Traženje podgrupa i iznimnih modela

Matej Mihelčić

Prirodoslovno-matematički fakultet, Sveučilište u Zagrebu

matmih@math.hr

04. travnja, 2026.

Traženje podgrupa - zadatak

- **Prediktivna indukcija pravila** - glavni cilj je stvoriti skup pravila koji radi klasifikaciju ili predviđanje (opisuje karakteristike ciljne labele deskriptivnim atributima).
- **Deskriptivna indukcija pravila** - glavni cilj je stvoriti pravila koja (individualno) opisuju zanimljive podskupove entiteta (uzorke) iz skupa podataka. Opisi trebaju biti simbolični i jednostavni.
- Traženje podgrupa je zadatak na **razmeđi** prediktivne i deskriptivne indukcije.
- Za zadani skup entiteta, opisanih skupom deskriptivnih atributa, te zadano svojstvo entiteta od interesa (ciljna varijabla), zadatak traženja podgrupa je pronaći podskupove entiteta koji su **statistički najinteresantniji**. Želimo da su podskupovi **najveći mogući** i da imaju distribuciju svojstva od interesa koja je u **najvećoj mogućoj mjeri neobična**.
- Distribuciju svojstva od interesa smatramo **neobičnom** ukoliko se značajno razlikuje od distribucije tog svojstva entiteta u cijelom skupu podataka.

Traženje podgrupa - sličnosti i razlike od učenja pravila

- Zadaci traženja podgrupa i učenja pravila koriste skupove podataka u identičnom formatu. Entiteti su opisani skupom deskriptivnih atributa, te postoji svojstvo od interesa (ciljna varijabla).
- Kod zadatka traženja podgrupa možemo tolerirati veći broj lažno pozitivno predviđenih primjera nego kod zadatka klasificiranja entiteta.
- Kod klasifikacijskih zadataka **penaliziramo lažno negativno** predviđene entitete.
- Kod traženja podgrupa **nagrađujemo stvarno pozitivno** predviđene entitete.
- Kod zadatka učenja pravila za klasifikaciju entiteta koristimo sva pravila iz skupa pravila (cijeli skup je jedan prediktivni model).
- Kod zadatka traženja podgrupa, svaka podgrupa je objekt koji opisuje jedan interesantni podskup entiteta (uzorak).

Traženje podgrupa

- Pravila imaju oblik Klasa $\leftarrow$ Uvjet. Svojstvo od interesa je sadržano u vrijednosti klase `Klasa`.
- Logičko pravilo (konjunkcija parova atribut-vrijednost) sadržano u dijelu `Uvjet` se još zove **antecedenta**, a dio `Klasa` se zove konzekventa pravila.
- `Uvjet` stvaramo koristeći vrijednosti atributa entiteta iz skupa za treniranje.
- Standardne pretpostavke zadatka učenja pravila: a) inducirana pravila moraju biti što točnija, b) inducirana pravila moraju biti što različitija (pokrivati različite dijelove prostora entiteta) se **dodatno relaksiraju** u zadatku traženja podgrupa.
- Jedna relaksacija prvog uvjeta je dodavanje uvjeta pokrivanja i neobičnosti distribucije u heuristiku uz uvjet točnosti. Drugi uvjet se relaksira dopuštanjem traženja podgrupa sa značajnim presjekom.
- Zbog relaksacije drugog uvjeta, otkriveni opisi su potencijalno redundantni s aspekta klasifikacije, međutim otkrivaju važna svojstva podskupova entiteta s deskriptivnog aspekta (razne poglede).

Traženje podgrupa - CN2-SD

- Algoritam CN2-SD je modifikacija algoritma za učenje pravila CN2 namijenjena traženju skupova podgrupa.
- Koristi težinsko pokrivanje primjera, kao i težinsku točnost za evaluaciju kvalitete podgrupe.
- Podržava stvaranje uređenih lista pravila i neuređenih skupova pravila, međutim CN2-SD izvodi predviđanje klase entitetu na identičan način, bez obzira na vrstu uređenja pravila.
- Glavna razlika između uređenih i neuređenih lista kod CN2-SD je način konstrukcije pravila. Kod uređenih lista, stvaramo podgrupe koje iterativno odvajaju entitete koji sadrže neku zadanu vrijednost ciljne klase, od entiteta koji sadrže bilo koju drugu vrijednost. Kod neuređenog skupa, tražimo podgrupe najveće točnosti, a klasu koju predviđa podgrupa zadajemo prema vrijednosti klase koju sadrži najveći broj pokrivenih entiteta.
- CN2-SD uvijek pravila reprezentira kao: `IF UVJET THEN VRIJEDNOST_KLASE` $(p_1, p_2, \ldots, p_k)$, gdje $p_1, \ldots p_k$ označavaju vjerojatnosti pojavljivanja primjera svake klase u podgrupi.

Traženje podgrupa - CN2-SD uređena lista

E - skup entiteta
A - skup atributa

function CN2-SD(E, A, M, γ)
$LP \leftarrow []$
InicijalizirajTezine(E, 1.0)
for $c \in$ VrijednostiCiljneKlase **do**
  Poz$\leftarrow$ dohvatiPozitivne(E, c)
  Neg$\leftarrow$ $E \setminus$ Poz
  repeat
    $Najbolji\_K \leftarrow$ PronadiNajboljePravilo(Poz, Neg, A, M)
    if $Najbolji\_K \neq \emptyset$ **then**
      Neka su E' entiteti pokriveni od $Najbolji\_K$
      AžurirajTežine(E', γ)
      Dodaj pravilo "If $Najbolji\_K$ then C" na kraj LP
    end if
  until $Najbolji\_K = \emptyset$ **or** $E = \emptyset$
end for
return LP

Traženje podgrupa - CN2-SD uređena lista

Procedure PronadiNajboljePravilo(Poz, Neg, A, M)
Neka je $B = \{\{\emptyset\}\}$
Neka je $Najbolji\_k = \emptyset$
while $B \neq \emptyset$ **do**
  Specijaliziraj sve konjunkcije u B:
  Neka je B' skup $\{x \wedge y \mid x \in B, y \in A\}$.
  Briši sve konjunkcije iz B' koje su u B (nisu specijalizirane) ili su prazne (npr. $big = y \wedge big = n$).
  for svaku konjunkciju $K_i \in B'$ **do**
    if K_i je statistički značajan (χ^2 test) i bolji od $Najbolji\_k$ prema **težinskoj točnosti** kada se testira na E **then**
      $Najbolji\_K = K_i$.
    end if
  end for
  repeat
    Izbriši najgoru konjunkciju iz B'.
  until $|B'| \leq M$
  $B \leftarrow B'$.
end while
return $Najbolji\_K$.

Traženje podgrupa - CN2-SD neuređeni skup

E - skup entiteta
A - skup atributa

function CN2-SD(E, A, M, γ)
$LP \leftarrow []$
InicijalizirajTezine(E, 1.0)
repeat
  $Najbolji\_K \leftarrow$ PronadiNajboljePravilo(E, A, M)
  if $Najbolji\_K \neq \emptyset$ **then**
    Neka su E' entiteti pokriveni od $Najbolji\_K$
    AžurirajTežine(E', γ)
    Neka je C najčešća vrijednost klase entiteta iz E'
    Dodaj pravilo "If $Najbolji\_K$ then C"
      na kraj LP
  end if
until $Najbolji\_K = \emptyset$ **or** $E = \emptyset$
return LP

Traženje podgrupa - CN2-SD neuređeni skup

Procedure PronadiNajboljePravilo(E, A, M)
Neka je $B = \{\{\emptyset\}\}$
Neka je $Najbolji\_k = \emptyset$
while $B \neq \emptyset$ **do**
  Specijaliziraj sve konjunkcije u B:
  Neka je B' skup $\{x \wedge y \mid x \in B, y \in A\}$.
  Briši sve konjunkcije iz B' koje su u B (nisu specijalizirane) ili su prazne (npr. $big = y \wedge big = n$).
  for svaku konjunkciju $K_i \in B'$ **do**
    $E_i \leftarrow$ primjeri pokiveni konjunkcijom K_i
    $c_i \leftarrow$ NajcescaKlasa(E_i)
    if K_i je statistički značajan (χ^2 test) i pravilo "IF K_i THEN c_i" je bolje od "IF $Najbolji\_k$ THEN c_s" prema **težinskoj točnosti** kada se testira na E **then**
      $Najbolji\_K = K_i$.
    end if
  end for
  repeat
    Izbriši najgoru konjunkciju iz B'.
  until $|B'| \leq M$
  $B \leftarrow B'$.
end while
return $Najbolji\_K$.

Traženje podgrupa - CN2-SD

- Težine pokrivenih primjera se reduciraju na vrijednost između 0 i 1, što daje do znanja algoritmu da se ne treba previše truditi pokriti taj entitet.
- CN2-SD može koristiti bilo koji od dva pristupa reduciranja težina:
  - **Multiplikativne težine** - za zadani $0 < \gamma < 1$, težina entiteta koji je pokriven od strane C pravila se određuje kao $w(e) = \gamma^C$. $\gamma = 1$ bi rezultirao pronalaženjem istog pravila u svakoj iteraciji, dok bi $\gamma = 0$ rezultirao običnim algoritmom pokrivanja.
  - Kod **aditivnog ažuriranja težina**, težine entiteta koji su pokriveni s C pravila se računaju kao $w(e) = \frac{1}{C+1}$.
- **Težinska točnost**:

  $$TezToc(Klasa \leftarrow Uvjet) = \frac{n'_{uvjet}}{N'} \cdot \left(\frac{n'_{Klasa \leftarrow Uvjet}}{n'_{uvjet}} - \frac{n'_{Klasa}}{N'}\right).$$

- $N' = \sum_{e \in E} w(e)$, $n'_{uvjet} = \sum_{e \in E'} w(e)$, E' skup entiteta pokrivenih uvjetom `Uvjet`. $n'_{Klasa \leftarrow Uvjet} = \sum_{e \in E' \ \wedge \ C = Klasa} w(e)$.

Traženje podgrupa - CN2-SD

- $n'_{Klasa} = \sum_{e \in E \ \wedge \ C = Klasa} w(e).$
- Statistička značajnost pravila se računa χ^2 testom uz vrijednost statistike: $\text{Sig}(r_i) = \text{Sig}(\text{Klasa} \leftarrow \text{Uvjet}) = 2 \cdot \sum_j n_{Klasa_j \leftarrow Uvjet} \cdot \log \frac{n_{Klasa_j \leftarrow Uvjet}}{n_{Klasa_j} \cdot p(Uvjet)}.$
- $p(Uvjet) = \frac{n_{Uvjet}}{N}$.
- Ukoliko je $Sig(r_i) > 9.24$, tada podgrupu smatramo značajnom s razinom značajnosti $\alpha = 0.01$.

Traženje podgrupa - deskriptivne mjere

- **Pokrivenost podgrupe** - mjeri postotak entiteta opisanih podgrupom: $Pok(r) = Pok(Klasa \leftarrow Uvjet) = \frac{n_{Uvjet}}{N}$.
- **Pokrivenost skupa podgrupa** - mjeri prosječnu pokrivenost podgrupa iz skupa: $Pok(\mathcal{R}) = \frac{1}{|\mathcal{R}|} \sum_{i=1}^{|\mathcal{R}|} Pok(r_i)$.
- **Potpora podgrupe** - mjeri postotak pozitivno opisanih entiteta podgrupom: $Pot(r) = \frac{n_{Klasa \leftarrow Uvjet}}{N}$.
- **Potpora skupa podgrupa** - mjeri postotak ukupno opisanih pozitivnih entiteta od strane svih podgrupa u skupu (pokrivenost svakog entiteta se broji samo jednom):

$$Pot(\mathcal{R}) = \frac{1}{N} \sum_{Klasa_j \; Uvjet_i | \; \exists r_k, r_k = Klasa_j \leftarrow Uvjet_i} \bigvee n_{Klasa_j \leftarrow Uvjet_i}$$

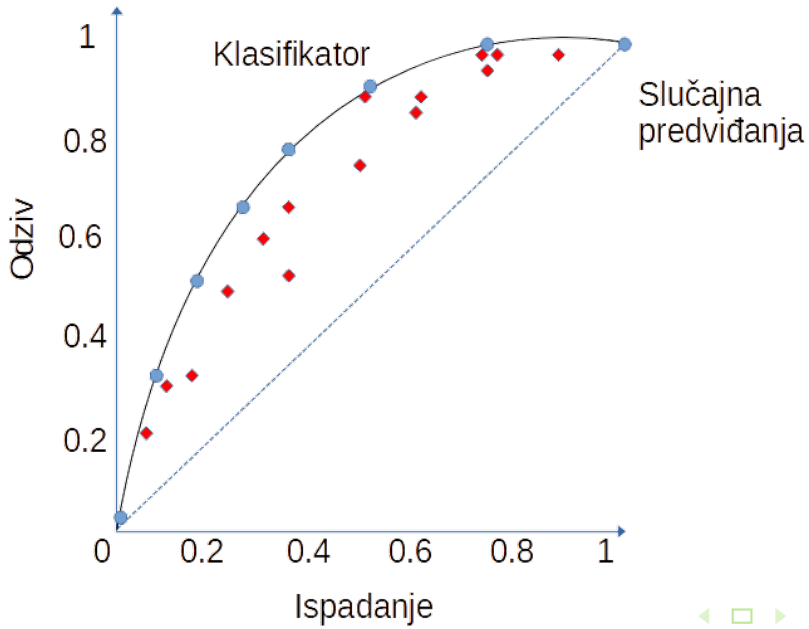
Traženje podgrupa - deskriptivne mjere

- **Veličina skupa podgrupa** - mjeri se kao broj podgrupa u skupu podgrupa: $Velicina(\mathcal{R}) = |\mathcal{R}|$.
- **Značajnost skupa podgrupa** - prosječna značajnost podgrupa sadržanih u skupu podgrupa: $Sig(\mathcal{R}) = \frac{1}{|\mathcal{R}|} \sum_{i=1}^{|\mathcal{R}|} Sig(r_i)$.
- **Neobičnost skupa podgrupa** - prosječna neobičnost podgrupa sadržanih u skupu podgrupa: $TezToc(\mathcal{R}) = \frac{1}{|\mathcal{R}|} \sum_{i=1}^{|\mathcal{R}|} TezToc(r_i)$.

Traženje podgrupa - površina ispod ROC krivulje

Postoje dva glavna načina računanja površine ispod ROC krivulje:

- Podgrupe se individualno crtaju u ROC prostoru prema njihovim odzivima i ispadanju. Iz dobivenih točaka se izračuna konveksna ljuska, koja obuhvaća podgrupe koje imaju najbolje performanse u određenom intervalu vrijednosti odziva i ispadanja. Površina ispod ROC krivulje dobivene od točaka konveksne ljuske se uzima kao procjena AUC skupa podgrupa.
  - Kod ovog načina, ne provjeravamo utjecaj preklapajućih podgrupa, a sve podgrupe izvan linije konveksne ljuske ignoriramo.



Traženje podgrupa - površina ispod ROC krivulje

- Druga metoda koristi sve podgrupe koje pokrivaju entitet da stvori predviđanje. Sigurnosti podgrupe u vrijendost klase se kombiniraju (računa se prosjek), te se predviđa najvjerojatnija vrijednost klase. Ukoliko je izračunata vrijednost veća od praga sigurnosti σ, skup podgrupa predviđa zadanu vrijednost klase.
- AUC računamo kao i kod klasifikatora, variranjem praga sigurnosti σ, te računanjem odziva i ispadanja.
- Za razliku od prvog načina, gdje svaka točka predstavlja točnost jedne podgrupe na ljusci, točke u drugom načinu označavaju performanse skupa podgrupa (promatranog kao klasifikator) uz drugačije zadan prag sigurnosti σ.

Traženje podgrupa - površina ispod ROC krivulje

- Prva metoda računanja AUC eliminira podgrupe koje nisu na konveksnoj ljusci krivulje iz razmatranja (radi probir).
- Prva metoda ne uzima u obzir preklapajuće podgrupe (što je često korisno). Zbog toga se ne može koristiti za usporedbu različitih algoritama. Npr. ova mjera jednako ocjenjuje tri disjunktne podgrupe koje pokrivaju sve entitete i imaju točnost 100% i tri podgrupe s maksimalnim prekapanjem i 100% točnosti koje pokrivaju jednu trećinu entiteta.
- Druga metoda je primjerenija za usporedbu različitih pristupa i primjeni kada podgrupe koristimo za predviđanje.
- Drugu metodu je jednostavnije primjeniti u sklopu postupaka krosvalidacije.

Traženje iznimnih modela

- Proširenje zadatka traženja podgrupa na način da svojstvo od interesa više ne mora nužno biti kategorijska ciljna varijabla.
- Iznimni modeli definiraju ciljni model koji opisuje svojstvo od interesa, te mjeru kvalitete podgrupe - ovisne o izabranom ciljnom modelu.
- Tražimo podgrupe čija svojstva mjerena ciljnim modelom se značajno razlikuju ili od a) tog svojstva **komplementa podgrupe** ili od b) tog svojstva **svih entiteta u skupu podataka**.
- Izbor a) ili b) radimo ovisno o domeni, ciljnom modelu, mjeri točnosti ili učinkovitosti računanja.

Traženje iznimnih modela

Entitet	A_1	A_2	A_3	. . .	A_m	C_1	C_2	. . .	C_k
E_1	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	. . .	$a_{1,m}$	$c_{1,1}$	$c_{2,1}$	. . .	$c_{1,k}$
E_2	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	. . .	$a_{2,m}$	$c_{2,1}$	$c_{2,2}$	. . .	$c_{2,k}$
. . .		. . .						. . .	
E_{n-1}	$a_{n-1,1}$	$a_{n-1,2}$	$a_{n-1,3}$	. . .	$a_{n-1,m}$	$c_{n-1,1}$	$c_{n-1,2}$	. . .	$c_{n-1,k}$
E_n	$a_{n,1}$	$a_{n,2}$	$a_{n,3}$	. . .	$a_{n,m}$	$c_{n,1}$	$c_{n,2}$	. . .	$c_{n,k}$

Traženje iznimnih modela

Za zadani skup podataka D, jezik opisa $\mathcal{D}$, mjeru kvalitete ϕ, prirodni broj k i skup uvjeta $\mathcal{C}$, zadatak pronalaženja top-k iznimnih modela je pronaći listu $d_1, \ldots, d_k$ opisa jezikom $\mathcal{D}$ tako da:

- $d_i,\ \forall 1 \leq i \leq k$ zadovoljava sve uvjete iz $\mathcal{C}$.
- $\forall i, j,\ i < j \Rightarrow \phi(D_i) \geq \phi(D_j)$.
- $\forall d \notin \{d_1, \ldots, d_k\}$, ukoliko d zadovoljava sve uvjete iz $\mathcal{C} \Rightarrow \phi(d) \leq \phi(d_k)$.

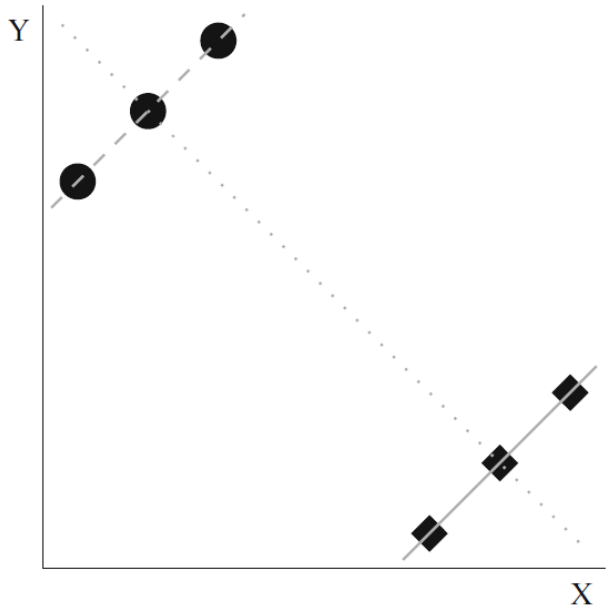
U mjeru kvalitete podgrupe se ugrađuje informacija o entropiji koja uzima u obzir informaciju o podgrupi i njezinom komplementu ili informacija o značajnosti (p-vrijednost) razlike parametara modela klase između podgrupe i komplementa.

- Glavni razlog: izbjegavanje podgrupa koje pokrivaju jako mali broj primjera. Kod jako malih podgrupa neobičnost (odstupanje) može biti visoko zbog slučajnosti u skupu podataka.

Traženje iznimnih modela

Izbor usporedbe podgrupe s komplementom ili sa svim entitetima iz skupa podataka može znatno utjecati na odluku je li otkrivena podgrupa značajna.

- Pretpostavimo da imamo problem kod kojeg su zadane dvije ciljne varijable.
- Želimo pronaći podgrupe (i njihove opise), takve da regresijski pravac pronađene podgrupe znatno odstupa od kontrolne grupe.



Traženje iznimnih modela

- Pretpostavimo da kružići označavaju entitete podgrupe.
- Ukoliko iznimnost podgrupe mjerimo preko koeficijenta smjera pravca, tada:
  - Podgrupa i njen komplement imaju isti koeficijent smjera pravca (1). Podgrupa u tom kontekstu nije iznimna.
  - Podgrupa ima značajno različit koeficijent smjera pravca (1) u odnosu na regresijski pravac svih entiteta iz skupa podataka (-1) i u tom kontekstu je iznimna.

Traženje iznimnih modela - algoritam

- Koristi se pretraživanje odozgo prema dolje uz upotrebu pretraživanja snopom (eng. beam search).
- Pretraživanje se provodi po razinama uzimajući u obzir kriterije složenosti opisa i veličine pronađenih podgrupa.
- Podgrupe se poboljšavaju korištenjem operatora poboljšanja (profinjenja).
- Jezik opisa podržan algoritmom koji opisujemo se sastoji od konjunkcija atributa (kao i kod učenja pravila i traženja podgrupa).
- Koristi se posebna vrsta diskretizacije numeričkih atributa.
  - Za zadani prirodni broj b, na najvišoj razini pretrage, radimo diskretizaciju u b pretinaca jednakih frekvencija.
  - Na nižim razinama pretrage se provodi diskretizacija u b pretinaca jednakih frekvencija, međutim na podskupu entiteta opisanih opisom u izgradnji.
- Pretinci se dinamički mijenjaju i razlikuju između opisa.

Traženje iznimnih modela - algoritam

Operator poboljšanja (profinjenja) radi tako da:

- U slučaju binarnog atributa a_i, iz postojećeg opisa d stvara: $d \wedge a_i$, $d \wedge \neg a_i$.
- U slučaju kategorijskog atributa a_i s kategorijskim vrijednostim $v_1, \ldots, v_k$, iz postojećeg opisa d stvara:
  $d \wedge a_i = v_1, \ldots, d \wedge a_i = v_k$, $d \wedge a_i \neq v_1, \ldots, d \wedge a_i \neq v_k$.
- U slučaju numeričkog atributa a_i. Poredamo vrijednosti entiteta pokrivenih opisom d, tako dobijemo $v_(1), \ldots, v(p)$. Izaberemo točke podjele $s_1, \ldots, s_{b-1}$, tako da $s_j = v_{(\lfloor j \cdot \frac{p}{b} \rfloor)}$. Stvaramo opise:
  $d \wedge (a_i \leq s_j)$, $d \wedge (a_i \geq s_j)$, $j = 1, \ldots, b-1$.

Traženje iznimnih modela - algoritam

Algoritam Pretraživanje snopom top-k kandidata iznimnih modela

Ulaz: Skup podataka D, mjera kvalitete ϕ, operator poboljšanja η, širina zrake w, dubina zrake d, veličina skupa rezultata k, Ograničenja $\mathcal{C}$
Izlaz: PriorityQueue *rezultat*

```
Queue redKandidata ← [{}]                          {Krećemo od praznog opisa}
rezultat ← new PriorityQueue(k)
for level ← 1; level ≤ d; level++ do
    snop ← new PriorityQueue(w)
    while redKandidata ≠ ∅ do
        generator ← redKandidata.dequeue()
        skupPoboljsanja ← η(generator)
        for all opis ∈ skupPoboljsanja do
            kvaliteta ← φ(opis)
            if opis.ZadovoljavaSveUvjete(C) then
                rezultat.insert(opis, kvaliteta)
                snop.insert(opis, kvaliteta)
            end if
        end for
    end while
    while snop ≠ ∅ do
        redKandidata.enqueue(snop.front())
    end while
end for
return rezultat
```

Traženje iznimnih modela - primjeri

Korelacijski model:

- Pretpostavljamo da imamo dva numerička cilja (c_1 i c_2).
- Zanima nas linearna asocijacija između ta dva cilja.
- Asocijaciju mjerimo koeficijentom korelacije nad podskupom entiteta S: $\hat{r} = \frac{\sum_{i \in ind(S)} (c_{1,i} - \bar{c_1})(c_{2,i} - \bar{c_2})}{\sqrt{\sum_{i \in ind(S)} (c_{1,i} - \bar{c_1})^2 \sum_{i \in ind(S)} (c_{2,i} - \bar{c_2})^2}}$
- Označimo s $\hat{r}_G$ i $\hat{r}_{G^C}$ procjenu koeficijenta korelacije izračunatog iz uzorka podgrupe G i njezinog komplementa G^C, a s ρ_G i ρ_{G^C} populacijske koeficijente korelacije.
- Možemo izračunati statistički test s hipotezom $H_0 : \rho_G = \rho_{G^C}$ i alternativom $H_1 : \rho_G \neq \rho_{G^C}$.
- Uz pretpostavku da imamo podgrupe koje pokrivaju više od 25 entiteta, možemo računati Fisher-ov z: $z' = \frac{1}{2} \ln(\frac{1+\hat{r}}{1-\hat{r}})$.

Traženje iznimnih modela - primjeri

Korelacijski model (nastavak):

- Distribucija z' je aproksimativno normalna, stoga statistika $z^* = \frac{z' - z'_C}{\sqrt{\frac{1}{n-3} + \frac{1}{n^C - 3}}}$ aproksimativno slijedi normalnu distribuciju s hipotezom H_0.
- Mjera kvalitete ϕ podgrupe može biti $1 - p$, gdje p označva p-vrijednost testa razlike u korelaciji između te podgrupe i njezinog komplementa.
- Kao alternativnu mjeru kvalitete ϕ možemo koristiti $H(G, G^c) \cdot |\hat{r}_G - \hat{r}_{G^C}|$.
  $H(G, G^c) = -p_{x \in G} log(p_{x \in G}) - p_{x \in G^c} log(p_{x \in G^C})$.

Traženje iznimnih modela - primjeri

Jednostavni regresijski model:

- Promatramo model: $y_i = \beta_0 + \beta_1 \cdot x_i + \varepsilon_i$. ε_i označava regresijski rezidual (razliku između predviđanja modela i prave vrijednosti varijable y).
- Promatramo razlike između jednostavnih regresijskih modela stvorenih nad entitetima sadržanim u grupi G i grupi G^C.
- Možemo promatrati razlike u koeficijentu smjera (β_1) ili slobodnom članu (β_0).
- Promatramo hipoteze $H_0 : \beta_1^G = \beta_1^{G^C}$ i $H_1 : \beta_1^G \neq \beta_1^{G^C}$.
- Koristimo procjenu metodom najmanjih kvadrata koeficijenta smjera β_1 (u oznaci $\hat{\beta}_1$) i nepristrani procjenitelj s^2 za varijancu parametra $\hat{\beta}_1$. $\hat{\beta}_1 = \frac{\sum_{i \in ind(S)} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i \in ind(S)} (x_i - \bar{x})}$, $s^2 = \frac{\sum_{i \in ind(S)} \hat{\varepsilon}_i^2}{(|S| - 2) \sum_{i \in S} (x_i - \bar{x})^2}$.

Traženje iznimnih modela - primjeri

Jednostavni regresijski model (nastavak):

- Test statistiku definiramo kao: $t' = \dfrac{\hat{\beta}_1^G - \hat{\beta}_1^{G^C}}{\sqrt{s_G^2 + s_{G^C}^2}}$ uz

  $k = \dfrac{(s_G^2 + s_{G^C}^2)^2}{\frac{s_G^4}{|S|-2} + \frac{s_{G^C}^4}{|E|-|S|-2}}$ stupnjeva slobode.
- Za $|E| > 40$ značajnost test statistike možemo računati iz t distribucije.
- Mjera kvalitete podgrupe ϕ je $1 - p$, gdje je p, p-vrijednost dobivena testiranjem gornje hipoteze definiranom statistikom.

Traženje iznimnih modela - primjeri

Klasifikacija:

- Pretpostavljamo $I_1, \ldots, I_{m-1}$ mogu biti binarne, kategorijske ili numeričke i $y = I_m$ je diskretna.
- Promotrimo model logističke regresije:
  $logit(P(y_i = 1|x_i)) = \ln(\frac{P(y_i = 1|x_i)}{P(y_i = 0|_i)}) = \beta_0 + x_i \cdot \beta_1$, gdje $y \in \{0, 1\}$, a $x \in \{I_1, \ldots, I_{m-1}\}$.
- Koeficijent β_1 nam govori o utjecaju x na pojavljivanje događaja y.
- Želimo odrediti ima li x_i znatno drugačiji utjecaj na model unutar neke podgrupe G sa skupom pokrivenih entiteta S, stoga definiramo:
  $$\text{logit}(P(y_i = 1|x_i)) = \begin{cases} (\beta_0 + \beta_1) + (\beta_2 + \beta_3) \cdot x_i & \text{za } i \in ind(S) \\ \beta_0 + \beta_2 \cdot x_i & \text{if } i \notin ind(S) \end{cases}$$
- Provodimo Wald-ov test za ispitivanje hipoteza, $H_0 : \beta_3 = 0$, $H_1 : \beta_3 \neq 0$.
- Testna statistika $W = \frac{b_3}{\hat{\sigma_{b_3}}}$. b_3 je dobiven kao komponenta argumenta koji maksimizira funkciju vjerodostojnosti modela logističke regresije.

Traženje iznimnih modela - primjeri

Klasifikacija (nastavak):

- Funkcija vjerodostojnosti se definira kao:
$$L(\beta) = \sum_{i=1}^{n} \left[y_i (\mathrm{x}_i^T \beta) - \ln(1 + e^{\mathrm{x}_i^T \beta}) \right]$$
- $\hat{\sigma}_{b_3}$ dobijemo kao korjen recipročne vrijednosti 4-tog elementa dijagonale Heissiana funkcije vjerodostojnosti L modela logističke regresije.
- Značajnost statistike provjeravamo iz normalne distribucije ($\mathcal{N}(0, 1)$).
- Kao mjeru kvalitete podgrupe ϕ koristimo $1 - p$, gdje je p, p-vrijednost dobivana iz Wald-ovog testa za procjenu vrijednosti parametra β_3.

Traženje podgrupa i iznimnih modela - važniji alati

Softver za kreiranje i evaluaciju podgrupa:

- Python (Orange paket) - `CN2-SD`.
- Python (Subgroups paket) - kolekcija algoritama (`VLSD`, `QFinder`, `DSLM`, itd.).

Softver za kreiranje i evaluaciju iznimnih modela:

- Python (EMM paket, `https://github.com/MathynS/emm`) - EMM algoritam i primjeri.