Вежбе 7

Рад са недостајућим подацима

7.1 Увод

У статистици, *импутација* је процес замене недостајућих података заменским вредностима. Непотпуни подаци честа су појава у било ком стварном проблему, било због природе самог проблема, било због људског фактора. Такви подаци производе много проблема, а неки од главних су повећана пристрасност и смањена ефикасност оцена, као и значајно тежа анализа података.

Договоримо се око терминологије. Податке које анализирамо сматраћемо смештеним у матрицу, где свака врста те матрице представља једну опсервацију[1] (енг. *observation*, рус. *наблюдение*), а свако поље те врсте представља регистровану вредност одговарајућег обележја за дату опсервацију. Одавде следи да свака колона садржи регистроване вредности једног обележја кроз различите опсервације. Неки од тих података могу да недостају.

Рубин и Литл су 1987. године у [6] недостајуће податке класификовали у три категорије, у зависности од тога од чега све зависи вероватноћа недостајања. Изложимо њихову класификацију.

Нека X означава горе наведену $n \times p$ матрицу. Нека X_{miss} означава податке који недостају (те вредности постоје, само су нама недоступне), а нека X_{obs} онзначава податке који су нам доступни. То свакако неће бити матрице, али нам за сада није битно у ком су облику, битно нам је да нам X_{miss} садржи вредности у X које не знамо, а X_{obs} вредности које знамо.

Нека је R једна $n \times p$ матрица чије је поље r_{ij} једнако 1 уколико имамо податак x_{ij}, а 0 уколико тај податак недостаје. Матрица R може да зависи од неколико ствари, од којих су нама од интереса зависност од доступних података и зависност од недостајућих података. Нека су у θ смештени параметри од којих зависи расподела од X.

1. Уколико R не зависи ни од доступних ни од недостајућих података, кажемо да подаци задовољавају **MCAR** услов (енг. *missing completely at random*). Прецизније, подаци су MCAR уколико је
$$\mathbf{P}\{R \mid X_{obs}, X_{miss}, \theta\} = \mathbf{P}\{R \mid \theta\}.$$

2. Уколико R зависи од посматраних, али не и од недостајућих података, кажемо да подаци задовољавају **MAR** услов (енг. *missing at random*). Формално, то значи да је
$$\mathbf{P}\{R \mid X_{obs}, X_{miss}, \theta\} = \mathbf{P}\{R \mid X_{obs}, \theta\}.$$

[1] Никако *обзервацију*. Консултовати [15]. Синоними би били *посматрање*, *опажање* итд, али се латински термин одомаћио у статистици.

3. Уколико R зависи и од доступних и од недостајућих података, кажемо да су подаци **MNAR** (енг. *missing not at random*).

$$\mathbf{P}\{R \mid X_{obs}, X_{miss}, \theta\} = \mathbf{P}\{R \mid X_{obs}, X_{miss}, \theta\}.$$

Детаљније дефиниције, појашњења и примери у вези са ова три појма могу се пронаћи у [18].

Пример 7.1. Један од стандардних модела недостајања података јесте онај који почива на логистичкој функцији, а омогућава да се генеришу и MCAR и MAR и MNAR недостајања, за одговарајући избор параметара.

Нека су нам подаци $X = (X_1, X_2)$ дати за n опсервација и $p = 2$ овележја и нека је $R = (R_1, R_2)$, где су X_1, X_2, R_1 и R_2 колоне (последње две - колоне нула и јединица). Тада генеришемо недостајања као:

$$P(R_2 = 0) = \psi_0 + \psi_1 \frac{e^{X_1}}{1 + e^{X_1}} + \psi_2 \frac{e^{X_2}}{1 + e^{X_2}},$$

где је $\psi = (\psi_0, \psi_1, \psi_2)$ параметар са ненегативним компонентама[2]. Наравно, подразумевамо да је горња једнакост узета у одговарајућим компонентама вектора. Уколико желимо да генеришемо MCAR недостајање у другој колони, поставићемо вредност параметра на $(\psi_0, 0, 0)$, за MAR на $(\psi_0, \psi_1, 0)$, $\psi_1 \neq 0$, а за MNAR на (ψ_0, ψ_1, ψ_2), $\psi_2 \neq 0$.

За добијене податке рачунамо $P(R_2 = 0)$, те ако је веће од одабраног прага (популарно - трешхолда), податак бришемо.

7.2 Игнорабилно и неигнорабилно недостајање

Наставићемо дискусију имајући на уму претходни пример. Ми, у суштини, имамо два типа параметара (потенцијално вишедимензионих). Први јесте параметар ψ од којег зависи расподела за R. Други и битнији параметар јесте параметар θ од којег зависи расподела самих података X.

Свака анализа података, барем између осталог, има за задатак да оцени вредност параметра θ, који је увек у вези са неком знајаном статистиком на подацима: стредњом вредношћу, дисперзијом, коефицијентом спљоштености код неких расподела итд. Стога, веома је битно имати квалитетне оцене параметра θ. С друге стране, параметар ψ је „сметајући'', јер је његово присуство последица несавршености самих података са којима радимо.

Сходно претходно реченом, ваљало би знати када можемо сазнати све о θ, а да не знамо ништа о ψ, то јест када можемо сазнати параметре података, а да не знамо параметре недостајања. Ако хоћемо да се изразимо строго формално, можемо уочити да подаци који су нама доступни заправо нису подаци X_{obs}, већ здружени подаци[3] (X, R). У општем случају, расподела од (X, R) ће зависити од (θ, ψ).

Дефиниција 7.1. У горњим ознакама, недостајање је *игнорабилно* ако расподела за (X, R) зависи само од θ.

Рубин и Литл су у [11] показали довољне услове за игнорабилност у случају игнорабилности при закључивању методом максималне веродостојности. То ћемо дати у следећој теореми.

Теорема 7.1. *Недостајање је игнорабилно за закључивање коришћењем максималне веродостојности ако важи:*

1. *Недостајање је MCAR или MAR.*

[2] И, наравно, природним ограничењима, јер вероватноћа не сме бити већа од 1

[3] Како недостајања могу бити „разбацана" свуда по X ово неће чинити матрицу, али се може направити матрицом тако што се на места где је $r_{ij} = 0$ у X врше разне манипулације индикаторима. То сада није од велике важности.

2. *Параметри θ и ψ су различити, у смислу да је параметарски простор за (θ, ψ) једнак Декартовом производу параметарских простора за θ и за ψ.*

Показали су и да за правилно закључивање на Бајесовски начин, поред ова два услова мора да важи и априорна независност параметара θ и ψ.

Напомена. У пракси, услов различитости параметара је углавном испуњен, те се сва пажња усредсређује на констатацију тога да ли су недостајања MAR. Тестови тог типа превазилазе овај рад.

Још једна напомена. Уочимо да то што је недостајање игнорабилно не значи да можемо потпуно да га занемаримо, јер чак да су подаци и MCAR, уколико је недостајање 70% све оцене ће бити неквалитетне. Рубинови и Литлови услови хоће да кажу да неће бити *систематске* пристрасности у оценама, али оне свакако могу бити неупотребљиве због превелике дисперзије или преспоре конвергенције ка правој вредности параметра.

7.2.1 Последице игнорабилности

Концепт игнорабилности игра значајну улогу у конструкцији импутационих модела. То је зато што је наша генерална идеја та да одредимо расподелу $X_{miss}|X_{obs}, R$, те да из ње генеришемо импутационе вредности. Рубин је показао ([5]) да при претпоставци игнорабилности важи да расподела за $X_{miss}|X_{obs}, R$ не зависи од R, што специјално значи да је она једнака расподели за $X_{miss}|X_{obs}, R = 1$, па апостериорну (у смислу након недостајања) расподелу можемо моделовати само на основу доступних података. Уколико не важи услов игнорабилности, она се мора моделовати узевши и R у обзир.

Примере игнорабилног и неигнорабилног недостајања овде нећемо наводити јер премашују оквире (а нарочито циљеве) овога рада, а заинтересовани се упућују на [18], страна 40.

7.3 Методи обраде недостајућих података који не подразумевају импутацију

Сада ћемо се кратко осврнути на два најчешћа начина за руковање недостајућим подацима који не подразумевају импутацију, а то су уклањање целих опсервација и уклањање по паровима.

7.3.1 Уклањање целих опсервација

Један од најједноставнијих начина за руковање недостајућим подацима јесте **уклањање целих опсервација** (енг. *listwise deletion/complete-case analysis*, рус. *анализ полных наблюдений*). Тај метод подразумева уклањање целе једне опсервације из узорка, уколико јој недостаје макар једно поље. Може се показати да уколико подаци које посматрамо задовољавају MCAR услов, оцене добијене на овај начин јесу непристрасне. У случају да MCAR услов није задовољен (врло често[4]), оцене нису чак ни непристасне, а сви остали проблеми остају и продудбљују се.

Главна погодност овог метода је лакоћа рада. Уклањање целе опсервације уклања потребу за специјализованим софтвером за рад са некомплетним подацима, као и за посебним техникама за рад с њима. Ипак, избацивање целих опсервација, ако ништа друго, смањује обим узорка, а дисперзија великог броја оцена расте са тим смањење. Сходно томе, у већини ситуација губици су већи од предности које доноси, тако да се овај метод сматра једним од најлошијих.

7.3.2 Уклањање по паровима

Следећи метод за руковање недостајућим подацима јесте тзв. **уклањање по паровима** (енг. *pairwise deletion/available-case analysis*, рус. *попарное удаление пропущенных наблюдений*). Суштина овог метода

[4]Класичан пример незадовољености MCAR услова јесу анкете. Врло често у анкетама појављују се питања која велики број људи сматра осетљивим, те одбија да на њих одговори. Рецимо, људи веома често одбијају да говоре о висини својих примања.

јесте да се опсервација уклања уколико јој поље недостаје за неку конкретну анализу, а иначе не. Канонски пример јесте рачунање коваријације међу колонама (обележјима): уколико за барем једну опсервацију недостаје вредност једног од два обележја, избацујемо ту опсервацију за оба.

Овај метод зна дати боље резултате од претходног, али врло често зна дати и потпуно бесмислене оцене параметара; на пример, дешава се да оцена коефицијента корелације буде већа од 1. Заиста, оцена коефицијента корелације између колона X_k и X_l дата је са

$$\frac{\sum_i (x_{ik} - \bar{x}_k)(x_{il} - \bar{x}_l)}{\bar{s}_{X_k} \bar{s}_{X_l}},$$

што у случају потпуних података не прави проблем. Међутим, неки софтверски пакети за оцене узорачких дисперзија из имениоца користе само оне врсте које су познате и за једну и за другу колону, док други дисперзију за једну колону оцењују на основу у њој познатих врста, аналогно за другу. Тада се може десити да оцена коефицијента корелације „испадне'' из интервала $[-1, 1]$, из очигледних разлога.

Такође, може се десити да овај метод да оцену коваријационе матрице која је непозитивно (уместо позитивно/ненегативно) дефинитна, што даље ствара проблеме код свих врста анализе података које користе коваријациону матрицу. Детаљније се може прочитати у [16] на страни 40.

7.3.3 Једнострука и вишеструка импутација

У неким случајевима смислено је недостајућу вредност заменити тачно једном конкретном вредношћу (углавном статистиком која зависи од доступних података), док у другим ситуацијама постоји више потенцијалних кандидата којима има смисла попунити празнину. Стога, понекад је довољно просто уписати вредност у празно поље, док је у другим ситуацијама потребно одабрати најбољу могућу оцену недостајућег податка, или све смислене оцене на неки начин објединити у једну. Отуда и потреба за појмом вишеструке импутације.

7.4 Класичне методе

Сада ћемо изложити пар класичних метода за импутацију недостајућих података, који не подразумевају употребу регресије, а то су импутација средњом вредношћу и импутација узорачком медијаном. Ове методе су прилично јасне саме по себи, тако да ће ова глава бити прилично кратка.

Код импутације средњом вредношћу посматрају се недостајућа поља по колонама. Срачуна се средња вредност поља која су доступна, и она се упише у сва недостајућа поља. Још једна идеја може деловати смислено, а то је да се поменута средња вредност упише само у једно празно поље, те да се поново срачуна средина доступних поља, која сада укључују и оно које смо попунили, те да тако итерирамо све док не попунимо сва доступна поља. Међутим, тривијално је показати да то попуњава празна поља на идентичан начин као и прва (и једноставнија) идеја.

Код импутације узорачком медијаном идеја је слична претходној: срачуна се узорачка медијана сваке од колона и упише на сва празна места. Као и малопре, то је еквивалентно уписивању поље по поље и рачунању медијане сваки пут.

7.5 kNN импутација

Нека је x врста матрице X и нека је x^* настало од x избацивањем поља којима недостаје вредност. Импутацију вршимо у 2 корака:

1. Рачунамо растојање[5] између x^* и сваке од врста у X^c (подматрица комплетних врста у X). Идентификујемо k најближих, $k \in \mathbb{N}$.

[5] У норми по избору - најчешће L^2.

2. Недостајуће поље од x замењујемо медијаном[6] идентификованих k врста.

Овај метод је прилично смислен и једноставан за разумети, али има једну великуману: одабир одговарајућег k. Не постоји „рецепт'' за тај одабир, а он се углавном врши на основу додатних сазнања о подацима (и обима узорка, наравно).

7.6 Вишеструка импутација

Тешко је рећи ко се и када први сетио да недостајуће податке мења нечим смисленим. Било како било, „оцем импутације'' као математичке дисциплине сматра се амерички математичар[7] Доналд Б. Рубин (1943 -), тренутно професор емеритус на Универзитету Харвард. Уколико се неко и не слаже с тиме да је Рубин отац импутације, сви се слажу да је Рубин отац вишеструке импутације.

Идеја вишеструке импутације почива у математичком појму неодређености[8] (ентропије). Пре извршења било ког експеримента (у вероватносном смислу), постоји неодређеност о његовом исходу. Након извршења експеримента та неодређеност се губи и прелази у информацију коју његова реализација носи. У нашем случају, то би значило да се неодређеност губи онда када сазнамо праву вредност недостајућег податка и упишемо је где треба. Како то није реалан случај, јер не знамо која вредност је ту била, ми „погађамо''. Свакако да две смислене процене дају више информација о непознатој вредности од једне смислене процене. Отуда и идеја да се, уместо једном вредношћу, недостајући податак мења скупом вредности.

Напомена. Сви горњи појмови (неодређеност, информација, „носити информацију") могу се формализовати, али то је тема за неки други рад.

Идеја вишеструке импутације потекла је из Рубиновог ума 1977. године, а своје коначно обличје доживела је у [4], што је допуњавано и издавано још неколико пута.[9]

Иако смислена идеја, вишеструка импутација пред себе поставља неколико озбиљних изазова:

1. Како генерисати сваку од импутационих вредности?
2. Који је њихов оптималан број?
3. Како комбиновати све те импутационе вредности у једну (јер је то крајњи циљ, наравно)?
4. Како мерити квалитет технике?

На ова питања је углавном одговорено, а на већину је одговорио сам Рубин. Ми ћемо укратко описати сваки од корака, али за почетак морамо поставити амбијент у коме радимо.

Параметар популације Q јесте било која функција вредности неког обележја на целој популацији (нпр. средња вредност, медијана итд). Када вучемо узорак, ми не можемо израчунати тачну вредност тог обележја, већ само њену *оцену* $\hat{Q}$. Оваква оцена треба да поседује одређен квалитет, а Рубин од ње захтева да буде *непристрасна* и *валидна у смислу поверења*. Непристрасност подразумева да је очекивање ове оцене (просек вредности преко свих могућих узорака[10]) једнако Q, односно[11] $E(\hat{Q}|X) = Q$.

[6] Може и средњом вредношћу, али медијана је робуснија.

[7] Почео да студира физику, дипломирао психологију, докторирао статистику. Свашта нешто по вокацији, математичар у пракси.

[8] Нећемо је овде формално дефинисати, већ ћемо је само описати, што је довољно за наше потребе. За детаљније видети [14].

[9] Отуда се и помиње више пута у литератури - нешто што има у једном издању, нема у другом.

[10] Дакле, имамо дискретну случајну величину где су њене вредности све могуће вредности оцене $\hat{Q}$ за различите узорке, а вероватноће су вероватноће извлачења тих узорака.

[11] Ознака представља малу злоупотребу нотације, али је задржавамо јер је користе и Рубин и Бурен, због лакшег комбиновања литературе.

Да бисмо објаснили валидност у смислу поверења, узмимо да је U оцена дисперзије за $\hat{Q}$. Кажемо да је оцена валидна у смислу поверења ако очекивање ове оцене, као њен просек преко свих узорака, задовољава:

$$E(U|X) \geq D(\hat{Q}|X),$$

где је $D(\hat{Q}|X)$ дисперзија саме оцене, проузрокована узорковањем. Статистички тест са номиналним прагом значајности α треба да одбацује нулту хипотезу у *највише* $100\alpha\%$ случајева у којима је она тачна. Процедура је валидна у смислу поверења ако ово важи. За више детаља видети [8], страна 3.

7.6.1 Рубинова правила

Убацимо сада у причу и недостајуће податке. Права вредност параметра популације Q је непозната. Претпоставимо да смо направили њену оцену $\hat{Q}$. Количина информације (која се може и формализовати) коју $\hat{Q}$ даје о Q зависи од тога шта знамо о X_{miss}. Када бисмо га могли реконструисати са стопроцентном тачношћу, онда бисмо знали и параметар. Како то, наравно, није могуће, ми морамо да посматрамо расподелу за Q при разним X_{miss}.

Могуће вредности параметра Q на основу нама знаних података X_{obs} садржане су у расподели $P(Q|X_{obs})$. Она се може записати као

$$P(Q|X_{obs}) = \int P(Q|X_{obs}, X_{miss})P(X_{obs}|X_{miss})dX_{miss}, \tag{7.1}$$

а то на основу формуле потпуне вероватноће. Горњи интеграл је само формалан запис, и по потреби постаје и сума, у дискретном случају[12].

Напомена. Шта је шта у горњој једначини? $P(Q|X_{obs})$ јесте апостериорна расподела за наш параметар, при услову доступних података. Ово је оно што нас занима. Откуд сад то? Па, приметимо какав нам је до сада ток мисли: желимо да оценимо параметар, али имамо недостајуће податке. Ми онда прво оценимо њих, па затим на основу попуњених података и параметар. То је дупли посао! Сва информација која је нама доступна садржана је у X_{obs}. Ако њу искористимо за добијање оцене за X_{miss}, то је само због тога што ми тиме знамо да рачунамо, а не зато што говори нешто више. Због тога желимо да „непречимо" пут, те да оценимо параметар на основу доступних података. ОК, и?

Рецимо да је од интереса оцењивање средње вредности обележја на популацији. Ако га оценимо аритметичком средином на доступним подацима, правимо потенцијално огромну грешку, јер подаци могу да недостају веома „пристрасно" (нпр. недостају сви већи од неке вредности). Стога тражимо расподелу параметра при доступним подацима, те тек онда оцењујемо шта треба.

$P(Q|X_{obs}, X_{miss})$ представља теоријску „идеалну" расподелу параметра при свим доступним подацима. $P(X_{miss}|X_{obs})$ представља расподелу недостајућих података при доступним.

Једначину је згодно тумачити здесна налево. Претпоставимо да користимо расподелу $P(X_{miss}|X_{obs})$ да „вучемо'' импутационе вредности за X_{miss}, које ћемо означити са $\dot{X}_{miss}$. Онда можемо користити $P(Q|X_{obs}, \dot{X}_{miss})$ да срачунамо Q за дато $\dot{X}_{miss}$. Понављамо ове кораке пролазећи кроз све могуће вредности за $\dot{X}_{miss}$, те саберемо/интегралимо. Овиме смо компликовану величину са леве стране у нашој једначини, изразили преко једноставнијих расподела, из којих можемо да вучемо узорак.

Може се показати да важи следећа формула:

$$E(Q|X_{obs}) = E(E(Q|X_{obs}, X_{miss})|X_{obs}).$$

Да не би било забуне, објаснићемо шта горња једнакост представља. Дакле, очекивање условне расподеле параметра Q при услову доступних података једнако је условном, при услову доступних података,

[12] И сума је интеграл по бројачкој мери, тако да заправо говоримо о интегралу чији је носач скуп свих вредности које може узети X_{miss}, макар и вишедимензионих. dX_{miss} заправо представља интеграцију по оној Лебег-Стилтјесовој вероватносној мери коју генерише функција расподеле од X_{miss}, макар и вишедимензиона.

очекивању средње вредности параметра Q за разна „извлачења'' недостајућих података (нотација унутрашњег очекивања је злоупотребљена, као и раније).

Претходна једнакост даје нам идеју како комбиновати резултате вишеструких импутација. Претпоставимо да је $\hat{Q}_l$ оцена добијена из l-те поновљене импутације. Онда је комбинована оцена дата са

$$\bar{Q} = \frac{1}{m}\sum_{l=1}^{m}\hat{Q}_l.$$

Напомена. Дозвољавамо да параметар буде вишедимензион; тада је свако $\hat{Q}_l$ један $k \times 1$ вектор, а уместо оцене дисперзије треба посматрати оцену коваријационе матрице.

Тада за дисперзију апостериорне расподеле важи позната формула Теорије вероватноћа:

$$D(Q|X_{obs}) = E(D(Q|X_{obs}, X_{miss})|X_{obs}) + D(E(Q|X_{obs}, X_{miss})|X_{obs}).$$

Први сабирак представља просечну дисперзију параметра Q, где се просек узима преко различитих X_{miss} (оних које смо раније „вукли''). Други сабирак преставља дисперзију (коваријациону матрицу) међу очекивањима параметра на основу комплетних података, пролазећи кроз различите X_{miss}. Први сабирак на енглеском носи назив *within-variance*, а други *between-variance*.

Нека $\bar{U}_\infty$ и B_∞ означавају граничне вредности оцена горњих двају сабирака када број импутација m тежи бесконачности. Тада се апостериорна (при услову доступних података, да се подсетимо) дисперзија за Q може оценити са

$$T_\infty = \bar{U}_\infty + B_\infty.$$

Сада се јасно наслућује како ћемо оценити T_∞ на основу коначног m. Рачунамо да је

$$\bar{U} = \frac{1}{m}\sum_{l=1}^{m}\bar{U}_l,$$

где је $\bar{U}_l$ оцена коваријационе матрице од $\hat{Q}_l$, у l-тој импутацији. Стандардна непристрасна оцена другог сабирка јесте

$$B = \frac{1}{m-1}\sum_{l=1}^{m}(\hat{Q}_l - \bar{Q})(\hat{Q}_l - \bar{Q})^T,$$

где претпостављамо потенцијалну вишедимензионалност параметра, па отуда и транспонат.

Било би примамљиво рећи да је (непристрасна) оцена за T_∞ једнака збиру $\bar{U} + B$. Међутим, ту би се заборавила чињеница да је и само $\bar{Q}$ оцењено, те само апроксимира $\bar{Q}_\infty$. Рубин је у [5] показао да је члан који у суми „фали'' систематски, и једнак B_∞/m. Ово значи да за велико m он бива занемарљив, али за мало m добијамо мању дисперзију него што јесте. Како B апроксимира B_∞, можемо писати

$$T \approx \bar{U} + B + B/m = \bar{U} + \left(1 + \frac{1}{m}\right)B.$$

Сада ћемо на једно место записати две централне једначине из претходне приче:

$$\bar{Q} = \frac{1}{m}\sum_{l=1}^{m}\hat{Q}_l, \tag{7.2}$$

$$T \approx \bar{U} + \left(1 + \frac{1}{m}\right)B. \tag{7.3}$$

$\bar{Q}$ је оцена параметра добијена вишеструком ипмутацијом, а T оцена дисперзије те оцене. Горње две једначине познате су као *Рубинова правила*.

7.6.1.1 Три извора варијабилности

Имамо, дакле, параметар од интереса, Q који желимо да оценимо на основу некомплетних података. Испоставља се да његова дисперзија долази из трију извора:

1. $\bar{U}$, дисперзија која је узрокована чињеницом да узимамо узорак уместо да посматрамо целу популацију; стандардна је статистичка мера варијабилности;
2. B, додатна варијабилност узрокована тиме што у нашем узорку имамо недостајање;
3. B/m, додатна варијабилност потекла од симулација, то јест из чињенице да је само $\bar{Q}$ оцењено за коначно m.

Последња ставка је неопходна да би вишеструка импутација „радила'' за мале m, јер ће иначе p-вредности бити исувише мале, а интервали поверења неприродно кратки (јер је дисперзија наизглед мања, а заправо није). Што је веће m, то је мањи утицај симулација на укупну дисперзију. Ранији савети говорили су да се за m узме 3. 5, или 10, док данашње искуство показује да је боље узети веће m и препоручује се 50. Ове препоруке су потекле из искуства, а не из формалног рачуна.

7.6.1.2 Валидна импутација

На почетку ове главе смо имали појам валидности оцене у смислу поверења, што је значило да стварни ниво поверења треба да буде једнак номиналном, или већи. У тој уводној причи нисмо имали недостајуће податке, те ћемо сада разрадити појам валидности у случају присуства недостајања. У суштини, анализу података можемо посматрати на три нивоа, а који су дати у наредној табели.

Популација	Комплетан узорак: $X = (X_{obs}, X_{miss})$	Некомплетан узорак: X_{obs}
Q	$\hat{Q},\ U \sim D(\hat{Q})$	$\bar{Q}, \bar{U}, B \sim D(\bar{Q})$

Табела 7.1: Нивои анализе података

„Најлакши'' ниво за рад јесте онај у коме имамо извршен попис, то јест имамо вредности нашег (наших) обележја на целој популацији. Ту не постоји никаква неодређеност о вредности параметра Q, те је ту ситуација јасна. Наредни ниво јесте ниво у коме извлачимо узорак из популације. Тада вредност $\hat{Q}$ која представља (макар и непристрасну, што и захтевамо барем асимптотски) оцену параметра Q не носи сву информацију о параметру, зато што не знамо у ком степену оцена „одступа'' од праве вредности параметра. О томе нам говори U, оцена коваријационе матрице/дисперзије оцене $\hat{Q}$. Тада пар $(\hat{Q}, U)$ садржи све што знамо о параметру.

Онај ниво који нама и јесте од интереса, а који ћемо с правом описати као „најтежи'', јесте онај када не само да имамо узорак, већ је он још и некомплетан. Сада оно што је било оцена, узима улогу онога што треба оценити, те добијамо $\bar{Q}$ као оцену за $\hat{Q}$, $\bar{U}$ као оцену за U, као и нову величину B, која „купи" варијабилност насталу чињеницом да недостајање постоји.

Процес импутације би требало, ако ништа друго, да барем даје адекватне оцене за $\hat{Q}$ и U. Изложићемо дефиницију дефиницију валидности дату у [18]. Импутациони процес је *валидан у смислу поверења* ако приближно (асимптотски), важе следећа три услова:

$$E(\bar{Q}|X) = \hat{Q} \tag{7.4}$$

$$E(\bar{U}|X) = U \tag{7.5}$$

$$\left(1 + \frac{1}{m}\right) E(B|X) \geq D(\bar{Q}). \tag{7.6}$$

Напомена. Овде се хипотетички комплетни подаци сматрају фиксним, то јест реалним бројевима при узимању очекивања. Шта је онда случајно? Случајна је матрица R, индикатор недостајања, и очекивање заправо иде по њеној расподели (јасно је да се за R мора претпоставити расподела).

Услови валидности су аналогни онима у случају без недостајања, те их нећемо посебно објашњавати. Нагласићемо само да валидност процеса импутације зависи и од самих параметара $(\hat{Q}, U)$, али и од расподеле недостајања. Стога се може десити да је иста процедура некад валидна, а некад није. Тестови валидности импутационих механизама свакако превазилазе овај рад, а радознали се упућују на [18].

7.6.1.3 Односи варијабилности

Задржаћемо досадашње ознаке, те посматрати количник:

$$\lambda = \frac{B + B/m}{T}.$$

Овај количник можемо тумачити као пропорцију варијабилности у подацима чије се порекло може приписати недостајању података. Он је једнак нули или уколико недостајања нема, или уколико са сигурношћу можемо реконструисати недостајуће податке.

Количник

$$r = \frac{B + B/m}{\bar{U}}$$

назива се *релативни пораст варијабилности потекао из недостајања* (енг. relative increase in variance due to nonresponse). Важи веза $r = \lambda/(1-\lambda)$.

Још једна мера коју ћемо размотрити јесте *удео информације о Q који немамо због недостајања.*[13] Ова мера дефинисана је са

$$\gamma = \frac{r + 2/(\nu + 3)}{1 + r}.$$

За ову меру неопходно је знати (оцену за) број степени слободе, о којем ћемо дискутовати у наредном пододељку.

Иако нисмо експлицитно нагласили, јасно је да су горње мере дефинисане за параметар који је скалар. Ако је параметар димензије k, тада се добијају аналогне величине:

$$\bar{\lambda} = \left(1 + \frac{1}{m}\right) \mathrm{tr}(BT^{-1})/k, \quad \bar{r} = \left(1 + \frac{1}{m}\right) \mathrm{tr}(B\bar{U}^{-1})/k.$$

За детаљније описе ових величина треба погледати [18], страна 47, као и литературу на коју се тамо упућује.

7.6.1.4 Степени слободе

Број степени слободе представља број опсервација (елемената узорка) умањен за број параметара у моделу. Рачунање броја степени слободе не може бити исто и кад имамо и кад немамо недостајање. У случају недостајућих података, Рубин је у [5] (једначина 3.1.6) извео да је

$$\nu_{old} = (m-1)\left(1 + \frac{1}{r^2}\right) = \frac{m-1}{\lambda^2},$$

где смо користили индекс *old* (енг. за *стар*), јер је оцена претрпела извесне корекције. Најнижа могућа вредност је $\nu_{old} = m - 1$, што значи да се сва варијабилност у подацима може приписати недостајању. Супротно од тога је ν_{old} значи да сва варијабилност потиче од узорковања, па или недостајања нема, или смо недостајуће податке савршено реконструисали.

[13] Појам је детаљније описан у [5], једначина 3.1.10.

Бернард и Рубин ([9]) су уочили да претходни израз може узети вредности веће од обима комплетног узорка, те да оцена има смисла само за велике узорке. За мале узорке неопходна је корекција оцене, да би иста имала смисла. Означимо са ν_{com}, број степени слободе за $\bar{Q}$ у хипотетички комплетним подацима. Ако је узорак обима n, а параметар димензије k, можемо ставити $\nu_{com} = n - k$. Барнард и Рубин су извели да је тада оцена броја степени слободе која урачунава информацију која недостаје дата са

$$\nu_{obs} = \frac{\nu_{com} + 1}{\nu_{com} + 3}\nu_{com}(1 - \lambda).$$

Оцена броја степени слободе коју они препоручују за тестирање јесте

$$\nu = \frac{\nu_{old}\nu_{obs}}{\nu_{old} + \nu_{obs}}. \tag{7.7}$$

Она је увек мања или једнака од ν_{com}, што смо и хтели. Ако је $\lambda = 0$, тада је $\nu = \nu_{com}$, што нам одговара јер знамо да том случају одговара да недостајање не утиче на варијабилност. Супротан случај јесте $\lambda = 1$ и тада је $\nu = 0$. Тада је сва варијабилност потекла од недостајања, те тада нема никаквог смисла радити било какве тестове, јер не поседујемо никакву информацију о параметру.

Постоји још корекција ове оцене, али ми их нећемо детаљније обрађивати. Набројане су у [18] на страни 48. Проћи ћемо кроз један пример у R-у.

7.6.1.5 Показни пример

У програмском језику R постоји мноштво пакета који се баве импутацијом недостајућих података, а онај који је специјализован за вишеструку импутацију назива се mice. Његов аутор је сам Стеф ван Буурен, аутор [18] и ученик Доналда Б. Рубина.

У бази nhanes налазе се подаци о разним здравственим параметрима пацијената. Наравно, у бази има недостајања. Два предиктора су категоричка (кодирана), а два нумеричка. Број опсервација је 25. Сада ћемо видети како се врши вишеструка импутација.

```
library(mice)
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
imp <- mice(nhanes, print = FALSE, m = 10, seed = 10000)
```

imp је класе mids, што је скраћеница за *Multiply imputed data set*. Функција сама бира механизам импутације, то јест начин на који се врше извлачења. То може да се предочи функцији као аргумент, али о том - потом. Подразумевани механизам познат је под називом *Predictive mean matching*, који ћемо ми и описати у наредној глави.

```
fit <- with(imp, lm(bmi ~ age))
```

Рачуна се оцена параметара за сваки импутирани dataset, на основу задате формуле. Овде су параметри два регресиона коефицијента, а формула је стандардна из линеарне регресије. Класа објекта fit јесте mira, што је скраћено за *Multiply imputed repeated analyses*.

```
estimate <- pool(fit)
estimate
```

```
## Class: mipo    m = 10
##          term  m estimate      ubar         b        t dfcom        df
## 1 (Intercept) 10    29.35 3.5768250 2.4717747 6.295777    23  9.649768
## 2         age 10    -1.72 0.9512832 0.4818339 1.481301    23 11.419504
##         riv    lambda       fmi
## 1 0.7601580 0.4318692 0.5216939
## 2 0.5571604 0.3578054 0.4468784
```

Видимо да је класа наше оцене `mipo`, што је скраћено од *Multiple imputation pooled object*. Наравно, pooling се врши уз помоћ раније наведених Рубинових правила. Прођимо сада кроз све елементе у `estimate`. Јасно, `m` означава број вишеструких импутација које смо задали. `estimate` даје оцене регресионих коефицијената, које смо и тражили. Оне одговарају оцени $\bar{Q}$ из претходних поглавља. Колоне `ubar`, `b` и `t` представљају оцене дисперзија које смо имали раније под истим именом (bar = црта изнад), узете за сваки коефицијет посебно (јер би иначе биле 2×2 матрице). Колона `dfcom` одговара оцени ν_{com}, а колона `df` одговара оцени ν добијеној Барнард-Рубиновом корекцијом. Релативни пораст варијабилности r дат је колоном `riv` (*relative increase in variance*). `lambda` је јасно. Мера γ дата је колоном `fiv`, што је од енглеског *fraction of missing information*. Наравно, и она је рачуната по параметру.

7.6.2 Интервали поверења и тестови

Поред саме тачкасте оцене параметра која нас занима, веома често је од интереса пронаћи интервал поверења за тај параметар/оцену, као и тестирати да ли параметар има неку вредност, или припада неком скупу. Да би се то успрешно спровело, треба знати расподелу оцене.

У свим до сада познатим методама претпоставља се да важи да је

$$Q - \hat{Q} \sim \mathcal{N}(0, U),$$

где су све ознаке исте као из претходних одељака.

Напомена. Када горњи услов није испуњен, тражи се нека функција параметра за коју горња нормалност барем приближно важи.

Суштински, ситуација може да се грана у два случаја: први, у којем је параметар скалар и други, у којем је параметар вектор (вишедимензион). Други случај је знатно компликован и ми се њиме нећемо овде бавити, а заинтересовани о њему могу пронаћи у [18], у 5. глави. Дакле, претпоставићемо да је параметар једнодимензион.

7.6.2.1 Закључивање у случају једнодимензионог параметра

У овоме случају U је број, а не матрица, а нормална расподела о којој је било речи је једнодимензиона.

Напомена. Случај једнодимензионог параметра може се примењивати и када је параметар вишедимензион, али се тада процедура понавља за сваку његову компоненту као параметар за себе.

Рубин је показао да асимптотски важи релација

$$\frac{Q - \bar{Q}}{\sqrt{T}} \sim t_\nu,$$

где је t_ν Студентова расподела са ν степени слободе, где је ν дефинисано једначином 7.7.

Стандардно, $100(1-\alpha)\%$ интервал поверења за Q рачуна се као

$$\bar{Q} \pm t_{\nu,1-\alpha/2}\sqrt{T},$$

где је $t_{\nu,1-\alpha/2}$ ознака за одговарајући квантил Студентове расподеле са ν степени слободе.

Уколико тестирамо нулту хипотезу $Q = Q_0$, за неку конкретну вредност Q_0, Рубин је показао да се p-вредност теста налази по формули:

$$p = P\left[F_{1,\nu} > \frac{(Q_0 - \bar{Q})^2}{T}\right],$$

где је $F_{1,\nu}$ ознака за случајну величину са Фишеровом расподелом са једним и ν степени слободе.

У `R`-у, претходне показатеље добијамо позивом функције `summary`, на објекат класе `mipo`. Ми ћемо искористити `estimate` из претходног примера који смо имали.

```
summary(estimate)
```

```
##          term estimate std.error statistic        df      p.value
## 1 (Intercept)    29.35  2.509139  11.69724  9.649768 5.143642e-07
## 2         age    -1.72  1.217087  -1.41321 11.419504 1.842675e-01
```

У нашем примеру (подсетимо се, проста линеарна регресија), променљива `statistic` представља класичну Валдову тест статистику на коју смо навикли код линеарне регресије. Уколико желимо да нам `summary` исписује и интервале поверења, довољно је као додатни аргумент додати `conf.int = TRUE`.

7.6.3 Квалитет импутационог метода

Рубин је показао да се бенефити вишеструке импутације „виде'' тек у случају да су импутациони методи валидни, у смислу дефиниције коју смо раније имали. Провера валидности углавном се врши симулацијама. Суштински, постоје два механизма која утичу на доступне податке: механизам узорковања и механизам недостајања. Симулацијама могу да се третирају сваки посебно, али и комбиновано. То нас доводи до три суштинска симулациона дизајна.

1. **Само механизам узорковања.** Одаберемо познат параметар, то јест расподелу са познатим параметром. Вадимо узорке $X^{(s)}$, на основу њих правимо оцене $\hat{Q}^{(s)}$ и $U^{(s)}$ и рачунамо просек за велико s. Требало би да добијемо вредности параметра и његове коваријационе матрице које су блиске правој.

2. **Оба механизма комбиновано.** За познат параметар вучемо узорке $X^{(s)}$, затим у сваком од њих генеришемо недостајања, те добијамо $X_{obs}^{(s,t)}$; импутирамо, оцењујемо $\hat{Q}^{(s,t)}$ и $T^{(s,t)}$ и упросечимо преко свих s и t.

3. **Само механизам недостајања**. За дату оцену $(\hat{Q}, U)$ генеришемо недостајања на подацима из које смо је добили, импутирамо велики број пута, те упросечимо добијене $(\bar{Q}, \bar{U})^{(t)}$, $B^{(t)}$. Предност овог приступа јесте што нам не треба модел података/параметра на популацији.

Неке честе мере квалитета које се користе јесу пристрасност оцене $\bar{Q}$, стопа покривања (пропорција интервала поверења који садрже праву вредност), просечна ширина интервала поверења, као и често виђани корен средње квадратне грешке. Сваки од показатеља има и своје предности и своје мане, које често зависе и од конкретне расподеле података, те их нећемо детаљније ни разматрати да не бисмо превише отишли у ширину.

7.7 Predictive mean matching

7.7.1 Како вишеструко импутирати? Показни пример у `R`-у

До сада смо, што код Рубинових правила, што код приче о евалуацији импутационог метода, говорили о кораку „импутирамо податке'', а затим са тако комплетним узорком радимо нешто. Ми смо тај

механизам којим импутирамо сматрали датим, само нама у датом тренутку непознатим/небитним (на енглеском се то зове *blackboxing*). Сада је дошло време да кажемо нешто о томе како се то ради. Кренућемо поступно, на примеру.

Напомена. Пример је преузет из [18], али је у извесној мери модификован. Није било сврхе смишљати неки оригиналнији, јер сваки погађа исту поенту, а база из овог примера је лепа за показивање идеја, због јаке линеарне везе предиктора.

Одабраћемо базу `whiteside` из пакета `MASS`. Човек презимена Уајтсајд[14] (енг. *Whiteside*) у дворишту своје куће у Лондону сваке недеље је мерио спољну температуру и количину утрошеног гаса за грејање. Мерења су у току грејних сезона 1960. и 1961. године. У подацима постоји и трећа променљива која је категоричка и говори о томе да ли је дато мерење вршено пре постављања изолације на кућу, или после. Учитајмо податке.

```
library(MASS)
podaci <- MASS::whiteside
```

Како су ови подаци комплетни (господин W. је био ревносан), то ћемо формално обрисати количину потрошеног гаса у опсервацији 47. Довољна нам је једна некомплетна опсервација да покажемо шта желимо, а више њих би само оптеретило запис и оштетило концентрацију.

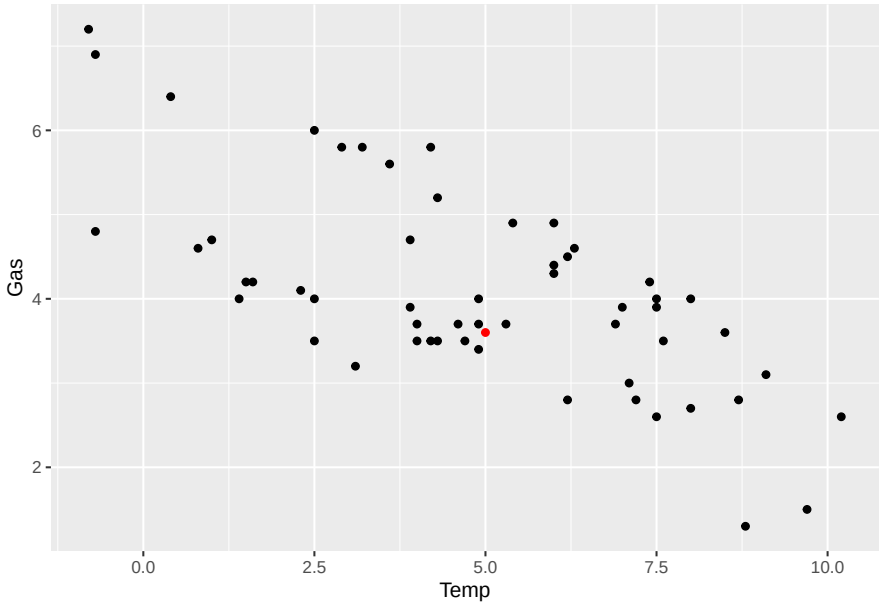
```
podaci_NA <- podaci
podaci_NA$Gas[47] <- NA
```

За почетак нацртајмо податке, без категоричке променљиве.

```
library(ggplot2)

ggplot(podaci_NA, aes(x = Temp, y = Gas))+
  geom_point()+
  geom_point(data = podaci[47, ], color = "red")
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



[14] Ја сам присталица руског начина транскрипције страних имена, то јест да се очувава изговор имена, а не сличност записа. Слично бих записао и *Уолт Уитман* уместо *Волт Витман*.

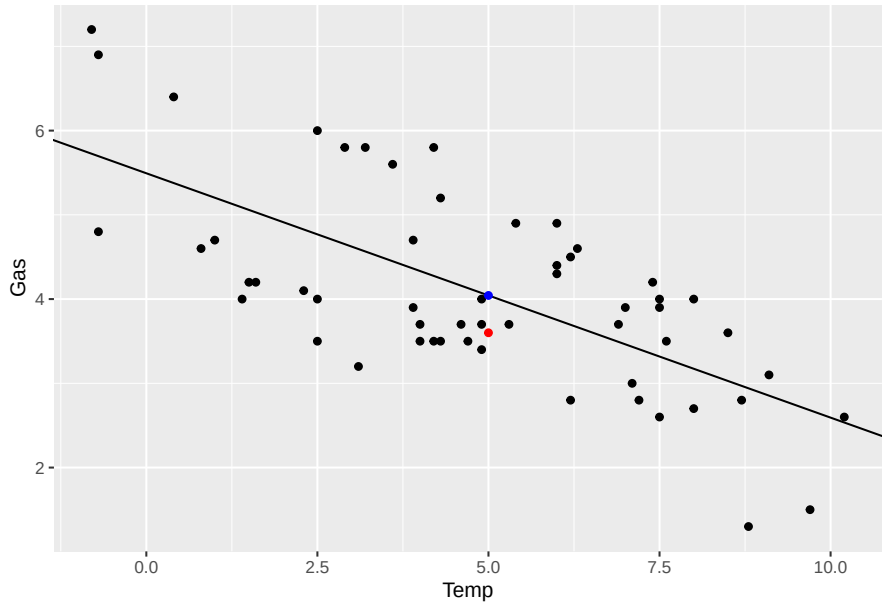
7.7.1.1 Регресиона права

Оно што нам прво пада на памет јесте да на основу доступних опсервација правимо линеарни модел `Gas` $\sim$ `Temp`, те да вредност те праве у 47. тачки узмемо као импутациону.

```
model <- lm(Gas ~ Temp, data = podaci_NA)
koef <- coef(model)
imput <- data.frame(Temp = podaci$Temp[47],
                    Insul = podaci$Insul[47],
                    Gas = koef[1] + koef[2] * podaci$Temp[47])

ggplot(podaci_NA, aes(x = Temp, y = Gas))+
  geom_point()+
  geom_point(data = podaci[47, ], color = "red")+
  geom_abline(intercept = koef[1], slope = koef[2])+
  geom_point(data = imput, color = "blue")
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



Овај начин импутације може се учинити „најсмисленијим'' и „најправилнијим'', а заправо не може бити даље од тога. Под претпоставком да важи линеарни модел, ова импутирана вредност (на слици плаво), јесте најбоља у средњеквадратном смислу. Међутим, она ни најмање не осликава несигурност коју имамо о недостајућој вредности, те не може бити коришћена као импутациона за вишеструку импутацију. Просто - функција у једној тачки не може узети више од једне вредности - па шта онда узети за импутационе вредности?

Може се учинити да овај метод добро „ради'' за једноструку импутацију, нарочито када је линеарна веза више него уочљива. Међутим, и ту треба бити опрезан. Уколико импутирамо вредношћу са регресионе праве, нарочито ако имамо велики број недостајућих вредности, ми **вештачки повећавамо линеарну везу међу предикторима**. То може значајно пореметити резултате стандардизованих тестова, те показати да су неки предиктори безначајни у присуству других (јер су јако корелисани), иако то нису.

7.7.1.2 Регресиона права + бутстреп + шум

Следећа идеја која нам може пасти на памет јесте да не импутирамо баш вредношћу са регресионе праве, већ да на ту вредност додамо неки случајан шум. Идеално би било да знамо расподелу података око регресионе праве, али ми то наравно не знамо. Та расподела може да се оцени, те да се из ње извуче m вредности које се користе као импутационе.

Изложена идеја је боља, али не сасвим добра. Да смо имали неки други узорак извучен из популације, оцене регресионих коефицијената и стандардне девијације грешака биле би различите. Што је мањи обим узорка, различитост је уочљивија. Стога на неки начин треба и ту варијабилност регресионих параметара укључити у импутациони механизам. Ту постоје два приступа:

1. *Бајесовски.* Извлачи параметре директно из њихове апостериорне расподеле.

2. *Бутстреп (Bootstrap).* Вуче узорке из доступних података и за сваки рачуна коефицијенте.

За оба случаја вучемо m параметара/узорака, и онда на основу њих импутирамо по принципу регресиона права + шум, с тим што се сада праве разликују, а и дисперзија шума јер је различито оцењена (у 2. случају).

Напомена. Често се, уместо из претпостављене (нормалне) расподеле грешака са оцењеном дисперзијом, шум извлачи узорковањем из резидуала модела. Наравно, претпостављамо да смо извршили потребне трансформације које гарантују барем приближну адекватност линеарног модела.

7.7.1.3 Регресиона права + још предиктора + бутстреп + шум

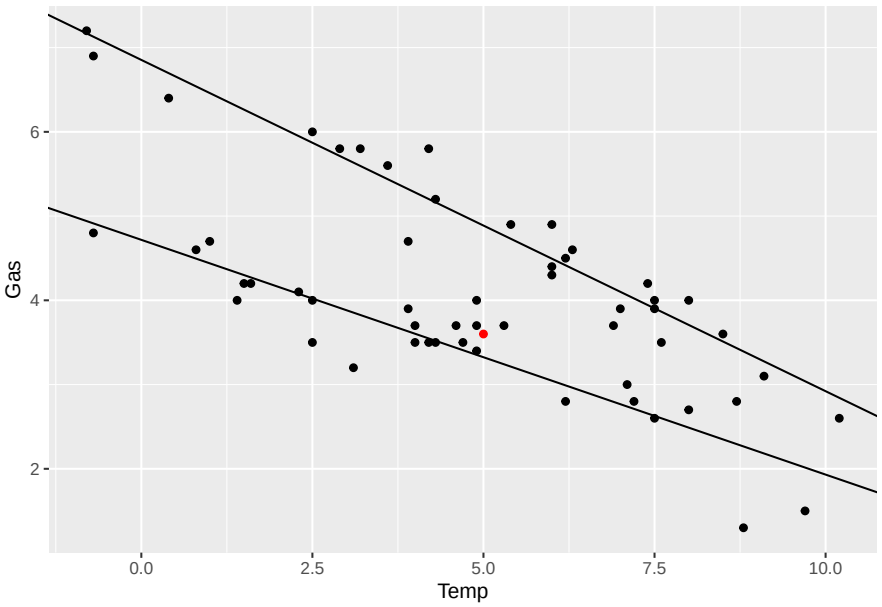
Следећа финеса коју можемо да додамо јесте још један предиктор, а да претходну причу поновимо са њим. Код нас је уочљиво да права `Gas` $\sim$ `Temp` варира у зависности од категорије променљиве `Insul`, што можемо да видимо и са слике.

```
podaci_NA_before <- podaci_NA[podaci_NA$Insul == "Before", ]
model_before <- lm(Gas ~ Temp, data = podaci_NA_before)
koef_before <- coef(model_before)

podaci_NA_after <- podaci_NA[podaci_NA$Insul == "After", ]
model_after <- lm(Gas ~ Temp, data = podaci_NA_after)
koef_after <- coef(model_after)

ggplot(podaci_NA, aes(x = Temp, y = Gas))+
  geom_point()+
  geom_point(data = podaci[47, ], color = "red")+
  geom_abline(intercept = koef_before[1], slope = koef_before[2])+
  geom_abline(intercept = koef_after[1], slope = koef_after[2])
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



Сада се може прво проверити (наравно ако је могуће) у којој је категорији недостајућа опсервација, те, на основу регресионе праве настале из података те категорије, генерисати m импутационих вредности по принципу права + бутстреп (Бајес) + шум.

7.7.1.4 Извлачење из доступних података

Прво провлачимо регресиону праву добијену на основу доступних података, и рачунамо регресионо предвиђање за недостајућу вредност. Затим извршимо предвиђање и за доступне вредности. Бирамо d доступних опсервација таквих да су њихове предвиђене вредности најближе предвиђеној вредности за недостајућу. На случајан начин бирамо једну од тих d и њено поље од интереса импутирамо на упражњено место.

Овај метод је на енглеском језику познат под називом ***Predictive mean matching*** и ми ћемо га у овом раду детаљније анализирати, у случају да више предиктора има недостајуће вредности.

7.7.2 Општи случај

Претходно поглавље посветили смо интуитивном увођењу алгоритма, тако што смо поново прошли кроз мисаони процес који је довео до његовог открића. Сада ћемо мало детаљније изложити алгоритам, испричати у којим се ситуацијама користи, а у којим не, те такође које су му предности и мане.

Име алгоритма, *Predictive mean matching*, предложио је Родерик Ј. А. Литл 1988. године ([7]), а оригинална идеја алгоритма је Рубинова из 1986.

За почетак, претпоставићемо да само једна променљива има недостајуће податке, а да су остале комплетне. Касније ћемо објаснити како се алгоритам може уопштити на податке који имају недостајања у више колона. Означимо променљиву (колону) која има недостајуће податке са Y, а остале, узете заједно као матрицу, означимо са X. Уколико је n обим узорка који имамо, са n_1 ћемо означити број доступних поља у Y, а са X_{obs} ћемо означити оне врсте у X које имају исти индекс као доступне у Y. Аналогно уводимо и n_0 као број недостајућих опсервација у Y и њега прати X_{miss}.

Елем, импутација PMM методом у општем случају спроводи се тако што се прави линеарни модел $Y_{obs} \sim X_{obs}$, те се и за доступне и за недоступне опсервације предвиђају вредности за Y у одговарајућим вредностима предиктора. Затим се, за предвиђену вредност недостајућег поља тражи d најближих. Ово „најближих'' може да се схвати на разне начине, а све у зависности од тога коју смо метрику одабрали.

Литл је у [7] предлагао, између осталог, Махалонобисово растојање[15]. Ипак, пракса је показала да се компликовањем не добијају исувише бољи резултати, па је у функцију `mice.impute.pmm` у пакету `mice` ушла стандардна апсолутна вредност, као једина метрика (не може да се бира).

Тих d најближих опсервација називамо *кандидатима за донора*. Дакле, свако недостајуће поље има својих d кандидата за донора. Затим се на случајан начин од њих бира један, чије се одговарајуће поље узима и импутира на упражњено место. Тај један назива се донор. Прича се понавља за свако недостајуће поље.

Напомена. Кандидати за донора се могу бирати и на друге начине, али ми смо посебно изложили онај који је рачунарски имплементиран. Андриц и Литл ([17]) разликују 4 начина за избор кандидата за донора (па и самог донора) за индекс j:

1. Бирамо неки праг (трешхолд) η и узимамо за доноре све i за које је $|\widehat{y}_i - \widehat{y}_j| < \eta$. Подразумевамо да је $|\cdot|$ она метрика која је одабрана. Кандидати су сви они који задовољавају једнакост, а од њих се на случајан начин бира донор.

2. Бирамо d најближих, и међу њима донора на случајан начин. Ово је имплементирано у `R`-у. Подразумевано је $d = 5$.

3. Као специјални случај претходног, бирамо један најближи. Јасно, он одмах постаје донор. Ово је познато под називом *hot deck* импутација, тако да је она специјални случај PMM импутације.

4. Бирамо само једног донора, дакле без кандидата, али тако што вероватноћа избора зависи од $|\widehat{y}_i - \widehat{y}_j|$. За ово је неопходно претпоставити неку расподелу.

Поред тога што смо видели како се све може одабрати донор (и који је начин имплементиран у `R`-у), корисно је погледати и на који се све начин може вршити упаривање оцена за доступне и недоступне податке, где под упаривањем мислимо на то како бирамо коефицијенте за предвиђање на доступним, а како на недоступним подацима.[16] И овде се могу разликовати 4 типа упаривања:

1. **Упаривање типа 0.** За доступне: $\hat{Y} = X_{obs}\hat{\beta}$; за недоступне: $\hat{Y} = X_{miss}\hat{\beta}$, где је $\hat{\beta}$ оцена у моделу $Y_{obs} \sim X_{obs}$.

2. **Упаривање типа 1.** За доступне: $\hat{Y} = X_{obs}\hat{\beta}$; за недоступне: $\dot{Y} = X_{miss}\dot{\beta}$, где је $\dot{\beta}$ случајно одабрана вредност параметра из апостериорне расподеле, под условом доступних података. Алтернатива овом бајесовском приступу јесте бутстреп.

3. **Упаривање типа 2.** За доступне: $\dot{Y} = X_{obs}\dot{\beta}$; за недоступне: $\dot{Y} = X_{miss}\dot{\beta}$.

4. **Упаривање типа 3.** За доступне: $\dot{Y} = X_{obs}\dot{\beta}$; за недоступне: $\dot{Y} = X_{miss}\ddot{\beta}$. Овде се врше два извлачења коефицијената из апостериорне расподеле, једно за доноре и једно за примаоце.

Упаривање типа 0 занемарује варијабилност оцене $\hat{\beta}$ за различите узорке и води ка лошим статистичким закључцима ([18]). Тип 2 покушава да реши овај проблем, мада може направити нови проблем за мали број предиктора, јер ће пречесто да фаворизује исте доноре. Тип 1 представља компромис претходна 2, и он је подразумеван у функцији `mice.impute.pmm`. Други се могу задати променом аргумента `matchtype`.

7.7.2.1 Предности и мане

Основна предност PMM алгоритма јесте та што на известан начин „штити'' механизам од потенцијалне грешке модела. Ако линеарни модел није баш најаднекватнији, ако је „довољно близу'' кандидати за донора ће бити приближно исти. Стога, за разлику од чисто параметарских алгоритама, PMM може да

[15] За дефиницију Махалонобисовог растојања може се погледати [22].

[16] Реч „упаривање" свакако није најадекватнија, али смо је овде задржали да би пратила широко коришћен енглески термин *matching*.

толерише благе грешке у моделовању зависности међу самим подацима. Такође, никада нећемо добити немогуће вредности као кандидата за импутационе.

Основна мана јесте што се у појединим ситуацијама донори понављају за различите недостајуће опсервације, што може смањити варијабилност унутар података и пореметити већину тестова. Такође, PMM треба избегавати када је број предиктора мали.

7.7.3 Алгоритам

Након што смо идејно објаснили како PMM функционише, ред је да дамо и формалан опис алгоритма. Алгоритам је следећи:

1. Срачунати производ $S = X_{obs}^T X_{obs}$.
2. Срачунати $V = (S + \kappa \mathbf{diag}(S))^{-1}$, за неко мало κ. Његова улога је да отклони потенцијалну мултиколинеарност (чија је последица то да нема инверз).
3. Спроводимо назубљену (ridge) линеарну регресију $Y_{obs} \sim X_{obs}$. Рачунамо регресионе коефицијенте $\hat{\beta} = V X_{obs}^T Y_{obs}$.
4. Нека је q број колона у X_{obs}. Извлачимо $\dot{g} \sim \chi^2_\nu$, где је $\nu = n_1 - q$.
5. Рачунамо $\dot{\sigma}^2 = (Y_{obs} - X_{obs}\hat{\beta})^T (Y_{obs} - X_{obs}\hat{\beta})/\dot{g}$.
6. Вучемо q независних вредности из нормалне $\mathcal{N}(0,1)$ расподеле и смештамо их у вектор $\dot{z}_1$.
7. Рачунамо $V^{1/2}$ преко Чолески декомпозиције.
8. Рачунамо $\dot{\beta} = \hat{\beta} + \dot{\sigma}\dot{z}_1 V^{1/2}$.
9. Рачунамо $\dot{\eta}(i,j) = |X_{obs,i\to}\hat{\beta} - X_{miss,j\to}\dot{\beta}|$, за све одговарајуће i, j.
10. Конструишемо n_0 вектора Z_j, при чему сваки садржи d „донора кандидата" из Y_{obs}, тако да је $\sum_d \dot{\eta}(i,j)$ минимално.
11. Из сваког Z_j бирамо донора i_j на случајан начин.
12. Импутирамо $y_j = y_{i_j}$, за свако $j \in \{1, 2, \ldots, n_0\}$.

7.7.3.1 Вишеструка импутација

Вишеструка импутација се добија тако што се део алгоритма од 4. корака па до краја понови m пута, те поступи по Рубиновим правилима.

7.7.4 Недостајања у више колона

До сада смо претпостављали да су наши подаци облика (Y, X) и да недостајања постоје само у Y. Сада треба пронаћи начин како да алгоритам уопштимо и на случајеве када недостајања постоје у више колона. Претпоставимо, зато, да су наши подаци облика (Y, X), али да је Y вишедимензион. Слично као и раније, вршимо линеарну регресију $Y_{obs} \sim X_{obs}$, а даље тражење донора настављамо као и малопре ([7]).

Напомена. Што је више колона у Y, а мање у X, резултати су (наравно) лошији, те се овај алгоритам и не препоручује у таквим ситуацијама. Он ће потпуно „пући" ако свака колона у подацима има барем једну недостајућу опсервацију. Тада треба посегнути за неким другим алгоритмом. Када би постојао савршен механизам импутације, онда би постојао тачно један механизам импутације.

7.8 PMM - Провера на подацима

7.8.1 Генерисање недостајања

Генерисање недостајања по MCAR принципу не би требало да представља велики проблем. Сетимо се да је у MCAR случају расподела матрице R независна и од недостајућих података и од доступних података. Сходно томе, довољно је генерисати примерак матрице R из одговарајуће расподеле (која се, поново, бира емпиријски, на основу додатних сазнања) и онде где се реализовала нула уписати `NA`. За велику већину расподела програмски језик `R` већ има уграђене функције за генерисање.

Међутим, генерисање MAR података неће ићи тако лако. Основни проблем лежи у томе што у MAR случају расподела матрице R зависи не само од унутрашњих параметара, већ и од доступних нам података. Та зависност може се остваривати на бесконачно много начина, а ми ћемо овде изложити неке од њих. За детаљније разматрање генерисања недостајућих података вреди погледати [18].

7.8.1.1 Показни пример

Као и обично, најбоље се види на примеру. Кренућемо од најједноставнијег случаја. Нека се наша дизајн матрица $X = (X_1, X_2)$ састоји од 1 000 узорака из дводимензионе нормалне расподеле $\mathcal{N}\left([3\ 3], \begin{bmatrix} 1 & 0.6 \\ 0.6 & 1 \end{bmatrix}\right)$.

```
set.seed(123)
n <- 1000
covMat_X1X2 <- matrix(c(1, 0.6,
                        0.6, 1),
                      nrow = 2)
library(MASS)
X <- mvrnorm(n, mu = c(3, 3), Sigma = covMat_X1X2)
```

За почетак ћемо генерисати недостајања **само у колони** X_2 по MAR принципу, и даћемо им да зависе од вредности у колони X_1. То се може урадити на више начина, а ми ћемо представити три која су у пракси од значаја. Нека је R_2 колона-индикатор недостајања у X_2 (у аналогији с матрицом R коју смо на почетку помињали). Дајемо три модела:

1. **MARRIGHT**: $\text{logit}(\mathbf{P}\{R_2 = 0\}) = X_1 - 3$
2. **MARMID**: $\text{logit}(\mathbf{P}\{R_2 = 0\}) = 0.75 - |X_1 - 3|$
3. **MARTAIL**: $\text{logit}(\mathbf{P}\{R_2 = 0\}) = |X_1 - 3| - 0.75$

Напомена. Бројеви су одабрани да би неке особине касније биле боље уочљиве на сликама - нису од пресудне важности.

Имена која смо им доделили ускоро ће постати јасна, а сада их прво применимо на нашим подацима. У `r-base` пакету `stats` дате су функције за рад са логистичком расподелом. `plogis` представља функцију расподеле вероватноћа логистичке расподеле, а она је ништа друго до инверз logit функције. Радимо прво са MARRIGHT моделом.

```
prob_R2jeNula_MARRIGHT <- plogis(X[, 1] -3)
prob_R2jeJedan_MARRIGHT <- 1 - prob_R2jeNula_MARRIGHT
```

У вектору `prob_R2jeJedan_MARRIGHT` су вероватноће да је одговарајуће поље присутно, тј. да није недостајуће. То нам одговара јер ћемо у зависности од тих вероватноћа да генеришемо вектор R_2. Урадимо то.

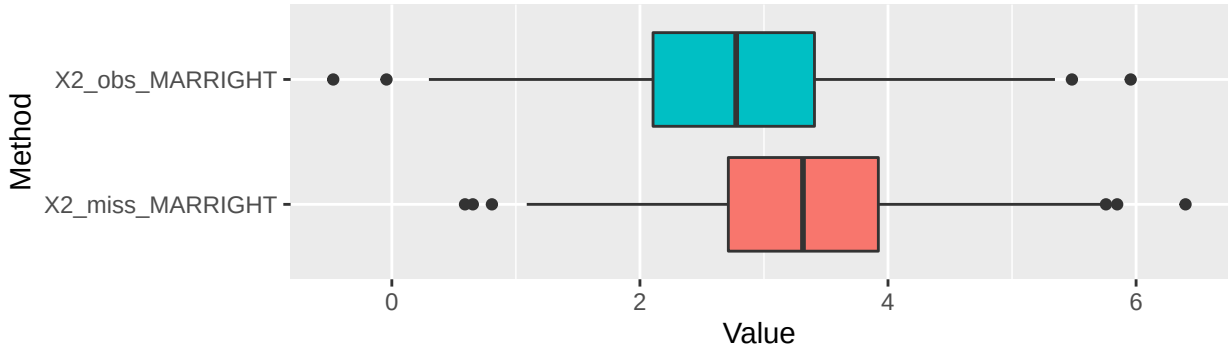
```
R2_MARRIGHT <- rbinom(n, 1, prob_R2jeJedan_MARRIGHT)
```

Сада ћемо раздвојити недостајуће и доступне податке.

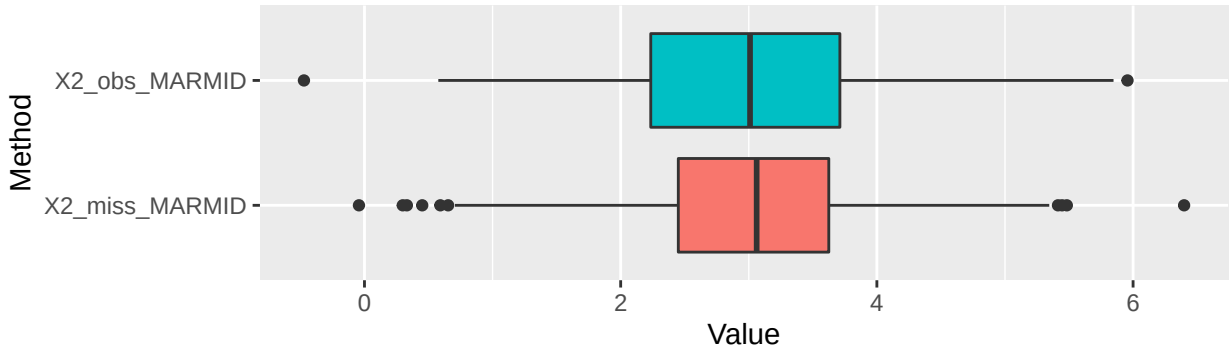
```
X2 <- X[, 2]
# kolona dostupnih vrednosti
X2_obs_MARRIGHT <- X2[R2_MARRIGHT == 1]
# kolona nedostajucih vrednosti
X2_miss_MARRIGHT <- X2[R2_MARRIGHT == 0]
```

На потпуно аналоган начин радимо и за MARMID и MARTAIL, разликује се само аргумент у првом реду.

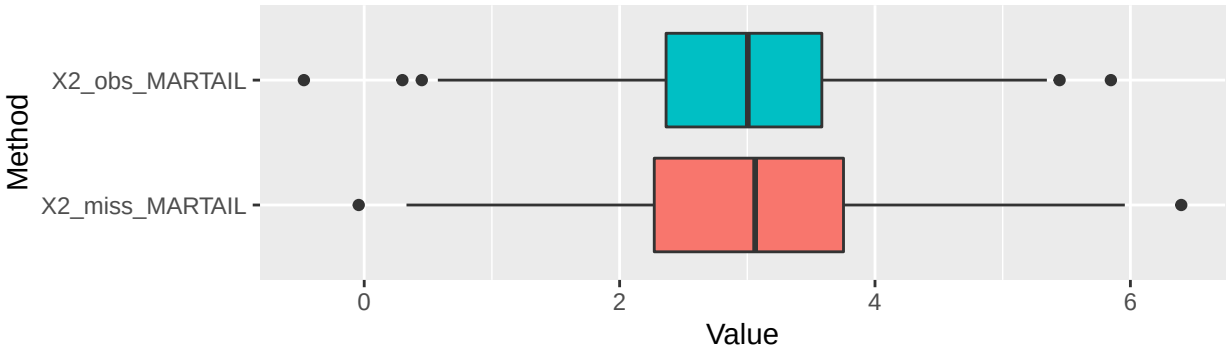
Сада ћемо нацртати боксплотове за сваки од претходних парова недостајућих и доступних података и постаће нам јасно откуд назив. На слици испод видимо боксплотове за доступне и недостајуће податке MARRIGHT модела. Јасно је да он „одузима'' више података за више вредности поља, тако да се вектор доступних вредности помера улево.



На следећој слици видимо боксплотове за MARMID модел. Он узима више података из околине средње вредности, тако да средња вредност остаје иста, али дисперзија доступних података расте.



Код MARTAIL модела видимо супротну ствар - одузима се више података далеких од средње вредности, тако да дисперзија оног што је остало опада.



Сваки од претходна три модела даје нам апроксимативно 50% недостајућих података, али сваки на свој начин. Ови модели недостајања су прилично екстремни и ретко се срећу на реалним подацима. Не само да имамо јаку везу између R_2 и X_1, него је и удео недостајућих података нереално висок за скоро па све реалне ситуације. Ипак, наравно да их нисмо узалуд наводили. Уколико се неки метод импутације добро покаже на овако екстремно оштећеним подацима, разумно је дати себи за право помислити да је тај метод квалитетан и да ће се понашати добро и у мање екстремним ситуацијама.

Са друге стране, прилично смо се ограничили. Имамо недостајање података само у једној колони, а уколико желимо недостајања и у другој, можемо то урадити само за оне индексе за које је у претходној податак доступан. Другим речима, за сада не умемо да генеришемо више недостајућих поља у врсти. Такође, не можемо директно да утичемо на проценат недостајућих података, а било би нам згодно када бисмо га могли проследити некој функцији као аргумент. Ствар се додатно компликује када у причу укључимо произвољан број колона. Срећом, и за то постоји решење, које ћемо дати у следећем пододељку.

7.8.1.2 Уопштење

Бранд је 1999. године у [10] је развио метод за генерисање MAR података за дизајн матрицу са p колона $X_1, X_2, \dots, X_p$ који ћемо ми сада изложити. Претпостављамо да је $X = (X_1, X_2, \dots, ..., X_p)$ на почетку комплетна. Овај метод захтева задавање следећих ствари:

- α, жељени удео недостајућих података;
- R_{pat}, бинарна матрица димензија $n_{pat} \times p$ која дефинише n_{pat} дозвољених шаблона по којима недостају подаци у врстама; дозвољени су сви осим свих нула и свих јединица;
- $f = (f_{(1)}, \dots, f_{(n_{pat})})$, вектор који садржи релативне френвенције појављивања сваког од шаблона;
- $\mathbf{P}\{R \mid X\} = \left(\mathbf{P}\{R_{(1)} \mid X_{(1)}\}, \dots, \mathbf{P}\{R_{(n_{pat})} \mid X_{(n_{pat})}\}\right)$, низ од n_{pat} модела недостајања, сваки за по један шаблон.

Напомена. У пакету `mice` постоји функција `ampute` која имплементира три горе наведена начина недостајања, као и MARLEFT у аналогији са MARRIGHT. Њен аргумент `type` може узети четири вредности: `"MID", "TAIL", "LEFT", "RIGHT"`. Подразумевана пропорција недостајања је 50%, а може се мењати аргументом `prop`.

7.8.2 Поређење са регресијом

Прави је моменат да направимо кратку рекапитулацију свега што смо до сада обрадили. Дефинисали смо типове недостајања података, детаљно смо прошли кроз идеју вишеструке импутације, те затим цело једно поглавље посветили PMM алгоритму. Објаснили смо зашто је он бољи од неких класичних метода, на пример регресије. Сада ћемо целу причу илустровати једним примером.

Треба нагласити да се од вишеструке импутације углавном не очекује да да најближе оцене недостајућих поља. То је најбоље радити регресијом, или неком другом параметарском методом, јер је познато да су те оцене, под претпоставком линеарног модела, најбоље у средњеквадратном смислу. Од вишеструке импутације се првенствено очекује да не наруши структуру података: варијабилност, предвидивост (тј. понављање шаблона унутар података) итд.

Генерисаћемо један вештачки пример, вишедимензиону нормалну расподелу димензије 6, где недостајања постоје у прве две колоне.

```
data <- mvrnorm(n = 1000,
                mu = c(0, 1, 2, 3, 4, 5),
                Sigma = diag(1, 6))
data <- data.frame(data)
colnames(data) <- c("V0", "V1", "V2", "V3", "V4", "V5")
```

Видимо да смо податке генерисали тако да је коваријациона матрица дијагонална, односно да су колоне независне. То смо урадили да би чињеница да регресија нарушава структуру података била још уочљивија. Генерисаћемо MAR недостајања пропорције 40%, по TAIL принципу. Слични резултати се добијају и за друге типове и пропорције недостајања, што је лако проверљиво, те нећемо оптерећивати запис.

```
data_Y <- data[, 1:2]

data_Y_amputed <- ampute(data_Y, prop = 0.4, mech = "MAR", type = "TAIL")$amp

data_incompl <- cbind(data_Y_amputed, data[, 3:6])
colnames(data_incompl) <- c("V0", "V1", "V2", "V3", "V4", "V5")

data_imput_pmm <- complete(mice(data_incompl)) # PMM je default
```

Сада ћемо исто урадити и за регресиону импутацију.

```
data_imput_reg <- complete(mice(data_incompl, method = "norm.predict"))
```

Сада ћемо цртати зависност колона `V0` и `V2` за оба метода. Бирамо једну колону која је имала недостајања и једну која није, јер је свеједно пошто су независне. Да бисмо импутиране вредности обојили различитом бојом од доступних, имаћемо мало физичког посла.

Прво ћемо из колоне `V0` у оштећеним подацима извући индикатор недостајања.

```
ind_fali <- is.na(data_incompl$V0)
ind_ne_fali <- as.logical(rep(1, length(data_incompl$V0)) - ind_fali)
```

Сада из импутираних база издвајамо посебно врсте које су биле комплетне, а посебно оне које су импутиране.

```
V0_V2_reg_obs <- data_imput_reg[ind_ne_fali, c(1, 3)]
V0_V2_pmm_obs <- data_imput_pmm[ind_ne_fali, c(1, 3)]

V0_V2_reg_imp <- data_imput_reg[ind_fali, c(1, 3)]
V0_V2_pmm_imp <- data_imput_pmm[ind_fali, c(1, 3)]
```

Сада смо спремни да цртамо. Плавом бојом означавамо импутиране вредности, а црном доступне.

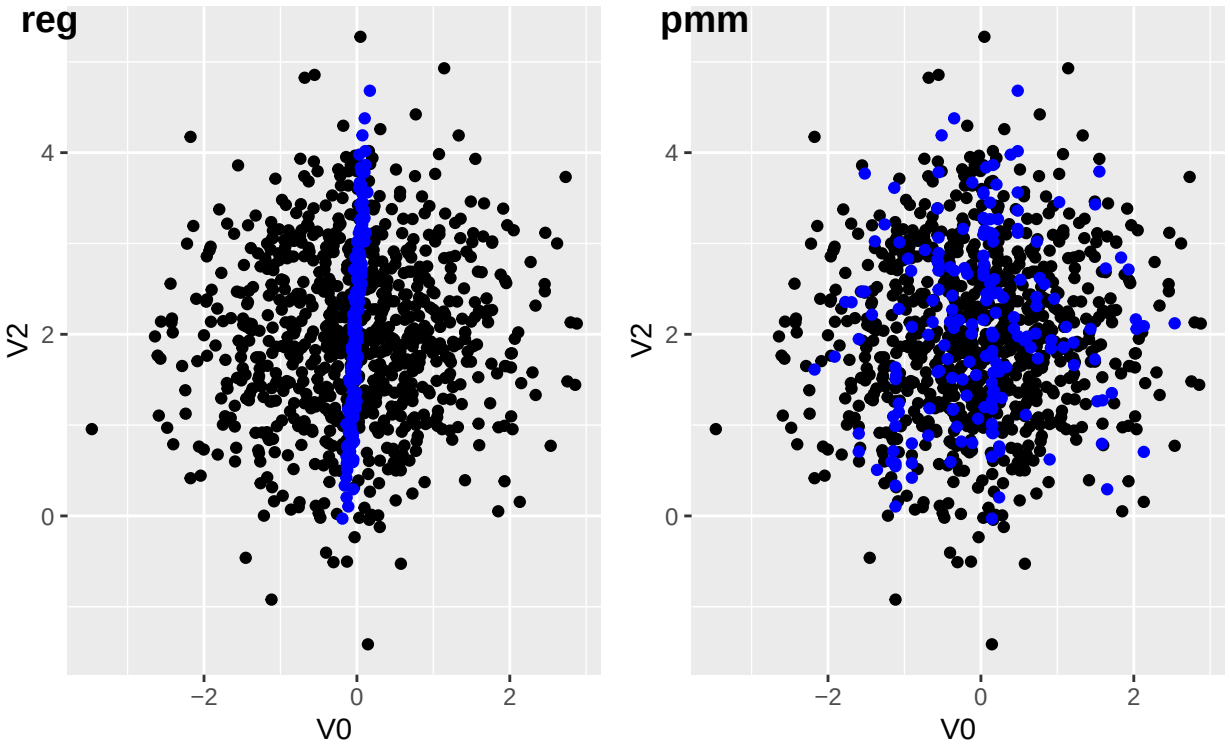
```
library(cowplot)

plot1 <- ggplot() +
```

```
  geom_point(data = V0_V2_reg_obs,
             mapping = aes(x = V0, y = V2)) +
  geom_point(data = V0_V2_reg_imp,
             mapping = aes(x = V0, y = V2),
             color = "blue")

plot2 <- ggplot() +
  geom_point(data = V0_V2_pmm_obs,
             mapping = aes(x = V0, y = V2)) +
  geom_point(data = V0_V2_pmm_imp,
             mapping = aes(x = V0, y = V2),
             color = "blue")

cowplot::plot_grid(plot1, plot2, labels = c("reg", "pmm"))
```



Више је него очигледно колико *Predictive mean matching* више очувава структуру података од класичне регресије (сама регресија и нема баш смисла за независне колоне, овде је узета чисто показно). Намерно смо одабрали независне колоне да би то било што уочљивије, а, наравно, исто важи и за зависне колоне. Ко жели и у то да се увери, може погледати [18], а можда и боље, поиграти се сам.

„Нарушавање структуре података'' може се мерити и разним иформационим показатељима, од којих је у последње време популаран `ApEn` (approximate entropy), о којему се може прочитати у [19]. На српском језику о овом показатељу може се прочитати у [23], а и видети његова примена на анализу временских серија.