PADĖKA

Norėčiau padėkoti mano baigiamojo darbo vadovui prof. V. Bagdonavičiui už suteiktą pagalbą darbo rašymo metu, svarbias pastabas ir pasiūlymus.
Dėkoju recenzentui doc. R.Lapinskui už naudingas pastabas.
Dėkoju docentei R.Levulienei už suteiktą mokymąsi priemonę ir naudingus patarimus naudojantis statistiniu paketu SAS.
Dėkoju Ingai Maslovai už vertingas pastabas.

Įvadas

Šiame darbe aprašomi du pagreitintų bandymų modeliai – parametrinis ir neparametrinis.

Pirmame skyrelyje aptartas parametrinio pagreitintų bandymų modelio atvejis.

Antrame skyrelyje aptartas semiparametrinis pagreitintų bandymų modelio atvejis.

Trečias skyrelis skirtas apkrovos pasikeitimo atsitiktiniais momentais atvejo modeliavimui esant neparametriniam pagreitintų bandymų modeliui. Šio skyrelio tikslas - neparametrinio patikimumo funkcijos įverčio, ir pagreitinimo konstantos įverčio apskaičiavimas iš pagreitintų bandymų duomenų.

Ketvirtame skyrelyje pateiktas apkrovos pasikeitimo atsitiktiniais momentais atvejo modeliavimas esant parametriniam pagreitintų bandymų modeliui.

Pagrindinė šio darbo užduotis - neparametriniais ir parametriniais metodais gautų patikimumo įvertinių charakteristikų ir savybių palyginimas, naudojant pagreitintu bandymų duomenis.

Visi skaičiavimai atlikti su programa „SAS“. Prieduose bus pateiktas SAS programinis kodas, bei rezultatai.

1. Parametrinis pagreitintų bandymų modelis

1.1 Pagreitintų bandymų modelio parametrizacija

Tegu

yra daugiamatė kintanti laike kovariantė; čia ir yra vienmatės kovariantės.

Pagal pagreitintų bandymų modelį, išgyvenamumo funkcija, kai kovariantė yra , yra

(1.1)

Jei kovariantė kintant laikui yra pastovi, tada (1.1) modelis užrašomas taip:

(1.2)

Funkcija r yra parametrizuojama tokia forma:

(1.3)

kur yra nežinomų parametrų vektorius ir yra žinomų funkcijų , su , vektorius.

Pagal parametrizuotą pagreitintų bandymų modelį, išgyvenamumo funkcija (1.1) yra:

(1.4)

Vistiek naudojame tą patį žymėjimą x.

Jei kovariantės yra pastovi, tada (1.4) modelis yra rašomas taip:

(1.5)

ir gedimo laiko logaritmas prie x gali būti užrašytas kaip , kur atsitiktinio kintamojo išgyvenimo funkcija nepriklauso nuo x ir yra .

Pažymime, kad lognormalaus laiko iki gedimo pasiskirstymo atveju, pasiskirstymas yra normalus ir turime standartinį daugialypės tiesinės regresijos modelį.

1.1.1 Tolydžios kovariantės

Iš pradžių, tarkime, kad kovariantės yra toludžios (krūvis, temperatūra, įtampa, slėgis).

Jei (1..2) modelyje x įgija reikšmes aibėje , tada visiems , :

, (1.6)

kur funkcija rodo mastelio kitimo laipsnį. Akivaizdu, kad .

Pirma, tarkime, kad x yra vienmatis. Mastelio kitimo greitis x atžvilgiu yra apibrėžiamas nykstamojo dydžio charakteristika:

(1.7)

Taigi, visiems , funkcija r(x) yra duota formulėje:

, (1.8)

kur yra fiksuota kovariantė.

Tarkim, kad yra proporcingas nustatytai funkcijai u(x):

.

Šiuo atveju

, (1.9)

kur yra pirmykštė funkcija, o , yra nežinomi parametrai.

1.1.2 Diskrečios ir nominalios kovariantės

Jei kovariantės yra diskrečios, tada funkcijų pavidalas turi tokį patį pavidalą, kaip ir tolydžių kovariančių atveju, t.y. gali būti , lnx arba .

Jei j-toji kovariantė yra nominali ir apima skirtingų reikšmių, tada yra suprantamas kaip -matis vektorius , apimantis skirtingas reikšmės.

ir yra -matis:

.

Taigi, jei j-toji kovariantė yra nominali ir kitos yra tolydžios arba diskrečios, tada

. (1.10)

Gautasis modelis yra ekvivalentus (1.5) modeliui su vienmačių kovariančių. Jei , kovariantė yra dichotominė(dalijanti), apimanti dvi reikšmes: 0 arba 1.

1.2 Regresijos koeficientų interpretacija

Tarkim, kovariantės kintant laikui yra konstantos. Tada, pagal pagreitintų bandymų modelį (1.5), p-asis gedimo laiko kvantilis yra

. (1.11)

Taigi, logaritmas

, (1.1**2)**

čia . Tegu yra vienetų prie x vidutinis ilgaamžiškumas.

Tada

(1.13)

ir logaritmas

(1.14)

yra tiesinė regresijos parametrų funkcija; čia .

Pažymime vidurkių ir kvantilių santykį atitinkamai:

ir (1.15)

Pagreitintų bandymų modeliui:

. (1.16)

Taigi, yra vidurkių santykis kovariantėms x ir y atitinkamai.

Aptarkime parametrų interpretaciją (1.5) modelyje.

1.2.1 Modeliai be sąveikos

a)Tolydžios arba diskrečios kovariantės

Tarkim, kad j-oji kovariantė yra tolydi arba diskreti. Tada

(1.17)

yra vidurkių santykis, atitinkantis pasikeitimą matavimo vienetu.

b)Nominalios kovariantės

Tarkim, kad yra nominali. Pirmoji jos reikšmė yra ir (i+1)-oji reikšmė yra , kur i-oji koordinatė yra vienetas. Tada

(1.18)

yra vidurkių santykis, atitinkantis pasikeitimą nuo pirmos iki (i+1)-os reikšmės.

1.2.2 Modeliai su iteracijomis

Jei j-osios kovariantės įtaka vidutiniam ilgaamžiškumui yra skirtinga esant įvairioms kovariančių reikšmėms, tada yra sąveika tarp kovarančių ir modelis turi būti pakeistas.

a) Sąveika tarp tolydžių arba diskrečių kovariančių

Jei yra dvi tolydžios arba diskrečios kovariantės ir yra sąveika tarp jų, tada

. (1.19)

Trims kovariantėms:

, (1.20)

ir taip toliau.

Dviejų kovariančių atveju, vidurkių santykis

(1.21)

priklauso nuo reikšmės.

Taigi,

(1.22)

yra vidurkių santykis, atitinkantis pakitimą matavimo vienetu, kai kitos kovariantes fiksuotos.

b) Sąveika tarp tolydžių arba diskrečių ir nominalių kovariančių

Tarkim, yra dvi kovariantės: yra tolydi arba diskreti ir yra nominali, su galimom reikšmėm. Tada

(1.23)

ir vidurkių santykis

(1.24)

piklauso nuo reikšmės.

Taigi, šiame pavyzdyje yra vidurkių santykis, atitinkantis pakitimą nuo pirmos iki (i+1)-osios reikšmės, kai kita kovariante fiksuota.

c) Sąveika tarp nominalių kovariančių

Tarkim, kad abi, ir , yra nominalios, kiekviena su trim reikšmėm. Tada:

, ,

ir

Šiuo atveju santykis

priklauso nuo .

Taigi, yra vidurkių santykis, atitinkantis pakitimą nuo pirmos iki antros reikšmės, kai kitos kovariantės fiksuotos.

Abibendrinimas yra akivaizdus, jei kovariantės apima tris ar daugiau reikšmių.

1.2.3 Priklausantys nuo laiko regresijos koeficientai

Apsvarstykime pagreitintų bandymų modelį su priklausančiomis nuo laiko koeficientais:

(1.25)

Aptarsime koeficientus formoje , (i=1,2,...,m), kur yra kažkokios apibrėžtos determinuotosios funkcijos arba numatomų procesų realizacijos. Tokiu atveju, pagreitintų bandymų modelis su priklausančiais nuo laiko koeficientais ir konstantomis arba priklausančiomis nuo laiko kovariantėmis gali būti užrašytas (5.4) forma su skirtinga kovariančių interpretacija.

Iš tikrųjų,

,

(1.26)

Tada

.

Taigi, pagreitintų bandymų modelis su priklausančiais nuo laiko regresijos koeficientais gali būti užrašytas formoje:

. (1.27)

Turime pagreitintų bandymų modelį, kur nežinomi parametrai ir kovaraintės yra apibrėžtos (1.26).

1.3 Patikimumo duomenų regresinė analizė: pasiskirstymų šeimos, priklausančios nuo mastelio ir formos parametrų

1.3.1 Modelis ir duomenys

Aptarkime pagreitintų bandymų modelį:

(1.28)

arba pagreitintų bandymų modelį (1.25) su priklausančiais nuo laiko regresijos koeficientais ir, tarkim, kad priklauso mastelio-formos išgyvenimo funkcijų klasei:

, .

Pavyzdžiui, t>0:

, , ,

tada gauname atitinkamai Veibulo, loglogistinį, lognormalųjį skirstinius. Čia yra standartinio normalaus dėsnio pasiskirstymo funkcija. Parametras gali būti įtrauktas į koeficientą , taigi, tarkime, kad , .

Pažymime, kad jei pagreitintų bandymų modelis su priklausančiais nuolaiko regresijos koeficientais yra nagrinėjamas, tada, netgi kai kovariantės yra konstantos, (1.25) modelis susiveda į (1.27) modelį, kuris yra eksivalentus (1.28) su priklausančiomis nuo laiko kovariantėmis .

Taigi, visi rezultatai, gaunami pagreitintų bandymų modeliui su priklausančiomis nuo laiko kovariantėmis, gali būti perrašyti pagreitintų bandymų modeliui su priklausančiais nuo laiko regresijos koeficientais ir konstantomis arba priklausančiomis nuo laiko kovariantėmis. Visose formulėse

m, ,

turi būti pakeistos atitinkamai į

2m, , .

Eksperimentų planas:

Stebima n gaminių. i-asis gaminys yra tikrinamas prie kovariantės , galimos kintančos laike.

Duomenys nepriklausomai cenzūruoti is dešinės. Tegu ir bus i-ojo vieneto gedimo ir cenzūravimo laikai , , .

Pažymime išgyvenamumo funkciją - . (1.4) modelis gali būti užrašytas formoje

. (1.29)

Jei yra konstanta, tada

, (1.30)

kur , .

Pažymime, kad atsitiktinio dydžio

(1.31)

išgyvenamumo funkcija yra . Pastoviai :

.

Pažymime:

, . (1.32)

Veibulo dėsniui:

, , , ; (1.33)

Loglogistiniam dėsniui:

, , , ; (1.34)

Lognormaliam dėsniui:

, , , , (1.35)

su .

1.3.2 Regresijos parametrų maksimalaus tikėtinumo įvertis

Maksimalaus tikėtinumo funkcija yra

x

x . (1.36)

Jei yra konstanta, tada tikėtinumo funkcija yra:

. (1.37)

Informantės yra

, (l=0,1,...,m),

; (1.38)

čia , ,

, (1.39)

ir funkcija h yra duota (1.32). Kai kovariantės yra konstantos, informantės yra

, (l=0,1,...,m),

, (1.40)

kur

, (1.41)

ir yra duotas (5.43).

Maksimalaus tikėtinumo įverčiai , gaunami sprendžiant lygčių sistemą

, (l=0,1,…,m+1).

1.3.3 Pagrindinių patikimumo charakteristikų įverčiai

Tarkim, kad = yra bet kokia kovariantė, kuri gali skirtis nuo , (i=1,...,n).

Išgyvenamumo funkcijos įvertis:

. (1.42)

Kai x yra konstanta, išgyvenamumo funkcijos įvertis yra

. (1.43)

Jei pagreitintų bandymų modelis su priklausančiais nuo laiko regresijos koeficientais yra geras, tada

,

kur , .

p-ojo kvantilio įvertis

Įvertis tenkina lygtį:

. (1.44)

Jei x yra konstanta, tada

. (1.45)

Gedimo laiko vidurkio įvertis

(1.46)

Jei x konstanta, tada

. (1.47)

Vidurkių santykių įverčiai

Vidurkių santykis MR(x,y) (žr. (5.27)) yra įvertinamas

. (1.48)

1) Modeliai be sąveikų.

a) Tolydi arba diskreti kovariantė .

Vidurkių santykis (žr. (5.28)) yra įvertinamas

. (1.49)

b) Nominali kovariantė .

Vidurkių santykis (žr. (5.29)) yra įvertinamas

. (1.50)

2) Modeliai su sąveikom.

a) Tolydi (diskreti) x Tolydi (diskreti) kovariantė.

Jei dvi tolydžios (diskrečios) kovariantės, pavyzdžiui, ir sąveikauja, tada vidurkių santykis, (žr. (5.32)), yra įvertinamas

. (1.51)

b) Tolydi (diskretūs) x nominali kovariantė.

Jei tolydi (diskreti) kovariantė sąveikauja su kovariante , kuri įgyja skirtingų reikšmių, tada vidurkių santykis (žr. (1.24) modelį) įvertinamas

$$. \qquad (1.52)$$

c) Nominali x nominali kovariantė.

Jei dvi nominalios kovarantės ir (su, sakykim, trim galimom reikšmėm) sąveikauja, tada vidurkių santykis yra įvertinamas

$$. \qquad (1.53)$$

2 Semiparametrinis vertinimas pagreitintuose bandymuose

Tarkim, kad bazinė išgyvenimo funkcija pagreitintų bandymų modelyje (2.1) yra visiškai nežinoma. Kadangi šiame darbe akcentuojamas neparametrinis vertinimas, tai naudosime ne anksčiau aprašytą eksperimentų planą, bet planą, kuris leidžia įvertinti nežinomas funkcijas S_0 ir r.

2.2.1 Antras eksperimentų planas: neapibrėžtos ir r

Tarkim, kad pagreitintų bandymų modelis (5.1) teisingas. Šis modelis gali būti užrašytas:

$$. \qquad (2.1)$$

čia yra patikimumo funkcija prie apkrovos , $x^{(0)}$ - pastovi normali apkrova, r=r(x) – funkcija, apibrėžta aibėje E. ir $r(x_0)=1$.

Tarkim, kad funkcijos r ir yra visiškai nežinomos. Tada naudojames antru eksperimento planu:

Tegu bus pagreitinta, pastovi apkrova .

Bandomos dvi grupės. Pirma vienetų grupė bandoma prie pagreitintos apkrovos ir gaunama pilna imtis . Antra vienetų grupė bandoma prie laiptuotos apkrovos

,

ir gaunama I tipo cenzūruota imtis , .

Čia yra antros grupės vienetų gedimų atsitiktinis skaičius.

Momentas turėtų būti toks, kad po perjungimo nuo apkrovos iki , intervale prie įprastos apkrovos įvyksta pakankamai didelis gedimų skaičius.

Pažymime:

 - pagreitinimo konstanta,

, , , (i=0,1,2),

,

Toliau funkcijos r reikšmę taške $x^{(0)}$ žymėsime taip pat r.

.

Apibrėžiam Nelson-Aalen tipo „įvertinį“:

(2.2)

ir Kaplano-Meierio tipo „įvertinį“

, (2.3)

kur .

Informantinė funkcija apibrėžiama kaip

, (2.4)

arba .

Tai didėjanti pažingsninė funkcija, , , su tikimybe 1.

Parametro r įvertis:

. (2.5)

Išgyvenimo funkcija yra apskaičiuojama:

. (2.6)

Kvantilis yra įvertinamas

(2.7)

ir vidutinis gedimo laikas

.

2.2.2 Asimptotinės įverčių savybės

Aptarkime asimptotines įverčio savybes. Visų pirma, mes gauname funkcijos įvertį, kuris gali būti užrašytas taip:

, (2.8)

kur yra empirinė išgyvenimo funkcija i-ajai vienetų grupei:

, , i=1,2.

Tarkim, kad , . Tada

intervale (2.9)

kur , (i=1) arba , (i=2), yra nulinio vidurkio Gauso procesas, toks, kad visiems :

, (i=1,2). (2.10)

Pažymime

. (2.11)

6.1 Teiginys:

Tarkim, kad yra absoliučiai tolydžios ir visiems . Tada

, (2.12)

kur

, ,

6.2 Teiginys:

Pagal 6.1 teiginio prielaidą ir pagreitintų bandymų modelį,

, (2.14)

ir

, . (2.15)

6.1 Pastaba:

Jei ir yra dideli, tada

,

kur , yra standartinio normalaus pasiskirstymo -kvantilis ir yra išgyvenimo funkcijos įvertis.

Taigi, apytikslus pasikliautinasis intervalas yra , kur

, . (2.16)

6.3 Teiginys:

Tarkim, kad tankiai yra tolydūs ir teigiami intervale , (i=1.2).

Tada

. (2.17)

6.2 Pastaba:

Gali būti įrodyta, kad

, (2.18)

kur yra gedimų intensyvumas prie įprastos apkrovos .

Aptarkime asimptotines įverčio savybes.

Tai gali būti užrašyta formoje:

(2.19)

ir

, (2.20)

kur , .

6.4 Teiginys:

Pagal 6.3 teiginio prielaidą

. (2.21)

6.1 Išvada:

Pagal teiginių prielaidas, visiems

, , (2.25)

, (2.26)

kur

. (2.27)

3 Apkrovos pasikeitimo atsitiktiniais momentais atvejo modeliavimas esant neparametriniam pagreitintų bandymų modeliui

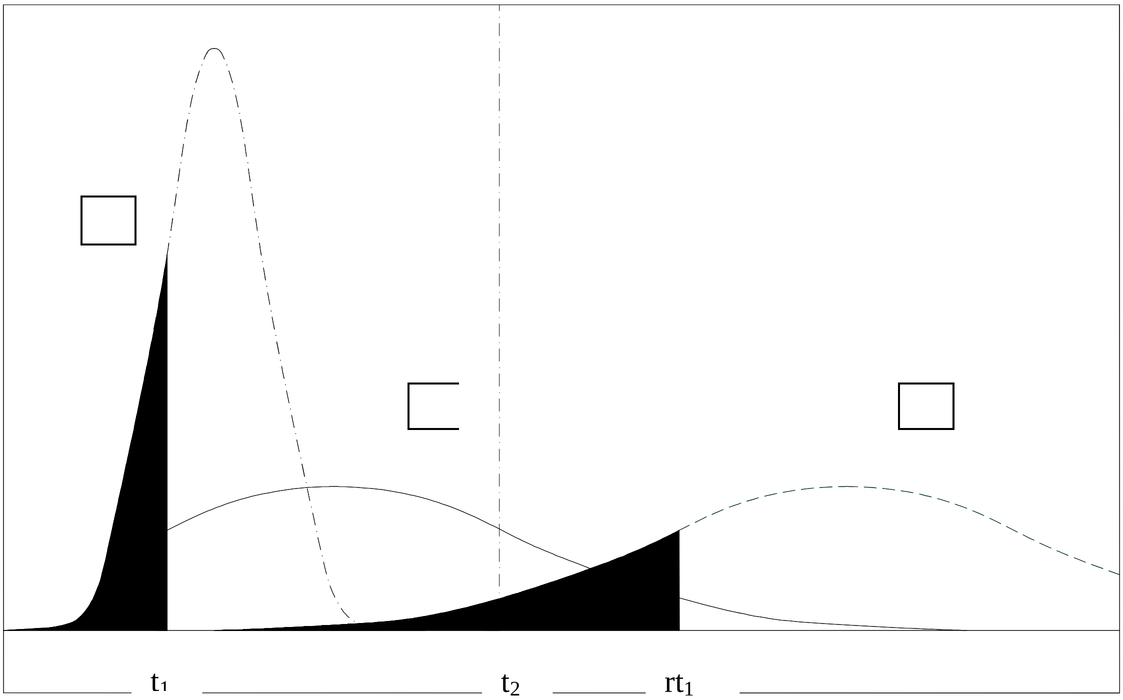
Kaip jau buvo minėta 2.2.1 skyriuje, tarkime bandomos os dvi gaminių grupės. Pirmoji gaminių grupė yra bandoma naudojant padidintą apkrovą ir gaunama pilna imtis . Antra gaminių grupė yra bandoma prie apkrovos iki momento ir po šio momento – prie įprastos apkrovos iki momento , t.y., prie apkrovos

ir gaunama I tipo cenzūruota imtis , .

Taigi, turime

gedimo momentus prie pagreitintos apkrovos ;

gedimo momentus prie apkrovos su perjungimu .



Išgyvenamumo funkcija

.

priklauso nuo nežinomų parametrų ir r. Taigi pagrindinė mūsų užduotis yra rasti išgyvenamumo funkcijos ir pagreitinimo konstantos r įverčius.

Tarkime, kad perjungimo momentas ; antro eksperimento pabaiga ; Turime, kad laiko iki gedimo momentų turi Veibulo skirtisnį

$T(x_0) \sim$ ir $T(x_1) \sim$.

Todel $ET(x_0)=$ ir $ET(x_1)=$. (3.1)

Mūsų modelyje parenkame vidurkius taip, kad pagreitinimo konstanta r=5. T.y kad pagreitintame režime vidutinis skaičius ivykusių gedimų būtų 5 kartus didesnis nei paprastame režime.

Taigi, tarkime, kad

$$ET(x_0)=1000 \text{ ir } ET(x_1)=200.$$

Formos parametras = 2.

Taigi patenkinta sąlyga: r = =5. Apskaičiuojame mastelio parametrus

ir , kur . (3.2)

Generuojame atsitiktinius dydžius w_{1i}, kurie pasiskyrstę tolygiai intervale (0;1) $w_{1i}\sim U(0;1)$, $i=1,\ldots,n_1$.

, ,

iš kur gauname, kad

, kur $i=1,\ldots n_1$.

Analogiškai generuojame atsitiktinius dydžius w_{2i}, $w_{2i}\sim U(0;1)$, $i=1,\ldots,m_1$.

Ieškome w^0: .

Jei $\mathbf{w_{2i}} \leq \mathbf{w^0}$, tai T_{2i} ieškome is lygties .

.

Jei $\mathbf{w_{2i}} > \mathbf{w^0}$, tai T_{2i} ieškome is lygties

; iš čia

.

Taigi turime gedimo momentus T_{1i} ir T_{2i}. Tam, kad apskaičiuotumėme , mums bus reikalinga funkcija U(r), tiksliau jos ženklas.

Kur

nesugedusių elementų skaičius iki momento u iš pirmos grupės;

nesugedusių elementų skaičius iki momento u iš antros grupės.

Analogiškai,

Įvertį ieškosime intervalų dalijimo pusiau metodu.

Tarkim, turim intervalą . Skaičiuojam . Toliau apskaičiuojame ir žiūrime i jos ženklą. Jei >0, tai šaknies ieškome intervale ir o . Jei <0, tai šaknies ieškosime intervale ir o . Nustatę funkcijos ženklą, vel daliname intervalą pusiau ir t.t kol nesurasime tokį r_k : r_m-r_{m-1}< ε. Tai ir bus mūsų ieškomas .

Mūsų atveju nagrinėjame intervalą [0,1;100] t.y. ir ; ε=0,01.

Radus , apskaičiuojame sukaupto intensyvumo funkcijos A_0 ir isgyvenamumo funkcijos S_0 įverčius ir .

.

Kai

+

+ .

Pažymime ,

,

.

Kadangi $s>rt_1$, tai = . Dėl tos pačios priežasties =0.

gedimo momentais, kai u= , atvejais kai ir gauname tokį sukauptą instnsyvumo funkcijos įvertį:

.

Analogiškai gauname . Šuoliuko N_2 trūkio taškai yra => trūkio taškai, t.y. u= .Kadangi = ir =0, kai gauname

= .

Dabar N_2 šuoliukai yra momentuose T_{2i} = , is kur .

Kadangi = ir = = - , gauname

= .

Taigi, kai

.

Kai , tai

.

Nes u ≤s, t.y arba . Turime, kad = ir =0.

Gedimo momentams T_{2i} formulės yra analogiškos, tik vietoj T_{1i} imame T_{2i}.

Kai gaunam ir , apskaičiuojam ir :

ir .

Atlikę auksčiau pateiktus skaičiavimus, randame įverčio r reiksmę

R
Result
4,6944

Ir patikmumo funkcijos įverčio grafikas atrodo taip:

Taigi šio skyrelio tikslas buvo - neparametrinio patikimumo funkcijos įverčio, ir pagreitinimo konstantos įverčio apskaičiavimas iš pagreitintų bandymų duomenų.

Kitame skyrelyje, aprašysime parametrinį pagreitintų bandymų modelį, apskaičiuosime r įvertį ir palyginsime gautus rezultatus.

4. Apkrovos pasikeitimo atsitiktiniais momentais atvejo modeliavimas esant parametriniam pagreitintų bandymų modeliui

Tarkime, turime funkciją

kur

$X_{1j}=T_{1j}$, $\delta_{1j}=1$, $X_{2j}=T_{2j}$, $\delta_{2j}=1$, $j=1,...,m_2$ ir $X_{2j}=t_2$, $\delta_{2j}=0$, $j=m_{2+1},...,n_2$

yra mums jau žinomi gedimo momentai.

Apibrėžkime

Kur ir yra žinomos funkcijos.

Taigi maksimalaus tikėtinumo funkcija bus

;

Pažymime

ir

.

Tada

Maksimizuojame funkciją l ir randame įverčius:

```
              PROC NLP: Nonlinear Maximization

                  Optimization Results
                   Parameter Estimates
                                   Gradient
                                  Objective
       N Parameter       Estimate       Function

       1 niu              1.853302    -0.000000111
       2 teta          1103.535354    -3.03485E-10
       3 r                4.919465     0.000000147
```

Matome, kad rezultatas yra beveik arti pradiniu priartėjimų, kurie yra lygus
=2, = 1128,3792 ir r =5.

Kadangi patikimumo funkcija yra . Jos įvertis bus ,
ir patikmumo funkcijos įverčio grafikas atrodys taip:

Palyginimui, apskaičiuosime abeju patikimumo funkcijų iverčių kvadratinius nuokrypius. t.y išsibarstymą apie tikrąją reiksmę:

ir

.

Tarkime, pasirenkame laiko momentus artimus 100,150,200,300,400 ir 450:
t1=100,3824
t2=149,3785
t3=206,0703
t4=300,8241
t5=401,2808
t6=460,7492

Generuojame N kartų (pakankamai didelį skaičiu, pvz 2000)

ir , ir apskaičiuojame jų reikšmes tuose taškuose.
Gauname, kad

.

IŠVADOS

Šiame darbe aprašyti neparametrinis ir parametrinis pagreitintų bandymų modeliai.

Pagrindinė šio darbo užduotis - neparametriniais ir parametriniais metodais gautų patikimumo įvertinių charakteristikų ir savybių palyginimas, naudojant pagreitintu bandymų duomenis.

Neparametrinio pagreitintų bandymų modelio atveju suskaičiavau

- mastelio parametrą
- informantės U(r) įvertį
- pagreitinimo konstantos r įvertį
- sukaupto intensyvumo funkcijos A(t) įvertį
- patikimumo funkcijos S(t) įvertį

bei nubraižiau patikmumo funkcijos įverčio grafiką.

Parametrinio pagreitintų bandymų modelio atveju suskaičiavau

- mastelio parametro įvertį
- formos parametro įvertį
- pagreitinimo konstantos r įvertį
- patikimumo funkcijos S(t) įvertį

bei nubraižiau patikmumo funkcijos įverčio grafiką.

Taigi galima teigti, kad parametrinis modelis yra tikslesnis negu neparametrinis modelis, tuo atveju, jei patikimumo funkcijos skirstinys yra teisingai parinktas. Kitu atveju, kai skirstinys parinktas neteisingai, parametrinis pagreitintų bandymų metodas netinka. Tuo tarpu, naudojantis neparametriniu pagreitintu bandymų modeliu nereikia žinoti funkcijos skirstino, todėl visada gauname gerus rezultatus.

Tą faktą įrodo ir kvadratinio nuokrypio apskaičiavimas. Suskaičiavau išsibarstymą aplink tikrąją reikšmę. Kadangi išsibarstymas neparametrinio modelio yra didesnis už išsibarstymą parametrinio modelio, reiškia gauti rezultatai parametriniu atveju yra tikslesni.

Priedas1.

Šiame priede pateiktas SAS programinis kodas.
Pirma programa “programa” apskaičiuoja □ neparametrinio modelio atveju.
Antra programa “duom” apskaičiuoja □ parametrinio modelio atveju.

```
data programa;          /*  pirma programa*/
input t1 t2 n1 n2 r r0 r1 ET0 ET1 niu;
datalines;
80 500 100 100 5 0.1 100 1000 200 2
;
run;

data teta;
 set programa (keep=ET0 ET1 niu);
 gama=Gamma(1.5);
 teta1=ET0/gama;
 teta2=ET1/gama;
run;

data w0;
 set programa(keep=r t1);
 set teta;
 w0=1-exp(-((r*t1)/teta1)**niu);
run;

data S;
 set programa;
 set teta;
 set w0;

do t=1 to n1;
  w1=UNIFORM(0);
  s1=(teta2)*((-log(1-w1)))**(1/niu);
  w2=UNIFORM(0);
  if w2<w0 then s2 = (teta2)*((-log(1-w2))**(1/niu));
  else if w2>=w0 then s2=t1-r*t1+teta1*((-log(1-
w2)))**(1/niu);
  output;
end;
run;

proc sort data=S out=T1(keep=s1);
 by s1;
run;
```

```
proc transpose data=T1 out=Transp1 prefix=Tn;
 var s1;
run;

proc sort data=S out=T2(keep=s2);
 by s2;
 where s2<=t2;
run;

proc transpose data=T2 out=Transp2 prefix=Tm;
 var s2;
run;

data R;
 set programa;
 set teta;
 set w0(keep=w0);
 set Transp1;
 set transp2;
 array T1i{100} Tn1-Tn100;
 array T2i{100} Tm1-Tm100;
 m2=0;
 epsilon=0.01;
 rk=0;
 r11=0.1;
 r22=100;

do t=1 to n1;
     if (T2i{t} <= t2) and (T2i{t}^=.) then m2=m2+1;
end;

DO while (abs(r22-r11)>=epsilon);
     rk=(r22+r11)/2;
     sumu1=0;
     sumu2=0;

  do t=1 to n1;
    if (T2i{t}>t1) and (T2i{t}^=.) then do;
      sumy1a=0;
      sumy2b1=0;

          do n=1 to n1;
if (T1i{n} >= (t1+((T2i{t}-t1)/rk))) and ((t1+((T2i{t}-
t1)/rk))^=.)  then y1a=1;

else if (T1i{n} < (t1+((T2i{t}-t1)/rk))) and ((t1+((T2i{t}-
t1)/rk))^=.) then y1a=0;
```

```
     sumy1a=sumy1a+y1a;
          end;

                    do nn=1 to m2;
                         if T2i{nn}>=T2i{t} then y2b1=1;
                         else if T2i{nn}<T2i{t} then y2b1=0;
                         sumy2b1=sumy2b1+y2b1;
                    end;

     if T2i{t}<=t2 and (T2i{t}^=.) then y2b2=(n2-m2)*1;
     else if T2i{t}>t2 and (T2i{t}^=.) then y2b2=(n2-m2)*0;
     sumy2t2=sumy2b1+y2b2;
     uu1=sumy1a/(sumy2t2+sumy1a);
     sumu1=sumu1+uu1;
                              end;



     sumy2b11=0;
     sumy1t1=0;
     if T1i{t} >t1 and (T1i{t}^=.) then do;

       do k=1 to n1;
         if T1i{k}>=T1i{t} and (T1i{t}^=.) then y1t1=1;
         else if T1i{k}<T1i{t} and (T1i{t}^=.) then y1t1=0;
         sumy1t1=sumy1t1+y1t1;
       end;
     do kk=1 to m2;
          if (T2i{kk}>=(t1+(rk*(T1i{t}-t1)))) and
          ((t1+(rk*(T1i{t}-t1)))^=.) then y2b11=1;
          else if (T2i{kk}<(t1+(rk*(T1i{t}-t1)))) and
          ((t1+(rk*(T1i{t}-t1)))^=.) then y2b11=0;
          sumy2b11=sumy2b11+y2b11;
     end;
if ((t1+(rk*(T1i{t}-t1)))<=t2) and ((t1+(rk*(T1i{t}-t1)))^=.)
then y2b22=(n2-m2)*1;

else if ((t1+(rk*(T1i{t}-t1)))>t2) and ((t1+(rk*(T1i{t}-
t1)))^=.) then y2b22=(n2-m2)*0;
sumy2b2=sumy2b11+y2b22;
uu2=sumy2b2/(sumy1t1+sumy2b2);
sumu2=sumu2+uu2;

end;
end;
     u=sumu1-sumu2;
     if u>0 then r22=rk;
            if u<0 then r11=rk;
```

```
 END;

result=(r22+r11)/2;
call symput("makro",m2);
run;

data At1;
set R(keep=t1 t2 n1 n2 m2 result);
set Transp1;
set Transp2;
array T1i{100} Tn1-Tn100;
array T2i{100} Tm1-Tm100;
array a11t1{100};
array a21t1{100};

   Do k=1 to n1;
     if T1i{k}<= result*t1 then do;
     suma01=0;
     suma02=0;
     do j=1 to n1;
     sumy01=0;
     sumy02=0;

     if T1i{j}<=T1i{k} then DO;
          do i=1 to n1;
            if T1i{i}>=T1i{j} then y01=1;
            else if T1i{i}<T1i{j} then y01=0;
            sumy01=sumy01+y01;
          end;

          do ii=1 to m2;
            if T2i{ii}>=T1i{j} then y02=1;
            else if T2i{ii}<T1i{j} then y02=0;
            sumy02=sumy02+y02;
          end;
          if T1i{j}<= t2 then y022=(n2-m2)*1;
          else if T1i{j}>t2 then y022=(n2-m2)*0;
          sumyy02=sumy02+y022;
          a01=1/(sumy01+sumyy02);
          suma01=suma01+a01;

                                                          END;

     if T2i{j}<T1i{k} and (T2i{j}^=.) then DO;

     sumy11=0;
     sumy12=0;
```

```
  do i=1 to n1;

     if (T2i{j}^=.) and (T1i{i}>=T2i{j}) then y11=1;
     else if (T1i{i}<T2i{j}) and (T2i{j}^=.) then y11=0;
     sumy11=sumy11+y11;
  end;
          do ii=1 to m2;
               if T2i{ii}>=T2i{j} then y12=1;
               else if T2i{ii}<T2i{j} then y12=0;
               sumy12=sumy12+y12;
          end;

     if (T2i{j}<=t2) and (T2i{j}^=.) then y122=(n2-m2)*1;
     else if (T2i{j}>t2) and (T2i{j}^=.) then y122=(n2-m2)*0;
     sumyy12=sumy12+y122;
     a02=1/(sumy11+sumyy12);
          end;
                                                   END;


 a11t1(k)=suma01+suma02;

 if (a11t1{k}^=.) then s11=exp(-a11t1{k});
 if (a11t1{k}=0) then s11=exp(-a11t1{k});
end;

else if T1i{k}>result*t1 then do;
suma3=0;
suma4=0;

do j=1 to n1;
     sumy31=0;
     sumy32=0;
     sumy41=0;
     sumy42=0;
     if (result*t1<T1i{j}) and (T1i{j}<=T1i{k}) then do;
          do h=1 to n1;
            if T1i{h}>=(T1i{j}) then y31=1;
            else if T1i{h}<(T1i{j}) then y31=0;
            sumy31=sumy31+y31;
          end;
          do hh=1 to m2;
            if T2i{hh}>=(t1+result*(T1i{j}-t1)) then y32=1;
            else if T2i{hh}<(t1+result*(T1i{j}-t1)) then
     y32=0;
            sumy32=sumy32+y32;
          end;
```

```
 if (t1+result*(T1i{j}-t1))<=t2 then y321=(n2-m2)*1;
 else if (t1+result*(T1i{j}-t1)) then y321=(n2-m2)*0;
 sumyy32=sumy32+y321;
 a3=1/(sumy31+sumyy32);
 suma3=suma3+a3;

end;

if (result*t1<T2i{j}) and (T2i{j}<=T1i{k}) and (T2i{j}^=.)
then DO;
     do i=1 to n1;
       if T1i{i}>=(t1+(T2i{j}-t1)/result) and (T2i{j}^=.)
then y41=1;
       else if T1i{i}<(t1+(T2i{j}-t1)/result) and (T2i{j}^=.)
then y41=0;
       sumy41=sumy41+y41;
     end;
          do ii=1 to m2;
            if T2i{ii}>=T2i{j} then y42=1;
            else if  T2i{ii}< T2i{j} then y42=0;
            sumy42=sumy42+y42;
          end;
if (T2i{j}<=t2) and (T2i{j}^=.) then y421=(n2-m2)*1;
else if (T2i{j}>t2) and (T2i{j}^=.) then y421=(n2-m2)*0;
sumyy42=sumy42+y421;
a4=1/(sumy41+sumyy42);
suma4=suma4+a4;
                                             END;
end;
     a21t1(k)=suma3+suma4;
     if (a21t1{k}^=.) then s11=exp(-a21t1{k});
     if (a21t1{k}=0) then s11=exp(-a21t1{k});
end;
output;
END;
run;

data At2;
set R(keep=t1 t2 n1 n2 m2 result);
set Transp1;
set Transp2;
array T1i{100} Tn1-Tn100;
array T2i{100} Tm1-Tm100;
array a12t2{100};
array a22t2{100};

DO k=1 to n1;
```

```
    if T2i{k}<= result*t1  then do;
         suma01=0;
         suma02=0;
           do j=1 to n1;
              sumy01=0;
              sumy02=0;
              if T1i{j}<=T2i{k} then do;
                 do i=1 to n1;
                   if T1i{i}>=T1i{j} then y01=1;
                   else if T1i{i}<T1i{j} then y01=0;
                   sumy01=sumy01+y01;
                 end;

                   do ii=1 to m2;
                        if T2i{ii}>=T1i{j} then y02=1;
                        else if T2i{ii}<T1i{j} then y02=0;
                        sumy02=sumy02+y02;
                   end;

                   if T1i{j}<= t2 then y022=(n2-m2)*1;
                   else if T1i{j}>t2 then y022=(n2-m2)*0;
                         sumyy02=sumy02+y022;
                         a01=1/(sumy01+sumyy02);
                         suma01=suma01+a01;

                                  end;

    if T2i{j}<T2i{k} and (T2i{j}^=.) then DO;
         sumy11=0;
         sumy12=0;
       do i=1 to n1;
         if (T2i{j}^=.) and (T1i{i}>=T2i{j}) then y11=1;
         else if (T1i{i}<T2i{j}) and (T2i{j}^=.) then y11=0;
         sumy11=sumy11+y11;
       end;
              do ii=1 to m2;
                   if T2i{ii}>=T2i{j} then y12=1;
                   else if T2i{ii}<T2i{j} then y12=0;
                   sumy12=sumy12+y12;
              end;

    if (T2i{j}<=t2) and (T2i{j}^=.) then y122=(n2-m2)*1;
    else if (T2i{j}>t2) and (T2i{j}^=.) then y122=(n2-m2)*0;
    sumyy12=sumy12+y122;
    a02=1/(sumy11+sumyy12);
                                         END;
END;
```

```
   a12t2(k)=suma01+suma02;
   if (a12t2{k}^=.) then s22=exp(-a12t2{k});
   if (a12t2{k}=0) then s22=exp(-a12t2{k});

end;

else if T1i{k}>result*t1 then do;
     suma3=0;
     suma4=0;
          do j=1 to n1;
            sumy31=0;
            sumy32=0;
            sumy41=0;
            sumy42=0;
          if (result*t1<T1i{j}) and (T1i{j}<=T2i{k}) then DO;

                     do h=1 to n1;
                         if T1i{h}>=(T1i{j}) then y31=1;
                         else if T1i{h}<(T1i{j}) then y31=0;
                         sumy31=sumy31+y31;
                     end;

         do hh=1 to m2;
          if T2i{hh}>=(t1+result*(T1i{j}-t1)) then y32=1;
          else if T2i{hh}<(t1+result*(T1i{j}-t1))then y32=0;
          sumy32=sumy32+y32;
         end;

       if (t1+result*(T1i{j}-t1))<=t2 then y321=(n2-m2)*1;
       else if (t1+result*(T1i{j}-t1)) then y321=(n2-m2)*0;
       sumyy32=sumy32+y321;
       a3=1/(sumy31+sumyy32);
       suma3=suma3+a3;
                                                    END;
if (result*t1<T2i{j}) and (T2i{j}<=T2i{k}) and (T2i{j}^=.)
then DO;

  do i=1 to n1;
    if T1i{i}>=(t1+(T2i{j}-t1)/result) and (T2i{j}^=.) then
y41=1;
    else if T1i{i}<(t1+(T2i{j}-t1)/result) and (T2i{j}^=.)
then y41=0;
    sumy41=sumy41+y41;
  end;
```

```
                     do ii=1 to m2;
                       if T2i{ii}>=T2i{j} then y42=1;
                       else if T2i{ii}< T2i{j} then y42=0;
                       sumy42=sumy42+y42;
                     end;

     if (T2i{j}<=t2) and (T2i{j}^=.) then y421=(n2-m2)*1;
     else if (T2i{j}>t2) and (T2i{j}^=.) then y421=(n2-m2)*0;
              sumyy42=sumy42+y421;
              a4=1/(sumy41+sumyy42);
              suma4=suma4+a4;

END;
end;

  a22t2(k)=suma3+suma4;
  if a22t2{k}^=. then s22=exp(-a22t2{k});
  if a22t2{k}=0 then s22=exp(-a22t2{k});
end;
output;
END;
run;

data Visa1;
 set At1(keep=s11);
 set T1;
run;

data Visa2;
 set At2(keep=s22);
 set T2;
run;

data Visa11;
 set Visa1;
 Tij=s1;
  pag=1; keep Tij pag;
run;

data visa22;
 set visa2;
 Tij=s2;
 pag=2; keep Tij pag;
run;
```

```
data duom;    / *antra programa */
 set visa11 visa22;
run;

proc nlp data=duom;
max logl;
parms niu=2, teta =1128, r=5;
bounds niu>0, teta>0, r>0;
t1=80;
t2=500;
n1=100;
n2=100;
s=(n1*log(r)-(n2-&makro)*(((r*t1+t2-
t1)/teta)**niu))/(n1+&makro);

if pag=1 then s+log(niu)-niu*log(teta)+(niu-1)*log(r*Tij)-
((r*Tij/teta)**niu);
if pag=2 and (Tij<t1)
then s+log(r)+log(niu)-niu*log(teta)+(niu-1)*log(r*Tij)-
((r*Tij/teta)**niu);

if pag=2 and (Tij>t1)and (Tij<=t2)
then s+log(niu)-niu*log(teta)+(niu-1)*log(r*t1+Tij-t1)-
(((r*t1+Tij-t1)/teta)**niu);
logl=s;

run;
```

Priedas2.

Šiame priede pateiksiu gautus rezultatus, atlikus programas „Programa“ ir „Duom“.

Šioje lentelėje suvedžiau visus turimus pradinius duomenis. Nuadodamasi formulėmis 3.1 apskaičiavau ET0 ir ET1.

Programa									
t1	t2	n1	n2	r	r0	r1	ET0	ET1	niu
80	500	100	100	5	0,1	100	1000	200	2

Naudodamasi formulėmis 3.2 apskaičiavau gana, teta1ir teta2.

Teta					
ET0	ET1	niu	gama	teta1	teta2
1000	200	2	0,8862	1128,3792	225,6758

Apskaičiuoju w0.

W0								
t1	r	ET0	ET1	Niu	gama	teta1	teta2	w0
80	5	1000	200	2	0,8862	1128,3792	225,6758	0,1181

Toliau modeliuoju atsitiktinius dydžius w1 ir w2 ir skaičiuojam ,..., ir ,..., . Šiuos skaičiavimus atlikau „S“ lentelėje. Gaunu tokią lentelę:

S				
T	w1	s1	w2	s2
1	0,1795	100,3824	0,5063	628,0065
2	0,6645	235,8465	0,948	1620,2464
3	0,6416	228,612	0,6811	886,2662
4	0,5766	209,2068	0,6421	823,7601
5	0,7599	269,5392	0,3831	464,255
6	0,2698	126,5519	0,8289	1179,4152
7	0,0514	51,8534	0,1334	106,9621
8	0,3423	146,0925	0,7016	920,8663
9	0,5673	206,5671	0,0876	68,3398
10	0,3436	146,428	0,7214	955,6341
11	0,7593	269,3407	0,9875	2041,5406
12	0,1401	87,6688	0,3049	360,4469
13	0,901	343,2077	0,5226	650,3348

S				
T	w1	s1	w2	s2
14	0,8428	306,955	0,9757	1855,7398
15	0,9614	407,1727	0,641	822,1114
16	0,2152	111,0891	0,7829	1074,5436
17	0,3916	159,0735	0,3463	415,6774
18	0,9889	478,8258	0,245	278,1812
19	0,5227	194,0812	0,5407	675,3171
20	0,9823	453,4071	0,0016	9,0371
21	0,2328	116,1785	0,3086	365,4439
22	0,461	177,4088	0,7199	952,9659
23	0,0541	53,2165	0,5081	630,3789
24	0,0326	41,097	0,4398	539,0006
25	0,522	193,8869	0,5176	643,3793
26	0,5934	214,0721	0,7419	993,1112

S				
T	w1	s1	w2	s2
27	0,6154	220,6075	0,0588	55,5727
28	0,8958	339,3771	0,3095	366,6221
29	0,3636	151,7084	0,7195	952,2552
30	0,0283	38,2462	0,7928	1095,6904
31	0,0576	54,962	0,7409	991,309
32	0,297	133,9723	0,551	689,7103
33	0,7671	272,4189	0,1146	78,7181
34	0,8308	300,8241	0,1277	97,0575
35	0,3839	157,0472	0,1065	75,7383
36	0,7988	285,7784	0,3534	425,0305
37	0,2632	124,71	0,6601	852,1314
38	0,6187	221,5925	0,6891	899,6731
39	0,2226	113,2393	0,0154	28,0723
40	0,3636	151,7173	0,3238	385,8499
41	0,8394	305,1986	0,316	375,3392
42	0,448	173,9748	0,1317	104,0238
43	0,5656	206,0783	0,1273	96,3354
44	0,92	358,6364	0,9029	1403,2592
45	0,8072	289,5572	0,4205	513,4827
46	0,2136	110,6221	0,0662	59,0765
47	0,3676	152,7647	0,4957	613,6821
48	0,6992	247,3667	0,4185	510,8293
49	0,1113	77,5389	0,2042	219,3364
50	0,4196	166,4484	0,3472	416,8529
51	0,5149	191,9349	0,6787	882,368
52	0,0289	38,6485	0,9417	1582,2174
53	0,7934	283,3976	0,9504	1635,7212
54	0,1476	90,1897	0,1322	104,814
55	0,5656	206,0703	0,3746	453,0055
56	0,7717	274,2731	0,275	319,8316
57	0,0125	25,3538	0,8286	1178,5465
58	0,8876	333,6573	0,3804	460,7492
59	0,8816	329,6191	0,2398	270,8511
60	0,1901	103,637	0,5622	705,4493
61	0,2505	121,1728	0,9084	1424,4487
62	0,722	255,3478	0,3506	421,372
63	0,4625	177,8057	0,8095	1133,0672
64	0,8932	337,534	0,0696	60,6232
65	0,1565	93,0973	0,6925	905,3137
66	0,9075	348,1834	0,2555	292,9498
67	0,3389	145,188	0,7047	926,2332
68	0,3703	153,4846	0,0552	53,7669
69	0,9993	608,1965	0,1504	135,5589
70	0,8622	317,6877	0,6803	884,9781
71	0,0274	37,6007	0,3242	386,3724
72	0,2494	120,8703	0,9306	1523,2349
73	0,4601	177,1843	0,712	938,9892
74	0,2412	118,5774	0,2477	282,0385
75	0,0821	66,0605	0,4343	531,7299
76	0,8989	341,6129	0,007	18,9454
77	0,9576	401,2808	0,6612	853,8608
78	0,4999	187,8523	0,9447	1599,6484
79	0,5865	212,069	0,1737	172,9448
80	0,1521	91,6755	0,2421	274,0946
81	0,7886	281,3242	0,997	2401,1694
82	0,3892	158,4428	0,0219	33,553
83	0,3162	139,1391	0,647	831,3906
84	0,0558	54,0538	0,1109	77,3658
85	0,8968	340,0654	0,4099	499,5018
86	0,9204	359,0347	0,5793	729,9173
87	0,48	182,4814	0,446	547,1785
88	0,1578	93,5147	0,9535	1656,5807
89	0,7051	249,3827	0,4304	526,5535
90	0,8721	323,6481	0,0722	61,7675
91	0,8252	298,0307	0,9361	1551,2891
92	0,5129	191,41	0,5926	749,2625
93	0,6175	221,2415	0,1589	149,3785
94	0,8099	290,7783	0,6929	906,0011
95	0,0656	58,781	0,4407	540,0931
96	0,8861	332,6356	0,0756	63,2605
97	0,2779	128,7607	0,5852	738,5298
98	0,4389	171,5568	0,018	30,444
99	0,4626	177,84	0,5721	719,67
100	0,0601	56,188	0,0764	63,61

Čia w1 yra sumodeliuoti atsitiktiniai dydžiai, pasiskirstę pagal tolygųjį skirtinį intervale (0;1). Jie naudojami ,..., skaičiavimui.

,..., SAS programoje yra pažymėti s1. Ši lentelė rodo rezultatus po kiekvienos ciklo iteracijos. Tą daro komanda output. Jei cikle nepanaudosime komandos output, tai į lentelę bus įrašytas tik paskutinės ciklo iteracijos rezultatas. Mum reikia visų ,..., reikšmių, tosėl ši lentelė išėjo tokia didelė. Analogiškai, w2 yra sumodeliuoti atsitiktiniai dydžiai, pasiskirstę pagal tolygųjį skirstinį intervale (0;1), naudojami ,..., (SAS programoje šios funkcijos žymimos s2) skaičiavimui.

Toliau reikia s1 ir s2 išrūšiuoti didėjančia tvarka, kad gautume gedimo momentus ir ,($m_2 \leq n_2$). Tai atliekam su procedūra „proc sort“. Išrūšiuotus duomenis surašome į dvi lenteles: „T1“ ir „T2“.
„T1“ lentelėje yra pirmoji gedimo momentų imtis (išrūšiuoti s1). „T2“ lentelėje – antroji gedimo momentų imtis (išrūšiuoti s2). Šios lentelės atrodo taip:

T1
S1
25,3538
37,6007
38,2462
38,6485
41,097
51,8534
53,2165
54,0538
54,962
56,188
58,781
66,0605
77,5389
87,6688
90,1897
91,6755
93,0973
93,5147
100,3824
103,637
110,6221
111,0891
113,2393
116,1785
118,5774
120,8703

T1
S1
121,1728
124,71
126,5519
128,7607
133,9723
139,1391
145,188
146,0925
146,428
151,7084
151,7173
152,7647
153,4846
157,0472
158,4428
159,0735
166,4484
171,5568
173,9748
177,1843
177,4088
177,8057
177,84
182,4814
187,8523
191,41

T1
S1
191,9349
193,8869
194,0812
206,0703
206,0783
206,5671
209,2068
212,069
214,0721
220,6075
221,2415
221,5925
228,612
235,8465
247,3667
249,3827
255,3478
269,3407
269,5392
272,4189
274,2731
281,3242
283,3976
285,7784
289,5572
290,7783

T1
S1
298,0307
300,8241
305,1986
306,955
317,6877
323,6481
329,6191
332,6356
333,6573
337,534
339,3771
340,0654
341,6129
343,2077
348,1834
358,6364
359,0347
401,2808
407,1727
453,4071
478,8258
608,1965

Šiuo atveju T1 yra 100 vienetų, o T2 tik 45, nes T2 turi būti mažiau nei eksperimento pabaiga. T.y mažiau nei t2=500.

T2
s2
9,0371
18,9454
28,0723
30,444
33,553
53,7669
55,5727
59,0765
60,6232
61,7675
63,2605
63,61
68,3398
75,7383
77,3658
78,7181
96,3354
97,0575
104,0238
104,814
106,9621
135,5589
149,3785

T2
s2
172,9448
219,3364
270,8511
274,0946
278,1812
282,0385
292,9498
319,8316
360,4469
365,4439
366,6221
375,3392
385,8499
386,3724
415,6774
416,8529
421,372
425,0305
453,0055
460,7492
464,255
499,5018

Kadangi skaičiuojant reikia naudoti abi gedimo momentų imtis, tai patogiau būtų jas turėti vienoje lentelėje. Tai darome „R“ lentelėje. Toje pačioje lentelėje skaičiuojame ir randame . „R“ lentelė atrodo taip:

R													
t1	t2	n1	n2	r	r0	r1	ET0	ET1	niu	gama	teta1	teta2	w0
80	500	100	100	5	0,1	100	1000	200	2	0,8862	1128,3792	225,6758	0,1181

R									
Tn1	Tn2	Tn3	Tn4	Tn5	Tn6	Tn7	Tn8	Tn9	Tn10
25,3538	37,6007	38,2462	38,6485	41,097	51,8534	53,2165	54,0538	54,962	56,188

R									
Tn11	Tn12	Tn13	Tn14	Tn15	Tn16	Tn17	Tn18	Tn19	Tn20
58,781	66,0605	77,5389	87,6688	90,1897	91,6755	93,0973	93,5147	100,3824	103,637

R									
Tn21	Tn22	Tn23	Tn24	Tn25	Tn26	Tn27	Tn28	Tn29	Tn30
110,6221	111,0891	113,2393	116,1785	118,5774	120,8703	121,1728	124,71	126,5519	128,7607

R									
Tn31	Tn32	Tn33	Tn34	Tn35	Tn36	Tn37	Tn38	Tn39	Tn40
133,9723	139,1391	145,188	146,0925	146,428	151,7084	151,7173	152,7647	153,4846	157,0472

R									
Tn41	Tn42	Tn43	Tn44	Tn45	Tn46	Tn47	Tn48	Tn49	Tn50
158,4428	159,0735	166,4484	171,5568	173,9748	177,1843	177,4088	177,8057	177,84	182,4814

R									
Tn51	Tn52	Tn53	Tn54	Tn55	Tn56	Tn57	Tn58	Tn59	Tn60
187,8523	191,41	191,9349	193,8869	194,0812	206,0703	206,0783	206,5671	209,2068	212,069

R									
Tn61	Tn62	Tn63	Tn64	Tn65	Tn66	Tn67	Tn68	Tn69	Tn70
214,0721	220,6075	221,2415	221,5925	228,612	235,8465	247,3667	249,3827	255,3478	269,3407

R									
Tn71	Tn72	Tn73	Tn74	Tn75	Tn76	Tn77	Tn78	Tn79	Tn80
269,5392	272,4189	274,2731	281,3242	283,3976	285,7784	289,5572	290,7783	298,0307	300,8241

R									
Tn81	Tn82	Tn83	Tn84	Tn85	Tn86	Tn87	Tn88	Tn89	Tn90
305,1986	306,955	317,6877	323,6481	329,6191	332,6356	333,6573	337,534	339,3771	340,0654

R									
Tn91	Tn92	Tn93	Tn94	Tn95	Tn96	Tn97	Tn98	Tn99	Tn100
341,619	343,2077	348,1834	358,6364	359,0347	401,2808	407,1727	453,4071	478,8258	608,1965

R									
Tm1	Tm2	Tm3	Tm4	Tm5	Tm6	Tm7	Tm8	Tm9	Tm10
9,0371	18,9454	28,0723	30,444	33,553	53,7669	55,5727	59,0765	60,6232	61,7675

R									
Tm11	Tm12	Tm13	Tm14	Tm15	Tm16	Tm17	Tm18	Tm19	Tm20
63,2605	63,61	68,3398	75,7383	77,3658	78,7181	96,3354	97,0575	104,0238	104,814

R									
Tm21	Tm22	Tm23	Tm24	Tm25	Tm26	Tm27	Tm28	Tm29	Tm30
106,961	135,5589	149,3785	172,9448	219,3364	270,8511	274,0946	278,1812	282,0385	292,9498

R									
Tm31	Tm32	Tm33	Tm34	Tm35	Tm36	Tm37	Tm38	Tm39	Tm40
319,8316	360,4469	365,4439	366,6221	375,3392	385,8499	386,3724	415,6774	416,8529	421,372

R									
Tm41	Tm42	Tm43	Tm44	Tm45	Tm46	Tm47	Tm48	Tm49	Tm50
425,0305	453,0055	460,7492	464,255	499,5018					

……………………………………………………………………………..
………………………………………………………………………………
………………………………………………………………………………

R			
Tm97	Tm98	Tm99	Tm100

R	
U	result
-0,0134	4,6944

„Visa1“ ir „Visa2“ lentelėse yra tik du stulpeliai: [ ], [ ] bei [ ], [ ] atitinkamai

Visa1	
s11	S1
0,995	25,3538
0,9898	37,6007
0,9847	38,2462
0,9796	38,6485
0,9745	41,097
0,9694	51,8534
0,9643	53,2165
0,9591	54,0538
0,954	54,962
0,9488	56,188
0,9436	58,781
0,9383	66,0605
0,9329	77,5389
0,9275	87,6688
0,922	90,1897
0,9166	91,6755
0,9112	93,0973
0,9057	93,5147
0,9002	100,3824
0,8947	103,637
0,8891	110,6221
0,8835	111,0891
0,8779	113,2393
0,8723	116,1785
0,8667	118,5774

Visa1	
s11	S1
0,8611	120,8703
0,8554	121,1728
0,8498	124,71
0,8442	126,5519
0,8386	128,7607
0,833	133,9723
0,8274	139,1391
0,8217	145,188
0,8161	146,0925
0,8104	146,428
0,8047	151,7084
0,799	151,7173
0,7934	152,7647
0,7877	153,4846
0,782	157,0472
0,7763	158,4428
0,7706	159,0735
0,7649	166,4484
0,7592	171,5568
0,7535	173,9748
0,7478	177,1843
0,742	177,4088
0,7363	177,8057
0,7306	177,84
0,7249	182,4814

Visa1	
s11	S1
0,7191	187,8523
0,7134	191,41
0,7077	191,9349
0,7019	193,8869
0,6962	194,0812
0,6905	206,0703
0,6847	206,0783
0,679	206,5671
0,6733	209,2068
0,6676	212,069
0,6618	214,0721
0,656	220,6075
0,6503	221,2415
0,6445	221,5925
0,6387	228,612
0,6329	235,8465
0,6271	247,3667
0,6214	249,3827
0,6156	255,3478
0,6098	269,3407
0,604	269,5392
0,5982	272,4189
0,5923	274,2731
0,5863	281,3242
0,5803	283,3976

Visa1	
s11	S1
0,5743	285,7784
0,5683	289,5572
0,5623	290,7783
0,5562	298,0307
0,5501	300,8241
0,544	305,1986
0,538	306,955
0,5319	317,6877
0,5257	323,6481
0,5196	329,6191
0,5134	332,6356
0,5073	333,6573
0,5011	337,534
0,495	339,3771
0,4888	340,0654
0,4827	341,6129
0,4765	343,2077
0,4704	348,1834
0,4643	358,6364
0,4581	359,0347
0,8064	401,2808
0,628	407,1727
0,4321	453,4071
0,2576	478,8258
0,0939	608,1965

Visa2	
s22	S2
1	9,0371
1	18,9454
0,995	28,0723
0,995	30,444
0,995	33,553
0,9643	53,7669
0,954	55,5727
0,9436	59,0765
0,9436	60,6232
0,9436	61,7675
0,9436	63,2605
0,9436	63,61

Visa2	
s22	S2
0,9383	68,3398
0,9383	75,7383
0,9383	77,3658
0,9329	78,7181
0,9057	96,3354
0,9057	97,0575
0,8947	104,0238
0,8947	104,814
0,8947	106,9621
0,833	135,5589
0,8104	149,3785
0,7592	172,9448

Visa2	
s22	S2
0,6618	219,3364
0,604	270,8511
0,5982	274,0946
0,5923	278,1812
0,5863	282,0385
0,5623	292,9498
0,5319	319,8316
0,4581	360,4469
0,4581	365,4439
0,4581	366,6221
0,4581	375,3392
0,4581	385,8499

Visa2	
s22	S2
0,4581	386,3724
0,4581	415,6774
0,4581	416,8529
0,4581	421,372
0,4581	425,0305
0,4581	453,0055
0,4581	460,7492
0,4581	464,255
0,4581	499,5018

Lentelės“Visa” s11 pažymėti išgyvenamumo funkcijų įverčiai , o s1 – pirmos imties gedimo momentai . Atitinkami žymėjimai įvesti ir “Visa2” lentelėje. Dabar sujungiu šias abi lenteles į vieną, išrūšiuoju išgyvenamumo funkcijos įverčius mažėjančia tvarka, o gedimo momentus T_{ij} didėjančia tvarka.

visa	
s11	s1
1	18,9454
1	9,0371
0,995	25,3538
0,995	33,553
0,995	30,444
0,995	28,0723
0,9898	37,6007
0,9847	38,2462
0,9796	38,6485
0,9745	41,097
0,9694	51,8534
0,9643	53,7669
0,9643	53,2165
0,9591	54,0538
0,954	55,5727
0,954	54,962
0,9488	56,188
0,9436	59,0765
0,9436	63,2605
0,9436	63,61
0,9436	60,6232
0,9436	58,781
0,9436	61,7675
0,9383	66,0605
0,9383	77,3658
0,9383	75,7383
0,9383	68,3398
0,9329	77,5389
0,9329	78,7181
0,9275	87,6688
0,922	90,1897
0,9166	91,6755
0,9112	93,0973
0,9057	93,5147
0,9057	96,3354
0,9057	97,0575
0,9002	100,3824

visa	
s11	s1
0,8947	104,0238
0,8947	104,814
0,8947	106,9621
0,8947	103,637
0,8891	110,6221
0,8835	111,0891
0,8779	113,2393
0,8723	116,1785
0,8667	118,5774
0,8611	120,8703
0,8554	121,1728
0,8498	124,71
0,8442	126,5519
0,8386	128,7607
0,833	133,9723
0,833	135,5589
0,8274	139,1391
0,8217	145,188
0,8161	146,0925
0,8104	149,3785
0,8104	146,428
0,8064	401,2808
0,8047	151,7084
0,799	151,7173
0,7934	152,7647
0,7877	153,4846
0,782	157,0472
0,7763	158,4428
0,7706	159,0735
0,7649	166,4484
0,7592	171,5568
0,7592	172,9448
0,7535	173,9748
0,7478	177,1843
0,742	177,4088
0,7363	177,8057
0,7306	177,84

visa	
s11	s1
0,7249	182,4814
0,7191	187,8523
0,7134	191,41
0,7077	191,9349
0,7019	193,8869
0,6962	194,0812
0,6905	206,0703
0,6847	206,0783
0,679	206,5671
0,6733	209,2068
0,6676	212,069
0,6618	214,0721
0,6618	219,3364
0,656	220,6075
0,6503	221,2415
0,6445	221,5925
0,6387	228,612
0,6329	235,8465
0,628	407,1727
0,6271	247,3667
0,6214	249,3827
0,6156	255,3478
0,6098	269,3407
0,604	269,5392
0,604	270,8511
0,5982	274,0946
0,5982	272,4189
0,5923	278,1812
0,5923	274,2731
0,5863	281,3242
0,5863	282,0385
0,5803	283,3976
0,5743	285,7784
0,5683	289,5572
0,5623	290,7783
0,5623	292,9498
0,5562	298,0307

visa	
s11	s1
0,5501	300,8241
0,544	305,1986
0,538	306,955
0,5319	317,6877
0,5319	319,8316
0,5257	323,6481
0,5196	329,6191
0,5134	332,6356
0,5073	333,6573
0,5011	337,534
0,495	339,3771
0,4888	340,0654
0,4827	341,6129
0,4765	343,2077
0,4704	348,1834
0,4643	358,6364
0,4581	460,7492
0,4581	453,0055
0,4581	425,0305
0,4581	386,3724
0,4581	464,255
0,4581	421,372
0,4581	416,8529
0,4581	415,6774
0,4581	385,8499
0,4581	375,3392
0,4581	366,6221
0,4581	365,4439
0,4581	360,4469
0,4581	359,0347
0,4581	499,5018
0,4321	453,4071
0,2576	478,8258
0,0939	608,1965

Iš šių rezultatų nubraižome patikimumo funkcijos įverčio grafiką

Į šią lentelę patalpinau gedimo momentus T1 ir T2 ir pagalbinį kintamajį 1ir 2 atitinkamai. Tokie pažymėjimai bus reikalingi tolesniuose apskaičiavimuose.

Duom	
Tij	pag
25,3538	1
37,6007	1
38,2462	1
38,6485	1
41,097	1
51,8534	1
53,2165	1
54,0538	1
54,962	1
56,188	1
58,781	1
66,0605	1
77,5389	1

Duom	
Tij	pag
87,6688	1
90,1897	1
91,6755	1
93,0973	1
93,5147	1
100,3824	1
103,637	1
110,6221	1
111,0891	1
113,2393	1
116,1785	1
118,5774	1
120,8703	1

Duom	
Tij	pag
121,1728	1
124,71	1
126,5519	1
128,7607	1
133,9723	1
139,1391	1
145,188	1
146,0925	1
146,428	1
151,7084	1
151,7173	1
152,7647	1
153,4846	1

Duom	
Tij	pag
157,0472	1
158,4428	1
159,0735	1
166,4484	1
171,5568	1
173,9748	1
177,1843	1
177,4088	1
177,8057	1
177,84	1
182,4814	1
187,8523	1
191,41	1

Duom	
Tij	pag
191,9349	1
193,8869	1
194,0812	1
206,0703	1
206,0783	1
206,5671	1
209,2068	1
212,069	1
214,0721	1
220,6075	1
221,2415	1
221,5925	1
228,612	1

Duom	
Tij	pag
235,8465	1
247,3667	1
249,3827	1
255,3478	1
269,3407	1
269,5392	1
272,4189	1
274,2731	1
281,3242	1
283,3976	1
285,7784	1
289,5572	1
290,7783	1
298,0307	1
300,8241	1
305,1986	1

Duom	
Tij	pag
306,955	1
317,6877	1
323,6481	1
329,6191	1
332,6356	1
333,6573	1
337,534	1
339,3771	1
340,0654	1
341,6129	1
343,2077	1
348,1834	1
358,6364	1
359,0347	1
401,2808	1
407,1727	1

Duom	
Tij	pag
453,4071	1
478,8258	1
608,1965	1
9,0371	2
18,9454	2
28,0723	2
30,444	2
33,553	2
53,7669	2
55,5727	2
59,0765	2
60,6232	2
61,7675	2
63,2605	2
63,61	2
68,3398	2

Duom	
Tij	pag
75,7383	2
77,3658	2
78,7181	2
96,3354	2
97,0575	2
104,0238	2
104,814	2
106,9621	2
135,5589	2
149,3785	2
172,9448	2
219,3364	2
270,8511	2
274,0946	2
278,1812	2
282,0385	2

Duom	
Tij	pag
292,9498	2
319,8316	2
360,4469	2
365,4439	2
366,6221	2
375,3392	2
385,8499	2
386,3724	2
415,6774	2
416,8529	2
421,372	2
425,0305	2
453,0055	2
460,7492	2
464,255	2
499,5018	2

Atlikę funkcijos maksimizavimą, kuris apskaičiuoja įverčius. Gauname tokius rezultatus:

```
              PROC NLP: Nonlinear Maximization

                   Optimization Results
                    Parameter Estimates
                                        Gradient
                                       Objective
N Parameter        Estimate         Function

1 niu               1.853302    -0.000000111
2 teta           1103.535354    -3.03485E-10
3 r                 4.919465     0.000000147
```

Kai pradiniai duomenys buvo tokie:

PROC NLP: Nonlinear Maximization

Optimization Start
Parameter Estimates

N	Parameter	Estimate	Gradient Objective Function	Lower Bound Constraint	Upper Bound Constraint
1	niu	2.000000	-10.436012	0	.
2	teta	1128.000000	-0.001481	0	.
3	r	5.000000	0.010826	0	.

Iš gautų rezultatų nubraižome patikimumo funkcijos įverčio grafiką: