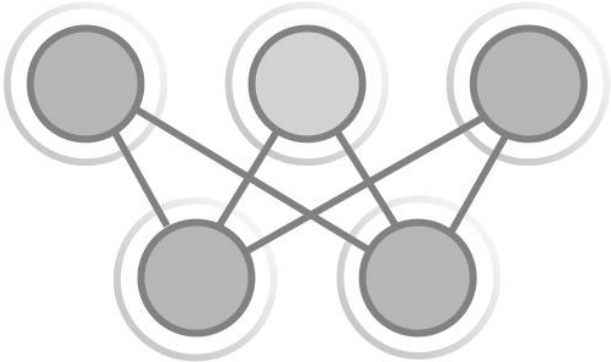
Strojno učenje

Fakultet elektrotehnike i računarstva
Zavod za elektroniku, mikroelektroniku, računalne i inteligentne sustave

Bojana Dalbelo Bašić, Jan Šnajder

Vrednovanje klasifikatora



Osnovne evaluacijske mjere – točnost

- Ukupan broj primjera = TP + TN + FP + FN = 150
- Broj točno klasificiranih primjera = TP + TN = 136
- Broj pozitivnih primjera = TP + FN = 8
- Broj negativnih primjera = TN + FP = 142

- Točnost *(*engl. *accuracy)* je udio točno klasificiranih primjera u skupu svih primjera.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$

  - U primjeru: Acc = 90,7%

Osnovne evaluacijske mjere – preciznost

- Preciznost *(engl. precision)* je udio točno klasificiranih primjera u skupu pozitivno klasificiranih primjera.

$$P = \frac{TP}{TP + FP}$$

  - U primjeru: P = 33,3%
  - Još se koristi naziv *positive predictive value (PPV).*
  - Vidimo da ovaj klasifikator nije precizan.

Osnovne evaluacijske mjere – odziv

- Odziv *(engl. recall)* je udio točno klasificiranih primjera u skupu svih pozitivnih primjera.

$$R = \frac{TP}{TP + FN}$$

  - U primjeru R = 75,0%
  - Drugi nazivi: osjetljivost *(engl. sensitivity*), *hit rate, true positive rate (TPR).*
  - Želimo da odziv bude što veći (npr. želimo otkriti što veći broj ljudi koji imaju neku bolest).

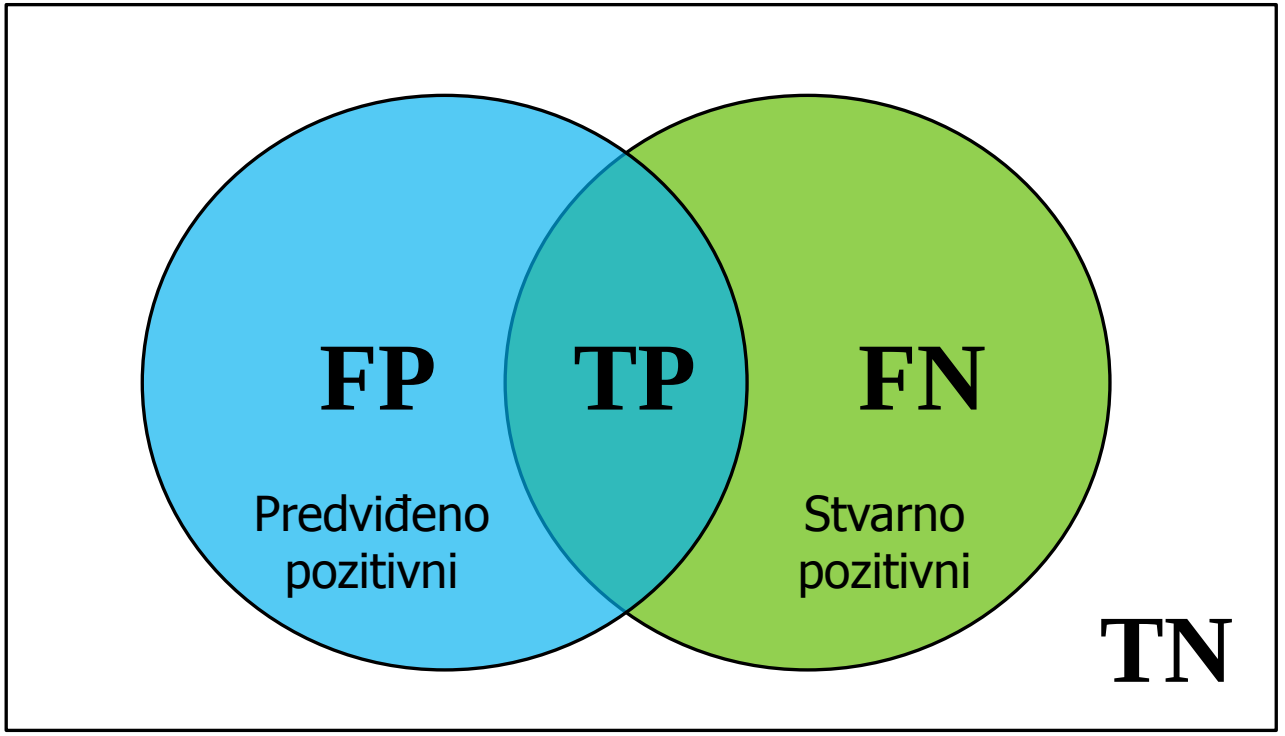
Osnovne evaluacijske mjere – specifičnost

- Specifičnost *(engl. specificity)* je udio točno klasificiranih primjera u skupu svih negativnih primjera.

$$\text{Specifičnost} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

  - U primjeru specifičnost = 91,6%
  - Još se koristi naziv *true negative rate (TNR).*
  - Ako je specifičnost 100% onda su svi zdravi ljudi prepoznati kao zdravi. Manja specifičnost znači da dio zdravih ljudi dobiva krivu dijagnozu.

Preciznost i odziv



$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN}$$

Osnovne evaluacijske mjere – zaključak

- Različite vrste mjera koristit će se u različitim područjima (domenama), ponajviše ovisno o uobičajenoj raspodijeli pozitivnih i negativnih primjera.
- Niti jedna od do sada navedenih mjera nije dovoljna sama za sebe
- Ako je skup izrazito neuravnotežen (puno pozitivnih primjera, a malo negativnih, ili obrnuto), lako je napraviti trivijalan klasifikator s visokom točnošću (klasifikator koji vraća apriorno najvjerojatniju klasu)
- Npr. ako je u 1000 primjera njih 990 negativno, a klasifikator sve primjere klasificira negativno, onda:
  - TP = 0, TN = 990, FP = 0, FN = 10
  - Točnost je 99%, ali odziv je 0% (preciznost je nedefinirana)

F-mjera

- F-mjera je harmonijska sredina preciznosti i odziva:

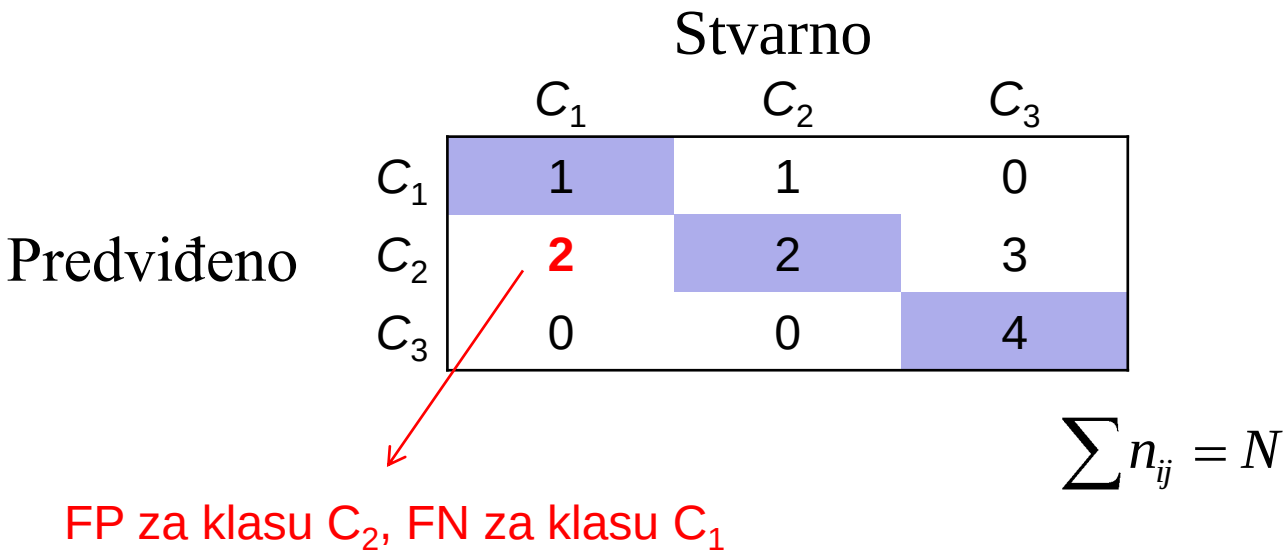
$$F = \frac{2}{(1/P)+(1/R)} = \frac{2PR}{P+R}$$

- U danom primjeru: F=46,1%
- U općem slučaju važnost preciznosti i odziva kontroliramo parametrom β (ako nam je važniji odziv koristit ćemo veću vrijednost parametra β).

$$F_\beta = \frac{(1+\beta^2)PR}{\beta^2 P + R}$$

- Tipično: mjera $\boldsymbol{F_1}$, rjeđe $F_{0,5}$ (naglašena preciznost) ili F_2 (naglašen odziv)

Višeklasna klasifikacija (K>2)



Za klasu C_j:

- TP_j = j-ti element dijagonale
- FP_j = zbroj nedijagonalnih elemenata j-tog retka
- FN_j = zbroj nedijagonalnih elemenata j-tog stupca
- $TN_j = N - TP_j - FP_j - FN_j$ (zbroj po elementima izvan retka j i stupca j)

Makro-F_1

- Preciznost i odziv klase C_j

$$P_j = \frac{TP_j}{TP_j + FP_j} \qquad R_j = \frac{TP_j}{TP_j + FN_j}$$

- F_1-mjera klase C_j

$$F_j = \frac{2P_jR_j}{P_j + R_j}$$

- Makro-uprosječena (engl. *macro-averaged*) F_1-mjera (kraće: **makro-F_1**)

$$F^{macro} = \frac{1}{K}\sum_{j=1}^{K} F_j$$

Mikro-F_1

- Zbrojimo TP, FP i FN po svim klasama:

$$TP = \sum_{j=1}^{K} TP_j \qquad FP = \sum_{j=1}^{K} FP_j \qquad FN = \sum_{j=1}^{K} FN_j$$

- Preciznost i odziv:

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN}$$

- Mikro-uprosječena (engl. *micro-averaged*) F_1-mjera (kraće: **mikro-F_1**)

$$F_1^{micro} = \frac{2PR}{P + R} = \frac{2TP}{2TP + FP + FN}$$

Mikro/makro-F_1 (primjer)

Predviđeno \ Stvarno	C_1	C_2	C_3
C_1	1	1	0
C_2	2	2	3
C_3	0	0	4

N=13

- TP_1=1, FP_1=1, FN_1=2 $\Rightarrow$ P_1=0.5, R_1=0.33 $\Rightarrow$ $F_{1,1}$=0.4
- TP_2=2, FP_2=5, FN_2=1 $\Rightarrow$ P_2=0.29, R_2=0.66 $\Rightarrow$ $F_{1,2}$=0.4
- TP_3=4, FP_3=0, FN_3=3 $\Rightarrow$ P_3=1, R_3=0.57 $\Rightarrow$ $F_{1,3}$=0.73

$\Downarrow$ $\Downarrow$ $\Downarrow$

TP=7 FP=6 FN=6

$\Downarrow$

P=R=0.54

$$F_1^{micro} = \frac{2 \cdot 0.54 \cdot 0.54}{0.54 + 0.54} = 0.54$$

$\Downarrow$

$$F_1^{macro} = \frac{1}{3}(0.4 + 0.4 + 0.73) = 0.51$$

Mikro-F_1 vs. makro-F_1

- Makro-F_1 sve klase tretira jednako
  - Zbog toga primjeri iz malih klasa imaju veći utjecaj na mjeru nego što bi imali kod mjere mikro-F_1
  - Razlika je vidljiva kod neuravnoteženih skupova
  - Npr.

	C_1	C_2	C_3
C_1	100	10	1
C_2	2	2	3
C_3	8	5	400

$$F_1^{micro} = 0.95$$

$$F_1^{macro} = 0.69$$

- Makro-F_1 je tipično manji od mikro-F_1
  - zato jer je na slabo zastupljenim klasama klasifikacija tipično lošija!
- U praksi se češće koristi mikro-F_1
  - ako ne piše što je, onda je mikro

Napomena

- Kod višeklasne klasifikacije (K>2) uvijek vrijedi *FP* = *FN*
  - *TP* = trag matrice
  - *FP* = *FN* = zbroj nedijagonalnih elemenata matrice
  - *TN* = *K*N* - *TP* - *FP* – *FN*

	C_1	C_2	C_3
C_1	1	1	0
C_2	2	2	3
C_3	0	0	4

N=13

  - *TP* = 7, *FP* = *FN* = 6, *TN* = 3*13-19 = 20
- Posljedično, vrijedi ***P =R = micro*** F_1
  - *P* = *R* = micro F_1 = 0.54

Napomene

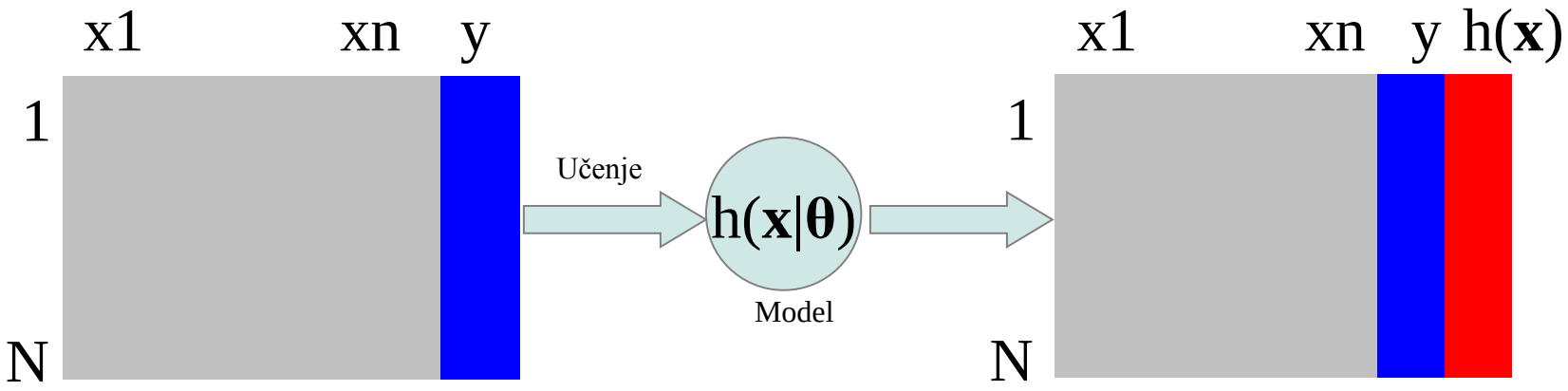
- Za svaku mjeru (točnost, preciznost, odziv, F1) može se izračunati mikro-prosjek ili makro-prosjek
  - Mikro-prosjek: računamo izravno na tablici zabune *K*x*K*
  - Makro-prosjek: računamo *K* vrijednosti mjera na tablicama zabune 2x2, i zatim uprosječimo
- Za *K*=2 klase također možemo raditi mikro-prosjek ili makro-prosjek

Procjena pogreške (eng. *error estimation*)

- Mjere vrednovanja izračunavaju se na slučajnom uzorku
- Dakle one su slučajne varijable, a vrijednost koju dobivamo je procjena
- Postavlja se pitanje kako dobiti dobru procjenu
- Dobra procjena je **“poštena”**, a to najčešće znači **pesimistična** (ako je procjena pesimistična, znamo da će u stvarnosti klasifikator raditi tako ili još bolje)
- Razvijen je niz postupaka za **procjenu pogreške** klasifikatora (eng. *error estimation*)
- Unatoč tome, mnogi još uvijek rade pogreške kod procjene pogreške :-)

Nepoštena procjena pogreške

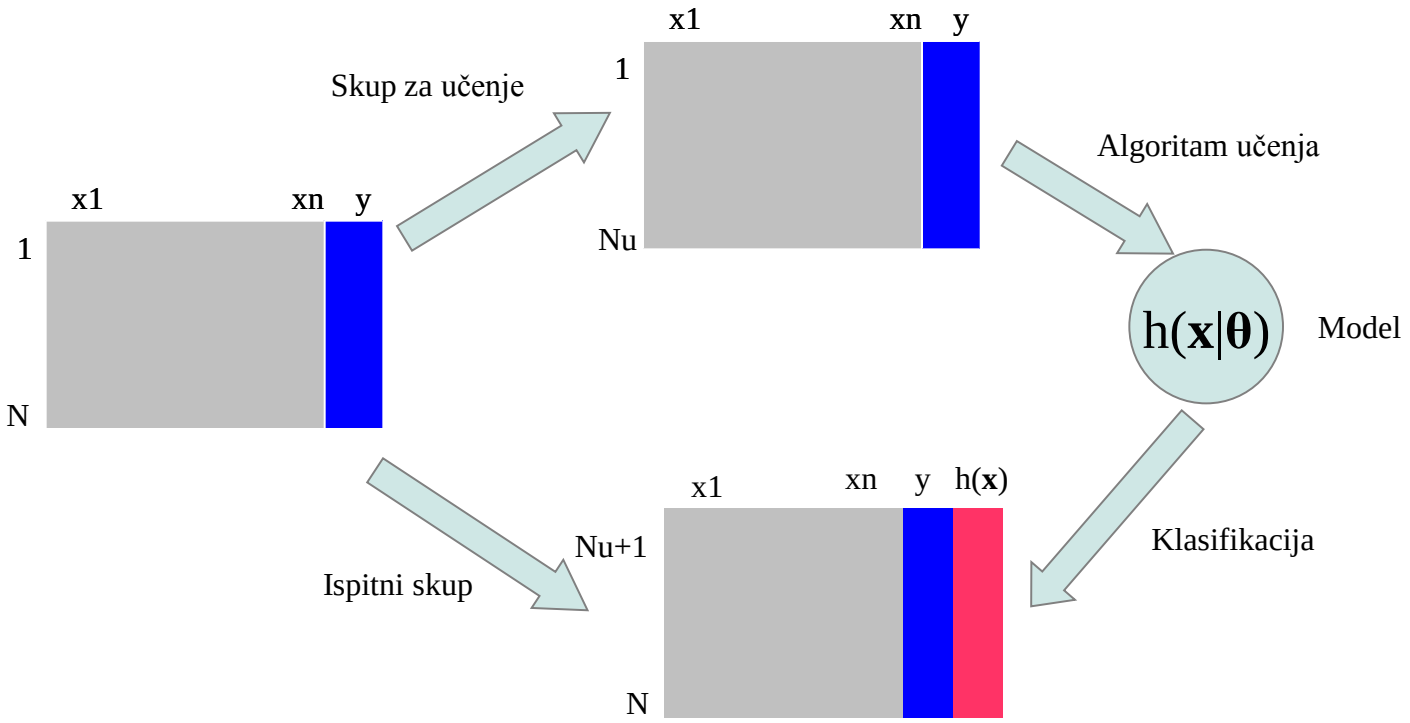
- Procjena pogreške (točnosti, preciznosti, odziva, mjere F1,...) na **istom skupu primjera na kojem je klasifikator učen**



- Ovo je loše jer ne mjerimo pogrešku generalizacije nego empirijsku pogrešku, koja je uvijek manja (i opada sa složenošću modela)

Metoda izdvajanja (engl. *holdout method*)

- Razdvajanje primjera na **skup za učenje** (engl. *training set*) i **ispitni skup** (engl. *test set*)
- Najjednostavnija varijanta unakrsne provjere

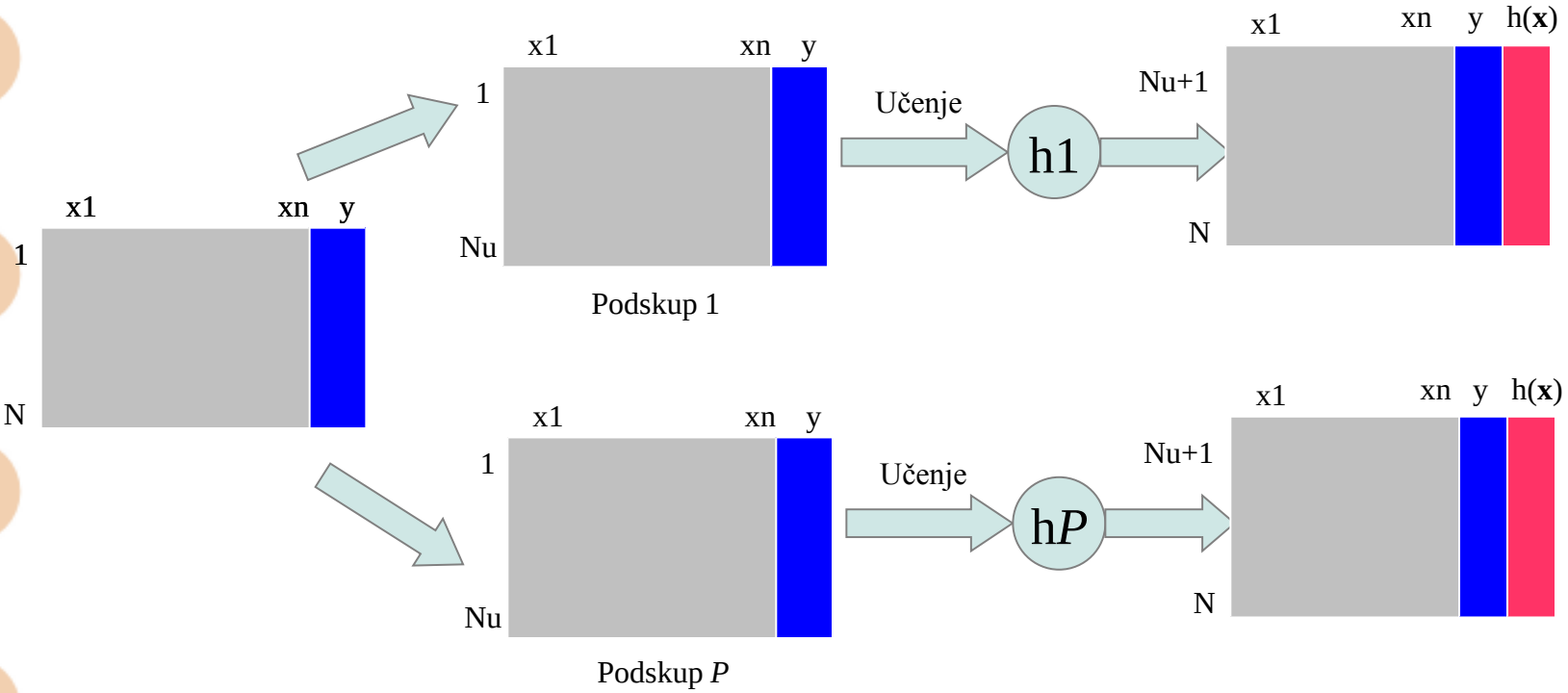


Metoda izdvajanja (engl. *holdout method*)

- Prednost: procjenjujemo pogrešku generalizacije
- Nedostatci:
  - **Gubitak primjera za učenje:** Budući da smo dio primjera morali ostaviti postrani zbog ispitivanja, imamo manje primjera za učenje i gubimo vrijednu informaciju. To je osobito problematično ako je primjera ukupno malo.
  - **Loša točnost procjene pogreške:** Naša procjena pogreške temelji se na samo jednom uzorku. Točnost procjene će doduše rasti što je ispitni skup veći, ali taj je skup uvijek ograničen.
- Oba nedostatka moguće je riješiti postupcima temeljenima na **ponovnom uzorkovanju** (engl. resampling)
  - Algoritam učimo na većini raspoloživih primjera
  - Dobivamo točnije procjene pogreške

Ponovljeno izdvajanje (engl. *repeated holdout*)

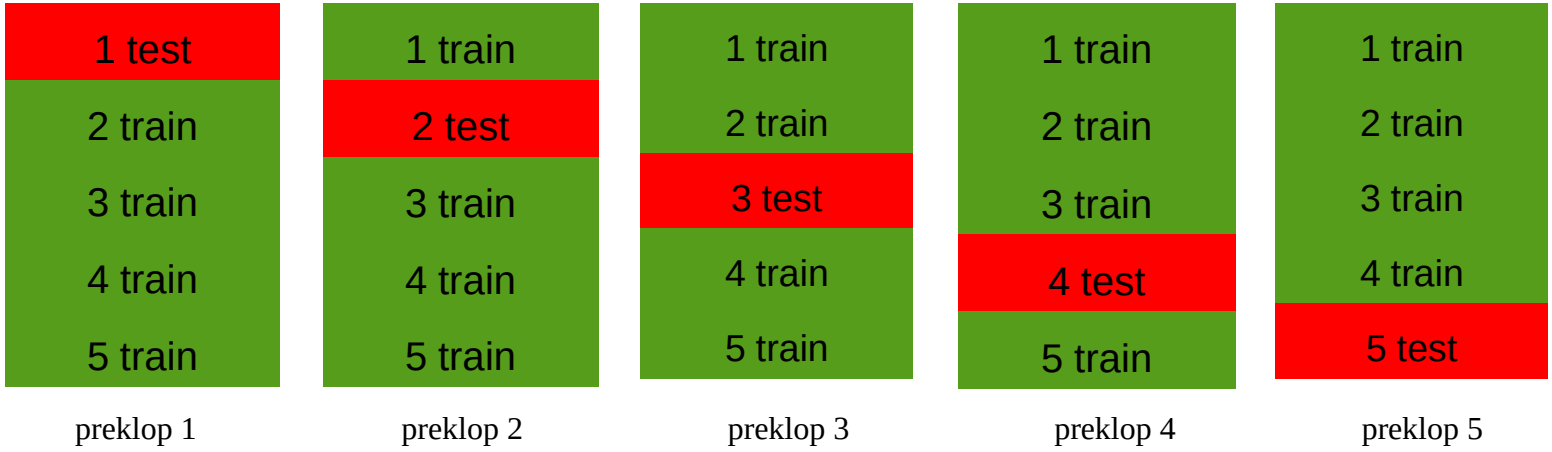
- Slučajan odabir *P* podskupova zadane veličine i podjela svakog takvog podskupa na skup za učenje i ispitni skup



- Procjena pogreške = prosjek pogrešaka pojedinih modela
- Problem: skupovi se preklapaju i **nemamo kontrolu** koji su primjeri i koliko puta upotrijebljeni

k-struka unakrsna provjera (engl. *k-folded cross validation*)

- Podjela skupa primjera u *k* particija odnosno preklopa (engl. *k-folded cross validation*)
- Učimo klasifikator na *k*-1 preklopa i ispitujemo ga na k-tom preklopu, pa to ponavljamo ukupno *k*-puta s pomicanjem ispitnog skupa



- Procjena pogreške = prosječna pogreška na *k* preklopa

k-struka unakrsna provjera

1 **za** i **= 1 do *k* radi:**

2 **nauči** model na skupu *D\Di* **(k-1)/k** od ukupnog skupa

3 **izračunaj** pogrešku na skupu *Di* **1/k** od ukupnog skupa

4 izračunaj prosjek pogreške na *k* preklopa

Stratificirana k-struka unakrsna provjera

- Podjela na skup za učenje i skup za ispitivanje može biti takva da ne zrcali pravu razdiobu primjera u skupu za učenje
  - Može rezultirati s pretjerano pesimističnom procjenom
- Rješenje je da se skupovi stratificiraju, odnosno da razdioba klasa bude sačuvana u oba skupa:
  - skup primjera podijeliti u K podskupova, po jedan za svaku klasu
  - svaki takav podskup podijeliti u k preklopa
  - združiti K preklopa (po jedan od svake klase) u jedan preklop

k-struka unakrsna provjera

- Tipično se uzima k=10 ili k=5
- Prednosti:
  - jednostavna izvedba (implementiran u mnogim alatima)
  - računalno ne suviše zahtjevno (ako k nije prevelik)
  - daje procjenu pogreške na nepreklapajućim podskupovima (točnija procjena)
- Nedostaci:
  - pojedinačnih k klasifikatora nije nezavisno jer međusobno dijele k-2 preklopa tj. (k-2)/k skupa za učenje
  - visoka varijanca procjene pogreške

Unakrsna provjera "izdvoji jednog"

- Engl. *Leave-one-out cross validation* (LOOCV)
- *k*-struka unakrsna provjera uz *k=N*
- U svakom preklopu **klasifikator se ispituje na samo jednom primjeru**
- Prednosti:
  - iskorištavamo gotovo potpun skup primjera
  - točnija procjena pogreške
- Nedostatci:
  - računalno prezahtjevno za veliki N
  - visoka varijanca procjene pogreške, osobito ako postoje vrijednosti koje odskaču
- Primjenjivo kada je skup primjera za učenje malen ili srednje velik

Procjena pogreške kod odabira modela

- Ako treba napraviti odabir modela (optimizaciju hiperparametara), unakrsnu provjeru radimo nad tri **disjunktna** skupa:
  - D1: **skup za učenje** (engl. training set)
  - D2: **skup za provjeru** (engl. validation set)
  - D3: **skup za ispitivanje** (engl. test set)
- Model učimo na skupu D1, a pogrešku generalizacije izračunavamo na skupu D2. To ponavljamo sve dok ne pronađemo optimalan model (optimalne hiperparametre) na skupu D2.
- Kada odabremo optimalan model na skupu D2, taj model učimo na skupu $D1 \cup D2$
- Zatim izračunamo pogrešku generalizacije tako naučenog modela na skupu ispitnom skupu D3
- To je pogreška koju objavljujemo

Procjena pogreške kod odabira modela

- Što ako želimo napraviti *k*-struku unakrsnu provjeru zajedno s odabirom modela?
- Moramo raditi nad tri skupa, pa imamo dvije ugniježđene petlje:
  - Vanjska petlja za učenje i testiranje (kao i prije)
  - Unutarnja petlja za odabir modela (za učenje i provjeru)
- Ovo nazivamo **ugniježđena *k*-struka validacija** (engl. *nested k-fold cross validation*)
- Dobivamo **točniju procjenu** pogreške nego s metodom izdvajanja (engl. *holdout*) jer:
  - Odabir modela radimo na temelju prosjeka pogreške
  - Konačna pogreška modela računa se na temelju prosjeka pogreške

k-struka ugniježđena unakrsna provjera

Primjer: provjera 5-10

1 vanjska petlja: **za i = 1 do 5 radi**:
2 za svaku odabranu vrijednost hiperparametra **θ radi:**
3 unutarnja petlja: **za j = 1 do 10 radi:**
4 **nauči** model na skupu za učenje (4/5*9/10 skupa)
5 **izračunaj** pogrešku na skupu za provjeru (4/5*1/10 skupa)
6 izračunaj prosjek pogreške na 10 unutarnjih preklopa
7 odaberi parametare koji minimiziraju prosjek pogreške
8 **nauči** model na skupu za učenje i provjeru (4/5 skupa)
9 **izračunaj** pogrešku generalizacije na ispitnom skupu (1/5 skupa)
10 izračunaj prosjek generalizacijske pogreške na 5 vanjskih preklopa

Napomene

- Apsolutno nikada **ne smijete ispitivati model na skupu na kojem ste ga učili**. To je nepoštena procjena pogreške
- Kada se odabere optimalan model, treba ga naučiti na **uniji skupa za učenje i skupa za provjeru**, da ne gubite informaciju, a zatim ispitati na izvdojenom ispitnom skupu
- Ni na koji način ne smije se za izgradnju modela koristiti informacija iz skupa na kojem se mjeri pogreška generalizacije
- Konačna pogreška modela računa se na temelju prosjeka pogreške. Može se izračunati i standardna devijacija
- Ugniježdena *k*-struka unakrsna provjera daje nam procjenu prosječne pogreške modela, ali nam ne daje jednoznačan odgovor na pitanje koji je model niti klasifikator zapravo najbolji (optimalni modeli mogu biti različiti u svakoj iteraciji vanjske petlje)
- **Važno:** parametar *k* se ne optimizira! Ne možete isprobavati s kojim *k* klasifikator radi najbolje! Odaberite *k* i držite se toga. (*k* je zapravo hiperhiperparametar)

Problem subjektivnosti

- Sve mjere za evaluaciju klasifikacije računaju se u odnosu na ispitni skup (test set)
- Ispitni skup je referentno označen (najčešće od strane ljudi). Taj skup često se naziva i **zlatni standard** (engl. *gold standard*)
- Problem predstavljaju subjektivni klasifikacijski problemi, kod kojih ne postoji 100% slaganje među ljudima
- Potrebno je na neki način kvantificirati subjektivnost problema
- **Mjera slaganja između označivača (engl. interannotator agreement, IAA)**
- Često se koristi: **Kappa-statistika**

Kappa-statistika (1)

- Kappa-statistika računa slaganje u oznakama dvaju označivača, ali korigirano s obzirom na vjerojatnost slučajnog slaganja (engl. *chance-corrected agreement*):

$$\kappa = \frac{A_o - A_e}{1 - A_e}$$

A_o – ostvareno slaganje, A_e – očekivano slaganje

- Oba parametra računamo metodom najveće izglednosti na temelju tablice zabune

Označivač 2

Označivač 1	*Da*	*Ne*	
Da	p_{11}	p_{12}	$p_{1.}$
Ne	p_{21}	p_{22}	$p_{2.}$
	$p_{.1}$	$p_{.2}$	p

$$A_o = p_{11} + p_{22}$$

$$A_e = p_{.1}\, p_{1.} + p_{.2}\, p_{2.}$$

Kappa-statistika: primjer

Primjer *i*	Označivač 1	Označivač 2
1	Da	Ne
2	Ne	Da
3	Ne	Ne
4	Ne	Ne
5	Da	Da
6	Ne	Da
7	Ne	Ne
8	Ne	Ne
9	Ne	Ne
10	Ne	Da

Označivač 1 \ Označivač 2	*Da*	*Ne*	
Da	1	1	2
Ne	3	5	8
	4	6	10

- $A_o = p_{11} + p_{22} = 6/10 = 0.6$
- $A_e = p_{.1}\, p_{1.} + p_{.2}\, p_{2.} = 4/10*2/10 + 6/10*8/10 = 0.56$

$$\kappa = \frac{A_o - A_e}{1 - A_e} = \frac{0.6 - 0.56}{1 - 0.56} = 0.09$$

Slaganje je vrlo loše jer je očekivana vjerojatnost da oba označivača primjer slučajno označe kao “Ne” već dosta velika (0.6*0.4=0.48)

Kappa-statistika (2)

- Ne postoji suglasnost oko interpretacije ☹
- Najčešće korišteno (Krippendorff 1980):
  - **Kappa >= 0.67** – sklonost slaganju (*tentative agreement*)
  - **Kappa >= 0.8** – dobro slaganje (*good agreement*)
- Ako je vaš skup za učenje subjektivan, obavezno izračunajte kappu. Ako kappa < 0.67, imate problem
  - **Važno:** označivači mogu imati dogovorene smjernice, ali se pri samom označavanju ne smiju međusobno dogovarati!
- Kappa-statistika je statistika: ima očekivanje i varijancu
  - Za izračunatu vrijednost može se izraziti interval pouzdanosti
- Proširenja:
  - na K>2: očigledno (koristimo tablicu zabune K×K)
  - Na više od 2 označivača: problematično (najbolje računati kappa-statistiku nezavisno po svim parovima označivača)