UNIVERZA V LJUBLJANI
FAKULTETA ZA DRUŽBENE VEDE

Blaž Simčič

Metoda glavnih komponent in manjkajoči podatki

Magistrsko delo

Ljubljana, 2014

UNIVERZA V LJUBLJANI
FAKULTETA ZA DRUŽBENE VEDE

Blaž Simčič

Mentor: doc. dr. Aleš Žiberna
Somentorica: red. prof. dr. Anuška Ferligoj

Metoda glavnih komponent in manjkajoči podatki

Magistrsko delo

Ljubljana, 2014

POVZETEK

Metoda glavnih komponent in manjkajoči podatki

Naloga obravnava vpliv različnih pristopov za obravnavo manjkajočih podatkov na obnašanje metode glavnih komponent v primeru neodgovora spremenljivke. Pri tem so uporabljene tri empirične baze podatkov. Ker se v praksi najpogosteje uporablja analizo na osnovi popolnih enot (*angl*. listwise deletion), vemo pa, da je tak pristop primeren le, če je delež manjkajočih podatkov ustrezno majhen, mehanizem za njihov nastanek pa povsem naključen (*angl*. Missing Completely at Random – MCAR), je bilo v literaturi predlaganih nekaj načinov, kako v takih primerih ravnati. Mednje poleg analize na osnovi popolnih enot sodijo analiza na osnovi razpoložljivih podatkov, vstavljanje aritmetične sredine, vstavljanje naključnih vrednosti, metoda k najbližjih sosedov, EM algoritem in večkratno vstavljanje. Te načine se je v nalogi v povezavi z metodo glavnih komponent medsebojno primerjalo, poskušalo pa se je ugotoviti tudi, kateri izmed njih je v različnih situacijah najprimernejši. Rezultati nakazujejo, da so za izvedbo metode glavnih komponent v primeru manjkajočih podatkov moderne metode za obravnavo manjkajočih podatkov, kamor uvrščamo EM algoritem in večkratno vstavljanje, večinoma primernejše od klasičnih.

KLJUČNE BESEDE: Metoda glavnih komponent, obravnava manjkajočih podatkov, metode vstavljanja, EM algoritem, večkratno vstavljanje.

ABSTRACT

Principal Component Analysis and Missing Data

It is not uncommon, especially for large data sets, for certain values of some of the variables to be missing. The most common strategy for dealing with such a situation is to completely delete any observation for which at least one of the variables has a missing value (i.e. listwise deletion). This is a satisfactory approach providing missing values are few and missing completely at random (MCAR), but clearly wasteful of information if a high proportion of observations have missing values and are not missing completely at random. To meet this problem, a number of alternative methods have been suggested. Apart from the listwise analysis these methods include: pairwise analysis, mean imputation, random values imputation, k nearest neigbours algorithm, EM algorithm and multiple imputation. The aim of the thesis is to explore the effect of these methods on the behaviour of Principal Component Analysis for univariate response pattern. The results suggest that for conducting the Principal Component Analysis modern methods, such as EM algorithm and multiple imputation, are more appropriate compared to the classic methods.

KEYWORDS: Principal Component Analysis, Treatment of Missing Data, Imputation Methods, EM algorithm, Multiple Imputation.

KAZALO VSEBINE

KAZALO TABEL

KAZALO SLIK

1 UVOD

S problemom manjkajočih podatkov se srečujemo tako v družboslovnih, vedenjskih, medicinskih kot tudi drugih znanostih; prisotni so praktično povsod, kjer se zbira podatke. Raziskovalci se pri analizi nepopolnih podatkov že desetletja zanašajo predvsem na različne »ad hoc« metode, ki skušajo podatke »popraviti« tako, da zavržejo nepopolne enote ali nadomestijo manjkajoče podatke. Te metode temeljijo na strogih predpostavkah glede vzroka za nastanek manjkajočih podatkov ter lahko povzročijo večja izkrivljenja rezultatov. »Ad hoc« metode v metodološki literaturi vse bolj izgubljajo na pomenu, medtem ko se v objavljenih znanstvenih člankih še na široko uporabljajo (Enders 2010). Tako manjkajoči podatki kot tudi neustrezne metode za njihovo obdelavo lahko povzročajo velike težave pri uporabi standardnih statističnih metod, kamor sodijo tudi multivariatne metode.

Ena od multivariatnih metod je metoda glavnih komponent, katere obnašanje v primeru manjkajočih podatkov (neodgovora spremenljivke) bomo preverili v tej nalogi. Osnovna ideja metode glavnih komponent (PCA) je zmanjšanje razsežnosti nabora podatkov, ki sestoji iz večjega števila medsebojno povezanih spremenljivk, pri čemer se poskuša ohraniti čim večji delež variabilnosti osnovnih spremenljivk.

Za izvedbo metode glavnih komponent je pogoj, da je podatkovna matrika polna, torej brez manjkajočih vrednosti. V primeru manjkajočih vrednosti večina statističnih računalniških programov enoto odstrani, četudi vrednost manjka le eni izmed obravnavanih spremenljivk; podatke torej analizira na osnovi popolnih enot. Ker tovrstna analiza pogosto vodi do pristranskih ocen statističnih parametrov (še posebno, če je delež manjkajočih vrednosti visok in podatki ne manjkajo povsem naključno), je pred izvedbo raziskave potrebno dobro premisliti, kako v čim večji meri zmanjšati verjetnost za pojav manjkajočih vrednosti.

Manjkajoči podatki prinašajo v statistično analizo vrsto težav, ki jih je v celoti nemogoče odpraviti. Dejanske vrednosti, ki obstaja – čeprav je ne poznamo –, namreč ne more zamenjati noben statistični konstrukt. Težave lahko s posebnimi postopki sicer omilimo, prava rešitev je pa seveda le v tem, da v fazi zbiranja podatkov problem preprečimo, česar običajno ni mogoče zagotoviti v celoti. Ko je zbiranje podatkov zaključeno – in podatki

dokončno manjkajo – se je treba jasno zavedati, da analiza na osnovi nepopolnih podatkov ne more biti enakovredna analizi popolnih podatkov (Vehovar 2007, 1).

V literaturi je predlaganih več različnih pristopov za ravnanje v primeru manjkajočih vrednosti, vsak izmed njih pa ima tako določene prednosti kot tudi slabosti (glej Jolliffe 2002: 363–366). Mednje sodijo analiza na osnovi popolnih enot (*angl*. listwise deletion), analiza na osnovi razpoložljivih podatkov (*angl*. pairwise deletion) ter različni načini vstavljanja podatkov: vstavljanje aritemetične sredine spremenljivk, vstavljanje naključnih vrednosti (RVI, *angl*. random values imputation), metoda k-najbližjih sosedov (KNN), večkratno vstavljanje (MICE-PMM)[1] in EM algoritem (Little in Rubin 2002; Joliffe 2002). Te pristope bomo v povezavi z metodo glavnih komponent v nalogi primerjali s pomočjo mer podobnosti in različnosti, in sicer s pomočjo koeficienta skladnosti (CC) ter korena povprečja kvadratov (RMS).

Predpostavke in omejitve

Ugotovitve naloge se nanašajo na splošno shemo manjkajočih podatkov (torej za ostale sheme, na primer monotono, ki je značilna za longitudinalne raziskave, ne veljajo) ter na neodgovor spremenljivke. Raziskava je bila narejena na treh empiričnih podatkovnih bazah, zato je pri posploševanju rezultatov potrebna določena mera previdnosti. Ker način generiranja manjkajočih podatkov lahko vpliva na rezultate simulacijske raziskave, je le-ta natančno opisan v poglavju 5.2.

Struktura naloge

Naloga je razdeljena na šest poglavij. Po uvodu (prvo poglavje) sledi opis metode glavnih komponent (drugo poglavje). Predstavljeno je matematično ozadje metode, testi za ugotavljanje smiselnosti izvedbe metode glavnih komponent ter pravila za ugotavljanje primernega števila glavnih komponent. Tretje poglavje je namenjeno manjkajočim podatkom. Obravnavana je njihova delitev, vzroki za njihov nastanek, predstavljeni pa so tudi načini

[1] Kratica je pojasnjena v poglavju 3.4.7.

obravnavanja le-teh. Četrto poglavje zajema pregled opravljenih raziskav na obravnavano temo ter predstavitev raziskovalnih vprašanj. V petem poglavju je predstavljen načrt raziskave (simulacije) ter opis podatkovnih baz. Šesto poglavje je namenjeno rezultatom raziskave, kjer so predstavljeni rezultati simulacije v tabelarični in grafični obliki ter njihova vsebinska interpretacija. Na koncu sledita razprava in zaključek (sedmo poglavje), kjer so predstavlje ključne ugotovitve naloge, odgovori na raziskovalna vprašanja ter predlogi za nadaljnje raziskovanje.

2 METODA GLAVNIH KOMPONENT

Metoda glavnih komponent (*angl.* principal component analysis, PCA) je ena najpogosteje uporabljenih multivariatnih metod. Osnovala sta jo Karl Pearson in Harold Hotelling. Osnovna ideja metode je zmanjšati razsežnost podatkov – medsebojno povezanih spremenljivk ter pri tem ohraniti čim večji del njihove skupne variabilnosti. Pri tem osnovni nabor spremenljivk spremenimo v nov nabor spremenljivk, glavnih komponent, ki so med seboj nepovezane in razvrščene tako, da jih prvih nekaj ohrani večji del variabilnosti osnovnih spremenljivk (Jolliffe 2002).

Glavnih komponent je toliko, kolikor je osnovnih spremenljivk, in so med seboj neodvisne (pravokotne). Glavne komponente se izražajo kot linearna kombinacija osnovnih spremenljivk in ohranjajo njihovo skupno variabilnost. Prva glavna komponenta je določena tako, da pojasni kar se da velik del celotne variance osnovnih spremenljivk. Druga glavna komponenta je določena tako, da je neodvisna od prve in pojasni kar se da velik del še nepojasnjene variance. Tretja glavna komponenta je neodvisna od prve in od druge glavne komponente in pojasni kar se da velik del še nepojasnjene variance itd. (Košmelj 2007).

Zaporedne glavne komponente so urejene po padajoči velikosti variance. Če so osnovne spremenljivke dovolj povezane, pojasnijo »pozne« glavne komponente majhen delež celotne variance in jih lahko zanemarimo. Bolj kot so izhodiščne spremenljivke med seboj povezane, bolj uspešna bo redukcija. Kot mero povezanosti uporabimo koeficient kovariance oz. korelacije, pri tem pa mora veljati, da je povezanost med spremenljivkami linearna (prav tam).

Matematično ozadje

Pri metodi glavnih komponent določimo uteži pri linearni kombinaciji spremenljivk tako, da je varianca te linearne kombinacije največja.

Linearne kombinacije $\mathbf{y}_j$ opazovanih spremenljivk $\mathbf{x}_j$ zapišemo kot:

$$\mathbf{y}_{j_i} = \sum_{k=1}^{p} x_{ik} a_{jk}, \qquad \begin{matrix} j=1,\ldots,p \\ k=1,\ldots,p \end{matrix} \qquad (2.1)$$

oziroma matrično:

$$\mathbf{y}_j = \mathbf{X}\mathbf{a}_j \qquad j=1,\ldots,p \qquad (2.2)$$

kjer je:

X – matrika podatkov in

$\mathbf{a}_j$ – vektor uteži.

Metoda glavnih komponent želi poiskati take uteži $\mathbf{a}_1$ za katere bo varianca $\mathbf{y}_1$ največja:

$$\text{var}(\mathbf{y}_1) = \text{var}(\mathbf{X}\mathbf{a}_1) = \max$$

Ko izračunamo prvo komponento z največjo varianco, poiščemo drugo komponento tako, da je nekorelirana s prvo in ima zopet največjo varianco, tretjo komponentno poiščemo tako, da je nekorelirana s prvo in drugo in ima spet najvišjo varianco, itd.

Lastni vektor $\mathbf{a}_1$ matrike varianc in kovarianc opazovanih spremenljivk $\boldsymbol{\Sigma}$ (v primeru standardiziranih spremenljivk je to tudi korelacijska matrika) pripada največji lastni vrednosti λ_1 iste matrike. Ta lastni vektor nam daje uteži za iskano prvo komponento, ki je tedaj enaka $\mathbf{y}_1 = \mathbf{X}\mathbf{a}_1$. Lastni vektorji so pravokotni med seboj. Naslednji lastni vektor, ki pripada naslednji največji lastni vrednosti, podaja uteži druge komponente itd.

Varianca glavne komponente $\mathbf{y}_j$ je enaka pripadajoči lastni vrednosti λ_j. Delež skupne variance, ki jo pojasni *j*-ta glavna komponenta, je $\lambda_j / sl\boldsymbol{\Sigma}$ $(sl\boldsymbol{\Sigma} = \sum_{j=1}^{p} \lambda_j)$. V primeru standardiziranih spremenljivk je delež pojasnjene skupne variance λ_j / p.

Dobljeni rezultati metode glavnih komponent so smiselni, če so lastne vrednosti pozitivna števila.

Na določanje glavnih komponent najbolj vpliva tista osnovna spremenljivka, ki ima največjo varianco. Če so osnovne spremenljivke merjene v različnih merskih enotah oz. če imajo iste merske enote in različen velikostni red vrednosti, je smiselno vpliv spremenljivk izenačiti. V takih primerih osnovne spremenljivke standardiziramo in s tem dosežemo, da imajo enaka povprečja in enake variance (povprečje 0 in varianco 1) (Košmelj 2007).

Izrek

Glavne komponente $\mathbf{y}_1, \mathbf{y}_2,\ldots,\mathbf{y}_p$ se izražajo kot linearna kombinacija izhodiščnih spremenljivk:

$\mathbf{y}_j = \mathbf{X}\mathbf{a}_j$, pri čemer velja (Košmelj 2007):

- vektorji $\mathbf{a}_j$ so lastni vektorji matrike $\mathbf{\Sigma}$. Rešujemo torej sistem: $|\mathbf{\Sigma} - \lambda \mathbf{I}| = 0$;

- varianca posamezne glavne komponente je enaka pripadajoči lastni vrednosti:

$Var(\mathbf{y}_j) = Var(\mathbf{X}\mathbf{a}_j) = \mathbf{a}_j^T \mathbf{\Sigma} \mathbf{a}_j = \lambda_j$;

- zaporedne lastne vrednosti uredimo po velikosti: $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p \geq 0$.

Smiselnost metode glavnih komponent

Za ugotavljanje smiselnosti metode glavnih komponent obstajajo različni testi, dva najpogosteje uporabljena, to sta Bartlettov test in Kaiser-Meyer-Olkinova mera (KMO), bosta v nadaljevanju tudi predstavljena. Če metodo glavnih komponent uporabljamo za pregledovalno analizo, teh testov ne uporabljamo.

Ali je povezanost spremenljivk dovolj velika, da je smiselno nadomestiti izhodiščne spremenljivke z glavnimi komponentami? Če lahko privzamemo, da so naši podatki vzorec iz populacije večrazsežno normalno porazdeljenih osnovnih spremenljivk, smemo uporabiti določene statistične teste. Ničelna domneva pravi, da je populacijska variančno-kovariančna matrika Σ diagonalna oz. da je populacijska korelacijska matrika ρ enaka enotni matriki. Metoda glavnih komponent je smiselna, če H_0 zavrnemo pri dovolj majhnem tveganju. Najbolj znan v te namene je Bartlettov test, ki temelji na χ^2 statistiki. Dejstvo je, da je omenjena predpostavka v praksi redko izpolnjena, zato je uporabnost tega testa relativno majhna (Košmelj 2007).

Druga mera za ugotavljanje smiselnosti metode glavnih komponent je Kaiser-Meyer-Olkinova mera (KMO). Metoda temelji na vrednostih korelacijskih koeficientov in parcialnih korelacijskih koeficientov (Hutcheson v Košmelj 2007) in vrednoti, ali bi spremenljivke lahko združili v skupine in posamezno skupino spremenljivk nadomestili z glavno komponento. Vrednost KMO mere je med 0 in 1, vrednosti blizu 1 kažejo, da bo redukcija uspešna, vrednosti pod 0,6 pa nakazujejo, da gre za nekoreliranost spremenljivk in neprimernost uporabe te metode (prav tam).

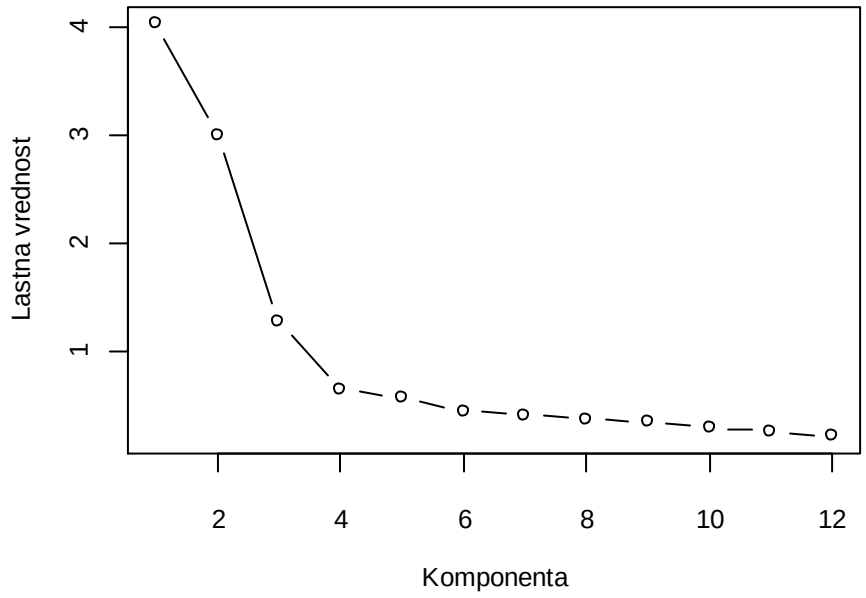
Določanje števila glavnih komponent

Ko izračunamo lastne vrednosti in lastne vektorje iz variančno-kovariančne oz. iz korelacijske matrike, se odločimo za redukcijo in privzamemo prvih *m*, $m < p$ glavnih komponent.

Določanje števila *m* je do določene mere subjektivno, pri tem se opiramo na različne hevristične postopke. Navajamo nekatere (Košmelj 2007; Ferligoj 2011, Jolliffe 2002; Jesenko in Jesenko 2007):

- vnaprej določen prag: npr. izbrano število glavnih komponent naj pojasni med 70 in 90 % skupne variabilnosti osnovnih spremenljivk;
- Kaiser-jevo pravilo: m = max(j, $\lambda_j \geq 1$). Število glavnih komponent je enako številu lastnih vrednosti, katerih vrednost je vsaj 1. Ideja, ki stoji za tem pravilom je, da vsaka glavna komponenta z lastno vrednostjo, manjšo od 1, vsebuje manjšo količino informacij kot ena izmed originalnih spremenljivk in je zato ni smiselno obdržati;
- odstotek pojasnjene variance zadnje upoštevane komponente naj bo vsaj 5;
- plaziščni diagram (*angl.* scree plot), ki prikazuje velikost lastne vrednosti glede na njeno zaporedno mesto. Lokacija »kolena« nakazuje število potrebnih komponent: do kolena vrednosti »strmo« padajo in pripadajoče glavne komponente upoštevamo, od kolena dalje so spremembe manjše in pripadajoče glavne komponente ne upoštevamo več.

Slika 2.1: Primer plaziščnega diagrama, ki ga uporabljamo za določanje potrebnega števila glavnih komponent



Na sliki 2.1, ki prikazuje plaziščni diagram, je »koleno« pri četrti glavni komponenti, kar pomeni, da bi se na podlagi plaziščnega diagrama odločili za štiri komponente.

3 MANJKAJOČI PODATKI

O **manjkajoči vrednosti** (*angl.* missing value) oziroma **manjkajočem podatku** (*angl.* missing data) govorimo, ko pri statistični enoti manjka vrednost določene spremenljivke. Na primer, anketirani v raziskavi o gospodinjstvih lahko zavrnejo odgovor na vprašanje o dohodku, v industrijskem eksperimetu so nekateri podatki manjkajoči zaradi mehaničnih okvar, nastalih v času eksperimenta, itd. (Little in Rubin 2002, 3).

Vzroki za nastanek manjkajočih podatkov

Ločimo **načrtovane** in **nenačrtovane** manjkajoče podatke. Med načrtovane manjkajoče podatke štejemo manjkajoče podatke enote, ki smo jo izločili iz vzorca, manjkajoče podatke, ki nastanejo zaradi preskokov v vprašalniku, in drugo. Za razliko od načrtovanih manjkajočih podatkov nenačrtovani niso pod nadzorom zbiralca podatkov. Primeri za nenačrtovane manjkajoče podatke so neodgovor na anketno vprašanje, zavrnitev sodelovanja in prekinitev sodelovanja pred koncem raziskave (van Buuren 2012, 29).

Neodgovori se lahko nanašajo na enote ali na spremenljivke.

Neodgovor enote (*angl.* unit non-response) pomeni manjkajoče vrednosti pri vseh spremenljivkah ene enote, zato ga lahko imenujemo tudi **popolni neodgovor** (*angl.* total non-response). Neodgovor enote je pogost v anketnih raziskavah, ko za nekatere vzorčne enote zaradi zavrnitve sodelovanja, nevzpostavitve stika ali kakšnega drugega razloga za to enoto nimamo nobenega podatka.

O **neodgovoru spremenljivke** (*angl.* item non-response) govorimo, ko pri spremenljivki manjkajo vrednosti le pri nekaterih enotah. V anketnih raziskavah se pojavi takrat, ko anketirani ne želijo odgovoriti na določena vprašanja (npr. o dohodku in drugih občutljivih temah), se ne morejo odločiti za nobeno izmed ponujenih možnosti, ali takrat, ko gre za preskok vprašanja.

V nadaljevanju naloge in simulaciji se bomo osredotočili le na neodgovor spremenljivke.

Tabela 3.1: Primeri vzrokov za nastanek manjkajočih podatkov glede na načrtovanost in neodgovor enote/neodgovor spremenljivke

	Načrtovani	**Nenačrtovani**
Neodgovor enote	vzorčenje	zavrnitev sodelovanja, vzorčenje po metodi samo-izbora
Neodgovor spremenljivke	preskok, zlivanje podatkov	neodgovor, napaka pri kodiranju

Pri manjkajočih podatkih v anketnih raziskavah v grobem ločimo nepokritje, neodgovore in druge vzroke (Vehovar 2007, 11):

- **nepokritje** (*angl.* non-coverage) vsebuje manjkajoče podatke, nastale v fazi načrtovanja raziskave. Določene enote iz ciljne populacije v vzorčni okvir sploh niso bile vključene, čeprav vanj sodijo;
- **neodgovor** (*angl.* non-response) nastane v fazi zbiranja podatkov (*angl.* field-work stage);
- **ostali vzroki manjkajočih podatkov** se nanašajo na manjkajoče podatke, nastale v drugih fazah statistične analize: urejanje podatkov, proces kontrole podatkov in obdelave podatkov.

Pred uporabo metod za obravnavo manjkajočih podatkov je smiselno razmisliti, ali »prava« vrednost, ki bi nadomestila manjkajočo vrednost, sploh obstaja. Manjkajočih vrednosti, ki na primer nastanejo zaradi preskoka v vprašalniku, ni smiselno obravnavati na enak način kot manjkajočih vrednosti, ki nastanejo zaradi zavrnitve odgovora, saj je v primeru preskoka »prava« vrednost ravno manjkajoča vrednost. Uporaba metod za obravnavo manjkajočih podatkov je torej smiselna predvsem takrat, ko »prave« vrednosti zaradi različnih razlogov nismo uspeli izmeriti.

Metode za obravnavo manjkajočih podatkov

Little in Rubin metode za obravnavo manjkajočih podatkov delita v naslednje kategorije (2002):

⇒ **metode, osnovane na popolnih enotah:** ko za nekatere enote manjka vrednost na določeni spremenljivki, je nepopolne enote najenostavneje zavreči ter analizirati le enote s popolnimi podatki. To je razmeroma enostavna strategija in je lahko uspešna pri majhnem deležu manjkajočih podatkov, vendar lahko privede do velikih pristranskosti in navadno ni zelo učinkovita, še posebej v primeru sklepanja na podpopulacije;

⇒ **metode uteževanja:** so metode za zmanjševanje pristranskosti, kjer skušamo z uteževanjem podatkov popraviti napake v ocenah, ki nastanejo zaradi manjkajočih vrednosti. Uteži dajejo nekaterim elementom v vzorcu večji relativni pomen kot drugim. Največkrat je potrebno zato, ker so bili elementi izbrani z različnimi verjetnostmi, pa tudi zaradi drugih razlogov (npr. poststratifikacija, neodgovori ipd.) (Kalton in Vehovar 2001);

⇒ **metode vstavljanja:** namesto manjkajočih vrednosti vstavimo nadomestne vrednosti, podatke pa analiziramo z uporabo standardnih statističnih metod. Sem spada t. i. *»hot deck« vstavljanje*, kjer manjkajoče vrednosti nadomestimo z obstoječimi vrednostmi drugih enot, *vstavljanje aritmetičnih sredin*, kjer manjkajoče vrednosti posamezne spremenljivke nadomestimo z aritmetično sredino vseh ne-manjkajočih vrednosti te spremenljivke, in *regresijsko vstavljanje*, kjer manjkajoče vrednosti posamezne enote ocenimo z regresijo na podlagi ne-manjkajočih vrednosti te enote;

⇒ **metode, ki temeljijo na modelskem pristopu** k statističnemu sklepanju, predpostavljajo ustrezen model za opazovane podatke (parametre takega modela se ocenjuje na podlagi različnih metod, kot je npr. metoda največjega verjetja). Osnovne prednosti takega pristopa so: prilagodljivost, izogibanje »ad hoc« metodam (predpostavke modela lahko prikažemo in ovrednotimo) ter razpoložljivost variančnih ocen ob upoštevanju dejstva, da so podatki nepopolni.

V empiričnem delu magistrske naloge (simulaciji) bodo od zgoraj opisanih metod obravnavane metode, osnovane na popolnih podatkih, metode vstavljanja in metode, ki temeljijo na modelskem pristopu (uporabljene metode so predstavljene v poglavju 3.3).

3.1 Delitev glede na mehanizem nastanka manjkajočih podatkov

Pri analizi nepopolnih podatkov se je najprej treba vprašati, zakaj podatki sploh manjkajo. Če podatki manjkajo zaradi povsem enostavnega in neškodljivega razloga, kot je spregledano vprašanje na anketnem listu, je manjkajoči podatek bolj nadloga kot problem, ki ga bi bilo treba rešiti; po drugi strani pa lahko podatki manjkajo pogojno na opazovane ali manjkajoče vrednosti. Poznavanje mehanizma za nastanek manjkajočih podatkov igra pomembno vlogo pri tem, kako bomo s temi podatki ravnali (Howell 2007, 208–209).

Rubin (1976) je definiral zelo jasno tipologijo manjkajočih podatkov, le-ta pa je postala standard za vse razprave, ki obravnavajo tovrstno tematiko. Ta tipologija sloni na vzrokih za nastanek manjkajočih podatkov. Rubin loči povsem naključno manjkajoče podatke (MCAR), naključno manjkajoče podatke (MAR) in nenaključno manjkajoče podatke (NMAR).

3.1.1 Povsem naključno manjkajoči podatki (MCAR)

Če razlog za obstoj manjkajočih podatkov ni odvisen od vrednosti ali potencialnih vrednosti določene spremenljivke, lahko rečemo, da podatki manjkajo povsem naključno – *angl*. missing completely at random (MCAR). Verjetnost za manjkajočo vrednost je torej v takem primeru neodvisna tako od njene vrednosti kot tudi od ostalih opazovanih vrednosti. Če imamo manjkajoče podatke, je to idealen primer, saj obravnava podatkov s povsem naključnimi manjkajočimi podatki MCAR ne vodi do pristranskih ocen statističnih parametrov (lahko pa se zgodi, da se moč raziskave zmanjša) (Howell 2007, 209).

PRIMERI:

- motorist (merjenec), ki se zaradi prometne nesreče ne udeleži testiranja;
- nenamerno spregledano vprašanje na anketnem listu;
- naključna izguba anketnega vprašalnika med transportom.

3.1.2 Naključno manjkajoči podatki (MAR)

Podatki pogosto ne manjkajo povsem naključno (MCAR), temveč jih lahko klasificiramo med naključno manjkajoče podatke – *angl*. missing at random (MAR). Za podatke, ki manjkajo

povsem naključno (MCAR), je verjetnost, da je x_{ij} manjkajoča, neodvisna tako od vrednosti x_{ij} kot od vrednosti drugih spremenljivk pri tej enoti. Manjkajoči podatki tipa MAR pa se pojavljajo pogojno na vrednosti opazovanih podatkov oz. ostalih (opazovanih) spremenljivk. Pogojno na vrednosti Y_{obs} (opazovane podatke) manjkajo podatki neodvisno od vrednosti Y_{mis} (manjkajočih podatkov) (Schaefer 1997; Graham 2009).

PRIMERI:

- poročeni respondenti v anketnih vprašalnikih raje podajo svoj osebni dohodek kot neporočeni in samski; verjetnost je torej odvisna od njihovega zakonskega statusa in ne od velikosti dohodka samega;
- podatki o količini padavin meteorološke postaje manjkajo v odvisnosti od hitrosti vetra; pri ekstremno močnem vetru je namreč večja verjetnost, da meteorološka postaja odpove.

3.1.3 Nenaključno manjkajoči podatki (NMAR)

Podatki manjkajo nenaključno – *angl*. not missing at random (NMAR), če nobena izmed zgornjih dveh definicij ne drži. Torej če podatki ne manjkajo vsaj MAR, manjkajo nenaključno. Verjetnost, da določena vrednost manjka, je pri NMAR odvisna od manjkajočih podatkov samih.

Ko podatki manjkajo NMAR, verjetno obstaja model, ki pojasnjuje nastanek manjkajočih podatkov. Če bi ta model poznali, bi lahko za naše podatke izračunali prave ocene parametrov. Na primer, če za osebe z nižjim osebnim dohodkom obstaja večja verjetnost, da le-tega ne bodo želele razkriti, kot za osebe z višjim osebnim dohodkom, lahko zapišemo enačbo takega modela (napovemo verjetnosti za manjkajoče vrednosti pri različnih velikostih dohodka). Tako enačbo lahko potem vključimo v kompleksejši model, s pomočjo katerega ocenimo manjkajoče vrednosti. Na žalost pa le redkokdaj vemo, kakšen je ta model (s katerim napovemo verjetnosti za manjkajoče vrednosti), zato ne moremo vedeti, kako nadaljevati. Poleg tega pa je vključitev modela pogosto težavna naloga in jo je treba pri vsakem vstavljanju prilagoditi (Howell 2007, 209).

Za uspešno reševanje NMAR je torej treba najti dodatne informacije o vzrokih za manjkajoče

podatke (kar je pogosto nemogoče) ali pa izvesti analizo tipa »kaj-če« (*angl*. what-if analysis), da ugotovimo, kako občutljivi so rezultati ob različnih scenarijih (van Burren 2012, 7).

PRIMERI:

- neudeležba preiskovancev v kliničnem preizkusu zaradi zdravstvenega stanja, povezanega s preizkusom – npr. zaradi hude depresije se preiskovanec ne udeleži ocenjevnja stopnje depresije (Palfy 2009, 43);
- osebe z višjim osebnim dohodkom v anketnih vprašalnikih raje podajo svoj osebni dohodek kot osebe z nižjim osebnim dohodkom; verjetnost je torej odvisna od velikosti dohodka samega.

3.1.4 Zanemarljiv in nezanemarljiv mehanizem za nastanek manjkajočih podatkov

Ko je mehanizem za nastanek manjkajočih podatkov NMAR, je analiza takih podatkov bolj zapletena kot v primeru MAR ali MCAR. V primeru NMAR lahko rečemo, da je mehanizem za nastanek manjkajočih podatkov **nezanemarljiv**. To pomeni, da ne smemo računati parametrov našega modela, če pred tem ne zapišemo modela, ki pojasnjuje način nastanka manjkajočih podatkov. Po drugi strani pa je mehanizem za nastanek manjkajočih podatkov **zanemarljiv**, če zadošča naslednjima dvema pogojema: 1. podatki manjkajo vsaj naključno – MAR; 2. parametri, ki določajo neodgovor, so neodvisni od parametrov, ki jih ocenujujemo in ki določajo porazdelitev nepopolne slučajne spremenljivke (*angl*. distinctness). Če je mehanizem zanemarljiv, lahko z analizo nadaljujemo, saj nam ni treba zapisati modela, ki pojasnjuje način nastanka manjkajočih podatkov, kar pa še ne pomeni, da lahko problem manjkajočih podatkov zanemarimo, saj imamo še zmeraj dovolj dela s tem, da poskušamo izračunati čim boljše ocene parametrov našega modela (Howell 2007; Rubin 1976).

Zanemarljiv mehanizem torej ne pomeni, da smo lahko glede manjkajočih podatkov popolnoma brezbrižni. Vzemimo primer mehanizma MAR, ko vrednosti spremenljivke $\mathbf{x}_2$ manjkajo pogojno na vrednosti spremenljivke $\mathbf{x}_1$. Veljavne ocene aritmetične sredine spremenljivke $\mathbf{x}_2$ ne moremo izračunati brez $\mathbf{x}_1$, kar pomeni, da moramo vrednosti spremenljivke $\mathbf{x}_1$ vključiti v izračun aritmetične sredine spremenljivke $\mathbf{x}_2$ (van Buuren 2012, 33).

3.2 Sheme manjkajočih podatkov

Poleg različnih mehanizmov za nastanek manjkajočih podatkov ločimo tudi različne sheme manjkajočih podatkov (*angl.* patterns of missing data). Manjkajoče vrednosti so lahko omejene na določene enote ali spremenljivke, lahko se pojavljajo slučajno, lahko pa zavzamejo tudi bolj urejene sheme. Medtem ko se mehanizmi za nastanek manjkajočih podatkov nanašajo na povezanost med (potencialnimi) vrednostmi spremenljivk in manjkajočimi vrednostmi, sheme manjkajočih podatkov ponazarjajo, katere vrednosti v podatkovni matriki so opazovane in katere manjkajoče (Little in Rubin 2002; Vehovar 2007).

Glede na shemo pojavljanja manjkajočih podatkov (glej sliko 3.1) Little in Rubin (2004, 4–8) ločita:

- neodgovor spremenljivke (*angl.* Univariate Nonresponse Pattern): manjkajoče vrednosti so omejene na eno samo spremenljivko. To je tudi eden izmed prvih problemov, ki se nanašajo na manjkajoče vrednosti, deležen pozornosti v statistični literaturi;

- neodgovor enote (*angl.* Multivariate Two Patterns oz. Unit Nonresponse Pattern): je zelo pogost v anketnih raziskavah, ko za nekatere vzorčne enote zaradi zavrnitve sodelovanja, nevzpostavitve stika ali kakšnega drugega razloga nimamo nobenega podatka, razen podatkov, ki smo jih pridobili pri načrtovanju raziskave (npr. lokacija gospodinjstva, velikost gospodinjstva ipd.);

- monotono shemo (*angl.* Monotone Pattern): tipično je povezana z longitudinalnimi raziskavami, ko sodelujoči »izstopijo« pred koncem raziskave (npr. zbolijo, imajo negativno reakcijo na določeno zdravilo, se odselijo ipd.), ter anketnimi raziskavami, ko respondenti pri določenem vprašanju prenehajo odgovarjati. Vizualno spominja na stopnice (glej sliko 3.1c);

- splošno shemo (*angl.* General Pattern): manjkajoče vrednosti so razpršene po celotni podatkovni matriki na videz naključno. Na videz naključen vzorec manjkajočih vrednosti je lahko zavajajoč, saj vrednosti lahko še zmeraj manjkajo sistematično (glej sliko 3.1d);

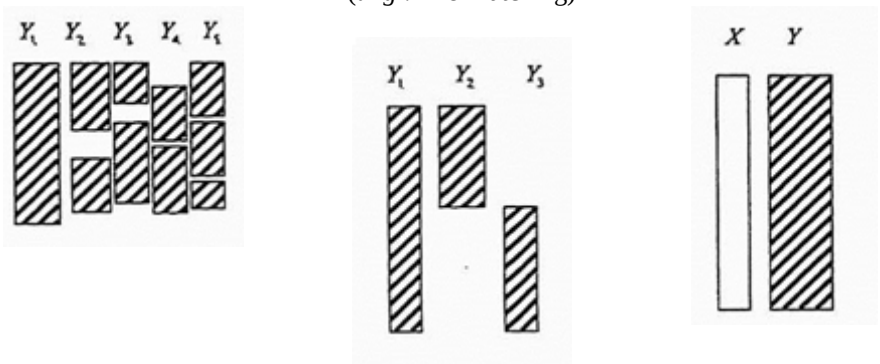
- ujemanje podatkov, ko dva niza spremenljivk nista nikoli opazovana skupaj (*angl*. File Matching, with Two Sets of Variables Never Jointly Observed): pri velikih količinah manjkajočih vrednosti je verjetnost, da spremenljivke niso nikoli opazovane skupaj, večja. Ko se to zgodi, se je treba problema jasno zavedati, saj lahko ocenjevanje nekaterih parametrov, ki se nanašajo na asociacijo med temi spremenljivkami, privede do napačnih oz. zavajajočih rezultatov. Problem je prisoten tudi pri metodah zlivanja podatkov (*angl.* data fusion methods);

- latentno spremenljivko (*angl.* Latent-Variable Pattern): včasih je neopazovane »latentne« spremenljivke smiselno obravnavati kot spremenljivke z manjkajočimi vrednostmi ter uporabiti ideje iz teorije manjkajočih vrednosti za ocenjevanje parametrov.

Slika 3.1: Sheme manjkajočih podatkov

(a) Neodgovor spremenljivke (b) Neodgovor enote (c) Monotona shema

(d) Splošna shema (e) Ujemanje podatkov (*angl*. File Matching) (f) Faktorska analiza



Vir: Little in Rubin 2004, 5.

Izbira metod za obravnavo manjkajočih podatkov ni odvisna le od mehanizma za nastanek manjkajočih podatkov, upoštevati je treba tudi njihovo shemo. V empiričnem delu naloge

smo se osredotočili na splošno shemo manjkajočih podatkov, kar pomeni, da rezultatov simulacije na ostale sheme ne moremo posplošiti.

Spodaj (glej sliko 3.2) je prikazan primer splošne sheme manjkajočih podatkov in lastnosti, s katerimi opišemo takšno podatkovno matriko.

Slika 3.2: Splošna shema manjkajočih podatkov

	spr1	**spr2**	**spr3**	**spr4**	**spr5**
e1					
e2		NA[2]			
e3					
e4			NA		
e5					
e6	NA				
e7					NA
e8		NA		NA	
e9					
e10					

Lastnosti, s katerimi opišemo takšno matriko:

- spremenljivke: spr1, spr2,…, spr5;
- enote: e1, e2,…, e10;
- število podatkov v matriki: 10 (enot) * 5 (spremenljivk) = 50;
- število manjkajočih podatkov v matriki: 6;
- odstotek manjkajočih podatkov v matriki: 6/50 * 100 = 12 %;
- število enot z manjkajočimi podatki v matriki: 5;
- odstotek enot z manjkajočimi podatki v matriki: 5/10 * 100 = 50 %.

[2] NA (*angl.* not available) je uveljavljen izraz za manjkajoče podatke.

3.3 Metode za obravnavo manjkajočih podatkov

Obstaja več različnih pristopov za obravnavo manjkajočih podatkov. S časom novi pristopi zamenjajo stare, vendar se ta proces odvija zelo počasi. Pogosto se torej starejše metode še naprej uporabljajo, po navadi zato, ker poteče precej časa, preden so novejše metode povsem razumljive in se jih vključi v učne načrte predmetov ter učbenike, včasih pa je potrebna tudi »zamenjava« generacij, če »stare« generacije ne osvežujejo svojega znanja. V tem poglavju bomo predstavili naslednje metode za obravnavo manjkajočih podatkov: analizo na osnovi popolnih enot, analizo na osnovi razpoložljivih podatkov, vstavljanje naključnih vrednosti glede na porazdelitev spremenljivke, vstavljanje aritmetične sredine, metodo k-najbližjih sosedov, EM algoritem in večkratno vstavljanje.

Ko manjkajočih vrednosti ne želimo upoštevati, imamo na voljo dve zelo enostavni možnosti: analizo na osnovi popolnih enot (*angl*. listwise deletion) in analizo na osnovi razpoložljivih podatkov (*angl*. pairwise deletion).

3.3.1 Analiza na osnovi popolnih enot

Analiza na osnovi popolnih enot statistično analizo omeji na niz enot, ki so brez manjkajočih vrednosti (take enote lahko imenujemo tudi popolne enote). Če želimo npr. izračunati variančno-kovariančno matriko za spremenljivke $\mathbf{x}_1$, $\mathbf{x}_2$,…, $\mathbf{x}_p$, bo analiza na osnovi popolnih enot izpustila vsako enoto, ki ima manjkajočo vrednost na vsaj eni izmed spremenljivk $\mathbf{x}_1$, $\mathbf{x}_2$,…, $\mathbf{x}_p$. Ta metoda je privzeta v mnogih statističnih programih (Schafer in Graham 2002, 155).

Prednost analize na osnovi popolnih enot je v tem, da je (1) enostavna in (2) omogoča primerljivost univariatnih statistik, saj so izračunane na istem vzorcu, njena slabost pa je potencialna izguba informacij in neupoštevanje nepopolnih enot. Izguba informacij ima dva vidika: izgubo preciznosti ter pristranskost, ko mehanizem za nastanek manjkajočih podatkov ni MCAR. Analiza na osnovi popolnih enot je lahko upravičena v smislu enostavnosti izvedbe takrat, ko sta izguba preciznosti ter pristranskost minimalni. To je bolj verjetno takrat, ko je delež popolnih enot visok, vendar pa stopnja pristranskosti in izguba preciznosti nista odvisni le od deleža popolnih enot in sheme manjkajočih podatkov (*angl*. pattern of

missing data), temveč tudi od podobnosti popolnih in nepopolnih enot ter od parametrov, ki nas zanimajo (Little in Rubin 2002, 41–42).

Analiza na osnovi popolnih enot je potencialno potratna pri univariatnih analizah, kot so ocene povprečij ter marginalnih frekvenčnih porazdelitev, saj so vrednosti spremenljivk zavržene, če pripadajo enotam z manjkajočimi vrednostmi na drugih spremenljivkah. Pri podatkovnih bazah z veliko spremenljivkami je lahko ta izguba zelo velika. Že v primeru, ko vrednosti vsake od 20 spremenljivk v podatkovni bazi manjkajo z 10 % verjetnostjo, je pričakovan delež popolnih enot $0.9^{20} \cong 0.12$, kar pomeni, da obdržimo le približno 12/0.9 = 13 % vseh vrednosti (Little in Rubin 2002, 53).

3.3.2 Analiza na osnovi razpoložljivih podatkov

Primer: za oceno standardnega odklona spremenljivke $\mathbf{x}_j$ lahko uporabimo vsako opazovano vrednost spremenljivke $\mathbf{x}_j$, za oceno kovariance med spremenljivkama $\mathbf{x}_j$ in $\mathbf{x}_k$ pa vsak opazovan par vrednosti spremenljivk $\mathbf{x}_j$ in $\mathbf{x}_k$. Problem analize na osnovi razpoložljivih podatkov je v tem, da rezultati slonijo na različnih sklopih podatkov ter različnih velikostih vzorcev. Ker parametri slonijo na različnih sklopih podatkov, je težko izračunati standardne napake ter ostale mere tveganja (Schafer in Graham 2002, 155).

Ker analiza na osnovi razpoložljivih podatkov upošteva vse podatke, ki so na voljo, bi lahko pričakovali, da je učinkovitejša od analize na osnovi popolnih enot. Kim in Curry (1977) sta ugotovila, da je to res takrat, ko je mehanizem za nastanek manjkajočih podatkov MCAR, povezanost med spremenljivkami pa šibka. Po rezultatih nekaterih drugih raziskav (Haitovsky 1968; Azen in Van Guilder, 1981) pa je analiza na osnovi popolnih enot učinkovitejša, ko je povezanost med spremenljivkami visoka. Seveda pa nobena izmed obeh metod v splošnem ni zadovoljiva (Little in Rubin 2002, 55).

3.3.3 Vstavljanje aritmetične sredine

Vstavljanje aritmetične sredine nadomesti vse manjkajoče vrednosti posamezne spremenljivke z aritmetično sredino vseh obstoječih vrednosti te spremenljivke. Tako ohranja

aritmetično sredino spremenljivke nespremenjeno, ne glede na odstotek manjkajočih vrednosti (Graham 2003).

Vstavljanje aritmetične sredine je hitra in enostavna rešitev za manjkajoče podatke, vendar zelo pogosto popolnoma neprimerna. Pri vstavljanju aritmetične sredine se vrednosti gostijo okrog aritmetične sredine, zato je naravna posledica takega vstavljanja podcenjevanje vrednosti variance. Ker ta metoda pokvari porazdelitev vzorčnih vrednosti, tudi nekatere druge ocene, kot so kvantili ali mere oblik frekvenčnih porazdelitev, niso ocenjene pravilno, če uporabljamo standardne formule za popolne podatke (glej Little in Rubin 2002, 61). Tudi ko imamo opravka z asimetrično porazdelitvijo vrednosti spremenljivke, je vstavljanje aritmetične sredine nesprejemljivo, vpliva pa tudi na odnose med spremenljivkami, saj »vleče« ocene za korelacijske koeficiente proti 0.

Ta metoda je ena izmed najslabših, če ne celo najslabša; upravičena je le za ocenjevanje aritmetične sredine v primeru, ko je mehanizem za nastanek manjkajočih podatkov MCAR (Little in Rubin 2002).

Pogosto vstavljamo namesto manjkajoče vrednosti razpoložljive vrednosti (*angl.* »hot-deck« imputation). Pri vseh enostavnih oblikah tovrstnega vstavljanja imamo za manjkajoče vrednosti potreben pogoj MCAR. Postopki se med seboj razlikujejo predvsem v načinu izbire nadomestnih vrednosti, za kar imamo več možnosti: enostavna slučajna izbira, zaporedno vstavljanje, vstavljanje podatkov znotraj razredov, vstavljanje najbližje enote, metoda k-najbližjih sosedov (Vehovar 2007, 90–92).

3.3.4 Vstavljanje naključnih vrednosti glede na porazdelitev spremenljivke

Pri tej preprosti metodi se iz obstoječih podatkov najprej ocenijo parametri (običajno normalne) porazdelitve, nato pa se na manjkajočih mestih generirajo naključne vrednosti s to porazdelitvijo (Denk in Weber 2011, 8; Gelman in Hill 2007, 534).

Nekoliko drugačna in prepostejša različica zgoraj opisane metode je, da iz obstoječih vrednosti neke spremenljivke naključno izbiramo vrednosti s ponavljanjem, ki jih vstavljamo na manjkajoča mesta. Njena prednost je v tem, da vrača samo legalne vrednosti iz zaloge

vrednosti posamezne spremenljivke (uvrščamo jo med »hot deck« metode) in zato za razliko od vstavljanja aritmetičnih sredin v večji meri ohrani obliko porazdelitve posamezne spremenljivke. Pri prvotno zastavljeni metodi se lahko zgodi, da vstavljamo vrednosti, ki niso naravna števila (kot bi na podlagi vprašalnika npr. morala biti), ali celo negativne vrednosti (neskončna repa normalne porazdelitve) (Little in Rubin 2002, 67–68; Schafer in Graham 2002, 159). Interna oznaka za to metodo vstavljanja naj bo RVI (*angl.* random values imputation method).

Ena od slabosti te metode je, podobno kot pri ostalih klasičnih pristopih, vstavljanje manjkajočih vrednosti za vsako spremenljivko posebej; metoda torej ne upošteva multivariatnega vidika vstavljanja kot npr. EM algoritem in večkratno vstavljanje (prav tam). Njeno uspešnost bomo preverili tudi s simulacijo v tej nalogi.

3.3.5 Metoda k-najbližjih sosedov

Metoda k-najbližjih sosedov (*angl.* K Nearest Neigbor (KNN) Algorithm) najbližje sosede izbere na podlagi mere podobnosti med enotami iz celotne podatkovne baze, če izvzamemo enote, ki imajo manjkajočo vrednost na enakem mestu kot enota, katere manjkajočo vrednost vstavljamo (Kim in drugi 2004). Med merami podobnosti je nabolj poznana Evklidska razdalja (poznamo pa še druge razdalje, kot so Manhatanova razdalja, Mahalanobisova razdalja, Pearsonova razdalja in druge). Evklidsko razdaljo med dvema enotama U in V, ki sta opisani s p spremenljivkami, izračunamo po formuli

$$d(U,V) = \sqrt{\sum_{i=1}^{p} (u_i - v_i)^2}\ , \qquad (3.1)$$

kjer je
p število spremenljivk;
u_i vrednost i-te spremenljivke prve enote (U) in
v_i vrednost i-te spremenljivke druge enote (V).

Dobljena vrednost d(U,V) je Evklidska razdalja med enotama U in V. Bolj kot sta si enoti podobni, manjša je razdalja med njima; med dvema popolnoma skladnima enotama bi bila ta razdalja enaka 0.

Ko s pomočjo ustrezne razdalje določimo najbližje sosede, vstavimo manjkajočo vrednost. Manjkajoča vrednost je zapolnjena z uteženo povprečno vrednostjo ustrezne spremenljivke k najbližjih enot (prav tam). Utež izračunamo po spodnji formuli (3.2):

$$W_i = \frac{\frac{1}{D_i}}{\sum_{i=1}^{k} \frac{1}{D_i}} \qquad (3.2)$$

kjer je

k število izbranih najbližjih sosedov (enot) ter

D_i razdalja med i-to enoto in enoto, katere manjkajočo vrednost vstavljamo.

Zaporedna metoda k-najbližjih sosedov

V simulaciji je bila uporabljena zaporedna metoda k-najbližjih sosedov, ki je posebna različica (nadgradnja) klasične metode k-najbližjih sosedov. Manjkajoče vrednosti vstavlja zaporedno: za vstavljanje uporabi najprej enote, ki imajo najmanj manjkajočih vrednosti, vstavljene vrednosti pa uporabi za nadaljnja vstavljanja. Čeprav metoda za vstavljanje uporablja že vstavljene vrednosti, deluje bolje od klasične metode k-najbližjih sosedov (Kim in drugi 2004).

Zaporedna metoda k-najbližjih sosedov se od običajne razlikuje v dveh točkah (prav tam):

1. pri zaporedni metodi k-najbližjih sosedov so enote razvrščene glede na delež manjkajočih vrednosti, vstavljanje je izvršeno zaporedno, pri čemer se za vstavljanje najprej uporabi enote z najmanj manjkajočimi vrednostmi. Zaporedno vstavljene enote se potem uporabi za nadaljnja vstavljanja. Algoritem podatkovno bazo razdeli na dva dela glede na to, ali enote vsebujejo manjkajoče vrednosti ali ne. Ko so vse manjkajoče vrednosti neke enote vstavljene, le-to premakne v del podatkovne baze brez manjkajočih vrednosti ter jo uporabi za nadaljnja vstavljanja;
2. zaporedna metoda k-najbližjih sosedov vse manjkajoče vrednosti posamezne enote vstavi hkrati (in je zato precej hitrejša), medtem ko klasična metoda išče najbližje sosede za vsako manjkajočo vrednost posebej.

Prednosti in slabosti metode k-najbližjih sosedov

Med prednostmi metode k-najbližjih sosedov Acuna in Rodriguez (2011, 4) omenjata naslednje: (1) primerna je tako za številske kot tudi opisne spremenljivke, (2) ni treba skonstruirati modela za manjkajoče podatke in (3) upošteva korelacijsko strukturo podatkov. Slabosti pa so: (1) subjektiven izbor merske razdalje (Evklidska, Manhatanova,

Mahalanobisova, Pearsonova in druge); (2) ker metoda išče najbolj podobne enote, je pri velikih podatkovnih bazah zelo počasna in (3) subjektiven izbor števila (k) najbližjih sosedov (če je izbrani k prevelik, algoritem pri napovedovanju manjkajoče vrednosti uporabi tudi enote, ki enoti z manjkajočo vrednostjo niso dovolj podobne ter s tem zmanjša natančnost ocene napovedane vrednosti).

3.3.6 EM algoritem za variančno-kovariančne matrike

EM algoritem je postopek, s pomočjo katerega izračunamo ocene za željene parametre po metodi največjega verjetja (v nadaljevanju ML, *angl.* Maximum Likelihood). Sestavljen je iz dveh korakov, in sicer pričakovanega ali E-koraka (*angl.* Expectation step) in maksimizacijskega ali M-koraka (angl. Maximization step). EM algoritem za večrazsežno normalno porazdelitev je iterativen postopek, kjer se izmenjujeta E- in M-korak (Graham 2012; Schafer 1997):

- E-korak: zamenja manjkajoče statistike z njihovimi pričakovanimi vrednostmi glede na opazovane podatke, ocenjene vrednosti uporabi kot parametre;
- M-korak: posodobi parametre z njihovimi ML ocenami glede na statistike iz E-koraka.

E-KORAK

Zadostne statistike za EM algoritem za variančno-kovariančno matriko so vsote, vsote kvadratov in vsote navzkrižnih produktov. Algoritem prebere vsako enoto podatkovne matrike posebej in vsakič posodobi statistike. Pri enotah brez manjkajočih vrednosti za izračun uporabi obstoječe vrednosti, pri enotah z manjkajočimi vrednostmi pa najprej izračuna najboljše ocene za manjkajoče vrednosti. Najboljša ocena za posamezno manjkajočo vrednost je regresijska ocena, kjer vse ostale spremenljivke nastopajo kot prediktorji. Za posodobitev vsote kvadratov in vsote navzkrižnih produktov, ko sta obravnavani vrednosti posamezne enote manjkajoči, se najboljši oceni doda korekcijski člen; korekcijski člen je ostanek pri regesiji, kjer vse ostale spremenljivke nastopajo kot prediktorji (Graham 2012, 54).

M-KORAK

Ko so vsote, vsote kvadratov in vsote navzkrižnih produktov ocenjene, lahko variančno-kovariančno matriko in vektor aritmetičnih sredin enostavno izračunamo. S tem se zaključi prva iteracija.

S pomočjo ocenjene variančno-kovariančne matrike in povprečij iz prve iteracije lahko izračunamo vse regresijske ocene, potrebne za napovedovanje vseh spremenljivk modela. Pri naslednji iteraciji te regresijske ocene uporabimo za posodobitev »najboljših ocen« za manjkajoče vrednosti. Ko so vsote, vsote kvadratov in vsote navzkrižnih produktov te iteracije izračunane, se izračuna novo variančno-kovariančno matriko in vektor aritmetičnih sredin ter oceni nove regresijske ocene za naslednjo iteracijo. Postopek ponavljamo toliko časa, dokler se variance, kovariance in aritmetične sredine med iteracijami razlikujejo tako malo, da lahko rečemo, da se ne spreminjajo več; takrat je EM algoritem skonvergiral. Variančno-kovariančna matrika in vektor aritmetičnih sredin z zadnje iteracije so končne ocene za te parametre. Te parametre lahko uporabimo za vsako statistično analizo, za katero kot vhodne parametre potrebujemo le variančno-kovariančno matriko in vektor aritmetičnih sredin (Graham 2012, 54–55).

EM algoritem izdela nepristranske oz. skoraj nepristranske ocene za aritmetično sredino, varianco in kovarianco. Še ena pozitivna lastnost EM algoritma je, da tudi če predpostavka o večrazsežni normalni porazdelitvi ne drži, izgleda, da dela izjemno dobro (Howell 2007, 219). Med negativne lastnosti EM algoritma lahko štejemo njegovo počasnost, posebno pri visokem deležu manjkajočih podatkov, ter neprimernost za testiranje hipotez, saj ne izračuna standardnih napak.

3.3.7 Večkratno vstavljanje

Večkratno vstavljanje ima veliko prednosti pred ostalimi metodami za obravnavo manjkajočih podatkov. Večkratno vstavljanje nadomesti manjkajoče vrednosti večkrat, tako da dobimo več »popolnih« matrik. Manjkajoče vrednosti nadomešča na podlagi opazovanih vrednosti posamezne enote in opazovanih relacij za ostale enote ob predpostavki, da so opazovane vrednosti vključene v model vstavljanja. Postopki večkratnega vstavljanja,

predvsem pa metoda MICE, so zelo fleksibilni in se lahko uporabljajo v različnih situacijah. Ker večkratno vstavljanje obsega ustvarjanje večkratnih napovedi za vsako manjkajočo vrednost, analiza večkratno vstavljenih podatkov upošteva negotovost vstavljanj in izdela točne standardne napake. Enostavno povedano, če opazovani podatki nudijo malo informacij o manjkajočih vrednostih, bodo vstavljene vrednosti zelo raznolike, kar bo privedlo do velikih standardnih napak pri analizah. Če pa opazovani podatki dobro napovedujejo manjkajoče vrednosti, bodo vstavljene vrednosti bolj skladne, kar bo privedlo do manjših in točnejših standardnih napak (Azur in drugi 2011, 40–41).

Večkratno vstavljanje z verižnimi enačbami (*angl.* The Chained Equation Approach to Multiple Imputation, v nadaljevanju MICE) je posebna metoda večkratnega vstavljanja. Metoda se je izkazala kot izredno uspešna v različnih simulacijskih raziskavah. Predpostavka MICE metode je, da manjkajoči podatki spremenljivk, uporabljenih v postopku vstavljanja, manjkajo naključno (MAR); izvajanje MICE v primeru nenaključno manjkajočih podatkov (NMAR) lahko privede do pristranskih ocen (van Buuren in Groothuis-Oudshoorn 2011, 7; Azur in drugi 2011, 41).

Mnoge prvotno razvite metode večkratnega vstavljanja predpostavljajo obsežen skupni model za vse spremenljivke, kot je npr. večrazsežna normalna porazdelitev. V primeru velikih podatkovnih baz z več sto spremenljivkami različnih tipov je takšna predpostavka redko ustrezna. MICE je alternativen in prilagodljiv pristop, ki ne predpostavlja večrazsežne normalne porazdelitve. Uporabljen je bil v podatkovnih bazah z več tisoč enotami in več sto spremenljivkami. MICE postopek temelji na vrsti regresijskih modelov, pri čemer je vsaka spremenljivka z manjkajočo vrednostjo modelirana pogojno v odvisnosti od ostalih spremenljivk. To pomeni, da lahko za vsako spremenljivko glede na njeno porazdelitev uporabimo drugačen model (npr. za dihotomne spremenljivke logistično regresijo, za zvezne pa linearno regresijo) (Azur in drugi 2011, 41–42).

MICE koraki

Postopek verižnih enačb lahko ponazorimo z naslednjimi koraki (Azur in drugi 2011, 42):

- Korak 1: za vsako manjkajočo vrednost v podatkovni bazi se izvede enostavno vstavljanje, kot je npr. vstavljanje aritmetične sredine.
- Korak 2: vstavljene vrednosti ene spremenljivke (»var«) nastavimo nazaj kot manjkajoče.

- Korak 3: opazovane vrednosti spremenljivke »var« s koraka 2 uporabimo v regresijskem modelu, kjer je »var« odvisna spremenljivka in vse ostale spremenljivke so neodvisne. Regresijski model tukaj deluje na enak način in pod enakimi predpostavkami, kot če bi izvedli linearni, logistični ali Poissonov regresijski model brez vstavljanja manjkajočih vrednosti.
- Korak 4: manjkajoče vrednosti spremenljivke »var« nadomestimo z napovedmi regresijskega modela. Ko spremenljivko »var« pozneje uporabimo kot neodvisno spremenljivko v regresijskih modelih (za nadomeščanje manjkajočih vrednosti ostalih spremenljivk), uporabimo tako njene opazovane kot tudi vstavljene vrednosti.
- Korak 5: korake 2–4 ponovimo za vsako spremenljivko z manjkajočimi podatki – ko ta postopek zaključimo, je zaključena ena iteracija oziroma en cikel. Na koncu cikla so vse manjkajoče vrednosti podatkovne matrike nadomeščene s pomočjo regresijskih modelov, ki odražajo opazovane povezanosti med spremenljivkami.
- Korak 6: korake 2–4 ponavljamo, z vsakim ciklom posodobimo vstavljene vrednosti. Na koncu obdržimo vstavljene vrednosti zadnjega cikla. Število ciklov določi raziskovalec sam; navadno je dovolj deset ciklov, vendar je za optimalno število ciklov potrebno natančno spremljanje rezultatov vstavljanja pod različnimi pogoji. Parametri, na podlagi katerih vstavljamo (npr. koeficienti regresijskega modela), morajo na koncu ciklov konvergirati tako, da postanejo stabilni. S tem se na primer izognemo vplivu vrstnega reda vstavljanja spremenljivk na vstavljene vrednosti. V praksi lahko raziskovalci konvergenco preverijo tako, da primerjajo regresijske modele med različnimi cikli. Obstaja sicer več različic metode MICE, vendar so si v osnovi zelo podobne.

Optimalno število popolnih podatkovnih matrik

Ko zaključimo z vnaprej določenim številom ciklov, smo izdelali eno popolno podatkovno matriko. Če želimo ustvariti več popolnih podatkovnih matrik, moramo celoten proces vstavljanja ponavljati (v simulaciji smo izdelovali le po eno podatkovno matriko). Opazovani podatki so v vseh izdelanih popolnih podatkovnih matrikah enaki; le vrednosti, ki so bile prvotno manjkajoče, so različne. Začetne raziskave so priporočale izdelavo 5–10 popolnih podatkovnih matrik; najnovejše raziskave pa ugotavljajo, da lahko (odvisno tudi od količine manjkajočih informacij v podatkih) izdelava do 40 popolnih podatkovnih matrik občutno poveča statistično moč. Žal v praksi vstavljanje v takšnem obsegu (npr. 40 podatkovnih matrik) ni vedno izvedljivo. Glede na velikost modela vstavljanja in razpoložljive

računalniške zmožnosti lahko za vstavljanje manjkajočih vrednosti ene podatkovne matrike potrebujemo nekaj minut ali celo ur. Graham in drugi (2007) navajajo rezultate simulacije, s katerimi si lahko pomagamo pri odločanju glede optimalnega števila popolnih podatkovnih matrik. Pri tem so ključnega pomena velikost podatkovne baze, količina manjkajočih podatkov in hitrost računalnika. Na primer, izdelava ene popolne matrike z več sto spremenljivkami, več tisoč enotami in med 5 in 80 % manjkajočih podatkov lahko traja nekaj ur, zato je lahko nepraktično izdelati 40 popolnih matrik; nasprotno pa je izdelava ene popolne matrike z 20 spremenljivkami in več sto enotami lahko izvedena v nekaj minutah, zato je izdelava 40 popolnih matrik popolnoma izvedljiva (Azur in drugi 2011, 43).

Izbira prediktorjev

Pri manjših in srednje velikih podatkovnih bazah s približno 20–30 spremenljivkami je kot prediktorje za posamezne spremenljivke najprimerneje izbrati vse ostale spremenljivke v podatkovni bazi, saj kot splošno pravilo velja, da le uporaba vseh razpoložljivih informacij vodi do čim bolj točnih in nepristranskih večkratnih vstavljanj. Pri večjih podatkovnih bazah, ki lahko obsegajo tudi po več sto spremenljivk in na tisoče enot, pa je pri izboru prediktorjev s ciljem, da se izognemo multikolinearnosti in drugim računskim problemom, smiselno upoštevati določena priporočila (glej van Buuren 2012, 128).

Metoda PMM

Metoda MICE zahteva, da se določi model vstavljanja za vsako spremenljivko z manjkajočimi vrednostmi. Za številske spremenljivke so na voljo metoda PMM, linearna regresija, Bayesova linearna regresija in vstavljanje aritmetične sredine.

Metoda PMM (*angl.* predictive mean matching), ki je bila uporabljena tudi v simulaciji, je metoda vstavljanja za številske (zvezne) spremenljivke. Podobna je regresijskemu vstavljanju, le da vsako manjkajočo vrednost nadomesti naključno iz množice k opazovanih vrednosti, katerih napovedane vrednosti so najbližje napovedani vrednosti za manjkajočo vrednost (Schenker in Taylor 1996).

Pri metodi PMM je treba določiti število najbližjih opazovanih vrednosti k (navadno 1, 3 ali 10). Metoda pri nizkih k deluje slabo (manjši kot je izbrani k, višja je korelacija med večkratnimi vstavljanji za posamezno manjkajočo vrednost), medtem ko velik k lahko privede do pristranskih cenilk (van Buuren 2012, 71; Schenker in Taylor 1996, 430).

Metoda PMM je »hot deck« metoda, saj manjkajočo vrednost nadomesti z obstoječo vrednostjo bližnje popolne enote. Ker linearno regresijo uporabi le za določanje razdalje med enotami, je manj občutljiva, ko so predpostavke modela kršene. Ena izmed prednosti metode PMM je tudi ta, da vstavlja le realne oz. obstoječe vrednosti (Schenker in Taylor 1996, 429).

Metoda MICE-PMM je najbolj priporočljiva metoda večkratnega vstavljanja, ko ima manjkajoče podatke manj kot 50 % enot in ko mehanizem za nastanek manjkajočih podatkov ni NMAR (Marshall in drugi 2010).

4 PREGLED OPRAVLJENIH RAZISKAV IN RAZISKOVALNE HIPOTEZE

4.1 PREGLED OPRAVLJENIH RAZISKAV IN ŠIRŠE LITERATURE

Metodologi se s problemom manjkajočih podatkov ukvarjajo že desetletja in v tem času so razvili veliko metod za njihovo obravnavo; nekatere od teh metod se na široko uporablja, nekatere pa so že utonile v pozabo. V objavljenih znanstvenih člankih in statističnih programih se še zmeraj večinoma srečujemo s t. i. klasičnimi metodami, ki pa v metodološki literaturi vse bolj izgubljajo na pomenu (Little in Rubin 2002 v Enders 2010). V tem podpoglavju bodo predstavljene ključne ugotovitve opravljenih raziskav, ki obravnavajo metode za obravnavo manjkajočih podatkov (tako klasične kot moderne) in so uporabljene v naši simulacijski raziskavi.

Analiza na osnovi popolnih enot in analiza na osnovi razpoložljivih podatkov sta najbolj pogosto uporabljeni metodi za obravnavo manjkajočih podatkov v družboslovnih vedah. Njuni prednosti sta enostavna uporaba in široka dostopnost v standardnih statističnih programih, vendar pa veliko empiričnih študij (Arbuckle 1996; Azen, Van Guilder in Hill 1989; Brown 1994; Enders 2001; Enders in Bandalos 2001; Haitovsky 1968; Kim in Curry 1997; Kromrey in Hines 1994; Wothke 2000 v Enders 2010) potrjuje, da sta ti dve metodi med dvema najslabšima.

Vstavljanje aritmetične sredine vrednosti gosti okrog aritmetične sredine, kar vodi v podcenjevanje vrednosti variance, korelacijske koeficiente pa »vleče« proti 0. To potrjujejo tudi empirične študije (Brown 1994; Enders in Bandalos 2001; Gleason in Staelin 1975; Kim in Curry 1975; Kromrey in Hines 1994; Olinsky, Chen in Harlow 2003; Raymond in Roberts 1987; Timm 1970; Wothke 2000 v Enders 2010). Simulacijske študije nakazujejo, da je vstavljanje aritmetične sredine ena najslabših, morda celo najslabša metoda za obravnavo manjkajočih podatkov (Little in Rubin 2002; Enders 2010).

Vstavljanje razpoložljivih vrednosti ali »hot deck« vstavljanje je metoda, kjer na mesto manjkajočih vstavljamo razpoložljive vrednosti. Kljub praktični pomembnosti in uporabnosti

»hot deck« metode je statistična literatura, ki bi obravnavala njeno teoretsko podlago in primerjavo z ostalimi metodami za obravnavo manjkajočih podatkov, močno omejena, kar seveda odpira veliko priložnosti za nadaljnje metodološko raziskovanje (Andridge in Little 2010).

Trenutno sta najbolj zanimivi in priporočeni metodi za obravnavo manjkajočih podatkov metoda največjega verjetja in večkratno vstavljanje (Enders 2010, van Buuren 2012; Schafer in Graham 2002). EM algoritem izdela nepristranske oz. skoraj nepristranske ocene za aritmetično sredino, varianco in kovarianco pri MCAR in MAR, pri NMAR pa ne, kar potrjujejo številne objavljene simulacijske raziskave (Arbuckle 1996; Enders 2001; Enders in Bandalos 2001; Gold in Bentler 2000; Muthen in drugi 1987; Olinsky, Chen in Harlow 2003; Wothke 2000 v Enders 2010). Podobno kot EM algoritem tudi večkratno vstavljanje izdela nepristranske ocene pri MCAR in MAR, pri NMAR pa ne, kar se sklada s teorijo na področju manjkajočih podatkov (Rubin 1976; Schafer 1997 v Enders 2010) in z rezultati simulacijskih raziskav (Allison 2000; Collins in drugi 2001; Graham in Schafer 1999; Newman 2003 v Enders 2010).

Večina dosedanjih raziskav, ki obravnavajo pojav manjkajočih podatkov v povezavi z metodo glavnih komponent, je zelo parcialnih. V objavljenih člankih gre večinoma za predlog novih algoritmov, medtem ko primerjav (simulacij) med različnimi metodami za obravnavo manjkajočih podatkov v povezavi z metodo glavnih komponent ni.

Največkrat objavljeno delo s področja manjkajočih podatkov Statistical Analysis with Missing Data (Little in Rubin 2007) metode glavnih komponent ne obravnava eksplicitno, temveč se osredotoča predvsem na ocenjevanje variančno-kovariančnih matrik na splošno. Članek avtorjev Tipping in Bishop pa je eden izmed redkih, ki obravnava prav metodo glavnih komponent v primeru manjkajočih podatkov (poudarek je na EM algoritmu). Tipping in Bishop predlagata iterativni algoritem, kjer EM algoritem za ocenjevanje verjetnostnega PCA modela združita z Little-Rubinovo metodologijo za ocenjevanje parametrov večkratne normalne porazdelitve (Joliffe 2002, 365).

Raziskave, ki obravnavajo metodo glavnih komponent in manjkajoče podatke, se večinoma osredotočajo predvsem na robustno ocenjevanje glavnih komponent. Namen robustnega ocenjevanja in obravnave manjkajočih podatkov pa je v splošnem zelo podoben; v obeh primerih identificiramo določene vrednosti, ki jih ne moremo uporabiti, ne da bi jih

prilagodili, saj so ali sumljivo ekstremne (robustno ocenjevanje) ali pa jih sploh ni (manjkajoče vrednosti). Če bi take vrednosti ignorirali, bi lahko zavrgli pomembno informacijo, zato jih poskušamo oceniti (Joliffe 2002, 366).

Članek avtorjev Ilin in Raiko (2010) s področja podatkovnega rudarjenja je po zasnovi še najbližje simulacijski raziskavi, ki je izvedena v tej nalogi, vendar pa se članek z vsebinskega vidika od naše simulacije razlikuje po tem, da se ne osredotoča na primerjavo različnih metod za obravnavo manjkajočih podatkov, temveč na primerjavo uspešnosti različnih verzij metode glavnih komponent v primeru manjkajočih podatkov.

4.2 RAZISKOVALNA VPRAŠANJA

Ker je primanjkljaj raziskav, ki bi natančno analizirale vpliv metod za obravnavo manjkajočih podatkov na metodo glavnih komponent, velik, smo se pri oblikovanju hipotez oprli predvsem na literaturo, ki obravnava različne metode za obravnavo manjkajočih podatkov ter mehanizme za nastanek manjkajočih vrednosti in ni neposredno povezana z metodo glavnih komponent.

V nalogi bomo poskušali odgovoriti na naslednji dve raziskovalni vprašanji:

(1) Ali so za izvedbo metode glavnih komponent v primeru manjkajočih podatkov moderni pristopi za obravnavo manjkajočih podatkov (metoda največjega verjetja in večkratno vstavljanje) primernejši od klasičnih?

Schafer in Graham (2002) metode za obravnavo manjkajočih podatkov delita v dve skupini, in sicer starejše pristope (klasične), med katere uvrščata analizo na osnovi popolnih enot, analizo na osnovi razpoložljivih podatkov, vstavljanje aritmetične sredine spremenljivk in vstavljanje razpoložljivih vrednosti, ter moderne pristope, med katere uvrščata metodo največjega verjetja in večkratno vstavljanje. S simulacijsko študijo sta prišla do ugotovitve, da so moderni pristopi v primerjavi s klasičnimi večinoma ustreznejši (glej Schafer in Graham 2002, 173). Zelo podobna simulacija je izvedena v magistrski nalogi, le da poudarek

ni na posledicah vstavljanja za posamezno spremenljivko, temveč na multivariatnem vidiku posledic vstavljanja.

(2) V kolikšni meri je primernost različnih metod odvisna od deleža manjkajočih podatkov in katere metode so primernejše glede na različne deleže manjkajočih podatkov ter glede na različne mehanizme za nastanek manjkajočih podatkov (MCAR, MAR, NMAR)?

Glede na prebrano literaturo analiza na osnovi popolnih enot vodi do nepristranskih ocen parametrov le pri izpolnjeni predpostavki MCAR, ena njenih ključnih pomanjkljivosti pa je potencialna izguba velikega števila enot. Nepristranske rezultate lahko torej pričakujemo le pri predpostavki MCAR in nizkem deležu manjkajočih podatkov. Podobno lahko pričakujemo tudi pri analizi na osnovi razpoložljivih podatkov ter vstavljanju aritmetičnih sredin spremenljivk, saj vse tri metode predpostavljajo MCAR (Howell 2007; Vehovar 2007).

Vstavljanje naključnih vrednosti naj bi bilo razmeroma uspešno le pri nizkem deležu manjkajočih podatkov, saj metoda ne upošteva multivariatnega vidika vstavljanja, metoda KNN pa naj bi bila med klasičnimi pristopi k obravnavi manjkajočih podatkov vsekakor najuspešnejša.

EM algoritem in večkratno vstavljanje (MICE-PMM) naj bi se izkazala kot ustrezna tako pri predpostavki MCAR kot tudi MAR (glej Schafer in Graham 2002), pri predpostavki NMAR pa naj bi se tudi ti dve metodi izkazali precej slabše (čeprav še zmeraj bolje od klasičnih pristopov).

5 NAČRT RAZISKAVE

Za preverjanje vpliva različnih pristopov za obravnavo manjkajočih podatkov na obnašanje metode glavnih komponent je bila oblikovana simulacijska študija na treh empiričnih podatkovnih bazah. Simulacija je bila implementirana v programskem jeziku R (oz. GNU S).

5.1 RAZDELAVA METODOLOŠKEGA OKVIRA PROUČEVANJA

Na popolnih podatkih (brez manjkajočih vrednosti) je na podlagi korelacijske matrike najprej izvedena metoda glavnih komponent; tako so na voljo rezultati, glede na katere se v nadaljevanju primerja kvaliteta različnih metod za obravnavo manjkajočih podatkov.

Podatkovne matrike z manjkajočimi podatki so generirane ločeno za tri podatkovne baze (za opis podatkovnih baz glej poglavje 5.4), glede na tri mehanizme za nastanek manjkajočih vrednosti MCAR, MAR in NMAR in glede na različne odstotke enot z manjkajočih vrednosti (glej odstavek spodaj). Za vsako od teh kombinacij je generiranih 1000 podatkovnih matrik.

Odstotek enot z manjkajočimi podatki zavzema vrednosti med 0 in 60 % s korakom po 5 %. Poleg odstotka enot z manjkajočimi podatki je v odvisnosti od odstotka enot z manjkajočimi podatki določen tudi odstotek manjkajočih podatkov v celotni matriki, ki zavzema vrednosti od 0 do 12 % s korakom po 1 %; pri 5 % enot z manjkajočimi podatki manjka 1 % vseh podatkov v matriki, pri 10 % enot z manjkajočimi podatki manjkata 2 % vseh podatkov v matriki … in pri 60 % enot z manjkajočimi podatki manjka 12 % vseh podatkov v matriki (glej tabelo 5.1).

Tabela 5.1: Odstotek enot z manjkajočimi podatki in odstotek manjkajočih podatkov v podatkovni matriki

Odstotek enot z manjkajočimi podatki	0	5	10	15	20	25	30	35	40	45	50	55	60
Odstotek manjkajočih podatkov v matriki	0	1	2	3	4	5	6	7	8	9	10	11	12

V nalogi je preizkušenih naslednjih sedem načinov obravnave manjkajočih podatkov, in sicer:

- analiza na osnovi popolnih enot (*angl.* listwise deletion),
- analiza na osnovi razpoložljivih podatkov (*angl.* pairwise deletion),
- vstavljanje aritmetične sredine,
- vstavljanje naključnih vrednosti (RVI),
- zaporedna metoda k-najbližjih sosedov (KNN),
- večkratno vstavljanje (MICE-PMM),
- EM algoritem.

Pri zaporedni metodi k-najbližjih sosedov smo preverili, kaj se z rezultati dogaja na različnih podatkovnih bazah pri različnih mehanizmih za nastanek manjkajočih podatkov, če vzamemo k najbližjih enot za različne k (v praksi v primeru manjkajočih vrednosti optimalnega k ne poznamo, saj nam podatkovna baza brez manjkajočih vrednosti za primerjavo ni na voljo). V empiričnem delu so zato prikazani rezultati za različne *k* (1, 3, 5, 10, 20).

Za izvedbo večkratnega vstavljanja oz. MICE algoritma je treba določiti število iteracij oz. ciklov za izdelavo ene podatkovne matrike: v empiričnem delu naloge so prikazani rezultati 10 iteracij, pri vsakem vstavljanju pa je bila izdelana ena popolna podatkovna matrika. Za pridobitev nepristranskih ocen za vstavljene vrednosti glede na opravljene simulacije (van Buuren 2012, 113) je pet iteracij MICE algoritma navadno dovolj.

Pri EM algoritmu za variančno-kovariančne matrike je treba določiti kriterij konvergence in maksimalno število iteracij. Kriterij konvergence je določen pri 0,000001, maksimalno število iteracij pa pri 10000. Algoritem se ustavi, ko je maksimalna relativna razlika vseh ocenjenih aritmetičnih sredin, varianc in kovarianc manjša ali enaka vrednosti kriterija konvergence oz. po maksimalnem številu iteracij, če ocene parametrov še niso konvergirale.

Pri vsaki metodi se na podatkovnih matrikah z manjkajočimi podatki[3] ponovi sledeče (algoritem 1000-krat ponovi naslednje):

1. izvedba metode za obravnavo manjkajočih podatkov,[4]
2. izvedba metode glavnih komponent na podlagi ocenjene korelacijske matrike – izračun uteži in varianc,
3. izračun mer podobnosti in različnosti CC in RMS[5].

Pri vsaki ponovitvi se izračunajo naslednje mere:

- RMS mera za uteži na izbranih (primernih) glavnih komponentah,[6]
- CC mera za uteži na izbranih (primernih) glavnih komponentah,
- RMS mera za lastne vrednosti izbranih (primernih) glavnih komponent.

Po 1000 ponovitvah se izračuna povprečne vrednosti zgornjih mer odstopanja.

Reševanje problema pozitivne definitnosti variančno-kovariančne oz. korelacijske matrike

Rezultati metode glavnih komponent so smiselni le, če so variance glavnih komponent pozitivna števila. To pa so, če je variančno-kovariančna matrika oz. korelacijska matrika pozitivno definitna (Ferligoj 2011; Jolliffe 2002). Ker se pri analizi na osnovi razpložljivih enot (v primeru manjkajočih vrednosti) lahko zgodi, da dobimo variančno-kovariančno oz. korelacijsko matriko, ki ni pozitivno definitna, je v programski kodi dodan pogoj, da v takem

[3] Podatkovne matrike z manjkajočimi podatki so shranjene v posebni datoteki, tako da vse metode za obravnavo manjkajočih podatkov uporabimo na istih nepopolnih matrikah.

[4] Pri analizi na osnovi popolnih enot je tukaj mišljeno brisanje nepopolnih enot, pri analizi na osnovi razpoložljivih podatkov ter EM algoritmu izračun korelacijske matrike, pri ostalih metodah (metodah vstavljanja) pa vstavljanje manjkajočih podatkov.

[5] Meri sta predstavljeni v poglavju 5.3.

[6] Glej opis podatkovnih baz, poglavje 5.4.

primeru program najde najbližjo možno pozitivno definitno variančno-kovariančno oz. korelacijsko matriko ter za nadaljnje izračune uporablja le-to (glej Schaefer in drugi 2011).

5.2 NAČINI GENERIRANJA PODATKOVNIH MATRIK

Vsi trije mehanizmi za nastanek manjkajočih podatkov (MCAR, MAR in NMAR) so bili implementirani v dveh korakih: v prvem koraku smo določenemu odstotku enot v podatkovni bazi (med 0 in 60 %) izbrisali po eno vrednost. V drugem koraku smo pri izbranih enotah (v prvem koraku) brisali vrednosti toliko časa, da smo dosegli želen odstotek manjkajočih podatkov v celotni podatkovni bazi (med 0 in 12 %). Vrednosti smo brisali pri vseh spremenljivkah v podatkovni bazi. Odstotek izbrisanih (manjkajočih) podatkov v podatkovni bazi v odvisnosti od odstotka enot z manjkajočimi podatki je prikazan v tabeli 5.1.

Generiranje podatkov tipa MCAR

Podatki, ki manjkajo po mehanizmu MCAR, manjkajo neodvisno od samih manjkajočih vrednosti in drugih atributov, torej manjkajo povsem naključno.

Opis postopka implementacije mehanizma MCAR

V prvem koraku naključno izberemo eno spremenljivko iz nabora vseh spremenljivk ter eno enoto iz nabora vseh enot brez manjkajočih vrednosti. Pripadajoči element podatkovne matrike izbrišemo (določimo kot manjkajočo vrednost). Postopek ponovimo tolikokrat, da dosežemo želen odstotkek enot z manjkajočimi podatki.

V drugem koraku naključno izberemo eno spremenljivko iz nabora vseh spremenljivk ter eno enoto iz nabora enot, ki že imajo eno ali več manjkajočih vrednosti, a ne pri tej (izbrani) spremenljivki. Pripadajoči element podatkovne matrike izbrišemo (določimo kot manjkajočo vrednost). Postopek ponovimo tolikokrat, da dosežemo želen odstotek manjkajočih vrednosti v celotni podatkovni matriki.

Koda algoritma za mehanizem MCAR v programskem jeziku R je prikazana na sliki 5.1.

Slika 5.1: Koda algoritma za mehanizem MCAR v programskem jeziku R

```
n<-dim(matrika)[1]       # število enot
m<-dim(matrika)[2]      # število spremenljivk
nme<-p/100*n             # število enot z manjkajočimi vrednostmi
nmv<-pmv/100*n*m    # število manjkajočih vrednosti v celotni bazi

tnme<-0
tnmv<-0

while(tnme<nme){
  iSpr<-sample(1:m,size=1) # naključno izberemo spremenljivko
  tEnote<-which(!apply(is.na(xMiss),1,any)) #dovoljene enote, ki so brez manjkajočih vrednosti
  iEnota<-sample(tEnote,size=1) # naključno izberemo enoto iz nabora dovoljenih enot
  xMiss[iEnota,iSpr]<-NA  #izbrano vrednost izbrišemo
  tnme<-tnme+1
  tnmv<-tnmv+1
  }
while(tnmv<nmv){
  iSpr<-sample(1:m,size=1) # naključno izberemo spremenljivko
  tEnote<- which(apply(is.na(xMiss),1,any)&(!is.na(xMiss[,iSpr]))) #dovoljene enote, ki že imajo
manjkajočo vrednost, a ne pri tej spremenljivki
  iEnota<-sample(tEnote,size=1)
  xMiss[iEnota,iSpr]<-NA
  tnmv<-tnmv+1
  }
```

Primer: v podatkovni bazi s 300 enotami in z 10 spremenljivkami brez manjkajočih vrednosti želimo pri 5 % enot izbrisati 1 % vseh vrednosti v celotni podatkovni bazi. Po zgoraj opisanem postopku bomo najprej naključno izbrali 300 * 5 % = 15 enot, pri katerih bomo naključno izbrisali 300 * 10 * 1 % = 30 vrednosti pod pogojem, da je za vsako izbrano enoto vsaj ena vrednost manjkajoča.

Generiranje podatkov tipa MAR

Podatki, ki manjkajo po mehanizmu MAR, manjkajo v odvisnosti od ene ali več spremenljivk v podatkovni matriki (torej manjkajo pogojno naključno). V podatkovni bazi »Mednarodna anketa« podatki manjkajo v odvisnosti od spremenljivke »famc2« (»Življenje je vredno živeti, kadar se ljudje popolnoma posvetijo družini.«), v podatkovni bazi »Prepoznavnost vin« podatki manjkajo v odvisnosti od spremenljivke »prolin« (vsebnost aminokisline prolin v vzorcu vina), v podatkovni bazi »Nova vozila« pa podatki manjkajo v odvisnosti od spremenljivke »engine« (prostornina motorja); enote z višjimi vrednostmi na teh spremenljivkah imajo višjo verjetnost za manjkajoče vrednosti.

Opis postopka implementacije mehanizma MAR

V prvem koraku naključno izberemo eno spremenljivko iz nabora vseh spremenljivk brez vnaprej določene spremenljivke **x** (glej zgornji odstavek) ter eno enoto iz nabora vseh enot brez manjkajočih vrednosti, pri čemer je verjetnost izbora enote odvisna od vrednosti spremenljivke **x** pri tej enoti; večja kot je vrednost te spremenljivke, večja je verjetnost izbora enote. Verjetnost izbora enote je določena na podlagi spodnje formule (5.1),

$$p_i = \frac{(z_i)^{10}}{\sum_{i=1}^{n} (z_i)^{10}}, \qquad (5.1)$$

kjer je

p_i verjetnost izbora i-te enote in

z_i standardizirana vrednost (z-vrednost) i-te enote spremenljivke **x**, ki ji prištejemo vrednost 10.[7]

Pripadajoči element podatkovne matrike izbrišemo (določimo kot manjkajočo vrednost). Postopek ponovimo tolikokrat, da dosežemo želen odstotek enot z manjkajočimi podatki.

V drugem koraku naključno izberemo eno spremenljivko iz nabora vseh spremenljivk brez vnaprej določene spremenljivke **x** ter eno enoto iz nabora enot, ki že imajo eno ali več manjkajočih vrednosti, a ne pri tej (izbrani) spremenljivki, pri čemer je enako kot v prvem koraku verjetnost izbora enote odvisna od vrednosti spremenljivke **x** v podatkovni bazi (glej formulo 5.1). Pripadajoči element podatkovne matrike izbrišemo (določimo kot manjkajočo vrednost). Postopek ponovimo tolikokrat, da dosežemo želen odstotek manjkajočih vrednosti v celotni podatkovni matriki.

Koda algoritma za mehanizem MAR se nahaja v prilogi C.

[7] Standardiziranim z-vrednostim prištejemo vrednost 10; tako so vse vrednosti v podatkovni matriki pozitivna števila.

Generiranje podatkov tipa NMAR

Podatki, ki manjkajo po mehanizmu NMAR, manjkajo v odvisnosti od dejanskih manjkajočih vrednosti. Večje kot so vrednosti posameznih spremenljivk, večja je verjetnost, da bodo ravno te vrednosti manjkajoče.

Opis postopka implementacije mehanizma NMAR

V prvem koraku naključno izberemo eno spremenljivko iz nabora vseh spremenljivk ter eno enoto iz nabora vseh enot brez manjkajočih vrednosti, pri čemer je verjetnost izbora enote odvisna od vrednosti izbrane spremenljivke; večja, kot je vrednost te spremenljivke, večja je verjetnost izbora enote, kjer je verjetnost izbora enote določena na podlagi formule 5.1 zgoraj. Pripadajoči element podatkovne matrike izbrišemo (določimo kot manjkajočo vrednost). Postopek ponovimo tolikokrat, da dosežemo vnaprej določen odstotek enot z manjkajočimi podatki.

V drugem koraku naključno izberemo eno spremenljivko iz nabora vseh spremenljivk ter eno enoto iz nabora enot, ki že imajo eno ali več manjkajočih vrednosti, a ne pri tej (izbrani) spremenljivki, pri čemer je verjetnost izbora enote odvisna od vrednosti izbrane spremenljivke; večja kot je vrednost te spremenljivke, večja je verjetnost izbora enote, kjer je verjetnost izbora enote določena na podlagi formule 5.1. Pripadajoči element podatkovne matrike izbrišemo (določimo kot manjkajočo vrednost). Postopek ponovimo tolikokrat, da dosežemo vnaprej določen odstotek manjkajočih vrednosti v celotni podatkovni matriki.

Koda algoritma za mehanizem NMAR se nahaja v prilogi C.

5.3 NAČINI EVALVACIJE REZULTATOV

Če želimo različne načine obravnavanja manjkajočih podatkov primerjati med seboj, moramo najprej definirati ustrezne mere podobnosti oz. različnosti za lastne vrednosti in uteži. Osnovna ideja je sledeča: po uporabi ene izmed metod za obravnavo manjkajočih podatkov izvedemo metodo glavnih komponent. Dobljene variance (lastne vrednosti) glavnih komponent in uteži primerjamo z variancami (lastnimi vrednostmi) in utežmi metode glavnih komponent na izhodiščnih popolnih podatkih. Večje kot je odstopanje, v manjši meri je

metoda za obravnavo manjkajočih podatkov prispevala k ohranitvi prvotne komponentne strukture.

Mera podobnosti oz. različnosti je po navadi eno samo število, kar ima tako svoje prednosti kot tudi slabosti. Prednost je gotovo dejstvo, da je primerjava med različnimi metodami obravnave manjkajočih podatkov zelo enostavna, slabost pa ta, da iz večdimenzionalnega podatka naredimo enodimenzionalen podatek. Taka mera vedno predstavlja nevarnost v smislu izgube informacije, zato jo moramo interpretirati karseda previdno. Najbolj učinkovito to težavo zaobidemo z določitvijo več mer podobnosti oz. različnosti, ki jih nato spremljamo hkrati.

Za primerjavo lastnih vrednosti in uteži bomo v nalogi uporabili dve uveljavljeni meri (Levine 1977; Lorenzo-Seva in ten Berge 2006), in sicer koren povprečja kvadratov (za primerjavo lastnih vrednosti in uteži) in koeficient skladnosti (za primerjavo uteži).

Pri metodi glavnih komponent so uteži lastni vektorji, ki predstavljajo korelacije med spremenljivkami in glavnimi komponentami. Lastni vektorji s predpisano dolžino so določeni do predznaka natančno, zato moramo, ko jih primerjamo med seboj (torej pred izračunom RMS in CC), poskrbeti, da kažejo v isto smer, tako da jih po potrebi množimo z -1. Sprememba smeri namreč ni odraz kakovosti metode za obravnavo manjkajočih podatkov, ampak lahko nenapovedano nastopi kadarkoli.

5.3.1 Koren povprečja kvadratov

Koren povprečja kvadratov (*angl.* Root Mean Square – RMS) je kvadratni koren povprečne kvadratne napake (*angl.* Mean Squared Error). Z RMS primerjamo vrednosti dveh nizov podatkov, kjer prvi niz navadno predstavljajo napovedane vrednosti hipotetičnega modela, drugi niz pa predstavljajo opazovane vrednosti.

Koren povprečja kvadratov za primerjavo uteži predstavlja koren povprečja kvadratov odstopanj uteži, dobljenih na popolnih podatkih, in uteži, dobljenih na nepopolnih podatkih. Mera je občutljiva tako na spremembe v razporeditvi uteži kot tudi na njihovo velikost, računamo pa jo lahko za poljubno število glavnih komponent (Levine 1977; Raiko in drugi

2007). Uteži (lastne vektorje) pred izračunom RMS po potrebi množimo z -1; tako poskrbimo, da kažejo v isto smer. Formula za izračun je sledeča:

$$RMS = \sqrt{\frac{\sum_{i=1}^{p}\sum_{j=1}^{m}\left(a_{ij} - a_{ij}^{'}\right)^{2}}{m \cdot p}}, \tag{5.2}$$

kjer je

a_{ij} element matrike, dobljene na popolnih podatkih;

a_{ij}' element matrike, dobljene na nepopolnih podatkih;

p število spremenljivk;

m število komponent.

RMS zavzema vrednosti med 0 in 2, pri čemer doseže vrednost 0 v primeru popolnega ujemanja uteži na popolnih in nepopolnih podatkih, vrednost 2 pa takrat, ko se uteži na popolnih in nepopolnih podatkih najbolj razlikujejo.

Koren povprečja kvadratov za primerjavo lastnih vrednosti predstavlja koren povprečja kvadratov odstopanj varianc (lastnih vrednosti) glavnih komponent, dobljenih na popolnih podatkih, in varianc (lastnih vrednosti) glavnih komponent, dobljenih na nepopolnih podatkih. Izračunamo ga po spodnji formuli (5.3):

$$RMS = \sqrt{\frac{\sum_{i=1}^{m}(\lambda_i - \lambda_i^{'})^{2}}{m}}, \tag{5.3}$$

kjer je

λ_i varianca (lastna vrednost) i-te glavne komponente, dobljena na popolnih podatkih;

λ_i' varianca (lastna vrednost) i-te glavne komponente, dobljena na nepopolnih podatkih;

m število komponent.

5.3.2 Koeficient skladnosti

Koeficient skladnosti (*angl*. Coefficient of Congruence – CC) je mera podobnosti, ki se jo uporablja za primerjavo uteži tako pri faktorski analizi kot tudi pri metodi glavnih komponent. Mero je predlagal Cyril Burt leta 1948, bolj poznana pa je postala nekaj let kasneje kot

Tuckerjev koeficient skladnosti (Lorenzo-Seva in ten Berge 2006; Herve 2007).

Podobno kot RMS je tudi koeficient skladnosti mera, ki je občutljiva tako na spremembe v razporeditvi uteži kot tudi na njihovo velikost. Računamo jo lahko za poljubno število komponent (Herve 2010; Teel in Verran 1991; Bedeian in drugi 1988). Uteži (lastne vektorje) pred izračunom CC po potrebi množimo z -1; tako poskrbimo, da kažejo v isto smer. Formula za izračun koeficienta skladnosti je sledeča (glej formulo 5.4):

$$CC_M = \frac{\sum_{j=1}^{m}\sum_{i=1}^{p} a_{ij} a_{ij}^{'}}{\sum_{j=1}^{m}\sqrt{\left(\sum_{i=1}^{p} a_{ij}^2\right)\left(\sum_{i=1}^{p} a_{ij}^{'2}\right)}}, \qquad (5.4)$$

kjer je

a_{ij} element matrike, dobljene na popolnih podatkih;

a_{ij}' element matrike, dobljene na nepopolnih podatkih;

p število spremenljivk;

m število komponent.

Maksimalna vrednost *CC* je 1; v tem primeru se uteži na popolnih in na nepopolnih podatkih popolnoma ujemajo. Kadar so vse uteži po absolutni vrednosti enako velike, sicer pa nasprotnega predznaka, je vrednost *CC* enaka -1. Kadar so uteži med seboj najbolj različne (tj. takrat, ko so komponente pravokotne ena na drugo), je vrednost *CC* enaka 0. Torej so vrednosti blizu 1 oz. -1 dobre, vrednosti okoli 0 pa govorijo o občutno spremenjeni strukturi. Konsenz glede mejne vrednosti, ki bi zagotavljala skladnost med komponentama, ne obstaja, zato navajamo Tuckerjeve smernice (Lorenzo-Seva in ten Berge 2006: 58): 0,98 do 1,00 = *odlična*, 0,92 do 0,98 = *dobra*, 0,82 do 0,92 = *mejna*, 0,68 do 0,82 = *slaba*, pod 0,68 = *nezadostna*. Te smernice je treba upoštevati z določeno stopnjo previdnosti, saj so bile vzpostavljene za primerjavo faktorjev oz. komponent dveh različnih raziskav.

Čeprav se CC uporablja zelo pogosto, je treba upoštevati, da dobimo visoke vrednosti CC vsakič, ko imata dve komponenti uteži pri veliko spremenljivkah z enakim predznakom (Levine 1977).

5.4 OPIS PODATKOVNIH BAZ

Za učinkovito napovedovanje uspešnosti različnih načinov vstavljanja manjkajočih podatkov so bile izbrane tri empirične baze podatkov z ordinalnimi in razmernostnimi spremenljivkami, primerne za izvedbo metode glavnih komponent (strogo gledano prva baza vsebuje ordinalne spremenljivke, vendar jih obravnavamo kot intervalne). Zaradi težnje po posploševanju rezultatov so bile podatkovne baze izbrane s treh vsebinsko popolnoma različnih področij. Za simulacijo so bili uporabljeni podatki iz raziskave Cross-Cultural Research za Slovenijo iz leta 2002 (Frieze 2011), podatki z raziskave Prepoznavnost vin iz leta 1991 (UCI Machine Learning Repository 2011) ter podatki o novih vozilih na ameriškem trgu leta 2004 (Shalizi 2011).

5.4.1 1. podatkovna baza: Mednarodna anketa o stališčih in izkušnjah študentov

Anketiranje se je izvajalo na podlagi mednarodne ankete o stališčih in izkušnjah študentov. Vprašalnik, ki sta ga pripravili Irene Frieze in Anuška Ferligoj, se je med letoma 1991 in 2004 uporabljalo na več univerzah v Evropi (Albanija, Bolgarija, Češka, Hrvaška, Litva, Madžarska, Nemčija, Norveška, Poljska, Rusija, Slovaška), Indiji, Pakistanu, na Japonskem in v ZDA (Frieze 2011). Obravnavana populacija so študentje Fakultete za družbene vede v Ljubljani v letu 2002, in sicer so praviloma to študentje drugega letnika, priložnostni vzorec pa so študentje, ki so bili v času izvajanja ankete prisotni na predavanjih, torej aktivni študenti. Takih študentov je bilo 328 (n = 328). Podatkovna baza ne vsebuje manjkajočih vrednosti.

Analiziranih je bilo 12 spremenljivk, ki merijo središčnost dela in družine (Family and Work Centrality). Vseh 12 spremenljivk je ordinalnih, kar pomeni, da lahko vrednosti razporedimo od najmanjše do največje (v našem primeru od 1 do 5). Te spremenljivke se, kot se to običajno počne v družboslovju, obravnava kot intervalne. Imena spremenljivk z ustreznimi vprašanji so navedena v spodnji tabeli (5.2).

Tabela 5.2: Opis spremenljivk za podatkovno bazo »Mednarodna anketa«

famc1	Ljudje bi se morali posvetiti družini.
famc2	Življenje je vredno živeti, kadar se ljudje popolnoma posvetijo družini.
famc3	Družina bi morala biti v življenju posameznika osrednjega pomena.
famc4	Posameznikovi življenjski cilji bi morali biti usmerjeni predvsem v družino.
famc5	Najpomembnejše stvari v življenju se tičejo družine.
famc6	Družina bi morala biti pomemben del posameznikovega življenja.
wrkc1	Najpomembnejše stvari v življenju se tičejo dela.
wrkc2	Delo bi moralo biti pomemben del posameznikovega življenja.
wrkc3	Ljudje bi se morali posvetiti delu.
wrkc4	Delo bi moralo biti v življenju posameznika osrednjega pomena.
wrkc5	Posameznikovi življenjski cilji bi morali biti usmerjeni predvsem v delo.
wrkc6	Življenje je vredno živeti, kadar se ljudje popolnoma posvetijo delu.

Lestvica možnih odgovorov:

1 – močno nasprotujem
2 – nasprotujem
3 – niti ne nasprotujem niti se ne strinjam
4 – strinjam se
5 – močno se strinjam

Opisne statistike spremenljivk in korelacijska matrika se nahajajo v prilogi A.

V spodnji tabeli (5.3) so prikazane lastne vrednosti komponent in odstotki celotne variance merjenih spremenljivk, ki jih opisuje vsaka od komponent.

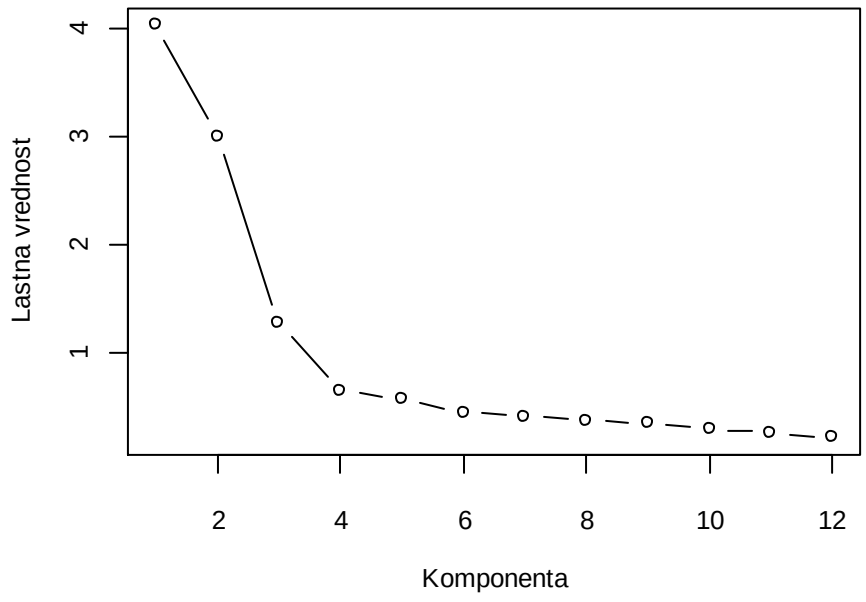
S pomočjo pravil, zapisanih v 2. poglavju (opis metode glavnih komponent), je treba določili število primernih komponent. Iz tabele 5.3 je razvidno, da prvi dve komponenti skupaj pojasnjujeta več kot 50 % celotne variance, lastne vrednosti prvih treh komponent pa so večje od 1.

Tabela 5.3: Lastne vrednosti na podatkovni bazi »Mednarodna anketa«

Komponente	Začetne lastne vrednosti		
	Lastne vrednosti	% Variance	Kumulativa (v %)
1	4,035	33,626	33,626
2	3,009	25,072	58,698
3	1,287	10,722	69,420
4	0,653	5,444	74,864
5	0,589	4,907	79,771
6	0,459	3,827	83,598
7	0,421	3,512	87,110
8	0,383	3,193	90,303
9	0,362	3,018	93,321
10	0,303	2,524	95,845
11	0,273	2,275	98,120
12	0,226	1,880	100,000

Oglejmo si še plaziščni diagram (glej sliko 5.2). Tam, kjer se diagram lomi, je predlog za število komponent.

Slika 5.2: Plaziščni diagram – podatkovna baza »Mednarodna anketa«



Na podlagi plaziščnega diagrama bi izbrali prve tri komponente, vendar se v primeru središčnosti dela in družine izkaže, da sta z vsebinskega vidika pomembni predvem prvi dve komponenti. Prva pojasnjuje 33,63 %, druga pa 25,07 % variabilnosti izmerjenih spremenljivk, skupaj torej 58,70 %. Rezultati zadoščajo zgoraj navedenim kriterijem, saj sta

tudi lastni vrednosti obeh glavnih komponent večji od 1 (glej tabelo 5.3). V spodnji tabeli (5.4) so prikazane uteži (korelacijski koeficienti med merjenimi spremenljivkami in glavnimi komponentami) za prvi dve glavni komponenti.

Tabela 5.4: Uteži – podatkovna baza »Mednarodna anketa«

	Komponente	
	1	2
famc1	**0,784**	0,185
famc2	**0,772**	0,294
famc3	**0,824**	0,274
famc4	**0,783**	0,265
famc5	**0,762**	0,298
famc6	**0,659**	0,132
wrkc1	-0,229	**0,700**
wrkc2	-0,193	**0,510**
wrkc3	-0,129	**0,621**
wrkc4	-0,336	**0,719**
wrkc5	-0,427	**0,713**
wrkc6	-0,342	**0,689**

Prva komponenta ima močne uteži na spremenljivkah, ki merijo središčnost družine, kar pomeni, večja kot bo vrednost na dani spremenljivki, večja bo vrednost na tej komponenti. Ta komponenta ima tudi v absolutnem smislu velike, vendar negativne vrednosti uteži na nekaterih spremenljivkah, ki merijo centralnost dela, kar pomeni, da bo v primeru večje vrednosti na dani spremenljivki manjša vrednost na tej komponenti. Največjo negativno utež ima s spremenljivko wrkc5, ki trdi, da bi posameznikovi cilji morali biti usmerjeni predvsem v delo. To je torej spremenljivka, ki posledično ne govori v prid družini, zato imamo na njej dokaj močno negativno korelacijo.

Druga komponenta ima močne uteži na spremenljivkah, ki merijo središčnost dela. Ta komponenta nima negativnih uteži na spremenljivkah, ki merijo centralnost družine.

Rezultati so v tem primeru pričakovani, saj merimo dve dimenziji, in sicer središčnost družine in središčnost dela. Višja vrednost na prvi komponenti pomeni torej višjo središčnost družine in nekoliko nižjo središčnost dela, višja vrednost na drugi komponenti pa višjo središčnost dela.

5.4.2 2. podatkovna baza: Prepoznavnost vin

Podatki so rezultat kemijske analize vin treh različnih pridelovalcev iz iste italijanske regije. Z analizo se je ugotavljalo prisotnost 13 različnih sestavin v 178 vzorcih (n = 178) treh različnih sort vin. Podatkovna baza ne vsebuje manjkajočih vrednosti.

Analiziranih je bilo 13 spremenljivk, ki predstavljajo vsebnost različnih sestavin vina. Vse spremenljivke so razmernostne (številske). Imena spremenljivk z opisi so navedena v tabeli 5.5.

Opisne statistike spremenljivk in korelacijska matrika se nahajajo v prilogi A.

Tabela 5.5: Opis spremenljivk za podatovno bazo »Prepoznavnost vin«

s1	Alkohol
s2	Malična kislina
s3	Usedline
s4	Alkalnost usedlin
s5	Magnezij
s6	Fenoli
s7	Flavonoidi
s8	Neflavonoidni fenoli
s9	Proantocianini
s10	Intenziteta barve
s11	Odtenek barve
s12	OD280/OD315 razredčenega vina
s13	Prolin

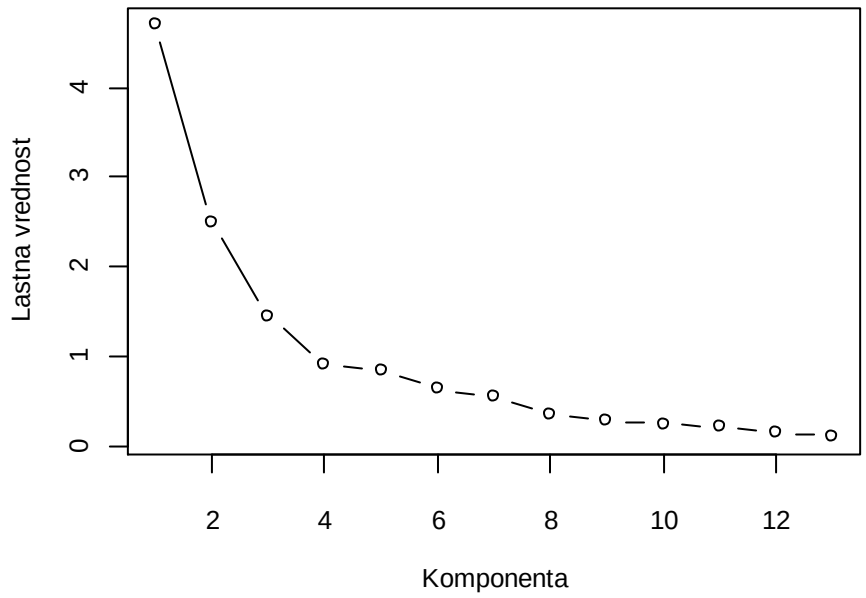
V spodnji tabeli (5.6) so prikazane lastne vrednosti komponent in odstotki celotne variance merjenih spremenljivk, ki jih opisuje vsaka od komponent. Iz tabele je razvidno, da prvi dve komponenti skupaj pojasnjujeta več kot 50 % celotne variance, prve tri komponente pa kar 66,53 %. Lastne vrednosti prvih treh komponent so večje od 1.

Tabela 5.6: Lastne vrednosti za podatkovno bazo »Prepoznavnost vin«

Komponente	Začetne lastne vrednosti		
	Lastne vrednosti	% Variance	Kumulativa (v %)
1	4,706	36,199	36,199
2	2,497	19,207	55,406
3	1,446	11,124	66,530
4	0,919	7,069	73,599
5	0,853	6,563	80,162
6	0,642	4,936	85,098
7	0,551	4,239	89,337
8	0,348	2,681	92,018
9	0,289	2,222	94,240
10	0,251	1,930	96,170
11	0,226	1,737	97,907
12	0,169	1,298	99,205
13	0,103	0,795	100

Oglejmo si še plaziščni diagram (glej sliko 5.3). Tam, kjer se diagram lomi, je predlog za število komponent.

Slika 5.3: Plaziščni diagram – podatkovna baza »Prepoznavnost vin«



Na primeru podatkovne baze »Prepoznavnost vin« so najpomembnejše prve tri komponente. Prva pojasnjuje 36,20 %, druga 19,21 % in tretja 11,12 % variabilnosti izmerjenih spremenljivk, skupaj torej 66,53 %. Rezultati zadoščajo zgoraj navedenim kriterijem, saj so

tudi lastne vrednosti vseh treh glavnih komponent večje od 1 (glej tabelo 5.6), tri glavne komponente pa lahko izberemo tudi na podlagi plaziščnega diagrama. V spodnji tabeli (5.7) so prikazane uteži (korelacijski koeficienti med merjenimi spremenljivkami in glavnimi komponentami) za glavne komponente z lastnimi vrednostmi, večjimi od 1.

Tabela 5.7: Uteži – podatkovna baza »Prepoznavnost vin«

	Komponente		
	1	2	3
s1	0,313	**0,764**	-0,249
s2	**-0,532**	0,355	0,107
s3	-0,004	0,499	**0,753**
s4	**-0,519**	-0,017	**0,736**
s5	0,308	0,473	0,157
s6	**0,856**	0,103	0,176
s7	**0,917**	-0,005	0,181
s8	**-0,648**	0,045	0,205
s9	**0,680**	0,062	0,180
s10	-0,192	**0,837**	-0,165
s11	**0,644**	-0,441	0,102
s12	**0,816**	-0,260	0,200
s13	**0,622**	**0,577**	-0,152

Interpretacija posameznih komponent je odvisna od tega, s katerimi merjenimi spremenljivkami komponente močno korelirajo (močne uteži na prvih treh komponentah so poudarjene). Uteži so koeficienti korelacije med merjenimi spremenljivkami in izbranimi komponentami.

Izkaže se, da so pomembne prve tri komponente. Prva komponenta ima močne uteži na večini spremenljivk, druga glavna komponenta ima močne uteži na spremenljivkah »vsebnost alkohola« in »intenziteta barve«, tretja pa na spremenljivkah »vsebnost usedline« in »alkalnost usedline« (glej tabelo 5.7). Za pravilno in korektno vsebinsko interpretacijo rezultatov bi bilo potrebno boljše poznavanje obravnavanega področja in spremenljivk, zato se ji bomo tukaj izognili, saj z vidika izvedbe simulacije in njenih končnih rezultatov ni bistvenega pomena.

5.4.3 3. podatkovna baza: Nova vozila na ameriškem trgu leta 2004

Podatkovna baza zajema tehnične specifikacije in ceno novih vozil na ameriškem trgu leta 2004. Podatki so bili pridobljeni s strani spletne revije za finančno svetovanje Kiplinger, z njihovim dovoljenjem pa jih je objavila revija Journal of Statistics Education na svojem podatkovnem arhivu. Podatki se nanašajo na ceno vozil, mere vozil ter njihovo energetsko učinkovitost, uporabljeni pa so bili le podatki za tista vozila, pri katerih ni bilo manjkajočih vrednosti (n = 387).

Analiziranih je bilo 10 spremenljivk, ki predstavljajo tehnične specifikacije in ceno novih vozil. Vse spremenljivke so razmernostne (številske). Imena spremenljivk z opisi so navedena v spodnji tabeli (5.8). Opisne statistike spremenljivk in korelacijska matrika se nahajajo v prilogi A.

Tabela 5.8: Opis spremenljivk za podatkovno bazo »Nova vozila«

retail	Priporočena maloprodajna cena
engine	Prostornina motorja (v litrih)
cylinders	Število valjev
horsepower	Moč motorja (KM)
citympg	Poraba goriva, mestna vožnja (v miljah na galon goriva)
highwaympg	Poraba goriva, izvenmestna vožnja (v miljah na galon goriva)
weight	Teža (v funtih)
wheelbase	Medosna razdalja (v inčih)
length	Dolžina (v inčih)
width	Širina (v inčih)

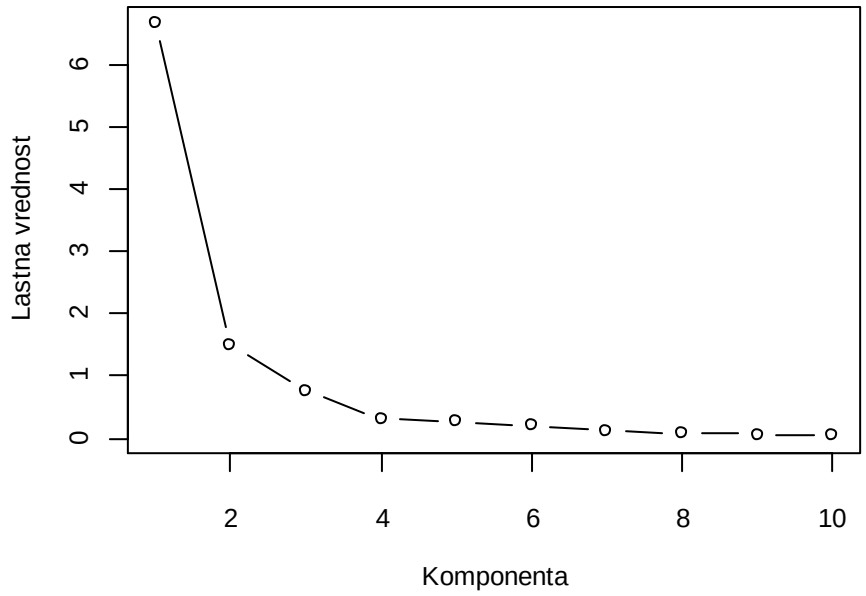
Tabela 5.9: Lastne vrednosti na podatkovni bazi »Nova vozila«

Komponente	Začetne lastne vrednosti		
	Lastne vrednosti	% Variance	Kumulativa (v %)
1	6,661	66,614	66,614
2	1,482	14,816	81,429
3	0,762	7,618	89,047
4	0,309	3,093	92,140
5	0,274	2,741	94,881
6	0,197	1,971	96,852
7	0,127	1,270	98,122
8	0,086	0,857	98,979
9	0,065	0,652	99,631
10	0,037	0,369	100,000

V tabeli (5.9) so prikazane lastne vrednosti komponent in odstotki celotne variance merjenih spremenljivk, ki jih opisuje vsaka od komponent. Iz tabele je razvidno, da prvi dve glavni komponenti skupaj pojasnjujeta kar 81,43 % celotne variance. To sta tudi glavni komponenti, katerih lastni vrednosti sta večji od 1.

Oglejmo si še plaziščni diagram (glej sliko 5.4). Tam, kjer se diagram lomi, je predlog za število komponent.

Slika 5.4: Plaziščni diagram – podatkovna baza »Nova vozila«



Na podlagi plaziščnega diagrama bi izbrali eno ali tri komponente, vendar se v primeru lastnosti avtov izkaže, da sta z vsebinskega vidika pomembni predvsem prvi dve komponenti (glej Shalizi 2011). Prva komponenta pojasnjuje 66,61 %, druga pa 14,82 % variabilnosti izmerjenih spremenljivk, skupaj torej 81,43 %. Rezultati zadoščajo zgoraj navedenim kriterijem, saj sta tudi lastni vrednosti obeh glavnih komponent večji od 1 (glej tabelo 5.9). V spodnji tabeli (5.10) so prikazane uteži (korelacijski koeficienti med merjenimi spremenljivkami in glavnimi komponentami) za prve tri glavne komponente (lastni vrednosti prvih dveh glavnih komponent sta večji od 1).

Tabela 5.10: Uteži – podatkovna baza »Nova vozila«

	Komponente		
	1	2	3
retail	**0,636**	**-0,602**	0,357
engine	**0,927**	-0,066	0,120
cylinders	**0,882**	-0,185	0,184
horsepower	**0,813**	**-0,452**	0,207
citympg	**-0,835**	0,169	0,457
highwaympg	**-0,825**	0,192	**0,510**
weight	**0,913**	0,141	-0,127
wheelbase	**0,747**	**0,567**	0,127
length	**0,720**	**0,570**	0,205
width	**0,819**	0,380	0,010

Prva komponenta ima močne uteži na vseh spremenljivkah, vendar so na spremenljivkah, ki merijo porabo, negativne (podatki za porabo so zapisani v prevoženih miljah na galon goriva). To pomeni, da obstaja negativna korelacija med porabo in vsemi ostalimi spremenljivkami. Prva komponenta nam pove, ali je vozilo veliko, drago in potratno ter z močnim motorjem, oziroma majhno, poceni in energetsko učinkovito ter z nekoliko šibkejšim motorjem (Shalizi 2011).

Druga komponenta ima močne negativne uteži na spremenljivkah »maloprodajna cena« in »moč motorja«, močne pozitivne uteži pa na spremenljivkah, ki merijo velikost vozila (medosna razdalja in dolžina, delno pa tudi širina vozila). Druga komponenta torej loči terenska vozila, enoprostorska vozila in manjša tovorna vozila (velika, ne pretirano draga in ne pretirano močna vozila) od dragih športnih avtomobilov (majhna, draga in močna vozila) (prav tam).

6 PREDSTAVITEV REZULTATOV

V pričujočem poglavju so predstavljeni rezultati simulacijske raziskave. Rezultati so za tri podatkovne baze (opis podatkovnih baz se nahaja v poglavju 5.4) predstavljeni ločeno. V prvem podpoglavju so predstavljeni rezultati simulacije za mehanizem MCAR, v drugem za mehanizem MAR in v tretjem za mehanizem NMAR (opis mehanizmov za nastanek manjkajočih podatkov se nahaja v poglavju 3.1).

Na grafih sta prikazani meri podobnosti (CC) in različnosti (RMS) za primerjavo uteži in lastnih vrednosti (opis mer se nahaja v poglavju 5.3) v odvisnosti od odstotka enot z manjkajočimi podatki, ki je vsakič zavzemal vrednosti od 0 do 60, prikazane pa so tudi tabele, kjer je za vsako podatkovno bazo prikazana razvrstitev metod od najboljše do najslabše glede na CC in RMS pri 60 % enot z manjkajočimi podatki (vrstni red metod pri različnih odstotkih enot z manjkajočimi podatki je lahko različen, zato takšna razvrstitev ne predstavlja absolutne mere učinkovitosti metod). Zaradi zagotavljanja preglednosti rezultatov so razponi ordinatne osi na grafih različni.

Rezultati metode k-najbližjih sosedov so prikazani za vrednost k = 10, na koncu vsakega sklopa pa so ločeno predstavljeni še rezultati metode k-najbližjih sosedov za različne k (k = 1,3,5,10,20).

Spodaj je podan seznam okrajšav imen metod za obravnavo manjkajočih podatkov, ki jih uporabljamo v tem poglavju.

listwise	analiza na osnovi popolnih enot
pairwise	analiza na osnovi razpoložljivih podatkov
vstavljanje povprečij	vstavljanje aritmetične sredine
RVI	vstavljanje naključnih vrednosti
KNN[8]	metoda k-najbližjih sosedov
EM algoritem	EM algoritem
MICE - PMM	večkratno vstavljanje z verižnimi enačbami (metoda PMM)

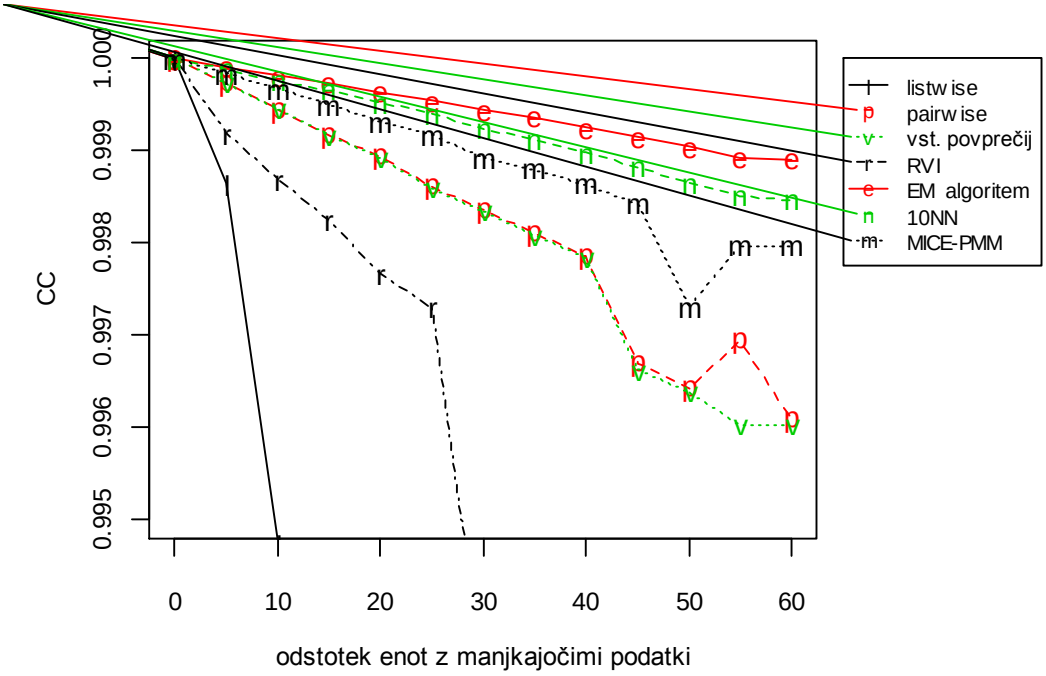
[8] Metoda je ob uporabi različnih k ustrezno preimonavana.

6.1 Predstavitev rezultatov MCAR

6.1.1 Rezultati za uteži na prvih dveh oz. treh glavnih komponentah

Po formulah 5.4 in 5.2 smo izračunali CC in RMS za uteži prvih dveh oz. treh komponent skupaj (glede na število »primernih« glavnih komponent za vsako podatkovno bazo, pri določanju števila »primernih« glavnih komponent smo se opirali na hevristične postopke iz 2. poglavja, glej tudi opis podatkovnih baz – poglavje 5.4).[9]

Slika 6.1: Mera podobnosti CC za uteži na prvih dveh komponentah za MCAR – podatkovna baza »Mednarodna anketa«



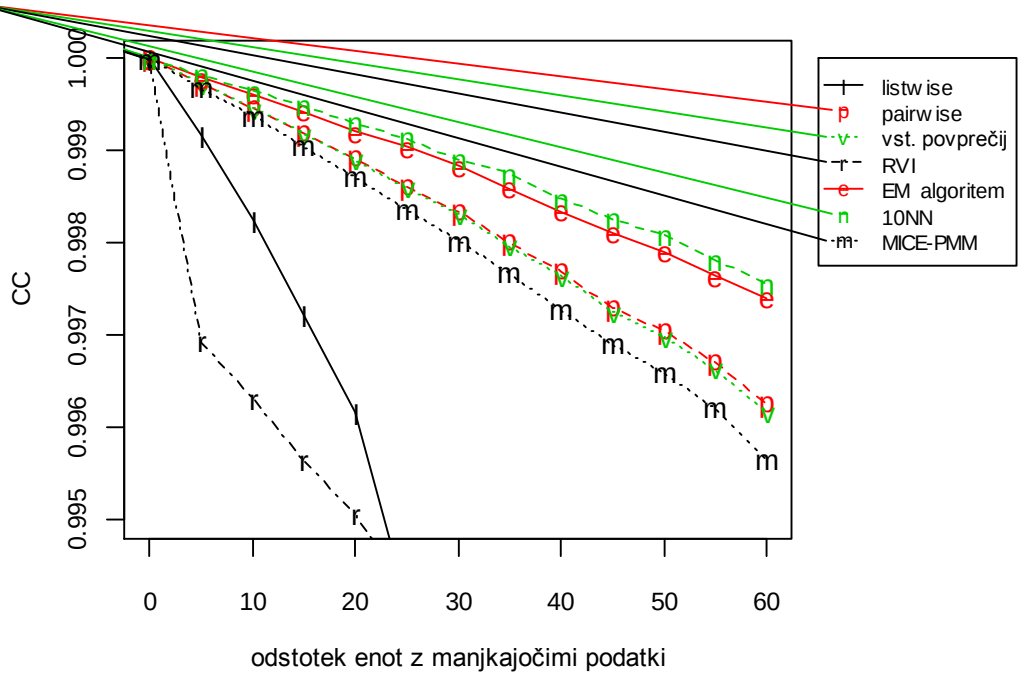
Na podatkovni bazi »Mednarodna anketa« (glej sliko 6.1) se glede na CC najbolje izkažejo metode EM algoritem, 10NN in MICE-PMM, najslabše pa analiza na osnovi popolnih enot, ki ji sledi RVI (obe metodi od ostalih močno odstopata). CC je za vse metode brez analize na

[9] Na podatkovnih bazah »Mednarodna anketa« in »Nova vozila« smo CC izračunali za prvi dve glavni komponenti, na podatkovni bazi »Prepoznavnost vin« pa za prve tri glavne komponente.

osnovi popolnih enot (ko ima manjkajoče podatke več kot 15 % enot) glede na Tuckerjeve smernice pri vseh odstotkih enot z manjkajočimi podatki odličen (med 0,98 in 1,00). Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.1 v prilogi B).

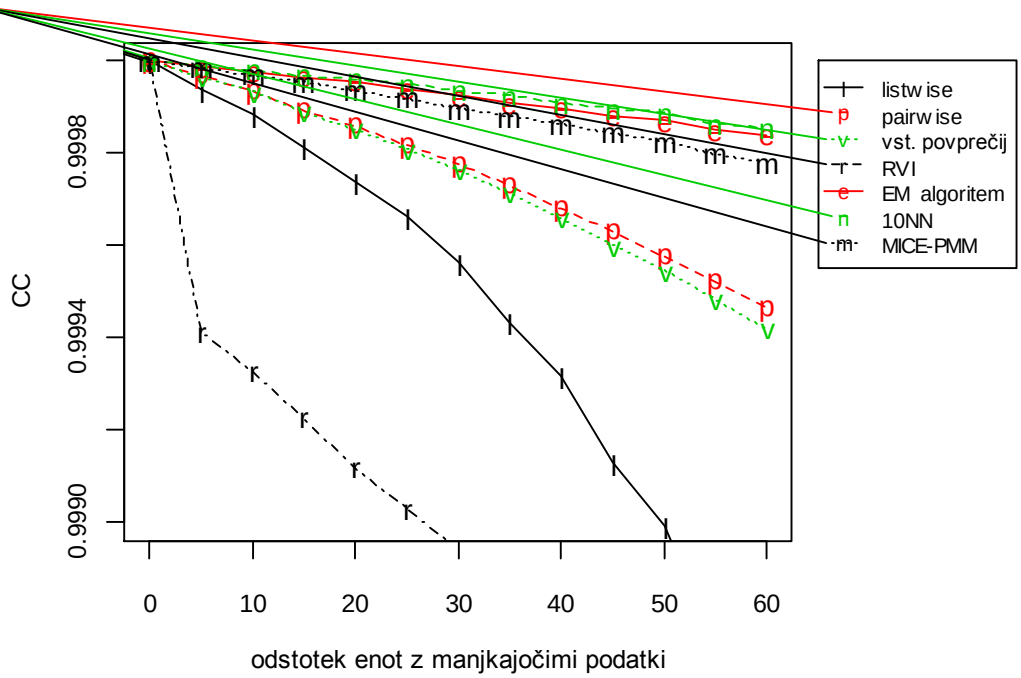
Na podatkovni bazi »Prepoznavnost vin« (glej sliko 6.2) se glede na CC najbolje izkažeta metodi 10NN in EM algoritem (obe metodi sta približno enakovredni, saj je CC pri različnih odstotkih enot z manjkajočimi podatkih pri obeh metodah skorajda identičen), najslabše pa se izkaže analiza na osnovi popolnih enot, ki ji sledi RVI. Skladnost je glede na Tuckerjeve smernice za vse metode brez analize na osnovi popolnih enot (ko ima manjkajoče podatke več kot 45 % enot) odlična (CC nad 0,98). Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.4 v prilogi B).

Slika 6.2: Mera podobnosti CC za uteži na prvih treh komponentah za MCAR – podatkovna baza »Prepoznavnost vin«



Na podatkovni bazi »Nova vozila« (glej sliko 6.3) se glede na CC najbolje izkažeta metodi 10NN in EM algoritem, sledi MICE-PMM, najslabše pa se izkaže RVI, ki ji sledi analiza na osnovi popolnih enot. Skladnost z utežmi na popolnih podatkih je za vse metode pri vseh odstotkih enot z manjkajočimi podatki odlična (CC je nad 0,98). Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.7 v prilogi B).

Slika 6.3: Mera podobnosti CC za uteži na prvih dveh komponentah za MCAR – podatkovna baza »Nova vozila«



S spodnje tabele (6.1), kjer primerjamo vrstni red metod glede na CC pri 60 % enot z manjkajočimi podatki, je razvidno, da se na vseh podatkovnih bazah najbolje izkažeta metodi 10NN in EM algoritem, najslabše pa metodi RVI in analiza na osnovi popolnih enot.

Tabela 6.1: Vrstni red metod glede na CC za uteži na prvih dveh oz. prvih treh komponentah pri 60 % enot z manjkajočimi podatki za MCAR

	Podatkovna baza		
	Mednarodna anketa	**Prepoznavnost vin**	**Nova vozila**
Najboljša	EM algoritem	10NN	10NN
	10NN	EM algoritem	EM algoritem
	MICE-PMM	pairwise	MICE-PMM
	pairwise	vstavljanje povprečij	pairwise
	vstavljanje povprečij	MICE-PMM	vstavljanje povprečij
	RVI	RVI	listwise
Najslabša	listwise	listwise	RVI

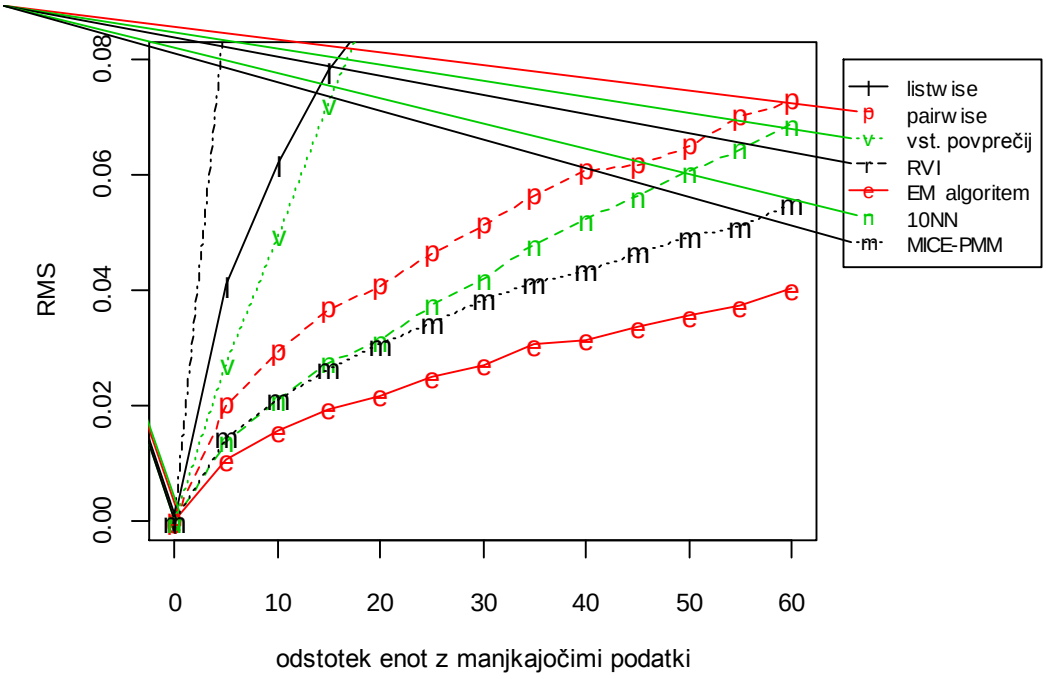
Preizkusili smo tudi, kaj se z rezultati algoritma KNN dogaja pri različnih k. Pri vseh odstotkih enot z manjkajočimi podatki se glede na CC najbolje izkaže 10NN (razen na bazi »Nova vozila«, kjer se poleg 10NN približno enako dobro izkaže tudi 5NN), vendar pa so razlike med rezultati KNN (k = 1,3,5,10,20) izredno majhne (glej prilogo B). Skladnost z

utežmi na popolnih podatkih je za vse k pri vseh odstotkih enot z manjkajočimi podatki glede na Tuckerjeve smernice odlična pri vseh treh podatkovnih bazah (CC je nad 0,98), iz česar lahko sklepamo, da je algoritem v tem primeru relativno neobčutljiv na vrednosti k (ko se k nahaja med 1 in 20). Rezultati za metodo KNN ob izbiri različnih k so prikazani v tabelah B.2, B.5 in B.8 v prilogi B.

6.1.2 Rezultati za lastne vrednosti prvih dveh oz. prvih treh glavnih komponent

Za vse tri podatkovne baze smo po formuli 5.3 izračunali RMS za lastne vrednosti prvih dveh oz. treh glavnih komponent (glede na število »primernih« glavnih komponent za vsako podatkovno bazo).

Slika 6.4: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za MCAR – podatkovna baza »Mednarodna anketa«

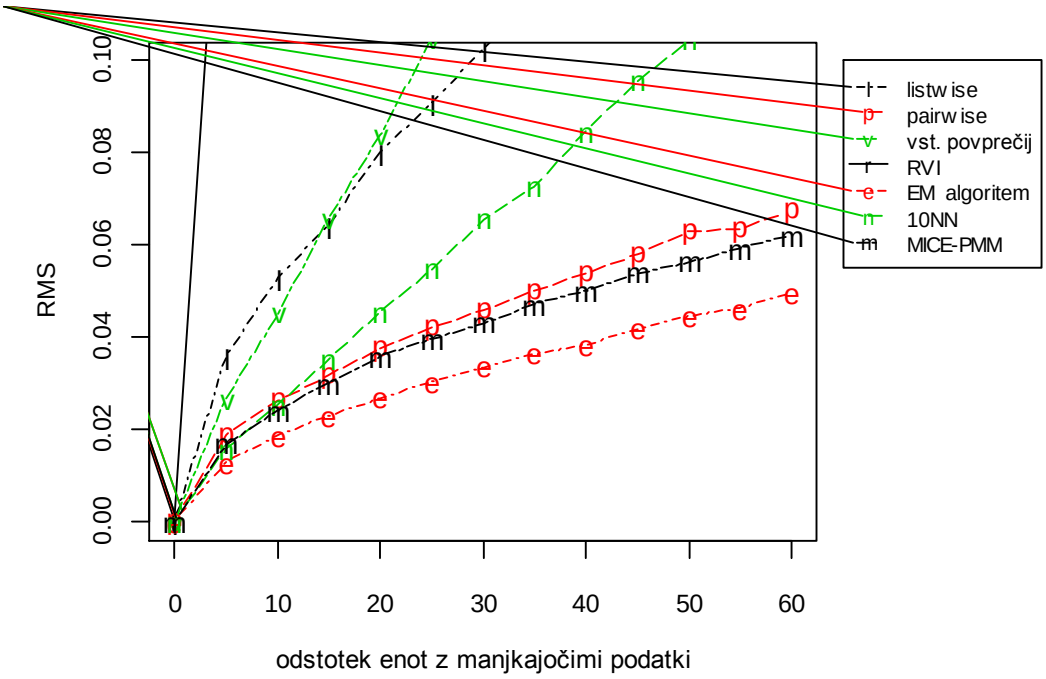


Na podatkovni bazi »Mednarodna anketa« (glej sliko 6.4) se najbolje izkaže EM algoritem, sledita MICE-PMM in 10NN (metodi sta približno enakovredni do 20 % enot z manjkajočimi podatki, nad 20 % pa se metoda MICE-PMM izkaže nekoliko bolje od 10NN), najslabše pa se izkaže metoda RVI, ki ji sledita vstavljanje povprečij in analiza na osnovi popolnih enot.

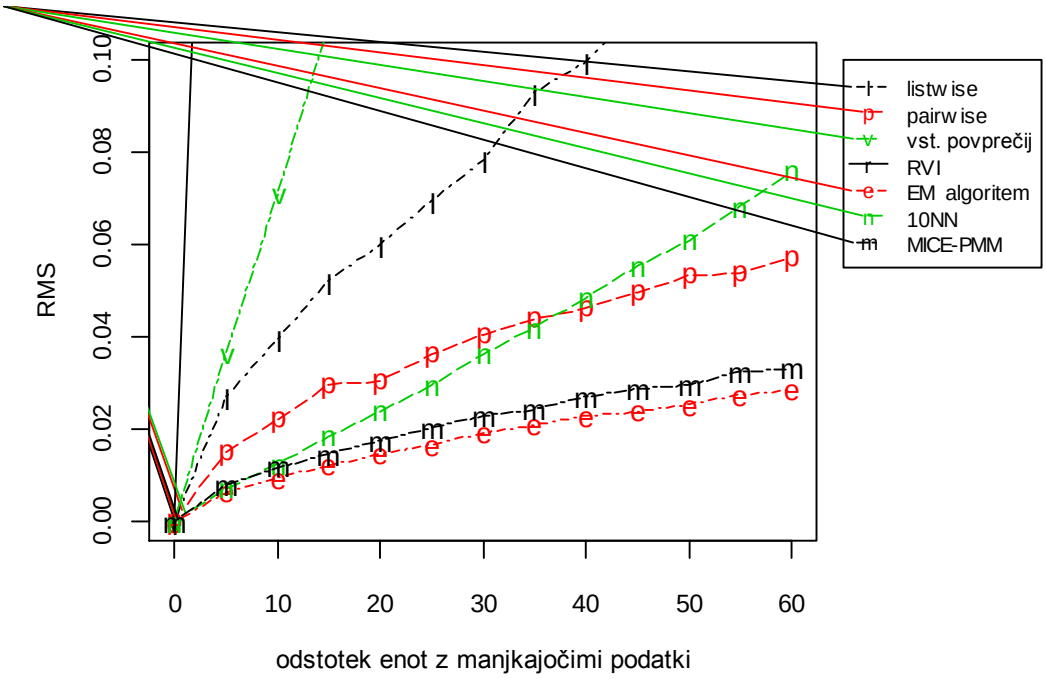
Na podatkovni bazi »Prepoznavnost vin« (glej sliko 6.5) se najbolje izkaže EM algoritem, sledita MICE-PMM in analiza na osnovi razpoložljivih podatkov (ki sta skorajda povsem

skladni), najslabše pa se izkaže RVI, ki ji sledita vstavljanje povprečij in analiza na osnovi popolnih enot. Metoda 10NN se na tej podatkovni bazi izkaže precej slabo.

Slika 6.5: Mera različnosti RMS za lastne vrednosti prvih treh glavnih komponent za MCAR – podatkovna baza »Prepoznavnost vin«



Slika 6.6: Mera različnosti RMS za lastne vrednosti prvih dveh glavnih komponent za MCAR – podatkovna baza »Nova vozila«



Na podatkovni bazi »Nova vozila« (glej sliko 6.6) se enako kot na prvih dveh podatkovnih bazah najbolje izkažeta EM algoritem in MICE-PMM (EM algoritem malenkost bolje od MICE-PMM), najslabše pa se ponovno izkaže RVI, ki ji sledita vstavljanje povprečij in analiza na osnovi popolnih enot.

Vrstni red metod glede na RMS za lastne vrednosti prvih dveh oz. treh glavnih komponent pri 60 % enot z manjkajočimi podatki je za vse tri podatkovne baze približno enak (glej tabelo 6.2). Kot najboljši se izkažeta metodi EM algoritem in večkratno vstavljanje, sledita analiza na osnovi razpoložljivih podatkov in 10NN, kot najslabša pa se izkaže metoda RVI.

Tabela 6.2: Vrstni red metod glede na RMS za lastne vrednosti prvih dveh oz. treh glavnih komponent pri 60 % enot z manjkajočimi podatki za MCAR

	Podatkovna baza		
	Mednarodna anketa	**Prepoznavnost vin**	**Nova vozila**
Najboljša	EM algoritem	EM algoritem	EM algoritem
	MICE-PMM	MICE-PMM	MICE-PMM
⇩	10NN	pairwise	Pairwise
	pairwise	10NN	10NN
	listwise	listwise	Listwise
	vstavljanje povprečij	vstavljanje povprečij	vstavljanje povprečij
Najslabša	RVI	RVI	RVI

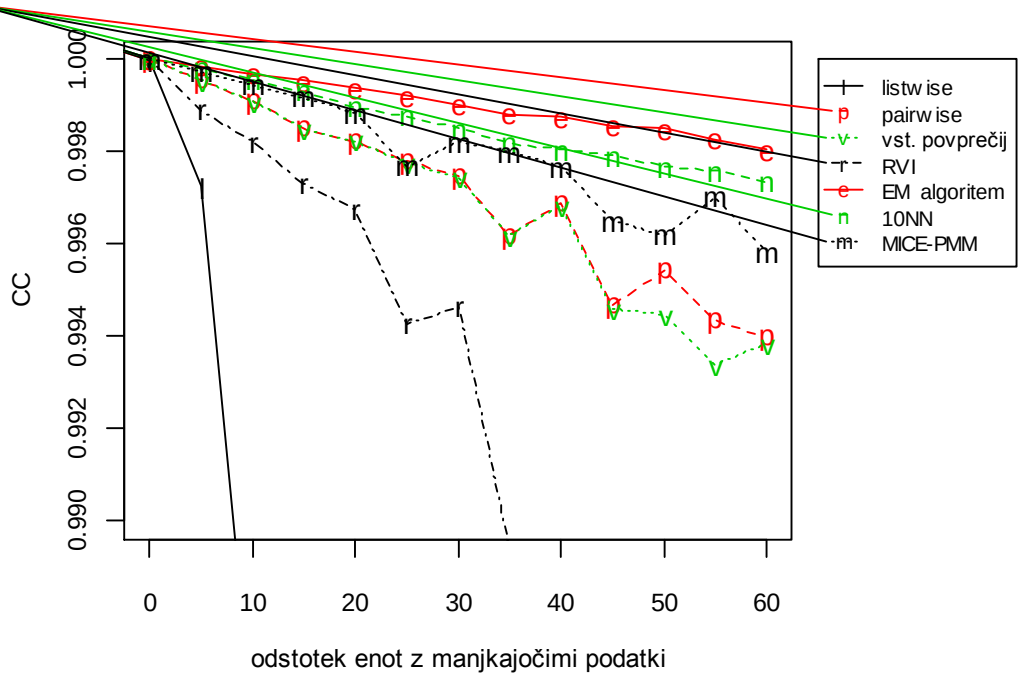
Preizkusili smo tudi, kaj se z rezultati algoritma KNN dogaja pri različnih k. Na bazi »Mednarodna anketa« se najbolje izkaže 20NN (ob izbiri različnih k se RMS spreminja minimalno), na bazah »Prepoznavnost vin« in »Nova vozila« pa 1NN (na obeh bazah nižji k pomeni tudi nižji RMS). Skladnost z lastnimi vrednostmi na popolnih podatkih je za vse k pri vseh odstotkih enot z manjkajočimi podatki relativno visoka pri vseh treh podatkovnih bazah (najvišja vrednost RMS je 0,135). Umestitev metode KNN glede na ostale metode se ob uporabi optimalnega k na bazi »Mednarodna anketa« ne bi spremenila, na podatkovni bazi »Prepoznavnost vin« bi se metoda umestila takoj za EM algoritem in MICE-PMM (torej bi se izkazala bolje od analize na osnovi razplożljivih podatkov), na podatkovni bazi »Nova vozila« pa bi se metoda KNN povsem približala MICE-PMM (podobno kot na podatkovni bazi »Prepoznavnost vin« bi se metoda izkazala bolje od analize na osnovi razpoložljivih podatkov). Rezultati za metodo KNN ob izbiri različnih k so prikazani v tabelah B.3, B.6 in B.9 v prilogi B.

6.2 Predstavitev rezultatov MAR

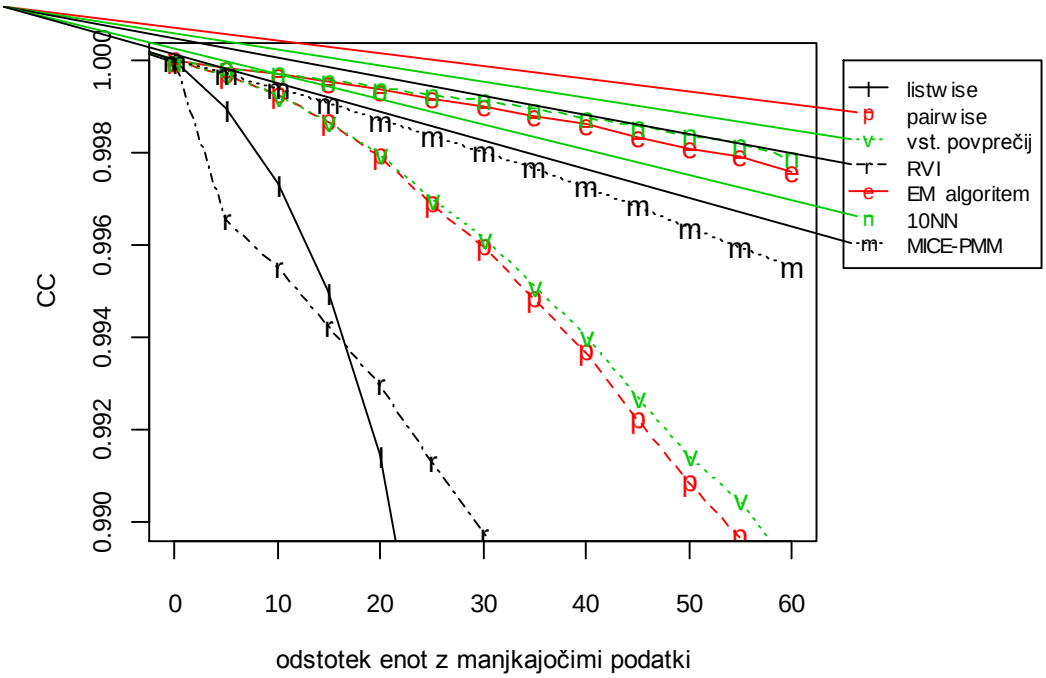
6.2.1 Rezultati za uteži na prvih dveh oz. treh glavnih komponentah

Na podatkovni bazi »Mednarodna anketa« (glej sliko 6.7) se glede na CC najbolje izkažeta metodi EM algoritem in 10NN, sledi jima metoda MICE-PMM, najslabše pa metoda analiza na osnovi popolnih enot, ki ji sledi RVI (ti dve metodi od ostalih močno odstopata). CC je za vse metode razen za RVI in analizo na osnovi popolnih enot glede na Tuckerjeve smernice pri vseh odstotkih enot z manjkajočimi podatki odličen (med 0,98 in 1,00). Metode so se izkazale podobno kot pri MCAR, le da je CC pri vseh metodah in pri vseh odstotokih enot z manjkajočimi podatki malenkost višji. Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.10 v prilogi B).

Slika 6.7: Mera podobnosti CC za uteži na prvih dveh komponentah za MAR – podatkovna baza »Mednarodna anketa«



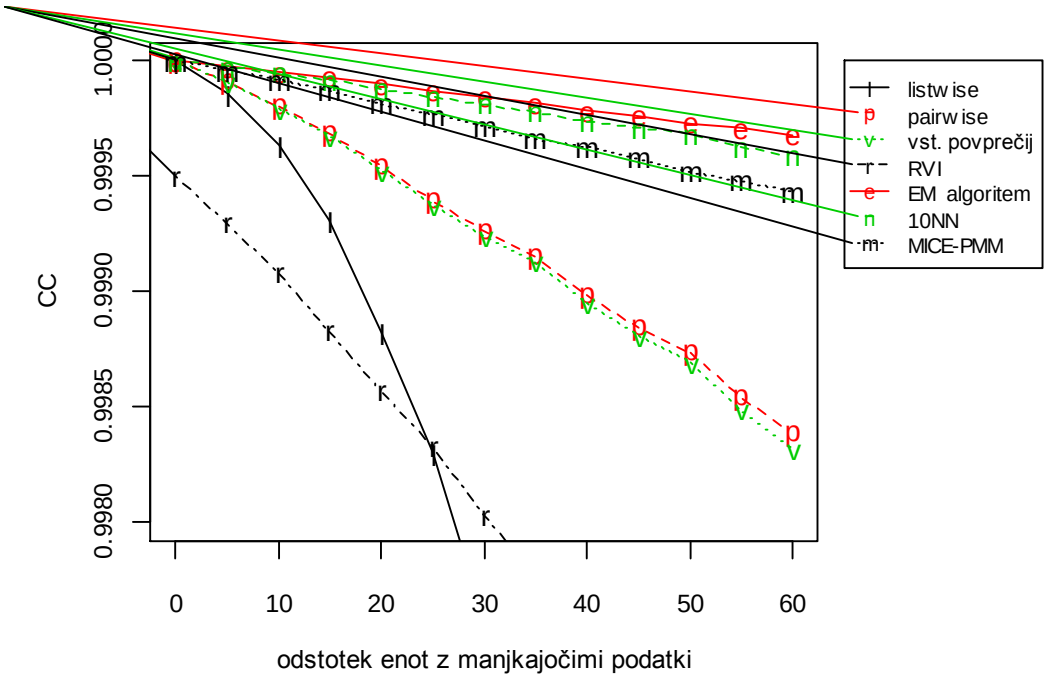
Slika 6.8: Mera podobnosti CC za uteži na prvih treh komponentah za MAR – podatkovna baza »Prepoznavnost vin«



Na podatkovni bazi »Prepoznavnost vin« (glej sliko 6.8) se glede na CC najbolje izkažeta metodi 10NN in EM algoritem, sledi MICE-PMM, najslabše pa ponovno metodi analiza na osnovi popolnih enot in RVI. Skladnost je glede na Tuckerjeve smernice za vse metode (brez analize na osnovi popolnih enot, ko ima manjkajoče podatke več kot 25 % enot in RVI, ko ima manjkajoče podatke 60 % enot) odlična (CC nad 0,98). Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.13 v prilogi B).

Na podatkovni bazi »Nova vozila« (glej sliko 6.9) se glede na CC najbolje izkažeta metodi EM algoritem in 10NN, sledi MICE-PMM, najslabše pa se izkaže metoda analiza na osnovi popolnih enot, ki ji sledi RVI. Skladnost z utežmi na popolnih podatkih je za vse metode pri vseh odstotkih enot z manjkajočimi podatki odlična (CC je nad 0,98). Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.16 v prilogi B).

Slika 6.9: Mera podobnosti CC za uteži na prvih dveh komponentah za MAR – podatkovna baza »Nova vozila«



Iz tabele 6.3, kjer primerjamo vrstni red metod glede na CC pri 60 % enot z manjkajočimi podatki, je razvidno, da se za izračun uteži pri mehanizmu MAR najbolje izkažejo metode EM algoritem, 10NN in MICE-PMM, najslabše pa metoda analiza na osnovi popolnih enot, ki ji sledi RVI.

Tabela 6.3: Vrstni red metod glede na CC za uteži na prvih dveh oz. prvih treh komponentah pri 60 % enot z manjkajočimi podatki za MAR

	Podatkovna baza		
	Mednarodna anketa	**Prepoznavnost vin**	**Nova vozila**
Najboljša	EM algoritem	10NN	EM algoritem
	10NN	EM algoritem	10NN
	MICE-PMM	MICE-PMM	MICE-PMM
	pairwise	vstavljanje povprečij	Pairwise
	vstavljanje povprečij	pairwise	vstavljanje povprečij
	RVI	RVI	RVI
Najslabša	listwise	listwise	Listwise

Preizkusili smo tudi, kaj se z rezultati algoritma KNN dogaja pri različnih k. Razlike med rezultati KNN (k = 1,3,5,10,20) so izredno majhne (glej prilogo B). Na bazah »Mednarodna anketa« in »Prepoznavnost vin« se najbolje izkažeta 10NN in 20NN, na bazi »Nova vozila«

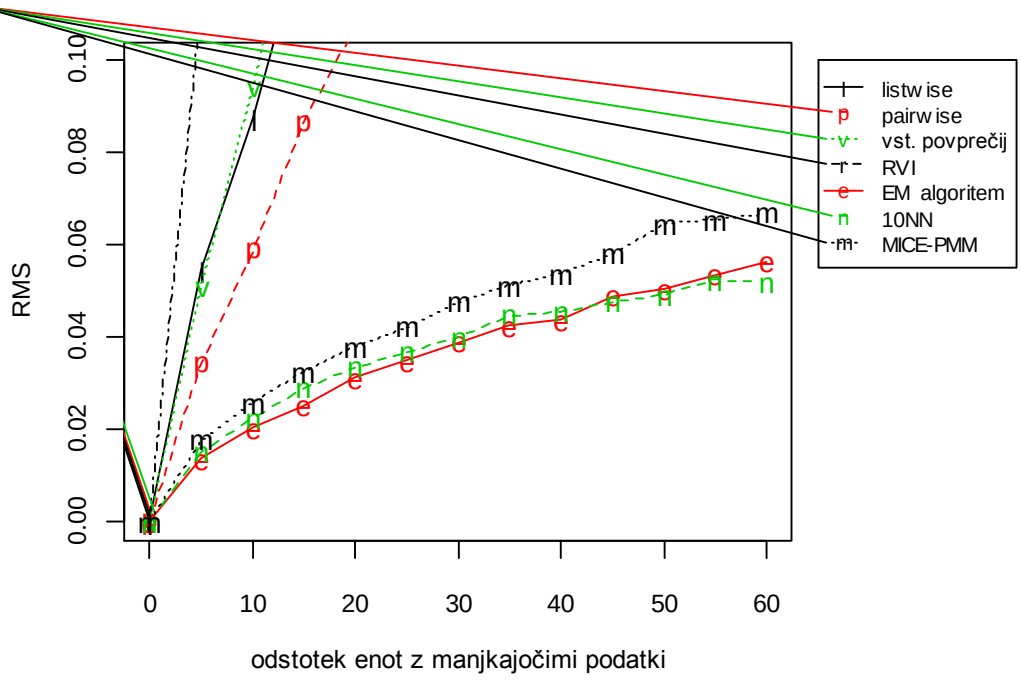
pa 5NN. Skladnost z utežmi na popolnih podatkih je za vse k pri vseh odstotkih enot z manjkajočimi podatki odlična pri vseh treh podatkovnih bazah (CC je nad 0,98), iz česar lahko sklepamo, da je algoritem tudi pri MAR (podobno kot pri MCAR) relativno neobčutljiv na vrednosti k (ko se k nahaja med 1 in 20). Umestitev metode KNN glede na ostale metode se ob uporabi optimalnega k na nobeni izmed treh podatkovnih baz ne spremeni. Rezultati za metodo KNN ob izbiri različnih k so prikazani v tabelah B.11, B.14 in B.17 v prilogi B.

6.2.2 Rezultati za lastni vrednosti prvih dveh oz. prvih treh glavnih komponent

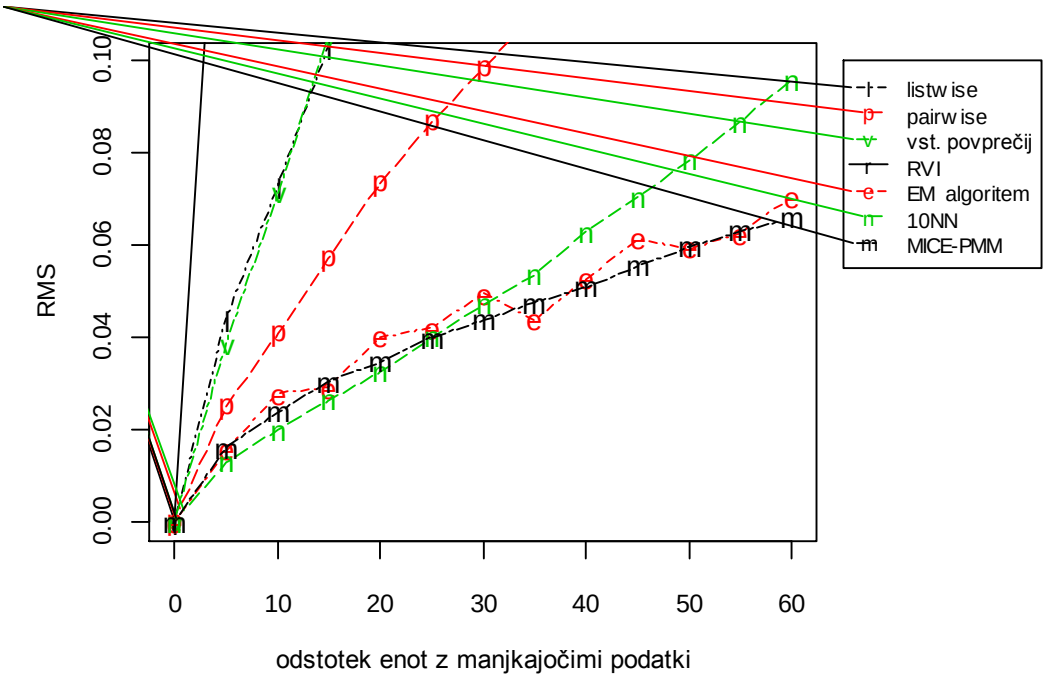
Na podatkovni bazi »Mednarodna anketa« (glej sliko 6.10) se glede na RMS najbolje izkažeta metodi 10NN in EM algoritem, sledi MICE-PMM, ostale metode pa se izkažejo precej slabo (najslabše se izkaže analiza na osnovi popolnih enot, ki ji sledita RVI in vstavljanje povprečij).

Na podatkovni bazi »Prepoznavnost vin« (glej sliko 6.11) se glede na RMS najbolje izkažeta metodi MICE-PMM in EM algoritem, sledi metoda 10NN, najslabše pa se izkaže metoda RVI, ki ji sledita vstavljanje povprečij in analiza na osnovi popolnih enot.

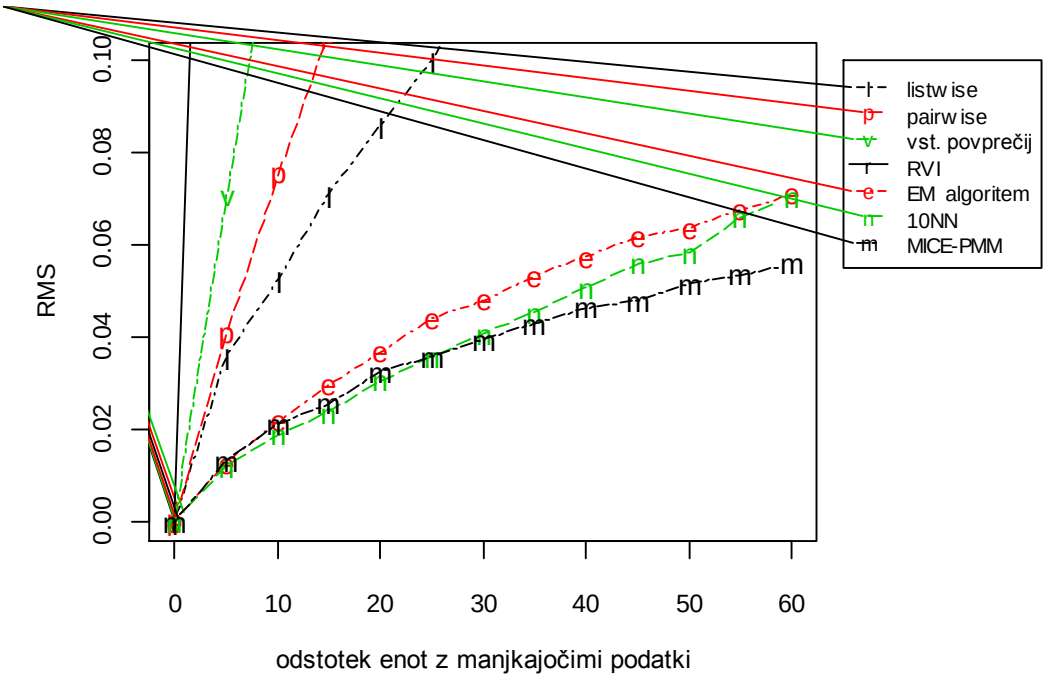
Slika 6.10: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za MAR – podatkovna baza »Mednarodna anketa«



Slika 6.11: Mera različnosti RMS za lastne vrednosti prvih treh glavnih komponent za MAR – podatkovna baza »Prepoznavnost vin«



Slika 6.12: Mera različnosti RMS za lastne vrednosti prvih dveh glavnih komponent za MAR – podatkovna baza »Nova vozila«



Na podatkovni bazi »Nova vozila« (glej sliko 6.12) se glede na RMS najbolje izkaže

metoda MICE-PMM, sledita 10NN in EM algoritem, najslabše pa se izkaže metoda RVI, ki ji sledi vstavljanje povprečij.

Vrstni red metod glede na RMS za lastne vrednosti prvih dveh oz. treh glavnih komponent pri 60 % enot z manjkajočimi podatki (glej tabelo 6.4) se za tri podatkovne baze nekoliko razlikuje: najbolje se izkažejo metode MICE-PMM, EM algoritem in 10NN.

Tabela 6.4: Vrstni red metod glede na RMS za lastne vrednosti prvih dveh oz. treh glavnih komponent pri 60 % enot z manjkajočimi podatki za MAR

	Podatkovna baza		
	Mednarodna anketa	**Prepoznavnost vin**	**Nova vozila**
Najboljša	10NN	MICE-PMM	MICE-PMM
	EM algoritem	EM algoritem	10NN
	MICE-PMM	10NN	EM algoritem
	pairwise	pairwise	Listwise
	listwise	listwise	Pairwise
	vstavljanje povprečij	vstavljanje povprečij	vstavljanje povprečij
Najslabša	RVI	RVI	RVI

Preizkusili smo, kaj se z rezultati algoritma KNN dogaja pri različnih k. Na bazi »Mednarodna anketa« se najbolje izkaže 10NN (ob izbiri različnih k se RMS spreminja minimalno, nekoliko slabše se izkaže le 1NN), na bazi »Prepoznavnost vin« se najbolje izkažeta 1NN in 3NN, na bazi »Nova vozila« pa 3NN. Metoda KNN bi se glede na ostale metode ob uporabi optimalnega k na podatkovni bazi »Prepoznavnost vin« povsem približala metodama MICE-PMM in EM algoritmu (vse tri metode bi bile približno enakovredne), na podatkovni bazi »Nova vozila« pa bi se KNN izkazala celo nekoliko bolje od metode MICE-PMM. Rezultati za metodo KNN ob izbiri različnih k so prikazani v tabelah B.12, B.15 in B.18 v prilogi B.

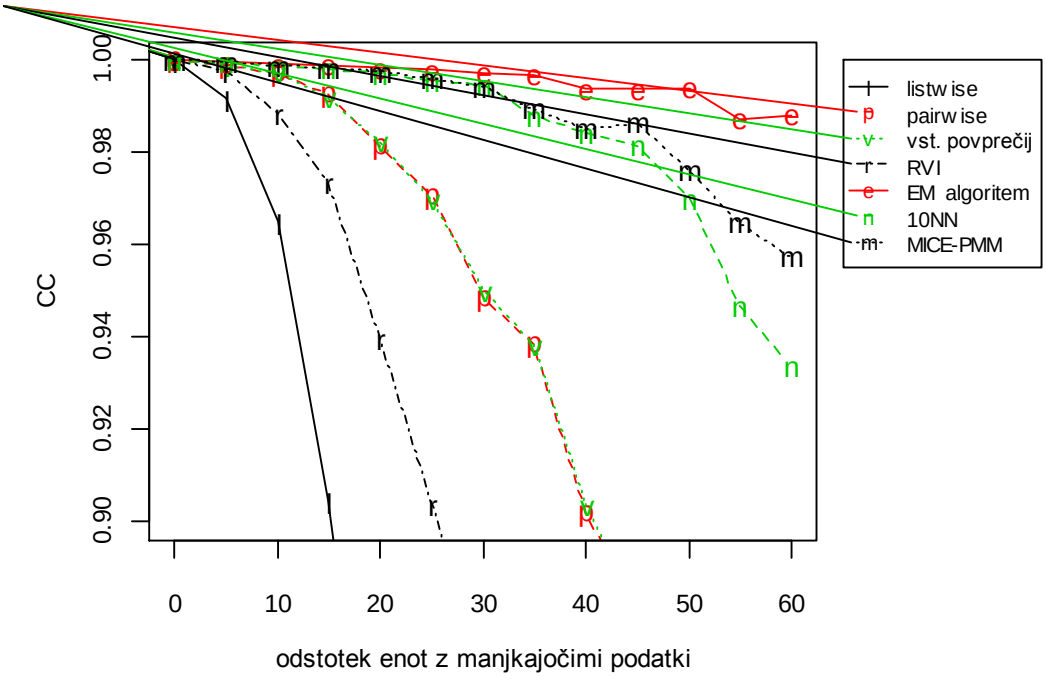
6.3 Predstavitev rezultatov NMAR

6.3.1 Rezultati za uteži na prvih dveh oz. treh glavnih komponentah

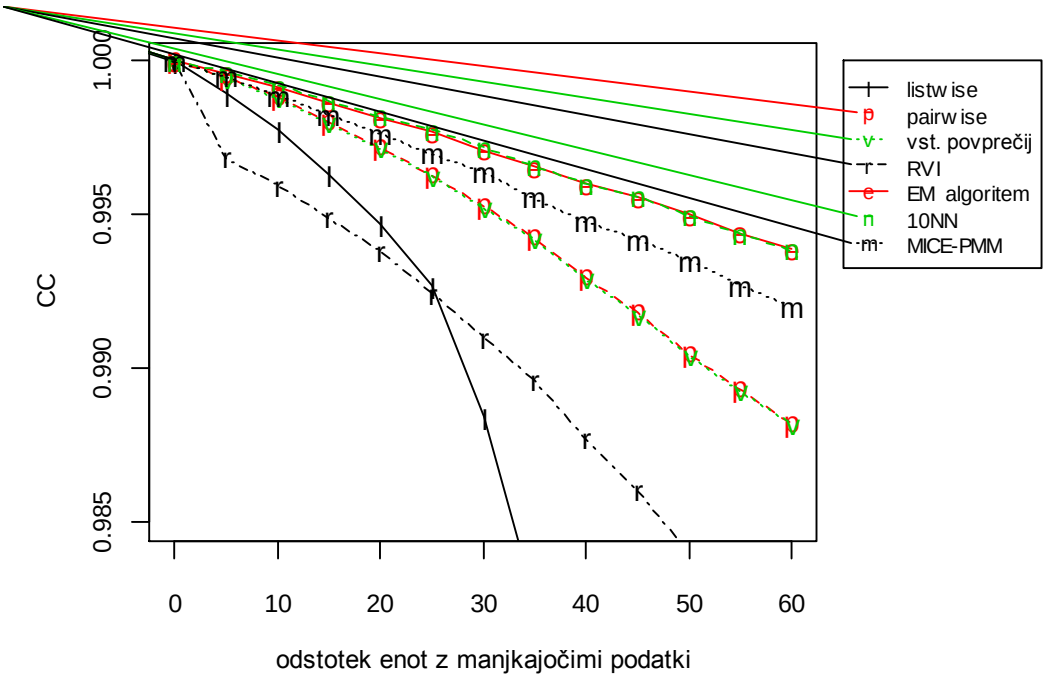
Na podatkovni bazi »Mednarodna anketa« (glej sliko 6.13) se glede na CC najbolje izkaže metoda EM algoritem, sledita metodi MICE-PMM in 10NN, najslabše pa se izkaže analiza na osnovi popolnih enot, ki ji sledi metoda RVI. CC je glede na Tuckerjeve smernice pri vseh odstotkih enot z manjkajočimi podatki odličen (med 0,98 in 1,00) le pri EM algoritmu. Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.19 v prilogi B).

Na podatkovni bazi »Prepoznavnost vin« (glej sliko 6.14) se glede na CC najbolje izkažeta metodi EM algoritem in 10NN (metodi se skorajda povsem prekrivata), sledi MICE-PMM, najslabše pa se ponovno izkaže analiza na osnovi popolnih enot, ki ji sledi RVI. Skladnost je glede na Tuckerjeve smernice za vse metode (brez analize na osnovi popolnih enot, ko ima manjkajoče podatke več kot 35 % enot, odlična (CC nad 0,98). Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.22 v prilogi B).

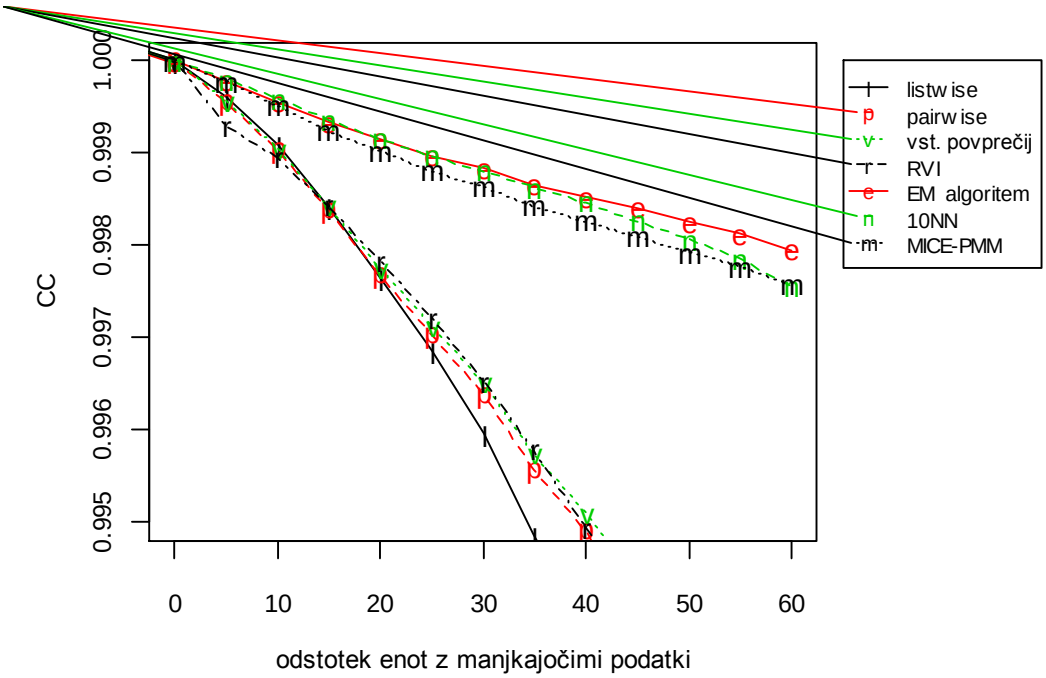
Slika 6.13: Mera podobnosti CC za uteži na prvih dveh komponentah za NMAR – podatkovna baza »Mednarodna anketa«



Slika 6.14: Mera podobnosti CC za uteži na prvih treh komponentah za NMAR – podatkovna baza »Prepoznavnost vin«



Slika 6.15: Mera podobnosti CC za uteži na prvih dveh komponentah za NMAR – podatkovna baza »Nova vozila«



Na podatkovni bazi »Nova vozila« (glej sliko 6.15) se glede na CC najbolje izkažejo metode EM algoritem, MICE-PMM in 10NN, preostale metode pa se izkažejo precej slabše.

Zanimivo je, da je skladnost z utežmi na popolnih podatkih za vse metode (tudi analizo na osnovi popolnih enot) pri vseh odstotkih enot z manjkajočimi podatki odlična (CC je nad 0,98), kar seveda pri NMAR nismo pričakovali. Najverjetneje je do tega prišlo zato, ker so spremenljivke v tej podatkovni bazi močno povezane med seboj (glej prilogo A) in ima zato brisanje podatkov po mehanizmu NMAR na korelacijsko matriko nekoliko manjši vpliv, kot bi ga imelo, če bi bila povezanost med spremenljivkami manjša. Med rezultati CC in RMS ni večjih razlik (za rezultate RMS glej tabelo B.25 v prilogi B).

Iz tabele 6.5, kjer primerjamo vrstni red metod glede na CC pri 60 % enot z manjkajočimi podatki, je razvidno, da je vrstni red metod za vse tri podatkovne baze zelo podoben. Najbolje se izkaže EM algoritem, sledita pa metodi MICE-PMM in 10NN. Ostale metode se izkažejo precej slabše, kar je bilo glede na mehanizem nastanka manjkajočih podatkov NMAR tudi pričakovano.

Tabela 6.5: Vrstni red metod glede na CC za uteži na prvih dveh oz. prvih treh komponentah pri 60 % enot z manjkajočimi podatki za NMAR

	Podatkovna baza		
	Mednarodna anketa	**Prepoznavnost vin**	**Nova vozila**
Najboljša	EM algoritem	EM algoritem	EM algoritem
	MICE-PMM	10NN	MICE-PMM
	10NN	MICE-PMM	10NN
	pairwise	pairwise	Listwise
	vstavljanje povprečij	vstavljanje povprečij	Pairwise
	RVI	RVI	RVI
Najslabša	listwise	listwise	vstavljanje povprečij

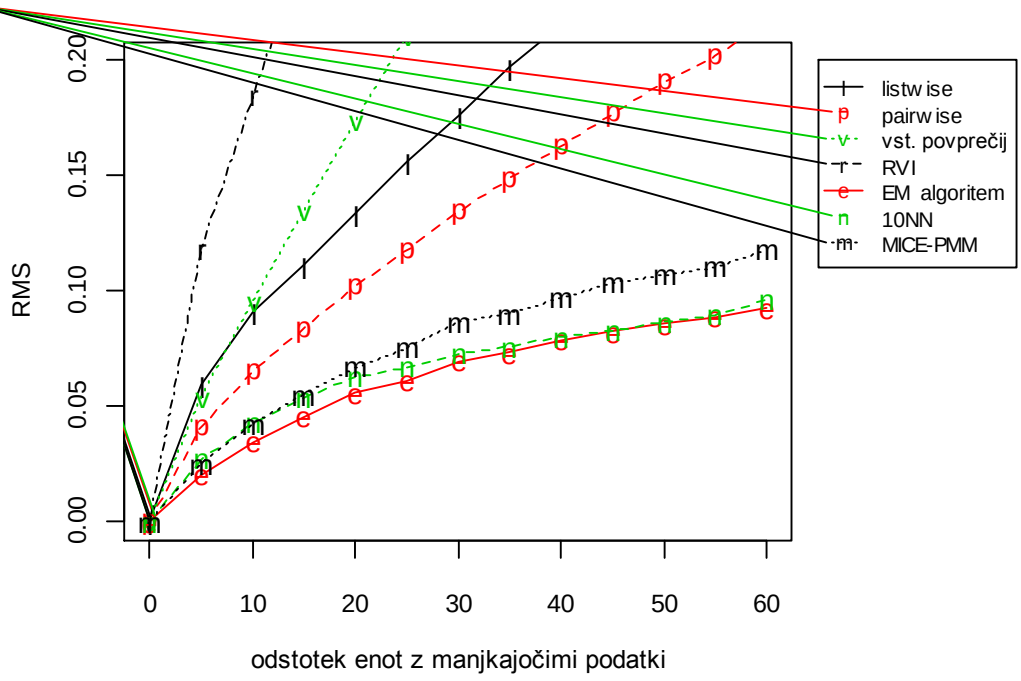
Preizkusili smo tudi, kaj se z rezultati algoritma KNN dogaja pri različnih k. Razlike med rezultati KNN (k = 1,3,5,10,20) so relativno majhne (glej prilogo B). Na bazah »Mednarodna anketa« in »Prepoznavnost vin« se glede na CC najbolje izkaže 5NN, na bazi »Nova vozila« pa 1NN. Umestitev metode KNN glede na ostale metode se ob uporabi optimalnega k na bazah »Mednarodna anketa« in »Prepoznavnost vin« ne bi spremenila, na podatkovni bazi »Nova vozila« pa bi se metoda izkazala bolje tako od EM algoritma kot tudi od metode MICE-PMM. Skladnost z utežmi na popolnih podatkih je za vse k pri vseh odstotkih enot z manjkajočimi podatki odlična na podatkovnih bazah »Prepoznavnost vin« in »Nova vozila« (CC je nad 0,98), na podatkovni bazi »Mednarodna anketa« pa dobra oz. mejna (CC nad 0,82

in pod 0,98). Rezultati za metodo KNN ob izbiri različnih k so prikazani v tabelah B.20, B.23 in B.26 v prilogi B.

6.3.2 Rezultati za lastni vrednosti prvih dveh oz. prvih treh glavnih komponent

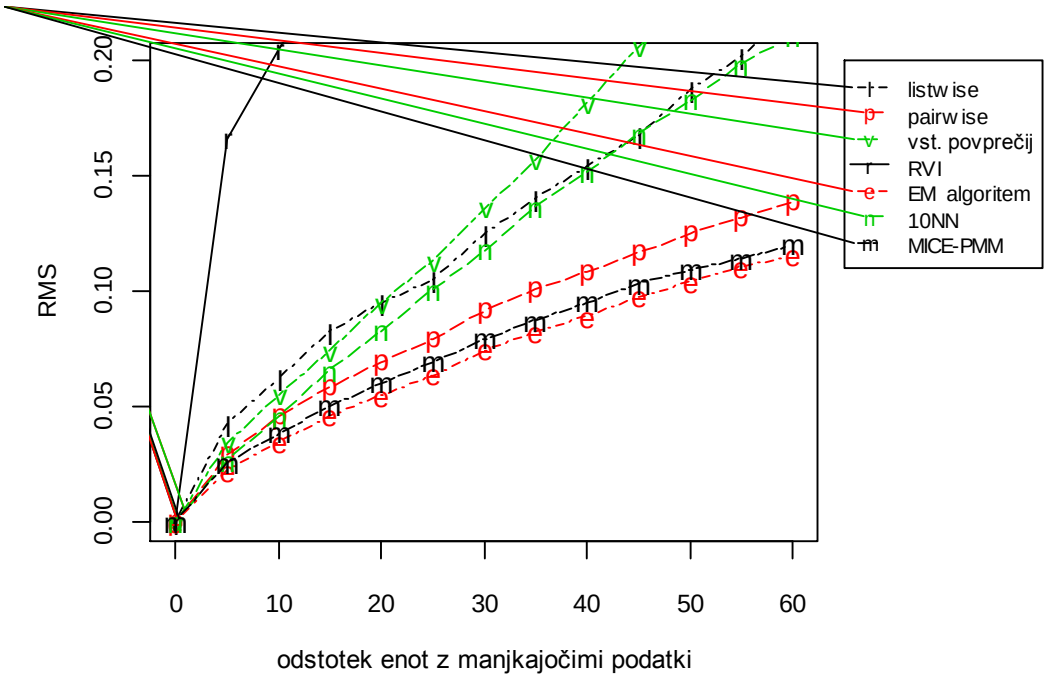
Na podatkovni bazi »Mednarodna anketa« (glej sliko 6.16) se najbolje izkažeta metodi EM algoritem in 10NN, sledi MICE-PMM, najslabše pa se izkaže metoda RVI, ki ji sledita vstavljanje povprečij in analiza na osnovi popolnih enot.

Slika 6.16: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za NMAR – podatkovna baza »Mednarodna anketa«

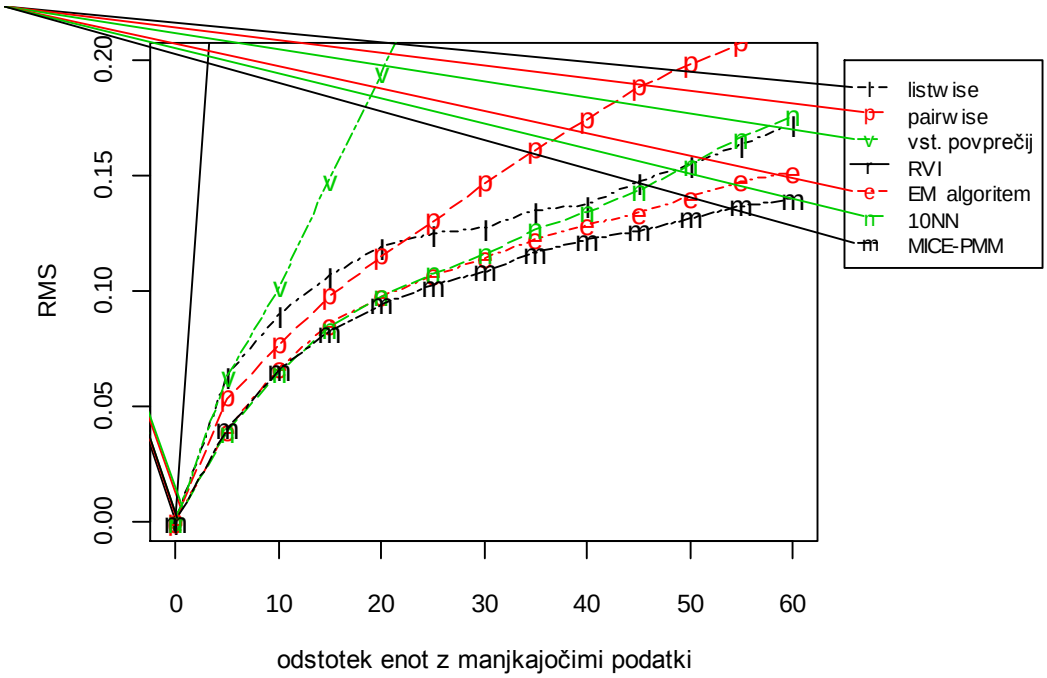


Na podatkovni bazi »Prepoznavnost vin« (glej sliko 6.17) se najbolje izkažeta EM algoritem in MICE-PMM, sledi metoda analiza na osnovi razpoložljivih enot, ki je izredno blizu najboljšima dvema, daleč najslabše pa se izkaže metoda RVI.

Slika 6.17: Mera različnosti RMS za lastne vrednosti prvih treh glavnih komponent za NMAR – podatkovna baza »Prepoznavnost vin«



Slika 6.18: Mera različnosti RMS za lastne vrednosti prvih dveh glavnih komponent za NMAR – podatkovna baza »Nova vozila«



Na podatkovni bazi »Nova vozila« (glej sliko 6.18) se najbolje izkažeta MICE-PMM in EM algoritem, ki jima sledita 10NN in analiza na osnovi popolnih enot (kar je zelo presenetljivo,

saj smo pričakovali, da se pri NMAR analiza na osnovi popolnih enot izkaže izredno slabo). Na tej podatkovni bazi se najslabše izkaže RVI, ki mu sledi vstavljanje povprečij.

Vrstni red metod glede na RMS za lastne vrednosti prvih dveh oz. treh glavnih komponent pri 60 % enot z manjkajočimi podatki (glej tabelo 6.6) se za tri podatkovne baze nekoliko razlikuje, vendar pa se še zmeraj najbolje izkažeta metodi EM algoritem in MICE-PMM, najslabše pa metoda RVI, ki ji sledi vstavljanje povprečij.

Tabela 6.6: Mera različnosti RMS za lastne vrednosti prvih dveh oz. treh glavnih komponent pri 60 % enot z manjkajočimi podatki za NMAR

	Podatkovna baza		
	Mednarodna anketa	**Prepoznavnost vin**	**Nova vozila**
Najboljša	EM algoritem	EM algoritem	MICE-PMM
	10NN	MICE-PMM	EM algoritem
	MICE-PMM	pairwise	Listwise
	pairwise	10NN	10NN
	listwise	listwise	Pairwise
	vstavljanje povprečij	vstavljanje povprečij	vstavljanje povprečij
Najslabša	RVI	RVI	RVI

Preizkusili smo, kaj se z rezultati algoritma KNN dogaja pri različnih k. Na bazi »Mednarodna anketa« se najbolje izkaže 5NN (ob izbiri različnih k se RMS ne spreminja veliko, nekoliko slabše se izkaže le 1NN), na bazah »Prepoznavnost vin« in »Nova vozila« pa se najbolje izkaže 1NN (nižji k pomeni tudi nižji RMS). Umestitev metode KNN glede na ostale metode bi ob uporabi optimalnega k na podatkovni bazi »Mednarodna anketa« ostala približno enaka, na podatkovni bazi »Prepoznavnost vin« bi se malenkost izboljšala in približala analizi na osnovi razpoložljivih podatkov, na podatkovni bazi »Nova vozila« pa bi se metoda KNN ob uporabi optimalnega k izkazala celo bolje tako od metode MICE-PMM kot tudi EM algoritma. Rezultati za metodo KNN ob izbiri različnih k so prikazani v tabelah B.21, B.24 in B.27 v prilogi B.

6.4 Časi izvajanja metod za obravnavo manjkajočih podatkov

V spodnji tabeli (6.7) so prikazani časi izvajanj za posamezne metode na stacionarnem računalniku s procesorjem Intel(R) Core(TM) i3.2100 CPU 3,1 GHz (4,00 GB RAM) ločeno za vse tri podatkovne baze.

Časi so izračunani ločeno za posamezne metode, mehanizme za nastanek manjkajočih podatkov in podatkovne baze. V prvem stolpcu vidimo imena načinov obravnave manjkajočih podatkov, katerih čas izvajanja se je meril, v drugem stolpcu uporabljeno funkcijo v programskem jeziku R, v četrtem, petem in šestem stolpcu pa čas izvajanja metode (za posamezne podatkovne baze).

Tabela 6.7: Časi izvajanja posameznih metod za 1000 ponovitev[10]

Metoda obravnave	**R funkcija**		Časi (ure:minute:sekunde) Mednarodna anketa	Prepoznavnost vin	Nova vozila
listwise	na.omit	MCAR	0:00:02	0:00:01	0:00:02
		MAR	0:00:02	0:00:01	0:00:02
		NMAR	0:00:02	0:00:01	0:00:02
pairwise	cov (pairwise.complete.obs)	MCAR	0:00:05	0:00:03	0:00:04
		MAR	0:00:05	0:00:03	0:00:04
		NMAR	0:00:05	0:00:03	0:00:04
vstavljanje povprečij	colMeans + ...	MCAR	0:00:01	0:00:01	0:00:01
		MAR	0:00:01	0:00:01	0:00:01
		NMAR	0:00:01	0:00:01	0:00:01
RVI	impute(, "random") + ...	MCAR	0:00:34	0:00:25	0:00:31
		MAR	0:00:31	0:00:23	0:00:29
		NMAR	0:00:34	0:00:24	0:00:32
10NN	SeqKNN	MCAR	0:07:37	0:03:09	0:09:25
		MAR	0:07:38	0:03:10	0:09:00
		NMAR	0:07:37	0:03:09	0:09:01
MICE-PMM	mice	MCAR	1:56:49	6:23:42	8:53:29
		MAR	1:51:48	6:00:09	16:55:02
		NMAR	1:56:17	6:22:36	17:37:36
EM algoritem	em.norm + ...	MCAR	0:00:41	0:00:39	0:00:39
		MAR	0:00:53	0:00:43	0:00:42
		NMAR	0:00:41	0:00:38	0:00:40

Najbolj zanimivi rezultati so za metodo MICE-PMM, ki je pričakovano daleč najpočasnejša (čeprav smo izdelovali le po eno popolno podatkovno matriko), in EM algoritem, ki je glede

[10] Za opis načina generiranja matrik z manjkajočimi podatki glej poglavje 5 Načrt raziskave.

na svojo učinkovitost izredno hiter. Te meritve nam dajejo uporaben vpogled v pogosto zanemarjen aspekt vstavljanja; pri velikih problemih časi izvajanja namreč niso več zanemarljivi.

Gornji vrstni red in same vrednosti so zgolj informativni, kajti hitro se lahko v programskem paketu R, ki se vztrajno in konstantno razvija, znajde nova, hitrejša in zoptimizirana verzija algoritma.

6.5 Povzetek rezultatov simulacijske raziskave

Na splošno lahko ob pogledu na rezultate za lastne vrednosti prvih dveh oz. prvih treh glavnih komponent v tem poglavju zaključimo, da ko podatki manjkajo po mehanizmu MCAR, sta za izračun uteži najprimernejši metodi EM algoritem in KNN, ki jima sledi metoda MICE-PMM, lastne vrednosti pa najnatančneje izračunamo z uporabo EM algoritma in MICE-PMM, vendar lahko tudi KNN, analiza na osnovi razpoložljivih podatkov in delno analiza na osnovi popolnih enot (le pri nižjih odstotkih enot z manjkajočimi podatki) privedejo do zadovoljivih rezultatov.

Pri mehanizmu za nastanek manjkajočih podatkov MAR sta za izračun uteži enako kot pri mehanizmu MCAR najprimernejši metodi EM algoritem in KNN, ki jima sledi metoda MICE-PMM, za izračun lastnih vrednosti pa se je najbolje izkazala metoda MICE-PMM, ki ji sledita EM algoritem in KNN.

Ko podatki manjkajo po mehanizmu NMAR, se za izračun uteži najbolje izkaže EM algoritem, ki mu sledita metodi MICE-PMM in KNN. EM algoritem se je najbolje izkazal tudi za izračun lastnih vrednosti, malenkost slabše se je izkazala metoda MICE-PMM (metoda KNN se je za izračun lastnih vrednosti pri mehanizmu NMAR izkazala nekoliko slabše, vendar samo v primeru, ko smo uporabili k = 10 in ne optimalnega k).

Metoda KNN se je na vseh treh podatkovnih bazah pri vseh treh mehanizmih za nastanek manjkajočih podatkov tako za izračun uteži kot tudi lastnih vrednosti izkazala dobro (razen

za izračun lastnih vrednosti pri mehanizmu NMAR), včasih celo bolje od modernih pristopov (EM algoritma in metode MICE-PMM).

Ostale štiri metode (analiza na osnovi razpoložljivih podatkov, vstavljanje povprečij, analiza na osnovi popolnih enot in RVI) so se večinoma izkazale slabše od modernih pristopov in metode KNN in jih v nobenem primeru ne bi priporočili, če so nam na voljo najboljše tri metode.

7 SKLEP

V sklopu prvega raziskovalnega vprašanja se je ugotavljalo, ali so za izvedbo metode glavnih komponent v primeru manjkajočih podatkov moderni pristopi za obravnavo manjkajočih podatkov (metoda največjega verjetja in večkratno vstavljanje) primernejši od klasičnih.

Na podlagi dobljenih rezultatov in njihove analize lahko na naše prvo raziskovalno vprašanje odgovorimo pritrdilno. Večinoma sta se pri vseh treh mehanizmih za nastanek manjkajočih podatkov tako za izračun uteži kot lastnih vrednosti najbolje izkazala EM algoritem in MICE-PMM. Razlika med modernimi in klasičnimi pristopi je večja pri višjih odstotkih enot z manjkajočimi podatki in pri NMAR. Treba pa je poudariti, da se je kot zelo učinkovita izkazala tudi metoda KNN, ki jo uvrščamo h klasičnim pristopom k obravnavi manjkajočih podatkov.

V sklopu drugega raziskovalnega vprašanja se je ugotavljalo, v kolikšni meri je primernost različnih metod odvisna od deleža manjkajočih podatkov ter katere metode so primernejše glede na različne deleže manjkajočih podatkov in glede na različne mehanizme za nastanek manjkajočih podatkov (MCAR, MAR, NMAR).

Ko podatki manjkajo po mehanizmu MCAR, so za izvedbo metode glavnih komponent moderni pristopi primernejši od klasičnih, vendar pa je tudi metoda KNN zelo učinkovita. Ko podatki manjkajo po mehanizmu MAR, so enako kot pri MCAR modernejši pristopi večinoma primernejši od klasičnih, ponovno se izredno dobro izkaže tudi metoda KNN. Ko podatki manjkajo po mehanizmu NMAR, se klasični pristopi izkažejo slabše kot pri MCAR in MAR; kot najboljši metodi sta se ponovno izkazali EM algoritem in MICE-PMM (metoda KNN se je pri mehanizmu NMAR izkazala nekoliko slabše od modernih pristopov).

Analiza na osnovi popolnih enot, ki je najpogosteje uporabljena kot osnovna v večini statističnih programov, se je v večini primerov tako za izračun uteži kot tudi lastnih vrednosti izkazala bistveno slabše od večine obravnavanih metod; še posebno v primeru podatkovnih baz z veliko manjkajočimi podatki je uporaba analize na osnovi popolnih enot nedopustna. Enako velja za metodo vstavljanje naključnih vrednosti, ki se je izkazala še slabše od analize

na osnovi popolnih enot. Predpostavljali smo, da bo metoda razmeroma uspešna le pri nizkem deležu manjkajočih podatkov, saj ne upošteva multivariatnega vidika vstavljanja, vendar pa se je že pri nizkem deležu manjkajočih podatkov izkazala precej slabše od ostalih (tako klasičnih kot tudi modernih) metod pri vseh mehanizmih za nastanek manjkajočih podatkov.

EM algoritem se po pričakovanjih iz literature odreže izredno dobro in bi ga nedvomno priporočali. V primerjavi z metodama MICE-PMM in KNN, ki sta se poleg EM algoritma izkazali kot zelo učinkoviti, je EM algoritem tudi izredno hiter.

Če bi uporabnik želel podatke vstavljati s trivialnejšo in zares izjemno hitro metodo, bi mu priporočali analizo na osnovi razpoložljivih podatkov, ki je, kljub kritikam v literaturi, ponekod izjemno blizu najboljšim, vendar je treba upoštevati, da so njeni rezultati pri različnih mehanizmih za nastanek manjkajočih podatkov in pri različnih odstotkih enot z manjkajočimi podatki zelo nekonsistentni in se ji je zato bolje izogibati (še posebno, če sta nam na voljo metodi, kot sta EM algoritem in MICE-PMM).

Metoda KNN se je glede na ostale klasične metode izkazala daleč najboljše (za izračun lastnih vrednosti nekoliko slabše kot za izračun uteži). Če bi vnaprej poznali idealen k, bi se po učinkovitosti zelo približala EM algoritmu in metodi MICE-PMM. V praksi v primeru manjkajočih podatkov optimalnega k žal ne poznamo, saj nam podatkovna baza brez manjkajočih podatkov za primerjavo ni na voljo, zato je to vsekakor slabost te metode.

Ob teh opažanjih velja opozoriti, da je bila simulacija narejena samo za splošno shemo manjkajočih podatkov na treh različnih empiričnih podatkovnih bazah. Kaj se z rezultati dogaja ob večjih podatkovnih bazah (več enot, več spremenljivk), ob višjih deležih manjkajočih podatkov ter kako so časi izvajanj odvisni od teh velikosti, bi lahko bil predmet poglobljene analize. Vsekakor bi bilo zanimivo preizkusiti še kakšno dodatno metodo za obravnavo manjkajočih podatkov, kot je npr. EM-SKNN algoritem (Kim in drugi 2004). V nalogi smo seveda obravnavali samo metodo glavnih komponent, tako da naši zaključki za ostale statistične metode ne veljajo.

Problem nepopolnih podatkov se raziskuje že skoraj stoletje, vendar sta se največji odkritji zgodili v sedemdestih letih prejšnjega stoletja: ocenjevanje parametrov po metodi največjega verjetja in večkratno vstavljanje. V tem času je Rubin (1976) orisal teoretični okvir problema

nepopolnih podatkov, ki je še danes aktualen. Metoda največjega verjetja in večkratno vstavljanje sta v zadnjih 30 letih v metodološki literaturi prejela veliko pozornosti, raziskovalci ju obravnavjo kot najsodobnejša pristopa k obravnavi mankajočih podatkov. Glede na klasične pristope sta teoretsko privlačnejša, saj zahtevata šibkejše predpostavke glede mehanizma za nastanek manjkajočih podatkov. S praktičnega vidika to pomeni, da ta dva pristopa dajeta manj pristranske ocene statističnih parametrov (Enders 2010).

Raziskovalci žal relativno počasi sprejemajo tako metodo največjega verjetja kot tudi večkratno vstavljanje, večinoma se še zmeraj zanašajo na klasične pristope k obravnavi manjkajočih podatkov. Vzrok za to lahko delno pripišemo pomanjkanju programskih možnosti, saj sta obe metodi postali široko dostopni šele v poznih devetdesetih letih prejšnjega stoletja, tehnična narava literature na področju obravnave manjkajočih podatkov pa najverjetneje predstavlja drugo bistveno oviro pri njunem sprejemanju in uporabi (prav tam).

V prihodnosti bo na področju obravnave manjkajočih podatkov treba poskrbeti, da bo razkorak med pristopi, ki so vgrajeni v standardne statistične pakete ter pristopi, ki se pojavljajo v objavljenih znanstvenih člankih, vse manjši. Vsekakor bo veliko treba narediti na tem, da se nova spoznanja na tem področju poskuša približati uporabnikom statističnih metod, predvsem kar se tiče modernih pristopov k obravnavi manjkajočih podatkov, saj bi bila res velika škoda, če znanja in programskih možnosti, ki so nam trenutno na voljo, ne bi poskušali izkoristili v celoti.

8 LITERATURA

Abdi, Herve, 2007. RV coefficient and congruence coefficient. V *Encyclopedia of Measurement and Statistics*, ur. Neil J. Salkind, 849–853. Thousand Oaks (CA): Sage.

Abdi, Herve, 2010. Congruence: Congruence coefficient, RV coefficient, and Mantel Coefficient. V *Encyclopedia of Research Design*, ur. Neil J. Salkind, 222–229. Thousand Oaks (CA): Sage.

Acuna, Edgar in Caroline Rodriguez. 2004. The treatment of missing values and its effect in the classifier accuracy V *Classification, Clustering, and Data Mining Applications*, ur. David Banks, Leanna House, Frederick R. McMorris, Phipps Arabie, Wolfgang Gaul, 639–648. Berlin-Heidelberg: Springer-Verlag.

Allison D., Paul. 2000. Multiple Imputation for Missing Data: A Cautionary Tale. *Sociological Methods and Research* 28 (3): 301–309.

American Statistical Association. *2004 New Car and Truck Data*. Dostopno prek: http://www.amstat.org/publications/jse/datasets/04cars.txt (6. april 2012).

Andridge, Rebecca R. in Roderick J.A. Little. 2010. A Review of Hot Deck Imputation for Survey Non-response. *Int Stat Rev.* 78 (1): 40–64.

Azen, S in Van Guilder, M. (1981). Conclusions regarding algorithms for handling incomplete data. *Proceedings of the Statistical Computing Section, American Statistical Association*: 53–56.

Azur, J. Melissa, Elizabeth A. Stuart, Constantine Frangakis, Philip J. Leaf . 2011. Multiple Imputation by Chained Equations: What is it and how does it work?. *International Journal of Methods in Psychiatric Research* 20 (1): 40–49.

Barret, Paul. 1986. Factor comparison: an examination of three methods. *Personality and Individual Differences* 7 (3): 327–440.

Bedeian, Arthur G., Achilles A. in Armenakis Alan W. Randolph. 1988. The Significance of Congruence Coefficients: A Comment and Statistical Test. *Journal of Management* 14 (4): 559–566.

Blejec, Andrej. 2008. *Multivariatne metode in manjkajoči podatki*. Dostopno prek: http://ablejec.nib.si/R/mva07.pdf (7. marec 2011).

Denk, Michaela, Michael Weber. 2011. *Avoid Filling Swiss Cheese with Whipped Cream: Imputation Techniques and Evaluation Procedures for Cross-Country Time Series*. Dostopno prek: http://www.imf.org/external/pubs/ft/wp/2011/wp11151.pdf (7. avgust 2011).

Enders**,** Craig Kyle. 2010. *Applied Missing Data Analysis*. New York: The Guilford Press.

Ferligoj, Anuška. 1989. *Razvrščanje v skupine. Teorija in uporaba v družboslovju.* Ljubljana: Fakulteta za sociologijo, politične vede in novinarstvo.

--- *Metoda glavnih komponent*. Dostopno prek: http://vlado.fmf.uni-lj.si/vlado/podstat/Mva.htm (17. marec 2011).

Frieze, Hanson Irene. *Cross – Cultural Research*. Dostopno prek: http://www.pitt.edu/~frieze/ccresrch.htm (19. februar 2011).

Garson, David. *Factor Analysis*. Dostopno prek: http://www2.chass.ncsu.edu/garson/pa765/factor.htm (19. februar 2011).

Gelman, Andrew in Jennifer Hill (2007): *Data Analysis Using Regression and Multilevel/Hierarchical Models.* New York: Cambridge University Press.

Graham, W. John. 2003. Adding Missing-Data-Relevant Variables to FIML-Based Structural Equation Models. *Structural Equation Modeling*. 10(1): 80–100.

Graham, W. John, Alison E. Olchowsky, Tamika D. Gilreath. 2007. How many imputations are really needed? Some practical clarifications of multiple imputation theory. Prevention science: the official journal of the Society for Prevention Research. 8(3); 206–213.

Graham, W. John. 2009. Missing Data Analysis: Making it Work in the Real World. *Annual Review of Psychology*. 60: 549–576.

Graham, W. John. 2012. *Missing Data: Analysis and Design (Statistics for Social and Behavioral Sciences)*. New York: Springer.

Grung, Bjorn in Rolf Manne. 1998. Missing values in principal component analysis. *Chemometrics and Intelligent Laboratory Systems* 42(1): 125–139.

Haitovsky, Yoel. 1968. Missing data in regression analysis. *Journal of the Royal Statistical Society*. B 30(1): 67–82.

Howell, David. 2007. The Treatment of Missing Data V *The SAGE Handbook of Social Science Methodology*, ur. William Outhwaite, Stephen P. Turner, 208–224. London: Sage.

Ilin, Alexander in Tapani Raiko. 2010. Practical Approaches to Principal Component Analysis in the Presence of Missing Values. *Journal of Machine Learning Research* 11: 1957–2000.

Jesenko, Jože in Manca Jesenko. 2007. *Multivariatne statistične metode*. Kranj: Založba Moderna organizacija.

Jolliffe, Ian. 2002. *Principal Component Analysis*. New York: Springer-Verlag.

Kalton, Graham in Vasja Vehovar. 2001. *Vzorčenje v anketah*. Ljubljana: Fakulteta za družbene vede, Univerza v Ljubljani.

Kim, KI-Yeol, Kim Byoung-Jin in Yi Gwan Su. 2004. Reuse of imputed data in microarray analysis increases imputation efficiency. *BMC Bioinformatics* 5:160.

Kim, Jae-On in James Curry. 1977. The treatment of missing data in multivariate analysis. *Sociological Methods Research* 6(2): 215–240.

Košmelj, Katarina. 2007. Metoda glavnih komponent: osnove in primer. *Acta agriculturae Slovenica* 89 (1): 159–172.

Levine, Mark S. 1977. Canonical Analysis and Factor Comparison. London: Sage Publications.

Little, J. A. Roderick in Donald B. Rubin. 2002. *Statistical Analysis with Missing Data*. New York: Wiley.

Lorenzo-Seva, Urbano in Jos M. F. ten Berge. 2006. Tucker's Congruence Coefficient as a Meaningful Index of Factor Similarity. *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences* 2 (2): 57–64.

Malešič, Kaja, Jože Rovan in Lea Bregar. *Ocena kakovosti sestavljenih kazalcev blaginje na podlagi glavnih komponent*. Dostopno prek http://www.stat.si/StatisticniDnevi/Docs/Radenci%202010/Malesic_Rovan_Bregar_SKB-prispevek.pdf (7. avgust 2011).

Marshall, A., Altman D. G., Royston P. in Holder R. L. 2010. Comparison of techniques for handling missing covariate data within prognostic modelling studies: a simulation study. *BMC Research Methodology* 10 (7): 1:16.

Palfy, Miroslav. 2009. *Nadomeščanje manjkajočih vrednosti s pomočjo rotacijskega regresijskega gozda. Doktorska disertacija*. Dostopno prek: http://dkum.uni-mb.si/Dokument.php?id=12468 (7. avgust 2011).

Radhakrishna, C. Rao. 1964. The use and interpretation of Principal Component Analysis in applied research. *Sankhya* A 26 (4): 329–358.

Raiko, Tapani, Alexander Ilin, in Juha Karhunen. 2007. Principal Component Analysis for Large Scale Problems with Lots of Missing Values V *Proceedings of the 18th European conference on Machine Learning*, ur. Joost N. Kok, Jacek Koronacki, Raomon Lopez Mantaras, Stan Matwin, Dunja Mladenič, Andrzej Skowron, 691–698, Berlin-Heidelberg: Springer-Verlag.

Rebonato, Riccardo in Peter Jackel. 2000. *The most general methodology to create a valid correlation matrix for risk management and option pricing purposes*. Dostopno prek: http://www.riccardorebonato.co.uk/papers/ValCorMat.pdf (7. junij 2011).

Rubin, Donald B. 1976: Inference and Missing Data. *Biometrika* 63 (3): 581–592.

Schaefer, Juliane, Rainer Opgen-Rhein, Verena Zuber, A. Pedro Duarte Silva, Miika Ahdesmäki in Korbinian Strimmer. 2012. *Package 'corpcor'*. Dostopno prek: http://cran.r-project.org/web/packages/corpcor/corpcor.pdf (6. junij 2012).

Schafer, L. Joseph. 1997. *Analysis of Incomplete Multivariate Data*. London: Chapman & Hall.

Schafer, L. Joseph in John W. Graham. 2002. Missing data: Our View of the State of The Art. *Psychological Methods* 7 (2): 147–177.

Schafer L. Joseph in Maren K. Olsen. 1998. Multiple imputation for multivariate missing-data. problems: a data analyst's perspective. *Multivariate Behavioral Research* 33 (4): 545–571.

Schenker, Nathaniel in Jeremy M. G. Taylor. 1996. Partially parametric techniques for multiple imputation. *Computational Statistics and Data Analysis*. 22: 425–446.

Shalizi, Cosma. 2009. *Principal Components: Mathematics, Example, Interpretation*. Dostopno prek: http://www.stat.cmu.edu/~cshalizi/350/lectures/10/lecture-10.pdf (13. september 2011).

Stanimirova, Ivana, Michal Daszykowski in Beata Walczak. 2006. Dealing with missing values and outliers in principal component analysis. *Talanta* 72 (1): 172: 178.

Teel, Cynthia in Joyce A. Verran. 1991. Focus on psychometrics. Factor comparison across studies. *Research in Nursing and Health* 14 (1): 67–72.

The Pennsylvania State University, Department of Statistics. *The multiple imputation FAQ page*. Dostopno prek: http://www.stat.psu.edu/~jls/mifaq.html (17. avgust 2011).

UCI Machine Learning Repository. 1991. *Wine Data Set*. Dostopno prek: http://archive.ics.uci.edu/ml/datasets/Wine (13. september 2011).

van Buuren, Stef in Karin Groothuis-Oudshoorn. 2011. Mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software* 45 (3): 1–67.

van Buuren, Stef. 2012. *Fexible imputation of missing data*. Boca Raton: Chapman & Hall/CRC.

Vehovar, Vasja (2007): *Nepopolni podatki v anketah*. Ljubljana: interno gradivo.

Wasito, Ito in Boris Mirkin. 2005. Nearest neighbour approach in the least-squares data imputation algorithms. *Information Sciences – ISCI* 169 (1-2): 1–25.

PRILOGA A: OPIS PODATKOVNIH BAZ

1. podatkovna baza: Mednarodna anketa o stališčih in izkušnjah študentov

Anketiranje se je izvajalo na podlagi mednarodne ankete o stališčih in izkušnjah študentov. Vprašalnik, ki sta ga pripravili Irene Frieze (ZDA) in Anuška Ferligoj (Slovenija), se je med letoma 1991 in 2004 uporabljalo na več univerzah v Evropi (Albanija, Bolgarija, Češka, Hrvaška, Litva, Madžarska, Nemčija, Norveška, Poljska, Rusija, Slovaška), Indiji, Pakistanu, na Japonskem in v ZDA (Frieze, 2011). Obravnavana populacija so študentje Fakultete za družbene vede v Ljubljani v letu 2002, in sicer so praviloma to študentje drugega letnika. Priložnostni vzorec pa so študentje, ki so bili v času izvajanja ankete prisotni na predavanjih, torej aktivni študenti. Takih študentov je bilo 328 (n = 328).

Tabela A.1: Opisne statistike spremenljivk

Spremenljivke	n	Povprečje	St. odklon	Asimetrija	Sploščenost
famc1	328	3,75	1,021	-0,686	0,101
famc2	328	2,93	1,020	0,113	-0,454
famc3	328	3,34	1,028	-0,298	-0,271
famc4	328	2,86	0,925	0,072	-0,116
famc5	328	3,24	0,966	-0,124	-0,237
famc6	328	3,98	0,793	-0,696	0,764
wrkc1	328	2,51	0,794	0,366	0,674
wrkc2	328	3,46	0,819	-0,484	0,213
wrkc3	328	3,18	0,794	-0,259	0,280
wrkc4	328	2,45	0,815	0,280	0,419
wrkc5	328	2,44	0,787	0,373	0,661
wrkc6	328	2,09	0,816	0,488	0,525

Tabela A.2: Korelacijska matrika

	famc1	famc2	famc3	famc4	famc5	famc6	wrkc1	wrkc2	wrkc3	wrkc4	wrkc5	wrkc6
famc1	1											
famc2	,646**	1										
famc3	,611**	,671**	1									
famc4	,54**	,67**	,713**	1								
famc5	,555**	,557**	,669**	,636**	1							
famc6	,517**	,421**	,49**	,396**	,49**	1						
wrkc1	-,1	,016	-,013	,048	,08	-,116*	1					
wrkc2	-,029	-,114*	-,063	-,145**	,021	,149**	,34**	1				
wrkc3	,074	-,003	,026	-,016	,076	,051	,324**	,536**	1			
wrkc4	-,136*	,003	-,04	-,083	-,073	-,135*	,502**	,267**	,334**	1		
wrkc5	-,203*	-,082	-,14*	-,129*	-,138*	-,154**	,466**	,318**	,401**	,6**	1	
wrkc6	-,118*	-,011	-,086	0	-,068	-,233**	,485**	,152**	,255**	,605**	,665**	1

(**) povezanost je stat. značilna pri stopnji tveganja $\alpha = 0,01$
(*) povezanost je stat. značilna pri stopnji tveganja $\alpha = 0,05$

2. podatkovna baza: Prepoznavnost vin

Podatki so rezultat kemijske analize vin treh različnih kultivarjev iz iste italijanske regije. Z analizo se je ugotavljalo prisotnost 13 različnih sestavin (spremenljivk) v 178 vzorcih vina (n = 178). Podatkovna baza ne vsebuje manjkajočih podatkov. Vse spremenljivke so intervalne (številske).

Tabela A.3: Opisne statistike spremenljivk

Spremenljivke	n	Povprečje	St. odklon	Asimetrija	Sploščenost
s1	178	13,00	0,812	-0,051	-0,852
s2	178	2,34	1,117	1,040	0,299
s3	178	2,37	0,274	-0,177	1,144
s4	178	19,50	3,340	0,213	0,488
s5	178	99,74	14,282	1,098	2,105
s6	178	2,30	0,626	0,087	-0,836
s7	178	2,03	0,999	0,025	-0,880
s8	178	0,36	0,124	0,450	-0,637
s9	178	1,59	0,572	0,517	0,555
s10	178	5,06	2,318	0,869	0,382
s11	178	0,96	0,229	0,021	-0,344
s12	178	2,61	0,710	-0,307	-1,086
s13	178	746,89	314,907	0,768	-0,248

Tabela A.4: Korelacijska matrika

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13
s1	1,000												
s2	,094	1,000											
s3	,212**	,164*	1,000										
s4	-,310**	,289**	,443**	1,000									
s5	,271**	-,055	,287**	-,083	1,000								
s6	,289**	-,335**	,129	-,321**	,214**	1,000							
s7	,237**	-,411**	,115	-,351**	,196**	,865**	1,000						
s8	-,156*	,293**	,186*	,362**	-,256**	-,450**	-,538**	1,000					
s9	,137	-,221**	,010	-,197**	,236**	,612**	,653**	-,366**	1,000				
s10	,546**	,249**	,259**	,019	,200**	-,055	-,172*	,139	-,025	1,000			
s11	-,072	-,561**	-,075	-,274**	,055	,434**	,543**	-,263**	,296**	-,522**	1,000		
s12	,072	-,369**	,004	-,277**	,066	,700**	,787**	-,503**	,519**	-,429**	,565**	1,000	
s13	,644**	-,192	,224**	-,441**	,393**	,498**	,494**	-,311**	,330**	,316**	,236**	,313**	1,000

(**) povezanost je stat. značilna pri stopnji tveganja $\alpha = 0,01$
(*) povezanost je stat. značilna pri stopnji tveganja $\alpha = 0,05$

3. podatkovna baza: Nova vozila na ameriškem trgu leta 2004

Podatkovna baza zajema tehnične specifikacije in ceno novih vozil na ameriškem trgu leta 2004. Podatki so bili pridobljeni s strani spletne revije za finančno svetovanje Kiplinger, z njihovim dovoljenjem pa jih je objavila revija Journal of Statistics Education na svojem podatkovnem arhivu. Podatki se nanašajo na ceno vozil, mere vozil ter njihovo energetsko učinkovitost, uporabljeni pa so bili le podatki za tista vozila, pri katerih ni bilo manjkajočih podatkov (n = 387). Analiziranih je bilo 10 spremenljivk, ki predstavljajo tehnične specifikacije in ceno novih vozil. Vse spremenljivke so intervalne (številske).

Tabela A.5: Opisne statistike spremenljivk

Spremenljivke	n	Povprečje	St. odklon	Asimetrija	Sploščenost
retail	387	33231,18	19724,630	2,837	14,145
engine	387	3,13	1,014	0,440	-0,493
cylinders	387	5,76	1,490	0,507	0,268
horsepower	387	214,44	70,263	0,884	1,526
citympg	387	20,31	5,262	2,967	16,709
highwaympg	387	27,26	5,636	1,430	6,979
weight	387	3532,46	706,004	0,676	1,118
wheelbase	387	107,21	7,087	0,317	0,380
length	387	184,96	13,238	-0,165	0,213
width	387	71,28	3,368	0,574	-0,179

Tabela A.6: Korelacijska matrika

	retail	engine	cylinders	horsepower	citympg	highwaympg	weight	wheelbase	length	width
retail	1									
engine	,599**	1								
cylinders	,654**	,911**	1							
horsepower	,835**	,779**	,791**	1						
citympg	-,485**	-,705**	-,671**	-,672**	1					
highwaympg	-,469**	-,707**	-,664**	-,650**	,941**	1				
weight	,476**	,812**	,731**	,631**	-,736**	-,789**	1			
wheelbase	,203**	,631**	,553**	,397**	-,481**	-,455**	,751**	1		
length	,210**	,625**	,546**	,381**	-,468**	-,389**	,653**	,867**	1	
width	,313**	,726**	,619**	,499**	-,587**	-,583**	,807**	,757**	,751**	1

(**) povezanost je stat. značilna pri stopnji tveganja $\alpha = 0,01$
(*) povezanost je stat. značilna pri stopnji tveganja $\alpha = 0,05$

PRILOGA B: REZULTATI SIMULACIJSKE RAZISKAVE

REZULTATI ZA MEHANIZEM MCAR

Tabela B.1: Mera različnosti RMS za uteži na prvih dveh komponentah – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,023380	0,010757	0,010919	0,021767	0,006620	0,007487	0,008828
10	0,035516	0,015838	0,016220	0,027719	0,009563	0,010916	0,012916
15	0,052110	0,019533	0,020221	0,032618	0,011476	0,013173	0,015716
20	0,065219	0,022399	0,023549	0,038240	0,013524	0,015604	0,018679
25	0,084718	0,025829	0,027241	0,042124	0,015241	0,017558	0,020540
30	0,105910	0,027990	0,029951	0,049818	0,016950	0,019727	0,023403
35	0,116693	0,030437	0,032841	0,053025	0,018020	0,021290	0,024682
40	0,138828	0,032544	0,035497	0,056792	0,019694	0,023110	0,026533
45	0,165987	0,035504	0,039204	0,066013	0,021007	0,024775	0,028458
50	0,173931	0,037550	0,041779	0,068887	0,022185	0,026509	0,031218
55	0,201735	0,038977	0,044695	0,075884	0,023654	0,027919	0,032113
60	0,225109	0,039969	0,045830	0,080795	0,023834	0,028326	0,032595

Tabela B.2: Mera podobnosti CC za uteži na prvih dveh komponentah za metodo KNN – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999813	0,999864	0,999874	0,999881	0,999877
10	0,999628	0,999731	0,999751	0,999758	0,999750
15	0,999497	0,999619	0,999651	0,999653	0,999634
20	0,999275	0,999468	0,999510	0,999522	0,999500
25	0,999058	0,999330	0,999382	0,999397	0,999374
30	0,998847	0,999159	0,999224	0,999244	0,999225
35	0,998679	0,999056	0,999127	0,999128	0,999090
40	0,998476	0,998859	0,998945	0,998979	0,998942
45	0,998167	0,998673	0,998785	0,998818	0,998775
50	0,997978	0,998521	0,998607	0,998650	0,998589
55	0,997692	0,998384	0,998487	0,998511	0,998440
60	0,997505	0,998285	0,998447	0,998466	0,998419

Tabela B.3: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za metodo KNN – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,015670	0,014025	0,013860	0,013793	0,013725
10	0,023036	0,020971	0,021148	0,021039	0,020735
15	0,029075	0,026766	0,027490	0,027497	0,026606
20	0,032676	0,030849	0,031849	0,031283	0,029642
25	0,038335	0,037034	0,038671	0,037556	0,035016
30	0,044619	0,041271	0,042914	0,041988	0,039179
35	0,048488	0,046571	0,049048	0,047956	0,043969
40	0,053853	0,051623	0,054011	0,052384	0,047667
45	0,056373	0,055802	0,058809	0,056401	0,050206
50	0,062566	0,059762	0,063648	0,060675	0,053379
55	0,067695	0,063349	0,067183	0,064533	0,056621
60	0,073465	0,067805	0,072075	0,068755	0,059780

Tabela B.4: Mera različnosti RMS za uteži na prvih treh komponentah – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,018686	0,010529	0,010672	0,038242	0,008683	0,008223	0,011294
10	0,027304	0,014927	0,015214	0,042166	0,012373	0,011943	0,016087
15	0,034419	0,018526	0,019035	0,046278	0,015351	0,014711	0,019962
20	0,040724	0,021537	0,022239	0,049691	0,018035	0,017346	0,023314
25	0,048413	0,024468	0,025429	0,054198	0,019958	0,019486	0,026285
30	0,054561	0,026854	0,028114	0,057705	0,022204	0,021983	0,029102
35	0,061699	0,029499	0,031052	0,061915	0,024313	0,023783	0,031439
40	0,070395	0,031643	0,033567	0,065837	0,026457	0,026198	0,034022
45	0,079043	0,034301	0,036513	0,071036	0,028290	0,028222	0,036472
50	0,093734	0,035879	0,038630	0,073939	0,029768	0,029682	0,038209
55	0,104927	0,037888	0,041074	0,077814	0,031599	0,031749	0,040264
60	0,122601	0,040461	0,043905	0,083349	0,033322	0,033768	0,043197

Tabela B.5: Mera podobnosti CC za uteži na prvih treh komponentah za metodo KNN – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999742	0,999815	0,999823	0,999827	0,999822
10	0,999497	0,999629	0,999646	0,999652	0,999646
15	0,999269	0,999462	0,999482	0,999489	0,999478
20	0,998983	0,999242	0,999283	0,999304	0,999293
25	0,998769	0,999079	0,999116	0,999132	0,999114
30	0,998393	0,998830	0,998894	0,998911	0,998895
35	0,998148	0,998640	0,998716	0,998745	0,998719
40	0,997799	0,998350	0,998433	0,998478	0,998448
45	0,997442	0,998134	0,998229	0,998256	0,998224
50	0,997128	0,997902	0,998019	0,998082	0,998054
55	0,996722	0,997591	0,997739	0,997814	0,997791
60	0,996424	0,997397	0,997517	0,997563	0,997541

Tabela B.6: Mera različnosti RMS za lastne vrednosti prvih treh glavnih komponent za metodo KNN – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,015685	0,014804	0,015217	0,015725	0,016162
10	0,022161	0,022490	0,023977	0,025468	0,026271
15	0,027526	0,030220	0,032609	0,035166	0,036526
20	0,033126	0,038338	0,041877	0,045380	0,047406
25	0,036860	0,045044	0,050143	0,054718	0,057419
30	0,041759	0,053558	0,059799	0,065716	0,068785
35	0,044431	0,058699	0,066125	0,072743	0,076439
40	0,049282	0,067389	0,076533	0,084574	0,088975
45	0,054530	0,075805	0,086227	0,095681	0,100573
50	0,057451	0,082148	0,094057	0,104476	0,109978
55	0,061441	0,091156	0,104215	0,115861	0,121605
60	0,065164	0,100476	0,116080	0,128687	0,134733

Tabela B.7: Mera različnosti RMS za uteži na prvih dveh komponentah – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,006617	0,004845	0,005431	0,029166	0,002558	0,002297	0,003076
10	0,009566	0,006966	0,008374	0,033286	0,003699	0,003350	0,004525
15	0,012442	0,009031	0,011396	0,037676	0,004799	0,004615	0,005729
20	0,014785	0,010485	0,013922	0,042169	0,005464	0,005445	0,006753
25	0,017140	0,012006	0,016598	0,046414	0,006439	0,006249	0,007755
30	0,019537	0,013334	0,019129	0,050586	0,007248	0,007091	0,008662
35	0,022237	0,014725	0,021929	0,055554	0,008143	0,007908	0,009595
40	0,024510	0,016115	0,024771	0,060693	0,008688	0,008809	0,010285
45	0,027518	0,017269	0,027622	0,065341	0,009327	0,009604	0,010995
50	0,030029	0,018481	0,030395	0,070334	0,009854	0,010166	0,011644
55	0,033356	0,019619	0,033193	0,074699	0,010629	0,011254	0,012595
60	0,037173	0,020855	0,036261	0,079907	0,011022	0,011830	0,013137

Tabela B.8: Mera podobnosti CC za uteži na prvih dveh komponentah za metodo KNN – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999985	0,999988	0,999989	0,999989	0,999989
10	0,999975	0,999980	0,999982	0,999982	0,999981
15	0,999958	0,999966	0,999968	0,999968	0,999965
20	0,999947	0,999957	0,999960	0,999960	0,999957
25	0,999932	0,999945	0,999949	0,999949	0,999945
30	0,999917	0,999932	0,999937	0,999937	0,999931
35	0,999902	0,999921	0,999926	0,999925	0,999918
40	0,999883	0,999906	0,999911	0,999909	0,999901
45	0,999864	0,999891	0,999897	0,999895	0,999886
50	0,999853	0,999884	0,999889	0,999887	0,999875
55	0,999828	0,999862	0,999867	0,999863	0,999850
60	0,999811	0,999851	0,999858	0,999853	0,999839

Tabela B.9: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za metodo KNN – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,006671	0,006501	0,006852	0,007336	0,008021
10	0,009918	0,010271	0,011143	0,012532	0,014045
15	0,013256	0,01442	0,016131	0,018552	0,021068
20	0,015994	0,018157	0,020690	0,023932	0,027235
25	0,017925	0,021613	0,025031	0,029511	0,033804
30	0,020971	0,026328	0,030793	0,036343	0,041753
35	0,022375	0,029700	0,035353	0,042152	0,048457
40	0,024668	0,033892	0,040347	0,048488	0,055789
45	0,026656	0,038216	0,046026	0,055317	0,063806
50	0,027870	0,041742	0,050406	0,060947	0,070108
55	0,031933	0,047425	0,056911	0,068417	0,078597
60	0,033071	0,052096	0,062956	0,076091	0,087476

REZULTATI ZA MEHANIZEM MAR

Tabela B.10: Mera različnosti RMS za uteži na prvih dveh komponentah – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,028757	0,014616	0,014966	0,025516	0,008186	0,009226	0,011503
10	0,048703	0,020980	0,021775	0,032957	0,012211	0,014126	0,016487
15	0,087607	0,027270	0,028578	0,040932	0,015063	0,018098	0,020042
20	0,117787	0,029929	0,032004	0,046447	0,017721	0,022118	0,023947
25	0,165208	0,033841	0,036665	0,053730	0,019926	0,024654	0,027441
30	0,191151	0,036649	0,040116	0,057563	0,022078	0,027044	0,029738
35	0,222917	0,040306	0,044474	0,066564	0,024439	0,029709	0,031641
40	0,257048	0,041001	0,046274	0,075315	0,025272	0,031236	0,034128
45	0,270149	0,045588	0,051450	0,079227	0,027117	0,032249	0,036712
50	0,294907	0,045707	0,053101	0,083805	0,027573	0,034168	0,039151
55	0,272551	0,047801	0,055970	0,093475	0,029724	0,035180	0,038890
60	0,288052	0,049989	0,058262	0,099027	0,031549	0,036596	0,041661

Tabela B.11: Mera podobnosti CC za uteži na prvih dveh komponentah za metodo KNN – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999734	0,999793	0,999807	0,999816	0,999814
10	0,999396	0,999530	0,999557	0,999572	0,999562
15	0,999033	0,999261	0,999294	0,999309	0,999311
20	0,998623	0,998938	0,998978	0,998981	0,998978
25	0,997353	0,998670	0,998748	0,998768	0,998766
30	0,997808	0,998379	0,998436	0,998493	0,998526
35	0,997285	0,998000	0,998129	0,998194	0,998225
40	0,994308	0,997875	0,997989	0,998043	0,998039
45	0,995809	0,997625	0,997807	0,997908	0,997911
50	0,991862	0,997338	0,997531	0,997663	0,997695
55	0,993440	0,997244	0,997440	0,997574	0,997578
60	0,991393	0,996068	0,996375	0,997357	0,997384

Tabela B.12: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za metodo KNN – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,018055	0,015724	0,015397	0,015137	0,015064
10	0,027646	0,023541	0,022948	0,022357	0,022209
15	0,038265	0,030374	0,029468	0,028838	0,028544
20	0,044825	0,034871	0,033979	0,033427	0,033647
25	0,054733	0,038951	0,037575	0,036778	0,038168
30	0,063485	0,042153	0,040496	0,039979	0,042495
35	0,072994	0,047632	0,045288	0,044674	0,048309
40	0,080650	0,048924	0,046512	0,045587	0,049776
45	0,088260	0,052132	0,048917	0,047601	0,052674
50	0,097329	0,054819	0,050754	0,049214	0,054989
55	0,105384	0,058075	0,054231	0,052584	0,059037
60	0,109979	0,059062	0,054187	0,051958	0,057726

Tabela B.13: Mera različnosti RMS za uteži na prvih treh komponentah – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,020619	0,011410	0,011580	0,040368	0,007449	0,007433	0,010766
10	0,033546	0,017928	0,018193	0,046352	0,011084	0,010823	0,015892
15	0,046039	0,024050	0,024413	0,052821	0,013744	0,013234	0,019680
20	0,060057	0,030207	0,030559	0,058632	0,016317	0,015760	0,023311
25	0,078316	0,036946	0,037237	0,065198	0,018623	0,017905	0,026318
30	0,093791	0,041957	0,042190	0,071132	0,020405	0,019484	0,029069
35	0,112916	0,047496	0,047685	0,076550	0,022546	0,021578	0,031509
40	0,135445	0,052611	0,052720	0,082148	0,024332	0,023401	0,034152
45	0,159650	0,058482	0,058377	0,088131	0,026513	0,025219	0,036862
50	0,180179	0,063380	0,063272	0,094148	0,028527	0,026977	0,039459
55	0,207950	0,067087	0,066783	0,098280	0,029777	0,028654	0,041731
60	0,240297	0,072204	0,071722	0,103155	0,032113	0,031012	0,043906

Tabela B.14: Mera podobnosti CC za uteži na prvih treh komponentah za metodo KNN – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999747	0,999839	0,999853	0,999866	0,999869
10	0,999485	0,999666	0,999699	0,999723	0,999727
15	0,999207	0,999507	0,999554	0,999593	0,999600
20	0,998926	0,999319	0,999381	0,999427	0,999431
25	0,998646	0,999125	0,999213	0,999270	0,999273
30	0,998424	0,998986	0,999077	0,999138	0,999141
35	0,998013	0,998746	0,998863	0,998950	0,998956
40	0,997684	0,998550	0,998699	0,998779	0,998778
45	0,997349	0,998328	0,998488	0,998584	0,998568
50	0,996985	0,998135	0,998303	0,998395	0,998352
55	0,996616	0,997869	0,998096	0,998205	0,998139
60	0,996162	0,997561	0,997784	0,997897	0,997773

Tabela B.15: Mera različnosti RMS za lastne vrednosti prvih treh glavnih komponent za metodo KNN – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,015883	0,013514	0,013240	0,013078	0,013097
10	0,023495	0,019792	0,019702	0,019922	0,020087
15	0,028641	0,025326	0,025893	0,026510	0,026821
20	0,033742	0,030524	0,031520	0,032664	0,033197
25	0,037789	0,036614	0,038455	0,040171	0,040769
30	0,042341	0,041396	0,044569	0,047260	0,048064
35	0,046564	0,046060	0,049906	0,053502	0,054553
40	0,051132	0,053810	0,058486	0,062818	0,063782
45	0,053916	0,059723	0,065802	0,070339	0,070910
50	0,057808	0,065592	0,072505	0,078446	0,078591
55	0,063272	0,072365	0,080246	0,086766	0,086534
60	0,065355	0,079742	0,088945	0,095940	0,094965

Tabela B.16: Mera različnosti RMS za uteži na prvih dveh komponentah – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,010274	0,008421	0,009388	0,031879	0,004058	0,004257	0,005395
10	0,016754	0,013078	0,015186	0,038619	0,006061	0,006273	0,007993
15	0,023112	0,016835	0,020232	0,045164	0,007617	0,007962	0,010056
20	0,030073	0,020459	0,025158	0,051695	0,009120	0,009746	0,012187
25	0,036252	0,023585	0,029590	0,057347	0,010414	0,011199	0,013655
30	0,043217	0,026311	0,033659	0,063509	0,011411	0,012428	0,015106
35	0,049101	0,028365	0,037100	0,068969	0,012435	0,013590	0,016565
40	0,056390	0,030880	0,040972	0,074453	0,013616	0,014851	0,017509
45	0,064668	0,032741	0,044237	0,079176	0,014269	0,015827	0,018604
50	0,072896	0,034271	0,047179	0,084772	0,015190	0,016523	0,019895
55	0,081221	0,036320	0,050543	0,089569	0,015861	0,018008	0,020656
60	0,092173	0,038011	0,053547	0,094653	0,016611	0,018920	0,021611

Tabela B.17: Mera podobnosti CC za uteži na prvih dveh komponentah za metodo KNN – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999966	0,999973	0,999974	0,999974	0,999972
10	0,999935	0,999948	0,999950	0,999948	0,999941
15	0,999900	0,999922	0,999923	0,999918	0,999906
20	0,999859	0,999888	0,999890	0,999881	0,999861
25	0,999820	0,999858	0,999859	0,999846	0,999820
30	0,999778	0,999828	0,999828	0,999812	0,999779
35	0,999748	0,999800	0,999802	0,999780	0,999743
40	0,999701	0,999759	0,999760	0,999737	0,999693
45	0,999659	0,999736	0,999734	0,999708	0,999658
50	0,999617	0,999710	0,999711	0,999683	0,999625
55	0,999553	0,999656	0,999661	0,999625	0,999562
60	0,999507	0,999623	0,999628	0,999589	0,999509

Tabela B.18: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za metodo KNN – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,013148	0,011586	0,011617	0,011946	0,012593
10	0,019484	0,017842	0,018195	0,018938	0,020163
15	0,023747	0,021312	0,022101	0,023795	0,025672
20	0,030005	0,026937	0,027944	0,030540	0,033443
25	0,034152	0,029887	0,032068	0,035818	0,039346
30	0,037436	0,032740	0,035473	0,040733	0,044807
35	0,041815	0,036586	0,039785	0,045333	0,049896
40	0,045971	0,040431	0,044462	0,050866	0,056396
45	0,049972	0,042833	0,048216	0,055945	0,061803
50	0,052987	0,044467	0,049648	0,058315	0,065410
55	0,058670	0,051046	0,056739	0,066044	0,073324
60	0,061830	0,051715	0,059516	0,070382	0,079484

REZULTATI ZA MEHANIZEM NMAR

Tabela B.19: Mera različnosti RMS za uteži na prvih dveh komponentah – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,036658	0,021644	0,021915	0,033990	0,011555	0,014904	0,013944
10	0,069930	0,032786	0,033382	0,052803	0,018249	0,023053	0,022838
15	0,124020	0,043420	0,044916	0,071713	0,022203	0,028335	0,027621
20	0,194630	0,059492	0,060262	0,104377	0,027434	0,035270	0,033108
25	0,250075	0,073216	0,075523	0,136746	0,031027	0,041221	0,038827
30	0,309904	0,095876	0,096615	0,167898	0,034993	0,046719	0,045470
35	0,380955	0,107182	0,109479	0,200451	0,037545	0,054622	0,051502
40	0,416027	0,137945	0,138479	0,240934	0,043311	0,062060	0,057780
45	0,458457	0,154017	0,156471	0,267356	0,044061	0,064706	0,058560
50	0,511466	0,174664	0,179338	0,308664	0,047051	0,078010	0,070498
55	0,549757	0,226085	0,227522	0,350290	0,054456	0,098577	0,081732
60	0,570747	0,260688	0,264896	0,394373	0,057219	0,112408	0,090575

Tabela B.20: Mera podobnosti CC za uteži na prvih dveh komponentah za metodo KNN – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999303	0,999452	0,999477	0,999443	0,999369
10	0,998460	0,998748	0,998805	0,998714	0,998575
15	0,997776	0,998234	0,998293	0,998177	0,997974
20	0,993996	0,996451	0,997390	0,997183	0,996866
25	0,989909	0,995683	0,995729	0,995461	0,994163
30	0,989654	0,994737	0,995715	0,994530	0,993964
35	0,975224	0,988142	0,989991	0,987882	0,985553
40	0,972645	0,988675	0,987216	0,984171	0,978348
45	0,965907	0,978546	0,982044	0,981531	0,977305
50	0,946570	0,969641	0,973970	0,969896	0,960512
55	0,922594	0,949048	0,951651	0,946726	0,936347
60	0,896691	0,937715	0,942774	0,933420	0,911091

Tabela B.21: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za metodo KNN – podatkovna baza »Mednarodna anketa«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,032096	0,027699	0,027010	0,027179	0,028327
10	0,051648	0,043620	0,042333	0,042737	0,044473
15	0,067428	0,055486	0,053262	0,053488	0,055964
20	0,083401	0,065526	0,062721	0,062930	0,066363
25	0,093750	0,069646	0,066276	0,066850	0,071847
30	0,107467	0,077084	0,072254	0,072948	0,079388
35	0,115803	0,079854	0,074647	0,075929	0,084077
40	0,124462	0,083852	0,078074	0,079816	0,089609
45	0,132876	0,087092	0,080694	0,083041	0,094899
50	0,145324	0,090203	0,083701	0,086773	0,100810
55	0,153401	0,092338	0,085283	0,089420	0,105467
60	0,165959	0,099027	0,091430	0,096280	0,114445

Tabela B.22: Mera različnosti RMS za uteži na prvih treh komponentah – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,020915	0,015377	0,015450	0,038743	0,012470	0,012226	0,014538
10	0,030909	0,022777	0,022853	0,044063	0,018580	0,018383	0,021601
15	0,039944	0,029609	0,029699	0,049407	0,023924	0,024005	0,027255
20	0,047870	0,035206	0,035336	0,054596	0,028245	0,028336	0,032074
25	0,056016	0,040435	0,040622	0,060448	0,031734	0,032279	0,036422
30	0,065896	0,045739	0,045997	0,066063	0,035704	0,036298	0,040077
35	0,075689	0,050261	0,050565	0,071035	0,039074	0,040076	0,044307
40	0,094094	0,055745	0,056100	0,077241	0,042239	0,043749	0,047982
45	0,122481	0,060192	0,060627	0,082560	0,044675	0,046246	0,050819
50	0,144454	0,064297	0,064694	0,087973	0,047326	0,049418	0,053808
55	0,184137	0,068781	0,069299	0,093705	0,050116	0,052384	0,057001
60	0,227331	0,072178	0,072784	0,098927	0,052403	0,054970	0,059677

Tabela B.23: Mera podobnosti CC za uteži na prvih treh komponentah za metodo KNN – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999583	0,999629	0,999628	0,999622	0,999607
10	0,999145	0,999221	0,999216	0,999205	0,999179
15	0,998590	0,998714	0,998708	0,998687	0,998644
20	0,998090	0,998256	0,998243	0,998221	0,998164
25	0,997530	0,997773	0,997782	0,997737	0,997658
30	0,996899	0,997204	0,997208	0,997174	0,997096
35	0,996233	0,996634	0,996642	0,996592	0,996499
40	0,995541	0,996009	0,996015	0,995986	0,995887
45	0,994989	0,995603	0,995630	0,995590	0,995459
50	0,994252	0,994993	0,995003	0,994957	0,994805
55	0,993544	0,994351	0,994428	0,994386	0,994221
60	0,992871	0,993802	0,993883	0,993841	0,993676

Tabela B.24: Mera različnosti RMS za lastne vrednosti prvih treh glavnih komponent za metodo KNN – podatkovna baza »Prepoznavnost vin«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,023728	0,025056	0,025787	0,026551	0,026984
10	0,037288	0,042486	0,044352	0,045919	0,046686
15	0,051220	0,060151	0,063104	0,065579	0,066557
20	0,062021	0,075645	0,079656	0,082733	0,083745
25	0,074287	0,092370	0,097280	0,100936	0,102377
30	0,085872	0,107238	0,113214	0,117890	0,119405
35	0,097293	0,124139	0,131371	0,136643	0,138307
40	0,106318	0,137678	0,145553	0,151833	0,153641
45	0,115381	0,151991	0,161347	0,168432	0,169983
50	0,124681	0,165318	0,175302	0,182743	0,184115
55	0,133586	0,179045	0,189872	0,198216	0,199083
60	0,139355	0,189881	0,201865	0,210866	0,211319

Tabela B.25: Mera različnosti RMS za uteži na prvih dveh komponentah – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	listwise	pairwise	vst. povprečij	RVI	EM algoritem	10NN	MICE-PMM
0	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,017556	0,018200	0,018298	0,031765	0,012019	0,011991	0,012705
10	0,027531	0,028447	0,028518	0,039474	0,018964	0,018571	0,020081
15	0,036442	0,037155	0,037358	0,048143	0,023906	0,023604	0,025354
20	0,044474	0,044675	0,045095	0,056361	0,027096	0,027311	0,029124
25	0,050994	0,050553	0,051283	0,064062	0,029988	0,030188	0,032303
30	0,057522	0,055768	0,056971	0,071645	0,032094	0,032567	0,034468
35	0,064610	0,061570	0,063005	0,078713	0,034402	0,035084	0,037108
40	0,070209	0,066184	0,068019	0,086004	0,036053	0,036992	0,039006
45	0,079567	0,071133	0,073328	0,092794	0,037463	0,039373	0,040793
50	0,087210	0,076165	0,078543	0,099374	0,039285	0,041550	0,042607
55	0,095586	0,080448	0,083135	0,104921	0,040696	0,043964	0,044100
60	0,105257	0,085759	0,088608	0,112088	0,042490	0,046863	0,045867

Tabela B.26: Mera podobnosti CC za uteži na prvih dveh komponentah za metodo KNN – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	1,000000	1,000000	1,000000	1,000000	1,000000
5	0,999823	0,999814	0,999806	0,999793	0,999771
10	0,999665	0,999634	0,999614	0,999580	0,999533
15	0,999491	0,999440	0,999411	0,999352	0,999278
20	0,999333	0,999273	0,999238	0,999153	0,999050
25	0,999187	0,999116	0,999078	0,998969	0,998836
30	0,999056	0,998989	0,998937	0,998805	0,998646
35	0,998912	0,998837	0,998774	0,998620	0,998425
40	0,998809	0,998724	0,998645	0,998465	0,998227
45	0,998651	0,998562	0,998472	0,998264	0,997984
50	0,998526	0,998421	0,998312	0,998067	0,997720
55	0,998377	0,998248	0,998128	0,997853	0,997440
60	0,998174	0,998028	0,997876	0,997551	0,997059

Tabela B.27: Mera različnosti RMS za lastni vrednosti prvih dveh glavnih komponent za metodo KNN – podatkovna baza »Nova vozila«

odstotek manjkajočih podatkov	1NN	3NN	5NN	10NN	20NN
0	0,000000	0,000000	0,000000	0,000000	0,000000
5	0,033057	0,036128	0,037484	0,038960	0,040899
10	0,050871	0,058875	0,061622	0,064738	0,067594
15	0,063579	0,075806	0,079356	0,083916	0,087582
20	0,073027	0,087490	0,091892	0,097949	0,101877
25	0,078445	0,095615	0,100831	0,107727	0,112145
30	0,082863	0,102389	0,108499	0,116271	0,121062
35	0,088736	0,110585	0,117947	0,126901	0,132506
40	0,090665	0,115686	0,124284	0,134385	0,140953
45	0,094412	0,122316	0,132495	0,143845	0,151061
50	0,098817	0,130846	0,142440	0,154936	0,163145
55	0,104284	0,140339	0,153088	0,166572	0,175487
60	0,107833	0,146851	0,161054	0,176286	0,186388

PRILOGA C: KODA V PROGRAMSKEM JEZIKU R

Primer kode v programskem jeziku R za podatkovno bazo »Mednarodna anketa«

```
################################################
### Brisanje podatkov MCAR:
################################################

GenerateMissing = function(A,p) {

if (p == 0) {pmv <- 0}
if (p != 5) {pmv <- p/5}

m<-dim(A)[2]
n<-dim(A)[1]
xMiss<-A
nme<-p/100*n
nmv<-pmv/100*n*m

tnme<-0
tnmv<-0

while(tnme<nme){
     #izbremo spremenjivko
     iSpr<-sample(1:m,size=1)
     #dovoljene enote (ki še nimajo manjkajoèe enote)
     tEnote<-which(!apply(is.na(xMiss),1,any))
     iEnota<-sample(tEnote,size=1) #izberemo enoto tako, da imajo velike
vrednosti veèjo verjetnost, da manjkajo
     xMiss[iEnota,iSpr]<-NA
     tnme<-tnme+1
     tnmv<-tnmv+1
}

while(tnmv<nmv){
     #izbremo spremenjivko
     iSpr<-sample(1:m,size=1)
     #dovoljene enote (ki že imajo manjkajoèo enoto, a ne pri tej
spremenljivki)
     tEnote<- which(apply(is.na(xMiss),1,any)&(!is.na(xMiss[,iSpr])))
     iEnota<-sample(tEnote,size=1)
     xMiss[iEnota,iSpr]<-NA
     tnmv<-tnmv+1
}

return(xMiss)
```

```
################################################
### Brisanje podatkov MAR:
################################################

GenerateMissingMAR = function(A,p) {

if (p == 0) {pmv <- 0}
if (p != 5) {pmv <- p/5}

m<-dim(A)[2]
n<-dim(A)[1]
xMiss<-A
nme<-p/100*n
nmv<-pmv*n*m/100

tnme<-0
tnmv<-0

m<-1:m

while(tnme<nme){
     #izbremo spremenjivko
        iSpr<-sample(m[m!=2],size=1)
     #dovoljene enote (ki še nimajo manjkajoèe enote)
     tEnote<-which(!apply(is.na(xMiss),1,any))
     iEnota<-sample(tEnote,prob=x[tEnote,2]^10,size=1)    #izberemo    enoto
tako, da imajo velike vrednosti veèjo verjetnost, da manjkajo
     xMiss[iEnota,iSpr]<-NA
     tnme<-tnme+1
     tnmv<-tnmv+1
}

while(tnmv<nmv){
     #izbremo spremenjivko
     iSpr<-sample(m[m!=2],size=1)
     #dovoljene  enote  (ki  že  imajo  manjkajoèo  enoto,  a  ne  pri  tej
spremenljivki)
     tEnote<- which(apply(is.na(xMiss),1,any)&(!is.na(xMiss[,iSpr])))
     iEnota<-sample(tEnote,prob=x[tEnote,2]^10,size=1)
     xMiss[iEnota,iSpr]<-NA
     tnmv<-tnmv+1
}

return(xMiss)

}
```

```
################################################
### Brisanje podatkov NMAR:
################################################

GenerateMissingNMAR = function(A,p) {

if (p == 0) {pmv <- 0}
if (p != 5) {pmv <- p/5}

m<-dim(A)[2]
n<-dim(A)[1]
xMiss<-A
nme<-p/100*n
nmv<-pmv*n*m/100

tnme<-0
tnmv<-0

while(tnme<nme){
     #izbremo spremenjivko
     iSpr<-sample(1:m,size=1)
     #dovoljene enote (ki še nimajo manjkajoèe enote)
     tEnote<-which(!apply(is.na(xMiss),1,any))
     iEnota<-sample(tEnote,prob=x[tEnote,iSpr]^10,size=1)  #izberemo  enoto
tako, da imajo velike vrednosti veèjo verjetnost, da manjkajo
     xMiss[iEnota,iSpr]<-NA
     tnme<-tnme+1
     tnmv<-tnmv+1
}

while(tnmv<nmv){
     #izbremo spremenjivko
     iSpr<-sample(1:m,size=1)
     #dovoljene  enote  (ki  že  imajo  manjkajoèo  enoto,  a  ne  pri  tej
spremenljivki)
     tEnote<- which(apply(is.na(xMiss),1,any)&(!is.na(xMiss[,iSpr])))
     iEnota<-sample(tEnote,prob=x[tEnote,iSpr]^10,size=1)
     xMiss[iEnota,iSpr]<-NA
     tnmv<-tnmv+1
}

return(xMiss)

}

GenerateMissingNMAR1 = function(A, p) {
N <- dim(A)[1]   ####number of cases
candidate<-which(A[,1]<3 | A[,2]<3 | A[,3]<3 | A[,4]<3 | A[,5]<3 | A[,6]<3
| A[,7]<3 | A[,8]<3 | A[,9]<3 | A[,10]<3 | A[,11]<3 | A[,12]<3)    #### I
want to sample all cases with at least 1 value lower than 3, so I have to
find candidates
idMiss <- sample(candidate, N * p / 100)    #### I sampled cases
for (i in idMiss){A[i,][which(A[i,] < 3)] <- NA}
return(A)
}
```

```
################################################
### Metode za obravnavo manjkajočih podatkov
################################################

# Vstavljanje aritmetične sredine

MeanImp = function(X)
{
    ids <- which(is.na(X), arr.ind = TRUE)
    means <- colMeans(X, na.rm = TRUE)
    X[ids] <- means[ids[,2]]
    return(X)
}

# Vstavljanje naključnih vrednosti (RVI)

RVI = function(X) {
   for (i in 1:dim(X)[2])
   {
      col = X[, i]
       X[, i] = as.integer(impute(col, "random"))
   }
   return(X)
}

# Metoda k najbližjih sosedov

KNNImp = function(X, k=1)
{
    return(ec.knnimp(X, k))
}


# MICE-PMM

MultipleImp = function(X)
{
   M <- mice(X, m=1, diagnostics=FALSE, printFlag=FALSE,maxit=10)
   return(complete(M))
}

# EM algoritem

EMImpNNN <- function(X)
{
    s <- prelim.norm(X)
    thetahat    <-    em.norm(s,showits=TRUE,    maxits    =    1000000,
criterion=0.000001)
    mu <- getparam.norm(s,thetahat,corr=TRUE)
    return(mu$r)
}
```

```
################################################
### Meri CC in RMS
################################################

# RMS za uteži, best = 0

WRMS = function(orig, weight)
{
   sqrt( sum( (orig - weight)^2 ) / length(orig) )
}


# CC za uteži

# CC za prvo utež, best = 1
measureCC1 = function(m1, m2)
{
   m = sqrt( sum(m1^2) * sum(m2^2) )
   return( sum(m1*m2) / m)
}


# CC za prvi dve uteži, best = 1
measureCC12 = function(w1, orig1, w2, orig2)
{
   m = sqrt( sum(w1^2) * sum(orig1^2) ) + sqrt( sum(w2^2) * sum(orig2^2) )
   return( ( sum(w1*orig1) + sum(w2*orig2) ) / m )
}

# CC za prve tri uteži, best = 1
measureCC13 = function(w1, orig1, w2, orig2, w3, orig3)
{
   m = sqrt( sum(w1^2) * sum(orig1^2) ) + sqrt( sum(w2^2) * sum(orig2^2)) +
sqrt( sum(w3^2) * sum(orig3^2))
   return( ( sum(w1*orig1) + sum(w2*orig2) + sum(w3*orig3)) / m )
}

# CC za vse uteži, best = 1
measureCC = function(w1, orig1, w2, orig2, w3, orig3, w4, orig4, w5, orig5,
w6, orig6, w7, orig7, w8, orig8, w9, orig9, w10, orig10, w11, orig11, w12,
orig12)
{
   m = sqrt( sum(w1^2) * sum(orig1^2) ) + sqrt( sum(w2^2) * sum(orig2^2) )
+ sqrt( sum(w3^2) * sum(orig3^2) ) + sqrt( sum(w4^2) * sum(orig4^2) ) +
sqrt( sum(w5^2) * sum(orig5^2) ) + sqrt( sum(w6^2) * sum(orig6^2) ) + sqrt(
sum(w7^2) * sum(orig7^2) ) + sqrt( sum(w8^2) * sum(orig8^2) ) + sqrt(
sum(w9^2) * sum(orig9^2) ) + sqrt( sum(w10^2) * sum(orig10^2) ) + sqrt(
sum(w11^2) * sum(orig11^2) ) + sqrt( sum(w12^2) * sum(orig12^2) )
   return( ( sum(w1*orig1) + sum(w2*orig2) + sum(w3*orig3) + sum(w4*orig4)
+  sum(w5*orig5)  +  sum(w6*orig6)  +  sum(w7*orig7)  +  sum(w8*orig8)  +
sum(w9*orig9) + sum(w10*orig10) + sum(w11*orig11) + sum(w12*orig12)) / m )
}
```

```
#################################################
### Vnos podatkov
#################################################

## (install packages mice, norm, dprep, Hmisc, (ade4), ...)

rm(list = ls(all = TRUE))

library(MASS)
library(nnet)
library(corpcor)

setwd("C:/Users/blaz/Desktop/R sim/prva baza")

source("functions.txt")

x = read.table("data.csv", header = TRUE, sep=";")
x = as.matrix(x[, 3:14])  # removal of demographic data

## parameters:

# odstotek manjkajočih podatkov
pmissing = seq(0, 60, by = 5)

# število ponovitev
rep = 1000


#################################################
### PCA: rezultati za matriko brez manjkajočih podatkov
#################################################
x<-scale(x, center = TRUE, scale = TRUE)+10
pc = princomp(x, cor = TRUE)
origload = (loadings(pc) %*% diag(pc$sdev))
origvar = pc$sdev^2

# screeplot and explained variances as in SPSS:
#   plot(1:12,   pc$sdev^2,   type="b",   xlab="Komponenta",   ylab="Lastna
vrednost")
# pc$sdev^2/12 * 100
```

```
################################################
### Generiranje matrik z manjkajočimi podatki
################################################

sez=list()
for (i in 1:length(pmissing))
{
   percent = pmissing[i]
sez2=list()
for (j in 1:rep)
{
sez2=c(sez2,list(
GenerateMissing(x,percent)))
}
sez=c(sez,list(sez2))
}
names(sez)<-pmissing


################################################
### Primer za izračun RMS (za uteži in lastne vrednosti) in CC (za uteži)
ob uporabi analize na osnovi popolnih enot
################################################

variance1 = vector(mode = "numeric", length = length(pmissing))
variance2 = vector(mode = "numeric", length = length(pmissing))
variance3 = vector(mode = "numeric", length = length(pmissing))

diffvar1 = vector(mode = "numeric", length = length(pmissing))
diffvar12 = vector(mode = "numeric", length = length(pmissing))
diffvar13 = vector(mode = "numeric", length = length(pmissing))
diffvar = vector(mode = "numeric", length = length(pmissing))
diff1 = vector(mode = "numeric", length = length(pmissing))
 diff12 = vector(mode = "numeric", length = length(pmissing))
diff13 = vector(mode = "numeric", length = length(pmissing))
diff = vector(mode = "numeric", length = length(pmissing))
diffCC1 = vector(mode = "numeric", length = length(pmissing))
diffCC12 = vector(mode = "numeric", length = length(pmissing))
diffCC13 = vector(mode = "numeric", length = length(pmissing))
diffCC = vector(mode = "numeric", length = length(pmissing))
units = vector(mode = "numeric", length = length(pmissing))

time = 0

for (i in 1:length(pmissing))
{
   percent = pmissing[i]

   var1 = 0   # average of first variance
   var2 = 0   # average of second variance
   var3 = 0   # average of second variance
   sumvar1 = 0    # first variance
   sumvar12 = 0 # first two variances
   sumvar13 = 0 # first three variances
   sumvar = 0 # all variances
   sum1 = 0   # first weight
   sum12 = 0  # first two weights
   sum13 = 0  # first three weights
   sum = 0    # all weights
   CC1 = 0    # CC for first weight
   CC12 = 0   # CC for first two weights
```

```
   CC13 = 0   # CC for first three weights
   CC = 0     # CC for all weights
   unit = 0  # units remain

   for (j in 1:rep)
   {
        if (percent == 0) {x2 = sez$`0`[[j]]}
           if (percent == 5) {x2 = sez$`5`[[j]]}
           if (percent == 10) {x2 = sez$`10`[[j]]}
           if (percent == 15) {x2 = sez$`15`[[j]]}
           if (percent == 20) {x2 = sez$`20`[[j]]}
           if (percent == 25) {x2 = sez$`25`[[j]]}
           if (percent == 30) {x2 = sez$`30`[[j]]}
           if (percent == 35) {x2 = sez$`35`[[j]]}
           if (percent == 40) {x2 = sez$`40`[[j]]}
           if (percent == 45) {x2 = sez$`45`[[j]]}
           if (percent == 50) {x2 = sez$`50`[[j]]}
           if (percent == 55) {x2 = sez$`55`[[j]]}
           if (percent == 60) {x2 = sez$`60`[[j]]}

        t1 = proc.time()
        x3v = na.omit(x2)
        t2 = proc.time()
        time = time + (t2 - t1)
        x3 = matrix(x3v, ncol = dim(x2)[2])

        pca = princomp(x3, cor = TRUE)

        var = pca$sdev^2
        load = (loadings(pca) %*% diag(pca$sdev))


        ## mesures:

        var1 = var1 + var[1]
        var2 = var2 + var[2]
           var3 = var3 + var[3]

           sumvar1 = sumvar1 + sqrt( sum((origvar[1]-var[1])^2))
     sumvar12 = sumvar12 + sqrt( sum((origvar[1:2]-var[1:2])^2) / 2 )
           sumvar13 = sumvar13 + sqrt( sum((origvar[1:3]-var[1:3])^2) / 3 )
           sumvar = sumvar + sqrt( sum((origvar[1:12]-var[1:12])^2) / 12 )
        # sumvar12 = sumvar12 + sqrt( ( (var[1]-origvar[1])^2 + (var[2]-
origvar[2])^2 ) / 2 )

        weight1 = load[, 1]
        if (sign(weight1[1]) != sign(origload[1,1])) weight1 = (-1) *
weight1
        sum1 = sum1 + WRMS(origload[, 1], weight1)
        CC1 = CC1 + abs(measureCC1(origload[, 1], weight1))

        weight2 = load[, 2]
        if (sign(weight2[1]) != sign(origload[1,2])) weight2 = (-1) *
weight2
        sum12 = sum12 + sqrt( 0.5 * WRMS(origload[, 1], weight1)^2 + 0.5 *
WRMS(origload[, 2], weight2)^2 )
        CC12 = CC12 + abs(measureCC12(weight1, origload[, 1], weight2,
origload[, 2]))

        weight3 = load[, 3]
```

```
        if (sign(weight3[1]) != sign(origload[1,3])) weight3 = (-1) *
weight3
            sum13 = sum13 + sqrt( (WRMS(origload[, 1], weight1)^2)/3 +
(WRMS(origload[, 2], weight2)^2)/3 + (WRMS(origload[, 3], weight3)^2)/3 )
        CC13 = CC13 + abs(measureCC13(weight1, origload[, 1], weight2,
origload[, 2], weight3, origload[, 3]))


        weight4 = load[, 4]
        if (sign(weight4[1]) != sign(origload[1,4])) weight4 = (-1) *
weight4
        weight5 = load[, 5]
        if (sign(weight5[1]) != sign(origload[1,5])) weight5 = (-1) *
weight5
        weight6 = load[, 6]
        if (sign(weight6[1]) != sign(origload[1,6])) weight6 = (-1) *
weight6
        weight7 = load[, 7]
        if (sign(weight7[1]) != sign(origload[1,7])) weight7 = (-1) *
weight7
        weight8 = load[, 8]
        if (sign(weight8[1]) != sign(origload[1,8])) weight8 = (-1) *
weight8
        weight9 = load[, 9]
        if (sign(weight9[1]) != sign(origload[1,9])) weight9 = (-1) *
weight9
        weight10 = load[, 10]
        if (sign(weight10[1]) != sign(origload[1,10])) weight10 = (-1) *
weight10
        weight11 = load[, 11]
        if (sign(weight11[1]) != sign(origload[1,11])) weight11 = (-1) *
weight11
        weight12 = load[, 12]
        if (sign(weight12[1]) != sign(origload[1,12])) weight12 = (-1) *
weight12


            sum = sum + sqrt( 1/12 * WRMS(origload[, 1], weight1)^2 + 1/12 *
WRMS(origload[, 2], weight2)^2 + 1/12 * WRMS(origload[, 3], weight3)^2 +
1/12 * WRMS(origload[, 4], weight4)^2 + 1/12 * WRMS(origload[, 5],
weight5)^2 + 1/12 * WRMS(origload[, 6], weight6)^2 + 1/12 * WRMS(origload[,
7], weight7)^2 + 1/12 * WRMS(origload[, 8], weight8)^2 + 1/12 *
WRMS(origload[, 9], weight9)^2 + 1/12 * WRMS(origload[, 10], weight10)^2 +
1/12 * WRMS(origload[, 11], weight11)^2 + 1/12 * WRMS(origload[, 12],
weight12)^2 )
        CC = CC + abs(measureCC(weight1, origload[, 1], weight2, origload[,
2], weight3, origload[, 3], weight4, origload[, 4], weight5, origload[, 5],
weight6, origload[, 6], weight7, origload[, 7], weight8, origload[, 8],
weight9, origload[, 9], weight10, origload[, 10], weight11, origload[, 11],
weight12, origload[, 12]))

            unit<-unit+(dim(x3)[1]/dim(x2)[1])

    }


    variance1[i] = var1 / rep
    variance2[i] = var2 / rep
    variance3[i] = var3 / rep

    diffvar1[i] = sumvar1 / rep
```

```
   diffvar12[i] = sumvar12 / rep
   diffvar13[i] = sumvar13 / rep
   diffvar[i] = sumvar / rep
   diff1[i] = sum1 / rep
   diff12[i] = sum12 / rep
   diff13[i] = sum13 / rep
   diff[i] = sum / rep
   diffCC1[i] = CC1 / rep
   diffCC12[i] = CC12 / rep
   diffCC13[i] = CC13 / rep
   diffCC[i] = CC / rep

   units[i] = unit / rep

}

cat(paste("Computation           time           (hh:mm:ss.ss):           ",
toString(round(time[3]/60/60)), ":", toString(round(time[3]/60)%%60), ":",
toString(round(time[3]%%60,2)), "\n", sep=""))


## mere:

PlotWriteTable(variance1, "listwiseVar1AVG")
PlotWriteTable(variance2, "listwiseVar2AVG")
PlotWriteTable(variance3, "listwiseVar3AVG")

PlotWriteTable(diffvar1, "listwiseRMSvar1")
PlotWriteTable(diffvar12, "listwiseRMSvar12")
PlotWriteTable(diffvar13, "listwiseRMSvar13")
PlotWriteTable(diffvar, "listwiseRMSvar")

PlotWriteTable(diff1, "listwiseRMS1")
PlotWriteTable(diff12, "listwiseRMS12")
PlotWriteTable(diff13, "listwiseRMS13")
PlotWriteTable(diff, "listwiseRMS")

PlotWriteTable(diffCC1, "listwiseCC1")
PlotWriteTable(diffCC12, "listwiseCC12")
PlotWriteTable(diffCC13, "listwiseCC13")
PlotWriteTable(diffCC, "listwiseCC")

PlotWriteTable(units, "units")
```