UNIVERSITETI I TIRANËS

FAKULTETI I SHKENCAVE TË NATYRËS

DEPARTAMENTI I INFORMATIKËS

DISERTACION

PËR MARRJEN E GRADËS "DOKTOR I SHKENCAVE"

DENIS SAATÇIU

KRIJIMI I MODELEVE INFORMATIKË PËR VLERËSIMIN E DENSITETEVE PROBABILITARË ZBATIME NË EKONOMI

Udhëheqës Shkencor: Ph.D. Altin TANKU

Tiranë, 2014

UNIVERSITETI I TIRANËS

FAKULTETI I SHKENCAVE TË NATYRËS

DEPARTAMENTI I INFORMATIKËS

Disertacion

i

Paraqitur nga

Z._____

Për marrjen e gradës shkencore

DOKTOR

Specialiteti : _____

Tema: _____

_____

_____

Mbrohet më dt.____ .____ ._____ para jurisë:

1._____Kryetar

2._____Anëtar (oponent)

3._____ Anëtar oponent)

4._____Anëtar

5._____Anëtar

6._____Anëtar

PËRMBAJTJA

PËRMBAJTJA iii
LISTA E TABELAVE v
LISTA E FIGURAVE vi
Hyrje vii
1. Modelimi 1
1.2 Modelimi i informatik 1
1.3 Kërkimi shkencor në modelimet informatike 2
1.4 Modelet informatikë në ekonomi 3
1.5 Ekonometria si shkencë 3
2. Ekonomia si ngjarje sociale, analiza e densiteteve të probabiliteteve të përbashkëta 4
3. Organizimi i të dhënave dhe testimi i tyre 7
Materiali dhe metodat 7
Manipulimi i të dhënave 8
Kategoria e parë (testi i limiteve) 9
Kategoria e dytë (testet spacialë) [19] [20] 9
Testi nr.1 9
Testi nr.2 10
Testi dispersionit [21] 10
Testi i Mann-it [22] 12
4. Particionimi i të dhënave 14
Metodologjia 14
Eksperimenti 14
5. Modelimi informatik i metodës së vlerësimit të densiteteve 18
5.1 Rasti njëdimensional 18
5.1.1 Bërthama normale 20
5.1.2 Bërthama trekëndore 20
5.1.3 Bërthama drejtkëndore 21
5.1.4 Bërthama Epanechnikov 21
5.2 Rasti d-dimensional 22

5.2.1 Bërthama Normale (Gaussian) 25
5.2.2 Bërthama Epanechnikov 25
5.3 Rasti d-dimensional , me kusht m-dimensional 27
5.3.1 Bërthama Normale (Gaussian) 30
5.3.2 Bërthama Epanechnikov 31
5.4 Ndërtimi i algoritmeve për momentet 32
6. Paralelizimi 35
Thread 35
Paralelizimi në metodën e paraqitur 38
7. Implementimi i cloud computing 41
7.1 Map/Reduce 41
7.2 MapReduce në gjenerimin e vlerave të densiteteve 43
Funksioni map 44
Funksioni reduce 44
7.3 Vlerësimi financiar 44
Përfundime 46
Referenca 48
Aneks 1: Simulimi në PHP i modelit të vlerësimit të densiteteve probabilitare 50
Aneks 2: Ekzekutime të aplikimit mbi të dhënat e ekonomisë 54
Falënderime 62
Bibliografi 63
Publikime: 63
Prezantime: 63
Përmbledhje 64
Abstract 65

LISTA E TABELAVE

Tabelë	**Faqe**
Tabelë 4.1 Specifikimet e serverit	14
Tabelë 4.2 Ndarja e të dhënave në particione	16
Tabelë 4.3 Koha së ekzekutimit të kërkesave në tabelën e particionuar dhe të paparticionuar	16
Tabelë 5.1 Bërthama e densiteteve dhe efiçenca e tyre, rasti njëdimensional	18
Tabelë 5.2 Vlerësimi i densiteteve sipas bërthamave, rasti njëdimensional	18
Tabelë 5.3 Bërthama e densiteteve, rasti d-dimensional	22
Tabelë 6.1 Koha e ekzekutimit të algoritmit të paparalelizuar dhe të paralelizuar ku numri i thread sa numri i pikave	36
Tabelë 6.2 Koha së ekzekutimit të algoritmit të paparalelizuar dhe të paralelizuar.	37
Tabelë A2.1 Variablat ekonomikë	54

LISTA E FIGURAVE

Figura	**Faqe**
Figura 4.1: Grafiku i diferencës së ekzekutimit të kërkesave midis tabelës së particionuar dhe të paparticionuar	17
Figura 6.1 Grafiku i së kohës së ekzekutimit midis algoritmit të paparalelizuar dhe të paralelizuar ku numri i thread sa numri i pikave	37
Figura 6.2 Grafiku i diferencës së kohës së ekzekutimit midis algoritmit të paparalelizuar dhe të paralelizuar	38
Figura 7.1 Ndarja e punëve në Map/Reduce	41
Figura 7.2 Krijimi i listave rezultat prej listave hyrëse	42
Figura 7.3 Agregimi i vlerave për një çelës	42
Figura A1.1: Pamje e parë e simulimit	50
Figura A1.2: Format i të dhënave ne Excel	51
Figura A1.3: Vlerat e variablave	51
Figura A1.4: Rangu i variablave	52
Figura A1.5: Tipet e bërthamave	52
Figura A1.6: Tipet e rrotullimit	52
Figura A2.1: Paraqitja surface RM2 vs CPI	55
Figura A2.2: Paraqitja contour RM2 vs CPI	55
Figura A2.3: Simultaneous	56
Figura A2.4: Lag 1	56
Figura A2.5: Lag 2	56
Figura A2.6: Lag 3	56
Figura A2.7: Lag 4	56
Figura A2.8: Lag 6	56
Figura A2.9: Pritja matematike dhe devijimi standard e M2 për ndryshimet e CPI	57
Figura A2.10: Pritja matematike dhe devijimi standard e CPI për ndryshimet e M2	57
Figura A2.11: Simultaneous	58
Figura A2.12: Lag 1	58
Figura A2.13: Lag 2	58
Figura A2.14: Lag 3	58
Figura A2.15: Lag 4	58
Figura A2.16: Lag 6	58
Figura A2.17: Paraqitje surface	59
Figura A2.18: Prerje e tërthortë	69
Figura A2.19: Simultaneous	60
Figura A2.20: Lag 1	60
Figura A2.21: Lag 2	60
Figura A2.22: Lag 3	60
Figura A2.23: Lag 4	60
Figura A2.24: Lag 6	60
Figura A2.25: Paraqitja surface	61
Figura A2.26: Prerja e tërthortë	61

Hyrje

Ngjarjet ekonomike, ashtu dhe sikur fenomenet natyrore, janë ngjarje stokastike. Studimi i tyre bëhet përmes teorive të probabilitetit.

Për këtë qëllim, studiuesit e kanë të nevojshme modelimin e fenomeneve stokastike dhe simulimin e këtij modelimi në paketa software-ike me efiçencë dhe gabim të vogël .

Qëllimi i këtij kërkimi është që të simulojë modelet matematikore që përdoren për të shpjeguar ngjarjet e rastit dhe për t'i modeluar ato në të ardhmen dhe aplikimin e saj në fushën e ekonomisë.

Ky simulim do të jetë një vegël e rëndësishme për ekonomistët për analizën dhe parashikimin e fenomeneve ekonomike me eficencë të lartë.

Simulimi bazohet ne teorinë propozuar nga Tanku dhe Ceca [(1) (2)]. Kjo teori paraqet një vështrim të ekonomisë bazuar mbi vlerësimin e densiteteve të bashkuara me dhe pa kusht. Teoria është aplikuar për variabla ekonomike, por mund te zgjerohet për çdo fenomen social natyror nga të dhënat e të cilit mund të nxjerrim përfundime.

Në këto shkenca, bazat e të dhënave janë të mëdha. Numri i të dhënave është relativisht shumë i madh dhe ndërtimi i simulimeve jo të menduara mirë mund të çojë në dështimin e modelit të simuluar. Në këtë studim, do të paraqesim një sërë teknikash për rritjen e efiçencës, si në nxjerrjen e të dhënave, ashtu edhe në përpunimin e tyre. Teknikat kryesore janë paralelizimi i procedurave llogaritëse dhe particionimi i të dhënave hyrëse.

Teza përdor metodën e eksperimentit për të hetuar nëse aplikimi i këtyre metodave sjell rritjen e efiçencës në ekzekutimin e algoritmeve.

Institucionet akademike, veçanërisht në shkencat sociale nuk kane fuqi llogaritëse. Gjithashtu shpesh herë, fuqia llogaritëse e nevojitur mund të ketë një kosto të papërballueshme. Prandaj mund të jetë më e leverdishme që kjo fuqi llogaritëse të përdoret në ambiente cloud ku kostoja e implementimit është më e ulët dhe mundësia e zgjerueshmërisë është më e lehtë.

Teza konsideron gjithashtu kostot financiare për përdoruesit.

Metoda është e aplikuar mbi të dhëna makroekonomike, por është lehtësisht e adaptueshme për çdo fenomen stokastik, pavarësisht nga natyra e tij.

Vazhdimi i këtij studimi do te fokusohet si me poshte:

Kapitulli i parë flet për modelimin. Pse ai është i rëndësishëm për të paraqitur jetën reale. Jepet një përshkrim i modelimit informatik dhe tregohet pse ai është kërkim shkencor. Më pas shikohet evoluimi i i modelimit informatik në ekonomi dhe tentohet të jepet një lidhje midis tij dhe shkencës së ekonometrisë.

Kapitulli i dytë jep një përshkrim të teorisë alternative ekonomike të propozuar të bazuar mbi vlerësimin e densiteteve të bashkuara dhe të densiteteve me kusht. Trajtohen teoria e VAR dhe kufizimet e tij.

Kapitulli i tretë trajton të dhënat të cilat do të përpunohen. Jepet një alternativë për ruajtjen e vlerave. Në këtë kapitull fokusohemi në cilësinë e të dhënave të servirura. Teknika të ndryshme testimi markuese e korrigjuese aplikohen mbi të dhënat, për të vlerësuar cilësinë e tyre.

Kapitulli i katërt trajton particionimin e të dhënave, si një teknikë me rëndësi në përshpejtimin e nxjerrjes së informacionit. Propozohet një teknikë për particionimin e tabelave me seri të gjata të dhënash si dhe kryhen eksperimente praktikë mbi kohën e nxjerrjes së të dhënave.

Kapitulli i pestë është ndër më të rëndësishmit. Në të paraqiten algoritmet në pseudokod të formulave të aparatit matematikor të teorisë ekonomike bazuar mbi vlerësimin e densiteteve të bashkuara. Algoritmet ndërtohen mbi bërthama një dhe shumë dimensionale, si dhe në bërthama d-dimensionale me kusht m-dimensional.

Kapitulli i gjashtë trajton paralelizimin e procedurave të algoritmeve të ndërtuar në kapitullin gjashtë. Diskutohen të mirat e paralelizmit dhe përfitimet që sjell. Trajtohet teknika të paralelizimit në php, multithread.

Kapitulli i shtatë trajton teknologjinë cloud dhe mundësinë e vendosjes së programit në cloud. Diskutohen përfitimet e mundshme si dhe propozohet modeli për maksimizimin e fitimit duke optimizuar koston . Jepet propozimi për implementimin e algoritmeve në cloud.

Aneks një paraqet simulimin e ndërtuar. Jep një shpjegim të funksionaliteteve dhe trajton mënyrën e ndërtimit të tij.

Aneks dy jep ekzekutime të simulimit mbi të dhëna reale të ekonomisë. Jepet një shpjegim i variablave ekonomikë.

1. Modelimi

Modelimi përbën një pjesë kyçe të një aktiviteti shkencor. Disiplina të ndryshme shkencore e trajtojnë modelimin sipas veçorive përkatëse. [3] Qëllimi i tij është të paraqesë objekte empirike apo procese të ndryshme në mënyrë të qartë, pa devijuar nga logjika bazë e vetë objektit apo procesit. Të gjithë modelet janë një paraqitje e thjeshtuar e realitetit, *in simulacra*. [4] Shpesh herë është e vështirë të gjejmë një model të plotë dhe që e paraqet tërësisht realitetin, por ne përpiqemi gjithnjë e më tepër të ndërtojmë modele sa më të përshtatshëm, duke ngritur dhe debate shkencore se cili prej modeleve është më i afërt me realitetin. [5]
Griffiths (2010) [6]i ndan modelet në katër grupe kryesore, modelet *konceptualë, in vivo, in vitro* dhe *in silico.*
Modelet e konceptualë janë modelet më të thjeshta. Ato përdoren për të modeluar veprimet e thjeshta rutinë.
Modelet *in vivo* janë organizma të gjallë laboratorikë, të cilët përzgjidhen për ngjashmëritë e tyre në fusha specifike dhe përdoren për të arritur rezultate eksperimentale.
Modelet *in vitro* kanë përmbajtje të ndryshme nga objekti konkret, por kanë aftësi të ofrojnë shërbime të njëjta me të.
Modelet *in silico* përdorin simulimin e modeleve matematikorë. Këto modele shpesh janë të bazuar në sisteme ekuacionesh dhe ofrojnë zgjidhje të tyre në situata komplekse.
Sipas Frigg dhe Hartmann (2012) [7], modelet mund të ndërtohen për dy qëllime kryesore. Së pari, modeli mund të jetë një paraqitje e një dukurie. Këto modele ndahen në modele fenomenesh dhe modele të dhënash.Së dyti, modelet mund të jenë paraqitje e një teorie.
Sipas tyre, modelet janë makina për të mësuar rreth botës. Duke studiuar një model, ne mund të zbulojmë tipare dhe fakte për vetë sistemin. Pyetjes se si mund të mësojmë nga një model, Hughes (1997) [8] iu përgjigj me një mjedis të përgjithshëm për mësimin nga një model. Procesi kalon në tre faza, orientimi, demonstrimi dhe interpretimi. Fillohet duke bërë lidhjet midis modelit dhe sistemit. Kontrollohen tiparet e modelit për të demonstruar realizimet e kryera dhe më pas këto të fundit krahasohen me vetë sistemin.
Morgan(1999) [9] e lidh procesin e të mësuarit nga një model me dy momente kryesore, ndërtimi i modelit dhe më pas manipulimi i tij. Mësimi nga një model mund të vijë vetëm si rrjedhojë e përdorimit të modelit. Tipe të ndryshëm modelesh, kërkojnë metodologji të veçanta.

1.2 Modelimi i informatik

Modelet informatikë janë simulime, që riprodhojnë sjelljen e një sistemi. Fillimet e tyre njihen menjëherë pas luftës së dytë botërore në meteorologji dhe fizikë nukleare. Më pas janë përdorur për modelime të proceseve në fusha të ndryshme si ekonomi, kimi, biologji, etj. Ato mund të përdoren për të shqyrtuar dhe për të fituar njohuri të reja si edhe për të vlerësuar performancën e sistemeve në zgjidhjet analitike të problemeve komplekse. [10]

Këto modele gjejnë një zbatim të gjerë në problemet me bazë matematikore, zgjidhja e të cilëve nuk mund të kryhet manualisht. Në këtë drejtim, kohët e fundit janë duke u kryer shumë studime shkencore. Simulimi kompjuterik na mundëson studimin e shumë modeleve komplekse si, progresi i një ekonomie, vendimmarrjet në një kompani etj. (11)
Këto simulime bëhen të domosdoshme kur metodat standarde të llogaritjeve nuk mund të arrijnë rezultatet e kërkuara.
Winsberg(2009) (12) përdor interpretime të ndryshme për simulimin kompjuterik.

(i) Simulimi është një algoritëm, që ekzekutohet në një kompjuter, dhe që studion sjelljen e përafërt të një modeli matematikor të përbërë nga ekuacione të pazgjidhshëm me mënyra analitike.
(ii) Simulimi konsiderohet si një metodë, që përdoret për studiuar ato sisteme, të cilët modelimin më të saktë e kanë të bazuar në ekuacionet e pazgjidhshëm me mënyra analitike.

Modelet informatikë mund të quhen modele të modeleve. Shpesh herë këto probleme hasin vështirësi për tu implementuar për arsye të ndryshme. Në pjesën më të madhe të rasteve ato shërbejnë për të simuluar një model të ndërtuar në një fushë të caktuar.
Vështirësitë mund të jenë të drejtpërdrejta, pra ne e kemi të vështirë të simulojmë një model për shkak të kompleksitetit të modelit, ose për shkak të pamundësisë së teknologjisë aktuale.
Vetë modeli i fenomenit mund të jetë i vështirë për tu ndërtuar, dhe si rrjedhojë edhe simulimi i tij do të jetë vështirë të arrihet i plotë e funksional. Në këtë rast shpesh haset një debat midis modeluesve të sistemit edhe simuluesve të modelit, sidomos kur akoma është e paqartë nëse problematikat janë në hapin e parë apo të dytë. Zgjidhja e problemeve të tilla kërkon mirëkuptim nga të dyja palët, dhe në më të shumtën e rasteve janë simuluesit ata që përpiqen të orientohen drejt të qënurit modelues të sistemit. Mund të përmendim rastet e informatikanëve që ndërtojnë aplikime ekonomike, inxhinierike, etj.
Këto vështirësi, natyrshëm lindin pyetjen, nëse duhet t'i besojmë rezultateve të arritura përmes simulimit. Simulimet mund të mos e paraqesin si duhet modelin, ose rezultatet e tyre mund të mos jenë të plota. Kjo ka çuar në zhvillimin e një teorie mbi metodat e testimit të rezultateve të arritura nga simulimi i modeleve.

1.3 Kërkimi shkencor në modelimet informatike

Ekziston një debat midis studiuesve nëse modelimet informatike përbëjnë risi po jo.
Nëse do i referohemi Frigg dhe Reiss (2009) (13), do vëmë re që ata pranojnë që simulimet kompjuterike kanë një ndikim tepër të madh në zhvillimet e shkencës, por ata nuk shohin të ketë risi shkencore në këto simulime.
Teoria e tyre është kundërshtuar nga Eric Winsberg(2009) dhe Paul Humphreys (14).
Humphreys paraqet pesë argumente për treguar që modelimet informatike përbëjnë një metodë shkencore:

1) Metafizik - simulimet krijojnë mjedise të përshtatshme ku eksperimentet mund të kryhen më lehtësisht se në botën reale
2) Epistemologjik – për të realizuar një simulim, kërkohen njohuri shtesë nga studiuesit.
3) Semantik – kërkohet te analizohet se si modelet i referojnë fenomenit konkret
4) Metodologjik – çdo simulim është i veçantë nga të tjerët dhe qëndron midis teorisë dhe eksperimentit
5) Progresi – progresi është i lidhur ngushtë me progresin teknologjik

1.4 Modelet informatikë në ekonomi

Simulimi në ekonomi pati një zhvillim kur ekonomia filloi të bazohej ne teorema e vërtetime. Fillimisht simulimi kishte një rol dytësor, kryesisht në gjetjen e analizave numerike, por më pas ai u pa si metodologji më vete [15] dhe si një mjet për të mbështetur eksperimentet (16). Përpjekja për të gjetur dallimin midis përdorimit të simulimit si instrument apo si teori ka ndihmuar në përforcimin e idesë se simulimi është një metodologji. Për më tepër interpretimi i ekonomisë përmes sistemeve komplekse, ka çuar në konceptimin e simulimit si një metodë modelimi.
Rritja e fuqisë llogaritëse të kompjuterëve, ka çuar në ndërtimin e shumë modeleve në këtë fushë duke ju lejuar studiuesve arritjen e rezultateve konkrete të zbatimit të teorive të ndryshme pa u ndalur në vështirësitë që nuk do të lejonin aplikimin praktik të tyre.

1.5 Ekonometria si shkencë

Ekonometria ka qëllim ti japë përmbajtje empirike marrëdhënieve ekonomike për testimin e teorive ekonomike. Duke theksuar aspektet sasiore të marrëdhënieve ekonomike, ekonometria kërkon unifikimin e matjeve dhe teorisë në ekonomi. Analizimi i problemeve ekonomikë nuk do ishte i plotë nëse bazohemi vetëm në teori, ose vetëm në matje. Si teoria, edhe matjet nuk janë të mjaftueshme për të na ndihmuar të kuptojmë fenomenet ekonomike, nëse përdoren të ndara nga njëra tjetra.
Modeli i parë makroekonomik u ndërtua nga Klein (1947,1950). Në një kohë të shkurtër u ndërtuan modele makroekonomike thuajse për çdo vend të industrializuar. Modelet makroekonomikë u përdorën për analizimin e politikave ekonomike si dhe parashikimin e tyre. Ndërtimi i këtyre modeleve solli në disa probleme përllogaritjeje, zgjidhja e të cilëve kishte rëndësi për zhvillimin e modelimit makroekonomik dhe për praktikat ekonometrike në përgjithësi. Zgjidhja e problemeve përllogaritës do të ishte e pamundur pa përdorimin e teknologjisë.

2. Ekonomia si ngjarje sociale, analiza e densiteteve të probabiliteteve të përbashkëta

Studimi i fenomeneve dhe ngjarjeve ekonomike bazohet në analizat empirike. Ky drejtim i kërkimit në ekonomi përdor të dhënat për variablat ekonomikë për të vërtetuar hipotezat teorike mbi marrëdhëniet që ekzistojnë ndërmjet variablave ekonomikë dhe për vlerësimin dhe implementimin e politikave ekonomike. Këto marrëdhënie janë rezultat i vendimmarrjes individuale dhe ndërveprimit ndërmjet të gjithë agjentëve (individëve, familjeve, kompanive, institucioneve financiare, dhe qeverisë e autoriteteve monetare) në një ekonomi. Hendry and Richard (1983) e përshkruajnë rezultatin e ndërveprimit të tyre si një fenomen social multidimensional, i cili ka karakteristikat e një ngjarjeje të rastit. Në perceptojmë këtë ngjarje të rastit përmes ndryshimeve të vlerave të variablave ekonomikë, të cilët përfaqësojnë dimensionet e saj të shumta. Në këtë drejtim, zhvillimet ekonomike reflektojnë ndërveprimin e të gjithë variablave ekonomikë. Studimi i ekonomisë fokusohet te ky ndërveprim, në procesin natyror që gjeneron rezultatin përfundimtar të vëzhguar ekonomik. Ky mekanizëm quhet procesi i gjenerimit të të dhënave (PGjDh). Ericson Hendry dhe Mison (1998) e përshkruajnë PGjDh në formën e një hapësire probabilitare (Ω, F, P(.)).

Në studimin e ekonomisë empirike, PGjDh është shprehur në formën e një procesi të kushtëzuar të kushteve fillestare (vlerat e variablave ekonomikë në kohën t=0) dhe një vektor me parametra

$$D_X(X_T \mid X_0,\zeta) = \prod_{(t=1)}^{T} D_X(x_t \mid X_{t-1},\zeta_t).$$

Ky formulim ka çuar në modelimin e ekonomisë në formën e sistemeve të diferencës dhe ekuacioneve diferenciale, ose si njihet ndryshe në ekonomi, procesi Vektor AutoRegresiv (VAR). Sims (1980) propozoi VAR si një mjet të vlefshëm dhe të nevojshëm të kërkimit empirik. VAR është bërë një mënyrë vërtet e favorshme për të eksploruar dhe analizuar sjelljen ekonomike.

Sidoqoftë, Tanku dhe Ceca (2013) vënë re që "implementimi praktik i këtij mjedisi teorik mundësohet vetëm nga një bashkësi kufizimesh e pengesash të ngurta, në të cilat ai mund të jetë joreal dhe t‘i largojë të dhënat nga natyra e tyre e reale, duke mbartur probleme të mundshme në interpretimin e rezultateve dhe analizës ekonomike.

Ata sugjerojnë një metodë të re alternative për përafrimin e kërkimit empirik bazuar në përcaktimin e ekonomisë si një ngjarje e rastit multidimensionale dhe te metoda e vlerësimit të densiteteve. Ata e përshkruajnë ekonominë si një ngjarje probabilitare shumë dimensionale, e cila në çdo moment përshkruhet nga realizimet (vlerat) e njëkohshme përgjatë të gjithë dimensioneve (domethënë të të gjithë variablave

ekonomikë. Pra ekonomia është një hapësirë d dimensionale, (ku d përfaqëson numrin e variablave të marrë në konsideratë) në të cilën çdo hapësirë m dimensionale e të njëjtave variabla (ku m<d) përfaqëson një nënhapësirë të ekonomisë të marrë në shqyrtim (të hapësirës origjinale d dimensionale). Ky përcaktim i hap rrugën përcaktimit të ekonomisë si një sekuencë hapësirash që zgjerohet në dimensione. Më tej Tanku dhe Ceca shtojnë se duke supozuar që të gjithë variablat (dimensionet) janë linearisht të varur me variablat origjinalë, çdo nënhapësirë ose hapësirë e re do të ruajë të njëjtën PGjDh, respektivisht me më pak apo më shumë dimensione.

Në parim këto janë kushtet mbi të cilat autorët mbështesin gjykimin e tyre mbi marrëdhëniet dhe ndërveprimin ndërmjet variablave ekonomikë. Ky gjykim bazohet në vlerësimin e densiteteve të përbashkëta shumë dimensionale duke përdorur metodën e vlerësimit të densiteteve. Investigimi empirik fokusohet në ndërtimin dhe vlerësimin e densiteteve duke përdorur vektorët d dimensionalë të të dhënave ekonomike simultane, ku d përfaqëson numrin e variablave të marrë në shqyrtim.

Sipas autorëve kjo metodë ka avantazh sepse studimi i ekonomisë përmes vlerësimit të densiteteve dhe krahasimit të tyre nuk e ka të nevojshme kufizimin e rrjedhimisht nuk kufizohet nga aplikimi i supozimeve teorike e praktike, që janë të domosdoshme në përdorimin e metodës së serive kohore.

Në parim Tanku dhe Ceca përdorin vlerësimin e densiteteve si një metodë për përafrimin e densitetit real me bërthama të njohura. Artikulli jep funksionin e densitetit për ngjarje një dhe multidimensionale respektivisht me bërthame gausiane dhe epanechnikov, duke mundësuar formën analitike funksionale të funksionit të densitetit shumëdimensional. Përdorimi i bërthamave të ndryshme përbën avantazh, pasi na jep mundësinë e zgjedhjes së bërthamës që i përshtatet më mirë shpërndarjes së panjohur. Për më tepër, zgjidhja analitike e tyre përmban parametra përmirësues shtesë, të cilët janë fokusuar në të mirën e përshtatjes së densitetit të panjohur që rrjedh prej PGjDh.

Pavarësisht nga fakti që ata japin formën funksionale analitike të funksionit të densitetit shumëdimensional, autorët propozojnë përdorimin e paraqitjes grafike si metodë investigimi për interpretimin e marrëdhënieve ekonomike. Të kufizuar nga pamundësia për të perceptuar dhe analizuar vetëm objekte tre dimensionale, Tanku dhe Ceca propozojnë interpretimin e PGjDh si përfaqësim i ngjarjes d dimensionale në nënhapësirën tre dimensionale, ku akset horizontale përfaqësojnë vetëm dy dimensione të ngjarjes, dhe aksi vertikal përfaqëson densitetin për çdo vlerë të mundshme në planin e krijuar nga dy dimensionet në fjalë. Vlerësimi i densitetit mund të gjendet për çdo pikë, pasi funksionet e densiteteve janë funksione të vazhdueshëm.

Se fundi autorët përshkruajnë intuitën dhe interpretimin praktik të densiteteve në analizën ekonomike. Në parim ky interpretim bazohet në formën e shpërndarjeve të

përbashkëta dy dimensionale dhe e ilustrojnë metodën për shpjegimin e sjelljes së inflacionit në ekonominë shqiptare.

Kjo metodë është zhvilluar më tej te Tanku dhe Ceca (2014), në të cilën autorët plotësojnë metodën e vlerësimit të densiteteve të përbashkëta shumë dimensionale duke përcaktuar metodologjinë dhe metrikën e leximit të grafikëve të densiteteve të përbashkëta të vlerësuara.

Për këtë qëllim autorët propozojnë "prerjen tërthore" të definuar si "densitetin e përbashkët të kushtëzuar" të hapësirës shumë dimensionale d-m (ku d-m përfaqëson hapësirën e variablave të varur, sjelljen e të cilave ne dëshirojmë të studiojmë) sipas çdo vlere të çfarëdoshme të secilit prej m (ku $m \leq (d-1)$) variablave të tjerë që i konsiderojmë si variabla shpjegues. Leximi dhe interpretimi bazohet në vlerësimin dhe interpretimin e karakteristikave numerike të densiteteve të përftuar për çdo "prerje tërthore". Praktikisht mund të gjenerohet një prerje tërthore për çdo variabël,

Procesi mund të përsëritet për gjithë d – 1 dimensione (variabla) në ekonomi. Duke krahasuar këto karakteristikë numerike të densiteteve të përbashkëta të kushtëzuara të variablave të varur për çdo vlerë të variablit shpjegues, studiuesi mund të kuptojë se variablat e varur evoluojnë në përgjigje të ndryshimeve në variablat shpjegues.

Për të vlerësuar densitetet me kusht Tanku dhe Ceca (2014) paraqesin densitetet me kusht ($\hat{f}\left(\underline{t}/t_k\right) = \frac{\hat{f}(\underline{t})}{\hat{f}_{\{k\}}(t_k)}$) në të cilën numëruesi përfaqëson densitetet e përbashkëta shumë dimensionale të dhënë nga Tanku e Ceca (2013), ndërsa emëruesi përfaqëson densitetin anësor të vlerësuesit të densitetit për çdo variabël shpjegues $t_k \in m$.

Se fundi Tanku dhe Ceca (2014) kanë propozuar riorganizimin e bazës së të dhënave duke i zhvendosur variablat e "varur" me një lag kohor L (ku L $\in \{1, 2, 3, 4, 5{,}6 \ldots\}$ në varësi të rastit). Kjo zhvendosje, që përputh vlerat aktuale të variablit apo grupit (nënhapësirës) së variablave të varur me vlerat e materializuara L periudha më parë të variablave shpjegues për të investiguar (vlerësuar) ekzistencën e shkakësisë dhe masën e reagimit me të cilin zhvillimet në variablat shpjegues transmetohen në vlerat e ardhme të variablave të varur L periudha më pas. Autorët ekspozojnë përdorimin e metodës së prerjeve tërthore të densiteteve të përbashkëta shumë dimensionale për të testuar lidhjen ndërmjet parasë dhe inflacionit (më saktë për të testuar vërtetësinë e ekuacionit të këmbimit) për rastin e ekonomisë amerikane për periudhën 1959-2013 gjithmonë duke përdorur rezultatet grafike për pritjen matematike të prerjeve tërthore dhe devijimin e tyre standard.

3. Organizimi i të dhënave dhe testimi i tyre

Zhvillimi i teknologjisë ka çuar në informatizimin e shumë proceseve. Këto procese shoqërohen me operacione ditore hedhjeje te dhënash si dhe me një migrim masim të të dhënave të mëparshme të cilat ndodhen në letër. Në më të shumtën e rasteve, migrimet kryhen para se aplikimi të vendoset në punë.

Hedhja e informacionit në aplikime të ndryshme shoqërohet gjithmonë me gabime. Rezultati i të dhënave të hedhura gabim në një proces dixhitalizimi kap një vlerë 5% të të dhënave të përgjithshme, pra një vlerë jo e pakonsiderueshme. Shumë prej aplikimeve kanë forma kontrolli për shmangien e gabimeve, por gjithsesi është e pamundur eliminimi total i tyre, sidomos në rastet e një procesi masiv dixhitalizimi në një kohë të përshpejtuar. Ato kërkojnë një proces testimi, në mënyrë që të prodhojnë rezultate të besueshme.

Më poshtë trajtohet modelimi informatik i procesit të testimeve që lidhen me cilësinë e hedhjes së të dhënave për seri të vazhdueshme kohore. Do të paraqitet një strukturë e bazës së të dhënave e përshtatshme për ruajtjen e serisë së të dhënave për variabla të ndryshëm. Mbi këto të dhëna do të aplikohen teste të ndryshëm tregues e korrigjues të gabimeve.

Për gjetjen e këtyre gabimeve janë zhvilluar metoda të ndryshme.

Metoda që ofron saktësinë më të madhe është metoda e quajtur "hedhje dopjo" [17], ku secila prej të dhënave hidhet nga dy operatorë të ndryshëm në dy kopje të sistemit. Në përfundim të hedhjes së të dhënave bëhet një kontroll vetëm mbi të dhënat që nuk korrespondojnë me njëra tjetrën. Niveli besimit për saktësinë mund të rritet duke shtuar operatorët që hedhin të njëjtat të dhëna. Kjo metodë e ul ndjeshëm numrin e gabimeve, por ka një kosto të lartë. Në këtë rast, kosto rritet në përpjesëtim të drejtë me numrin e operatorëve. Ajo mund të përdoret kur numri i të dhënave është i vogël, ose në rastin kur toleranca ndaj gabimeve është e papranueshme.

Metodë tjetër për kontrollin e të dhënave, është përdorimi i kontrolluesve të të dhënave të hedhura nga operatorët. Në këtë rast, kosto është më e ulët se në rastin e parë, pasi numri i kontrolluesve mund të jetë më i ulët se numri i operatorëve, p.sh. mund të kemi një kontrollues për çdo katër operatorë. Por numri i gabime mund të jetë më i madh.

Metodat e mësipërme mundësojnë eliminimin e gabimeve gjatë hedhjes së të dhënave, por nuk na japin informacion për cilësinë e matjes së këtyre të dhënave. Për të kontrolluar cilësinë e të dhënave, si dhe në rastin kur nuk mund të rikontrollojmë të dhënat me operatorë shtesë, sygjerohet përdorimi i metodave statistikore, të cilat tregojnë ato të dhëna që potencialisht mund të jenë të gabuara. [18]

Materiali dhe metodat

Përgjithësisht të dhënat mund të ndodhen në skedarë tekst të formateve të ndryshëm, p.sh. Excel.

Në skedarë të tillë, matjet për secilin variabël mund të vendosen në kolona të veçanta, ku secila kolonë mban vlerat e variablit të regjistruara gjatë kohës.

Bëjmë kalimin e të dhënave në një format të ri për të kryer procedura automatizimi dhe testime më të ndërlikuara.

Si sistem për menaxhimin e bazës së të dhënave është përdorur SQL Server 2008 R2.

Gjuha për krijimin e objekteve është T-SQL.

Krijohet tabela "seriteBruto" e të dhënave bruto

```
CREATE TABLE seriteBruto(
        [id] [int] IDENTITY(1,1) NOT NULL,
        [Variabel] [nvarchar](50) NULL,
        [Vendndodhje] [nvarchar](50) NULL,
        [Data] [datetime] NULL,
        [Vlera] [nvarchar](50) NULL
)
```

Në këtë tabelë mund të regjistrohen të dhëna bazë. Sipas kërkesës, tabela mund të zgjerohet duke shtuar fusha të reja, p.sh. matje të ndryshme për të njëjtën datë, etj.

Vihet re që të në këtë tabelë regjistrojmë dy atribute kryesore për të dhënat, variablin dhe vendndodhjen. Zakonisht, seritë e të dhënave janë për çdo dyshe të këtyre atributeve.

Të dhënat duhet të merren nga aplikimet dhe pas transformimeve përkatëse te ngarkohen në tabelën e mësipërme. Procesi i transformimit mund të automatizohet nëpërmjet ndërtimit të proçedurave T-SQL.

Kalimi i të dhënave në një vend të vetëm na krijon mundësinë të ndërtojmë vetëm një proçedurë për secilin test të pavarur nga variabli, i cili do të testohet. Në këtë rast ky variabël mund të kalojë si parametër.

Para se të kalohet testimet, duhet të bëhet unifikimi i vlerave. Të dhënat që nuk janë plotësuar markohen me një vlerë të caktuar, p.sh. -55555.

Update seriteBruto set vlera=-55555 where vlera is null

Manipulimi i të dhënave

Studiuesit, për të parë marrëdhëniet ndërmjet variablave, përveç vlerave të tyre janë të prirur të shikojnë edhe marrëdhëniet e diferencave të para dhe ndryshimet në përqindje.

Tanku dhe Ceca (2012) thonë që duke marrë parasysh marrëdhënien midis variablave, ndryshimet në një variabël duhet të vërehen edhe të variablat e tjerë.

Shtojmë dy kolona te tabela *seriteBruto*

Alter table seriteBruto add column Diferena

Alter table seriteBruto add column Perqindje

Komanda për plotësimin e vlerave të diferencës së parë

```
Update seriteBruto set Diferenca= Vlera -(
        Select Vlera from seriteBruto sB2
        Where seriteBruto.variabel=sb2.variabel
          And seriteBruto.Vendndodhje =sb2.Vendndodhje
          And seriteBruto.Data-1=sb2.Data )
```

Komanda për plotësimin e vlerave të diferencës së dytë

```
Update seriteBruto set Perqindje= Vlera/(
        Select Diferenca from seriteBruto sB2
        Where seriteBruto.variabel=sb2.variabel
          And seriteBruto.Vendndodhje =sb2.Vendndodhje
          And seriteBruto.Data-1=sb2.Data
)
```

Kategoria e parë (testi i limiteve)

Testi i limiteve kontrollon që vlerat e një variabli të jenë brenda limiteve të caktuara, p.sh. çmimi i bukës vendoset nga 0 në 120 lekë, pasi dihet që gjatë kohës nuk ka pasur çmim jashtë këtij limit. Çdo vlerë jashtë këtij limiti do quhet vlerë e gabuar dhe do kalohet për rishikim.

Automatizimi i testit kryhet si me poshtë:

Krijojmë tabelën ku vendosim testet

```
CREATE TABLE [testet](
        [id] [int] IDENTITY(1,1) NOT NULL,
        [testues] [nvarchar](50) NULL,
        [kategori] [nvarchar](200) NULL,
        [tabela] [nvarchar](200) NULL,
        [kushti] [nvarchar](200) NULL,
        [pershkrim] [nvarchar](200) NULL,
        [fusha1] [nvarchar](200) NULL,
        [fusha2] [nvarchar](200) NULL,
        [limiti_poshtem] [float] NULL,
        [limiti_siperm] [float] NULL
)
```

Komanda për listën e testeve është:
select id, kategori, tabela, kushti, pershkrim, fusha1, fusha2, limiti_poshtem, limiti_siperm from testet where kategori=@kategori order by id

Duke krijuar një kursor mbi setin e të dhënave të marra më sipër, mund të nxjerrim listën e vlerave që nuk plotësojnë kushtin për secilin test me anë të komandës:

*Exec("select * from "+@tabela+" where "+ @kushti)*

Kategoria e dytë (testet spacialë) [19] [20]

Këto teste kanë qëllim kontrollues – korrigjues, bazuar në konceptin e fqinjësisë gjeografike të çdo vendndodhjeje me vendndodhjet përreth. Në formë të përgjithshme shënohet t(i) vlera e variablit në ditën e i-të.

Testi nr.1

Nëse |t(i)-t(i+1)|> limit_1

Nëse për çdo k €{1,2,3}: |[t(i,k)-t(i+1,k)]-[t(i)-t(i+1)]|>limit_2

Testi dështon, vlera t(i) duhet të kontrollohet

Shënim: k €{1,2,3} i referohet tre vendndodhjeve të ndryshme.

t(i,k)-jep vlerën e serisë në ditën e i-të në vendndodhjen e k-të

Vlerat limit_1 dhe limit_2 duhen të vendosen sipas rastit të serisë së të dhënave në shqyrtim. P.sh. në rastin e një serie të temperaturave ditore, mund të themi që lëvizja prej 10 °c, midis dy vlerave të njëpasnjëshme, është e dyshimtë, duke vendosur kështu limit_1=10. Gjithashtu mund të themi që ndryshimet e diferencave të vlerave të njëpasnjëshme midis dy vendndodhjeve fqinje nuk duhet të jenë më shumë se 3 °c, duke vendosur kështu limit_2=3.

Nëse ndodh që diferenca e vlerave të njëpasnjëshme të jetë më e madhe se limit_1, atëherë kjo mund të ketë ndodhur ose si rrjedhojë e ndonjë anomalie të sistemit, ose si rrjedhojë e hedhjes gabim së të dhënës. Shikohet diferenca midis vlerave të njëpasnjëshme në vendndodhjet fqinjë, përgjithësisht tre janë të mjaftueshëm. Nëse në secilën vendndodhje, diferenca, midis vlerave të njëpasnjëshme, ndryshon më shumë se një vlerë e caktuar (limit_2) nga diferenca t(i)-t(i+1), atëherë vlera t(i) markohet si e dyshimtë, përndryshe mendohet si e saktë.

Testi nr.2

Nëse për çdo k € {2,3,4}: |t(i,l)-t(i,k)|>limit

Ath t(i, l) quhet matje e matur keq.

Apliko interpolimin gjeografik si më poshtë:

t(i,l)=[t(i,2)+t(i,3)+t(i,4)]/3

Shënim: k €{1,2,3} i referohet tre vendndodhjeve të ndryshme fqinje.

t(i,k)-jep vlerën e serisë në ditën e i-të në vendndodhjen e k-të

Nëse ndodh që për një datë të caktuar, diferenca e vlerës së variablit me secilën prej vlerave të variablit në vendndodhjet fqinje të jetë më e madhe se *limit*, atëherë kjo mund të ketë ndodhur ose si rrjedhojë e ndonjë anomalie të sistemit, ose si rrjedhojë e hedhjes gabim së të dhënës. Në këtë rast vlera t(i) markohet si e dyshimtë. Në qoftë se kërkohet korrigjimi i kësaj vlere ajo interpelohet duke u zëvendësuar me mesataren e vlerave të njëjtit variabël në të njëjtën datë në vendndodhjet fqinje.

Vlera limit duhet të vendoset sipas rastit të serisë së të dhënave në shqyrtim.

Testi dispersionit [21]

Llogariten vlerësimet e mesatares dhe dispersionit të serisë për 30 ditë para ditës i, sipas formulave:

$$\overline{t(\imath)} = \frac{1}{30}\sum_{k=i-1}^{i-30} t(k);$$

$$\widetilde{\sigma(\imath)^2} = \frac{1}{30}\sum_{k=i-1}^{i-30}\left[t(k) - \overline{t(\imath)}\right]^2$$

Vlera është në rregull nëse vlera e matur pasardhëse t(i) është:

$$\overline{t(\imath)} \pm 1.96\left(\frac{\widetilde{\sigma(\imath)}}{\sqrt{30}}\right)$$

Metoda më sipër nënkupton që shpërndarja e serisë t(i) është normale.

Niveli i besimit 95 % mund te ndryshohet.

Më poshtë paraqesim pjesën transact SQL të procedurës që gjen vlerat e dyshuara për një variabël të caktuar.

Proçedura përdor dy tabela të përkohshme, vleraTemp, e cila përdoret për llogaritje brenda procedurës, si dhe vleraTest2, në të cilën regjistrohen vlerat e dyshuara të serisë. Procedura do të marrë si parametër emrin e variablit.

```
create table  vleraTemp( id int, vlera float)

create table  vleraTest(
        variabli nvarchar(50),
        stacioni nvarchar(50),
        data date,
        t float,
        s float,
        vlera float
)
```

Në këtë proçedurë, variabli merret si parametër *@variabli*. Atributi i dytë i nevojshëm për tu përcaktuar është vendndodhja. Ndërtojmë një kursor që bredh listën e vendndodhjeve:

```
declare kursoriSt cursor  FOR  select vendndodhje from vendndodhjet
```

Kursori do e kalojë listën e vendndodhjeve te @vendndodhje.

Për secilën vendndodhje, pastrojmë tabelën e përkoshshme vleraTemp dhe inicializojmë @i=1:

```
set @i=1

delete vleraTemp
```

Ndërtojmë një kursor që bredh datat një e nga një:

```
declare kursoridate cursor  FOR  select data,vlera from seritebruto where vendndodhje=@vendndodhje and variabli=@variabli order by data
```

Kursori do e kalojë listën te *@data* dhe *@vl.*

Në ciklin e parë, kemi *@i=1*, prandaj do kemi incializim te *@data1* dhe jo *@data2*. Kjo sjell që diferenca e *@data1* dhe *@data2* nuk do të jetë një, pasi @data2 ka një vlerë të mëparshme. Vini re që kjo ndodh vetëm në ciklin e parë të kursorit të datës për secilën vendndodhje, prandaj ne duhet të pastrojmë tabelën e përkohshme vleraTemp.

```
if @i=1
    set @data1=@data
else
    set @data2=@data
```

```
if dateadd(day,1,@data1)!=@data2
begin
    delete vleraTemp
    set @data1=@data
    set @i=1;
end
```

Për 30 ditët e para nuk mund të kryejmë veprime, pasi nuk kemi se c'mesatare të gjejmë, prandaj për 30 vlerat e para vetëm i vendosim te *vleraTemp*.

Nëse kemi kaluar 30 ditëshin e parë, gjejmë mesataren e tabelës dhe e vendosim te @t. Gjejmë dispersionin dhe e vendosim te @s.

Kontrollojmë vlerën nëse është në rregull sipas formulës më sipër. Nëse jo, regjistrojmë në tabelën vleraTest datën, vlerën, mesataren dhe shangien e dispersionit.

Vini re që në tabelën *vleraTemp*, mbajmë vetëm vlerat e 30 ditëve të fundit. Kjo arrihet duke fshirë vlerën më të hershme të regjistruar në tabelë.

Inicializimi *@data1=@data* në fund të ciklit dhe *@data2=@data*,pasi *@i*!=1, në fillim të ciklit pasardhës mundëson *@data2=@data1+1* duke mos pastruar tabelën vleraTemp dhe vendosur *@i*=1.

```
if @i<31
begin
  insert into vleraTemp values (@i,@vl);
  set @i=@i+1;
end
else
begin
  select @t=round(sum(vlera)/30,2) from vleraTemp
  select @s=round(sum(power(vlera-@t,2))/30,2) from vleraTemp
  if abs(@t-@vl)>2.05*round(sqrt(@s),2)*round(sqrt(31/30),2)
    insert into vleraTest values (@seri, @stacioni, @data, @t,
2.05*round(sqrt(@s),2)*round(sqrt(31/30),2), @vl)
  insert into vleraTemp values (@i,@vl)
  delete vleraTemp where id=@i-30
  set @i=@i+1;
end
set @data1=@data
```

Testi i Mann-it [22]

Shënim: Testi i Manit kryhet mbi seritë vjetore të të dhënave, pasi kërkon një numër të caktuar të dhënash për të pasur rezultate me kuptim.

Fillimi i Testit:

Për serinë që do të testohet, vjetore, (p.sh. N=10 vjet) jepet seria **t(i)**
i varion nga 1 deri ne 10
1. S(1)=0

2. Fiksojmë i € {2, 10}.
3. Llogaritet statistika: S(i) = S(i-1) + {"Numrin e rasteve që termi t(i) > t(k) per k=**1** deri **i-1**"} + 0.5 * {"Numrin e rasteve që termi t(i) = t(k) per k=**1** deri **i-1"}**
4. Për çdo **i** llogariten pritja matematike dhe varianca
   E(i)=i*(i-1)/4 Var(i)=i*(i-1)*(2*i+5)/72
5. Llogaritet Statistika e Testit Mann sipas formulës:
   U(i) = [S(i)-E(i)]/SQRT[Var(i)]
6. Nqs U(i) > 1.96, atëherë seria ka heterogjenitet në vitin **i** me probabilitet 95%
7. Fund

Më poshtë paraqesim pjesën transact SQL të procedurës që gjen vlerat e dyshuara për një variabël të caktuar. Procedura do të marrë si parametër emrin e variablit.

Në këtë proçedurë, variabli merret si parametër @variabli. Atributi i dytë i nevojshëm për tu përcaktuar është vendndodhja. Ndërtojmë një kursor që bredh listën e vendndodhjeve:

declare kursoriSt cursor FOR select vendndodhje from vendndodhjet

Kursori do e kalojë listën e vendndodhjeve te @vendndodhje.

Bëjmë inicializimet e mëposhtme. Variabli @i do tregojë vitin. @s është shumë progresive gjatë viteve, prandaj inicializohet jashtë ciklit të viteve.

set @i=1
set @s=0

Testi i Mannit shërben për të treguar vitet ku seria ka heterogjenitet me probabilitet 95%. Ndërtojmë një kursor që bredh vitet me radhë:

declare kursori cursor FOR select viti,avg(vlera) from seritebruto where variabli=@variabli and vendndodhje=@vendndodhje group by viti order by viti

Kursori do e kalojë listën te @viti dhe @Sasia

set @s_re=0

select @s_re=sum(a.mes) from (select viti,case when avg(vlera)<@Sasia then 1 when avg(vlera)=@Sasia then 0.5 else 0 end mes from seritebruto where variabli=@variabli and *vendndodhje=@vendndodhje* and viti<@viti group by viti) a

```
set @s=@s+isnull(@s_re,0)
set @e=@i*(@i-1)/cast(4 as float)
set @v=@i*(@i-1)*(2*@i+5)/cast(72 as float)
if(@i>1)
  set @u=(@s-@e)/sqrt(@v)
if (abs(@u)>=2)
 insert into vleraTestMan values (@stacioni,@viti,@u)
set @i=@i+1
```

Cikli përsëritet për të gjithë vitet.

4. Particionimi i të dhënave

Skema bazës së të dhënave e propozuar në kapitullin më sipër, përmban tabela të cilat kanë një numër të konsiderueshëm të dhënash. Kjo do të sjellë vonesa në ekzekutimin e kërkesave në bazën e të dhënave. Implementimi i particionimit, do të rrisë performancën dhe do të ulë kohën e ekzekutimit të këtyre kërkesave.

Particionimi është një teknikë që ristrukturon të dhënat në tabelë. Fizikisht tabela ndahet në pjesë, të quajtura particione, dhe secila prej këtyre pjesëve mund të ruhet në disqe të ndryshme. Pra kjo mundëson lexim paralel të të dhënave.

Particionimi na lejon të ndajmë rreshtat e një tabele bazuar mbi një shprehje logjike të një kolone, e quajtur çelësi i particionimit.

Particionimi i tabelës mund të bëhet si në fillim të krijimit të tabelës, edhe pasi tabela është mbushur me të dhëna. Kjo na krijon mundësinë, që të përdorin si çelës particionimi, vetë variablin. Në këtë mënyrë nuk do ketë rëndësi se sa variabla ruajmë në tabelë, pasi ato duke u ruajtur të ndarë, mund të tërhiqen më lehtë.

Më poshtë trajtohet particionimi i tabelë seriteBruto. Do të paraqitet një teknikë particionimi e përshtatshme për ruajtjen e serisë së të dhënave për variabla të ndryshëm.

Testimet janë janë ekzekutuar në një server me specifikimet e mëposhtme:

Tabelë 4.1 Specifikimet e serverit

Specifikime teknike	
Modeli	HP PROLIANT DL 380 G5
Shpejtësia e CPU	2.83 GHz
Memorja RAM	8 GB
Numri i CPU-ve	4
Sistem Operativ	Windows Server 2008 64-Bit

Si sistem për menaxhimin e bazës së të dhënave është përdorur SQL Server 2008 R2.

Koha e ekzekutimit është matur duke punuar me SQL Server Profiler.

Metodologjia

Është krijuar një tabelë e particionuar seriteBruto me çelës particionimi të tipit nvarchar.

Tabela është populluar me të dhëna mbi katër variabla makroekonomikë.

Përpara ekzekutimit të kërkesave, është pastruar cache-ja dhe buffer-at.

Eksperimenti

Krijohet funksioni i particionimit. Funksioni ka për qëllim të përcaktojë sa particione do të krijohen dhe cilat do të jenë vlerat e secilit prej tyre. Duke qenë se duam të particionojmë tabelën mbi variablat e ruajtur, nuk mund të dimë saktësisht numrin e particioneve. Kjo sepse çdo shtim i një variabli në tabelë, duhet të shoqërohet me shtimin e një particioni të ri.

Fillimisht funksioni krijohet me një particion fiktiv.

CREATE PARTITION FUNCTION PartRang (nvarchar(50))
AS RANGE LEFT FOR VALUES ('a')

Funksioni i particionimit është quajtur partRang. Fusha çelës është e tipit nvarchar.

Krijohet skema e particionimit, e cila do t'i referohet funksionit të particionimit të krijuar më lart

CREATE PARTITION SCHEME PartSkeme AS
PARTITION PartRang
ALL TO ([PRIMARY])

Tabela e particionuar

```
CREATE TABLE seriteBruto(
        [id] [int] IDENTITY(1,1) NOT NULL,
        [Variabel] [nvarchar](50) NULL,
        [Vendndodhje] [nvarchar](50) NULL,
        [Data] [datetime] NULL,
        [Vlera] [nvarchar](50) NULL
)
ON
PartSkeme (Variabel)
```

Vlera e kolonës *variabël* do të përcaktojë se në cilin particion do të ruhet rreshti.

Tabela do të popullohet me të dhëna për katër variabla: ER, CPI, R, M.

Për çdo variabël të ri të regjistruar, p.sh. ER, kryejmë procedurën si më poshtë.

Shtojmë një filegroup të ri, i cili do shërbejë si vendruajtje për variablin e ri.

```
ALTER DATABASE [baza e të dhënave] ADD FILEGROUP FG_ER;
GO
ALTER DATABASE [baza e të dhënave]
ADD FILE
(
   NAME = dat1_ER,
   FILENAME = 'C:\Program Files\Microsoft SQL
Server\MSSQL10_50.MSSQLSERVER\MSSQL\DATA\dat1_ER.ndf',
   SIZE = 5MB,
   MAXSIZE = 100MB,
   FILEGROWTH = 5MB
)
TO FILEGROUP FG_ER;
```

Përcaktojmë file grup të ri vendruajtjen e vlerave të variablit të ri.

ALTER *PARTITION SCHEME PartSkeme next used [FG_ER]*

Ndryshojme funksionin e particionimit.

ALTER PARTITION FUNCTION partRang () SPLIT RANGE ('ER');

Pas regjistrimit të vlerave në tabelë, vëmë re se të dhënat janë ruajtur në particionet përkatëse.

```
SELECT partition_number as Numri_particionit, rows as Rreshta
FROM sys.partitions
WHERE OBJECT_ID = OBJECT_ID('seriteBruto')
```

Tabelë 4.2 Ndarja e të dhënave në particione

Numri_particionit	Rreshta
1	16000
2	16000
3	16000
4	16000

Ndërtojmë një tabelë kopje të tabelës seriteBruto të paparticionouar.

```
CREATE TABLE seriteBruto2(
        [id] [int] IDENTITY(1,1) NOT NULL,
        [Variabel] [nvarchar](50) NULL,
        [Vendndodhje] [nvarchar](50) NULL,
        [Data] [datetime] NULL,
        [Vlera] [nvarchar](50) NULL
)
```

Ekzekutojmë kërkesën

```
select max(vlera) from seriteBruto2 where variabel='R'
```

```
select max(vlera) from seriteBruto where variabel='R'
```

Kërkesat ekzekutohet pesë herë, duke ndryshuar sasinë e të dhënave në tabelë. Pas çdo ekzekutimi pastrohet cache-ja.

```
DBCC FREESYSTEMCACHE('ALL')
DBCC FREESESSIONCACHE
DBCC FREEPROCCACHE
```

Rezultatet janë vendosur në tabelën më poshtë. Koha është e llogaritur me milisekonda.

Tabelë 4.3 Koha së ekzekutimit të kërkesave në tabelën e particionuar dhe të paparticionuar

Numri	Tabela		Diferenca
	Particionuar	Paparticionuar	
4000	5	13	8
8000	20	213	193
16000	37	377	340
32000	53	435	382
64000	67	513	446

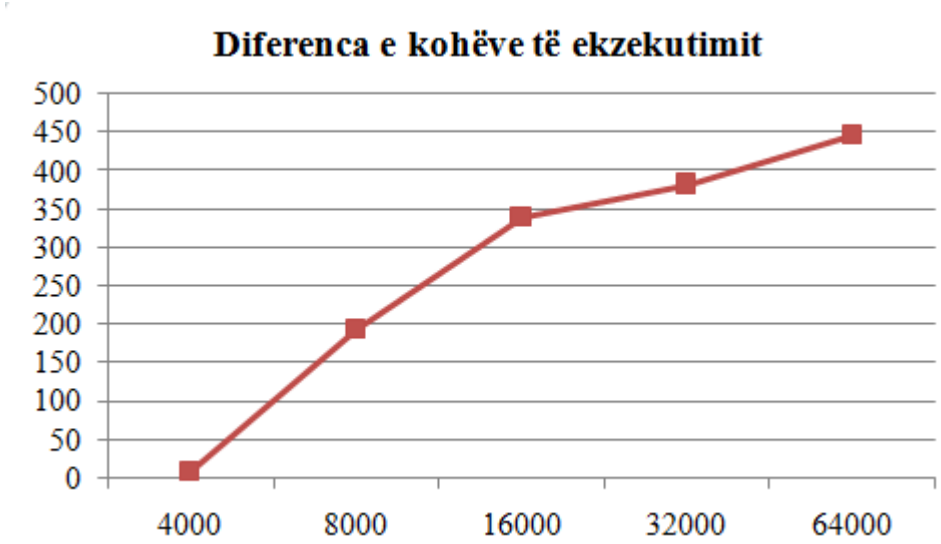


Figura 4.1 Grafiku i diferencës së ekzekutimit të kërkesave midis tabelës së particionuar dhe të paparticionuar

Në grafik vihet re që përfitimi në kohën e ekzekutimit rritet linearisht me sasinë e të dhënave.

Pra në rastin e të dhënave që kemi marrë në shqyrtim, teknika e particionimit luan rol në uljen e kohës së ekzekutimit të kërkesave në bazën e të dhënave, pa prekur shtresën e aplikacionit.

5. Modelimi informatik i metodës së vlerësimit të densiteteve

Algoritmet e mëposhtëm janë ndërtuar bazuar mbi teorinë e propozuar nga Tanku, Ceca(2012). Ata janë ndërtuar për të dy rastet, njëdimensionale dhe shumë dimensionalë për bërthama të ndryshme. Modelimi dhe simulimi i këtyre algoritmeve përbëjnë dhe qëllimin kryesor të këtij punimi.

5.1 Rasti njëdimensional

Për vlerësimin e densiteteve, në rastin një dimensional, Tanku, dhe Ceca(2012) marrin parasysh katër bërthama të paraqitura në tabelën 5.1.

Efiçencat janë llogaritur nga Silverman(1986).

Tabelë 5.1 Bërthama e densiteteve dhe efiçenca e tyre, rasti njëdimensional

	Bërthama	Densiteti	Efiçenca
1.	Gausiane	$K(t)=\frac{1}{\sqrt{2\pi}}e^{-\frac{t^2}{2}}$, $t\epsilon R$	$\sqrt{\frac{36\pi}{125}}\approx 0.9512$
2.	Trekëndore	$K(t)=\begin{cases}1-\lvert t\rvert, \lvert t\rvert<1\\ 0, t\ tjetër\end{cases}$	$\sqrt{\frac{243}{250}}\approx 0.9859$
3.	Rektangulare	$K(t)=\begin{cases}\frac{1}{2}, \lvert t\rvert<1\\ 0, t\ tjetër\end{cases}$	$\sqrt{\frac{108}{125}}\approx 0.9295$
4.	Epanechnikov	$K(t)=\begin{cases}\frac{3}{4\sqrt{5}}(1-\frac{1}{5}t^2), \lvert t\rvert<\sqrt{5}\\ 0, t\ tjetër\end{cases}$	1

Për të llogaritur densitetin e shpërndarjes së një variabli ekonomik të rastit, Tanku dhe Ceca bëjnë supozimet e mëposhtme.

Le të jetë X një variabël i rastit me densitet f(.) për tu vlerësuar. Ne kemi n vlera të variablit, respektivisht $x_1,x_2,...,x_n$

Për të vlerësuar densitetin f(.), Tanku dhe Ceca (2012) përdorin bërthamat që ndodhen në tabelën 5.2. Pas zëvendësimeve dhe transformimeve, Tanku dhe Ceca na japin format përfundimtare:

Tabelë 5.2 Vlerësimi i densiteteve sipas bërthamave, rasti njëdimensional

	Bërthama	Densiteti
1.	Gausiane (normale)	$\forall t\in R, llogarit\ \widehat{f(t)}=\frac{1}{nh}*\frac{1}{\sqrt{2\pi}}\sum_{i=1}^{n} exp\left[\frac{-(t-x_i)^2}{2h^2}\right]$
2.	Trekëndore	$a_i=\begin{cases}h-\lvert t-x_i\rvert, (h-\lvert t-x_i\rvert)>0\\ 0,\ (h-\lvert t-x_i\rvert)\le 0\end{cases}, t\in R$ $\forall t\in R, llogarit\ \widehat{f(t)}=\frac{1}{nh^2}\sum_{i=1}^{n} a_i$

3.	Drejtkëndore	$a_i = \begin{cases} 1, (h - \lvert t - x_i \rvert) > 0 \\ 0, \ (h - \lvert t - x_i \rvert) \leq 0 \end{cases}, t \in R$ $\forall t \in R, llogarit\ \widehat{f(t)} = \frac{1}{2nh} \sum_{i=1}^{n} a_i$
4.	Epanechnikov	$\forall t \in R, nqs\ \frac{\lvert t - x_i \rvert}{h} < \sqrt{5}$ $llogarit\ \widehat{f(t)} = \frac{3}{4\sqrt{5}} \frac{1}{nh} \sum_{i=1}^{n} \left[1 - \frac{1}{5} \left(\frac{t - x_i}{h} \right)^2 \right]$

Më poshtë do japim algoritmet në pseudokod, për vlerësimet e mësipërme.

Vini re që vlerësimet e mësipërme janë për një pikë të boshtit koordinativ. Pra për një pikë *x* ne vlerësojmë densitetin e saj $\hat{f}(x)$. Që densiteti i kësaj pike të jetë pozitiv, duhet që pika të ndodhet midis vlerave ekstreme të matura, pasi përndryshe ngjarja nuk mund të ndodhë dhe densiteti do të jetë zero.

Në algoritmet e ndërtuara më poshtë, vlerat e matjeve te variablit vendosen në një vektor të emërtuar Koeficient.

Gjejmë vlerën më të madhe dhe më të vogël të arritur. Këto vlera do jenë dhe pikat ekstreme të boshtit koordinativ. Përcaktojmë numrin e pikave që duam të vlerësojmë për të ndërtuar grafikun duke përcaktuar largësinë e pikave ne bosht.

Përdorim funksionet e gatshme, të cilat i gjejmë në shumicën e gjuhëve të programimit.

exp(x) = e^x

abs(x) = vlera absolute e *x*

pow(x,z) = x^y

sqrt(x) = $\sqrt{x}$

h është parametri i lëmimit dhe duhet përcaktuar nga studiuesit në varësi të rastit.

Vlerat e arritura vendosen në vektorin produkt, i cili më pas përdoret për të ndërtuar grafikët.

Veprimet më poshtë vlejnë për të katër rastet e bërthamave.

```
dif=largësia e përcaktuar
sasiMatje=numri i matjeve reale të variablit
vlera_max=-9999
vlera_min=9999
matje=lexo matjen e radhës
Per i = 1 deri në sasiMatje  bëj
      Koeficientet[i] = matje
      Nqs matje > vlera_max
        vlera_max=matje
     Nqs matje < vlera_min
        vlera_min=matje
```

matje=lexo matjen e radhës

5.1.1 Bërthama normale

Implentimi i algoritmit:

```
X_vlera= vlera_min
Derisa X_vlera<= vlera_max
    Fx_vlere = 0
    Për i = 1 deri në i <= sasiMatje bëj                //Llogarit F(x)
          Fx_vlere += exp(-1 * (pow((X_Vlera - Koeficientet[i]),2) / (pow(h,2) * 2)))
    Fx_vlere = Fx_vlere / (sasiMatje * h * 2.506631205423)
    produkt[X_vlera]=Fx_vlere
    X_vlera=X_vlera + dif
}
```

5.1.2 Bërthama trekëndore

Implentimi i algoritmit:

```
X_vlera= vlera_min
Derisa X_vlera<= vlera_max
      Fx_vlere = 0
      Për i = 1 deri në i <= sasiMatje bëj                //Llogarit F(x)
        Nqs (abs((X_vlera - Koeficientet[i]) / h) < 1)
            Fx_vlere = Fx_vlere + 1 - abs((X_Vlera - Koeficientet[i]) / h)
       Fx_vlere = Fx_vlere / (sasiMatje * h);
       produkt[X_vlera)]=Fx_vlere
       X_vlera=X_vlera + dif
   }
```

5.1.3 Bërthama drejtkëndore

Implentimi i algoritmit:

```
X_vlera= vlera_min
Derisa X_vlera<= vlera_max
      Fx_vlere = 0
      Për i = 1 deri në i <= sasiMatje bëj                //Llogarit F(x)
        Nqs (abs((X_vlera - Koeficientet[i]) / h) < 1)
            Fx_vlere = Fx_vlere + 0.5
       Fx_vlere = Fx_vlere / (sasiMatje * $h)
       produkt[X_vlera]=Fx_vlere;
       X_vlera=X_vlera + dif
   }
```

5.1.4 Bërthama Epanechnikov

Implentimi i algoritmit:

```
X_vlera= vlera_min
Derisa X_vlera<= vlera_max
      Fx_vlere = 0
      Për i = 1 deri në i <= sasiMatje bëj                //Llogarit F(x)
        Nqs (abs((X_vlera - Koeficientet[i]) / h) < sqrt(5))
            Fx_vlere = Fx_vlere + 1- pow((X_vlera - Koeficientet[i]) / h) , 2)/5
       Fx_vlere = 3*Fx_vlere / (4*sqrt(5)* sasiMatje * h)
       produkt[X_vlera]=Fx_vlere;]
       X_vlera=X_vlera + dif
   }
```

5.2 Rasti d-dimensional

Tanku dhe Ceca ndërtojmë vlerësimin e densiteteve për rastin d-dimensional me madhësi n: $x_1,x_2,...,x_n$ duke përdorur bërthamat d-dimensionale. Në këtë rast densitetet janë në një hapësirë d+1 dimensionale: $(t,\hat{f}(t)) \in R^{d+1}$, $\forall t \in R^d$.

Në paraqitjen grafike të tyre, ne mund të arrijmë deri në rastin kur d=2, pra:

$(t,\hat{f}(t)) \in R^3$, $\forall t \in R^2$

Për vlerësimin e densiteteve, në rastin d-dimensional, Tanku dhe Ceca(2012) marrin parasysh dy bërthama të paraqitura në tabelën 5.3.

Tabelë 5.3 Bërthama e densiteteve, rasti d-dimensional

	Bërthama	Densiteti
1.	Gausiane	$K(\underline{t})=\frac{1}{(2\pi)^{\frac{d}{2}}} e^{-\frac{t't}{2}}$, $\underline{t} \in R^d$, $K(\underline{t}) \in R$
2.	Epanechnikov	$K_e(\underline{t})=\begin{cases}\frac{3}{2Cd}(d+2)(1-t't), & t't<1\\ 0, & t\ tjetër\end{cases}$ C_d është vëllimi i sferës njësi d dimensionale $\underline{t} \in R^d$, $K_e(\underline{t}) \in R$

Për të llogaritur densitetin e shpërndarjes së një variabli ekonomik të rastit, Tanku dhe Ceca bëjnë supozimet e mëposhtme.

Le të jetë $\underline{t}=(t_1,t_2,...,t_d)^1 \in R^d$ një variabël i rastit me densitet f(.) për tu vlerësuar.

Supozojmë se kemi n vlera të variablit d-dimensional $\underline{t}$, respektivisht $\underline{t}_1,\underline{t}_2,...,\underline{t}_n$.

$\underline{t}_1 = (t^1_1,t^2_1,....,t^d_1)$

$\underline{t}_2 = (t^1_2,t^2_2,....,t^d_2)$

.....

$\underline{t}_n = (t^1_n,t^2_n,....,t^d_n)$

Për të vlerësuar densitetin f(.), Tanku dhe Ceca (2012) përdorin bërthamat që ndodhen në tabelën 6.2.1. Pas zëvendësimeve dhe transformimeve, Tanku dhe Ceca na japin format përfundimtare:

Bërthama Normale (Gaussian)

$\forall \underline{t} \in R^d$, llogarit:

$$\hat{f}(\underline{t}) = \frac{1}{nh^d(2\pi)^{d/2}} \cdot \sum_{i=1}^{n} exp\left\{-\frac{1}{2h^2}\left[(t_1 - x_i^1)^2 + (t_2 - x_i^2)^2 + \cdots + \left(t_d - x_i^d\right)^2\right]\right\}.$$

Bërthama Epanechnikov

Llogarit: $C_d = \begin{cases} \frac{\pi^{d/2}}{\left(\frac{d}{2}\right)!}, \ d - çift \\ \frac{2^{\frac{d+1}{2}} \cdot \pi^{\frac{d-1}{2}}}{d!!}, \ d - tek \end{cases}$.

$\forall \underline{t} \in R^d$, llogarit:

$$a_i = \begin{cases} \left\{1 - \frac{1}{h^2}\left[(t_1 - x_i^1)^2 + (t_2 - x_i^2)^2 + \cdots + \left(t_d - x_i^d\right)^2\right]\right\}, \ nqs \ \{\ldots\} > 0 \\ 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad , \ nqs \ \{\ldots\} \leq 0 \end{cases}.$$

Hapi 4: Llogarit: $\hat{f}(\underline{t}) = \frac{1}{nh^d}\frac{d+2}{2 \cdot C_d} \cdot \sum_{i=1}^{n} a_i$.

Pra i bije qe llogarisim:

$$\hat{f}(\underline{t}) = \frac{1}{nh^{d+2}}\frac{d+2}{2 \cdot C_d} \cdot \sum_{i=1}^{n} \left\{ \overbrace{h^2 - \left[(t_1 - x_i^1)^2 + (t_2 - x_i^2)^2 + \cdots + \left(t_d - x_i^d\right)^2\right]}^{\{\ldots\} > 0; \ te \ tjerat \ 0} \right\}$$

Më poshtë do japim algoritmet në pseudokod, për vlerësimet e mësipërme.

Vini re që vlerësimet e mësipërme janë për variabël $\underline{x} \in R^d$. Pra për një pikë $\underline{x} \in R^d$ ne vlerësojmë densitetin e saj $\hat{f}(\underline{x})$. Që densiteti i kësaj pike të jetë pozitiv, duhet që pika të ndodhet midis vlerave ekstreme të matura, pasi përndryshe ngjarja nuk mund të ndodhë dhe densiteti do të jetë zero.

Siç e theksuam dhe më sipër, në paraqitjen grafike ne mund të arrijmë deri në rastin kur d=2. Pra mund të zgjedhim vetëm dy variabla si akse të boshteve koordinativë, të cilët më poshtë po i quajmë variabla segment.

Variablat e tjerë mund të merren vetëm si vlerë fikse.

Në algoritmet e ndërtuara më poshtë, vlerat e matjeve të dy variablave segment vendosen në një matrice të emërtuar Koeficient.

Vlerat e matjeve të variablave fiks vendosen në një matrice të emërtuar Koeficient_Fiks.

Vlerat e matjeve të variablave segment vendosen në një vektor të emërtuar Vlerat.

Gjejmë vlerën më të madhe dhe më të vogël të arritur për matjet e variablave segment. Këto vlera do jenë dhe pikat ekstreme të dy boshteve koordinativ. Përcaktojmë numrin e pikave që duam të vlerësojmë për secilin aks të grafikut duke përcaktuar largësinë e pikave ne bosht.

Përdorim funksionet e gatshme, të cilat i gjejmë në shumicën e gjuhëve të programimit.

exp(x) = e^x

abs(x) = vlera absolute e *x*

pow(x,z) = x^y

sqrt(x) = $\sqrt{x}$

h është parametri i lëmimit dhe duhet përcaktuar nga studiuesit në varësi të rastit.

Vlerat e arritura vendosen në matricën produkt, e cili më pas përdoret për të ndërtuar grafikët.

Veprimet më poshtë vlejnë për të dy rastet e bërthamave.

```
Dif_nje=largësia e përcaktuar për variablin e parë
Dif_dy=largësia e përcaktuar për variablin e dytë
sasiMatje=numri i matjeve reale të variablave
nrVarFix= numri i variablave fiks të përdorur
sasiVarPerdorur = nrVarFix + 2

Për i = 1 deri në nrVarFix bëj
      Var_fiks = variabëli fiks i radhës
      matje=lexo vlerën e Var_fiks
      Vlerat[i] = matje
      Për j = 1 deri në sasiMatje  bëj
         Koeficientet_Fix[i][j] = matje

vlera_max_nje = -9999
vlera_min_nje = 9999
vlera_max_dy = -9999
vlera_min_nje = 9999

Për i = 1 deri në sasiMatje  bëj
       matje=lexo matjen e radhës për variablin e parë segment
       Koeficientet[1][i] = matje
       Nqs matje > vlera_max_nje
         vlera_max_nje=matje
       Nqs matje < vlera_min_nje
         vlera_min_nje =matje
       matje=lexo matjen e radhës për variablin e dytë segment
       Koeficientet[2][i] = matje
       Nqs matje > vlera_max_dy
         vlera_max_dy =matje
       Nqs matje < vlera_min_dy
         vlera_min_dy =matje
```

5.2.1 Bërthama Normale (Gaussian)

Implentimi i algoritmit:

```
X1_vlera= vlera_min_nje
Derisa X1_vlera <= vlera_max_nje
        X2_vlera= vlera_min_dy
        Derisa X2_vlera<= vlera_max_dy
                Fx_vlere = 0
                Për i = 1 deri  sasiMatje
                   a=pow((X1_vlera - Koeficientet[1][i]),2)+pow((X2_vlera - Koeficientet[2][i]),2)
                   shumaVarFix=0
                   Për j = 1 deri  nrVarFix
                      shumaVarFix +=pow((vlera[j] – Koeficientet_Fix[j][i]),2)
                   a=a + shumaVarFix
                   Fx_vlere = Fx_vlere + exp(-1 * a /  (pow(h,2)*2))
                Fx_vlere = Fx_vlere / (sasiMatje * pow(h,sasiVarPerdorur) * pow(2*π,sasiVarPerdorur/2))
                produkt[X1_vlera][X2_vlera]=Fx_vlere
                X2_vlera=X2_vlera + Dif_dy
X1_vlera=X1_vlera + Dif_nje
```

5.2.2 Bërthama Epanechnikov

Për llogaritjen e vlerave të densitetit në rastin e bërthamës Epanechnikov, përdorim funksionet e ndërtuara më poshtë.

```
funksion faktorial(numri)
        Nqs (numri==0 ose $numri==1)
            kthe 1
        kthe numri*faktorial(numri-1);
```

```
funksion Cd(d)
        if(d%2==0)
        kthe pow(π, d/2)/faktorial(d/2);
        kthe pow(2,( d +1)/2)*pow(π,( d -1)/2)/ faktorial(d)
```

Implentimi i algoritmit.

```
X1_vlera= vlera_min_nje
Derisa X1_vlera <= vlera_max_nje
        X2_vlera= vlera_min_dy
        Derisa X2_vlera<= vlera_max_dy
                Fx_vlere = 0
                Për i = 1 deri  sasiMatje
                   a=pow((X1_vlera - Koeficientet[1][i]),2)+pow((X2_vlera -
Koeficientet[2][i]),2)
                   shumaVarFix=0
                   Për j = 1 deri  nrVarFix
                      shumaVarFix +=pow((vlera[j] – Koeficientet_Fix[j][i]),2)
                   a=a + shumaVarFix
                   a=a/pow(h,2)
                   Nqs (a<1)
                         Fx_vlere = Fx_vlere + (1 - a)
                Fx_Vlere = (sasiVarPerdorur +2)*Fx_vlere / (sasiMatje *
pow(h,sasiVarPerdorur) *2*Cd(sasiVarPerdorur))
                produkt[X1_vlera][X2_vlera]=Fx_vlere
                X2_vlera=X2_vlera + Dif_dy;
        X1_vlera=X1_vlera + Dif_nje
```

5.3 Rasti d-dimensional , me kusht m-dimensional

Tanku dhe Ceca ndërtojmë vlerësimin e densiteteve për rastin d-dimensional me kusht m-dimensional me madhësi n: $x_1,x_2,...,x_n$ duke përdorur bërthamat d-dimensionale. Edhe në këtë rast densitetet janë në një hapësirë d+1 dimensionale:

$(t,\hat{f}(t)) \in R^{d+1}, \forall t \in R^d$.

Në paraqitjen grafike të tyre, ne mund të arrijmë deri në rastin kur d=2, pra:

$(t,\hat{f}(t)) \in R^3, \forall t \in R^2$

Për të llogaritur densitetin e shpërndarjes së një variabli ekonomik të rastit, Tanku dhe Ceca bëjnë supozimet e mëposhtme.

Le të jetë $\underline{t}=(t_1,t_2,...,t_d)^1 \in R^d$ një variabël i rastit me densitet f(.) për tu vlerësuar.

Supozojmë se kemi n vlera të variablit d-dimensional $\underline{t}$, respektivisht $\underline{t}_1,\underline{t}_2,...,\underline{t}_n$.

$\underline{t}_1 = (t^1_1,t^2_1,....,t^d_1)$

$\underline{t}_2 = (t^1_2,t^2_2,....,t^d_2)$

.....

$\underline{t}_n = (t^1_n,t^2_n,....,t^d_n)$

Për të vlerësuar densitetin f(.) të kushtëzuar, Tanku dhe Ceca (2014) përdorin bërthamat Gausiane dhe Epanecnikov, format analitike të të cilave për rastin e kushtëzimeve m-dimensionale janë paraqitur më poshtë.

Bërthama Normale (Gaussian)

Hap 1: Fikso m parametra $k+1, k+2, \dots, k+m \in \{1, \dots, d\}$ (fikson m variabla ekonomike),

ku $1 \le m \le d-1$.

Hap 2: Fikso një vlerë për secilën $t_{k+1}, t_{k+2},\dots,t_{k+m}$. (fikson vlerat e variablave ekonomike)

Hap 3: Fikso një parameter $h > 0$.

Hap 4: $\forall \underline{t} = (t_1, \dots, t_{k+1}, \dots, t_{k+m}, \dots, t_d) \in R^d$(ku $t_{k+1}, t_{k+2},\dots,t_{k+m}$ janë të fiksuara) llogarit:

$$\hat{f}_{\{1,\dots,k,k+m+1,\dots,d\}}(t_1, \dots, t_k, t_{k+m+1}, \dots, t_d)$$
$$= \frac{1}{h^{d-m}(2\pi)^{\frac{d-m}{2}}}$$
$$\cdot \frac{\sum_{i=1}^{n} exp\left\{-\frac{1}{2h^2}\left[(t_1 - x_i^1)^2 + \dots + \overbrace{(t_{k+1} - x_i^{k+1})^2 + \dots + (t_{k+m} - x_i^{k+m})^2}^{m \text{ copë}} \dots + (t_d - x_i^d)^2\right]\right\}}{\sum_{i=1}^{n} exp\left\{-\frac{1}{2h^2}\left[\underbrace{(t_{k+1} - x_i^{k+1})^2 + \dots + (t_{k+m} - x_i^{k+m})^2}_{m \text{ copë}}\right]\right\}}$$

Bërthama Epanechnikov

Hap 1: Fikso m parametra $k+1, k+2, \dots, k+m \in \{1, \dots, d\}$ (fikson m variabla ekonomike),

ku 1 ≤ m ≤ d-1.

Hap 2: Fikso një vlerë për secilën $t_{k+1}, t_{k+2}, \dots, t_{k+m}$. (fikson vlerat e variablave ekonomike)

Hap 3: Fikso një parameter $h > 0$.

Hap 4: $\forall \underline{t} = (t_1, \dots, t_{k+1}, \dots, t_{k+m}, \dots, t_d) \in R^d$(ku $t_{k+1}, t_{k+2}, \dots, t_{k+m}$ janë të fiksuara) llogarit:

$$\hat{f}_{\{1,\dots,k,k+m+1,\dots,d\}}(t_1, \dots, t_k, t_{k+m+1}, \dots, t_d) =$$

$$= \frac{\Gamma\left(1+\frac{d-m}{2}\right)}{h^{d-m} \cdot \pi^{\frac{d-m}{2}}} \cdot$$

$$\cdot \frac{\sum_{i=1}^{n} \left\{ \overbrace{h^2 - \left[\underbrace{(t_1 - x_i^1)^2 + \dots + \overbrace{\left(t_{k+1} - x_i^{k+1}\right)^2 + \dots + \left(t_{k+m} - x_i^{k+m}\right)^2}^{m \text{ copë}} \dots + \left(t_d - x_i^d\right)^2}_{d \text{ copë}} \right]}^{\{\dots\} > 0; \ \text{te tjerat } 0} \right\}}{\frac{2nh^2}{d-m+2} - \sum_{i=1}^{n} \left[\underbrace{\left(t_{k+1} - x_i^{k+1}\right)^2 + \dots + \left(t_{k+m} - x_i^{k+m}\right)^2}_{m \text{ copë}} \right]}$$

Llogaritja e $\Gamma\left(1+\frac{d-m}{2}\right)$

Dihet qe: Γ(1) = 1, Γ(n) = (n-1)!, dheΓ $\left(\frac{1}{2}+n\right) = \frac{(2n)!}{4^n n!}\sqrt{\pi}$.

$$\Gamma\left(1+\frac{d-m}{2}\right) = \begin{cases} \left(\frac{d-m}{2}\right)! & (d-m) \ cift \\ \frac{(1+d-m)!}{4^{\frac{1+d-m}{2}} \cdot \left(\frac{1+d-m}{2}\right)!}\sqrt{\pi} & (d-m) \quad tek \end{cases}$$

Më poshtë do japim algoritmet në pseudokod, për vlerësimet e mësipërme.

Vini re që vlerësimet e mësipërme janë për variabël $\underline{x} \in R^d$. Pra për një pikë $\underline{x} \in R^d$ ne vlerësojmë densitetin e saj $\hat{f}(\underline{x})$. Që densiteti i kësaj pike të jetë pozitiv, duhet që pika të ndodhet midis vlerave ekstreme të matura, pasi përndryshe ngjarja nuk mund të ndodhë dhe densiteti do të jetë zero.

Siç e theksuam dhe më sipër, në paraqitjen grafike ne mund të arrijmë deri në rastin kur d=2. Pra mund të zgjedhim vetëm dy variabla si akse të boshteve koordinativë, të cilët më poshtë po i quajmë variabla segment.

Variablat e tjerë mund të merren vetëm si vlerë fikse.

Në algoritmet e ndërtuara më poshtë, vlerat e matjeve të dy variablave segment vendosen në një matrice të emërtuar Koeficient.

Vlerat e matjeve të variablave fiks vendosen në një matrice të emërtuar Koeficient_Fiks.

Vlerat e matjeve të variablave segment vendosen në një vektor të emërtuar Vlerat.

Gjejmë vlerën më të madhe dhe më të vogël të arritur për matjet e variablave segment. Këto vlera do jenë dhe pikat ekstreme të dy boshteve koordinativ. Përcaktojmë numrin e pikave që duam të vlerësojmë për secilin aks të grafikut duke përcaktuar largësinë e pikave ne bosht.

Përdorim funksionet e gatshme, të cilat i gjejmë në shumicën e gjuhëve të programimit.

exp(x) = e^x

abs(x) = vlera absolute e x

pow(x,z) = x^y

sqrt(x) = $\sqrt{x}$

h është parametri i lëmimit dhe duhet përcaktuar nga studiuesit në varësi të rastit.

Vlerat e arritura vendosen në matricën produkt, e cili më pas përdoret për të ndërtuar grafikët.

Veprimet më poshtë vlejnë për të dy rastet e bërthamave.

```
Dif_nje=largësia e përcaktuar për variablin e parë
Dif_dy=largësia e përcaktuar për variablin e dytë
sasiMatje=numri i matjeve reale të variablave
nrVarFix= numri i variablave fiks të përdorur
sasiVarPerdorur = nrVarFix + 2

Për i = 1 deri në nrVarFix bëj
    Var_fiks = variabëli fiks i radhës
    matje=lexo vlerën e Var_fiks
    Vlerat[i] = matje
    Për j = 1 deri në sasiMatje  bëj
       Koeficientet_Fix[i][j] = matje

vlera_max_nje = -9999
vlera_min_nje = 9999
vlera_max_dy = -9999
vlera_min_nje = 9999

Për i = 1 deri në sasiMatje  bëj
      matje=lexo matjen e radhës për variablin e parë segment
      Koeficientet[1][i] = matje
      Nqs matje > vlera_max_nje
```

```
  vlera_max_nje=matje
Nqs matje < vlera_min_nje
  vlera_min_nje =matje
matje=lexo matjen e radhës për variablin e dytë segment
Koeficientet[2][i] = matje
Nqs matje > vlera_max_dy
  vlera_max_dy =matje
Nqs matje < vlera_min_dy
  vlera_min_dy =matje
```

5.3.1 Bërthama Normale (Gaussian)

Implentimi i algoritmit:

```
//do gjejmë emëruesin që është i përbashkët për të gjithë
Për i = 1 deri  sasiMatje
        shumaVarFix[i]=0
        Për j = 1 deri  nrVarFix
                shumaVarFix[i]+=pow((vlera[j] – Koeficientet_Fix[j][i]),2)
X1_vlera= vlera_min_nje
Derisa X1_vlera <= vlera_max_nje
        X2_vlera= vlera_min_dy
        Derisa X2_vlera<= vlera_max_dy
                Fx_vlere = 0
                emeruesiFix=0
                Për i = 1 deri  sasiMatje
                   a=pow((X1_vlera - Koeficientet[1][i]),2)
                   b=pow((X2_vlera - Koeficientet[2][i]),2)
                   numerues=a+ shumaVarFix[i]+b
                   Fx_vlere = Fx_vlere + exp(-1 * numerues /  (pow(h,2)*2))
                   Nqs variabelKusht=ipari
                        emeruesiFix += exp(-1 * (a+ shumaVarFix [i])/(pow(h,2)*2))
                   PRN
                        emeruesiFix += exp(-1 * (b+ shumaVarFix [i])/(pow(h,2)*2))
```

```
        Fx_vlere = Fx_vlere / (emeruesiFix * pow(h,sasiVarPerdorur-1-
nrVarFix) * pow(2*π,(sasiVarPerdorur - 1 – nrVarFix) /2))

        produkt[X1_vlera][X2_vlera]=Fx_vlere

        X2_vlera=X2_vlera + Dif_dy

    X1_vlera=X1_vlera + Dif_nje
```

5.3.2 Bërthama Epanechnikov

Për llogaritjen e vlerave të densitetit të kushtëzuar në rastin e bërthamës Epanechnikov, përdorim funksionet e ndërtuara më poshtë.

```
funksion faktorial(numri)

    Nqs (numri==0 ose numri==1)

        kthe 1

    kthe numri*faktorial(numri-1);


funksion theta(d,m){

    Nqs ((d-m)%2==0)

        kthe faktorial((d-m)/2)

    kthe sqrt(π)*(d-m)*faktorial(1+d-m)/(pow(4,(1+d -m)/2)*faktorial(1+d-m))
```

Implentimi i algoritmit.

```
X1_vlera= vlera_min_nje

Derisa X1_vlera <= vlera_max_nje

    X2_vlera= vlera_min_dy

    Derisa X2_vlera<= vlera_max_dy

        Fx_vlere = 0

        shumaEmerues=0

        Për i = 1 deri  sasiMatje

            a=pow((X1_vlera - Koeficientet[1][i]),2)+pow((X2_vlera -
Koeficientet[2][i]),2)

            shumaVarFix=0

            Për j = 1 deri  nrVarFix
```

```
                    shumaVarFix +=pow((vlera[j] – Koeficientet_Fix[j][i]),2)
                a=a + shumaVarFix
                Nqs (a< pow(h,2))
                    Fx_vlere = Fx_vlere + (pow(h,2) - a)
                    shumaEmerues=shumaEmerues+shumaVarFix
            emeruesiFix=2*sasiVarPerdorur*pow($h,2)/(sasiVarPerdorur-
nrVarFix +2)-shumaEmerues
            Fx_Vlere = theta(sasiVarPerdorur, nrVarFix)*Fx_vlere /
(pow(h,sasiVarPerdorur- nrVarFix)*pow(π,(sasiVarPerdorur- nrVarFix)/2)*
emeruesiFix)
            produkt[X1_vlera][X2_vlera]=Fx_vlere
            X2_vlera=X2_vlera + Dif_dy;
        X1_vlera=X1_vlera + Dif_nje
```

5.4 Ndërtimi i algoritmeve për momentet

Pasi ndërtojmë vlerësimin e densiteteve për rastin d-dimensional me kusht m-dimensional me madhësi n: $x_1,x_2,...,x_n$ duke përdorur bërthamat d-dimensionale, ne mund të përdorim vlerat e këtyre densiteteve për të dhënë vlerësime empirike mbi pritjen matematike dhe dispersionin.

Vlerësimi empirik i pritjes:

Një vlerësim i përafërt i $E(X) = \int_a^b xf(x)dx$, jepet nga përafrimi i grafikut të funksionit xf(x), si më poshtë.

$$E(X) \approx \sum_{i=1}^{n-1} \left[\frac{x_i f(x_i)+x_{i+1}f(x_{i+1})}{2}(x_{i+1} - x_i)\right]$$

Vleresimi empirik i dispersionit:

Një vlerësim i përafërt i $D(X) = \int_a^b [x - E(X)]^2 f(x)dx$, jepet nga përafrimi i grafikut të funksionit [x – E(X)]2f(x), si më poshtë.

$$\sigma^2 = D(X) \approx \sum_{i=1}^{n-1} \left[\frac{(x_i-EX)^2 f(x_i)+(x_{i+1}-EX)^2 f(x_{i+1})}{2}(x_{i+1} - x_i)\right]$$

Algoritmi i grafikut te pritjes dhe dispersionit sipas Tanku dhe Ceca:

Ndërtohet grafiku (t1, t2) → f(t1, t2).

Kur bëhet prerja sipas t2 = a, për prerjen (t1, a) → f(t1, a),

llogaritet E{f(t1, a)}, σ^2 = D{f(t1, a)}

Implentimi i algoritmit në rastin kur gjejmë pritjen e variablit të dytë.

Përdorim funksionet e gatshme, të cilat i gjejmë në shumicën e gjuhëve të programimit.

pow(x,z) = x^y

sqrt(x) = $\sqrt{x}$

Vlerat e arritura vendosen në vektoret Pritja, Varianca, ariancaPlus, variancaMinus, të cilët më pas përdoren për të ndërtuar grafikët.

Dif_nje=largësia e përcaktuar për variablin e parë
Dif_dy=largësia e përcaktuar për variablin e dytë
sasiMatje=numri i matjeve reale të variablave

vlera_max_nje = -9999
vlera_min_nje = 9999
vlera_max_dy = -9999
vlera_min_nje = 9999

Për i = 1 deri në sasiMatje bëj
matje=lexo matjen e radhës për variablin e parë segment
Nqs matje > vlera_max_nje
vlera_max_nje=matje
Nqs matje < vlera_min_nje
vlera_min_nje =matje
matje=lexo matjen e radhës për variablin e dytë segment
Nqs matje > vlera_max_dy
vlera_max_dy =matje
Nqs matje < vlera_min_dy
vlera_min_dy =matje

Matrica produkt është ndërtuar duke shfrytëzuar algoritmet e vlerësimeve të densiteteve d dimensionale me kusht m dimensional.

```
X1_vlera= vlera_min_nje

Derisa X1_vlera <= vlera_max_nje

        E=0

        X2_vlera= vlera_min_dy

        Derisa X2_vlera< vlera_max_dy

                E=E+(( X2_vlera* produkt[X1_vlera][X2_vlera]+(X2_vlera+dif_dy)*
produkt[X1_vlera][X2_vlera+dif_dy])/2)* dif_dy

                X2_vlera=X2_vlera + Dif_dy;

        Pritja[X1_vlera] = E
```

```
        D=0

        Derisa X2_vlera< vlera_max_dy

                D=D+ ((pow(X2_vlera- Pritja[X1_vlera],2) *
produkt[X1_vlera][X2_vlera] + pow((X2_vlera+dif_dy)- Pritja[X1_vlera],2)*
produkt[X1_vlera][ X2_vlera+dif_dy])/2) *  dif_dy

                X2_vlera=X2_vlera + Dif_dy;

        Varianca[X1_vlera] = D

        X1_vlera=X1_vlera + Dif_nje

        variancaPlus[i]=Pritja[i]+2*sqrt(Varianca[i]);

        variancaMinus[i]=Pritja[i]-2*sqrt(Varianca[i]);
```

6. Paralelizimi

Kapitulli i mëparshëm jep algoritmet për gjenerimin e matrices së densiteteve të vlerësuara. Algoritmet marrin si parameter vlerën minimale dhe maksimale të vlerave në boshtet koordinativë, si dhe largësinë e pikave nga njëra tjetra. Algoritmet gjejnë vlerën e *f(.)* për çdo dy pika. Pra numri vlerave të funksionit të përllogaritura do të jetë:

((Vlera_max_nje-vlera_min_nje)/dif_nje) * ((Vlera_max_dy-vlera_min_dy)/dif_dy)

Vëmë re që sa më të vogla të jenë vlerat e dif_nje dhe dif_dy, aq më shumë rritet numri i vlerave që duam të llogarisim. Ky numër mund të jetë sado i madh. Nga ana tjetër, sa më të imta të jenë këto vlera, aq më i plotë dhe i kuptueshëm do jetë grafiku rezultat.

Duke qenë se përllogaritjet e këtyre janë të pavaruara nga njëra tjetra, mund ti ekzekutojme në paralel.

Programimi në paralel është një model programimi i dizenjuar për procesimin e të dhënave me volum të madh duke e ndarë punën në një bashkësi punësh më të vogla dhe të pavarura. Ky modul shpjegon natyrën e ketij modeli programimi dhe se si ai mund të përdoret për të shkruar programe të cilat ekzekutohen në paralel.

Më poshtë po propozojme disa mënyra për të zbatuar parimin e paralelizimit.

Thread

Thread-et janë pjesë e një procesi dhe zakonisht ndajnë të njëjtën memorie dhe burime. Për të përshpejtuar ekzekutimin e punëve, ne ndajmë punën midis thread-eve të ndryshme, ku secili prej tyre kryen një punë më të vogël. Pra në një sistem me shumë procesorë apo me disa bërthama procesorësh, do kemi ndarje të punëve që duhen kryer. Pra punët mund të kryhen njëkohësisht dhe jo njëra pas tjetrës. Punët nuk do të bllokojnë njëra tjetrën, pasi në PHP memorja nuk ndahet.

Zgjidhja e ofruar nga PHP për multithreading është klasa pthreads e saj:

```
class ChildThread extends Thread {
        public $data;
        public function run() {
                /* algoritmi */
                $this->data = ‘rezultati i punës’;
        }
}
```

```
$thread = new ChildThread();

if ($thread->start()) {

        /* nënproçedura */

        // pret derisa thread të përfundojë

        $thread->join();

}
```

Kodi më sipër ndan punën në dy thread-e. Pas përfundimit të punës të thread të dytë të krijuar, rezultatet i tij i kalojnë procesit prind për të përpunuar më tej.

Eksperimenti

Testimet janë janë ekzekutuar në serverin e përdorur për particionimin (shih kapitull V, tabela 5.1)

Është krijuar skedari i të dhënave me 200 matje për secilin prej variablave ekonomikë CPI, M1, M2, LEK/EURO, LEK/USD.

Rasti 1

Algoritmi fillimisht është ekzekutuar i paparalelizuar dhe më pas i paralelizuar. Testimi është përsëritur duke rritur numrin pikave në boshtet koordinativë. Kemi thirrur threade për përllogaritjen e densitetit për çdo dyshe pikash (x,y). Pra numri i thread-eve të përdorur është i barabartë me numrin e pikave.

Vihet re që paralelizimi me këtë metodë sjell një rritje të kohës së ekzekutimit, e cila përkeqësohet në përpjesëtim të drejtë me rritjen e numrit të pikave, të cilave do u përllogaritet densiteti.

Rezultatet janë vendosur në tabelën më poshtë. Koha është e llogaritur me sekonda.

Tabelë 6.1 Koha e ekzekutimit të algoritmit të paparalelizuar dhe të paralelizuar ku numri i thread sa numri i pikave.

	Koha në sekonda	
Numri pikave	**Paralelizuar**	**Paparalelizuar**
800	1.9	0.4
20000	41	8.5
800000	165	34
4000000	1073	168
20000000	4560	845

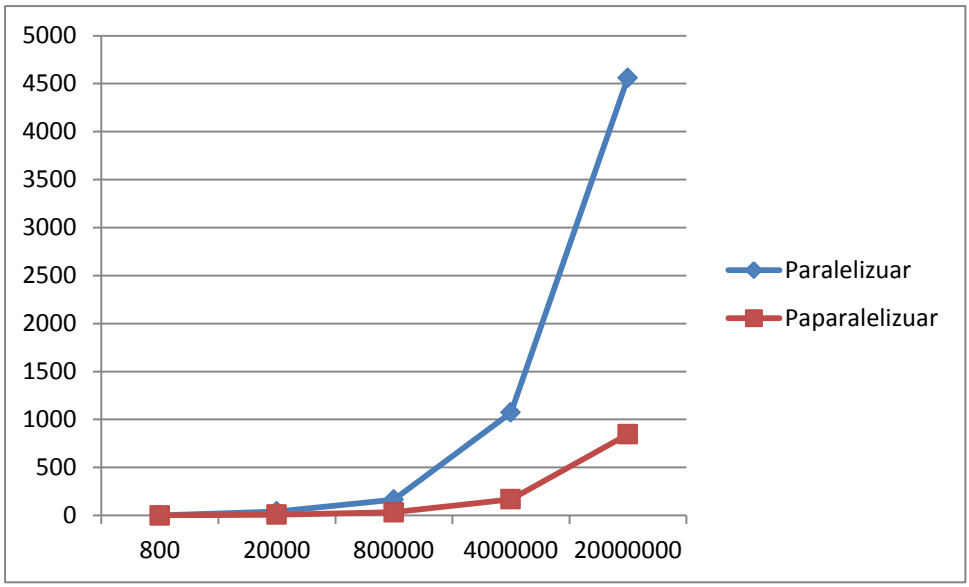


Figura 6.1 Grafiku i së kohës së ekzekutimit midis algoritmit të paparalelizuar dhe të paralelizuar ku numri i thread sa numri i pikave

Rasti 2

Algoritmi fillimisht është ekzekutuar i paparalelizuar dhe më pas i paralelizuar. Testimi është përsëritur duke rritur numrin pikave në boshtet koordinativë. Numri i thread-eve të përdorur është 4.

Rezultatet janë vendosur në tabelën më poshtë. Koha është e llogaritur me sekonda.

Tabelë 6.2 Koha e ekzekutimit të algoritmit të paparalelizuar dhe të paralelizuar.

Numri pikave	**Koha në sekonda**		**Diferenca**
	Paralelizuar	**Paparalelizuar**	
800	0.3	0.4	0.1
20000	5.9	8.5	2.6
800000	23	34	11
4000000	110	168	58
20000000	563	845	282

Vihet re që paralelizimi me këtë metodë sjell një ulje të kohës së ekzekutimit, e cila përmirësohet në përpjesëtim të drejtë me rritjen e numrit të pikave, të cilave do u përllogaritet densiteti.

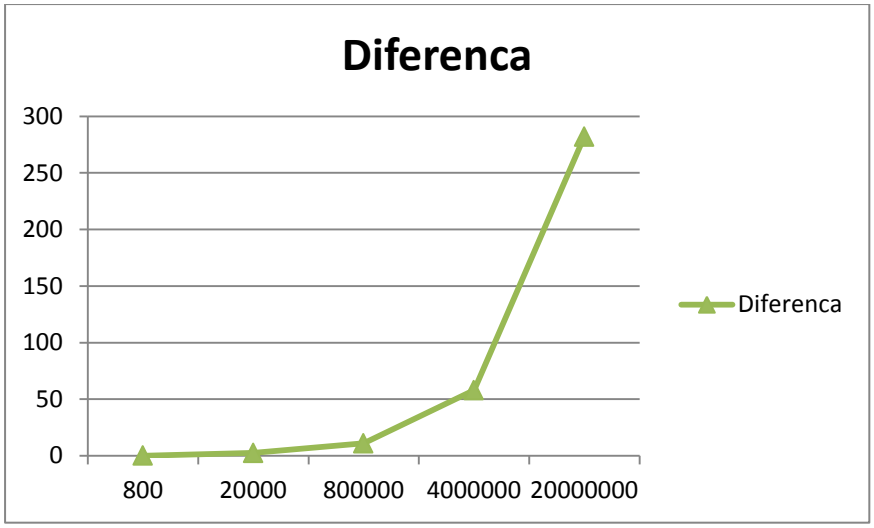


Figura 6.2 Grafiku i diferencës së kohës së ekzekutimit midis algoritmit të paparalelizuar dhe të paralelizuar.

Në grafik vihet re që përfitimi në kohën e ekzekutimit rritet me rritjen e numrit të pikave për llogaritje.
Pra në rastin e të dhënave që kemi marrë në shqyrtim, teknika e paralelizimit luan rol të ndjeshëm në uljen e kohës së ekzekutimit.

Paralelizimi në metodën e paraqitur

Duke qenë që grafiku ndërtohet nga një bashkësi pikash (x,y), paralelizimi i propozuar është pikërisht ndarja e përllogaritjeve të f(.) për grupe pikash.

Eksperimenti më sipër tregoi që krijimi i një thread për llogaritje të densitetit për çdo pike (x,y) nuk rekomandohet, sepse në këtë rast koha e nevojshme për ekzekutimin e thread është më e madhe se koha e nevojshme për llogaritjen e vlerës së densitetit.

Rekomandohet të bëhet paralelizim i përllogaritjes së grupeve të pikave.

Fillimisht bëhet leximi i vlerave nga skedarët në memorie. Kjo na mundëson elimimin e veprimeve I/O me diskun, duke përjashtuar pritjen për operacione të ndryshme. Meqënëse aplikimi mund të klasifikohet në aplikim i fokusuar në cikle përllogaritjesh, rekomandohet që numri i thread-eve të jetë:

N_thread =Numër procesorësh * numër bërthamash për procesor (6.1)

Algoritmet do ndahen në dy procese. Procesi prind bën ndarjen e boshtit të abshisave në segmente të barabarta. Numri i segmenteve do të jetë numri i thread-eve që do përdoren. Më pas, procesi do thërrasë thread-et bij. Ekzekutimi i tyre kryhet në paralel. Secili prej tyre do të kthejë një nënmatricë rezultat.

Procesi prind bën bashkimin e rezultateve në një matricë përfundimtare.

Procesi prind

Vihet re që procesi prind është i njëjtë në të gjitha rastet.

```
$nrthread=floor((($X1_Vlera_Max-$X1_Vlera_Min)/$diff1)/N_thread);
$min=$X1_Vlera_Min;
$max=$X1_Vlera_Min+$nrthread*$diff1;
$i=1;
 while($max<=$X1_Vlera_Max && $min<=$X1_Vlera_Max){
          $thread[$i] = new AsyncOperation(parametra);
          $thread[$i]->start();
          $min=$max+$diff1;
          if(($min+$nrthread*$diff1)<=$X1_Vlera_Max)
               $max=$max+$nrthread*$diff1;
          else
             $max=$X1_Vlera_Max;
           $i++;
}
$min=$X1_Vlera_Min;
 $max=$X1_Vlera_Min+$nrthread*$diff1;
$i=1;
 while($max<=$X1_Vlera_Max && $min<=$X1_Vlera_Max){
    $thread[$i]->join();
    $rez=$thread[$i]->getValue();
   $X1_Vlera=$min;
   while ($X1_Vlera<=$max){
        $X2_Vlera=$X2_Vlera_Min;
        while ($X2_Vlera<=$X2_Vlera_Max){
$product[$X1_Vlera][$X2_Vlera]=$rez[$X1_Vlera][$X2_Vlera];
               $X2_Vlera=$X2_Vlera + $diff2;
       }
      $X1_Vlera=$X1_Vlera + $diff1;
    }
   $min=$max+$diff1;
   if(($min+$nrthread*$diff1)<=$X1_Vlera_Max)
      $max=$max+$nrthread*$diff1;
   else
      $max=$X1_Vlera_Max;
   $i++;
 }
```

Procesi bir:

Procesi bir është i ndryshëm për bërthama të ndryshme.

Më poshtë po japin algoritmin e ekzekutimit në rastin e bërthamës d dimensionale Gaussiane. Për bërthamat e tjera mjafton të ndryshojme procedurën e brendshme të procesit bir.

```
$X1_Vlera=$this->X1_Vlera_Min;
while ($X1_Vlera<=$this->X1_Vlera_Max){
        $X2_Vlera=$this->X2_Vlera_Min;
        while ($X2_Vlera<=$this->X2_Vlera_Max){
                $Fx_Vlere = 0;
                for ($i = 2; $i <= $nr_matje; $i++)  {
                 $a=pow(($X1_Vlera - $Koeficientet[1][$i]),2) +pow(($X2_Vlera -
$Koeficientet[2][$i]),2);
                 $Fx_Vlere = $Fx_Vlere + exp(-1 * $a / (pow($h,2)*2));
                 }
                $Fx_Vlere = $Fx_Vlere / ($nr_matje * pow($h,2) *
pow(2*3.14128,2/2));
               $R[$X1_Vlera][$X2_Vlera]=$Fx_Vlere;
               $X2_Vlera=$X2_Vlera + $this->diff2;
        }
         $X1_Vlera=$X1_Vlera + $this->diff1;
   }
 $this->llogaritjet=$R;
```

7. Implementimi i cloud computing

Cloud computing është një paradigmë e re e cila na mundëson kujtesë e përpunim me kosto më të lirë. Kjo metodë lejon përdoruesit të përdorin makina me kujtesë të madhe dhe me fuqi përpunuese të lartë, pa pasur nevojë të blejnë pajisje të tilla, po duke huazuar ato prej kompanive të mëdha.

Cloud mund të shihet si një rrjet nyjesh të vendosura në nivelin server.

Klientët mund të aksesojnë aplikimin dhe të dhënat e tyre në cdo kohë, në çdo vend të botës. Ata mund të aksesojnë sistemin cloud computing duke përdorur një kompjuter të thjeshtë, mjafton të jenë të lidhur me internet.

Cloud computing na jep mundësinë dhe opsionin e ruajtjes së të dhënave në hardware-in e dikujt tjeter, duke na lehtësuar nga nevoja e gjetjes së hapësirës fizike në anën klient.

Nëse pjesa back end e sistemit cloud computing është një sistem grid computing, atëherë klienti do të mund të shfrytëzonte të gjithë fuqinë e procesimit të të gjithë rrjetit cloud computing. Algoritmet e studiuara më sipër kërkojnë llogaritje të shumta e komplekse. Në një sistem grid computing, klienti e dërgon problemin për procesim në cloud. Sistemi cloud do të përdorë fuqinë procesuese të të gjithë komputerave në rrjet për zgjidhjen e problemit, duke rritur kështu shpejtësinë e përllogaritjes.

7.1 Map/Reduce

Modeli i propozuar për simulimin e teorisë është MapReduce. Ky model është një model programimi i ndërtuar për procesimin e të dhënave të mëdha duke realizuar përpunimin e tyre në paralel. Ky model është i përshtatshëm për rastin tonë pasi paralelizimi i ofruar nuk kërkon shkëmbim të informacionit midis nënpunëve të krijuara.

Platforma e përdorur është Apache Hadoop [23]. Ajo ka një tolerance të lartë ndaj gabimeve. Platforma mundëson rifillimin e nënpunëve të përfunduara me dështim.

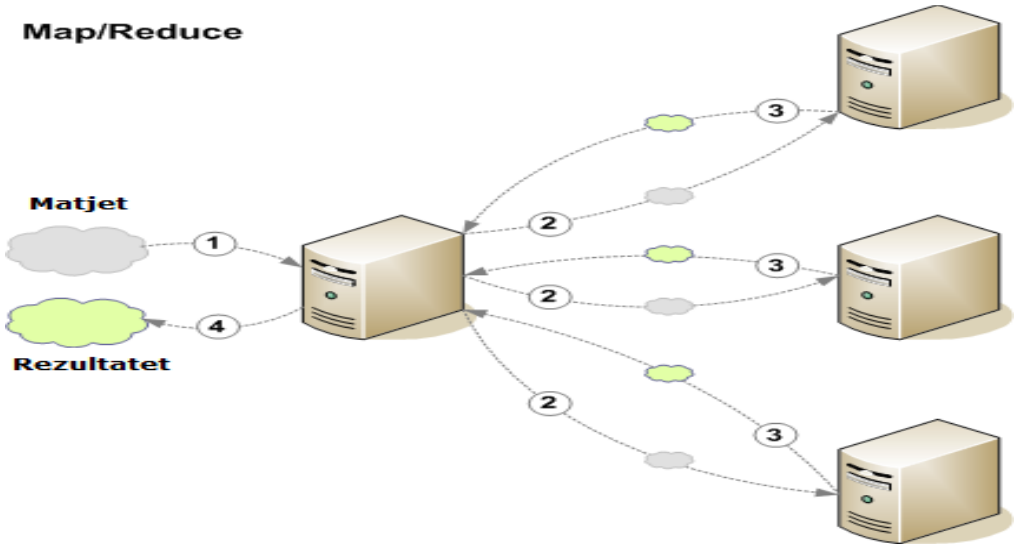


Figura 7.1 Ndarja e punëve në Map/Reduce

Si shihet në figurën më sipër, ne japim një set me të dhëna për tu përpunuar, dhe është vetë platforma e cila i dërgon këto të dhëna për përpunim në makina të ndryshme.

Kjo paradigmë kërkon që të dhënat hyrëse të kenë një format standard. Të dhënat do të ruhen në skedarë, ku secili rresht do jetë një dyshe (çelës-vlerë). Për këtë duhet të bëjmë transformimin e të dhënave tona në formatin e ri.

Pasi të dhënat kalojnë formën e re, të dhënat do të ndahen në skedarë të ndryshëm, të cilët i jepen për përpunim makinave të ndryshme. Kjo përbën dhe fazën e parë të paralelizmit

Programet MapReduce, marrin një listë me të dhëna hyrëse dhe i transformojnë ato në të dhëna rezultat. Kjo kryhet përmes dy idiomave, map dhe reduce.

Gjatë fazës Map, për çdo dyshe çelës-vlerë, ne ekzekutojmë funksionin e duhur dhe i transformojmë të dhënat në çiftet e reja (çelës-vlerë rezultat).

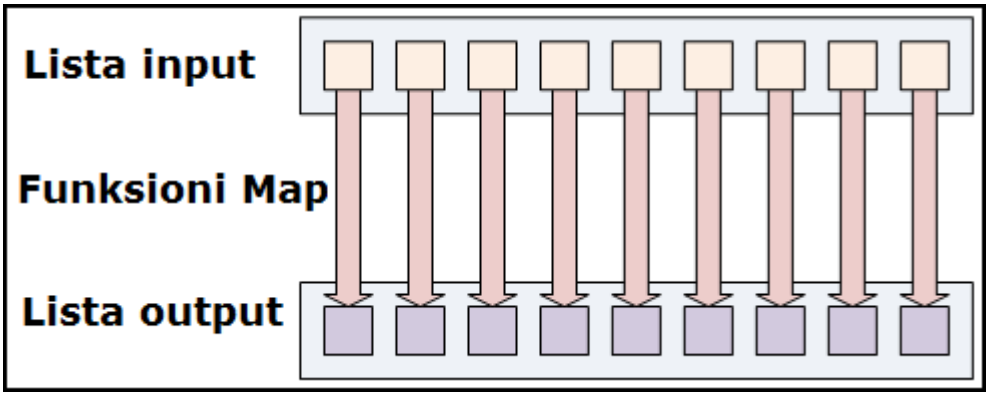


Figura 7.2 Krijimi i listave rezultat prej listave hyrëse

Faza Reduce nuk është e nevojshme në të gjitha rastet. Ajo do të përdoret në rastet kur për një çelës kemi më shumë se një rezultat dhe këto rezultate duam t'i agregojmë në një rezultat të vetëm.

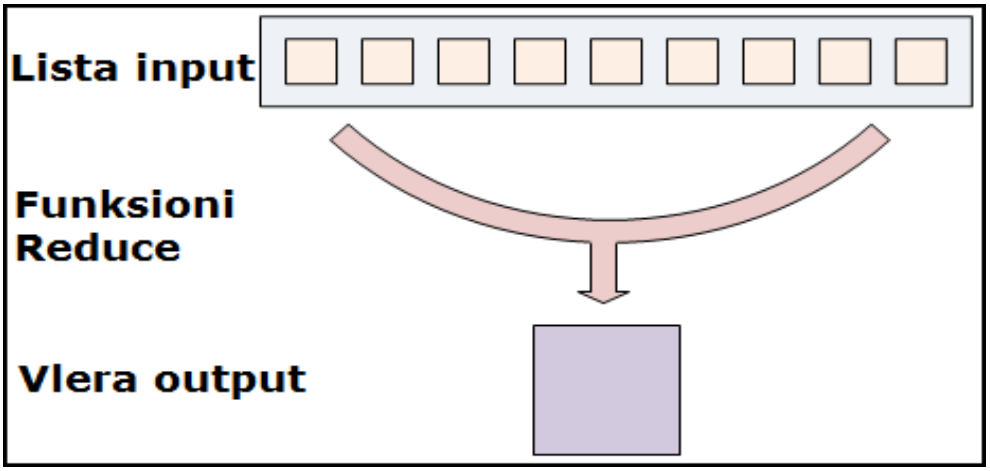


Figura 7.3 Agregimi i vlerave për një çelës

Në përfundim të procesit, rezultatet, të cilët ndodhen në skedarë të ndryshëm, do të bashkohen në një skedar të vetëm përfundimtar.

Të dyja fazat i jepen për ekzekutim makinave të ndryshme, duke na mundësuar një nivel të lartë paralelizimi.

Mjedisi i punës Hadoop MapReduce kërkon dy funksione kryesore për realizimin e fazave më sipër, *mapper* dhe *reducer*. Secili prej këtyre funksioneve përpunon të dhënat, pavarësisht nga njëra tjetra. Kjo krijon mundësinë e ekzekutimit të tyre në makina të ndryshme.

Pra paralelizmi është i pranishëm në të dyja fazat.

Kjo platformë mund të instalohet edhe lokalisht, por që të arrihet performanca e dëshiruar duhet të përdorim një numër të konsiderueshëm makinash për paralelizim. Për këtë arsye, rekomandohet të përdoret cloud, i cili mundëson infrastrukturë të tillë me kosto të pranueshme.

7.2 MapReduce në gjenerimin e vlerave të densiteteve

Le të jetë $\underline{t}=(t_1,t_2,...,t_d)$[1] një variabël i rastit me densitet f(.) për tu vlerësuar.

Supozojmë se kemi n vlera të variablit d-dimensional $\underline{t}$, respektivisht $\underline{t}_1,\underline{t}_2,...,\underline{t}_n$.

$\underline{t}_1 = (t^1_1,t^2_1,....,t^d_1)$

$\underline{t}_2 = (t^1_2,t^2_2,....,t^d_2)$

.....

$\underline{t}_n = (t^1_n,t^2_n,....,t^d_n)$

Matjet e mësipërme përdoren për vlerësimin e densitetit për vlera të fikse të variablave duke ekzekutuar algoritmat e propozuar.

Nqs përdorim d dimensione, densiteti do të vlerësohet për çdo d-she të vlerave të dimensioneve.

Çelësi i propozuar është: Vlerë_dim1-vlerë_dim2-...- vlerë_dimd

Si vlerë do të vendosen matjet e variablave, psh $(t^1_1,t^2_1,....,t^d_1)$

Vihet re që për një çelës do të kemi n përsëritje në skedar, ku n është numri i matjeve që kemi për variablat.

Nqs shënojmë me nr_pika_dim_i numri e pikave për të cilat do vlerësojmë densitetin për variabëlin *i*, atëherë numri i reshtave të skedarit hyrës do të jetë:

Nr_rreshta=1

Për i=1 në d bej

Nr_rreshta= Nr_rreshta nr_pika_dim_i*

Nr_rreshta= Nr_rreshta n* *(7.1)*

Për secilin rresht, do kryhen llogaritje me kompleksitet O(d).

Numri i veprime të përgjithshme do të ketë kompleksitet:

Nr_veprime = Nr_rreshta O(d)* *(7.2)*

Më poshtë po japin funksionet *map* e *reduce* për rastin e bërthamës d dimensionale Gaussiane. Për bërthamat e tjera mjafton të ndryshojme procedurën e brendshme të llogaritjes së vlerave.

Funksioni map

```
mapper  (skedari,  përmbajtje)
        Për secilin rresht në përmbajtje
            Çelës=fjala e parë në rresht
            Vektor=vendosVleraVektor(Çelës)
            Vlerë=0
             i=0
            Për secilëm fjalë në rresht
                Vlerë = Vlerë + pow((Vektor [i] –fjalë),2)
                 i=i+1
            Vlerë =exp(-1 * Vlerë /  (pow(h,2)*2))
            Emit(Çelës,Vlerë)
```

Funksioni reduce

```
reducer  (çelës,  vlerat)
        vlerë_rezultat=0
        Për secilëm vlerë te vlerat
                vlerë_rezultat = vlerë_rezultat + vlerë
        vlerë_rezultat = vlerë_rezultat / (n * pow(h,d) * pow(2*π,d/2)
         Emit(çelës, vlerë_rezultat)
```

Skedari fillestar ndahet në skedarë më të vegjël. Secili prej ketyre skedarëve i jep si parametër instancave të funksionit *map* në makina të ndryshme. Numri i skedarëve është mirë që të jetë të paktën sa numri i makinave në *cluster*. Funksionet map gjenerojnë vlerat e ndërmjetme të çelësave, të cilëve i jepen si parametër instancave të funksioneve *reduce* në makina të ndryshme. Funksionet *reduce* bëjnë llogaritjen përfundimtare të vlerës së një çelësi dhe e vendosin atë në një skedar

7.3 Vlerësimi financiar

Përdorimi i cloud rrit ndjeshëm efiçencën financiare në procesin e përpunimit të të dhënave dhe përllogaritjen e densiteteve, pasi ai shkurton në mënyrë drastike kostot fikse (dmth kostot që lidhen me blerjen dhe mirëmbajtjen e hardware) duke e lejuar përdoruesin që të blejë kohë në cloud vetëm kur ka të nevojshëm përpunimin. Rrjedhimisht kosto e hardware bëhet kosto variable (që dmth është një kosto që duhet paguar vetëm në procesin e përpunimit) duke minimizuar koston totale të procesit pasi në këtë mënyrë e shpërndan koston fikse (atë të blerjes së hardware) ndërmjet mijëra përdoruesve.

Nga ky këndvështrim përllogaritja e densiteteve të të dhënave në cloud përbën një avantazh ekonomik, sepse shkurton kohën e përpunimit të të dhënave dhe marrjen e rezultateve në një kohë sa më të shpejtë. Marrja e rezultateve sa më shpejtë është e dëshirueshme sepse shkurton kohën e punës dhe rrjedhimisht edhe koston monetare të përpunimit të të dhënave. E parë nga ky këndvështrim, kërkuesi do të ishte i interesuar të minimizonte kohën e përllogaritjes, që do te thote përdorimin e një numri sa më të madh kompiuterash (blerjen e një kapaciteti sa më të madh) në cloud. Por blerja e sa më shumë kapacitet rrit njëkohësisht koston (sepse çdo hapësirë shtesë do të kërkonte një pagesë monetare). E thënë ekonomikisht kjo definohet: për ds kohë kosto marzhinale e një njësie shtesë është më e madhe se 0, kosto totale rritet për çdo "njësi" cloud të blerë. Rrjedhimisht përdoruesi (ai që përllogarit densitetet) që të rrisë efiçencën ekonomike duhet gjithashtu të minimizojë koston variable dmth të blejë minimumin e nevojshëm të njësive (kapacitetit përllogaritës) në cloud. Në këtë mënyrë efiçenca ekonomike definohet si shkurtimi i kohës së përllogaritjes me minimumin e kapacitetit të blerë në cloud.

Kjo efiçencë arrihet në atë pikë ku kosto monetare në terma të kohës së perpunimit të të dhënave (funksioni i kohës së përpunimit në varësi të njësive (hapësirës së blerë) cloud të blera), që është një funksion rrënës me rritjen e hapësirës së blerë në cloud, barazohet me koston monetare të njësive cloud të blera, që është një funksion rritës me rritjen e hapësirës së blerë në cloud.

Koha totale është e barabartë:

$$T = (Nr_veprime*t') / x \quad (7.3)$$

ku është t' koha e përpunimit në orë ose minuta në mjedisin cloud dhe x është numri i makinave.

Kjo kohë është funksion i performancës relative në cloud.

Kosto monetare e kohës së përpunimit është:

$$C = a - T * c'1 \quad (7.4)$$

ku c'1 është kosto monetare e një njësie kohe dhe a është kosto oportune e mosvlerësimit të densiteteve.

Kosto monetare e njësive (ose hapësirës) cloud është:

$$C' = b + c_x*x \quad (7.5)$$

ku b është kosto fikse e abonomit në cloud, nëse ka të tillë, c_x është kosto marzhinale, dhe x është numri i njësive cloud të blera.

Kjo do të bëjë të mundur krahasimin e të dy funksioneve C dhe C' në të njëjtën njësi, njësia monetare.

Pra maksimizimi i fuqisë përllogaritëse me kosto minimiale arrihet kur:

$$C = C' \quad (7.6)$$

Zgjidhja e ekuacionit (7.6) për vlerën e x, jep numrin optimal të makinave dhe minimizon koston.

t' në formulën (7.3) është gjithashtu funksion i gjuhës së programimit. Rrjedhimisht edhe kosto optimale dhe numri i makinave mund të ndryshojë për gjuhë të ndryshme. Eksplorimi i këtij fenomeni do të jetë subjekt i kërkimit ë mëtejshëm.

Përfundime

Ky punim do të paraqesë analizimin dhe aplikimin praktik të modelit matematikor qe përshkruan metodën e vlerësimit të densiteteve të përbashkëta shumëdimensionale.

Kontributi do të jetë zhvillimi i një instrumenti në duart e përdoruesve dhe politikë-bërësve për të analizuar fenomenet stokastike, në rastin e kësaj teme ngjarjet ekonomike dhe testuar modelet teorik ekonomikë duke përdorur teorinë e probabilitetit.

Rrjedhimisht në pjesën e parë u algoritmua ky model dhe u paraqitën rezultatet në formë grafike. Rezultatet duken të pranueshme dhe konfirmojnë pritshmeritë mbi karakteristikat e ekonomisë shqiptare.
I gjithë modeli është shkuar në gjuhën php. Për më tepër modeli është përshatur më procedura që lehtësojnë përdorimin dhe manipulimin e të dhënave nga ekonomistët.

Ndërtimi i simulimit në platforma web lehtëson përdoruesit në përdorimin e këtyre modeleve pasi nuk kërkon kompjuter me fuqi të lartë për çdo përdorues. Atyre u nevojitet vetëm një web browser.

Meqenëse procesi bazohet në të dhëna, modeli është plotësuar me procedura për testimin e të dhënave.

Më tej teza është fokusuar në optimizmin e algoritmeve dhe optimizmin e efiçencës.

Të dhënat e sistemeve ekonomike dhe ngjarjeve stokastike në përgjithësi kanë një volum të madh, prandaj edhe procedurat për ekzekutimin e modeleve mbi to do të kërkojnë burime të shumta, si në kohë edhe në fuqi përpunuese të makinave.

Për zgjidhjen e këtij problemi, tema propozon teknikat më poshtë:

1) Teknika e particionimit

Teknika e particionimit krijon mundësinë ndarjes fizike të të dhënave në disqe të ndryshme duke ulur kohën e leximit të këtyre të dhënave nga sistemet e menaxhimit të bazave të të dhënave. Duke shfrytëzuar këtë teknikë, teza propozon një model particionimi, performanca e të cilit u testua mbi matjen e katër variablave makroekonomikë. Rezultatet e aplikimit konkret treguan që koha e përpunimit zvogëlohet me shpejtësi rritëse.

2) Teknika të paralelizimit

Modelimi informatik i teorisë së vlerësimit të densiteteve të bashkuara rezultoi në ndërtimin e procedurave llogaritëse, të cilat kryejnë një sërë veprimesh të pavarura nga njëra tjetra për nxjerrjen e rezultatit përfundimtar. Përdorimi i paralelizimit krijon mundësinë e uljes së kohës së ekzekutimit duke na mundësuar që procedurat të japin rezultate më të plota duke përllogaritur më shumë vlera dhe me efiçencë më të lartë.

Në tezë janë propozuar dy metoda të paralelizmit dhe është testuar njëra prej tyre.

2.1) Teknika multithreading

Në lidhje me këtë u eksperimentua me dy teknika.

Teknika e parë thërret një numër thread-esh të barabartë me numrin e pikave.

Eksperimenti tregoi që paralelizimi me këtë metodë sjell një rritje të kohës së ekzekutimit, e cila përkeqësohet në përpjesëtim të drejtë me rritjen e numrit të pikave, të cilave do u përllogaritet densiteti.

Në teknikën e dytë, numri i thread-eve është dhënë si funksion i numrit i procesorëve dhe numrit bërthamave të këtyre procesorëve. Eksperimenti tregoi që paralelizimi me këtë metodë, ndryshe nga metoda e parë, sjell një ulje të kohës së ekzekutimit, e cila përmirësohet në përpjesëtim të drejtë me rritjen e numrit të pikave, të cilave do u përllogaritet densiteti.

Rrjedhimisht, teza propozon që teknika e dytë të përdoret.

2.2) Teknika cloud

Rritja e performancës së algoritmit me metodën më sipër kërkon rritje të numrit të procesorëve. Kjo sjell dy probleme.

Problemi i parë është rritja e kostos financiare si rezultat e rritjes së numrit të procesorëve.

Problemi i dytë qëndron në numrin e kufizuar të procesorëve në një makinë.

Teza propozon që të zgjidhë këto dy probleme duke shfrytëzuar avantazhet që ofrojnë sistemet cloud.

Për këtë qëllim, teza propozon një metodologji të përgatitjes dhe përshtatjes së të dhënave për strukturën cloud dhe ndërton algoritme për modelin teorik Tanku dhe Ceca (2013)(2014).

Së fundi, duke i kushtuar një rëndësi elementit financiar, teza propozon një model për vlerësimin e numrit optimal të makinave në funksion të fuqisë përpunuese, kostos financiare për njësi pune dhe ambientit të programimit.

Nga ky këndvështrim, ka rëndësi eksplorimi i këtij funksioni për ambiente të ndryshme. Në të ardhmen tema do të zgjerohet në eksplorimin e efiçencës në mjedise alternative.

Produkti i gjeneruar nga ky studim ka potencial të japë një kontribut të rëndësishëm në studimin e të gjithë fenomeneve stokastikë, pavarësisht nga disiplina shkencore në të cilën ato vërehen, qofshin këto fizike, sociale, natyrore, biologjike, mjekësore etj.

Referenca

1. **K., Tanku A. & Ceca.** Density estimation for economic variables – a genuine application. *Seria e materialeve te diskutimit te BSH.* 2013.

2. **C, Tanku A. & Ceca.** Stastistical estimation of macroeconomic indicators addressing densitz estimation techiques. *Seria e materialeve te diskutimit te BSH.* 2014. Pranuar per botim 2014.

3. **I., Hacking.***Representing and Intervening.* s.l. : Cambridge University Press, 1983, Vëll. i Introductory Topics in the Philosophy of Natural Science.

4. **Box, George E.P. & Draper, N.R.** Empirical Model-Building and Response Surfaces. s.l. : Wiley, 1987, f. 424.

5. **Hagedorn, R. et.al.** *http://www.ecmwf.int/staff/paco_doblas/abstr/tellus05_1.pdf.* 2005.

6. **Griffiths, E. C.** What is a model? [Në linjë] 2010. [Cituar më: 25 04 2014.] https://63587809-a-bf11e6be-s-sites.googlegroups.com/a/ncsu.edu/emily-griffiths/whatisamodel.pdf.

7. **S., Frigg R. & Hartmann.** Models in Science. [re.] Edward N. Zalta. *The Stanford Encyclopedia of Philosophy.* Fall 2012 Edition, 2012. http://plato.stanford.edu/archives/fall2012/entries/models-science/.

8. **Hughes, R. I. G.** *Theoretical practices of physics: philosophical essays.* s.l. : Oxford University Press. 289 pp, 2010.

9. **Morgan, Mary.** Learning from Models. [re.] M.S. Morgan and M.C. Morrison. *Models as Mediators.* s.l. : Cambridge University Press, 1999, fv. 347-388.

10. **I., Hacking.** *Representing and Intervening. Introductory Topics in the Philosophy of Natural Science.* s.l. : Cambridge University Press, 1983.

11. **Hartmann, Stephan.** The World as a Process: Simulations in the Natural and Social Sciences. [re.] R. Hegselmann et al. *Modelling and Simulation in the Social Sciences from the Philosophy of Science Point of View.* 1996, fv. 77-100.

12. **Winsberg, Eric.** *Computer Simulation and the Philosophy of Science.* [re.] Blackwell Publishing Ltd. 5, September 2009, Philosophy Compass, Vëll. i 4, fv. 835–845.

13. **Roman Frigg, Julian Reiss.***The philosophy of simulation: hot new issues or same old stew?* 3, s.l. : Springer Netherlands, August 2009, Synthese, Vëll. i 169, fv. 593-613 .

14. **Humphreys, Paul.***The philosophical novelty of computer simulation.* 3, 2009, Synthese, Vëll. i 169, fv. 615–626.

15. **R., AXELROD.** *Advancing the Art of Simulation in the Social Sciences.* [re.] Hegselmann R., Terna P. Conte R. Berlin : s.n., 1997, Simulating Social Phenomena, fv. 21-40.

16. **R, EPSTEIN JM and Axtell.** *Growing Artificial Societies.* s.l. : Brookings Institution Press MIT Press, 1996.

17. **Aksel Paulsen, Søren Overgaard,Jens Martin Lauritsen.** Quality of Data Entry Using Single Entry, Double Entry and Automated Forms Processing–An Example Based on a Study of Patient-Reported Outcomes. *P L o S One.* 2012, Vëll. i 7, 4.

18. **Ramírez, José G. and Ramírez, Brenda S.** Analyzing and Interpreting Continuous Data Using JMP: A Step-by-Step Guide.

19. **Getis, A.** *Spatial statistics.* 2009.

20. **Gatrell, T.C Bailey & A.C.** *Interactive Spatial Data Analisys.* Harlow : Longman.

21. **José G. Ramírez, W.L. Gore and Associates Inc.** Statistical Intervals: Confidence, Prediction,Enclosure. *http://www.sascommunity.org/mwiki.* [Në linjë]

22. **Niranjan Nagarajan, Uri Keich.** *Reliability and efficiency of algorithms for computing the significance of the Mann–Whitney test.* 4, 2009, Computational statistics, Vëll. i 24, fv. 605-622.

23. http://hadoop.apache.org/. [Në linjë]

24. **R., Solow.** *Economic History and Economics.* May 1985, The American Economic, Vëll. i 75. 2.

Aneks 1: Simulimi në PHP i modelit të vlerësimit të densiteteve probabilitare

Në këtë kapitull, trajtohet procesi i algoritmimit dhe i kodimit në gjuhën php i përllogaritjes së vlerave të densiteteve probabilitare të ndërtuara bazuar mbi metoda të vlerësimit të densiteteve me anë të bërthamave. Programi synon që të ngrejë një algoritëm optimal, për të lehtësuar vrojtimin e marrëdhënieve kovaracionale dhe sjelljeve të ndërsjella të vektorëve të rastit shumë dimensionalë. Vrojtimi i këtyre marrëdhënieve realizohet nëpërmjet tenzorëve të vlerave të përllogaritura të densiteteve të vlerësuara dhe të vizualizimit nëpërmjet një aparati grafik të ndërtuar për këtë rast.

Simulimi është përshtatur për të lexuar të dhënat nga një skedar i jashtëm. Ai pranon një numër çfarëdo kolonash (variablash) dhe për secilën prej tyre përllogarit vlerat minimale dhe maksimale për përdorim të mëpasshëm. Ofrohet mundësia e zgjedhjes deri në 2 variabla për gjithë vlerat e tyre dhe përdorimi i vlerave të fiksuara për variablat e tjera.

Matrica e formuar nga vlerat e përllogaritura të densiteteve të pozuara përdoret për të llogaritur tenzorin n +1 dimensional të variablave dhe vlerave të funksionit të vlerësuar. Tenzori i formuar transformohet në një vektor, i cili iu dha si atribut klasës grafike që u përdor në php. Janë bërë optimizime të duhura në procesin e përllogaritjës së vlerave të tenzorit. Realizimi i prerjeve n – 2 dimensionale të dhëna më parë, të çfarëdoshme, është bërë i mundur për të dhënë paraqitje grafike të shpërndarjeve të të dhënave të pozuara.

Programi realizon ndërtim grafikesh 3 dimensionalë dhe rrotullimin e tyre horizontal e vertikal për lehtësi vrojtimi.

Pamja e aplikacionit

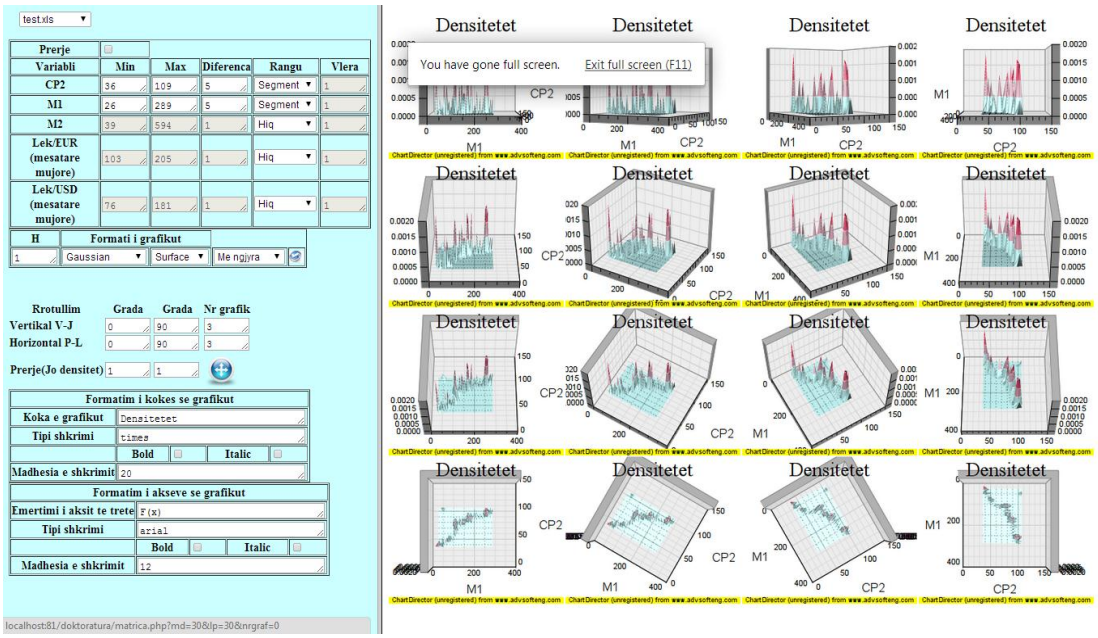


Figura A1.1 Pamje e parë e simulimit

Të dhënat

Në këtë version, të dhënat ruhen ne skedar Excel.

Programi mund të ruajë skedarë të ndryshëm. Zgjedhja e tyre bëhet përmes ashensorit në pjesën e sipërme të faqes kryesore.

Në faqen e punës, vendosim vlerat e matura të secilit variabël në kolona të veçanta, duke filluar nga kolona e parë. Programi do të konsiderojë kolonat në fillim derisa të arrijë fjalën kyçe *Fund.*

Për secilin variabël. Programi do të përpunojë të dhëna derisa të arrijë fjalën kyçe *Fund.* Për të gjithë variablat, kemi të njëjtin numër matjesh.

	A	B	C	D	E	F	G
1	CPI	M1	M2	Lek/EUR (mesatare mujore)	Lek/USD (mesatare mujore)	fund	
2	37.1	27.62	40.15	114.76	102.47		
3	37.7	28.03	41.22	114.12	103.03		
4	38.0	28.00	42.41	115.41	102.47		
5	41.4	27.83	42.87	115.30	101.95		
6	42.1	28.21	43.55	114.88	100.04		
7	42.8	29.73	46.13	113.26	96.76		
8	41.4	28.74	45.72	110.87	91.37		
9	40.5	31.32	48.78	106.08	86.92		
10	40.6	31.47	48.93	104.86	84.64		
11	41.0	32.22	49.79	110.49	87.25		
12	41.3	33.39	51.41	115.91	92.20		
13	42.1	38.77	57.47	117.51	95.22		
14	42.9	39.33	59.08	119.54	95.15		
15	fund	fund	fund	fund	fund		
16	43.8	39.62	59.92	120.42	93.76		
17	44.0	40.75	61.62	127.15	93.28		
18	44.7	41.32	62.99	129.97	93.23		
19	44.5	42.59	64.80	127.20	92.56		
20	43.7	44.05	66.96	127.11	92.19		
21	42.2	46.94	70.34	126.38	90.60		
22	42.6	48.47	72.92	123.71	91.57		
23	43.0	49.86	75.15	124.94	94.43		

Figura A1.2: Format i të dhënave ne Excel

Programi automatikisht gjen vlerat ekstreme për çdo variabël.

Variabli	Min	Max	Diferenca	Rangu	Vlera
CPI	36	43	1	Hiq	1
M1	26	40	1	Hiq	1
M2	39	60	1	Hiq	1
Lek/EUR (mesatare mujore)	103	120	1	Hiq	1
Lek/USD (mesatare mujore)	83	104	1	Hiq	1

Figura A1.3: Vlerat e variablave

Për secilin variabël zgjedhim rangun e vlerave. Mund të zgjedhim deri në dy variabla segment.

Duhet të vendosni vlerat minimale dhe maksimale për variablat segment. Duhet të zgjidhni një vlerë për variablat fiks.

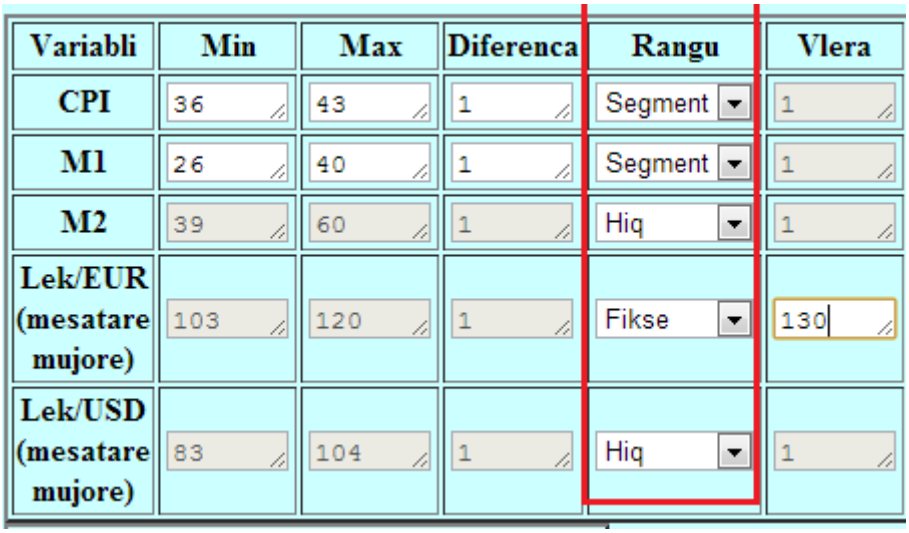


Figura A1.4: Rangu i variablave

H është parametri i lëmimit. Jepet mundësia të përdoret një nga dy bërthamat, Gaussian dhe Epanechnikov. Grafiku mund të jetë me ngjyra ose bardh e zi.

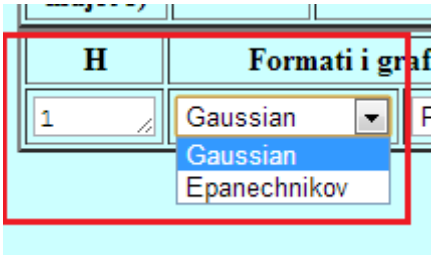


Figura A1.5: Tipet e bërthamave

Mundësohen dy tipe kryesorë grafikësh: Contour dhe Surface.
Grafiku mund të prodhohet në kënde të ndryshme vrojtimi. P.sh. duke vendosur numrin e grafikëve 3, me gradacion nga 0 në 90 për rrotullimin vertikal, atëherë grafikët do të paraqiten të rrotulluar 0,45,90 grade.

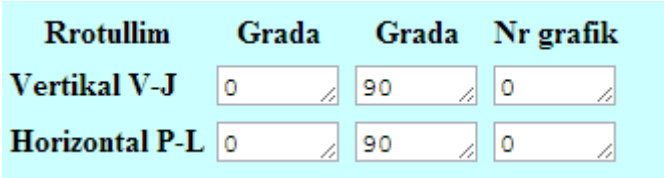


Figura A1.6: Tipet e rrotullimit

Programi jep grafikët e densiteteve të vlerësuara të variablave si edhe prerjen tërthore të variablave në varësi nga njëri tjetri.

Për realizimin e programit është përdorur gjuha e programimit PHP 5.3.8.

Klasa *Spreadsheet_Excel_Reader* është përdorur për të komunikuar me skedarët Excel.

Për gjenerimin e grafikëve është përdorur libraria grafike ChartDirector.

Për grafikët tredimensionalë, ajo merr si parametër dy vektorë me vlerat e akseve dhe vektorin me të dhënat rezultat.

Për grafikët dydimensionalë, ajo merr si parametër vektorin me vlerat e aksit dhe vektorin me të dhënat rezultat.

Aneks 2: Ekzekutime të aplikimit mbi të dhënat e ekonomisë

Më poshtë do të paraqesim grafikë rezultat të nxjerrë nga aplikimi në web nga të dhëna reale.

Të dhënat

Tanku dhe Ceca (2014) kanë përdorur observime tremujore te parasë nominale, CPI dhe GDP. Të dhënat janë marrë nga FRED dhe mbulojnë periudhën 1959 – 2013. Figurat paraqiten në USD, për paratë dhe GDP dhe në formë indeksi për CPI. Të dhënat do të riorganizohen, duke u transformuar nga koncepti i zakonshëm i serive kohore, në mënyrën e paraqitur nga Tanku dhe Ceca (2013):

$$\begin{cases} \underline{x}_1 = \left(x_1^1, x_1^2, \dots, x_1^d\right) \\ \underline{x}_2 = \left(x_2^1, x_2^2, \dots, x_2^d\right) \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots \\ \underline{x}_n = \left(x_n^1, x_n^2, \dots, x_n^d\right) \end{cases} \quad \text{ose} \quad X_{1\times n} = \mathcal{F}_{d\times n}$$

Përshkrimet e variablave janë dhënë në tabelën A2.1

Tabelë A2.1 Variablat ekonomike

Tipi	**Variabli**	**Përshkrimi**
Të dhëna	CPI	Indeksi CPI
	RM2	M2 real
Të dhëna në diferencën e parë	Dcpi	Indeksi CPI (Diferenca e parë)
	dRM2	M2 real (Diferenca e parë)
Të dhëna në përqindje relative	INF	Inflacioni
	pRM2	M2 real (Diferenca e parë)

Rasti 1. Densitetet e probabiliteteve të përbashkëta

RM2 vs CPI (ndryshimet në përqindje, observime tremujore)

Paraqitja surface

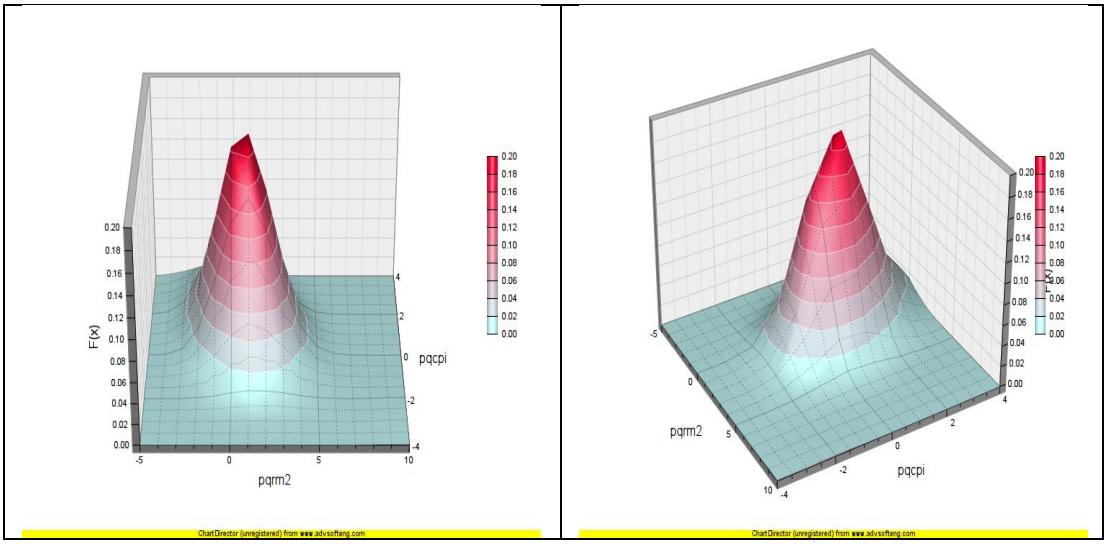


Figura A2.1: Paraqitja surface RM2 vs CPI

Paraqitja contour

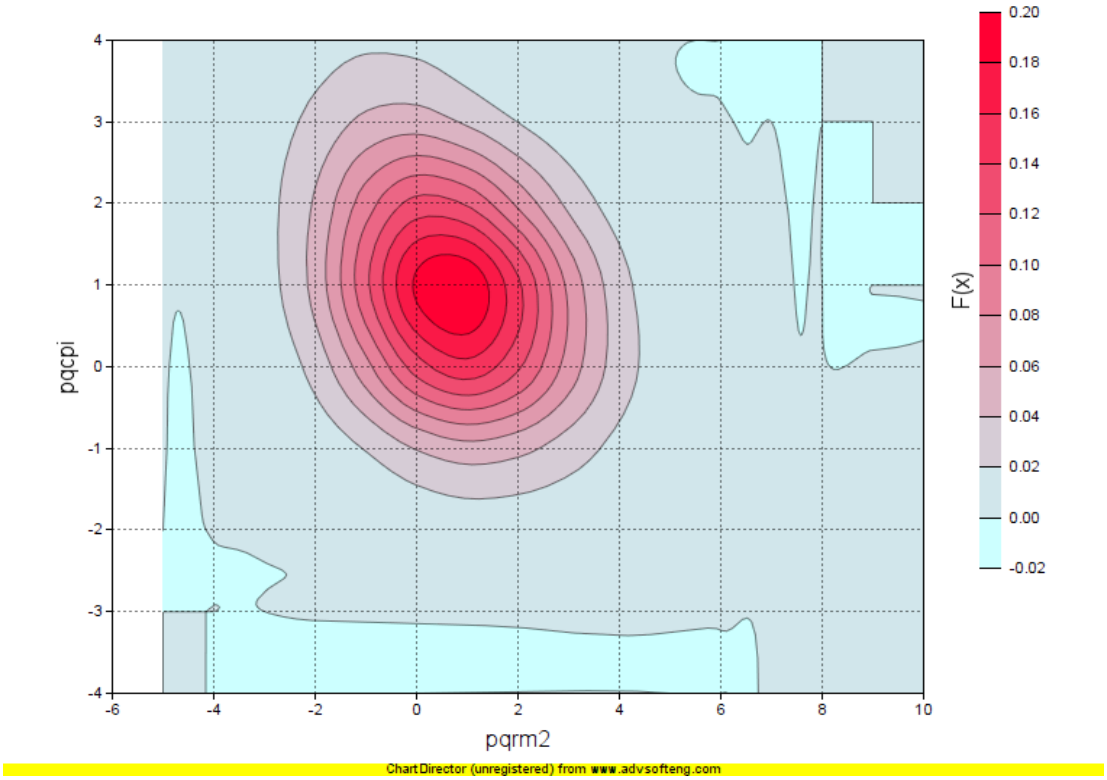
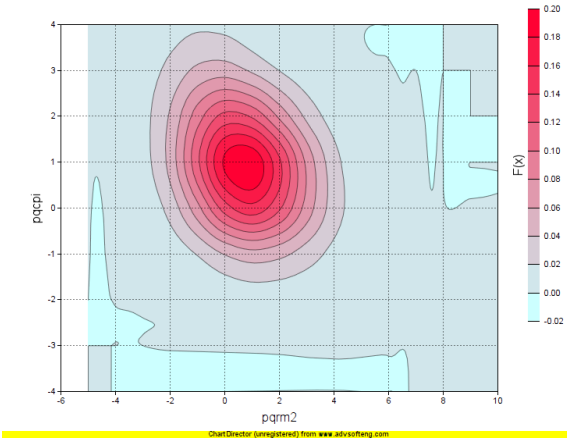


Figura A2.2: Paraqitja contour RM2 vs CPI

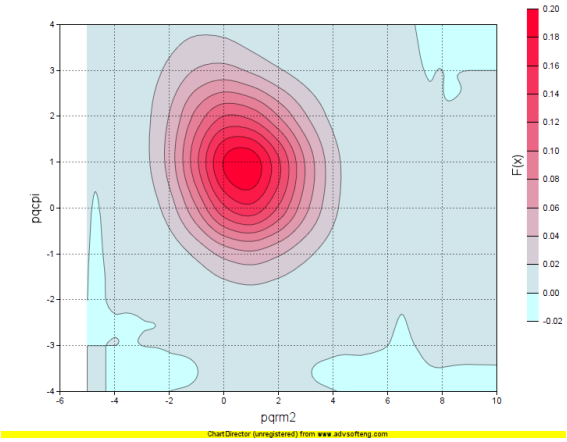
Rasti 2. Densitetet e probabiliteteve të përbashkëta

RM2 vs CPI (ndryshimet në përqindje, observime tremujore)

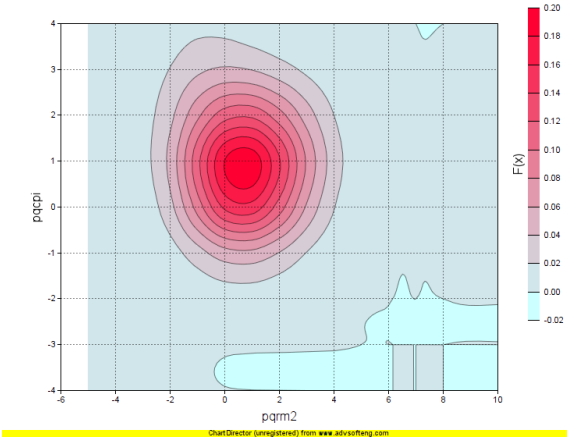
a) **Figura A2.3**: Simultaneous



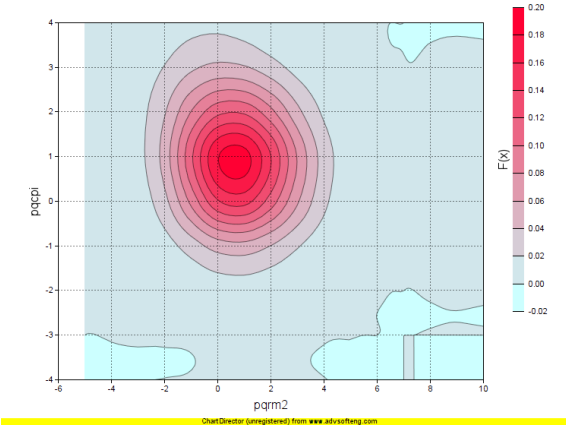
b) **Figura A2.4**: Lag 1



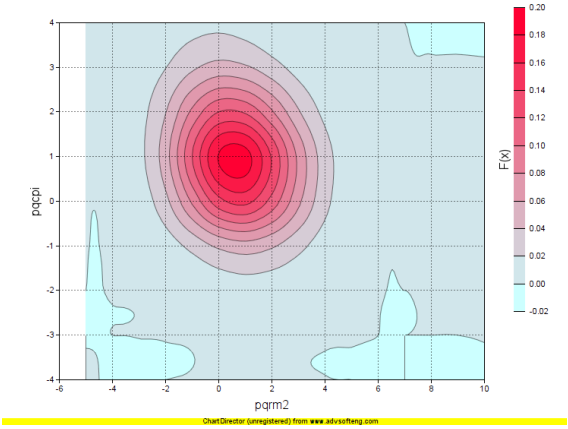
c) **Figura A2.5**: Lag 2



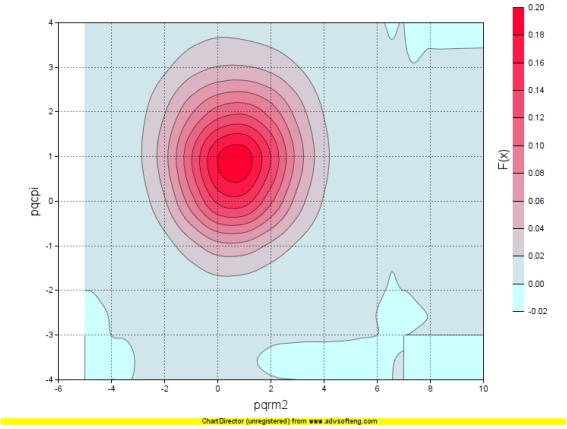
d) **Figura A2.6**: Lag 3



e) **Figura A2.7**: Lag 4



f) **Figura A2.8**: Lag 6



Rasti 3. Densitetet e probabiliteteve të përbashkëta të kushtëzuara dRM2 vs dCPI, për figurën A2.4

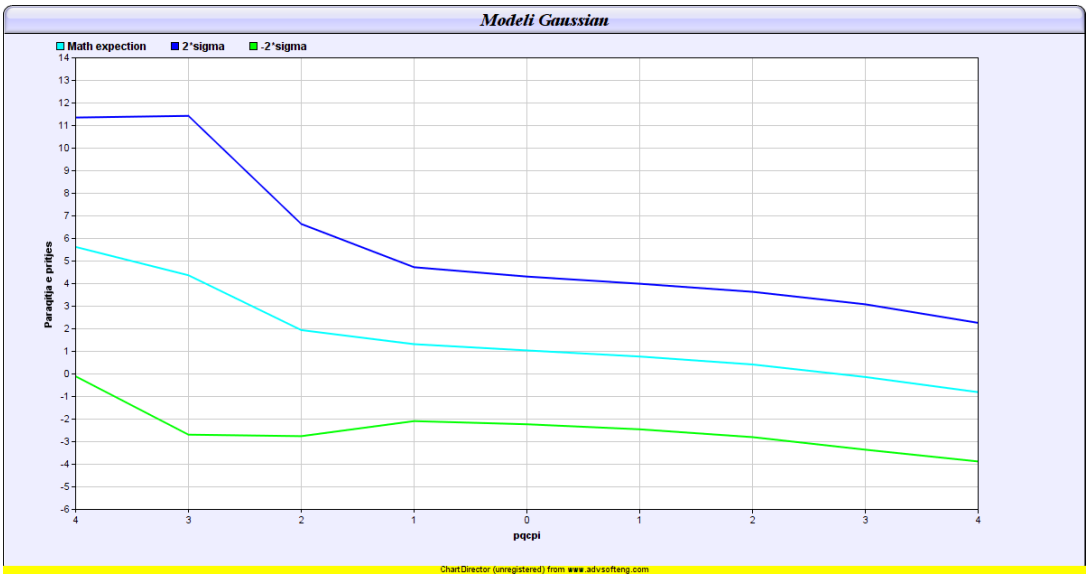


Figura A2.9: Pritja matematike dhe devijimi standard e M2 për ndryshimet e CPI

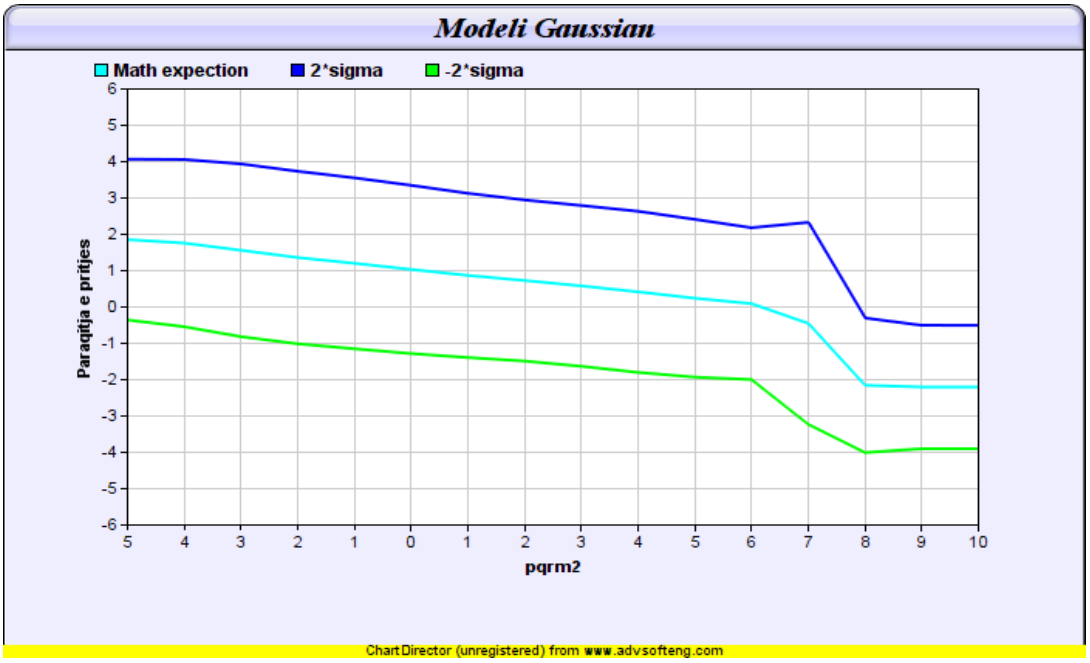
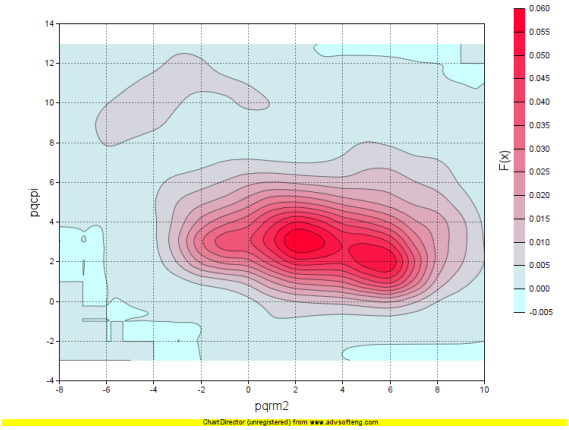


Figura A2.10 Pritja matematike dhe devijimi standard i CPI për ndryshimet e M2

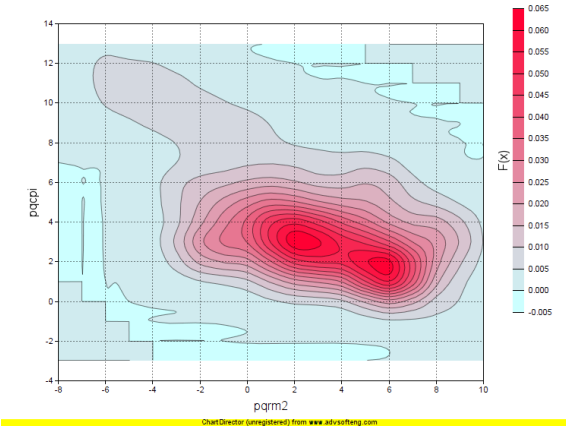
Rasti 4. Densitetet e probabiliteteve të përbashkëta

RM2 vs CPI (ndryshimet në përqindje, observime vjetore)

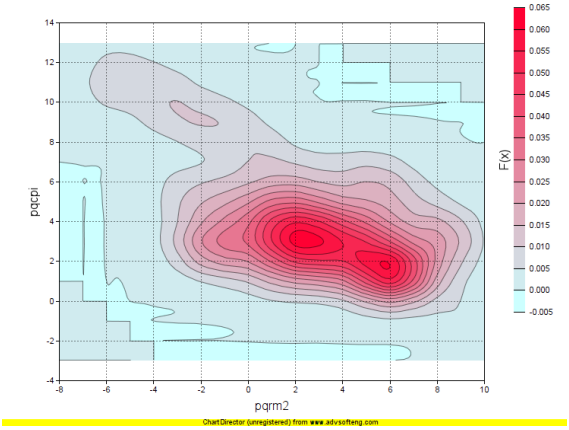
a) **Figura A2.11**: Simultaneous



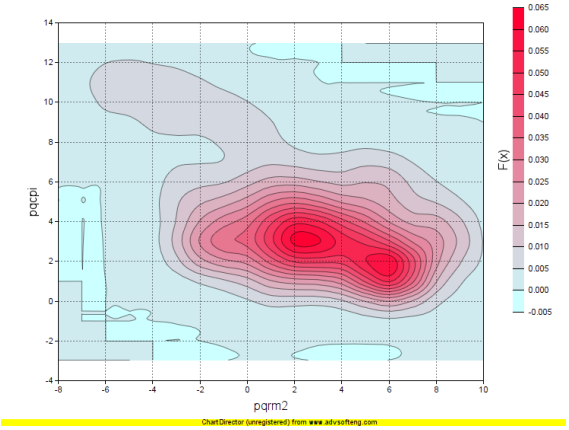
b) **Figura A2.12**: Lag 1



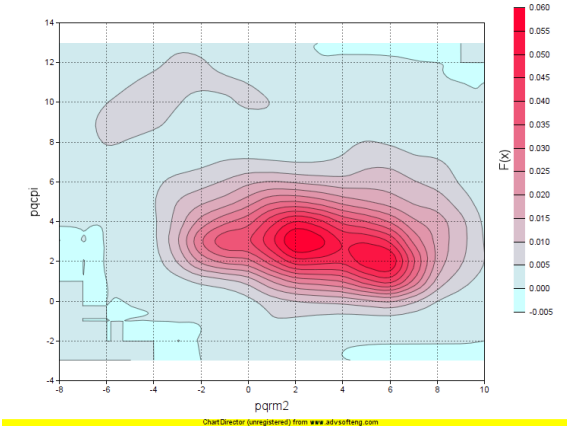
c) **Figura A2.13**: Lag 2



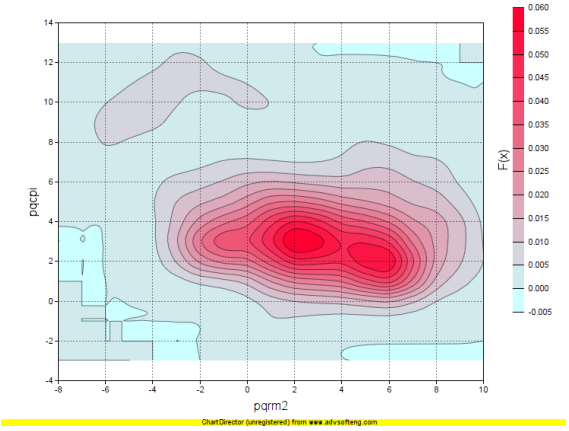
d) **Figura A2.14**: Lag 3



e) **Figura A2.15**: Lag 4



f) **Figura A2.16**: Lag 6



Rasti 5. Paraja reale dhe inflacioni, observime vjetore, Lag 2

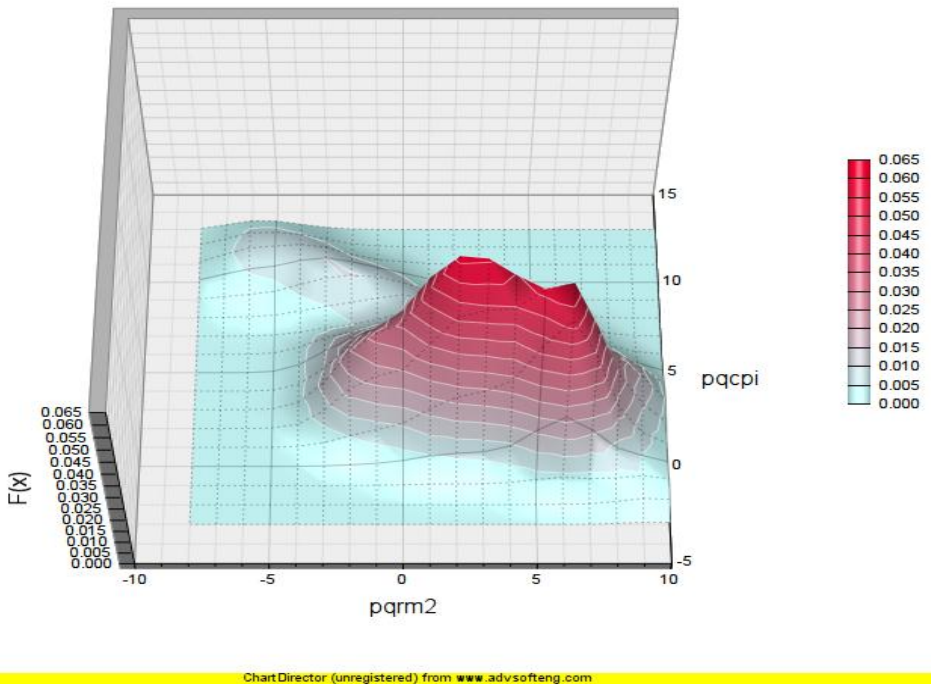


Figura A2.17: Paraqitja surface e figurës A2.13

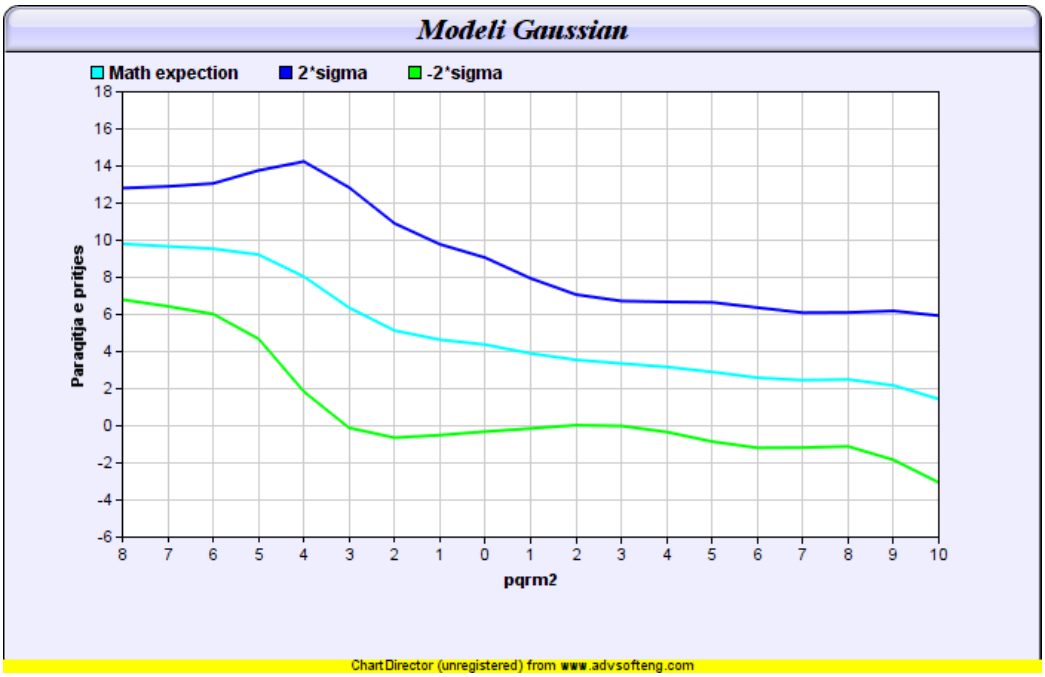
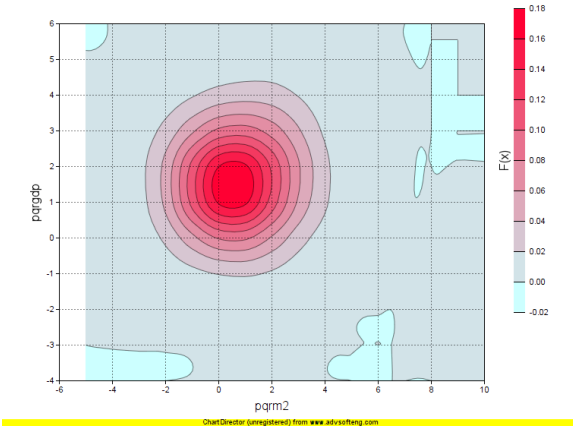


Figura A2.18: Prerja e tërthortë e figurës A2.13

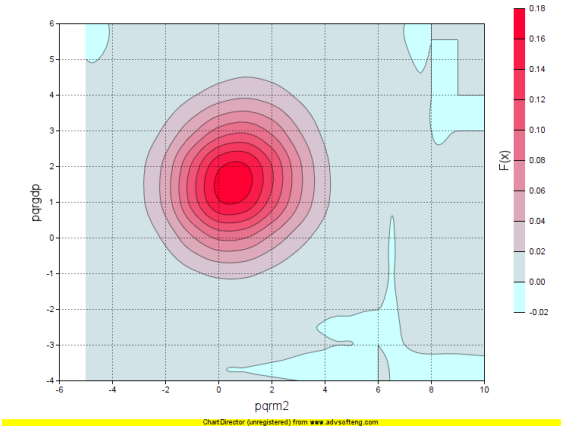
Rasti 6. Densitetet e probabiliteteve të përbashkëta

RM2 vs RGDP (ndryshimet në përqindje, observime tremujore)

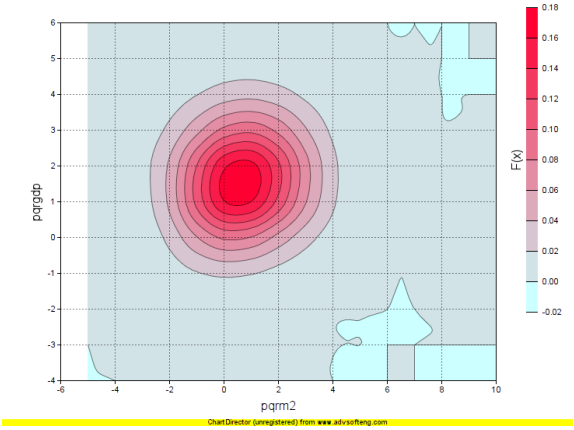
a) **Figura A2.19**: Simultaneous



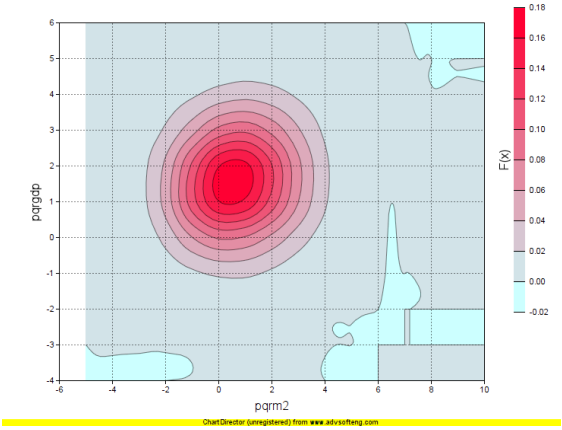
b) **Figura A2.20**: Lag 1



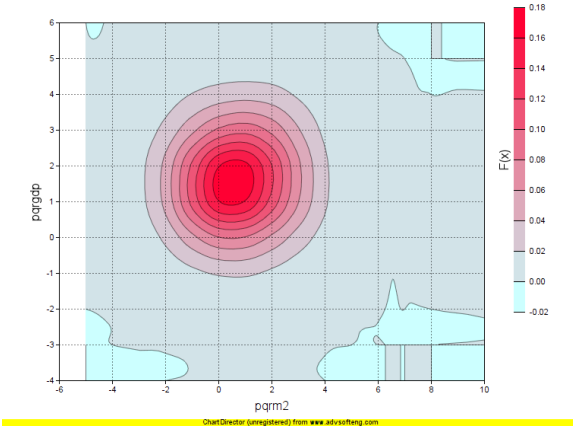
c) **Figura A2.21**: Lag 2



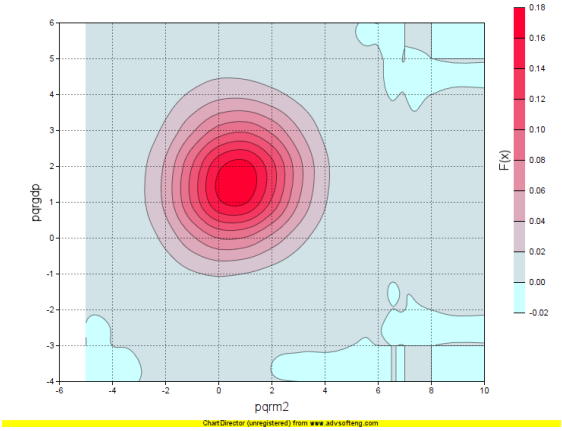
d) **Figura A2.22**: Lag 3



e) **Figura A2.23**: Lag 4



f) **Figura A2.24**: Lag 6



Rasti 7. Densitetet e probabiliteteve të përbashkëta

RM2 vs RGDP (ndryshimet në përqindje, observime tremujore)

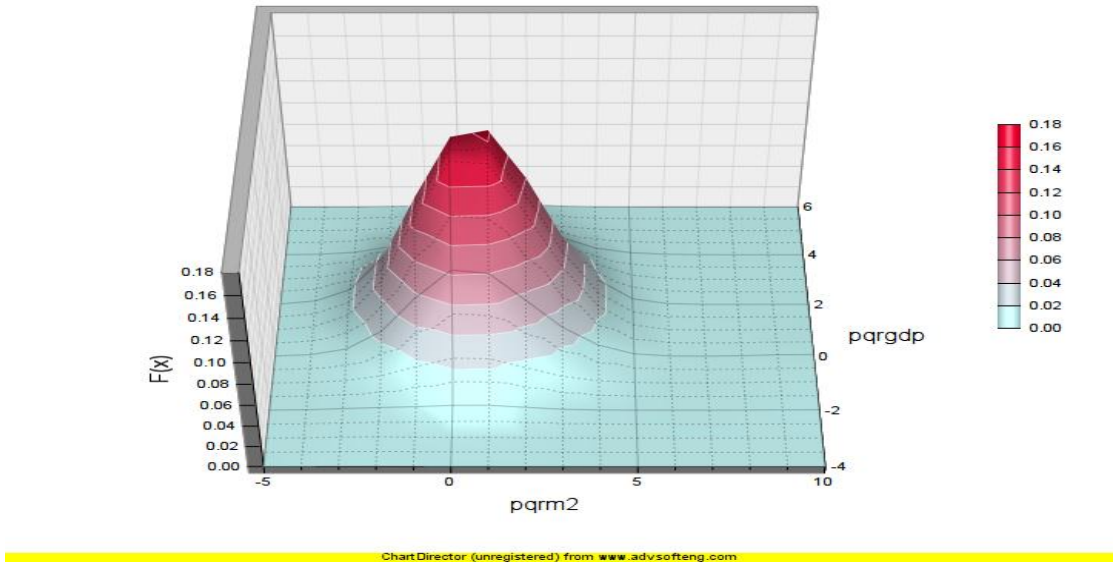


Figura A2.25: Paraqitja surface e figurës A2.20

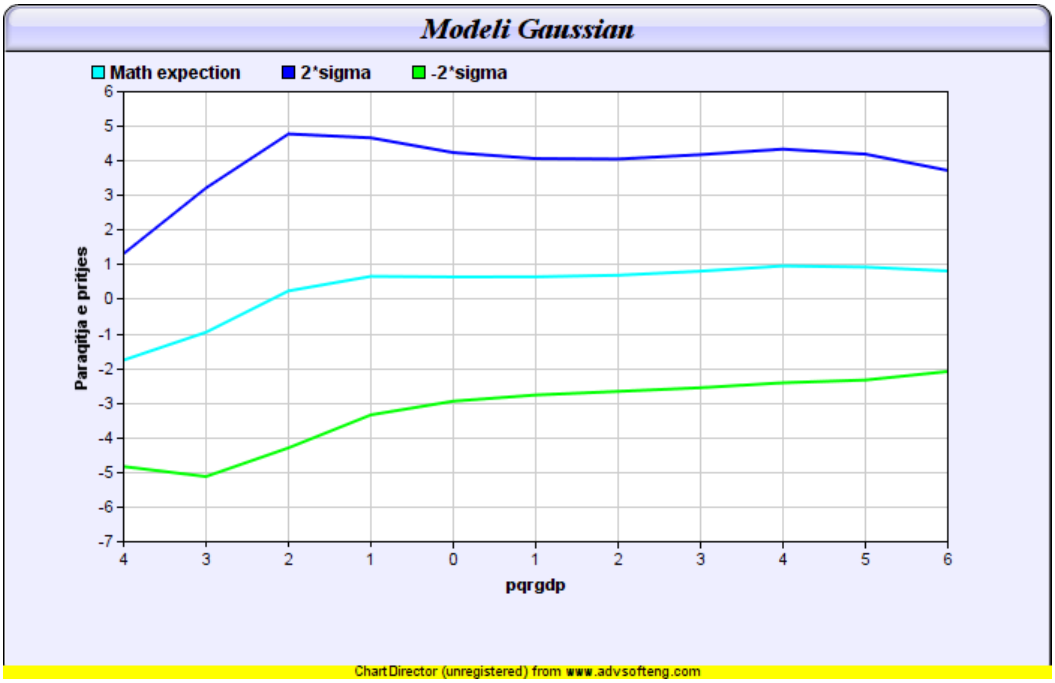


Figura A2.26: Prerja e tërthortë e figurës A2.20

Falënderime

I shpreh mirënjohjen dhe falenderimet e mia udhëheqësit tim Altin TANKU dhe profesorit tim Kliti CECA. Jam ndier i privilegjuar dhe i nderuar që më kanë drejtuar, pasi me profesionalizmin e tyre, optimizmin, energjinë pozitive, dashamirësinë, këshillat, mbështetjen, ndihmën e pakursyer dhe inkurajimin e tyre, gjithmonë kanë ditur të më drejtojë në rrugën e duhur për realizimin e kësaj teme.

I shpreh mirënjohjen dhe falenderimet e mia profesorit tim Endri Xhina. E falënderoj për mbështejen në të gjithë rrugëtimin tim. Me cilësitë më të mira personale dhe profesionale që e karakterizojnë, e falenderoj për idetë që më dha për fokusin e studimeve të doktoratës.

Një falënderim i veçantë për të gjithë grupin e profesorëve të nderuar të Fakultetit të Shkencave të Natyrës për kontributin që kanë dhënë në formimin tim akademik.

Së fundi falenderimet e mia do të shkojnë për familjen time. Mbështetja e tyre dhe dashuria e madhe kanë qenë faktori kryesor në edukimin tim profesional dhe njerëzor. Ata kanë qenë modeli i dy prindërve shumë të përkushtuar dhe më kanë dhuruar gjithmonë dashuri pa kufi.

Bibliografi

Publikime:

SAATCIU, D. & DIBRA D. 2014. Ndërtimi i grafikëve 3-dimensionalë në Matlab për shpërndarjen e densiteteve probabilitarë. *Buletini i Shkencave Natyrore Nr. 17 FSHN, UT*

SAATCIU, D. & BIMBARI B. 2014. Modelimi i testimeve logjike e probabilitare për të dhënat e vazhdueshme. *Buletini i Shkencave Natyrore Nr. 17 2011 FSHN, UT*

SAATCIU, D. 2014. Algorithms for scalene economic variables aproximation. Mean and dispersion. *International Journal Sciences, Innovation and new Technology*

Prezantime*:*

FEJZAJ J. & XHINA E. & SAATCIU D. & BIMBARI B. 2011. Performance improvement through table partitioning (comparison of table partitioning in SQL server 2008). *1st International Symposium on Computing in Informatics and Mathematics. Durrës June 2011*

SAATCIU, D. 2012. Ngritja e disa algoritmeve optimale për përafrimin me anë të metodës së vlerësimit të densiteteve të variablave ekonomike të çfarëdoshëm; *Konferencë Shkencore: Fakulteti i Shkencave Natyrore në 100 vjetorin e pavarësisë, Tiranë, Nëntor 2012*

SAATCIU, D. 2013. Computer models in probability density estimation. Applications in economy; *International Conference: Economic research in south east Europe, Bank of Albania, Tirana, December 2013*

DHËMBI A. & ZHAKA L. & XHINA E. & SAATCIU D. 2013. Implementation of Cloud Computing in Web Applications. *Information Systems and Technology Innovations: towards a digital Economy. June*

Përmbledhje

Autoritetet ekonomikë e kanë të nevojshme të kuptojnë dhe të parashikojnë sjelljen e treguesve kryesore makroekonomikë (sjellje e treguesve të konsumit e të prodhimit, të treguesve financiarë dhe monetarë, të kursit të këmbimit etj) dhe ndërveprimin e tyre. Gjithashtu, ata duhet të vlerësojnë efektivitetin e treguesve së politikave (politika e normës së interesit, politikat fiskale, si dhe kufizimet direkte e indirekte në aktivitetin ekonomik). Për të realizuar këtë qëllim, ekonomistët kanë propozuar modele të ndryshme empirike. Disa nga këto modele janë tejet të ngurtë për të përshkruar ekonominë dhe strukturën e saj stokastike – ata vuajnë nga supozime teorike joreale, të cilat imponojnë kufizime shtesë. Këto modele bazohen ne metodologjinë e vektorëve autoregresivë, sistemet e ekuacioneve diferenciale dhe ekuacioneve të diferencës së parë. Për të shmangur këto probleme, studiuesit kanë propozuar përdorimin e metodës të vlerësimit të densiteteve të përbashkëta, duke përfituar nga përcaktimi i ekonomisë si një ngjarje rasti shumëdimensionale. Kjo metodë bazohet në llogaritjen e densiteteve shumëdimensionale dhe interpretimit të tyre grafik për të gjykuar mbi ngjarjet e fenomenet ekonomike.

Ky punim do të paraqesë analizimin dhe aplikimin praktik të modelit matematikor qe përshkruan metodën e vlerësimit të densiteteve të përbashkëta shumëdimensionale.

Kontributi do të jetë zhvillimi i një instrumenti në duart e përdoruesve dhe politikë-bërësve për të analizuar ngjarjet ekonomike dhe testuar modelet teorik ekonomikë duke përdorur teorinë e probabilitetit.

Kontributi i dytë është optimizimi i algoritmeve dhe shkurtimi i kohës së ekzekutimit nëpërmjet paralelizimit të proçedurave llogaritëse.

Dhe së fundmi, vendosja e aplikimit në sistemet cloud për të përfituar aftësinë përpunuese me kosto minimale.

Vërtetësia e hipotezave në këtë tezë është testuar përmes disa eksperimenteve.

Fjalë Kyçe: *modelim informatik, shpërndarje probabilitare,* optimizim, p*aralelizim, vizualim grafik, cloud, vlerësim empirik, modelim ekonometrik, analize kosto*

Abstract

Economic authorities need to understand and forecast the behavior of macroeconomic variables (consumption, production, financial and monetary indicators, exchange rate, etc.) and their interactions. In addition, they need to evaluate the effectiveness of their policy variables (policy interest rate, fiscal policies, and other direct and indirect restrictions on economic activity). Economists have proposed several empiric models to achieve this goal. Some of these models are too rigid to fully describe the economy and its stochastic structure – they suffer from unrealistic theoretical assumptions, which in turn impose additional restrictions. These models are based on vector autoregressive methodology, systems of difference and differential equations.

To overcome these difficulties, researchers have recently proposed a new method based on multidimensional joint density estimation. The application of this method benefits from the definition of the economy as a multidimensional random event. It is based on the estimation of multidimensional density functions and the interpretation of their graphical representation.

This study will focus on the analysis and implementation of mathematical model which the multidimensional joint density estimation method is built upon.

The contribution will be the development of an instrument at the hands of the researchers and policymakers to analyze economic events and test economic theoretic models using probability theory.

Optimizing algorithms and reducing execution time through the parallel computing of programming procedures is the second contribution.

Finally, this research aims to achieve efficiency and reducing financial cost using the cloud to run the application.

The hypotheses in this thesis are tested though a number of experiments.

Keywords: *computer model, probability density,* optimization, *parallelism, graphical visualization, cloud, empirik analzsis, econometrics, cost effectiveness*