R nyelv – regressziós modellek

Vadon Viktória

2024/2025/I. félév

Tartalomjegyzék

1. Lineáris regresszió

1.1. Regressziós feladat

- regresszió alapfeladata: adottak $(\underline{x}_i, y_i)$ párok, valamilyen $(\underline{X}, Y)$ együttes eloszlásból
- cél: Y becslése, előrejelzése $\underline{X}$ függvényében
- azaz, szeretnénk y_i-t közelíteni $y_i \approx f(\underline{x}_i)$ + véletlen hiba formájában
- $\underline{x}$ több különböző változót is magában foglalhat, amik befolyásolják y-t, pl. ha egy autó fogyasztását szeretnénk megbecsülni, benne lehet a hengertérfogat, autó tömege, gyártás éve, stb.

1.2. Lineáris regresszió elméletben: legkisebb négyzetek módszere

- legegyszerűbb, speciális eset **lineáris regresszió**: $y_i \approx a + b \cdot x_i + h_i$
- feltétel: (X, Y) folytonos változók
- h_i: véletlen hiba. feltesszük, hogy független, azonos eloszlásúak, normális/Gauss eloszlásból, 0 várható értékkel (!)
- átrendezve: $h_i = y_i - (a + b \cdot x_i)$
- szokás a becsült/előrejelzett értéket $\hat{y}_i = a + b \cdot x_i$-nek jelölni.
- tartsuk észben: most (x_i, y_i) adottak, a, b-t kell meghatároznunk!
- hogyan? optimalizálási feladat a, b paraméterekre
- **legkisebb négyzetek módszere**: hibák négyzetösszegét minimalizáljuk: $\sum_{i=1}^{n} h_i^2 = \sum_{i=1}^{n} \big(y_i - a - bx_i\big)^2 \to \min_{a,b}$
- megoldás: R kiszámolja és kiköpi, nem kell tudnunk.
  - képlet:
  - $b = \mathrm{Cov}(X, Y)/\mathrm{Var}(X)$
  - $a = \overline{Y} - b \cdot \overline{X}$
  - jelmagyarázat:
  - $\overline{X} = \big(\sum_{i=1}^{n} x_i\big)/n$, $\overline{Y} = \big(\sum_{i=1}^{n} y_i\big)/n$ mintaátlagok
  - $\overline{X^2} = \big(\sum_{i=1}^{n} x_i^2\big)/n$
  - $\overline{XY} = \big(\sum_{i=1}^{n} x_i y_i\big)/n$
  - empirikus szórásnégyzet $\mathrm{Var}(X) = \overline{X^2} - \big(\overline{X}\big)^2$
  - empirikus kovariancia $\mathrm{Cov}(X, Y) = \overline{XY} - \overline{X} \cdot \overline{Y}$
  - levezetés: numerikus analízis anyaga (parciális deriválás, és/vagy túlhatározott LER Gauss-féle normálegyenlete)
- kapunk egy $a + bx$ úgynevezett **regressziós egyenest** – az összes létező egyenes közül ez minimalizálja a pontoktól való négyzetes távolságokat.
- ha tényleg van egy lineáris trend az adatokban, akkor az egyenes követi a pontfelhőt.

1.3. Lineáris regresszió R-ben

- legyenek x_i egy x vector-ban adottak, y_i pedig egy y vector-ban (lehetnek pl data.frame oszlopai)
- hívás regressziós modellre: `reg <- lm(y ~ x)`
- lm(): linear model
  - hasában formula megadása, **speciális szintaxis**!
  - lineárisra elég egyszerű `y ~ x`, jelentése: y-t becsüljük x (lineáris) függvényeként
  - alapértelmezés: a konstans tagot implicit magában foglalja!
  - konstans tag megadása explicit: `lm(y ~ 1 + x)`

  - konstans tag kizárása: `lm(y ~ 0 + x)`
- lm() eredménye egy „lm” osztályú objektum
  - modell objektum hívásával: formuláját és a becsült együtthatókat írja ki (intercept: konstans tag, többi együtthatót a függvénnyel/változóval címkézi)
  - adattagjait lekérhetjük $ szintaxissal vagy különböző függvényhívásokkal.
  - pl. reg$coef, coef(reg): coefficients, együtthatók: (a, b) vector – első tag a konstans, második a meredekség! (ebben a formában mehet rögtön abline() hasába pl.)
  - ref$fitted, fitted(reg): $\hat{y}_i = a + bx_i$ becslések vector-ban
  - reg$resid, resid(reg): h_i hibák vector-ban
- előrejelzés: predict(reg,uj.adatok=data.frame), a modell alapján megjósolja / előrejelzi / megbecsli y_i-t a data.frame-ből származó x_i-ekre
  - statisztikai gyakorlat: általában az adatok kb. 80%-át tanuló adatoknak veszik, az alapján építik a modellt, a maradék kb. 20%-on pedig ellenőrzik az illeszkedést.
- pontfelhő + regressziós egyenes ábrázolása: plot(x,y); abline(coef(reg))

1.4. Modell „jóságának” ellenőrzése

- legegyszerűbb: pontfelhő + regressziós egyenes. tényleg látjuk-e a lineáris trendet? „egyenletesen” szóródik a pontfelhő a regressziós egyenes körül?
- summary(reg): több különböző adat
  - modell formula emlékeztető
  - hibák eloszlása: néhány kvantilis
  - együtthatók táblázatban: becslés (estimate), + ennek a (becsült) szórása (std.error)
  - szintén együtthatók táblázatban: t value, Pr(>|t|) oszlop = p-value: t-teszt az együtthatóra, lényegesen különbözik-e 0-tól? kis p-value: lényegesen különbözik, nagy p-value: valószínűleg 0.
  - alul R-squared, R^2: 0 és 1 közti szám, egy korreláció jellegű mérőszám X és Y között; a modellnek akkor van értelme, ha R^2 nem túl kicsi.
  - szintén alul: F-stat, p-value: úgynevezett F-teszt az egész modellre, H_1: az illesztett lineáris modell jobb a triviális $y_i \approx \overline{Y}$ becslésnél; H_0: a két modell statisztikailag ekvivalens (azaz a regressziós modellünk haszontalan, nincs létjogosultsága) – szokásos módon kis p-value jelenti H_1-t, és hogy a modellünk „szignifikáns”, azaz van értelme.
- feltételek ellenőrzése:
- hibák standard Gauss-e? qqnorm(reg$resid)
- h_i hibák *elvileg* függetlenek x_i-től és a becsült $\hat{y}_i$-től
  - plot(x,reg$resid), plot(reg$fitted,reg$resid): „struktúrálatlan”, egyenletesen szétszórt pontfelhőt kellene látnunk.
  - pl. ha plot(x,reg$resid) úgy néz ki, mintha egy függvényt követne, az azt je-

lenti, a hibák még X függvénye – akkor a lineáris modell nem elég, valamilyen más $y_i \approx f(x_i)$ függvény formájában kellene Y-t közelíteni!
- plot(reg): különböző beépített diagnosztikai ábrák; nem kell mindet értenünk, de itt is ott a Gauss Q-Q plot, és a plot(reg$fitted,reg$resid).

2. Komplex regressziós modellek

2.1. Modell építése

- lineáris regresszió általánosítása: $y_i \approx af_1(x_i) + bf_2(x_i) + \ldots$ típusú közelítések, tetszőleges f_j függvények lineáris kombinációjaként, vagy szerepelhetnek benne egyéb változók is, pl. $y_i \approx a + b \cdot x_i + c \cdot z_i$ formában is közelíthetünk.
- honnan jönnek a változók? változóknak nyilván az adattábla oszlopai jönnek szóba.
- függvények választása: az adatok grafikus elemzéséből, vagy szakértői véleményből. tipikusan: polinom, exp, log, néha trigonometrikus.
- hogyan választunk a lehetőségek közül? több kapott modellek illeszkedését elemezzük.
- széles körben használható a lineáris regresszió, mert némi átalakítással más típusú függvényt is lineáris kombináció alakra lehet hozni:
  - pl. ha tudjuk, hogy $y \approx ax^b$, polinom ismeretlen kitevővel, akkor $\ln(y) \approx \ln(a) + b \cdot \ln(x)$.
  - vagy ha tudjuk, hogy $y \approx a^x$, exponenciális ismeretlen alappal, akkor $\ln(y) \approx \ln(a) \cdot x + 0$ formában keressük.
  - a becsült paraméterrel utána visszatranszformálhatjuk, hogy az additív konstansokat is meg tudjuk becsülni, stb.
- **modell építési folyamata:**
- minél egyszerűbb modellel kezdünk, pl. lineáris, kevés változóval.
- ha a residuals valamilyen mintát mutat, ez alapján próbálunk újabb változót vagy függvényt bevenni a modellbe.
- ha a t-teszt 0-nak sejt egy változót, megpróbáljuk kivenni belőle.
- a különböző modelleket összehasonlítjuk (lentebb tárgyaljuk, hogyan)
- cél: kompromisszum, egy elég jól illeszkedő modell, de nem túl sok változóval.
  - nyilván szeretnénk egy olyan modellt, ami elég jól előrejelzi Y-t
  - de nem akarunk túlságosan ráfókuszálni a konkrét adathalmaz véletlen sajátosságaira, mert akkor új adatokra az előrejelzéseink torzulnak! „overfitting”

2.2. Formula szintaxis

- az egyszerűség kedvéért most folytonos változókra szorítkozunk.
- **függvények az lm(formula) hívásban**:

- különböző változók: pl. `lm(y ~ x + z)`
- vigyázzunk: a formulában mást jelentenek a műveleti jelek, sokszor I() (identitás) függvénybe kell csomagolni őket!
- pl. másodfokú polinom, $y_i \approx a + bx_i + cx_i^2$, akkor `lm(y ~ 1 + x + I(x^2))`
- más függvények lineáris kombinációjaként pl. $y_i \approx a + b \cdot e^x + c \cdot 1/x$: `lm(y ~ 1 + exp(x) + I(1/x))`
- korábbi modell módosítása egy-egy tag vagy új változó hozzáadásával, vagy kivételével: update()
  - pl. `reg1 <- lm(y ~ x)`
  - vegyük bele x^2-et is: `reg2 <- update(reg1, . ~ . + I(x^2))`
  - itt: reg1 a régi modell, amiből indulunk.
  - `. ~ .`: régi formula bal és jobb oldalát helyettesíti. fenti módosítás: jobb oldalhoz hozzávettük x^2-et.
  - most vegyük ki a lineáris tagot: `reg3 <- update(reg2, . ~ . - x)` – ezzel reg3 formulája `y ~ 1 + I(x^2)`.
  - módosítható a bal oldal is; pl. ha $y_i \approx a^{x_i}$ alakot sejtünk, és $\ln(y_i) \approx \ln(a) \cdot x_i$ alakban szeretnénk egy modellt: `reg4 <- update(reg1,log(.) ~ . - 1)`

2.3. Regressziós görbe kirajzolása

- regressziós egyenes esetén abline(reg$coef) gyors megoldás
- de ha nem egyenest illesztünk, hogy rajzoljuk ki a regressziós görbét?
  - a pontok: plot(x,fitted(reg)) formában
  - de ha töröttvonalat szeretnénk, plot(x,fitted(reg),type="l") csak akkor működik jól, ha x növekvő rendben van! (rajzolás és összekötés az indexek sorrendjében) – nehézség: x-szel együtt át kell rendeznünk a reg$fitted vector-t is!
  - megoldás: order(x) visszaadja azt az indexvektort, ami növekvő rendbe rendezi x-et, azaz x[order(x)] ugyanaz, mint sort(x) – akkor reg$fitted[order(x)], a számolt függvényértékeket is átrendezhetjük vele együtt!
  - ha túl ritkák az alappontok, akkor vegyünk fel alappontokat egy z vektorban pl. seq() segítségével, a függvényértékeket pedig predict(reg,as.data.frame(z)) formában tudjuk kiszámolni (a coersion kell, mert a predict csak list vagy data.frame adatokat fogad el)

2.4. Modellek összehasonlítása

- fentebb láttuk, hogy egy-egy modellre hogyan tudjuk tesztelni, van-e a teljes modellnek létjogosultsága, illetve egyes együtthatói lényegesen különböznek-e 0-tól. (ez alapján szűkíthetjük a modellt, majd teszteljük, a szűkített modell is ugyanolyan jó magyarázó erővel bír-e, avagy ugyanolyan jól közelíti Y-t.)

- két modellt csak akkor van értelme összehasonlítani, ha az egyik részmodellje a másiknak!
- pl. a fenti példákkal:
  `reg1 <- lm(y ~ 1 + x)`,
  `reg2 <- lm(y ~ 1 + x + I(x^2))`,
  `reg3 <- lm(y ~ 1 + I(x^2))`,
  akkor reg1, reg2 pár összehasonlítható, és reg2, reg3 pár is; de reg1, reg3 pár nem (mindkettőben van olyan tag, ami a másikban nincs!)
- összehasonlítás: anova(reg1,reg2) – analysis of variance
  - RSS: residual sum of squares, hibák négyzetösszege az adott modellre – nyilván a kisebb jobb, de mennyire lényeges a különbség, megéri-e a plusz tago(ka)t, változó(ka)t? (tipikusan 1-1 tag/változó különbséggel hasonlítjuk össze)
  - F, Pr(>F) oszlopok: ún. F-teszt (RSS-ek hányadosán, relatív hibán alapul). H_0: a két modell gyakorlatilag ekvivalens, H_1: számottevő a különbség – itt is kis Pr(>F)=p-value esetén választunk H_1-t, ilyenkor megéri a komplexebb modellt használni, mert jelentősen jobb közelítést ad.