Neparametarska regresija

Sertić, Tin

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Tin Sertić

NEPARAMETARSKA REGRESIJA

Diplomski rad

Voditelj rada:
doc. dr. sc. Azra Tafro

Zagreb, Prosinac, 2023

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Zahvaljujem dragome Bogu, Blaženoj Djevici Mariji, svetome Antunu Padovanskom, dragoj obitelji, strpljivoj mentorici, prijateljima, profesorima i dobročiniteljima koji su svojim dobrim primjerom, mudrim savjetom i neograničenom potporom podržavali i usmjeravali moje obrazovanje.

Sadržaj

Uvod

U ovom diplomskom radu cilj je predstaviti metode neparametarske regresije, pri čemu regresiju smatramo nastojanjem da iz dostupnih podataka (npr. kvadratura nekretnina i njihovih cijena) naučimo procjenjivati vrijednost neprekidne varijable (npr. procjenjivati cijenu nekretnine za neku proizvoljnu kvadraturu). Kako bismo to učinili, potrebno je oblikovati procjenitelja (našu procjenu) regresijske funkcije (stvarne veze). To možemo učiniti metodama parametarske regresije u kojima procjenitelja prilagođavamo nekom fiksnom obliku funkcije za koji smo se odlučili. Budući da sam oblik regresijske funkcije (stvarne veze) nama može biti neprepoznatljiv iz samog grafičkog prikaza, predstavit ćemo metode neparametarske regresije u kojima ćemo se služiti pomoćnim metodama kako bismo odredili oblik funkcije kojem ćemo nastojati prilagoditi procjenitelja.

Kako bismo to ostvarili, u prvom poglavlju uvest ćemo pojam regresije, napraviti kratak pregled linearne regresije kao najpoznatijeg predstavnika parametarskih metoda te iznijeti detaljniju motivaciju za uvođenje neparametarske regresije. Isto tako, predstavit ćemo model neparametarske regresije uz koji ćemo navesti pretpostavke koje će nas ograničiti na klasu linearnih procjenitelja iz koje će dolaziti svi procjenitelji koje ćemo formirati. Na kraju prvog poglavlja ćemo diskutirati odnos pristranosti i varijance prilikom formiranja procjenitelja.

U drugom poglavlju predstavit ćemo metodu lokalnih prosjeka, metodu procjene jezgrom i metodu lokalne linearne regresije zajedno s pomoćnim metodama unakrsne validacije koje će nam pomoći odabrati parametar koji će određivati oblik procjenitelja. Predstavljene metode ilustrirat ćemo na jednakim umjetno generiranim podacima u programskom jeziku R.

U trećem poglavlju usporedit ćemo modele dobivene primjenom predstavljenih metoda te ćemo odabrati model koji je najbolje formirao procjenitelja poznate regresijske funkcije iz koje smo generirali podatke. Uz to, na primjeru povezanosti mjesečnih primanja i ugleda predstavit ćemo modeliranje predstavljenim metodama zajedno s odabirom modela u primjeru gdje nam regresijska funkcija nije poznata.

Poglavlje 1

Osnove

U ovom poglavlju nastojat ćemo ponoviti pojmove koji se obrađuju na razini preddiplomskog studija matematike, a koji će zajedno s predstavljenom teorijom i primjerima tvoriti smislenu i zaokruženu cjelinu. Zbog opsežnosti materije, teško je u potpunosti ponoviti sve relevantne pojmove, stoga upućujemo čitatelja da po potrebi prouči [3] i [6] gdje se detaljnije obrađuju pojmovi na čijim osnovama temeljimo daljnje rezultate. Nakon toga, prikazat ćemo primjer iz [2] koji će nas motivirati da predstavimo pojam neparametarske regresije koji ćemo u daljnjim poglavljima detaljnije razraditi. Definicije i rezultati iz ovog poglavlja temeljeni su na [4] i [7].

1.1 Općenito o regresiji

U svijetu oko sebe primjećujemo raznovrsne varijable za koje smatramo da ovise jedne o drugima, kao na primjer kvadratura nekretnine i njezina cijena ili visina inflacije i razina nezaposlenosti. Kako bi otkrili tu vezu, odnosno *regresijsku funkciju*, koristimo *metode regresije* kako bi dobili *procjenitelja* kojim možemo predviđati npr. cijene nekretnina i razine nezaposlenosti za proizvoljne kvadrature i visine inflacije.

Definicija 1.1.1. *Neka je dan uzorak od n opažanja* $(x_i, Y_i), i = 1, ..., n$, *pri čemu* x_i *nazivamo kovarijatama, a* Y_i *vrijednostima varijable odziva. Regresijski model definiramo kao vezu varijable odziva Y i kovarijate x sljedećim jednadžbama*

$$Y_i = f(x_i) + \varepsilon_i, \mathbb{E}(\varepsilon_i) = 0, i = 1, ..., n, \tag{1.1}$$

pri čemu pretpostavljamo da je funkcija $f : \mathbb{R}^p \to \mathbb{R}$ *nepoznata glatka funkcija koju želimo procijeniti te koju nazivamo regresijskom funkcijom. Nadalje, pretpostavljamo da varijanca greške iznosi* $\mathbb{V}(\varepsilon_i) = \sigma^2$ *i ne ovisi o x. Metode koje koristimo kako bi procijenili regresijsku funkciju nazivamo metodama regresije. Primjenom metoda regresije dolazimo do*

funkcijske veze kojom procjenjujemo f(x), a koju nazivamo procjeniteljem ili izglađivačem i označavamo s $\hat{f}_n(x)$.

Dakle, regresija je općeniti pojam koji obuhvaća sve pojmove iz gornje definicije, a predstavlja nastojanje da iz dostupnih podataka (npr. kvadratura nekretnina i njihovih cijena) naučimo procjenjivati vrijednost neprekidne varijable (npr. procjenjivati cijenu nekretnine za neku proizvoljnu kvadraturu).

Napomena 1.1.2. *U 1.1 pretpostavljamo da su vrijednosti kovarijate* x_i *fiksne. Umjesto toga, mogli bismo pretpostaviti da su nasumične, pa bi u tom slučaju zapisivali podatke kao* $(X_1, Y_1), ..., (X_n, Y_n)$ *te bi* $f(x)$ *interpretirali kao uvjetno očekivanje Y s obzirom na* $X = x$*, odnosno*

$$f(x) = \mathbb{E}(Y|X = x). \tag{1.2}$$

Kako bismo pronašli procjenitelja regresijske funkcije, možemo koristiti *parametarske* i *neparametarske* metode regresije, pri čemu parametrom smatramo numeričku vrijednost o kojoj ovisi regresijska funkcija. U metodama *parametarske* regresije unaprijed pretpostavljamo oblik regresijske funkcije f i unaprijed određujemo broj parametara koje zatim procjenjujemo korištenjem istih *parametarskih* metoda regresije. S druge strane, metodama *neparametarske* regresije težimo istovremeno otkriti oblik regresijske funkcije, odrediti broj parametara i nakon svega procijeniti te parametre.

Prije nego što počnemo obrađivati metode neparametarske regresije, ukratko ćemo predstaviti metodu linearne regresije kao najpoznatijeg predstavnika metoda parametarske regresije.

1.2 Kratak pregled linearne regresije

Linearna regresija je metoda parametarske regresije koja pretpostavlja da je regresijska funkcija linearna, odnosno da je ona oblika 1.3. Istovremeno, ona želi unutar *linearnog regresijskog modela* procijeniti parametre $\beta_1, ..., \beta_p$ o kojima smo pretpostavili da ovisi regresijska funkcija.

Definicija 1.2.1. *Neka je dan uzorak od n opažanja* $(x_i, Y_i), i = 1, ..., n$*„ pri čemu je* $Y_i \in \mathbb{R}$ *te* $x_i = (x_{i1}, ..., x_{ip})^T \in \mathbb{R}^p$*. Linearni regresijski model pretpostavlja da je*

$$Y_i = f(x_i) + \varepsilon_i \equiv \sum_{i=1}^{p} \beta_j x_{ij} + \varepsilon_i, i = 1, ..., n, \tag{1.3}$$

pri čemu je $\mathbb{E}(\varepsilon_i) = 0$ *te* $\mathbb{V}(\varepsilon_i) = \sigma^2$.

Napomena 1.2.2. *Najčešće želimo uključiti odsječak u model pa ćemo koristiti konvenciju da je* $x_{i1} = 1, i = 1, ..., n$.

Definicija 1.2.3. *Matricu sustava X definiramo kao nxp matricu*

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix}.$$

Napomena 1.2.4. *Ako koristimo notaciju* $Y = (Y_1, ..., Y_n)^T, \varepsilon = (\varepsilon_1, ..., \varepsilon_n)^T$ *i* $\beta = (\beta_1, ..., \beta_p)^T$ *tada jednadžbu 1.3 možemo zapisati kao:*

$$Y = X\beta + \varepsilon. \tag{1.4}$$

Kako bi pronašli procjenitelja regresijske funkcije, koristit ćemo verziju linearne regresije zvanu metodom najmanjih kvadrata, prije čijeg predstavljanja ćemo uvesti sljedeću definiciju:

Definicija 1.2.5. *Sumu kvadrata grešaka definiramo kao*

$$SKG = (Y - X\beta)^T(Y - X\beta) = \sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2. \tag{1.5}$$

Dakle, suma kvadrata grešaka nije ništa drugo nego suma kvadriranih pogrešaka koje smo učinili pri procjeni svakog od pojedinih $Y_i, i = 1, ..n$. Nadalje, procjeniteljem metodom najmanjih kvadrata nazivat ćemo upravo onu p-torku koja će minimizirati tu sumu te koja će jednoznačno odrediti regresijsku funkciju pretpostavljenu u linearnom regresijskom modelu 1.3.

Definicija 1.2.6. *Procjenitelj metodom najmanjih kvadrata* $\hat{\beta} = (\hat{\beta}_1, ..., \hat{\beta}_p)^T$ *je vektor koji minimizira sumu kvadrata grešaka danu s 1.5.*

Teorem 1.2.7. *Ako je* X^TX *invertibilna matrica, procjenitelja metodom najmanjih kvadrata možemo dobiti kao*

$$\hat{\beta} = (X^TX)^{-1}X^TY. \tag{1.6}$$

Dokaz. Želimo pronaći procjenitelja metodom najmanjih kvadrata $\hat{\beta}$, odnosno onog koji minimizira

$$SKG(\beta) = (Y - X\beta)^T(Y - X\beta).$$

Prije svega, primijetimo da se $SKG(\beta)$ može zapisati kao

$$SKG(\beta) = Y^TY - \beta^TX^TY - Y^TX\beta + \beta^TX^TX\beta = Y^TY - 2\beta^TX^TY + \beta^TX^TX\beta,$$

pri čemu smo u drugoj jednakosti koristili činjenicu da je $\beta^T X^T Y$ kao skalar jednak svojoj transponiranoj vrijednosti, odnosno da je jednak $Y^T X\beta$. Nadalje, po rezultatima matematičke analize znamo da $\hat{\beta}$ mora zadovoljavati

$$\frac{\partial S\,KG}{\partial \beta}(\hat{\beta}) = -2X^T Y + 2X^T X\hat{\beta} = 0,$$

što se ljepše može zapisati kao

$$X^T X\hat{\beta} = X^T Y. \tag{1.7}$$

Napokon, množenjem obje strane jednadžbe 1.7 inverzom matrice $X^T X$ dobivamo

$$\hat{\beta} = (X^T X)^{-1} X^T Y.$$

□

Nadalje, nakon što smo definirali procjenitelja metodom najmanjih kvadrata i pronašli njegovu formulu u zatvorenom obliku, definirat ćemo i procjenu tim istim procjeniteljem:

Definicija 1.2.8. *Procjena regresijske funkcije metodom najmanjih kvadrata $f(x)$ u točki $x = (x_1, ..., x_p)^T$ je dana s*

$$\hat{f}_n(x) = \sum_{j=1}^{p} \hat{\beta}_j x_j = x^T \hat{\beta}. \tag{1.8}$$

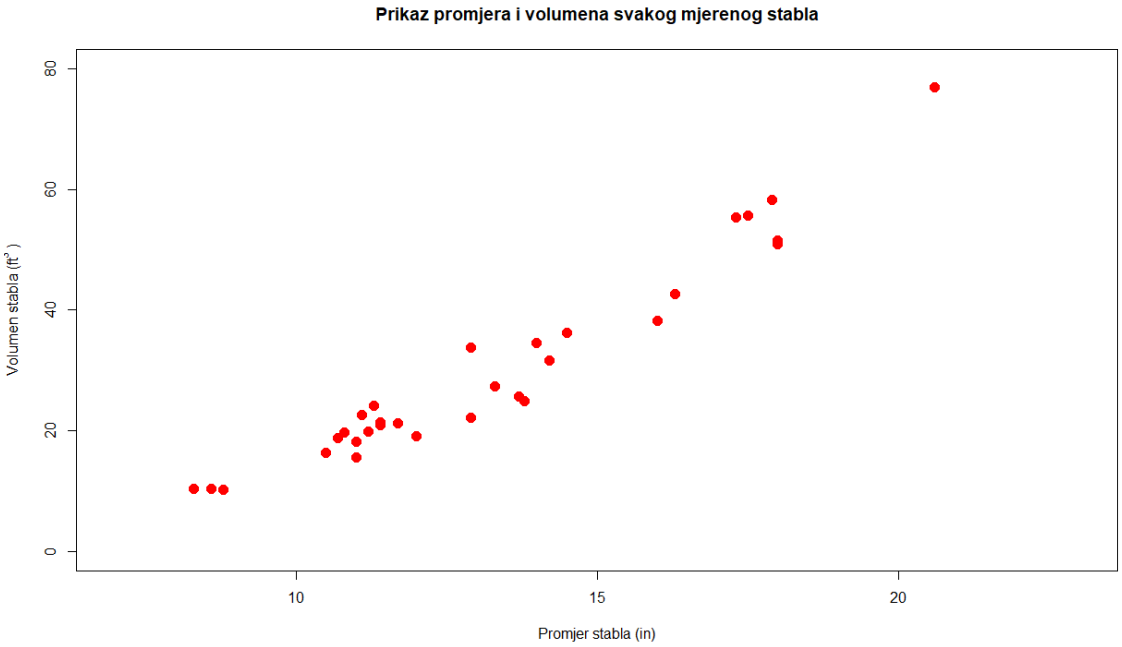
Vektor procijenjenih vrijednosti $\boldsymbol{f} = (\hat{f}_n(x_1), ..., \hat{f}_n(x_n))^T$ tada možemo zapisati kao

$$\boldsymbol{f} = X\hat{\beta}. \tag{1.9}$$

Primjer 1.2.9. *Kako bi ilustrirali metodu, promotrit ćemo podatke iz skupa podataka* **trees**[1] *koji prikazuje podatke o promjeru, visini i volumenu 31 posječenog drveta crne trešnje. Upotrijebit ćemo linearnu regresiju kako bi opisali odnos između promjera drveta (mjerenog u inčima na visini od 4 stope i 6 inča) i njegovog volumena (mjerenog u kubičnim stopama) te ćemo dobivenog procjenitelja iskoristiti kako bi procijenili volumen drveta crne trešnje promjera 20 inča (~50.8 cm) .*[2]

[1] Skup podataka unutar osnovnog paketa programskog jezika R[5]

[2] 1 stopa = 12 inča, 1 inč = 2.54 cm



Slika 1.1: Grafički prikaz promjera i volumena svakog mjerenog stabla

Nakon što smo na slici 1.1 grafički prikazali podatke o promjeru i volumenu svakoga stabla, možemo naslutiti da kroz podatke možemo dobro provući pravac (odnosno provesti linearnu regresiju metodom najmanjih kvadrata) kako bi pronašli procjenitelja (našu procjenu) regresijske funkcije (pravog odnosa promjera stabla i njegovog volumena). Primjenom linearne regresije dolazimo do procjenitelja (naše procjene) regresijske funkcije (pravog odnosa promjera stabla i njegovog volumena) koji je grafički prikazan na slici 1.2, a dan je sljedećom formulom:

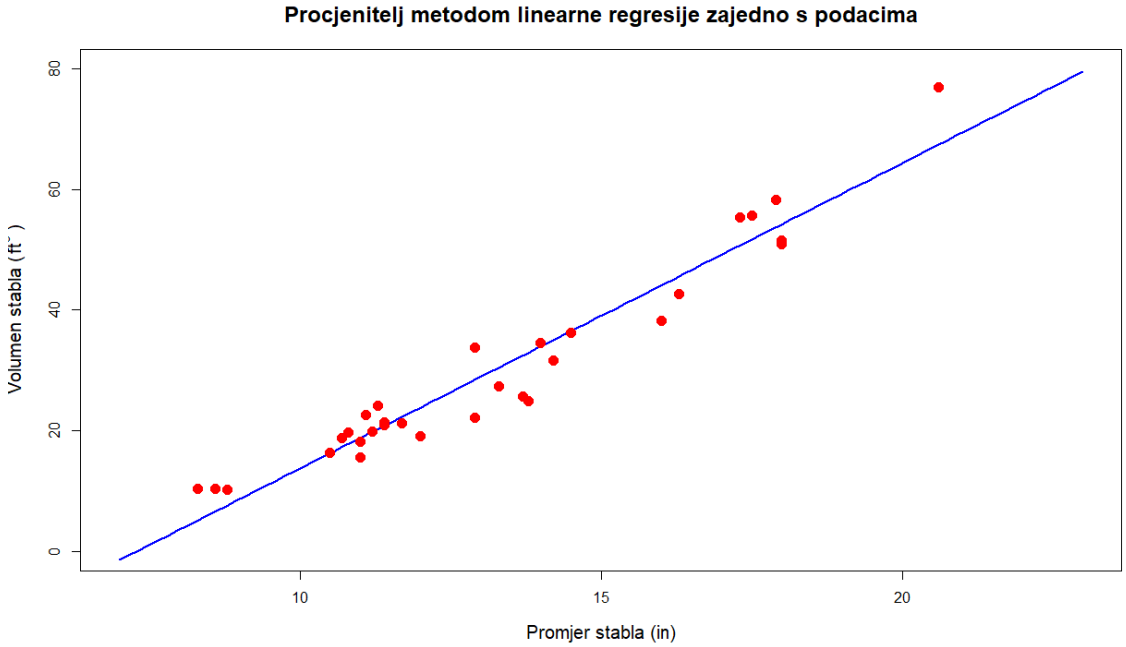
$$volumen = -36.9435 + 5.0659 * promjer.$$

Naposljetku, iskoristit ćemo gornju formulu kako bi procijenili volumen crnog trešnjinog drveta promjera 20 inča (~50.8 cm). Kako bi dobili procjenu metodom linearne regresije, potrebno je uvrstiti 20 u dobivenu formulu kako bi dobili procjenu da volumen takvog drveta iznosi 64.37367 kubičnih stopa (~1.822859 kubičnih metara).

1.3 Motivacija za uvođenje neparametarske regresije

Promotrit ćemo četiri skupa podataka identične uzoračke varijance i prosjeka koje nazivamo Anscombeovim kvartetom [3] , a koje smo tablično prikazali u tablici 1.1. Oni su

[3] Francis John Anscombe (1918.-2001.), engleski statističar



Slika 1.2: Grafički prikaz procjenitelja metodom linearne regresije zajedno s podacima

izvorno oblikovani kako bi naglasili potrebu korištenja grafičkih alata uz regresijske modele, ali istovremeno daju izvrsnu motivaciju za uvođenje neparametarske regresije.

Naime, ukoliko na svakom skupu podataka napravimo linearnu regresiju, dolazimo do četiri linearna modela s jednakim koeficijentima smjera pravca i odsječka na y osi te s jednakim koeficijentom determinacije. [4]

Kako bi se u to uvjerili, detaljnije pogledajmo grafički prikaz linearnih modela na slici 1.3 na kojoj možemo primijetiti da sva četiri modela daju jednakog procjenitelja koji ima ”dobar” koeficijent determinacije, odnosno koji je ”dobro” prilagođen svakom od četiri skupa podataka. Detaljnije promatrajući sliku 1.3 dolazimo do sljedeća tri nedostatka ”slijepog” primjenjivanja linearne regresije, kao i svakog drugog modela parametarske regresije:

[4] Jednakost u ovom poglavlju se smatra jednakošću do na 10^{-2}

Br	X1	Y1	X2	Y2	X3	Y3	X4	Y4
1	10.00	8.04	10.00	9.14	10.00	7.46	8.00	6.58
2	8.00	6.95	8.00	8.14	8.00	6.77	8.00	5.76
3	13.00	7.58	13.00	8.74	13.00	12.74	8.00	7.71
4	9.00	8.81	9.00	8.77	9.00	7.11	8.00	8.84
5	11.00	8.33	11.00	9.26	11.00	7.81	8.00	8.47
6	14.00	9.96	14.00	8.10	14.00	8.84	8.00	7.04
7	6.00	7.24	6.00	6.13	6.00	6.08	8.00	5.25
8	4.00	4.26	4.00	3.10	4.00	5.39	19.00	12.50
9	12.00	10.84	12.00	9.13	12.00	8.15	8.00	5.56
10	7.00	4.82	7.00	7.26	7.00	6.42	8.00	7.91
11	5.00	5.68	5.00	4.74	5.00	5.73	8.00	6.89

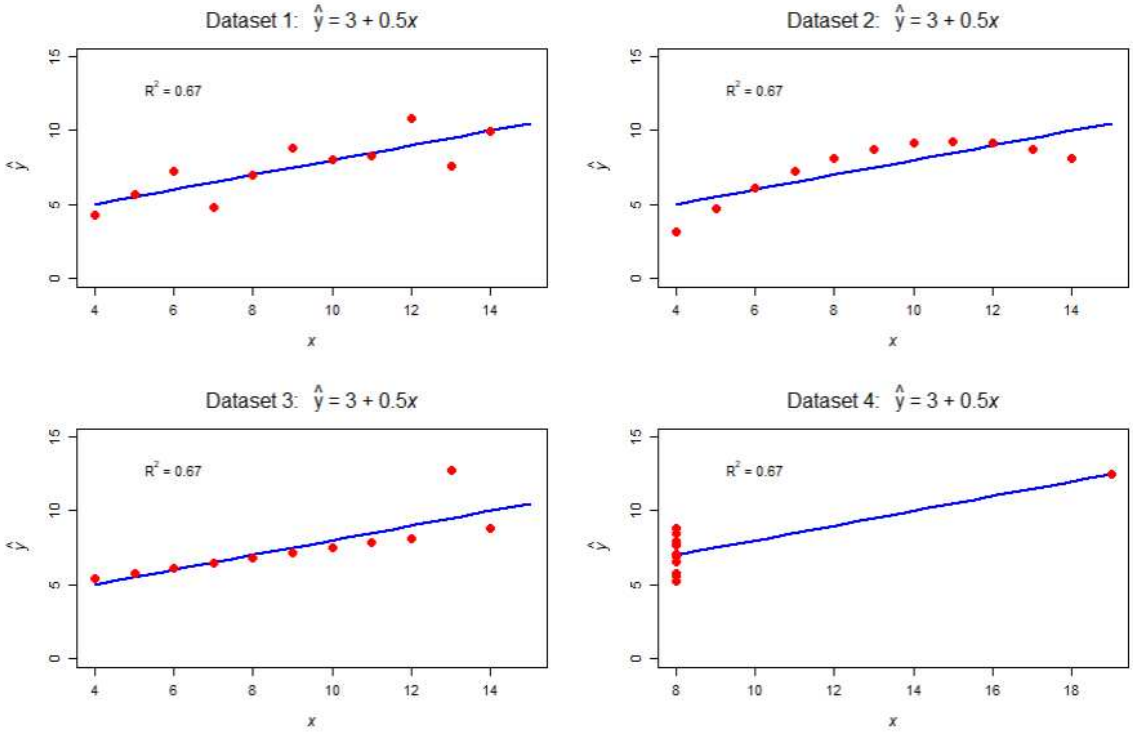
Tablica 1.1: Anscombeov kvartet podataka

1. Model koji ima ”visok” koeficijent determinacije, tj. ”dobru” prilagođenost izvornim podacima, ne mora *a priori* biti dobar model.

2. Modeli parametarske regresije ne moraju otkriti niti prepoznati ”pravu” vezu među podacima, koja može biti apstraktna, tj. nama neprepoznatljiva iz samog grafičkog prikaza i u stvarnosti drugačija od onog modela kojem smo mi željeli prilagoditi podatke i one metode kojom smo željeli pronaći procjenitelja.

3. Kako bi modeliranje bilo postepeno i smisleno, prije odabira konačnog modela potrebno je grafički prikazati podatke te promotriti nekoliko modela (parametarskih i/ili neparametarskih), kako bi se između njih odlučili za najbolji kojim ćemo raditi eventualne daljnje procjene, kao što ćemo i mi učiniti na kraju ovog diplomskog rada u primjeru 3.2.1.

Primijetimo da bi neki modeli mogli biti zamijenjeni boljim alternativama parametarske regresije (na primjer, u slučaju trećeg modela, mogli bismo se odlučiti za linearnu regresiju uz minimizaciju neke druge funkcije umjesto sume kvadrata grešaka prikazane formulom 1.5, čijim korištenjem greške koje uzrokuju ekstremne vrijednosti nebi imale toliki utjecaj na konačni oblik procjenitelja). Ipak, mi ćemo se u ovom radu odlučiti za



Slika 1.3: Prilagodba Anscombeovog kvarteta linearnoj regresiji

prezentaciju metoda neparametarske regresije koje imaju potencijal znatno proširiti paletu metoda kojima čitatelj raspolaže za analizu podataka.

1.4 Model neparametarske regresije

Model neprametarske regresije se poput onog parametarske regresije temelji na općenitoj definiciji 1.1.1. Na tu definiciju zatim dodajemo dodatne pretpostavke, kao što smo u slučaju parametarske (linearne) regresije to učinili u definiciji 1.2.1. Tako ćemo i u ovom diplomskom radu na tu definiciju dodati određene pretpostavke, kako bi dobili podskupinu procjenitelja neparametarske regresije zvanu linearni procjenitelji. Dakle, svaki od neparametarskih procjenitelja koje ćemo prezentirati u ovome diplomskom radu bit će neki oblik linearnog procjenitelja čija formalna definicija slijedi u nastavku.

Definicija 1.4.1. *Procjenitelj $\hat{f}_n(x)$ od f je **linearni procjenitelj** ako za svaki x postoji vektor $l(x) = (l_1(x), ..., l_n(x))^T$ takav da je*

$$\hat{f}_n(x) = \sum_{i=1}^{n} Y_i l_i(x). \tag{1.10}$$

Vektor procijenjenih vrijednosti definiramo kao

$$\boldsymbol{f} = (\hat{f}_n(x_1), ..., \hat{f}_n(x_n))^T. \tag{1.11}$$

pri čemu je $Y = (Y_1, ..., Y_n)^T$. Iz toga lako slijedi da se $\boldsymbol{f}$ može zapisati kao

$$\boldsymbol{f} = LY, \tag{1.12}$$

pri čemu je L n x n matrica čiji je i-ti redak $l(x_i)^T$, odnosno za elemente matrice L vrijedi $L_{ij} = l_j(x_i)$. Vrijednosti i-tog retka možemo protumačiti kao dodijeljene težine svakom od Y_i – eva u formiranju procjene $\hat{f}_n(x_i)$.

Dakle, linearni procjenitelj je onaj procjenitelj kojem se za dane kovarijate može pronaći matrica L koju kad pomnožnimo s vektorom originalnih vrijednosti Y dobivamo vrijednost vektora procjene tog linearnog procjenitelja. Stoga je od velike važnosti definirati i imenovati matricu L.

Definicija 1.4.2. *Matricu L koja zadovoljava 1.12 nazivamo matricom izglađivanja (eng. smoothing matrix) ili kapa matricom (eng. hat matrix). i-ti redak matrice L nazivamo efektivnom jezgrom za procjenjivanje $f(x_i)$. Nadalje, definiramo efektivne stupnjeve slobode kao:*

$$v = tr(L). \tag{1.13}$$

Kako bi bolje razumjeli gornju definiciju, linearne procjenitelje možemo zamisliti kao one koji svoju procjenu (npr. procjenu cijene nekretnine) temelje na linearnoj kombinaciji originalnih vrijednosti (npr. kombinaciji svih cijena nekretnina), pri čemu svaka dodijeljena težina ovisi o kovarijati koju procjenjujemo (npr. veću važnost će imati cijena stanova slične kvadrature).

Napomena 1.4.3. *Težine u svim linearnim procjeniteljima koje ćemo predstaviti imat će svojstvo da, za svaki x, $\sum_{i=1}^{n} l_i(x) = 1$. To svojstvo povlači da će ti procjenitelji čuvati konstantne krivulje. Odnosno, ukoliko je $Y_i = c$ za svaki i, onda će biti i $\hat{f}_n(x) = c$.*

U sljedećem poglavlju ćemo predstaviti neke od linearnih procjenitelja zajedno sa pripadnim primjerima koje ćemo nadopunjavati potrebnim teoretskim rezultatima.

1.5 Odnos pristranosti i varijance prilikom formiranja procjenitelja

U sljedećem poglavlju predstavit ćemo kriterije i metode koji će nam pomoći odabrati "najboljeg" procjenitelja metodama neparametarske regresije, dok ćemo u ovom potpoglavlju u nešto većoj općenitosti predstaviti pojmove pristranosti i varijance te njihovu ulogu u odabiru procjenitelja. Kako bismo efikasno odredili procjenitelja regresijske funkcije, potrebno je imati ciljanu metriku, odnosno neku mjeru pogreške koju želimo minimizirati. Kako bismo ju definirali, najprije definiramo kvadratnu grešku (kvadriranu pogrešku učinjenu u točki x) kao:

$$L(f(x), \hat{f}_n(x)) = (f(x) - \hat{f}_n(x))^2. \tag{1.14}$$

Nadalje, uzimamo očekivanje učinjene kvadratne greške te dolazimo do izraza koji nam govori koliku pogrešku možemo očekivati, a koji nazivamo srednje kvadratnom greškom (SKG):

$$SKG = \mathbb{E}(L(f(x), \hat{f}_n(x))) = \mathbb{E}(f(x) - \hat{f}_n(x))^2. \tag{1.15}$$

Taj izraz možemo rastaviti na sljedeći način:

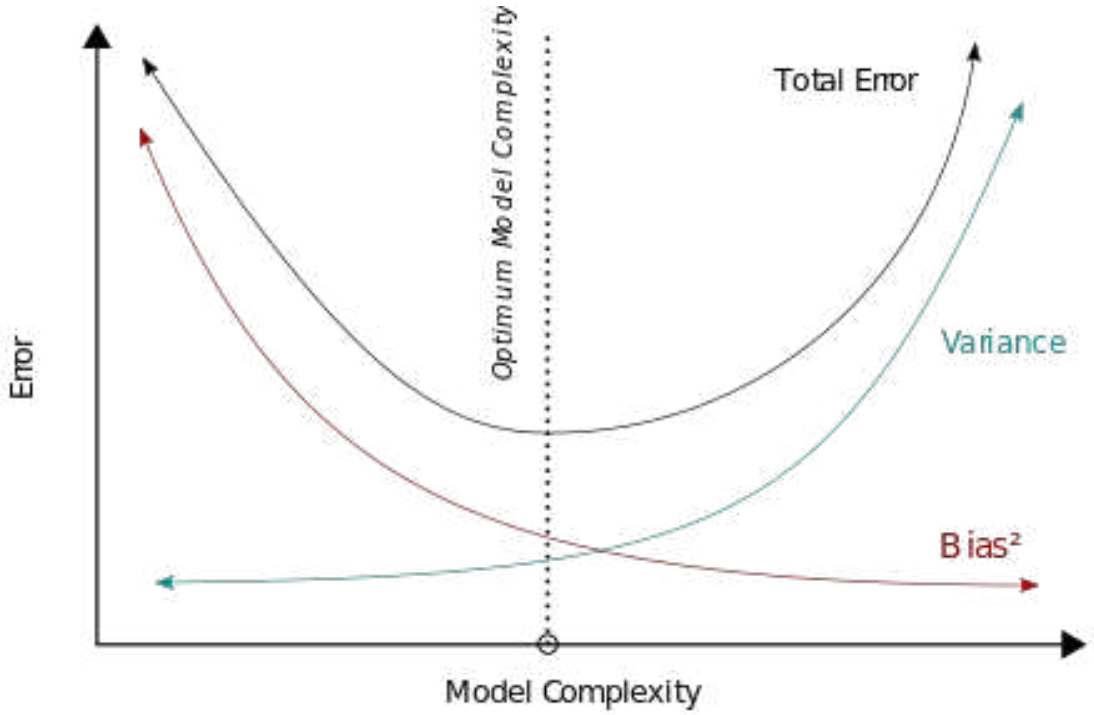
$$SKG = \mathbb{E}(f(x) - \hat{f}_n(x))^2 = (\mathbb{E}(\hat{f}_n(x)) - f(x))^2 + \mathbb{E}((\hat{f}_n(x) - \mathbb{E}(\hat{f}_n(x)))^2) \tag{1.16}$$

Prvi sumand nazivamo kvadriranom pristranošću, a drugi varijancom. Pristranost predstavlja koliko je naša procjena različita od stvarnih vrijednosti, dok varijanca odražava koliko bi se naš procjenitelj razlikovao u procjenama ukoliko bi ga procijenili na nekim drugim ulaznim podacima. Kako bismo bolje razumjeli pojmove pristranosti i varijance, promotrimo dvije krajnosti:

1. Procjenitelj koji je previše jednostavan (npr. za svaki ulazni podatak daje jednaku procjenu). Taj će model imati nisku varijancu, ali visoku pristranost.

2. Procjenitelj koji je previše složen, tj. kompliciran (npr. u potpunosti se prilagođava dostupnim podacima). Taj će model imati visoku varijancu, ali nisku pristranost.

Stoga, kako bismo izbjegli degenerirane slučajeve, možemo naslutiti da je potrebno odabrati procjenitelja optimalne složenosti, odnosno u pravoj mjeri ga prilagoditi podacima kako bi izraz 1.16 bio što manji, a on će to biti ukoliko dovoljno dobro izbalansiramo dva obrnuto proporcionalna izraza, pristranost i varijancu. To nastojanje možemo vidjeti na slici 1.4, a to balansiranje ćemo implicitno učiniti koristeći metode unakrsne validacije koje će nam pomoći odabirati modele optimalne složenosti.



Slika 1.4: Rastav ukupne pogreške na dvije komponente, pristranost i varijancu, prikazan u ovisnosti o složenosti modela [5]

[5] https://commons.wikimedia.org/wiki/File:Bias_and_variance_contributing_to_total_error.svg

Poglavlje 2

Odabrane metode s primjerima

Kroz ovo poglavlje prolazit ćemo kroz odabrane metode neparametarske regresije zajedno s ilustrativnim primjerima koji će u većoj mjeri pratiti bilješke načinjene za potrebe predavanja na University of Minnesota [2]. Definicije i rezultati iz ovog poglavlja temeljeni su na [7].

2.1 Metoda lokalnih prosjeka

Metoda lokalnih prosjeka (eng. *local averages*)

Prva metoda koju ćemo predstaviti u ovom diplomskom radu naziva se metodom lokalnih prosjeka kojoj je cilj uzeti segment radijusa $h > 0$ te procjenjivati nepoznate vrijednosti y^* kao aritmetičku sredinu onih točaka koje se nalaze u okolini x^* , odnosno unutar segmenta $[x^* - h, x^* + h]$.

Definicija 2.1.1. *Fiksirajmo širinu pojasa $h > 0$ te definirajmo skup $B_x = \{i : |x_i - x| \leq h\}$. Neka je n_x broj točaka u B_x. Za svaki x za koji je $n_x > 0$ definiramo*

$$\hat{f}_n(x) = \frac{1}{n_x} \sum_{i \in B_x} Y_i. \tag{2.1}$$

Ovaj procjenitelj nazivamo procjeniteljem metodom lokalnih prosjeka od $f(x)$, što je poseban slučaj linearnih procjenitelja koje smo predstavili u prethodnom poglavlju.

Napomena 2.1.2. *Kako bi se u to uvjerili, primijetimo da $\hat{f}_n(x)$ možemo zapisati kao*

$$\hat{f}_n(x) = \sum_{i=1}^{n} Y_i l_i(x), \tag{2.2}$$

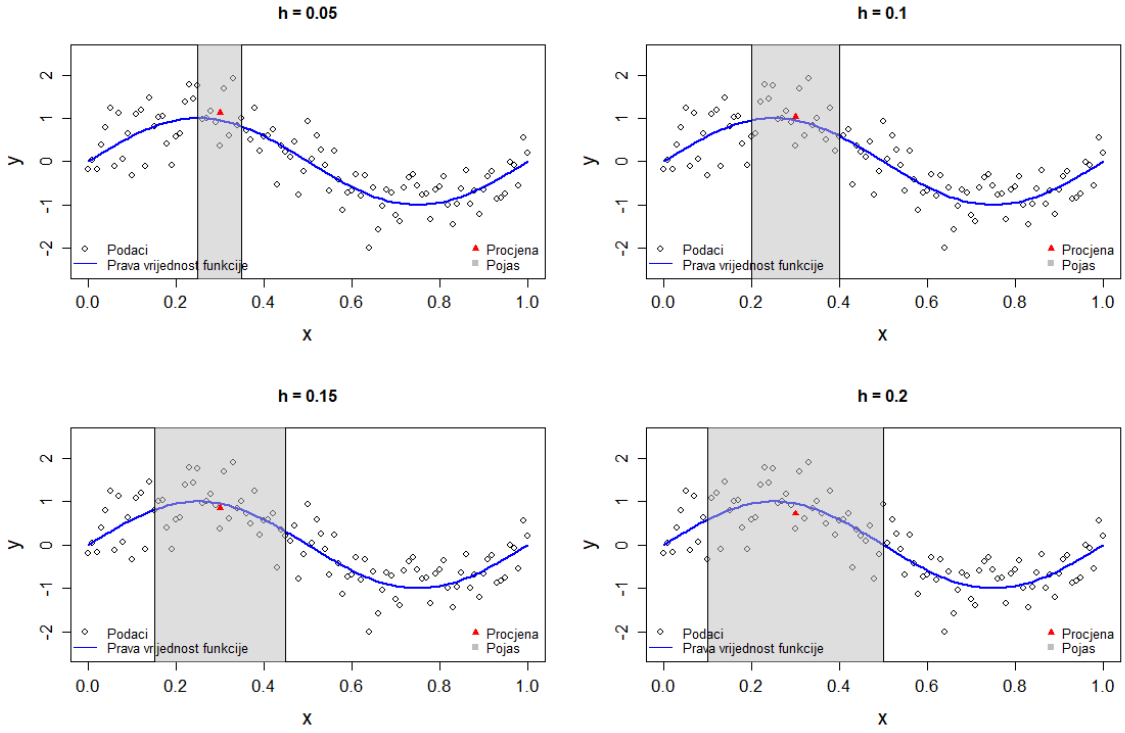
pri čemu je

$$l_i(x) = \begin{cases} \frac{1}{n_x}, & ako\ je\ |x_i - x| \leq h \\ 0, & inače \end{cases}. \tag{2.3}$$

Primjer 2.1.3. *Generirat ćemo dvodimenzionalne podatke, pri čemu će prva koordinata predstavljati podatke generirane iz uniformne distribucije na* $[0, 1]$*, a druga koordinata će pratiti sljedeću transformaciju sinusa prilagođenu za nasumičan šum:*

$$f(x) = \sin(2\pi x). \tag{2.4}$$

Na grafičkom prikazu ćemo uz podatke prikazati i gornju funkciju iz koje su generirani podaci te ćemo korištenjem metode lokalnih prosjeka procijeniti "nepoznatu" vrijednost funkcije za $x^* = 0.3$*.*



Slika 2.1: Metoda lokalnih prosjeka

Kako bi na podacima primijenili metodu lokalnih prosjeka, fiksirat ćemo različite širine intervala $h > 0$ *te ćemo za svaku od njih pokušati procijeniti nepoznatu vrijednost funkcije za* $x^* = 0.3$*, a to ćemo učiniti uzimanjem prosjeka po svim točkama koje upadaju u interval radijusa* h *oko točke* $x^* = 0.3$ *(sivo područje na svakom podsegmentu slike 2.1).*

Nakon što primijenimo gore opisani algoritam, za svaki h ćemo općenito dobiti drugu procjenu za y^* (crveni trokutić), jer smo y^* procjenjivali uzimajući prosjek većeg ili manjeg broja točaka. Kako bi algoritmom što bolje procjenjivali nove vrijednosti, potrebno je "optimalno" odabrati širinu intervala h - što će biti tema sljedećeg pododjeljka.

Odabir parametara procjenitelja

U primjeru predstavljenom u prethodnom pododjeljku mogli smo naslutiti da svaki od neparametarskih procjenitelja u znatnoj mjeri ovisi o vrijednosti koju nazivamo parametar procjenitelja. U ovom odjeljku predstavit ćemo pomoćnu metodu koju ćemo koristiti za odabir najpogodnijeg parametra procjenitelja.

No prije predstavljanja same metode, prvo ćemo prikazati izraz koji korištenjem metode želimo minimizirati:

$$R(h) = \mathbb{E}\left(\frac{1}{n}\sum_{i=1}^{n}(\hat{f}_n(x_i) - f(x_i))^2\right). \tag{2.5}$$

Dakle, želimo pronaći parametar h koji će odrediti oblik $\hat{f}_n$ te koji će minimizirati očekivanu prosječnu kvadratnu grešku. Idealno bi bilo kada bismo mogli odabrati h koji će minimizirati $R(h)$, no to nije moguće direktno učiniti budući da $R(h)$ ovisi o nepoznatoj regresijskoj funkciji $f(x)$.

Stoga, mogli bismo pokušati zamijeniti $R(h)$ prosječnom kvadratnom greškom na skupu dostupnih podataka, koju nazivamo i greškom treniranja:

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{f}_n(x_i))^2. \tag{2.6}$$

No takva zamjena je nedovoljno dobra, budući da podcjenjuje pravu vrijednost $R(h)$ koja je očekivanje izraza unutar zagrada, a mi smo našu zamjenu odabrali kao najmanju moguću realiziranu vrijednost unutar zagrada, što ne opisuje dovoljno dobro očekivanje. Stoga će i h koji na taj način odaberemo zajedno sa procjeniteljem kojeg na taj način dobijemo biti previše pristran i previše prilagođen podacima.

Kao alternativu, umjesto pokušaja zamjene 2.6, minimizirat ćemo sličan izraz kako bi dobili h, koji će isto tako mjeriti prosječnu kvadratnu grešku na skupu dostupnih podataka, no ovoga puta će ta greška biti između vrijednosti Y_i i pomoćnog procjenitelja koji je oblikovan na dostupnim podacima bez i-tog para (x_i, Y_i).

Definicija 2.1.4. *Vrijednost leave-one-out*[1] *unakrsne validacije definiramo kao*

$$CV = R(\hat{h}) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{f}_{(-i)}(x_i))^2, \tag{2.7}$$

pri čemu je $\hat{f}_{(-i)}$ *procjenitelj dobiven ispuštanjem i-tog para* (x_i, Y_i)*, tj.*

$$\hat{f}_{(-i)}(x) = \sum_{j=1}^{n} Y_j l_{j,(-i)}(x), \tag{2.8}$$

gdje je

$$l_{j,(-i)}(x) = \begin{cases} 0, & ako\ je\ j = i \\ \frac{l_j(x)}{\sum_{k\neq i} l_k(x)}, & ako\ je j \neq i \end{cases}. \tag{2.9}$$

Ukoliko za zamjenu $R(h)$ odaberemo *vrijednost leave-one-out unakrsne validacije*, h koji ćemo odabrati i procjenitelj kojeg ćemo dobiti bit će prilagođeniji općenitijem slučaju (nepoznatim podacima), jer svaki sumand unutar izraza 2.7 prikazuje odnos parametra h i namjerno skrivenog (nepoznatog) podatka.

Napomena 2.1.5. *Pomoćna metoda kojom određujemo parametar h minimiziranjem gornje vrijednosti naziva se leave-one-out unakrsnom validacijom, što je jedna od mnogih metoda unakrsne validacije.*

U završnim primjerima 3.1.1 i 3.2.1 u potpunosti ćemo ilustrirati metodu lokalnih prosjeka, pri čemu ćemo koristiti pomoćnu metodu leave-one-out unakrsne validacije kako bismo odabrali optimalan parametar h.

2.2 Metoda procjene jezgrom

Metoda procjene jezgrom (eng. *kernel smoothing*)

Kako bismo mogli procijeniti regresijsku funkciju metodom procjene jezgrom, ponajprije uvodimo jezgru kao glatku funkciju s određenim svojstvima te navodimo najpoznatije primjere jezgara zajedno s grafičkim prikazom kako bismo mogli intuitivnije shvatiti predstavljeni pojam.

Definicija 2.2.1. *Jezgru definiramo kao glatku funkciju K, takvu da je* $K(x) \geq 0$ *te za koju vrijedi*

$$\int K(x)dx = 1, \tag{2.10}$$

[1] u slobodnom prijevodu na hrvatski jezik *izostavi-jednog-van*

$$\int xK(X)dx = 0, \tag{2.11}$$

i

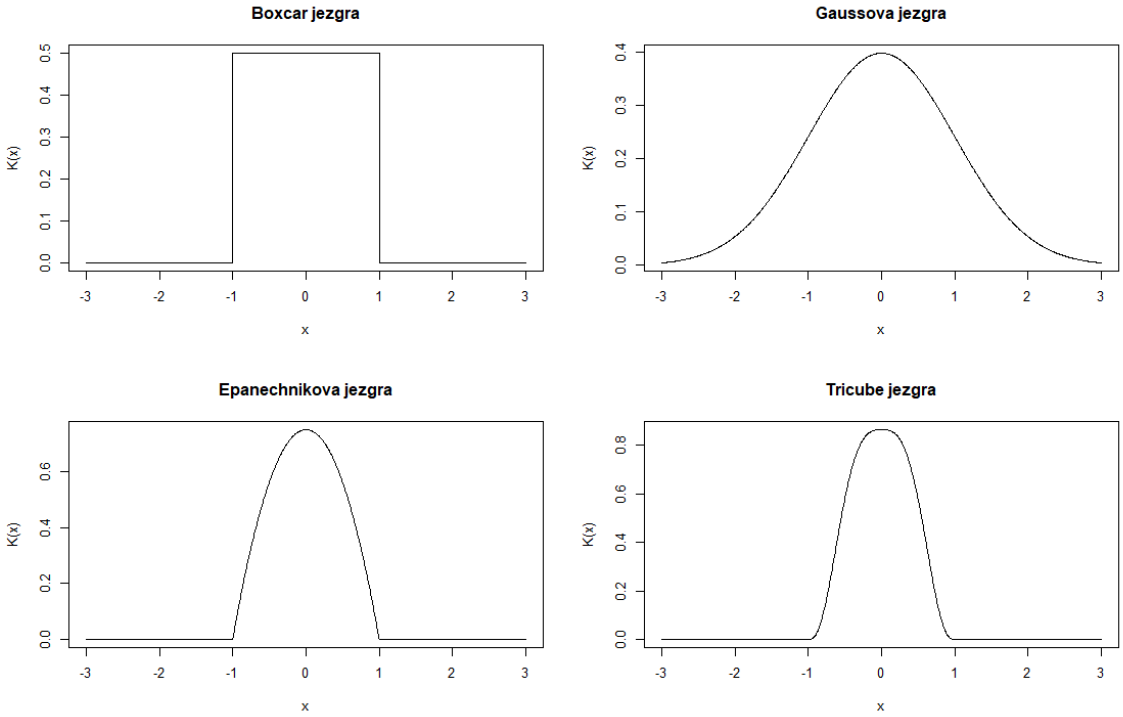
$$\sigma_K^2 \equiv \int x^2 K(x)dx > 0. \tag{2.12}$$

Primjer 2.2.2. *U ovom primjeru, odnosno na slici 2.2 prikazat ćemo najčešće korištene jezgre u metodi procjene jezgrom, a to su:*

***Boxcar** jezgra*	$K(x) = \frac{1}{2}I(x)$
***Gaussova** jezgra*	$K(x) = \frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}$
***Epanechnikova** jezgra*	$K(x) = \frac{3}{4}(1-x^2)I(x)$
***Tricube** jezgra*	$K(x) = \frac{70}{81}(1-\lvert x\rvert^3)^3 I(x)$

pri čemu je:

$$I(x) = \begin{cases} 1, & ako\ je\ |x| \leq 1 \\ 0, & ako\ je\ |x| > 1 \end{cases}.$$



Slika 2.2: Grafički prikaz najčešće korištenih jezgara u metodi procjene jezgrom

Nakon definicije jezgre, promotrimo poseban slučaj procjenitelja jezgrom nazvanim Nadaraya-Watsonovim procjeniteljem jezgrom koji procjenitelja regresijske funkcije $f(x)$ oblikuje kao težinski prosjek Y_i-eva, pri čemu veće težine (veću važnost) pridodaje onim točkama bližima x.

Definicija 2.2.3. *Neka je $h > 0$ pozitivan realan broj kojeg nazivamo širina.* ***Nadaraya-Watson*** *procjenitelj jezgrom definiramo kao:*

$$\hat{f}_n(x) = \sum_{i=1}^{n} l_i(x) Y_i, \tag{2.13}$$

pri čemu je K jezgra iz definicije 2.2.1, a težine $l_i(x)$ su dane s

$$l_i(x) = \frac{K(\frac{x-x_i}{h})}{\sum_{j=1}^{n} K(\frac{x-x_j}{h})}. \tag{2.14}$$

U sljedećem primjeru grafički ćemo prikazati na koji način širina $h > 0$ iz prethodne definicije utječe na težine (važnost) koje pridodajemo svakom Y_i-u pri izračunu prosjeka.

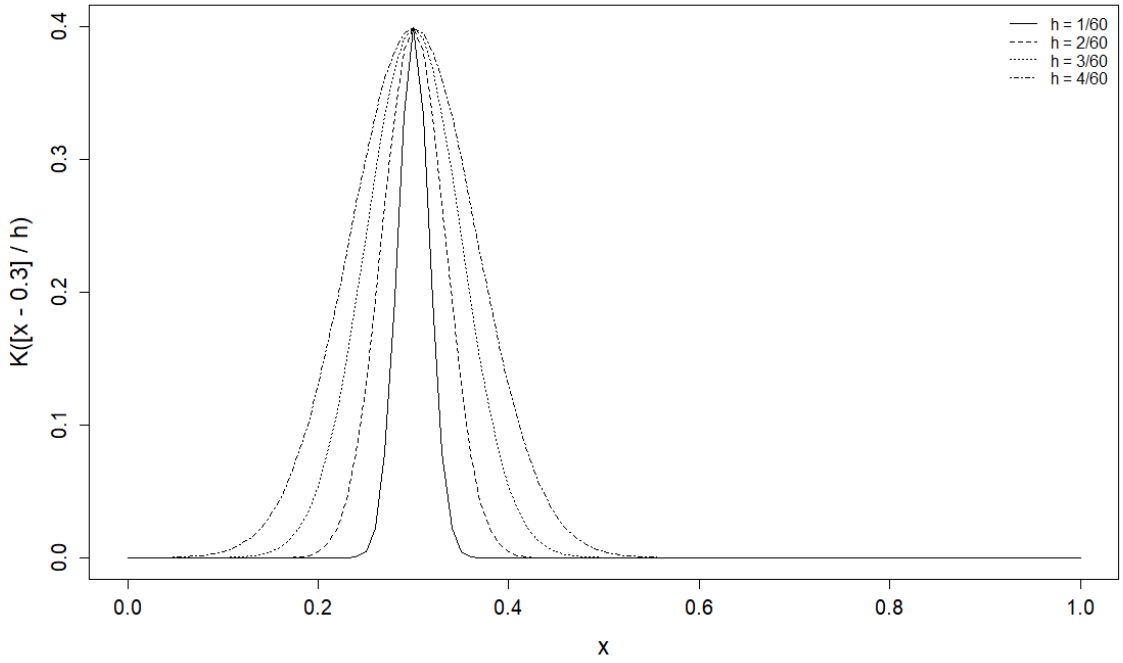
Primjer 2.2.4. *Pretpostavimo da imamo regresijski problem u kojem želimo procijeniti vrijednost nepoznate odzivne varijable za $x^* = 0.3$ te da smo se odlučili koristiti Nadaraya-Watsonovog procjenitelja jezgrom. Ukoliko za rješavanje danog problema odaberemo Gaussovu jezgru, tada na slici 2.3 možemo vidjeti koliku težinu (važnost) Gaussova jezgra pridodaje kojoj točki pri izračunu težinskog prosjeka. Iz slike 2.3 možemo primijetiti da neovisno o širini h, najveće težine (važnost) se pridodaju onim točkama u neposrednoj blizini točke $x^* = 0.3$. Isto tako, možemo naslutiti da kako $h \to 0$ procjenitelj jezgrom pri formiranju svoje procjene koristi samo one Y_i-eve koji su jako blizu x te isto tako da kako $h \to \infty$, procjenitelj jezgrom koristi i one Y_i-eve koje su relativno daleko od $x^* = 0.3$.*

Nakon što smo u prethodnom primjeru uvidjeli kolike težine (važnost) Gaussova jezgra pridodaje kojoj točki u blizini $x^* = 0.3$ pri formiranju procjene, nadopunimo zadobivenu intuiciju konkretnim formiranjem procjene za različite parametre širine $h > 0$.

Primjer 2.2.5. *Primjer ćemo prikazati na identičnim dvodimenzionalnim podacima kao u prethodnom potpoglavlju, u kojima je prva koordinata generirana iz uniformne distribucije na* $[0, 1]$*, a druga koordinata generirana iz sljedeće transformacije sinusa prilagođene za nasumičan šum:*

$$f(x) = \sin(2\pi x). \tag{2.15}$$

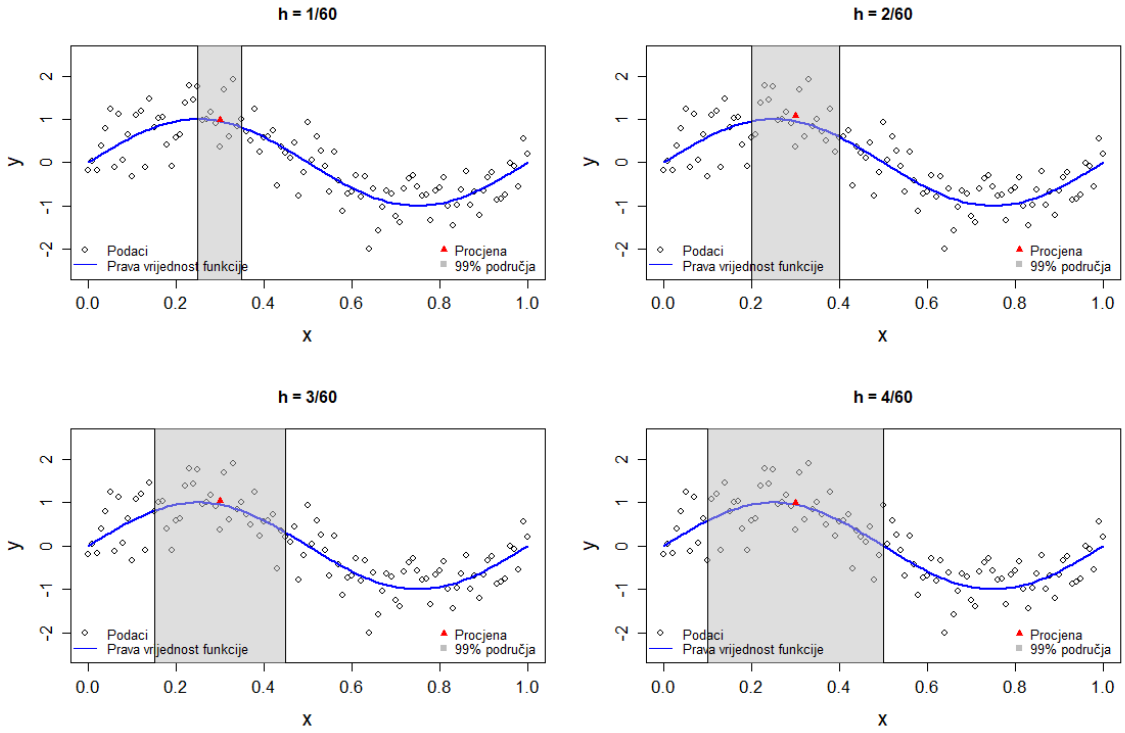
Na grafičkom prikazu ćemo uz podatke prikazati i gornju funkciju iz koje su generirani podaci te ćemo korištenjem metode procjene jezgrom procijeniti "nepoznatu" vrijednost funkcije za $x^* = 0.3$.



Slika 2.3: Grafički prikazane težine pri korištenju Gaussove jezgre u svrhu procjene odzivne varijable za vrijednost $x^* = 0.3$ za različite širine $h > 0$.

Kako bi na podacima primijenili metodu procjene jezgrom, fiksirat ćemo različite širine $h > 0$ *te ćemo za svaku od njih pokušati procijeniti nepoznatu vrijednost funkcije za* $x^* = 0.3$*, a to ćemo učiniti uzimanjem težinskog prosjeka pri čemu ćemo težine dodjeljivati koristeći Gaussovu jezgru predstavljenu u primjeru 2.2.2. Dodijeljene težine možemo vidjeti na slici 2.3, a sivim područjem na svakom podsegmentu slike 2.4 prikazano je područje zajedno sa točkama koje su unutar 99% površine ispod krivulje težina na slici 2.3 (one točke sa značajnijom težinom).*

Nakon što primijenimo gore opisani algoritam, za svaki h ćemo općenito dobiti drugu procjenu za y^* *(crveni trokutić), jer smo* y^* *procjenjivali uzimajući težinski prosjek pri čemu smo svakoj točki općenito dodjeljivali različitu težinu. Kako bi algoritmom što bolje procjenjivali nove vrijednosti, potrebno je "optimalno" odabrati širinu intervala h - što će biti tema sljedećeg pododjeljka.*



Slika 2.4: Metoda procjene jezgrom

Odabir širine

U odjeljku u kojem smo predstavili metodu lokalnih prosjeka, predstavili smo i pomoćnu metodu leave-one-out unakrsne validacije koja nam je koristila za odabir parametra procjenitelja. U ovom pododjeljku predstavit ćemo općenitiju pomoćnu metodu, prije čijeg ćemo uvođenja predstaviti motivacijski teorem bez dokaza.

Teorem 2.2.6. *Neka je $\hat{f}_n$ linearan procjenitelj. Tada vrijednost leave-one-out unakrsne validacije $\hat{R}(h)$ možemo zapisati kao*

$$\hat{R}(h) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i - \hat{f}_n(x_i)}{1 - L_{ii}}\right)^2, \tag{2.16}$$

pri čemu je $L_{ii} = l_i(x_i)$ i-ti dijagonalni element matrice izglađivanja L.

Slično kao što je leave-one-out unakrsna validacija imala za cilj odabrati h koji će minimizirati izraz 2.16, tako će i generalizirana unakrsna validacija imati za cilj minimizirati sličan izraz, u kojemu će vrijednost L_{ii} (koliko i-ta kovarijata x_i utječe pri formiranju

procjene $\hat{f}(x_i)$) biti zamijenjena uprosječenom vrijednosti $n^{-1}\sum_{i=1}^{n} L_{ii} = v/n$, pri čemu $v = tr(L)$ prikazuje efektivne stupnjeve slobode.

Definicija 2.2.7. *Vrijednost generalizirane unakrsne validacije definiramo kao*

$$GCV(h) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i - \hat{f}_n(x_i)}{1 - \frac{v}{n}}\right)^2, \tag{2.17}$$

pri čemu v prikazuje efektivne stupnjeve slobode, odnosno trag matrice L.

Napomena 2.2.8. *Pomoćna metoda kojom odabiremo h minimizacijom vrijednosti 2.17 naziva se metodom generalizirane unakrsne validacije.*

Upravo tu pomoćnu metodu generalizirane unakrsne validacije koristit ćemo u završnim primjerima 3.1.1 i 3.2.1 za odabir optimalnog parametra širine h kako bismo u potpunosti ilustrirali metodu procjene jezgrom.

2.3 Metoda lokalne linearne regresije

Metoda lokalne linearne regresije (eng. *local linear regression*)

U ovom odjeljku promotrit ćemo generalizaciju metode procjene jezgrama zvanu metodom lokalne linearne regresije. Kako bi motivirali uvođenje ove metode promotrimo konstantnog procjenitelja $\hat{f}_n(x) \equiv a$ koji minimizira sumu kvadrata grešaka, tj.

$$\sum_{i=1}^{n}(Y_i - a)^2. \tag{2.18}$$

Rješenje tog problema je procjenitelj $\hat{f}_n(x) = \overline{Y}$, koji očito nije dobar procjenitelj regresijske funkcije $f(x)$.
Nadalje, ukoliko definiramo težinsku funkciju koristeći pojam jezgre definirane u prošlom odjeljku $w_i(x) = K(\frac{x_i - x}{h})$ i definiramo konstantnog procjenitelja $\hat{f}_n(x) \equiv a$ koji minimizira težinsku sumu kvadrata grešaka:

$$\sum_{i=1}^{n} w_i(x)(Y_i - a)^2, \tag{2.19}$$

izvođenjem elementarnih operacija dolazimo do procjenitelja:

$$\hat{f}_n(x) = \frac{\sum_{i=1}^{n} w_i(x) Y_i}{\sum_{i=1}^{n} w_i(x)}, \tag{2.20}$$

što je upravo procjenitelj metodom procjenitelja jezgrama. Nadalje, gornjom diskusijom dolazimo do zanimljive interpretacije procjenitelja metodom procjenitelja jezgrama, a to je da je on lokalno konstantan procjenitelj - dobiven dodjeljivanjem težina metodi najmanjih kvadrata.

To nas potiče da poboljšamo procjenitelja na način da konstantu a u izrazu 2.19 zamjenimo nekom drugom funkciju, u našem slučaju linearnom funkcijom, čime dolazimo do izraza:

$$\sum_{i=1}^{n} w_i(x)(Y_i - (a_{0,x} + a_{1,x}(x_i - x)))^2. \tag{2.21}$$

Minimiziranjem tog izraza dolazimo do procjenitelja koji se pokušava približiti linearnoj funkciji u maloj okolini svake točke, odnosno do procjenitelja metodom lokalne linearne regresije.

Budući da je u gornjem izrazu implicitno skriven parametar h unutar izraza za težinu $w_i(x)$ o kojem procjenitelj ovisi, riječ je o metodi neparametarske regresije, jer je istovremeno potrebno odrediti parametar h koji diktira oblik funkcije i procijeniti dva parametra $a_{0,x}$ i $a_{1,x}$ o kojima unutarnja linearna funkcija ovisi.

Pokažimo formalno da je riječ o metodi neparametarske regresije, odnosno za fiksan h izvedimo izraz za procjenitelja te se uvjerimo da je on linearni procjenitelj definiran u definiciji 1.4.1.

Teorem 2.3.1. *Procjenitelj metodom lokalne linearne regresije je*

$$\hat{f}_n(x) = \sum_{i=1}^{n} l_i(x) Y_i, \tag{2.22}$$

pri čemu je $l(x)^T = (l_1(x), ..., l_n(x))$ *vektor oblika:*

$$l(x)^T = e_1^T (X_x^T W_x X_x)^{-1} X_x^T W_x. \tag{2.23}$$

Nadalje, definiramo $e_1 = (1, 0)^T$, $a = (a_{0,x}, a_{1,x})$ *te* X_x *i* W_x *kao:*

$$X_x = \begin{bmatrix} 1 & x_1 - x \\ 1 & x_2 - x \\ \vdots & \vdots \\ 1 & x_n - x \end{bmatrix}, \tag{2.24}$$

$$W_x = \begin{bmatrix} w_1(x) & 0 & \cdots & 0 \\ 0 & w_2(x) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_n(x) \end{bmatrix}. \tag{2.25}$$

Dokaz. Primijetimo da smo procjenitelja metodom lokalne linearne regresije definirali kao onog koji minimizira izraz 2.21. Taj izraz možemo ljepše zapisati kao:

$$(Y - X_x a)^T W_x (Y - X_x a). \tag{2.26}$$

Primjenjujući pravila transponiranja i matričnog množenja te primjećujući jednakost središnjih članova, dolazimo do izraza:

$$Y^T W_x Y - 2Y^T W_x X_x a + a^T X_x^T W_x X_x a. \tag{2.27}$$

Kako bismo minimizirali ovaj izraz, moramo pronaći a koji minimizira ovu kvadratnu jednadžbu. Njega ćemo pronaći ukoliko riješimo sustav jednadžbi u kojima ćemo izjednačiti parcijalne derivacije po a sa 0.

$$\frac{\partial}{\partial a} = -2X_x^T W_x Y + 2X_x^T W_x X_x a. \tag{2.28}$$

$$-2X_x^T W_x Y + 2X_x^T W_x X_x a = 0. \tag{2.29}$$

$$2X_x^T W_x X_x a = 2X_x^T W_x Y. \tag{2.30}$$

Konačno podijelit ćemo obje strane jednadžbe s $2X_x^T W_x X_x$ kako bismo dobili procjenitelja a:

$$a = (X_x^T W_x X_x)^{-1} X_x^T W_x Y. \tag{2.31}$$

Dobiveni a je minimum kvadratne forme 2.27, budući da je da je riječ o kvadratnoj formi po a, pri čemu je izraz $X_x^T W_x X_x$ pozitivno definitan.

Budući da je dobiveni procjenitelj a oblikovan kako bi procijenio vrijednost funkcije u točki x te kako je u jednadžbi 2.21 u svakom sumandu prisutan opis ponašanja procjenitelja u okolini točke x_i dolazimo do zaključka da će se u okolini točke x procjenitelj ponašati kao (ukoliko zamijenimo x_i s x):

$$\hat{f}_n(x) = a_{0,x},$$

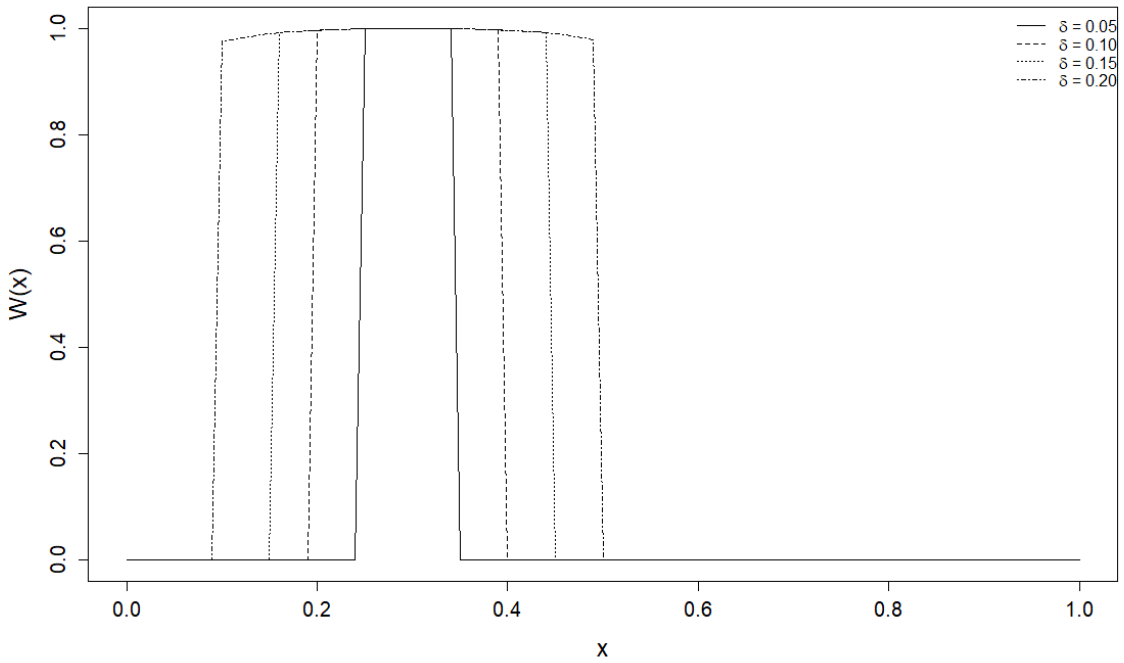
čime dolazimo do izraza 2.23. □

Napomena 2.3.2. *Iako smo u ovom potpoglavlju iz pedagoških razloga težinu definirali kao $w_i(x) = K(\frac{x_i - x}{h})$, odnosno kao transformaciju neke od jezgara, u primjeru koji slijedi odlučit ćemo se definirati ju preko tricube funkcije:*

$$w_i(x) = \begin{cases} (1 - |x - x_i|^3)^3, & \textit{ako je } |x - x_i| < \delta \\ 0, & \textit{inače} \end{cases}.$$

Tricube funkcija dodjeljuje težine, odnosno opisuje koliki utjecaj ima pogreška učinjena u kojoj točki pri formiranju procjene za x. Na slici 2.5 možemo vidjeti kolike težine dodjeljuje

tricube funkcija kojoj točki pri formiranju procjene za $x = 0.3$ u ovisnosti o širini intervala δ.



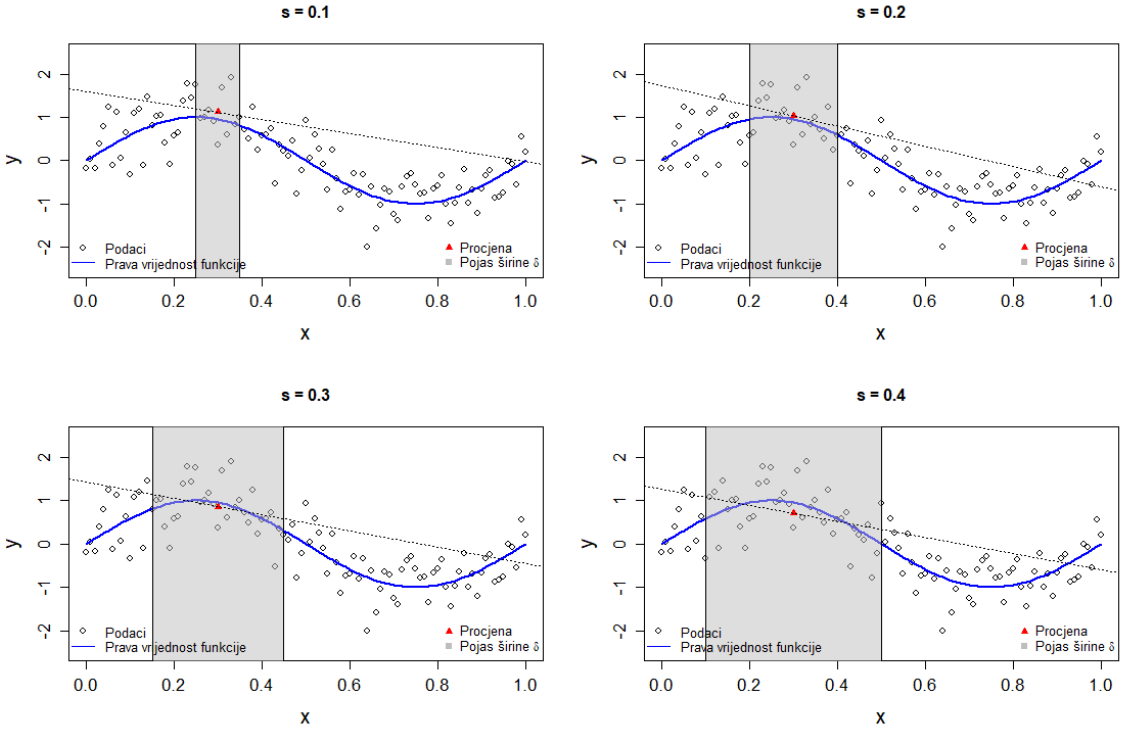
Slika 2.5: Težine koje dodjeljuje tricube funkcija pri formiranju procjene u $x = 0.3$

Isto tako, kao što je u teoretskom dijelu $h > 0$ određivao oblik težinske funkcije i samog procjenitelja, tako će i u primjerima raspon $s \in (0, 1]$ određivati oblik težinske funkcije i samog procjenitelja na način da će predstavljati proporciju točaka sn koje imaju pozitivnu težinu. Dakle iz parametra $s \in (0, 1]$ konstruirat ćemo potrebnu širinu intervala δ za tricube funkciju koja će sadržavati sn točaka oko x.

Primjer 2.3.3. *Primjer ćemo prikazati na identičnim dvodimenzionalnim podacima kao u prethodnom potpoglavlju, u kojima je prva koordinata generirana iz uniformne distribucije na* $[0, 1]$*, a druga koordinata generirana iz sljedeće transformacije sinusa prilagođene za nasumičan šum:*

$$f(x) = \sin(2\pi x). \tag{2.32}$$

Na grafičkom prikazu ćemo uz podatke prikazati i gornju funkciju iz koje su generirani podaci te ćemo korištenjem metode lokalne linearne regresije procijeniti "nepoznatu" vrijednost funkcije za $x^* = 0.3$.



Slika 2.6: Metoda lokalne linearne regresije

Kako bismo na podacima primijenili metodu lokalne linearne regresije, fiksirat ćemo različite proporcije $s \in (0, 1]$ *te ćemo za svaku od njih pokušati procijeniti nepoznatu vrijednost funkcije za* $x^* = 0.3$. *Kako bismo to učinili, prvo ćemo konstruirati interval radijusa* δ *oko točke* $x^* = 0.3$ *koji će sadržavati sn najbližih točaka oko* x^* *(sivo područje na svakom podsegmentu slike 2.6). Zbog uniformnosti podataka na* $[0, 1]$ *taj će* δ *iznositi* $s/2$. *Definiranjem* δ *jedinstveno određujemo težine pa možemo primijeniti teoretski rezultat iz teorema 2.3.1 pri izračunu procjene.*

Nakon što primijenimo gore opisani algoritam, za svaku proporciju s ćemo općenito dobiti drugu procjenu za y^* *(crveni trokutić), jer smo pri procjenjivanju* y^* *različitim točkama općenito dodjeljivali veću ili manju težinu (važnost). Kako bi algoritmom što bolje procjenjivali nove vrijednosti, potrebno je "optimalno" odabrati proporciju s - što će biti tema sljedećeg pododjeljka.*

Odabir proporcije

U odjeljku u kojem smo predstavili metodu procjene jezgrom, predstavili smo i pomoćnu metodu generalizirane unakrsne validacije koja nam je koristila za odabir parametra procjenitelja. Upravo tu pomoćnu metodu generalizirane unakrsne validacije koristit ćemo u završnim primjerima 3.1.1 i 3.2.1 za odabir optimalnog parametra proporcije s kako bismo u potpunosti ilustrirali metodu lokalne linearne regresije.

Poglavlje 3

Završni primjeri

U ovom poglavlju usporedit ćemo procjenitelje dobivene korištenjem metoda iz prethodnog poglavlja na istim tim umjetno generiranim podacima. Uz to prikazat ćemo još jedan realističniji primjer u kojem nam regresijska funkcija nije poznata, a koji će u većoj mjeri pratiti bilješke načinjene za potrebe predavanja na University of Minnesota. [2]

3.1 Usporedba modela

Primjer 3.1.1. *Primjer ćemo prikazati na identičnim dvodimenzionalnim podacima kao u prethodnom poglavlju, u kojima je prva koordinata generirana iz uniformne distribucije na* $[0, 1]$*, a druga koordinata generirana iz sljedeće transformacije sinusa prilagođene za nasumičan šum:*

$$f(x) = \sin(2\pi x). \tag{3.1}$$

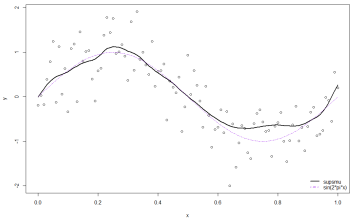
Dakle, generirali smo podatke iz funkcije prikazane u 3.1, pri čemu smo drugoj koordinati umjetno nadodali slučajan šum (što bi mogli smatrati greškom mjerenja u nekom realističnijem primjeru).

U ovom primjeru ćemo koristeći te podatke formirati i grafički prikazati procjenitelje dobivene svakom od tri predstavljene metode. Uz to ćemo usporediti koji je od procjenitelja u prosjeku učinio najmanju kvadratnu grešku, ako greškom smatramo koliko je procjena procjenitelja odstupila od stvarnih vrijednosti (onih bez slučajnog šuma).

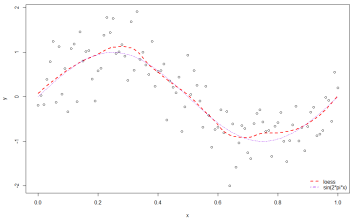
Koristeći pomoćne metode unakrsne validacije implementirali smo metode, pri čemu način implementacije s popratnom diskusijom možemo proučiti unutar sljedeća dva potpoglavlja.

Nakon što primijenimo svaku od predstavljenih metoda dolazimo do slike 3.1 na kojoj su prikazani odvojeno i istovremeno procjenitelji zajedno s regresijskom funkcijom.

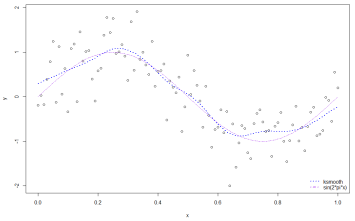
Detaljnijim promatranjem slike 3.1 možemo naslutiti da će velik utjecaj na prosječnu kvadratnu grešku imati područje oko vrijednosti $x = 0.8$*, budući da su tu odstupanja pro-*



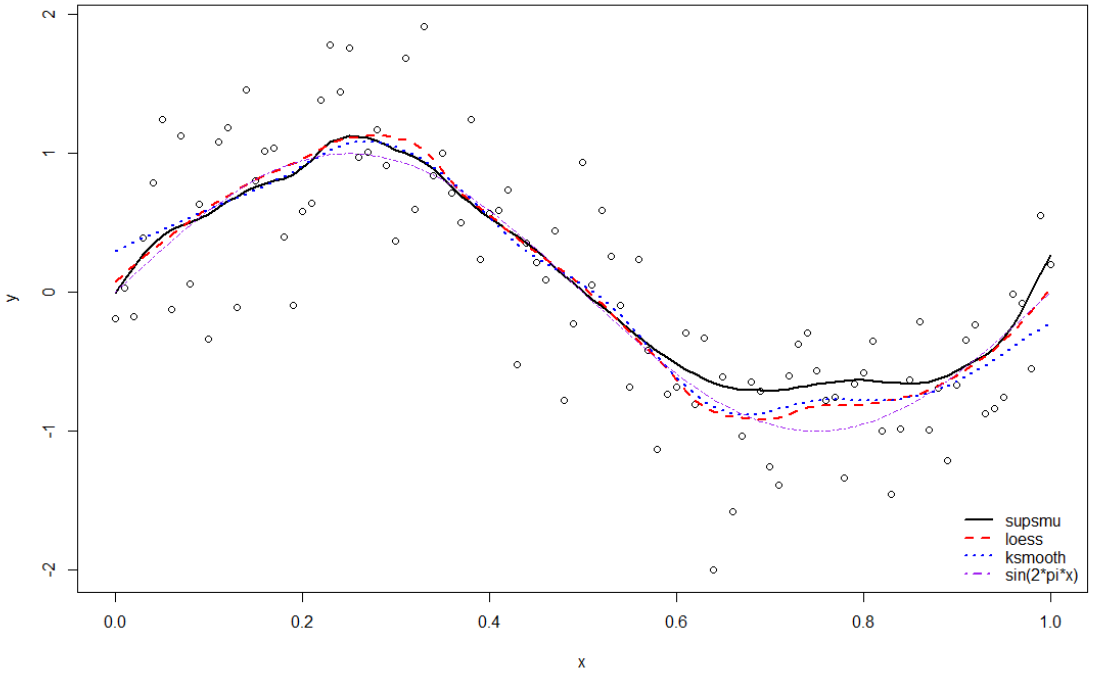
(a) Metoda lokalnih prosjeka



(b) Metoda lokalne regresije



(c) Metoda procjene jezgrom



(d) Prikaz primjene triju metoda zajedno s podacima

Slika 3.1: Grafički prikaz prilagođavanja podataka modelu

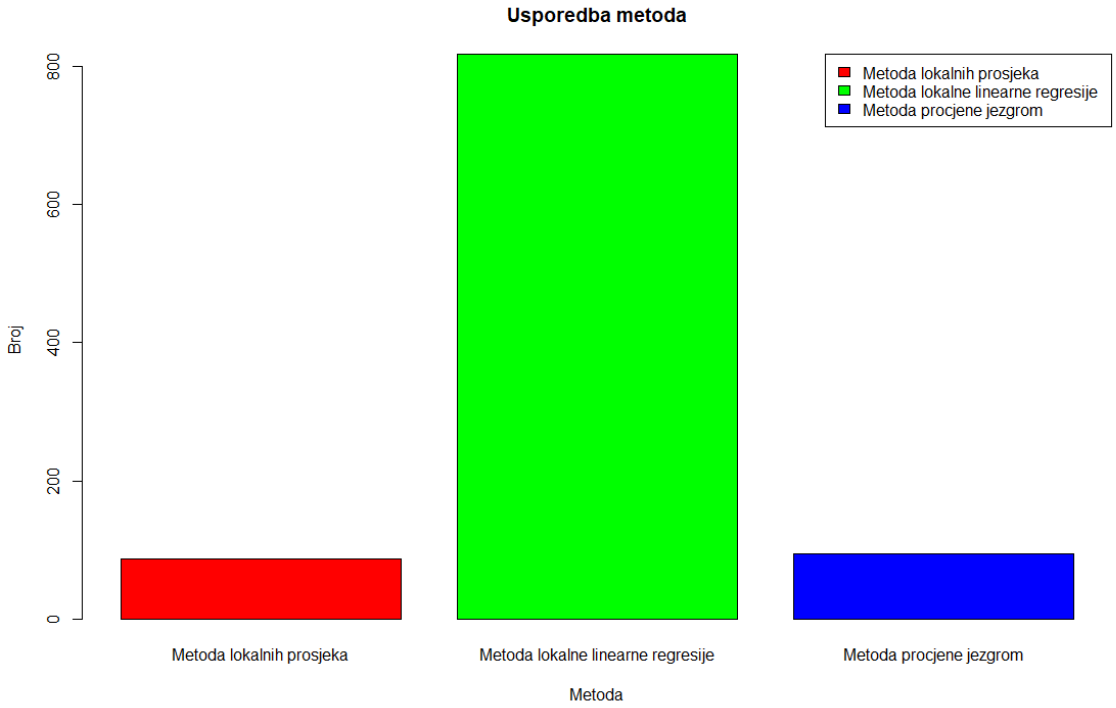
cjenitelja od regresijske funkcije najveća. Svoje slutnje možemo potvrditi sljedećim rangiranjem:

1. *Metoda lokalne linearne regresije - 0.006180024*
2. *Metoda procjene jezgrom - 0.01186096*
3. *Metoda lokalnih prosjeka - 0.01918818*

koje je dobiveno nakon izračuna učinjene prosječne kvadratne greške prikazane sljedećom formulom:

$$\frac{1}{n}\sum_{i=1}^{n}(\hat{f}_n(x_i) - f(x_i))^2. \tag{3.2}$$

Ako bismo svoje zaključivanje sprovodili samo na gore učinjenim kvadratnim greškama, tada bismo se odlučili odabrati procjenitelja dobivenog metodom lokalne linearne regresije. No kako bi naše zaključivanje bilo primjenjivo i na nekim drugim podacima, potrebno je generirati podatke N puta te usporediti u koliko je slučajeva koja od metoda postizala najbolje rezultate. Taj postupak učinili smo $N = 1000$ puta za istu tu sinusoidnu funkciju prikazanu formulom 3.1 te smo dobivene rezultate prikazali na slici 3.2.



Slika 3.2: Metoda lokalnih prosjeka

Promatranjem slike 3.2 možemo uvidjeti da bi sinusoidnu funkciju prikazanu formulom 3.1 najbolje bilo procjenjivati metodom lokalne linearne regresije.

Navedeno zaključivanje i modeliranje bi nam bilo korisno ako bismo željeli aproksimirati sporo i teško primjenjivu poznatu formulu za izračun vrijednosti neke fizikalne veličine nekom jednostavnijom aproksimacijom.

Budući da nam često regresijska funkcija neće biti poznata, u sljedećem potpoglavlju promotrit ćemo još jedan primjer u kojem ćemo prezentirati procjenitelje dobivene metodama neparametarske regresije te u kojem ćemo se argumentirano pokušati odlučiti za onaj najprikladniji.

3.2 Primjer povezanosti mjesečnih primanja i ugleda

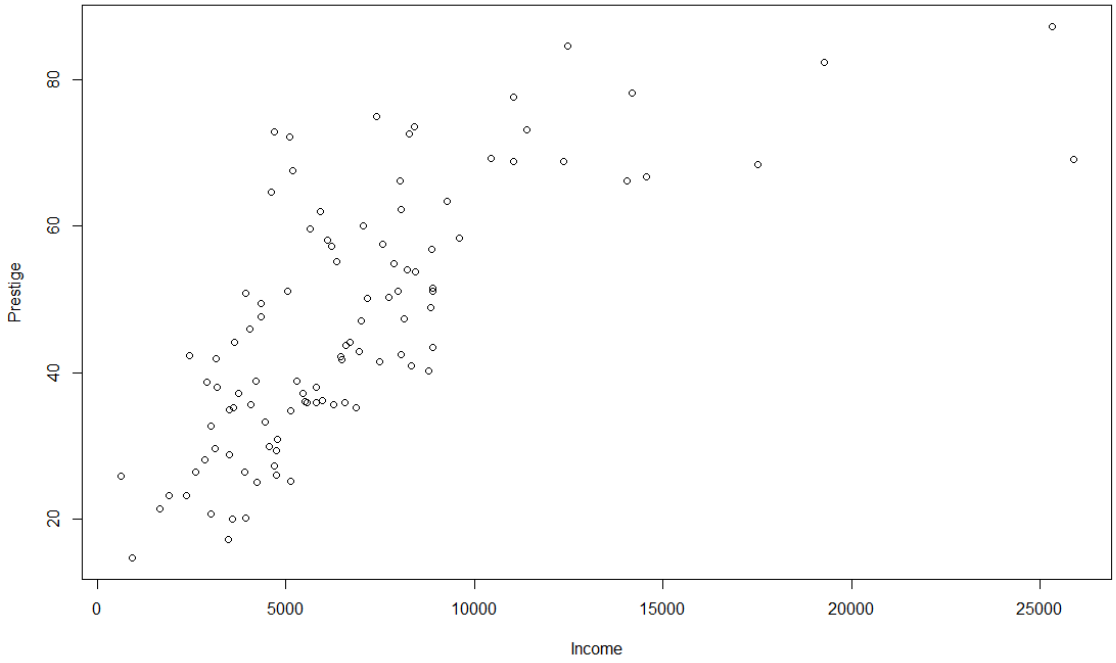
	education	income	women	prestige	census	type
gov.administrators	13.11	12351	11.16	68.8	1113	prof
general.managers	12.26	25879	4.02	69.1	1130	prof
accountants	12.77	9271	15.70	63.4	1171	prof
purchasing.officers	11.42	8865	9.11	56.8	1175	prof
chemists	14.62	8403	11.68	73.5	2111	prof
physicists	15.64	11030	5.13	77.6	2113	prof
biologists	15.09	8258	25.65	72.6	2133	prof
architects	15.44	14163	2.69	78.1	2141	prof
civil.engineers	14.52	11377	1.03	73.1	2143	prof
mining.engineers	14.64	11023	0.94	68.8	2153	prof

Slika 3.3: Prvih deset redaka tablice Prestige

Primjer 3.2.1. *Promotrimo podatke iz skupa podataka Prestige, koji se nalaze unutar paketa car[1]. Unutar skupa podataka nalaze se karakteristike svakog od 102 zanimanja koja su bila ponuđena kao odgovor na pitanje o zaposlenju tijekom popisa stanovništva u Kanadi 1971. godine. Točnije, za svako zanimanje možemo pronaći prosjek duljine trajanja edukacije, mjesečnih primanja u dolarima, postotka žena na određenom zanimanju zajedno s koeficijentom ugleda koji su ljudi dodjeljivali tom zanimanju, šifrom za identifikaciju i kategorijom zanimanja koja nam pomaže grupirati slične poslove prema razini odgovornosti. Detaljniji prikaz podataka unutar tablice Prestige možemo vidjeti na slici 3.3.*

U ovom primjeru nastojat ćemo povezati podatke o prosječnim mjesečnim primanjima svakog od 102 zanimanja zajedno s Pineo-Porter ocjenom ugleda - ocjenom između 10 i 90 koja odražava mišljenje generalne populacije o tom zanimanju.

Rezultati modeliranja i analiza ovakve tematike mogu biti zanimljivi psiholozima i sociolozima, kao i mladim ljudima koji svoje zanimanje žele odabrati uzimajući u obzir "mišljenje" i "potencijalno poštovanje" drugih.

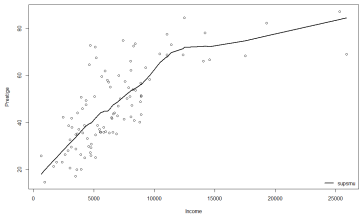


Slika 3.4: Raspršeni grafikon primanja i ugleda za svako od 102 zanimanja

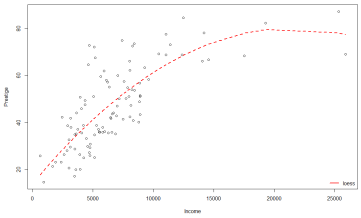
Odnos prihoda i ugleda za svako od 102 zanimanja možemo vidjeti na raspršenom grafikonu na slici 3.4. Prikazani odnos čini se nelinearnim, osim za zanimanja s prihodom manjim od $10000, za koja možemo pretpostaviti da postoji linearna veza između primanja i ugleda. No, za zanimanja koja zarađuju između $10000 i $25000 ta veza je slabija, nejasnija i manjeg nagiba ukoliko bi ju modelirali linearnim modelom. Stoga je smisleno odabrati neki nelinearni model kojim bi modelirali podatke za cjelokupan raspon mjesečnih primanja. U našem slučaju prikazat ćemo prilagodbu podataka svakom od tri već predstavljena modela neparametarske regresije te ćemo se promatrajući grafički prikaz odlučiti za onaj model koji "najprirodnije", tj. "najizglađenije" aproksimira taj odnos, odnosno s najboljim omjerom prilagodbe podacima i glatkoće dobivene funkcije.

Prije nego što prokomentiramo dobivene grafove na slici 3.5, objasnimo na koji način smo ih dobili. Na njima su prikazane procjene ugleda svakom od tri metode za već dostupna zanimanja koje smo onda međusobno spojili crtkastom linijom, pri čemu smo:

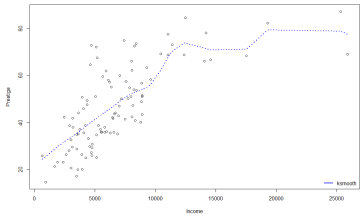
1. *Metodu lokalnih prosjeka sproveli koristeći već dostupnu funkciju supsmu()[5] koju smo koristili kako bismo odabrali optimalan parametar h koristeći leave-one-out unakrsnu validaciju te formirali dobivene procjene za optimalan parametar h.*



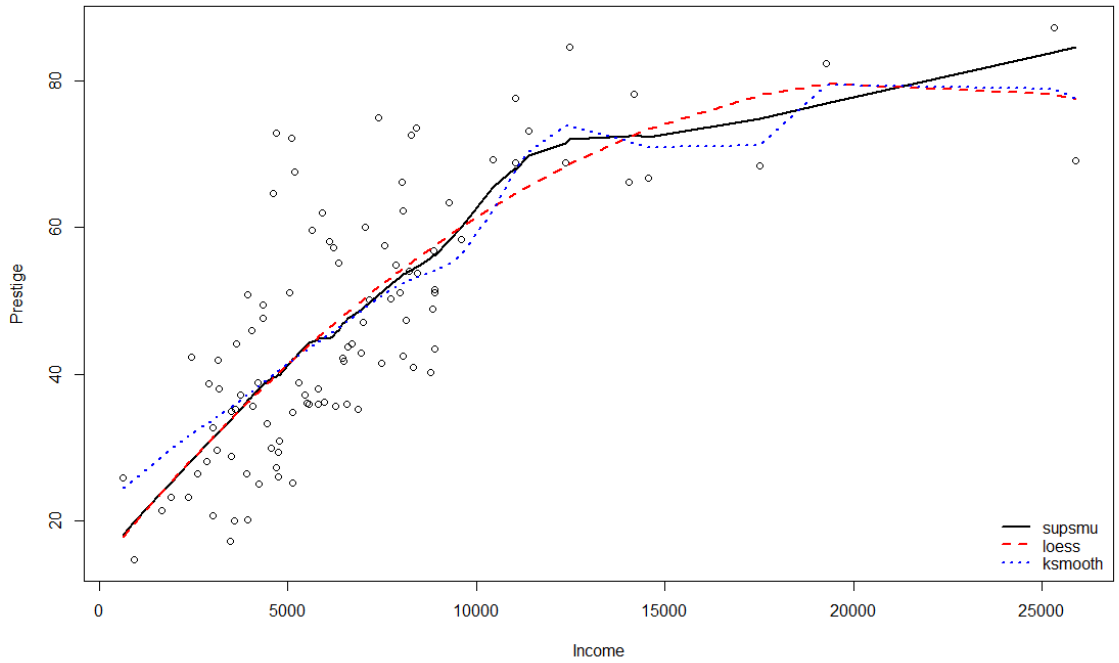
(a) Metoda lokalnih prosjeka



(b) Metoda lokalne regresije



(c) Metoda procjene jezgrom



(d) Prikaz primjene triju metoda zajedno s podacima

Slika 3.5: Grafički prikaz prilagođavanja podataka modelu

NAPOMENA: *Unutar funkcije supsmu() parametar h je nešto drugačije implementiran. Naime, u teoretskom dijelu predstavili smo h kao mjeru veličine okoline točke x unutar koje smo uzimali y vrijednosti za računanje prosjeka, dok je unutar funkcije supsmu() h predstavljen kao broj između* $(0, 1]$ *koji predstavlja proporciju najbližih točaka, tj. sn točaka čije ćemo y vrijednosti uzimati za računanje prosjeka.*

2. *Metodu lokalne linearne regresije sproveli koristeći pomoćnu funkciju loess.gcv(), u kojoj smo koristili dostupnu funkciju loess()[5] koja je težine dodjeljivala koristeći*

tricube funkciju te kojom smo odabrali optimalan parametar s koristeći generaliziranu unakrsnu validaciju te formirali dobivene procjene za optimalan parametar s.

NAPOMENA: *loess() funkcija koju smo upotrijebili koristi nešto općenitiju tricube funkciju od one koju smo koristili u ilustrativnom primjeru u prethodnome poglavlju. Točnije, ona koristi istu tu funkciju u kojoj je udaljenost $|x - x_i|$ skalirana za maksimalnu udaljenost δ (jer ovaj put podaci nisu na [0,1] kao u našem primjeru):*

$$w_i(x) = \begin{cases} (1 - |\frac{x-x_i}{\delta}|^3)^3, & \text{ako je } |x - x_i| < \delta \\ 0, & \text{inače} \end{cases}.$$

3. *Metodu procjene jezgrom sproveli koristeći pomoćnu funkciju ksmooth.gcv(), u kojoj smo koristili Gaussovu jezgru, odabrali optimalan parametar h koristeći generaliziranu unakrsnu validaciju te formirali dobivene procjene za optimalan parametar h.*

Promatrajući dobivene grafove možemo primijetiti da se onaj iscrtkan plavom linijom previše prilagođava primanjima većim od $15000, odnosno da je on procjenitelj čiji će izraz za varijancu unutar izraza za grešku 1.16 biti prevelik te da neće dobro procjenjivati ugled ljudi s primanjima većim od $15000.

Promatrajući crno i crveno iscrtakni graf na slici 3.5 ne možemo pronaći dovoljno snažan argument zašto bi se isključivo odlučili za jedan umjesto drugog. No ipak, kako bi naše izlaganje bilo potpuno, iznijet ćemo argument zašto bismo odabrali onaj iscrtkan crvenom linijom na slici 3.5.

Promatrajući graf iscrtkane crvene linije možemo primijetiti da je on najizglađeniji što nas uz promatranje prirode samog problema može potaknuti da njega odaberemo kao "najprirodniji" (jer se i naše mišljenje najvjerojatnije "glatko" prilagođava). Njegovim odabirom smatramo da je on onaj koji je najbolje izbalansirao izraze pristranosti i varijance unutar izraza za grešku 1.16. Dakle, on je onaj koji je dovoljno dobro pratio podatke, no ne toliko da bi neprirodno iskrivio procjenitelja.

Stoga, odabrat ćemo model dobiven metodom lokalne linearne regresije kao konačan rezultat našeg modeliranja, odnosno odlučit ćemo njega uzeti kao procjenitelja (reprezentaciju) regresijske funkcije (veze primanja i ugleda koji smo pokušali opisati).

3.3 Kod uz završni primjer

```
library(car)
data(Prestige)

#Tockasti grafikon
plot(Prestige$income, Prestige$prestige,
     xlab = "Income", ylab = "Prestige")

#Pomocne funkcije
loess.gcv <- function(x, y){
  nobs <- length(y)
  xs <- sort(x, index.return = TRUE)
  x <- xs$x
  y <- y[xs$ix]
  tune.loess <- function(s){
    lo <- loess(y ~ x, span = s)
    mean((lo$fitted - y)^2) / (1 - lo$trace.hat/nobs)^2
  }
  os <- optimize(tune.loess, interval = c(.01, .99))$minimum
  lo <- loess(y ~ x, span = os)
  list(x = x, y = lo$fitted, df = lo$trace.hat, span = os)
}

ksmooth.gcv <- function(x, y){
  nobs <- length(y)
  xs <- sort(x, index.return = TRUE)
  x <- xs$x
  y <- y[xs$ix]
  xdif <- outer(x, x, FUN = "-")
  tune.ksmooth <- function(h){
    xden <- dnorm(xdif / h)
    xden <- xden / rowSums(xden)
    df <- sum(diag(xden))
    fit <- xden %*% y
    mean((fit - y)^2) / (1 - df/nobs)^2
  }
  xrng <- diff(range(x))
```

```
37   oh <- optimize(tune.ksmooth, interval = c(xrng/nobs, xrng)
        )$minimum
38   if(any(oh == c(xrng/nobs, xrng)))
39     warning("Minimum found on boundary of search range.\nYou
          should retune model with expanded range.")
40   xden <- dnorm(xdif / oh)
41   xden <- xden / rowSums(xden)
42   df <- sum(diag(xden))
43   fit <- xden %*% y
44   list(x = x, y = fit, df = df, h = oh)
45 }
46
47 #Metoda lokalnih prosjeka (CV odabir parametra)
48 locavg <- with(Prestige, supsmu(income, prestige))
49
50 #Metoda lokalne regresije (GCV odabir parametra)
51 locreg <- with(Prestige, loess.gcv(income, prestige))
52 locreg$df
53
54 #Metoda procjene jezgrama (GCV odabir parametra)
55 kern <- with(Prestige, ksmooth.gcv(income, prestige))
56 kern$df
57
58 #Generiranje slika
59 plot(Prestige$income, Prestige$prestige, xlab = "Income",
      ylab = "Prestige")
60 lines(locavg, lwd = 2)
61 lines(locreg, lwd = 2, lty = 2, col = "red")
62 lines(kern, lwd = 2, lty = 3, col = "blue")
63 legend("bottomright", c("supsmu", "loess", "ksmooth"),
64        lty = 1:3, lwd = 2, col = c("black", "red", "blue"),
           bty = "n")
65
66 plot(Prestige$income, Prestige$prestige, xlab = "Income",
      ylab = "Prestige")
67 lines(locavg, lwd = 2)
68 legend("bottomright", "supsmu", lwd = 2, col = "black", bty
      = "n")
69
```

```
70 plot(Prestige$income, Prestige$prestige, xlab = "Income",
     ylab = "Prestige")
71 lines(locreg, lwd = 2, lty = 2, col = "red")
72 legend("bottomright", "loess", lwd = 2, col = "red", bty = "
     n")
73
74 plot(Prestige$income, Prestige$prestige, xlab = "Income",
     ylab = "Prestige")
75 lines(kern, lwd = 2, lty = 3, col = "blue")
76 legend("bottomright", "ksmooth", lwd = 2, col = "blue", bty
     = "n")
```

Listing 3.1: Kod uz primjer mjesečnih primanja i ugleda [2]

Bibliografija

[1] J. Fox i S. Weisberg, *An R Companion to Applied Regression*, Thousand Oaks, CA, 2019, https://socialsciences.mcmaster.ca/jfox/Books/Companion/.

[2] N. E. Helwig, *Nonparametric Regression (Smoothers) in R*, http://users.stat.umn.edu/~helwig/notes/smooth-notes.html, [Online; pristupljeno 15.9.2023.].

[3] M. Huzak, *Matematička Statistika*, Zagreb, 2020.

[4] E. Peck, G. Vining i D. Montgomery, *Introduction to Linear Regression Analysis*, Wiley series in probability and statistics, 2012.

[5] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2022, https://www.R-project.org/.

[6] N. Sandrić i Z. Vondraček, *Vjerojatnost*, Zagreb, 2019.

[7] L. Wasserman, *All of Nonparametric Statistics*, Springer New York, 2006.

Sažetak

U ovom diplomskom radu predstavili smo metode neparametarske regresije kao alternativu poznatijim metodama parametarske regresije. Za razliku od metoda parametarske regresije, koje nastoje procijeniti procjenitelja nekog određenog oblika s konačnim brojem parametara, neparametarske metode nastoje prvo procijeniti oblik funkcije (ili broj parametara) i onda prilagoditi procjenitelja određenom obliku. Kako bismo procijenili oblik funkcije, prezentirali smo pomoćne metode *leave-one-out* i *generalizirane unakrsne validacije*. Nakon toga, predstavili smo *metodu lokalnih prosjeka, metodu procjene jezgrom* i *metodu lokalne linearne regresije* kao glavne predstavnike klase linearnih procjenitelja. Predstavljene metode ilustrirali smo pojedinačno i zajedno na nekoliko primjera u programskom jeziku R kako bismo čitatelju uz teoretsku osnovu, pružili i praktično znanje iz obrađenog područja.

Summary

In this master's thesis, we presented nonparametric regression methods as an alternative to more well-known parametric regression methods. Unlike parametric regression methods, which aim to estimate a predictor of a specific form with a finite number of parameters, nonparametric methods first seek to estimate the shape of the function (or the number of parameters) and then adapt the estimator to a specific form.

To estimate the shape of the function, we introduced auxiliary methods such as *leave-one-out* and *generalized cross-validation*. Subsequently, we introduced the *local averaging* method, *kernel estimation* method, and *local linear regression* method as key representatives of the class of linear smoothers. We illustrated these methods individually and collectively on several examples in the R programming language to provide the reader with both theoretical foundations and practical knowledge in the covered area.

Životopis

Rođen dana 28.6.1999. u Sisku, od oca Gorana i majke Danijele, rođ. Medjed. Osnovno obrazovanje sam stekao u Osnovnoj školi Viktorovac Sisak, dok sam srednjoškolsko obrazovanje završio u Gimnaziji Sisak. Nakon završene srednje škole, 2018. godine upisujem preddiplomski sveučilišni studij matematike na Prirodoslovno-matematičkom fakultetu u Zagrebu, čijim završavanjem 2021. godine stječem zvanje sveučilišnog prvostupnika matematike. Svoje obrazovanje tijekom studija upotpunjem iskustvom rada u odjelu modeliranja kreditnog rizika u Zagrebačkoj banci te akademijom za računalnu analizu podataka u organizaciji Sofascora. Odmah pri završetku preddiplomskog studija, upisujem diplomski sveučilišni studij financijske i poslovne matematike kako bi stekao zvanje magistra matematike.