LATVIJAS UNIVERSITĀTE
FIZIKAS UN MATEMĀTIKAS FAKULTĀTE
MATEMĀTIKAS NODAĻA

KOLMOGOROVA-SMIRNOVA TESTS DATIEM AR ILGLAICĪGU ATMIŅU

KURSA DARBS

Autors: **Dmitrijs Bosņaks**
Stud. apl. db08132
Darba vadītājs: doc. Dr.math. Jānis Valeinis

RĪGA 2012. g. 27. janvārī

Saturs

Ievads

Termins ilglaicīga atmiņa kļuva aktuāls ap 1960.gadu. Kopš tā laika tas ieņem būtisko lomu laikrindu analīzē. Mūsdienās daudzus procesus var aprakstīt tieši ar šiem modeļiem. Pielietojumi ilglaicīgas atmiņas procesiem atrodami makroekonomikā, hidroloģijā, fizikā, astronomijā, telekomunikācijās, agronomijā un finanšu jomā. Šī darba mērķis ir pielietot testus teorētiskās sadalījuma funkcijas pārbaudei datiem ar ilglaicīgu atmiņu. Kaut no praktiskā redzes viedokļa mūs interesē saliktā hipotēžu pārbaude, tomēr sākumam šajā darbā mēs apskatīsim vienkāršo hipotēžu pārbaudi par datu sadalījuma funkciju. Tā ir relatīvi jauna problemātika, par kuru izdevās atrast tikai vienu publikāciju. Konkrēti mēs apskatīsim Kolmogorova-Smirnova testa vienas izlases gadījuma pielietošanu no publikācijas [1].

Pirmajā nodaļā mēs definēsim kas ir ilglaicīgas atmiņas procesi. Otrajā nodaļā mēs definēsim ARFIMA klases procesu, kuru mēs apskatīsim kā ilglaicīga procesa piemēru šajā darbā. Trešajā nodaļā mēs aprakstīsim testa statistiku un tas sadalījumu pie H_0. Ceturtajā nodaļā mēs veiksim jaudas analīzi simulētajam alternatīvam.

1. Ilglaicīgas atmiņas procesi

Šajā nodaļā mēs definēsim kas ir process ar ilglaicīgu atmiņu. Definīcijas paņemtas no [2].

Definīcija 1. Procesu sauksim par stacionāru vājā nozīmē, ja tas apmierina nosacījumus:

$$1)\ \forall\ t\ Ex_t = const$$

$$2)\ cov(x_t, x_{t+\tau}) \text{ ir atkarīga tikai no } \tau.$$

Turpmāk, vāji stacionārus procesus mēs sauksim vienkārši par stacionāriem.

Definīcija 2. Par stacionāra procesa autokovariāciju funkciju sauc

$$\gamma(h) = cov(x_t, x_{t+\tau}) = E(x_t - \mu)(x_{t+\tau} - \mu).$$

Definīcija 3. Pieņem, ka $\gamma(h)$ ir stacionāra procesa autokovariāciju funkcija, tad saka, kā procesam piemīt ilglaicīga atmiņa, ja

$$\sum_{h=-\infty}^{\infty} |\gamma(h)| = \infty.$$

Ilglaicīgai atmiņai ir arī alternatīvas definīcijas un viena no tām ir saistīta ar autokovariāciju hiperbolisko dilšanu.

Definīcija 4. Pozitīvu mērojamu funkciju l, definētu kādā bezgalības apkārtnē $[a, \infty)$ sauc par lēni variējošu Karamata nozīmē tad un tikai tad, ja katram $c > 0$ izpildās

$$\lim_{x \to \infty} \frac{l(cx)}{l(x)} = 1.$$

Piemēram funkcijas $l(x) = log(x)$ un $l(x) = c$, kur c ir pozitīva konstante ir lēni variējošas funkcijas.

Definīcija 5. Saka, kā procesam piemīt ilglaicīga atmiņa, ja

$$\gamma(h) \sim h^{2d-1} l(h)$$

kad $h \to \infty$, kur d ir tā saucamais ilglaicīgas atmiņas parametrs un l ir lēni variējošā funkcija.

2. ARFIMA procesi

Plaši pazīstama ilglaicīgas atmiņas modeļu klase ir autoregresīvie frakcionāli integrētie slīdošā vidējā ($ARFIMA$) procesi, kurus mēs arī izmantosim ilglaicīgas atmiņas simulācijām šajā darbā. Pēc būtības tie ir $ARIMA(p, d, q)$ procesu vispārinājums kur parametrs d var būt daļskaitlis.

Definīcija 6. $\{y_t\}$ ir $ARFIMA(p, d, q)$ process, ja

$$\phi(B)y_t = \theta(B)(1 - B)^{-d}\varepsilon_t,$$

kur $\phi(B) = 1+\phi_1 B+\cdots+\phi_p B^p$ un $\theta(B) = 1+\theta_1 B+\cdots+\theta_q B^q$ ir attiecīgi autoregresīvais un slīdošā vidēja operatori, $(1-B)^{-d}$ ir frakcionālais diferenču operators definēts ar binomiālu izvirzījumu

$$(1 - B)^{-d} = \sum_{j=0}^{\infty} \eta_j B^j = \eta(B),$$

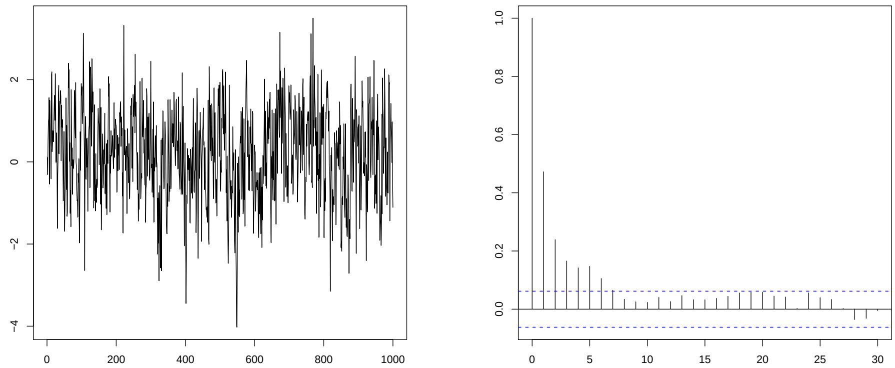
kur

$$\eta_j = \frac{\Gamma(j + d)}{\Gamma(j + 1)\Gamma(d)},$$

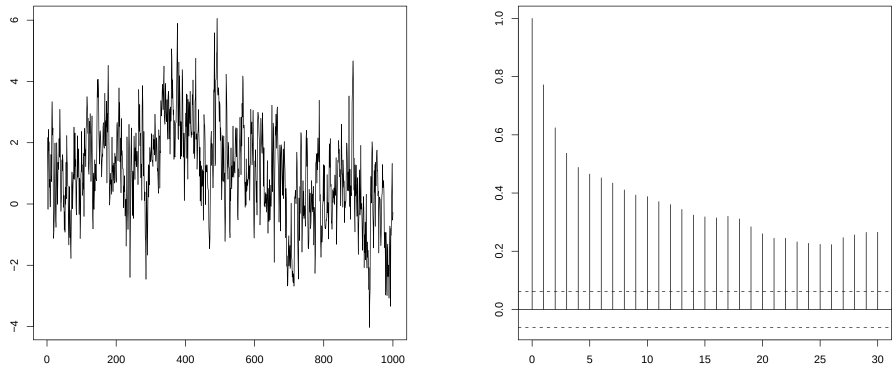
priekš $d < \frac{1}{2}$, $d \notin Z$ un $\{\varepsilon_t\}$ ir baltais troksnis. $\Gamma(x)$ ir gamma funkcija, kuru definē

$$\Gamma(x) = \int_0^{\infty} e^{-t} t^{x-1}\, dt$$

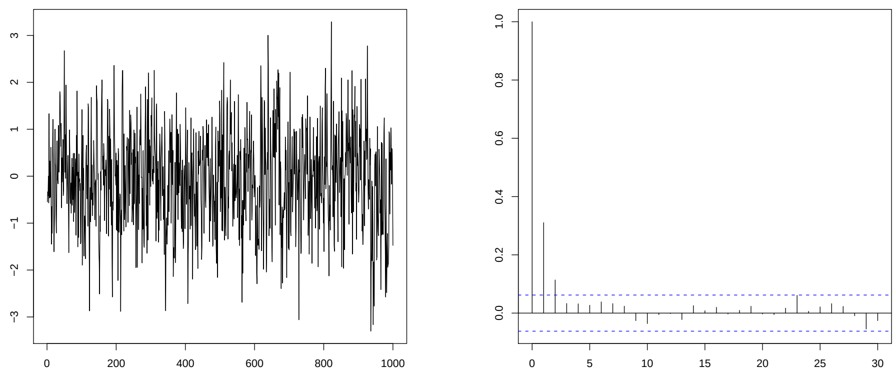
Tagad ilustrēsim dažus piemērus ARFIMA procesiem un tā ACF salīdzinājumā ar ARIMA procesu. Laikrindas garumā 1000 simulētās ar datorprogrammas R palīdzību.



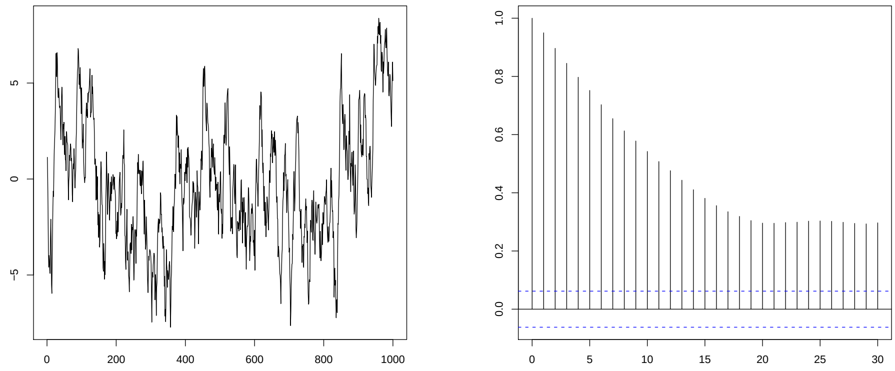
2.1. att. $ARFIMA(1, d, 0)$ process un tā ACF pie $\phi = 0.3$ $d = 0.1$.



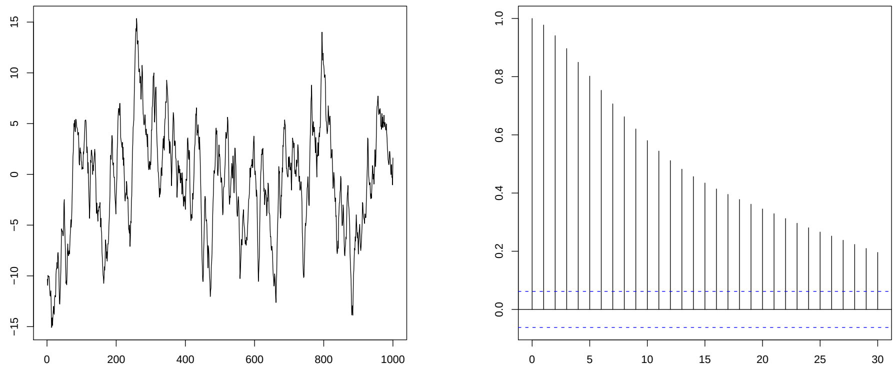
2.2. att. $ARFIMA(1, d, 0)$ process un tā ACF pie $d = 0.4$ un $\phi = 0.3$.



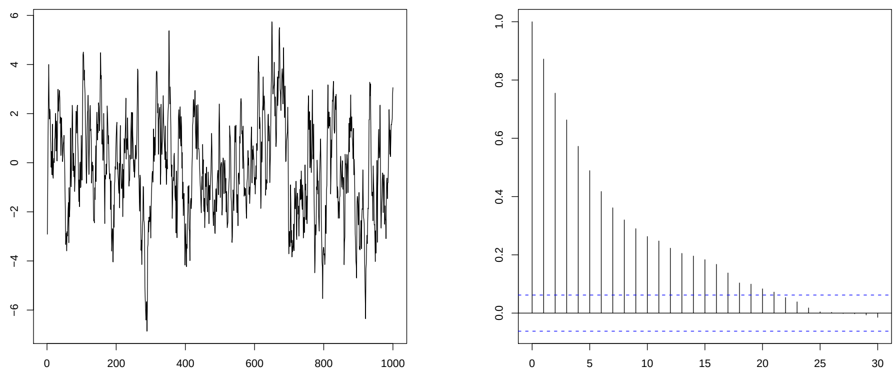
2.3. att. $ARIMA(1, 0, 0)$ process un tā ACF pie $\phi = 0.3$.



2.4. att. $ARFIMA(1, d, 0)$ process un tā ACF pie $\phi = 0.9$ un $d = 0.1$.



2.5. att. $ARFIMA(1, d, 0)$ process un tā ACF pie $\phi = 0.9$ un $d = 0.4$.



2.6. att. $ARIMA(1, 0, 0)$ process un tā ACF pie $\phi = 0.9$.

2.1. ARFIMA procesa $MA(\infty)$ un $AR(\infty)$ reprezentācijas

Definīcija 7. Stacionāru procesu $\{X_t\}$ sauc par apgriežamo, ja

$$X_t - \mu = \sum_{j=1}^{\infty} \pi_j (X_{t-j} - \mu) + \varepsilon_t,$$

kur $\sum_{j=1}^{\infty} \pi_j < \infty$ un $\mu = EX_t$.

Teorēma 1. *[2] Pieņem, ka ARFIMA procesam polinomiem ϕ un θ nav kopīgo sakņu un $d \in (-1, \frac{1}{2})$, tad*

a) Ja ϕ saknes ir ārpus vienības riņķa $\{z : |z| = 1\}$, tad eksistē viens vienīgs stacionārs atrisinājums

$$y_t = \sum_{j=-\infty}^{\infty} \psi_j \varepsilon_{t-j}$$

b) Ja θ saknes ir ārpus slēgta vienības riņķa $\{z : |z| \leq 1\}$, tad atrisinājums $\{y_t\}$ ir apgriežams.

Pēc teorēmas, pie pieņemuma, ka $\phi(B)$ un $\theta(B)$ saknes atrodas ārpus slēgta vienības riņķa $\{z : |z| \leq 1\}$ un $d \in (-1, \frac{1}{2})$, ARFIMA(p, d, q) process ir stacionārs un apgriežams. Tādā gadījumā mēs varam uzrakstīt

$$y_t = (1 - B)^{-d} \phi(B)^{-1} \theta(B) \varepsilon_t = \psi(B) \varepsilon_t$$

un

$$\varepsilon_t = (1 - B)^{d} \phi(B) \theta(B)^{-1} y_t = \pi(B) y_t.$$

$MA(\infty)$ koeficienti ψ_j un $AR(\infty)$ koeficienti π_j apmierina sekojošas asimptotiskās sakarības:

$$\psi_j = \frac{\theta(1)}{\phi(1)} \frac{j^{d-1}}{\Gamma(d)} + O(j^{-1})$$

$$\pi_j = \frac{\phi(1)}{\theta(1)} \frac{j^{-d-1}}{\Gamma(-d)} + O(j^{-1})$$

3. Kolmogorova-Smirnova tests vienkāršu hipotēžu pārbaudei

Mūsu mērķis ir vienkārša hipotēžu pārbaude datiem ar ilglaicīgu atmiņu

$$H_0 : F = F_0 \text{ pret } H_1 : F \neq F_0$$

Tai mēģināsim pielāgot Kolmogorova-Smirnova statistiku

$$K_n = \sup_{x \in R} |F_n(x) - F_0(x)|,$$

kur F_n ir empīriskā sadalījuma funkcija, kuru definē

$$F_n(x) = \frac{1}{n} \sum_{i=1}^{n} I\{x_i \leq x\},$$

kur x_i ir procesa novērojumi un I ir indikator-funkcija.

Pieņem, ka mums ir zināma interesējošā procesa $MA(\infty)$ reprezentācija

$$y_t = \sum_{k=0}^{\infty} b_k \varepsilon_{t-k},$$

kur ε_t ir ar nulles vidējo vērtību un vienības dispersiju. Pieņem, ka konstantes b_k apmierina $b_k = 0 \ \forall k < 0, b_1 = 0$ un $b_t \sim c_0 t^{-(1-d)}$, kad $t \to \infty$ kādām $0 < c_0 < \infty$ un $0 < d < \frac{1}{2}$.

No publikācijas [1] mums ir zināms, ka pie H_0

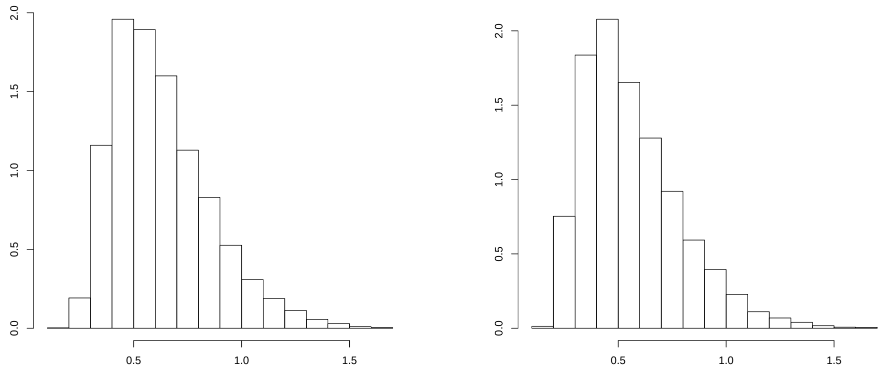
$$n^{\frac{1}{2}-d} K_n \to_D c^{\frac{1}{2}}(\theta)|Z|\|f_0\|_\infty \Longrightarrow \frac{n^{\frac{1}{2}-d} K_n}{c^{\frac{1}{2}}(\theta)\|f_0\|_\infty} \to_D |Z|,$$

kur $Z \sim N(0,1)$, $\|f_0\|_\infty = \sup_{x \in R} f_0(x)$, $\theta = (c_0, d)$ un

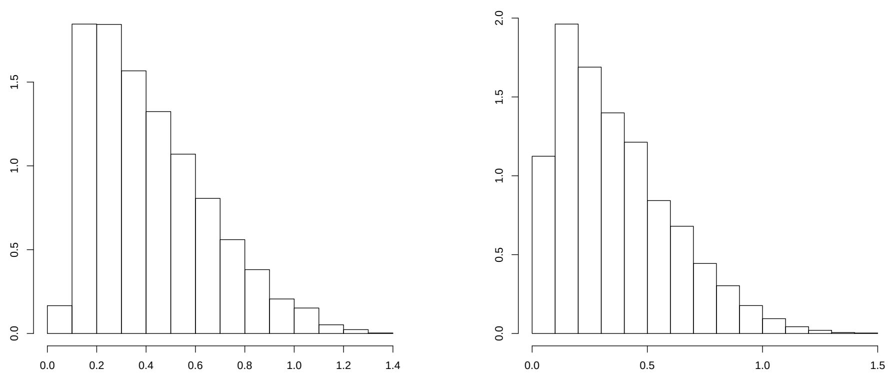
$$c(\theta) = \frac{c_0^2 B(d, 1-2d)}{d(1+2d)},$$

kur $B(a,b) = \int_0^1 x^{a-1}(1-x)^{b-1}\, dx$.

Ar datorprogrammas R palidzību simulēsim statistikas $n^{\frac{1}{2}-d} K_n$ sadalījumu, pie $n = 100$ un $n = 1000$, simulāciju skaitā $m = 10000$ ARFIMA(0,d,0) procesam.



3.1. att.: Simulētais attiecīgi pie $n = 100$ un $n = 1000$ statistikas sadalījums $ARFIMA(0, d, 0)$ procesam pie $d = 0.1$.



3.2. att.: Simulētais attiecīgi pie $n = 100$ un $n = 1000$ statistikas sadalījums $ARFIMA(0, d, 0)$ procesam pie $d = 0.4$.

4. Simulācijas

Sājā nodaļā mēs mēģināsim pielietot ieprēkšaprakstīto testu simulētiem datiem. Sākuma ar datorprogrammas R palīdzību ģenerēsim procesu ar ilglaicīgu atmiņu kā ARFIMA procesu X_t. Tālāk mēs pielietosim transformāciju

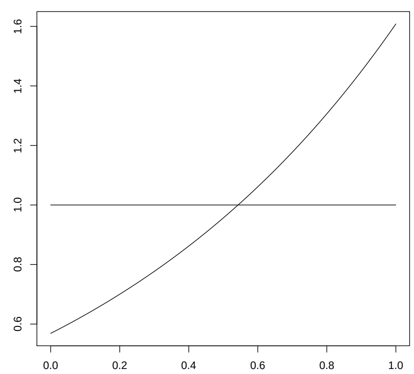
$$Y_t = F(X_t),$$

kur $F(x)$ ir procesa X_t teorētiskā sadalījuma funkcija. Tad process Y_t ir vienmērīgi sadalīts intervālā $(0, 1)$. Šādā veidā mēs simulēsim kritisko vērtību. Nākošā solī mēs veiksim testa jaudas analīzi. Šīm nolūkam mēs ģenerēsim alternatīvu no [3] izskatā

$$g(x) = \exp(\sum_{j=1}^{k} \theta_j \pi_j(x) - c(\theta)),$$

kur θ ir parametru vektors, $\pi_j(x)$ ir ortonromālie Ležandra polinomi un $c(\theta)$ ir konstante tāda, lai integrālis no $g(x)$ būtu 1. Mūsū konkrētā gadījumā izvēlēsimies $k = 1$

$$g(x) = \exp(0.3\sqrt{12}(x - 0.5) - c).$$



4.1. att. Alternatīva 4. kopā ar vienmērīgo blīvuma funkciju.

Ar $G(x)$ apzimēsim alternatīvas sadalījuma funkciju

$$G(x) = \int_0^x g(x)\, dx$$

Tagad mēs pielietojam transformāciju

$$A_t = G^{-1}(Y_t)$$

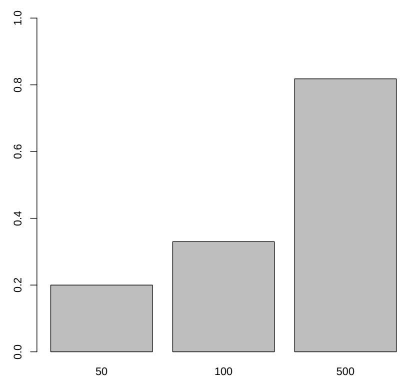
un iegūstam procesu A_t kurš nav vienmērīgi sadalīts. Priekš jaudas analīzes mēs 1000 reizes simulēsim procesu A_t un aprēķināsim, kādu daļu no simulācijam mūsu test noraidīs

$$H_0\colon\ A_t \sim U(0,1)$$

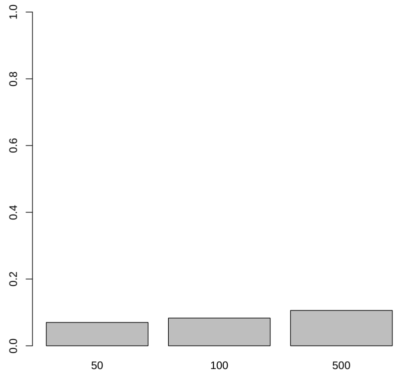
Simulāciju laikā radās grūtības ģenerēt ARFIMA procesu ar noteikto teorētisku sadalījumu, tāpēc bija ģenerēts process garumā nm, kur m ir simulāciju skaits un n ir simulēta procesa plānots garums. Teorētisko sadalījuma funkciju mēs aproksimējam ar empīrisko sadalījuma funkciju un sakotnējus datus sadalām m apakškopas izmērā n. Pēc tam mēs aprēķinam testa statistiku katrai datu apakškopai un pārbaudam cik rezultāti pārsniedz simulēto kritisko vērtību. Zemāk ir aplūkoti simulāciju rezultāti.

4.1. tabula: Jaudas analīzes rezultāti alternatīvai 4. ARFIMA$(1, d, 0)$ procesam (ar Cr apzīmēsim simulēto kritisko vērtību)

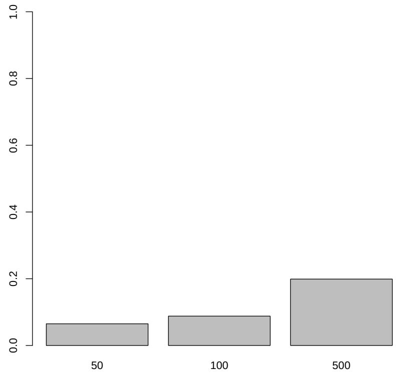
ϕ	d	n	Cr	Jauda
0.3	0.1	50	1.290	0.200
0.3	0.1	100	1.259	0.330
0.3	0.1	500	1.247	0.818
0.3	0.4	50	0.884	0.070
0.3	0.4	100	0.784	0.083
0.3	0.4	500	0.769	0.106
0.9	0.1	50	2.879	0.065
0.9	0.1	100	2.868	0.088
0.9	0.1	500	2.961	0.199
0.9	0.4	50	1.259	0.061
0.9	0.4	100	1.160	0.063
0.9	0.4	500	1.080	0.079



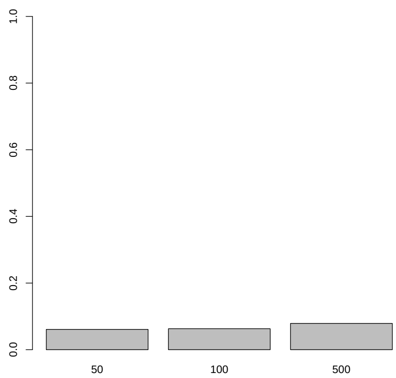
4.2. att. Jaudas atkarība no datu apjoma n pie $\phi = 0.3$ un $d = 0.1$.



4.3. att. Jaudas atkarība no datu apjoma n pie $\phi = 0.3$ un $d = 0.4$.

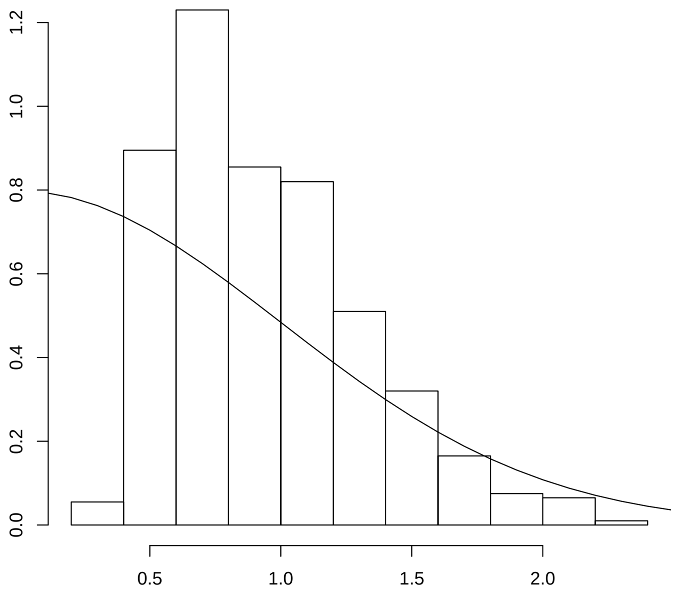


4.4. att. Jaudas atkarība no datu apjoma n pie $\phi = 0.9$ un $d = 0.1$.

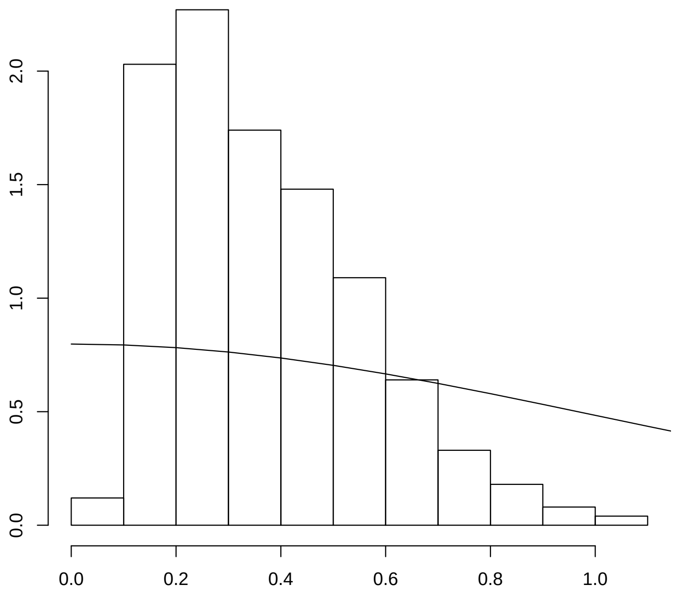


4.5. att. Jaudas atkarība no datu apjoma n pie $\phi = 0.9$ un $d = 0.4$.

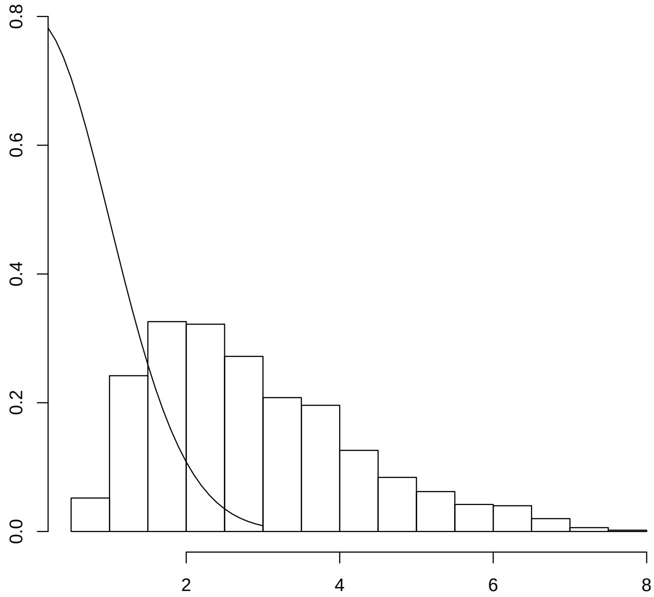
Zemāk attēlosim šādi simulētiem datiem statistikas $\frac{n^{\frac{1}{2}-d}K_n}{c^{\frac{1}{2}}(\theta)\|f_0\|_\infty}$ empīrisko un teorētisko sadalījumus pie $n = 100$.c



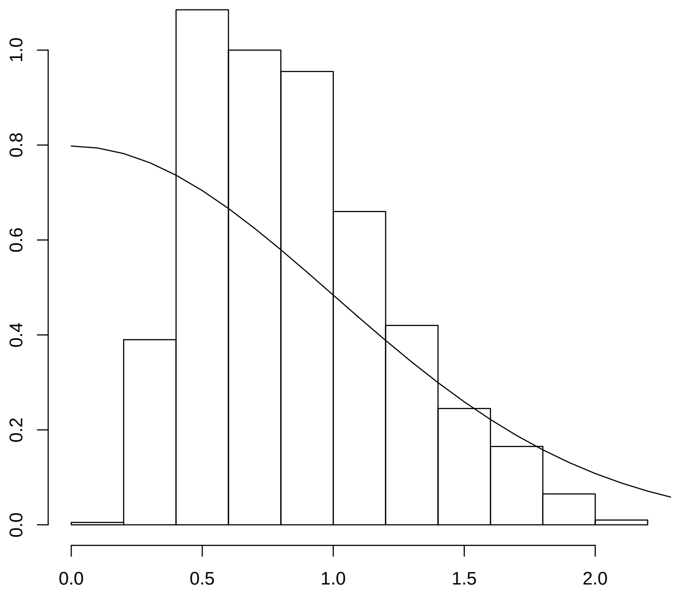
4.6. att. Statistikas empīriskais un teorētiskais sadalījumi pie $\phi = 0.3$ un $d = 0.1$.



4.7. att. Statistikas empīriskais un teorētiskais sadalījumi pie $\phi = 0.3$ un $d = 0.4$.



4.8. att. Statistikas empīriskais un teorētiskais sadalījumi pie $\phi = 0.9$ un $d = 0.1$.



4.9. att. Statistikas empīriskais un teorētiskais sadalījumi pie $\phi = 0.9$ un $d = 0.4$.

Secinājumi

No veiktas jaudas analīzes ir redzams, kā pie lielākiem parametru ϕ un d vērtībām testa jauda ir stipri mazāka par 1, uz kuru tai teorētiski jākonverģē.

Darba gaitā radās problēmas ģenerēt ilglaicīgi atkarīgu procesu ar noteiktu sadalījumu ar noteiktiem parametriem. Tāpēc kā teorētiskās funkcijas aproksimāciju nācās izmantot empīrisko sadalījuma funkciju, ka protams, ietekmē rezultātu precizitāti. Vēl viens jautājums ir, vai apskatīt statistiku $n^{\frac{1}{2}-d}K_n$, kurai sadalījums ir atkarīgs no procesa parametriem, bet paša statistika nav ierobežota ar ARFIMA procesu klasi, kā mēs arī darījām šajā darbā. Otrais variants ir apskatīt statistiku $\frac{n^{\frac{1}{2}-d}K_n}{c^{\frac{1}{2}}(\theta)\|f_0\|_\infty}$, kurai ir fiksēts sadalījums un, kā sekas, ir zināmās kritiskās vērības, bet reāliem datiem tad būtu jāpielāgo ARFIMA modeli un jānovērtē parametri.

Nākotnē vajadzētu pārbaudīt testa jaudu vairākām alternatīvām un jāizmēģina, kā tas darbosies uz reāliem datiem.

Izmantotā literatūra un avoti

[1] Donatas Surgailis Hira L. Koul. Goodness-of-fit testng under long memory. *Journal of Statistical Planning and Inference*, 2010.

[2] Wilfredo Palma. *Long-Memory Time Series Theory and Methods*. John Wiley & Sons, 2007.

[3] Teresa Ledwina Wilbert C. M. Kallenberg. Consistency and monte carlo simulation of a data driven version of smooth goodness-of-fit tests. *The Annals of Statistics*, 23, 1995.

Programmas R kods

```
###################
#Procesa simulēšana
###################
library(fracdiff)
a<-0.9
d<-0.4
b<-NULL
x<-fracdiff.sim(1000, ar=a, ma=b, d)$series
ts.plot(x,ylab='',xlab='',main='')
acf(x,ylab='',xlab='',main='')

y<-arima.sim(n = 1000, list(ar = a, ma =b))
ts.plot(y,ylab='',xlab='',main='')
acf(y,ylab='',xlab='',main='')

######################
#Sadalījuma simulēšana
######################
library(fracdiff)
library(nortest)
n<-100
a<-NULL
b<-NULL
d<-0.1
k<-c()
disp<-c()

#Dispersija
for (i in 1:1000)
{
x<-fracdiff.sim(1000, ar=a, ma=b, d)$series
disp[i]<-sd(x)
}
sdt<-mean(disp)
sdt

f0<-pnorm(0,0,sdt)

for(i in 1:10000)
{
x<-fracdiff.sim(n, ar=a, ma=b, d)$series
Kn<-ks.test(x,"pnorm",0,sdt)$statistic[[1]]
k[i]<-n^(1/2-d)*Kn
}
hist(k,prob=T,main='',xlab='',ylab='')

###############
#Jaudas analīze
```

```
###############
library(fracdiff)
disp<-c()
rez<-c()
k<-c()
m<-1000
n<-100
a<-0.9
b<-NULL
d<-0.4

set.seed(1)

dati<-fracdiff.sim(m*n, ar=a, ma=b, d)$series

#Empiriska sadalijuma funkcija
Fn<-ecdf(dati)

#Kritiska vertiba
for(i in 1:m)
{
x<-dati[((i-1)*n+1):(i*n)]
xu<-Fn(x)
Kn<-ks.test(xu,"punif",0,1)$statistic[[1]]
rez[i]<-n^(1/2-d)*Kn
}
crit<-quantile(rez,0.95)[[1]]
crit

### Lezandra polinomi
P1<-function(x) (x-0.5)*sqrt(12)
P2<-function(x) sqrt(5)*(6*(x-0.5)^2-0.5)
P3<-function(x) sqrt(7)*(20*(x-0.5)^3-3*(x-0.5))
P4<-function(x) 210*(x-0.5)^4 - 45*(x-0.5)^2 + 9/8
P5<-function(x) sqrt(11)*(63*(2*x-1)^5-70*(2*x-1)^3+15*(2*x-1))/8
P6<-function(x) sqrt(13)*(231*(2*x-1)^6-315*(2*x-1)^4+105*(2*x-1)^2-5)/16
P7<-function(x) sqrt(15)*(429*(2*x-1)^7-693*(2*x-1)^5+315*(2*x-1)^3-35*(2*x-1))/16
P8<-function(x) sqrt(17)*(6435*(2*x-1)^8-12012*(2*x-1)^6+6930*(2*x-1)^4-1260*(2*x-1)^2+35)/128
P9<-function(x) sqrt(19)*(12155*(2*x-1)^9-25740*(2*x-1)^7+18018*(2*x-1)^5-4620*(2*x-1)^3+315*(2*x-1))/128
P10<-function(x) sqrt(21)*(46189*(2*x-1)^{10}-109395*(2*x-1)^8+90090*(2*x-1)^6-30030*(2*x-1)^4+3465*(2*x-1)^2-63)/256

## Alternativa g1 (exponential family) no Ledw&Kal(1995)
theta<-0.3
f.const<-function(x) exp(theta*P1(x))
const<-log(integrate(f.const,0,1)[[1]]); const
alt.exp<-function(x){
exp(theta*P1(x) - const)
}
```

```
##bisektrisu metode
bis<-function(x0,y0,funf,prec)
{
a<-x0;b<-y0 # funkcijai jabut punkta x0 negativai un punkta y0 pozitivai
while (b-a>prec)
{
if (funf(a+(b-a)/2)>0) b<-a+(b-a)/2 else a<-a+(b-a)/2
}
a+(b-a)/2
}

alt<-function(n,i)
{
x<-dati[((i-1)*n+1):(i*n)]
xu<-Fn(x)
Data<-c()
for(i in 1:n)
{
aimfun<-function(x)
{
integrate(alt.exp,0,x)[[1]]-xu[i]
}
Data[i]<-bis(0,1,aimfun,0.001)
}
Data
}

#######################
# Alternativas generesana #
#######################
for (i in 1:m)
{
xa<-alt(n,i)
Kn<-ks.test(xa,"punif",0,1)$statistic[[1]]
k[i]<-n^(1/2-d)*Kn
}
jauda<-length(k[k>crit])/m
jauda

####################
#Teoretiskais sadalijums
####################
f0<-dunif(0,0,1)
c0<-(1+sum(b))/(1+sum(a))/gamma(d)
B<-function(a,b)
{
z<-function(x)
{
x^(a-1)*(1-x)^(b-1)
```

```
}
integrate(z,0,1)$value
}

c<-c0^2*B(d,1-2*d)/(d*(1+2*d))
kteor<-c()
for (i in 1:m)
{
x<-dati[(n*(i-1)+1):(n*i)]
xu<-Fn(x)
Kn<-ks.test(xu,"punif",0,1)$statistic[[1]]
kteor[i]<-n^(0.5-d)*Kn/(c^(0.5)*f0)
}
xx<-seq(0,3,by=0.1)
hist(kteor, prob=T, main=NULL, xlab=NULL, ylab=NULL)
lines(xx,2*dnorm(xx,0,1))

crit
jauda
```

darbs “Kolmogorova-Smirnova tests datiem ar ilglaicīgu atmiņu” izstrādāts LU Fizikas un Matemātikas fakultātē.

Ar savu parakstu apliecinu, ka pētījums veikts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai.

Autors: Dmitrijs Bosņaks

____________________ ________________

(paraksts) (datums)

Rekomendēju darbu aizstāvēšanai.

Vadītājs: doc. Dr.math. Jānis Valeinis

____________________ ________________

(paraksts) (datums)

Recenzents:

____________________ ________________

(paraksts) (datums)

Darbs iesniegts Matemātikas nodaļā ________________

(datums)

____

(darbu pieņēma)

Darbs aizstāvēts gala pārbaudījuma komisijas sēdē

________________ prot. Nr. ________, vērtējums________

(datums)

Komisijas sekretārs/-e: ________________ ________________

(Vārds, Uzvārds) (paraksts)