

Nina Ružič Gorenjec

Grafični prikazi učinkov v regresijskih modelih

Povzetek. Pri statističnih analizah podatkov je pogosta uporaba regresijskih modelov z več neodvisnimi spremenljivkami, v katere nekatere spremenljivke vključimo z nelinearnimi učinki in dodamo interakcije med nekaterimi neodvisnimi spremenljivkami. Teh učinkov praviloma ni mogoče povzeti številčno na intuitiven način, zato so grafični prikazi nujni za njihovo povedno predstavitev v poročilih ali znanstvenih člankih. V uporabljenem regresijskem modelu ocenimo napovedane vrednosti izida ali njegove pretvorjene različice in jih predstavimo na grafikonu skupaj z intervali zaupanja. Neodvisne spremenljivke, katerih učinek predstavljamo, pri tem variirajo, preostalem spremenljivkam v multiplu regresijskem modelu pa določimo izbrane konstantne vrednosti. V članku predstavimo, kako vrednosti preostalih spremenljivk v modelu vplivajo na širino intervalov zaupanja za napovedane vrednosti izida, zaradi česar grafično presojanje statistične značilnosti nelinearnosti učinka na podlagi intervalov zaupanja v multipli regresiji ni verodostojno. Prikažemo tudi, da je risanje intervalov zaupanja pri grafičnih predstavitvah interakcij med neodvisnimi spremenljivkami lahko zavajajoče, saj nas njihovo (ne)prekrivanje lahko vodi v zmotno prepričanje o statistični (ne)značilnosti interakcije. V drugem delu članka na kratko predstavimo knjižnico `rms` iz programa R, s katero lahko učinke v raznovrstnih regresijskih modelih enostavno prikažemo. Izvirna koda za tvorbo podatkov in vseh grafičnih prikazov iz članka je prosto dostopna na spletu.

Graphical Displays of Effects in Regression Models

Abstract. Regression models with multiple independent variables are frequently used in statistical data analysis. Independent variables are often included with non-linear effects, and interactions between independent variables are added to the model. Because it is usually not possible to present these effects numerically in an intuitive way, graphical presentations are necessary to properly illustrate these effects in reports and scientific papers. For a chosen regression model, we estimate the predicted value of the outcome or its transformed version and plot it together with the corresponding confidence intervals. Independent variables that are included in the effect that we are presenting vary, while all other variables in the model are fixed to certain values. In the paper, we present how the values for which the other variables are adjusted influence the width of the confidence intervals for predicted values of the outcome, implying that graphical inspection of statistical significance of non-linearity of an effect based on confidence intervals is not sensible. Furthermore, we show that confidence intervals in graphical presentations of interactions between independent variables can be misleading, since the (lack of) intersection between them can give wrong intuition about the (lack of) statistical significance of the interaction. In the second part of the paper, we concisely present the `rms` library from R software, which is very useful for graphical presentations of effects in various regression models. Source code for data simulation and all graphical presentations from the paper is freely available online.

■ **Infor Med Slov** 2020; 25(1-2): 19-24

Institucija avtorice / Author's institution: Medicinska fakulteta, Univerza v Ljubljani.

Kontaktna oseba / Contact person: asist. dr. Nina Ružič Gorenjec, Inštitut za biostatistiko in medicinsko informatiko, Medicinska fakulteta, Univerza v Ljubljani, Vrazov trg 2, 1000 Ljubljana, Slovenija. E-pošta / E-mail: nina.ruzic.gorenjec@mf.uni-lj.si.

Prispelo / Received: 17. 7. 2020. Sprejeto / Accepted: 26. 11. 2020.

Uvod

Pri statističnih analizah podatkov je za preverjanje raziskovalnih domnev pogosto potrebna uporaba regresijski modelov,¹ ki omogočajo hkratno preučevanje povezanosti več pojasnjevalnih spremenljivk z izidom v enem modelu. Še posebej v medicini, ki preučuje kompleksne pojave, so v raziskavah in znanstvenih člankih velikokrat uporabljeni regresijski modeli (linearna regresija, logistična regresija in preostali posplošeni linearni modeli, mešani modeli, nelinearni modeli, Coxov regresijski model idr.).

Med analizo prileganja modela se pogosto izkaže, da je potrebno nekatere spremenljivke vključiti v model nelinearno (na primer v obliki kvadratne funkcije ali kubičnega zleпка z omejitvami, angl. *restricted cubic spline*) in da je potrebno dodati interakcije med spremenljivkami, ali pa imamo vsebinske razloge za vključitev takšnih učinkov. Omenjenih učinkov ni mogoče predstaviti številčno na intuitiven način (če niso vse vpletene spremenljivke opisne z malo kategorijami), zato so za njihovo povedno predstavitev nujni grafični prikazi. Z regresijskim modelom ocenimo napovedane vrednosti izida ali njegove pretvorjene različice (npr. v logistični regresiji raje predstavimo verjetnost dogodka kot logaritem obetov za dogodek) in jih predstavimo na sliki skupaj z intervali zaupanja. Neodvisne spremenljivke, katerih učinek predstavljamo, pri tem variirajo, preostalem spremenljivkam v modelu multiple regresije pa določimo konstantne vrednosti.

V članku bomo predstavili, kako vrednosti preostalih spremenljivk v modelu vplivajo na širino intervalov zaupanja za napovedane vrednosti izida in na kaj moramo biti pozorni pri njihovi interpretaciji. Prikazali bomo tudi, zakaj je risanje intervalov zaupanja pri grafičnih predstavitvah interakcij med neodvisnimi spremenljivkami lahko zavajajoče.

V drugem delu članka bomo na kratek in preprost način predstavili knjižnico `rms` (avtorja Franka E. Harella)² iz programa R,³ s katero lahko učinke v raznovrstnih regresijskih modelih enostavno prikažemo. Programska koda za tvorbo podatkov in izdelavo vseh grafičnih prikazov iz članka je prosto dostopna na spletu.⁴

Pasti risanja intervalov zaupanja za napovedane vrednosti izida

Skozi članek bomo prikazali nekaj linearnih in logističnih regresijskih modelov; pri vseh bomo uporabljali iste neodvisne spremenljivke, ki jih bomo simulirali na naslednji način. Za 200 enot bomo simulirali X_1 in X_2 iz standardne normalne porazdelitve, Z_1 in Z_2 pa iz Bernoullijeve porazdelitve s parametrom 0,2, kjer bomo Z_2 premaknili za 1, tako da bo zavzemala vrednosti 1 in 2 (zgolj zaradi hitrejšega razločevanja med dvojiškima spremenljivkama na grafikonih). Za linearne modele bomo napako ε simulirali iz normalne porazdelitve s povprečjem 0 in standardnim odklonom 3. Rezultati v članku so dobljeni pri uporabi `set.seed(6)` v programu R, verzija 3.6.3.³

Pri vrednotenju rezultatov bo vrednost p statistično značilna, če bo manjša od stopnje tveganja 0,05.

Nelinearni vpliv neodvisne spremenljivke

Za povedno predstavitev nelinearnega vpliva neodvisne spremenljivke na izid je grafični prikaz nujen, pogosto pa nam poleg vrednosti p služi tudi kot dodatno orodje za presojanje statistične značilnosti in strokovne pomembnosti nelinearnosti učinka. Pri slednjem se je potrebno zavedati, da moramo v primeru multiple regresije ob risanju napovedanih vrednosti izida preostale spremenljivke v modelu »fiksirati« (tj. predpisati jim določene vrednosti), kar vpliva na širino intervalov zaupanja in s tem tudi na našo zaznavo pomembnosti učinka. To bomo prikazali s primerom.

Izid naj bo generiran iz linearnega modela

$$Y = 1 + X_1 + X_1^2 + X_2 + Z_1 + Z_2 + 3X_2Z_1 + 3X_2Z_2 + \varepsilon.$$

V modelu imamo torej poleg kvadratnega vpliva številske spremenljivke X_1 še tri druge pojasnjevalne spremenljivke in dve interakciji med njimi, ki bosta služili za predstavitev preprostosti risanja s knjižnico `rms` v programu R v naslednjem razdelku. Zaradi enostavnosti smo si za vrsto nelinearnega učinka izbrali kvadratno funkcijo; namesto te bi lahko uporabili tudi kakšno vrsto zleпка. Generiranim podatkom prilagodimo model multiple linearne regresije, v katerega vključimo X_1 s kvadratnim učinkom, X_2 , Z_1 in Z_2 ter pripadajoči interakciji. Za kvadratni učinek dobimo vrednost $p < 0,0001$.

Na sliki 1 je predstavljena napovedana vrednost izida glede na različne vrednosti X_1 , pri čemer je na levem grafu $Z_1 = 0$ in $Z_2 = 1$, na desnem pa $Z_1 = 1$ in $Z_2 = 2$, medtem ko je X_2 na obeh grafih enak mediani vzorca. Marsikdo (vključno z avtorico tega članka) statistično značilnost nelinearnosti učinka grafično ovrednoti »čez palec« tako, da presodi, ali bi lahko v pas, ki ga omejujejo intervali zaupanja, vrisali premico – če tega ne moremo narediti, potem se nam učinek zdi statistično značilno nelinearen. To je le približna grafična metoda, poleg tega so intervali zaupanja izračunani za vsako točko posebej (celoten pas skupaj torej nima primerne pokritja), zato seveda statistično značilnost presojamo na podlagi vrednosti p . Na levi sliki naša intuicija sovpada z visoko značilno vrednostjo p , medtem ko za desno sliko to ne drži. Spremenljivki Z_1 in Z_2 smo na desni sliki nastavili na vrednosti, ki ju zavzame le 18 % oziroma 19 % enot v naših podatkih.

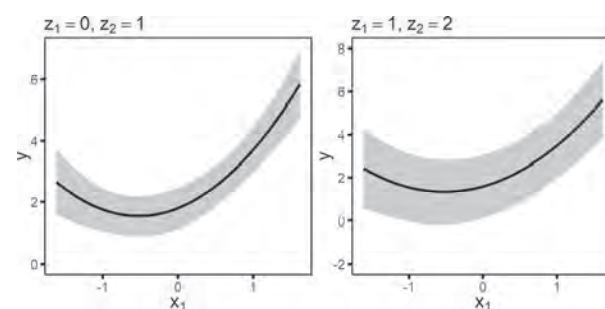
V grafične prikaze nelinearnih učinkov je priporočljivo vključiti intervale zaupanja, saj dodajo informacijo o variabilnosti ocene in omogočajo presojanje strokovne pomembnosti za populacijo. Pri tem pa se je potrebno zavedati, da lahko nastavljene vrednosti preostalih spremenljivk bistveno vplivajo na njihovo širino. Praviloma so intervali zaupanja ožji, če so preostale spremenljivke nastavljene na svoj modus oziroma mediano, ni pa to nujno in zato je v kompleksnejših modelih z veliko spremenljivkami pogosto potrebno vložiti kar nekaj truda za jasno prikazan nelinearen učinek, predhodno nakazan z značilno vrednostjo p .

Interakcija med neodvisnimi spremenljivkami

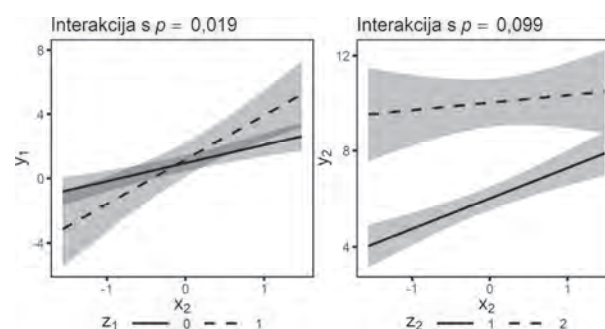
Če imamo v modelu interakcijo med dvema neodvisnima spremenljivkama, je velikost učinka prve neodvisne spremenljivke odvisna od vrednosti druge in obratno. Prisotnost oziroma odsotnost interakcije bomo podrobneje predstavili na dveh primerih, namen pa je preko njiju razmisliti o smiselnosti vključevanja intervalov zaupanja v grafične prikaze interakcij.

V prvem primeru bo izid generiran iz linearnega modela $Y_1 = 1 + X_2 + Z_1 + 2X_2Z_1 + \varepsilon$, v drugem pa iz modela $Y_2 = 1 + X_2 + 5Z_2 + \varepsilon$. V obeh primerih imamo torej v model vključeni dve neodvisni spremenljivki, eno številsko in eno dvojiško, pri čemer je v prvem modelu prisotna interakcija, v drugem pa ne. Tako generiranim podatkom (uporabimo že generirane neodvisne spremenljivke in napako) prilagodimo dva modela multiple linearne regresije, enega za izid Y_1 glede na neodvisni spremenljivki X_2

in Z_1 ter drugega za izid Y_2 glede na spremenljivki X_2 in Z_2 , kjer v obeh modelih dovolimo še interakcijo. V prvem modelu dobimo statistično značilno interakcijo z vrednostjo $p = 0,019$, v drugem modelu pa je vrednost $p = 0,099$ in interakcija torej ni statistično značilna.



Slika 1 Grafični predstavitev nelinearnega učinka v linearnem modelu za Y s 95 % intervali zaupanja za napovedane vrednosti, pri čemer je na levem grafu $Z_1 = 0$ in $Z_2 = 1$, na desnem pa $Z_1 = 1$ in $Z_2 = 2$, medtem ko je X_2 na obeh grafih enak mediani vzorca.



Slika 2 Grafični predstavitev interakcije v linearnem modelu za Y_1 (levo) in Y_2 (desno) s 95 % intervali zaupanja za napovedane vrednosti, kar je lahko za vrednotenje statistične značilnosti interakcije (vrednost p nad grafom) zavajajoče.

Na sliki 2 sta interakciji predstavljene grafično, tj. narisane so napovedane vrednosti izida glede na številsko neodvisno spremenljivko pri različnih vrednostih opisne spremenljivke. Če bi bila tudi druga spremenljivka številsko, potem bi narisali napovedane vrednosti pri nekaterih izbranih vrednostih te spremenljivke. Osredotočimo se najprej le na premice brez intervalov zaupanja. Na levi sliki lahko nazorno vidimo grafično predstavitev interakcije – naklona premic sta različna. Skladno s tem opazimo, da sta si na desni sliki naklona premic bolj podobna, kar sovpada z večjo vrednostjo p – predpostavka modela brez interakcije je vzporednost premic (v praksi na vzorcu seveda praktično nikoli ne dobimo točno vzporednih premic). Pri tem lahko opazimo tudi, da imata premici na desni sliki zelo različno začetno vrednost (presečišče z ordinatno osjo), kar je

posledica dejstva, da ima Z_2 v tem modelu velik učinek – odsotnost učinka spremenljivke in s tem potreba po enostavnejšem modelu bi se kazala v podobnosti premic.

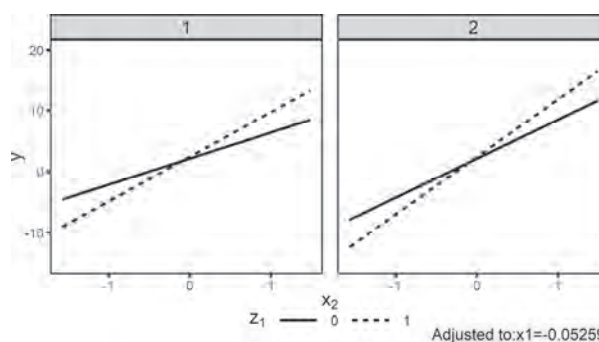
Čeprav je večini bralcev zgoraj napisano popolnoma jasno, pa lahko grafično preučevanje interakcij ob narisanih intervalih zaupanja marsikoga hitro zavede. Ob prvem pogledu na levi graf na sliki 3 bi namreč lahko zmotno pomislili, da interakcija zaradi prekrivajočih se intervalov zaupanja ni statistično značilna, medtem ko bi nam intervali zaupanja na desni sliki lahko nakazovali statistično značilnost, čeprav je v resnici ravno obratno. Čeprav je širina intervalov zaupanja odvisna od variabilnosti, ki je povezana s statistično značilnostjo, pa je prekrivanje intervalov zaupanja oziroma sekanje premic odvisno predvsem od začetnih vrednosti, ki nikakor niso povezane z interakcijo. Risanje intervalov zaupanja pri grafičnih predstavitvah interakcij v splošnem zato ni priporočljivo.

Uporaba knjižnice rms v programu R

V programu R³ je v eno izmed osnovnih knjižnic `stats` vgrajena funkcija `predict`, ki izračuna napovedane vrednosti izida skupaj z intervali zaupanja za velik nabor regresijskih modelov, kot so linearni modeli (preko funkcije `lm`), posplošeni linearni modeli (`glm`) in s tem logistična regresija in nelinearni modeli (`nls`). Z vključitvijo knjižnice `survival`⁵ dobimo še možnost izračuna napovedi za Coxov regresijski model (`coxph`). Pri vseh je potrebno v argumentu `newdata` v obliki podatkovnega okvirja (`data.frame`) podati vrednosti neodvisnih spremenljivk, pri katerih bo izračunana napoved izida, funkcija `predict` pa vrne matriko s tremi stolpci, v katerih so shranjene napovedi in njihove spodnje ter zgornje meje intervalov zaupanja (če je argument `interval` nastavljen na vrednost `'confidence'`). Že ko imamo v modelu le nekaj več neodvisnih spremenljivk, postane ročno določanje vrednosti vseh neodvisnih spremenljivk mučno. Če bi na primer želeli v našem prvem modelu za izid Y predstaviti interakcijo, ki vključujeta tri neodvisne spremenljivke X_2 , Z_1 in Z_2 (kot na sliki 3), potem bi ob 100 vrednostih za številsko spremenljivko X_2 morali za argument `newdata` eksplicitno podati podatkovni okvir velikosti 400×4 (v njem bi morali med drugim 400-krat ponoviti neko vrednost za X_1), pa čeprav so pri opisnih spremenljivkah Z_1 in Z_2 možne vrednosti jasne. Ta problem je bolj elegantno rešen v knjižnici `rms` s

funkcijo `Predict` (z veliko začetnico), katere glavne lastnosti in prednosti bomo predstavili v nadaljevanju. Avtor knjižnice je priznani biostatistik Frank E. Harrell, ki je v svoji knjigi o regresijskem modeliranju¹ knjižnici namenil posebno poglavje. V članku smo uporabili njeno verzijo 6.0-0.²

Uporabo funkcije bomo predstavili na primeru prvega modela za Y in pripadajoče slike 3.



Slika 3 Grafična predstavitev interakcij v linearnem modelu za Y , narisana z uporabo funkcije `Predict` iz knjižnice `rms` v programu R.

Privzete vrednosti neodvisnih spremenljivk

Pri uporabi funkcije `Predict` je ključno, da si najprej v globalnih nastavitvah za neodvisne spremenljivke shranimo privzete vrednosti, pri katerih bomo napovedi izračunali. Če so naši podatki shranjeni v podatkovnem okviru z imenom podatki, to naredimo preprosto z:

```
dd=datadist(podatki);
options(datadist='dd')
```

Z `datadist` v `options` je s tem med drugim določeno:

- na katero vrednost bodo nastavljene spremenljivke, ki bodo pri napovedi fiksne (npr. X_1) – funkcija `datadist` nastavi modus za opisne spremenljivke (tiste, ki so shranjene kot `factor`) in mediano za številске;
- na katerem razponu bodo izračunane napovedi za številsko spremenljivko, ki bo pri napovedi variirala (npr. X_2) – funkcija `datadist` nastavi 5. in 95. percentil pri velikost vzorca 200 ali manj, za večje vzorce pa 10. najmanjšo in največjo vrednost.

Če želimo, lahko v objektu `dd` nekatere od nastavljenih privzetih vrednosti prepisemo s svojimi in šele nato `dd` zapišemo v `options`. Tako definirane privzete vrednosti bodo veljale za vse modele, ki jih bomo nato prilagodili podatkom.

Funkcija Predict

Čeprav lahko s funkcijo `Predict` izračunamo napovedane vrednosti izida za nekoliko manjši nabor regresijskih modelov kot s funkcijo `predict`, je ta še zmeraj obsežen. Za prilagajanje modela pa moramo uporabiti funkcije iz knjižnice `rms`, na primer za linearne modele funkcijo `ols`, za logistično regresijo `lrm`, za posplošene linearne modele `Glm` (velika začetnica) in za Coxov regresijski model `cph`. Za nelinearne učinke v modelu prav tako uporabimo funkcije iz knjižnice `rms`, na primer za polinome funkcijo `pol`, za kubični zlepek z omejitvami (angl. *restricted cubic spline*) pa bi uporabili funkcijo `rCs`.

Za naš primer torej prilagodimo model linearne regresije s funkcijo `ols` (uporabimo `pol`, shranimo v `fit`) in ker imamo privzete vrednosti že nastavljene, dobimo sliko 3 (oziroma eno izmed njenih različic) preprosto z ukazoma `p=Predict(fit,x2,z1,z2); ggplot(p)`.

Na levem in desnem grafu so narisane napovedane vrednosti pri spremenljivki Z_2 enaki 1 oziroma 2. Za sliko 3 smo v funkciji `Predict` dodatno nastavili argument `conf.int` na `FALSE`, saj smo v prejšnjem razdelku videli, da risanje intervalov zaupanja ob prikazovanju interakcij ni priporočljivo.

Pod grafom se privzeto izpišejo vrednosti, na katere so preko `datadist` nastavljene preostale spremenljivke v modelu. To lahko onemogočimo z uporabo logičnega argumenta `adj.subtitle` znotraj funkcije `ggplot`. Če bi želeli X_1 nastaviti na kakšno drugo vrednost, to določimo znotraj funkcije `Predict`, npr. z `x1=0`. Globalnih nastavitvev in `datadist` nam torej ni potrebno spreminjati.

V našem primeru smo za risanje uporabili funkcijo `ggplot` (oz. `ggplot.Predict`), ki uporablja knjižnico `ggplot2`,⁶ lahko pa bi uporabili tudi `plot`, ki uporablja knjižnico `lattice`.⁷ Če bi variirali le dve spremenljivki namesto treh, bi lahko uporabili še `plotp`, ki uporablja knjižnico `plotly`.⁸

Znotraj funkcije `ggplot` lahko nastavimo še nekaj drugih argumentov. V našem primeru smo `aestype` nastavili na `'linetype'`, privzeta vrednost je `'color'`, možni pa sta še `'size'` in `'shape'`. Sliko lahko prilagodimo svojim željam z verižnim dodajanjem grafičnih ukazov preko znaka `+`, kot je običajno za risanje z `ggplot2`, ali pa iz dobljenih izračunanih napovedanih vrednosti (shranjene so v predmet `p`) naredimo svoj podatkovni okvir in na njem sami zgradimo celoten `ggplot` predmet. To je enostavno, saj so v predmetu `p` shranjene vrednosti

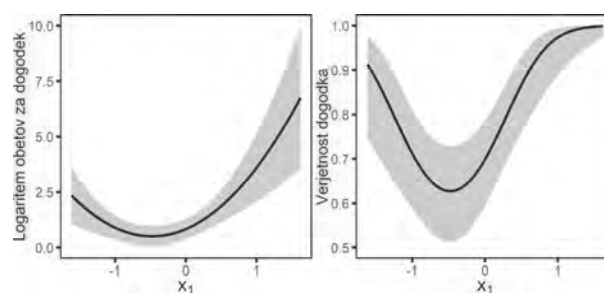
vseh neodvisnih spremenljivk, pri katerih so napovedane vrednosti izračunane.

Kakor pri vseh knjižnicah oziroma programih, je za končno sliko potrebno nastaviti in popraviti mnogo podrobnosti (oznake, velikosti, razmiki ipd.), za vmesna poročila in preučevanje podatkov ter modelov pa zadostujejo že grafi, ki jih dobimo z nekaj kratkimi vrsticami kode, predstavljene zgoraj. Kot smo navedli že uvodoma, je programska koda za vse slike iz članka in njihove preprostejše različice dostopna na spletu.⁴

Če imamo v programu R hkrati naloženo še knjižnico `car`,⁹ je namesto `Predict` nujno uporabiti `rms::Predict`, saj ima knjižnica `car` istoimensko funkcijo, ki je različica funkcije `predict` iz knjižnice `stats`, uporabna le za linearne modele s funkcijo `lm`.

Prikaz pretvorjenega izida

Oglejmo si še zadnjo prednost funkcije `Predict`, in sicer izračun napovedanih vrednosti za pretvorjen izid. To potrebujemo na primer v modelu logistične regresije, kjer je predstavitev napovedanih vrednosti na osnovni lestvici logaritma obetov za dogodek sicer primerna za preučevanje učinkov s statističnega vidika, ni pa koristna za poveden prikaz strokovni javnosti. Za slednjo je potrebno predstaviti napovedane vrednosti za verjetnost dogodka. Kako lahko to enostavno naredimo z uporabo funkcije `Predict`, prikazuje spodnji primer.



Slika 4 Predstavitev nelinearnega učinka v modelu logistične regresije na napovedane vrednosti logaritma obetov za dogodek (levo) in verjetnosti za dogodek (desno) s 95 % intervali zaupanja, narisana z uporabo funkcije `Predict` iz knjižnice `rms` v programu R.

Generiramo izid iz modela logistične regresije, v katerem je linearni prediktor na lestvici logaritma obetov enak $1 + X_1 + X_1^2$ (uporabimo že generiran X_1), in prilagodimo model logistične regresije preko funkcije `lrm` s kvadratnim učinkom za X_1 (uporabimo `pol`, shranimo v `fit`). Z uporabo ukaza `Predict(fit,x1)` dobimo levo stran slike 4, na

kateri so napovedane vrednosti logaritma obojev za dogodek, desno stran slike pa dobimo tako, da znotraj `Predict` dodatno uporabimo `fun=plogis`.

Potreba po prikazu pretvorjenega izida se lahko naravno pojavi tudi v linearni regresiji. Če so predpostavke modela linearne regresije s prvotnim izidom kršene, jih lahko skušamo popraviti tako, da izid pretvorimo. Poskusimo lahko uporabiti različne strogo naraščajoče funkcije. V primeru pozitivnega izida lahko uporabimo na primer pretvorbo Box-Cox,¹⁰ ki je enoparametrična družina potenčnih funkcij, kjer parameter ocenimo s pomočjo metode največjega verjetja. V programu R lahko uporabimo knjižnico `car`,⁹ ocenimo parameter s funkcijo `powerTransform` in nato uporabimo ustrezno pretvorbo Box-Cox s funkcijo `bcPower`.

Čeprav je model morda statistično primerneje narediti s pretvorjenim izidom, pa nas z vsebinskega vidika zanima učinek spremenljivk na prvotni izid. Izid iz modela linearne regresije moramo torej pretvoriti nazaj v prvotnega, kar lahko naredimo tako, da v programu R definiramo funkcijo za inverzno pretvorbo (shranimo v `f`) in nato znotraj funkcije `Predict` uporabimo `fun=f`. Izračunamo lahko torej napovedane vrednosti za poljubno pretvorjen izid.

Razprava

V članku smo opozorili, na kaj moramo biti pazljivi pri preučevanju in interpretaciji intervalov zaupanja za napovedane vrednosti izida v regresijskih modelih. Gotovo bi se našlo še marsikaj, na kar moramo biti pozorni pri grafičnih prikazih modelov. Osredotočili smo se na nelinearne učinke in interakcije, saj jih praviloma ni mogoče povzeti številčno na intuitiven način in so grafični prikazi nujni za njihovo povedno predstavitev v poročilih ali znanstvenih člankih.

Drugi del članka ponudi uporabnikom programa R enostaven način za računanje napovedanih vrednosti (prvotnega ali pretvorjenega) izida in njihovo risanje s pomočjo knjižnice `rms`. Za analizo podatkov seveda obstaja še mnogo drugih (statističnih) programov z dobrimi možnostmi grafičnega prikazovanja, prav tako so v programu R na voljo tudi drugi načini in knjižnice. Predstavljena kratka navodila omogočajo

hitro usvojitev preprostega načina prikazovanja učinkov za uporabnike z osnovnim poznavanjem programa R.

Zaključek

V regresijskih modelih so intervali zaupanja priporočljivi pri grafičnih prikazih nelinearnih učinkov na napovedane vrednosti izida (podajajo informacijo o variabilnosti ocene in omogočajo presojanje strokovne pomembnosti za populacijo), pri čemer pa se je potrebno zavedati, da je njihova širina odvisna od tega, katere vrednosti smo določili preostalim spremenljivkam v modelu. Grafično presojanje statistične značilnosti nelinearnosti učinka na podlagi intervalov zaupanja v multipli regresiji zato ni verodostojno.

Vključevanje intervalov zaupanja na grafične predstavitev interakcij med neodvisnimi spremenljivkami odsvetujemo, saj nas njihovo (ne)prekrivanje lahko vodi v zmotno prepričanje o statistični (ne)značilnosti interakcije.

Reference

1. Harell FE: *Regression modelling strategies* (2nd ed.). New York 2015: Springer.
2. Harell FE: *rms: Regression modeling strategies*, R package version 6.0-0. 2020, <https://cran.r-project.org/package=rms/> (8. 7. 2020)
3. R Core Team: *R: A language and environment for statistical computing*. Vienna 2020: Foundation for Statistical Computing, <https://www.r-project.org/> (8. 7. 2020)
4. Ružič Gorenjec N: *Grafični prikazi učinkov v regresijskih modelih, programska koda*. <http://ibmi.mf.uni-lj.si/files/graficniPrikaziUcinokov.r> (8. 7. 2020)
5. Therneau TM: *A package for survival analysis in R*, R package 3.2-3. 2020, <https://cran.r-project.org/package=survival/> (8. 7. 2020)
6. Wickham H: *ggplot2: Elegant graphics for data analysis* (2nd ed.). New York 2016: Springer.
7. Sarkar D: *Lattice: multivariate data visualization with R*. New York 2008: Springer.
8. Sievert C: *Interactive web-based data visualization with R, plotly, and shiny*. Florida 2020: CRC Press.
9. Fox J, Weisberg S: *An R companion to applied regression* (3rd ed.). Thousand Oaks CA 2019: Sage.
10. Box GEP, Cox DR: An analysis of transformations. *J R Stat Soc Series B Stat Methodol* 1964; 26(2): 211-252.