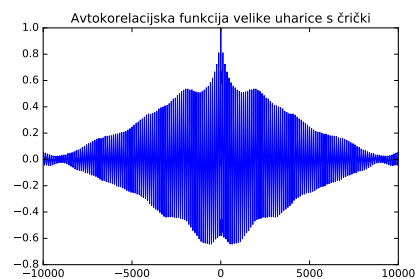
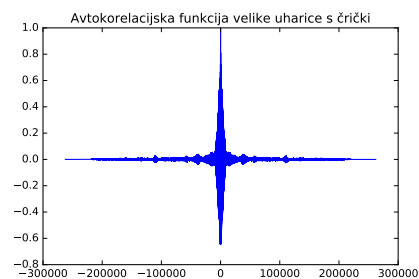
### 2.1 Avtokorelacijske funkcije

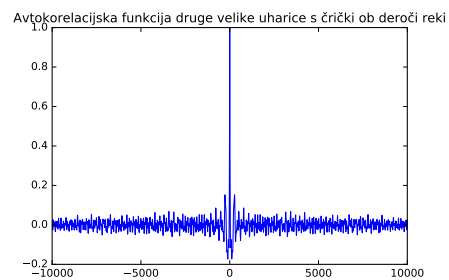
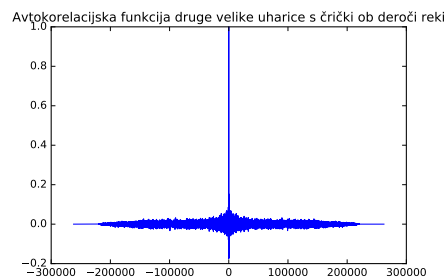
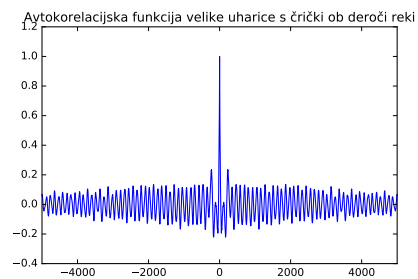
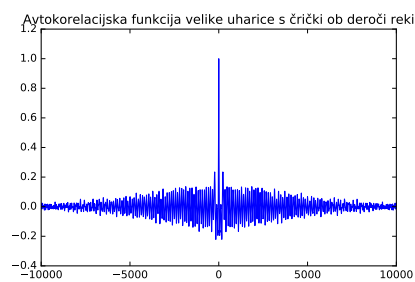
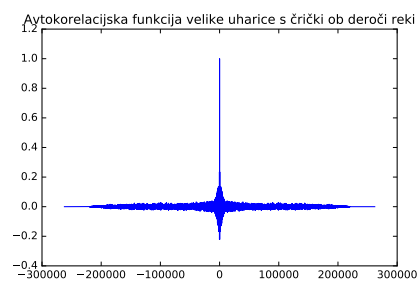
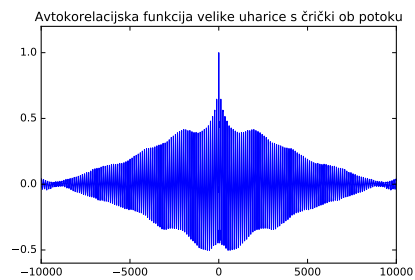
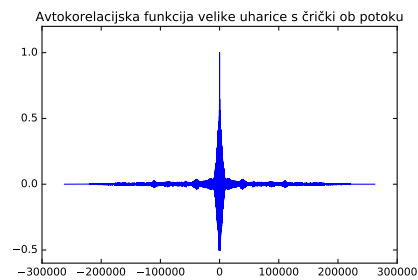


Seveda je največja korelacija pri  $x=0$  (Saj takrat primerjamo signal točno samim s sabo, vidi se tudi iz definicije reskalirane avtokorelacijske funkcije) in potem pada. Opazimo še manjše lokalne maksimume/minimуме



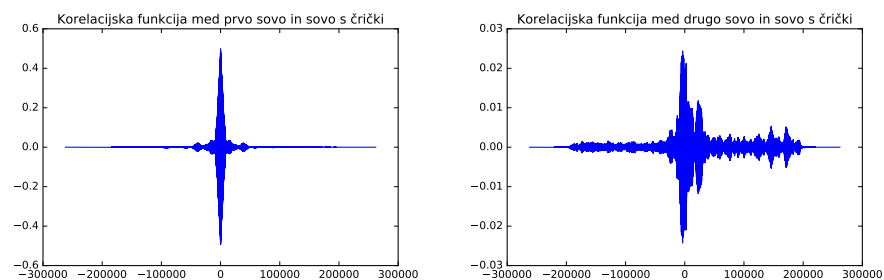
V tem primeru se očitno avtokorelacija izgublja počasneje kot prej



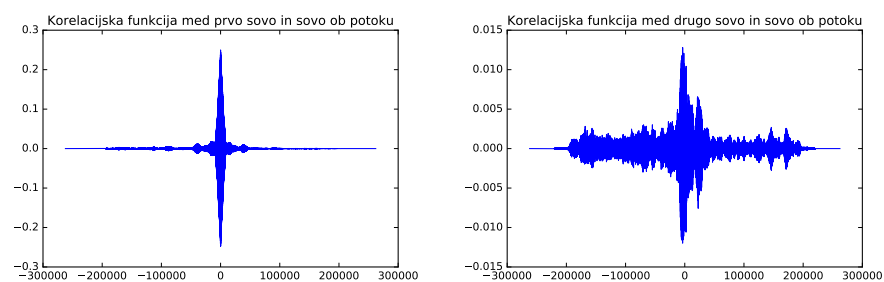


Za uharice ob deroči reki opazimo, da se avtokorelacija izgublja zelo hitro

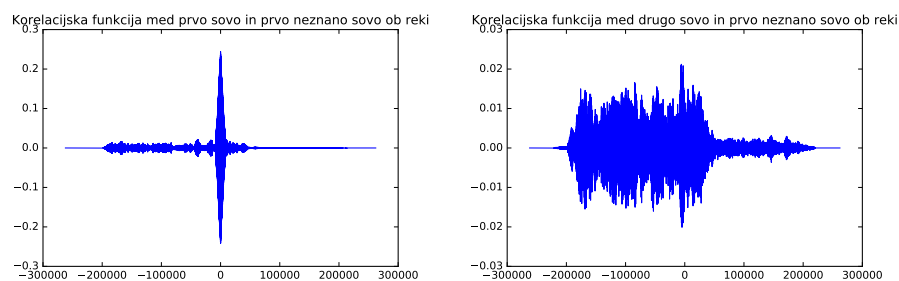
## 2.2 Korelacijske funkcije



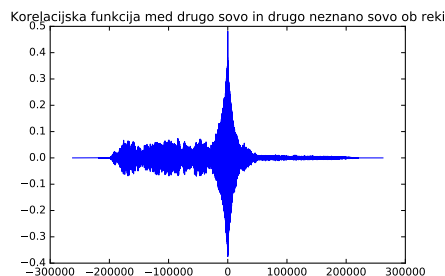
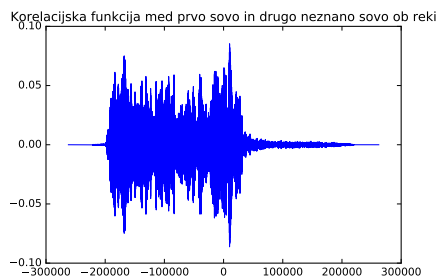
Sova ob črčkih je torej prva sova



Sova ob potoku je tudi prva sova



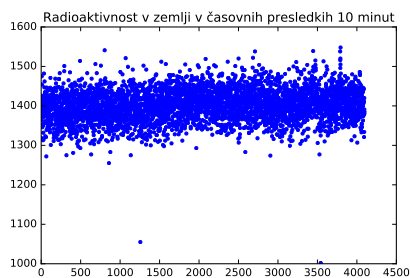
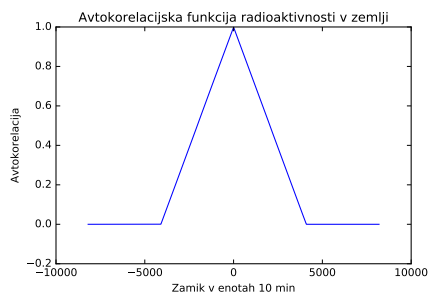
Prva neznana sova ob deroči reki je spet naša prva sova



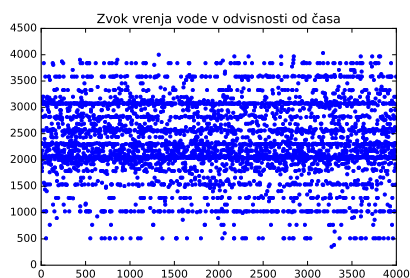
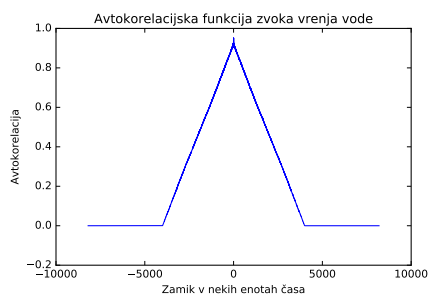
Druga neznana sova ob deroči reki pa je naša druga velika uharica

### 3 Ostale avtokorelacijske funkcije

Poglejmo si še nekaj zanimivih primerov. Pokazal bom nekaj podatkov, ki sem jih našel na spletni strani in avtokorelacijske funkcije le-teh.<sup>1</sup>

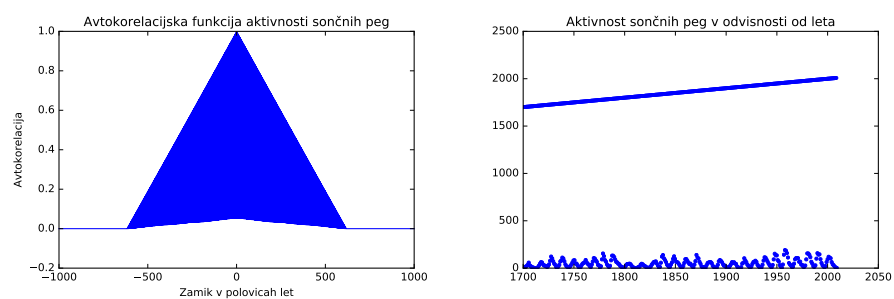


Vidimo, da se korelacija samo izgublja. Vrednosti grejo le od 0 do 1, ker so tudi moji vhodni podatki imeli le pozitivne vrednosti



Podobna slika kot prej. Tudi na desni nič kaj ne izgleda, da bi bilo kaj avtokoreliranosti.

<sup>1</sup><http://predmeti.fmf.uni-lj.si/mafiprak/naloga5>



Tukaj na desni vidimo, da neka periodičnost je prisotna. To se v avtokorelacijski funkciji vidi tako, da avtokoreliranost skače med približno 0 pa neko vrednostjo vidno nad 0