SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

Metode ispitivanja statističke značajnosti u strojnom učenju

Filip Boltužić
Voditelj: *Prof. dr. sc. Bojana Dalbelo Bašić*

Zagreb, veljača 2014.

SADRŽAJ

1. Uvod

U strojnom učenju (engl. *machine learning – ML*) uvijek postoji potreba provjere izgrađenog sustava (za prevođenje, klasifikaciju govornih činova (engl. *speech acts*) (Pratt, 1977), za izgradnju stabla međuovisnosti (engl. *dependency tree*) (Collins, 2003), za sažimanje teksta (engl. *text summarizing*), za ekstrakciju ključnih riječi (engl. *keyword extraction*) ...). Uz argumentirano obrazloženje autora sustava o dobivenim performansama sustava, potrebno je priložiti statistički dokaz kojime se nepobitno pokazuje kako je uistinu ostvaren originalan znanstveni doprinos. Ispitivanjem statističke značajnosti pokazuje se koliko su rezultati eksperimenata vrijedni, hoće li ponavljanje eksperimenta u istim uvjetima ponovno dovesti do sličnih rezultata i u kolikoj mjeri su pouzdani. Ovaj korak je ili bi barem trebao biti sastavni dio svakog broja ML članka u kojem se predstavlja nova tehnika. Korištenjem statističkih testova u istraživanju standardizira se provođenje eksperimenata.

Dobro provedeno istraživanje trebalo bi biti sadržavati provjeru pretpostavki korištenih u statističkim analizama. Prikupljanje i odabir eksperimentalnog skupa podataka trebao bi biti opravdan i detaljno opisan, kako bi se pokazalo postoje li ograničenja skupa podataka. Primjerice, ukoliko je čitav skup podataka iz jednog novinskog izvora, a opće je poznata i prihvaćena činjenica kako su članci u tim novinama uglavnom pristrani jednoj političkoj opciji, može se pretpostaviti kako će sadržaj novinskih članaka biti sadržajem pristran.

U literaturi je moguće pronaći provjeravanje statističke značajnosti iz različitih perspektiva. Tako se (Chinchor, 1992) bavi analizom rezultata MUC-4 (engl. *Message Understanding Conference*), (Koehn, 2004) i (Zhang et al., 2004) ispituju značajnost rezultata strojnog prevođenja (engl. *machine translation*), (Bisani i Ney, 2004) analiziraju rezultate automatskog prepoznavanja govora (engl. *automated speech recognition*). (Berg-Kirkpatrick et al., 2012), (Yeh, 2000).

U ovom radu prvo će se objasniti osnove ispitivanja statističke značajnosti. Nakon toga pričat će se o mogućim testovima značajnosti u području strojnog učenja. Svaki spomenuti test bit će kratko objašnjen, s pružanjem prikladnih referenci. Objasnit će

se mehanizam pojedinih statističkih testova te prikladnost ili neprikladnost statističkog testa u kontekstu specifičnih okolnosti. U praktičnom dijelu rada pokazat će se kako odabir statističkog testa može biti jako bitan prilikom dokazivanja statističke značajnosti rezultata. Konačan cilj rada je dati čitatelju uvid u metodologije ispitivanja statističke značajnosti u kontekstu strojnog učenja, kako bi što bolje odabrao odgovarajuću tehniku u vlastitim eksperimentima.

2. Testiranje hipoteza

Testiranje hipoteza dominantan je način verifikacije rezultata prilikom objavljivanja znanstvenih radova. Ronald Fisher, otac moderne statistike, pokazao je kako je moguće dokazati ili opovrgnuti rezultate eksperimenta koristeći statističke postupke (Fisher, 1922). Prema Fisherovom testiranju hipoteza moguće je ispitati jesu li rezultati dobiveni eksperimentom statistički značajni, što bi značilo da rezultati nisu dobiveni pukom slučajnošću, već da ih je moguće reproducirati u istovjetnim uvjetima. Razvojem statistike utemeljen je uobičajen način ispitivanja statističke značajnosti prema kojem se:

1. postavlja inicijalna hipoteza istraživanja,
2. definiraju nulta H_0 i alternativna hipoteza H_1 (engl. *null and alternative hypothesis*),
3. promatraju statistička obilježja podataka nad kojima će se provoditi odgovarajući statistički testovi (engl. *statistical tests*),
4. na temelju rezultata prethodnog koraka odabire odgovarajući statistički test,
5. odabire relevantna statistička mjera T (engl. *test statistic*),
6. računa distribucija odabrane statističke mjere T pod pretpostavkom da je nulta hipoteza zadovoljena (ove vrijednosti često su unaprijed izračunate i pohranjene u tablicama (Wilcoxon et al., 1973))
7. odabire razina značajnosti α (engl. *significance level*), razina vjerojatnosti ispod koje se odbacuje nulta hipoteza (česti odabiri su 1% ili 5%, ovisno o eksperimentu),
8. računaju granične vrijednosti (engl. *critical region*) distribucije statističke mjere T za razinu značajnosti α,

9. računa statistička mjera T_{obs} dobivena iz eksperimentalnih (stvarnih) podataka,

10. donosi odluka o odbacivanju nulte hipoteze ukoliko je dobivena vrijednost T_{obs} unutar graničnih vrijednosti.

Moguć je i alternativan scenarij prema kojem se na temelju dobivene T_{obs} vrijednosti računa p-vrijednost (engl. *p-value*), vjerojatnost $P(H_0|D)$ dobivenih podataka D pod pretpostavkom nulte hipoteze H_0 na temelju koje se donosi odluka o odbacivanju nulte hipoteze. U slučaju kada je p-vrijednost manja od postavljene razine statističke značajnosti α, nije moguće odbaciti nultu hipotezu zbog nedovoljno dokaza. P-vrijednost naziva se i empirijska razina značajnosti, a ne ovisi o vrstu testa i odabranoj statističkoj mjeri. Iz tog razloga u većini radova interpretacija rezultata statističkog testiranja tumači se putem p-vrijednosti.

Eksperiment kojim je Fisher predstavio testiranje hipotezi je damin test kušanja čaja (engl. *lady tasting tea*). Testom se pokušalo ustanoviti može li dama (u Fisherovom slučaju Muriel Bristol) temeljem okusa razlikovati čaj s mlijekom prema načinu spravljanja napitka (prvo čaj, zatim mlijeko ili obratno). U ovom slučaju nulta hipoteza je pretpostavka da dama ne može razlikovati čaj s mlijekom sudeći samo prema okusu. Dami je predstavljeno osam šalica čaja s mlijekom za koje je morala opisati način pripreme. Dobiveni rezultati uspoređuju se s pravim vrijednostima, te se na temelju toga provodi statistička verifikacija rezultata. Eksperiment je detaljno objašnjen u Fisherovom radu Fisher (1935). Test se smatra izuzetno bitnim za razvoj polja statistike (Potter, 2001).

2.1. Vrste pogrešaka

Prilikom provođenja statističkog testa, na temelju dobivenih statističkih mjera ili p-vrijednosti, donosi se odluka o prihvaćanju ili odbacivanju nulte hipoteze. Sama odluka može biti pogrešna i to na dva načina:

- pogrešno je odbačena istinita nulta hipoteza – pogreška tipa I (engl. *type I error*) ili
- je donesena odluka kojom se ne odbacuje pogrešna nulta hipoteza – pogreška tipa II (engl. *type II error*).

Pogreška tipa I pogrešno zaključuje da je zadovoljena pretpostavka koja se eksperimentom htjela statistički dokazati. Pogrešna tipa II odbacuje istinitu pretpostavku. Primjerice, konstruirana su dva klasifikatora C_A i C_B. Nulta hipoteza pretpostavlja

kako su njihove performanse jednake. Pogreška tipa I bi se dogodila u slučaju da prihvatimo da je C_A bolji od C_B, a da u stvarnosti rade jednako dobro. Kada bi uistinu C_A radio bolje od C_B, a takva teza bi se odbacila statističkim testom, to znači kako je došlo do pogreške tipa II. Pogreške tipa I i II su osnovne statističke pogreške. Postoje dodatna proširenja ovih osnovnih pogrešaka, a često ovise o području primjene (medicina, računalna sigurnost, telekomunikacije ...).

2.2. Kritike testiranja hipoteza

Testiranje hipoteza primjenjuje se u gotovo svim znanstvenim disciplinama. No, postupak dokazivanja statističke značajnosti je podložan brojnim kritikama. Objavljen je izuzetno velik broj radova koji kritiziraju provedbe sumnjivih statističkih postupaka, kao što su (Hedges et al., 1985), (Dar et al., 1994), (Yoccoz, 1991). Najprodavanija knjiga iz statistike *How to lie with statistics* (Huff, 2010) pokušava, na način pristupačan široj publici, demonstrirati na koji je način moguće slučajno ili namjerno iskoristiti moć statističkog zaključivanja na pogrešan način. Razlozi pogrešnog provođenja statističkog testa su pogrešno tumačenje ili preskakanje jednog od koraka navedenih na početku poglavlja.

Bruce Thompson (Thompson, 1993) navodi neke kritike konvencionalnih metoda statističkog testiranja:

1. Nulta hipoteza će se **uvijek** odbaciti, ako uzme u obzir dovoljno velika populacija. Thompson kaže:

> Ispitivanje statističke značajnosti može biti vođeno tautologijom. Umorni istraživači, nakon prikupljanja skupa podataka, rade statističke testove kako bi se uvjerili da je prikupljena dovoljno velika količina podataka, što je podatak koji već znaju. Ovakva tautologija učinila je mnogo štete znanstvenoj zajednici.

2. Korištenje analize varijance (engl. *analysis of variance – ANOVA*) može dovesti do pogrešnih usporedbi. U višedimenzionalnim analizama primjenom hijerarhijskog pristupa moguće je povlačiti usporedbe između podataka iz različitih dimenzija, analogno poslovici: usporedba krušaka i jabuka.

3. ANOVA zahtjeva spajanje varijanci prilikom izračuna srednje devijacije (u nazivniku). Ova operacija je dozvoljena samo u slučaju da su varijable homogene, međusobno usporedive. Slično tome, analiza kovarijance (engl. *analysis of covariance –*

ANCOVA) pretpostavlja da je zadovoljen uvjet *homogenosti regresije* (engl. *Homogeneity of Regression Slopes*). Ovo su preduvjeti koje brojni istraživači ne provjeravaju (Thompson, 1993).

3. Bayesov pristup testiranju hipoteza

Testiranje hipoteza opisano u poglavlju 2 naziva se frekvencijski pristup testiranju. Takav način testiranja uvijek uključuje postavljanje dvije oprečne hipoteze H_0 i H_1 i ispitivanje testne statistike, često putem p-vrijednosti ili intervala pouzdanosti. Frekvencijski pristup testiranju daje zaključke o eksperimentu koji su valjani ukoliko se razmatra iznimno velik broj ponavljanja eksperimenta. Bayesovim pristupom pokušava se kompenzirati nemogućnost višestrukog ponavljanja eksperimenta, već se, prilikom izračuna statističke značajnosti, uzimaju u obzir prethodno stečena znanja poznata prije odvijanja eksperimenta (*apriori* znanja). U pojedinim situacijama moguće je primjeniti samo Bayesov način testiranja, jer nije moguće ponoviti eksperiment više od jednom. Primjerice, donošenje sudskih odluka moguće je samo na temelju dokaza dostupnih za jedan, konkretni slučaj.

Bayesov pristup testiranju alternativa je frekvencijskom pristupu testiranja. U Bayesovu slučaju moguće je razmatrati više od dvije hipoteze. Temelji se na dobro poznatom Bayesovom teoremu (Pawlak, 2002) u kojem modeliramo svijet temeljem *a priori* i *a posteriori* vjerojatnosti. Primjerice, potrebno je provjeriti ispravnost hipoteza H_1 i H_2, na temelju dobivenih podataka x. Potrebno je izračunati *posteriori* vjerojatnosti $P(H_1|x)$ i $P(H_2|x)$ pomoću Bayesove formule:

$$P(H_1|x) = \frac{P(x|H_1)P(H_1)}{P(x)} \tag{3.1}$$

$$P(H_2|x) = 1 - P(H_1|x) \tag{3.2}$$

A priori vjerojatnost dobivenih podataka $P(x)$ je vjerojatnost podataka prema svim pretpostavljenim hipotezama (ovdje H_1 i H_2):

$$P(x) = \sum_i P(x|H_i)P(H_i) \tag{3.3}$$

Prednost Bayesovog načina testiranja je mogućnost dodavanja vlastitih pretpostavki o eksperimentu prije obavljanja testiranja. Temeljem zdravog razuma, prethodnog iskustva ili predrasuda moguće je definirati a priori vjerojatnosti ishoda eksperimenta. Primjerice, želimo li ispitivati je li osoba duge kosa muškog ili ženskog spola, vjerojatno ćemo postaviti visoku a priori vjerojatnost temeljem stvarnog iskustva. Takve procjene nije uvijek lako donositi, što je jedna često raspravljana tema u okviru Bayesovog pristupa testiranju (Gelman i Shalizi, 2012). Također, moguće je provoditi ispitivanje statističke značajnosti na Bayesov način koristeći Bayesov faktor (engl. *Bayes factor*), omjer vjerojatnosti hipoteza temeljem podataka:

$$\frac{P(x|H_1)}{P(x|H_2)} \tag{3.4}$$

Frekvencijsko testiranje se najčešće koristi prilikom statističkog ispitivanja značajnosti, zbog lakoće aproksimacija procesa frekvencijskim vjerojatnostima. Korištenje Bayesovog načina testiranja zahtjeva procjenjivanje parametara, što je, do napretka dobivenih *Monte Carlo, MC* (Hammersley et al., 1965) metodama, bilo izuzetno matematički zahtjevno. Kombinacijom današnje računalne snage i MC metoda moguće je promatrati ispitivanje statističke značajnosti na Bayesov način, te se smatra kako će ovakav način testiranja postati dominantan u literaturi kroz nekoliko godina (Gelman i Shalizi, 2012).

4. Mjerenje rezultata sustava u strojnom učenju

Ispitivanje statističke značajnosti rezultata često se obavlja u kontekstu strojnog učenja. Izrada novih tehnika i metoda strojnog učenja zahtjeva statističku analizu i dokazivanje.

4.1. Algoritmi učenja u strojnom učenju

U ovom odjeljku objasnit ćemo pojmove vezane za strojno učenje korištene u ostatku rada. Postoji mnogo referenci za upoznavanje s područjem strojnog učenja, kao što su (Anderson et al., 1986) i (Bishop i Nasrabadi, 2006). Kako se strojno učenje primjenjuje u sve više raznovrsnih područja, tako postoji i literatura za upoznavanje sa strojnim učenjem, kao što je (Baldi et al., 2001).

Algoritmima učenja izgrađujemo klasifikatore, kojima automatski kategoriziramo stvari: klasifikacija novčanica u aparatima za kavu, klasifikacija otiska prstiju, prepoznavanje prometnih znakova . . . Razlika između algoritma učenja i klasifikatora je u tome što je klasifikator istrenirani proizvod bez mogućnosti daljnjih modifikacija ponašanja. Algoritam učenja može proizvesti jedan ili više klasfikatora. Ponekad se uspoređuju algoritmi učenja, a ponekad klasifikatori, ovisno o primjeni. Primjerice, potrebno je izgraditi stroj za automatsko prepoznavanje vrsta riječi u obradi prirodnog teksta: moguće je pokušati uporabiti različite algoritme učenja za dobivanje klasifikatora. Moguće je uspoređivati rezultate algoritama učenja, ali i samih klasifikatora.

Riječ	lopata	život	ja	ljudi	cvijet	ili	svijet	aha	ritam	latica
Ispravno	T	F	F	T	F	F	T	T	T	F
Dobiveno	T	T	T	F	F	F	T	T	T	T

Tablica 4.1: Rezultat klasifikatora

U kontekstu učenja kod čovjeka, moguće je uspoređivati tehnike učenja, primjerice tzv. kampanjski način učenja s redovitim učenjem, što bi odgovaralo usporedbi algoritama učenja. Uspoređivanje rezultata studenata na ispitima odgovara usporedbi klasifikatora.

4.2. Mjerenje performansi modela strojnog učenja

Primjerice, radi se na automatskom prepoznavanju ključnih riječi u tekstu temeljem algoritama učenja baziranih na strojnom učenju. Algoritmi proizvode klasifikatore koji obavljaju prepoznavanje ključnih riječi. Cilj istraživanja je usporedba i verifikacija performansi dvaju klasifikatora ključnih riječi, u svrhu pronalska optimalnog klasifikatora. Ulaz klasifikatoru je dokument, a izlaz klasifikatora je niz ključnih riječi. Postoje referentne vrijednosti (engl. *gold standard*), koje se smatraju ispravnim ključnim riječima pojedinog dokumenta s kojima se uspoređuje rezultat dobiven klasifikatorom. U tablici 4.1 dan je primjer ispitivanja rezultata klasifikatora. Jedan stupac tablice govori kako je klasificirana riječ u odnosu na točan rezultat (*T* je točno, *N* netočno). Iz navedene tablice možemo vidjeti kako su moguće četiri kombinacije podudaranja izlaza klasifikatora i stvarne vrijednosti.

Popisivanje svih rezultata može biti nepregledno, pogotovo kod većeg broja testnih primjera, stoga se koriste tablice kontigencije (engl. *contigency tables*). Njima se dobivaju agregirane vrijednosti ispitivanja klasifikatora, a daju pregled koliko ima

- ispravno klasificiranih pozitivnih primjera (engl. *true positives*) –TP,
- ispravno klasificiranih negativnih primjera (engl. *true negatives*) – TN,
- pozitivnih primjera pogrešno svrstanih u negativne (engl. *false positives*) – FP i
- negativnih primjera pogrešno svrstanih u pozitivne (engl. *false negatives*) – FN.

Upravo su ovo sve četiri moguće kombinacije ispitivanja rezultata klasifikatora. Općenit oblik tablice kontigencije prikazan je na tablici 4.2. Tablicu kontigencije za primjer iz tablice 4.1 moguće je vidjeti u tablici 4.3. Ovakav oblik tablica kontigencija često se koristi u strojnom učenju, kao u radu Hall (1999).

Tablica kontigencije predstavlja polaznu točku nekim naprednijim mjerama performansi, a najčešća korištena mjera je preciznost (engl. *accuracy – ACC*), računa se

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

.

		Ispravno	
		T	F
Dobiveno	T	TP	FP
	F	FN	TN

Tablica 4.2: Oblik tablice kontigencije

		Ispravno	
		T	F
Dobiveno	T	4	3
	F	1	2

Tablica 4.3: Primjer tablice kontigencije

U praksi, koriste se i mnoga druga mjerila izvedbe klasifikatora, mnoga opisana u radu Powers (2011). Mjere često ovise i o prirodi samog zadatka. Primjerice, strojni prevoditelji rangiraju se prema *BLEU* rezultatu, opisanom u (Papineni et al., 2002). U ovome radu spominjat će se neke druge mjere, no neće se eksplicitno objašnjavati.

4.3. Metodologija ispitivanja značajnosti

Potraga za novim, optimalnim algoritmom učenja za zadatak strojnog učenja često uključuje izgradnju više modela s različitim algoritmima i promatranje ponašanja tih modela nad različitim skupovima podataka. Ciljevi takvih istraživanja mogu biti: kako konstruirati klasifikator koji u prosjeku radi vrlo dobro nad svim dostupnim skupovima podataka, koji klasifikator pruža najbolje rezultate nad malim (specifičnim) skupovima podataka, kada koristiti algoritme učenja, a kada klasifikator i sl. Ukoliko se radi o novom zadatku, ne postoji referentni sustav s kojim bi se novoizgrađeni sustav mogao usporediti. U slučaju napadanja poznatog, donekle rješenog problema potrebno je usporediti perfomanse novodobivenih postupaka s trenutno najboljim. Također, moguće je pretpostaviti da je na raspolaganju jedan ili više skupova podataka. Ti skupovi podataka možda dolaze iz različite domene. Na primjer, izdvajanje ključnih riječi iz skupova podataka S_1 i S_2, gdje je S_1 prikupljen uglavnom iz novinskih članaka, dok se do S_2 došlo sabiranjem proznih djela.

Najjednostavniji (generički) slučaj statističkog dokaza izrade klasifikatora zahtjeva:

istraživačku hipotezu,

Novi klasifikator C_A je precizniji od trenutno općeprihvaćenog klasifikatora C_B. Viši iznos vrijednosti znači da klasifikator radi bolje (točnije).

nultu hipotezu H_0,

Ne postoji razlika u performansama između klasifikatora *A* i *B*.

skup podataka,

Postoji jedinstven skup podataka S (engl. *dataset*) – populacija. Moguće je uzimati uzorke x iz populacije.

način mjerenja

f_A je rezultat klasifikatora C_A, f_B je rezultat klasifikatora C_B. $d_{A,B} = f_A - f_B$ je razlika u perfomansama između klasifikatora C_A i C_B

Provođenje testiranja može se odvijati na čitavoj populaciji, ili se može uzorkovati nad populacijom i ispitivati ponašanje C_A i C_B na uzorcima x veličine n iz populacije S. U slučaju uzorkovanja, testiranje hipoteze procjenjuje kolika je vjerojatnost:

$$p(d_{C_A,C_B}(X) > d_{C_A,C_B}(x)|H_0) < \alpha \tag{4.2}$$

gdje je *X* slučajna varijabla mogućih dobivenih uzoraka veličine *n*, a $d_{C_A,C_B}(x)$ promatrana (konstantna) vrijednost. Ako vrijedi 4.2 za odabranu vrijednost α (najčešće 0.05) onda se odbacuje nulta hipoteza. Dva su osnovna načina izračuna p-vrijednosti:

- izravnim izračunom,
- procjenom.

U idućem poglavlju bit će govora o oba načina. Prvo će se pokazati na primjerima iz strojnog učenja kako je moguće izračunati p-vrijednost s obzirom na različite okolnosti eksperimenta. Aproksimacija p-vrijednosti koristi se kao alternativa u uvjetima kada nije moguće analitički izračunati p-vrijednost. Na čitatelju ostaje prepoznati uvjete vlastitog istraživanja te se prema tome odlučiti za jedan od ova dva načina.

5. Ispitivanje statističke značajnosti u strojnom učenju

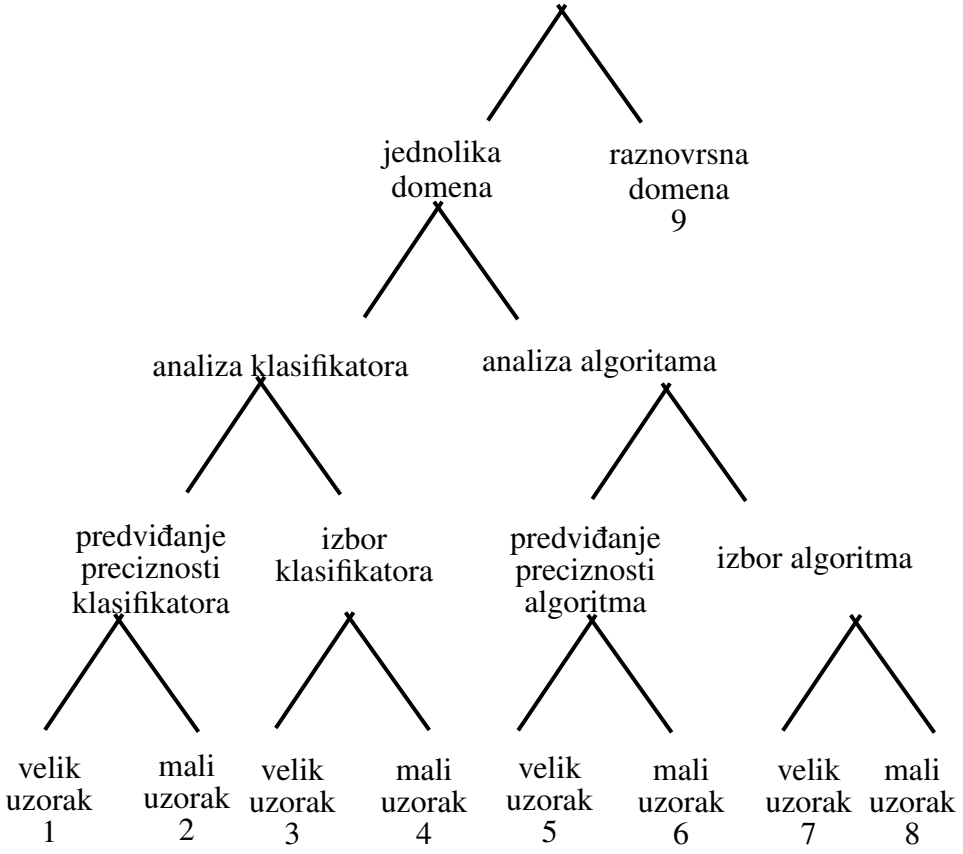
Prema (Dietterich, 1998) postoji devet temeljnih statističkih pitanja u strojnom učenju, prikazanih na slici 5.1. Pitanja su hijerarhijski organizirana u četiri razine, razina:

- domene – jednolika ili raznolika,
- tehnike – odabir klasifikatora ili algoritma učenja,
- vrste zadatka – predviđanje ili usporedba algoritama/klasifikatora i
- veličine populacije – mala ili velika. .

Svaki list stabla predstavlja jedno pitanje, a do pitanja se dolazi spajanjem vrijednosti pojedinih čvorova prethodnika sve dok se ne dođe do korijena. Primjerice, kako predvidjeti preciznost klasifikatora u velikom, jednolikom skupu podataka je pitanje označeno brojem 1 na slici. Također, bitno je napomenuti kako je bitan faktor i brojnost korištenih metoda tj. broj algoritama učenja ili klasifikatora (Demšar, 2006).

5.1. Terminologija

Moguće je uspoređivati najmanje dva algoritma učenja ili klasifikatora, ali i više. Prilikom usporedbe dvaju algoritama učenja koristit će se imena A i B, a proizvodi tih algoritama učenja, klasifikatori, imat će oznate C_A i C_B. Niz cjelobrojnih vrijednosti $1 \ldots i$ služiti razlikovanju više od dva algoritma učenja, a pripadajući klasifikatori označavat će se sa $C_1 \ldots C_i$. Skup podataka bit će označen znakom S, a njegova veličina oznakom n. Svi ostali pojmovi bit će objašnjeni u odjeljku u kojem se prvi put spominju.



Slika 5.1: Statistička pitanja u strojnom učenju

5.2. Statistička pitanja usporedbe u strojnom učenju

Ovaj odjeljak pokušat će odgovoriti na statistička pitanja vezana uz usporedbu algoritama učenja/klasifikatora u strojnom učenju (3, 4, 7, 8), usporediti algoritme učenja i klasifikatore nad raznolikim skupovima podataka (pitanje 9) te opisati načine ispitivanja statističke značajnosti u okruženju s više od dva algoritma učenja ili klasifikatora.

5.2.1. Pitanje 3 – usporedba klasifikatora na velikom skupu podataka

Konstruirani su klasifikatori C_A i C_B. Moguće je izdvojiti zaseban skup podataka T nad kojim će se testirati dobiveni klasifikatori. Potrebno je mjeriti performanse svakog klasifikatora na posebnom skupu za testiranje te primjeniti McNemarov test.

McNemarov test

Za McNemarov test, opisan u (Everitt, 1992) potrebno je populaciju S podijeliti na skupove za treniranje R i testiranje T. Algoritmi A i B proizvode klasifikatore C_A i C_B koji produciraju rezultate klasifikacije nad skupom podataka za testiranje. Provođenje testa zahtjeva izgradnju tablice kontigencije:

broj primjera koje su C_A i C_B pogrešno klasificirali	broj primjera koje je C_A pogrešno, a C_B ispravno klasificirao
broj primjera koje je C_B pogrešno, a C_A ispravno klasificirao	broj primjera koje su C_A i C_B ispravno klasificirali

skraćeno: gdje je $n = n_{00} + n_{01} + n_{10} + n_{11}$ ukupan broj primjera u skupu za

n_{00}	n_{01}
n_{10}	n_{11}

testiranje T. Prema nultoj hipotezi, oba klasifikatora trebala bi imati jednak broj pogrešaka $n_{10} = n_{01}$. McNemarov test, zasnovan na χ^2 testu, ispituje koliko vjerojatnost raspodjele broja pogrešno klasificiranih primjera prema nultoj hipotezi, a očekivana distribucija može se prikazati kontigencijskom tablicom:

n_{00}	$(n_{01} + n_{10})/2$
$(n_{01} + n_{10})/2$	n_{11}

U ovom slučaju χ^2 statistika iznosi:

$$T_{obs} = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \tag{5.1}$$

Kako bi se odbacila nulta hipoteza, potrebno je da vrijednost dobivena iz testnog skupa podataka veća od granične vrijednosti odgovarajućih stupnjeva slobode ($\chi^2_{1,0.95} = 3.84$).

5.2.2. Pitanje 4 – usporedba klasifikatora na malom skupu podataka

Dva klasifikatora, C_A i C_B dobiveni su treniranjem algoritama učenja A i B nad malim skupom podataka S. Ovaj problem (Dietterich, 1998) izjednačava s pitanjem 8 u odjeljku 5.2.4, gdje će biti objašnjene potencijalne metode za rješenje problema.

5.2.3. Pitanje 7 – usporedba algoritama učenja na velikom skupu podataka

Algoritmi učenja A i B treniraju se i testiraju na velikom skupu podataka S. Odgovor na ovo pitanje dobiveno je u istraživanju u okviru DELVE projekta (Hinton et al.): potrebno je podijeliti skup S na disjunktne skupove za trening te izdvojiti jedan skup za test. Klasifikatori dobiveni treniranjem algoritama učenja nad svim skupovima za treniranje se ispituju nad rezultatima testova. Moguće je primijeniti ANOVU s varijablama

- izbora algoritma učenja,
- izbora skupa za treniranje i
- za svaku jedinku iz skupa za testiranje.

Kvazi-F testom (Eisen, 1966) se utvrđuje postoji li statistička značajna razlika između različitih algoritama učenja.

5.2.4. Pitanje 8 – usporedba algoritama učenja na malom skupu podataka

Dva algoritma učenja A i B potrebno je istrenirati na skupu S i provjeriti koji će od algoritama dati bolje rezultate na novom skupu jednake veličine kao i S. U radu (Dietterich, 1998) opisuje se pet različitih statističkih testova za usporedbu algoritama učenja s ograničenim skupom podataka.

McNemarov test

Ovdje je ponovno moguće je uporabiti McNemarov test opisan u odjeljku 5.2.1. No, nedostatak McNemarovog testa je činjenica da se skup za treniranje R ($|R| << |S|$) odabire jednom, a nad njim se potom rade sve usporedbe. Prema tome, McNemarov test je primjenjiv ukoliko je varijabilnost podataka mala, jer se radi o podskupu mnogo većeg (populacijskog) skupa S. Pretpostavlja se kako sve što vrijedi za odabrani skup R, vrijedi i za čitavu populaciju S, što ne mora biti istina u praksi.

Test razlika

Test razlika (engl. *Test for the difference of two proportions*) (Snedecor i Cochran, 1980) mjeri razinu pogreške kod algoritama A i B preko vjerojatnosti:

$$p_A = \frac{n_{00} + n_{01}}{n} \tag{5.2}$$

$$p_B = \frac{n_{00} + n_{10}}{n} \tag{5.3}$$

Ako su p_A i p_B su vjerojatnosti pogrešne klasifikacije algoritama A i B, broj pogrešno klasificiranih N primjera zadovoljava binomnu raspodjelu srednje vrijednosti Np_A i varijance $p_A(1 - p_A)N$. Binomna raspodjela prelazi u normalnu za dovoljno velik broj primjera n. Ako pretpostavimo da su p_A i p_B nezavisni, veličina $p = (p_A + p_B)/2$ također zadovoljava normalnu raspodjelu. Da bi nulta hipoteza bila zadovoljena, srednja vrijednost izvedene veličine p bit će nula, a standardna pogreška i z vrijednost:

$$se = \sqrt{\frac{2p(1-p)}{n}}. \tag{5.4}$$

$$z = \frac{p_A - p_B}{\sqrt{2p(1-p)/n}}. \tag{5.5}$$

Nulta hipoteza se odbija ukoliko je dobivena z vrijednost van graničnih vrijednosti ($|z| > Z_{0.975} = 1.96$ kod dvostranog testa s $\alpha = 0.05$).

Prilikom postavljanja testa, pogrešno se pretpostavlja nezavisnost mjera p_A i p_B, mjerenih nad istim skupom podataka T. Također, test, kao i McNemarov, uzima u obzir jedinstveni skup za treniranje R.

T-test s resempliranjem

T-test s resempliranjem je najkorišteniji test u strojnom učenju, prema (Dietterich, 1998). Prvi korak je izrada uzoraka R, unaprijed poznate veličine, iz skupa S i kreiranje testnih skupova T na temelju uzoraka. Uzorkovanje se ponavlja; često korišten broj ponavljanja je 30. p_A^i i p_B^i su vjerojatnosti pogrešne klasifikacije algoritama klasifikatora dobivenih algoritmima A i B nad i-tim skupom podataka. Pod pretpostavkom da je uzorkovanje neovisno mjera $p^{(i)} = p_A^{(i)} - p_B^{(i)}$ moguće je primjeniti Studentov t-test, gdje se t-statistika dobije formulom:

$$\hat{p} = \frac{1}{n}\sum_{i=1}^{n} p^{(i)} \tag{5.6}$$

$$t = \frac{\hat{p} \cdot \sqrt{n}}{\frac{\sqrt{\sum_{i=1}^{n}(p^{(i)} - \hat{p})^2}}{\sqrt{n-1}}}. \tag{5.7}$$

Nulta hipoteza se odbija ako je dobivena t vrijednost veća od odgovarajuće (tablične) t vrijednosti (primjerice, $|t| > t_{29,0.975} = 2.05$ za 30 uzorkovanja). Ponovno, radi se pretpostavka nezavisnosti varijabli $p_A^{(i)}$ i $p_B^{(i)}$ do kojih se dolazi uzorkovanjem iz istog skupa podataka. Druga pogrešna pretpostavka nezavisnosti radi se prilikom zbrajanja

$p^{(i)}$ vrijednosti iz različitih uzorkovanja. Ovdje pretpostavka nezavisnosti ne stoji zbog mogućih preklapanja uzoraka.

Upareni t-test s k-strukom unakrsnom validacijom

Jedna od mana t-testa s resempliranjem je (ponekad) neispravna pretpostavka o nezavisnosti podataka u različitim uzorcima. Upareni t-test s k-strukom unakrsnom validacijom dijeli skup podataka S na k disjunktnih skupova za testiranje jednake veličine $T_1, \ldots, T_k$. Skup za treniranje u koraku i dobiva se unijom svih skupova $T_j, j \neq i$. Sve ostalo identično je kao i u prethodno objašnjenom t-testu. Prema (Dietterich, 1998) jedina mana ovog testa je eksperimentalno dobiveni visoki korelacijski faktor između različitih skupova za treniranje.

5x2cv upareni t-test

Kako bi prebrodili mane uparenog t-testa s k-strukom unakrsnom validacijom (Dietterich, 1998) je primijetio kako bilo kakvo preklapanje u skupovima podataka za testiranje ili treniranje narušava performanse testa. Iz tog razloga, osmišljen je 5x2cv upareni t-test. U tom testu pet puta se ponavlja unakrsna validacija s dva preklapanja (engl. *5 replications of 2-fold cross validation*). Prilikom svake podjele skupa podataka S dobivaju se skupovi S_1 i S_2. Algoritmi A i B koriste oba skupa koriste se za treniranje i testiranje. Ispitivanjem dobivenih klasifikatora dobiju se vjerojatnosti: $p_A^{(1)}$, $p_A^{(2)}$, $p_B^{(1)}$, $p_B^{(2)}$, iz njih razlike $p^{(1)} = p_A^{(1)} - p_B^{(1)}$ i $p^{(2)} = p_A^{(2)} - p_B^{(2)}$, iz razlika varijanca $s^2 = (p^{(1)} - \hat{p})^2 + (p^{(2)} - \hat{p})^2$, s time da je $\hat{p} = (p^{(1)} + p^{(2)})/2$. s_i^2 je varijanca dobivena u i-itom (od 5) ponavljanju. Iz toga moguće je dobiti vrijednost t statistike, 5x2cv $\tilde{t}$ vrijednost iz:

$$T_{obs} = \frac{p_1^{(1)}}{\sqrt{\frac{1}{5}\sum_{i=1}^{5} s_i^2}} \tag{5.8}$$

Prema nultoj hipotezi dobivena 5x2cv $\tilde{t}$ statistika poprima obilježja t-raspodjele s pet stupnjeva slobode. (Dietterich, 1998) je teorijski i eksperimentalno dokazao ovu tvrdnju. Alpaydm u svom radu (Alpaydm, 1999) primjećuje kako vrijednost dobivene $\tilde{t}$ statistike ovisi o izboru $p_i^{(j)}$ vjerojatnosti (ovdje $p_1^{(1)}$). Iz tog razloga predlaže poboljšanje 5x2cv t-testa, kombinirani 5x2cv F test koji uzima u obzir sve dobivene $p_i^{(j)}$ vjerojatnosti.

5.2.5. Pitanje 9 – usporedba na raznolikom skupu podataka

Analizom objavljenih radova na konferencijama *International Conference of Machine Learning* između 1999. i 2003. godine (Demšar, 2006) detektirane su metode usporedbi klasifikatora nad različitim skupovima podataka:

- prosjek preciznosti u različitim skupovima podataka,
- t-test,
- upareni t-test jedan protiv ostalih,
- upareni t-test svatko protiv svakog,
- prebrojavanje pobjeda/nerješenih rezultata/poraza i
- prebrojavanje statistički značajnih pobjeda/nerješenih rezultata/poraza

Opće je prihvaćeno da je korištenje t-testa za usporedbu klasifikatora na različitim skupovima podataka neprimjereno. U radu (Demšar, 2006) preporučuje se Wilcoxonov test rangova s predznacima (engl. *Wilcoxon ranked sign test*) (Wilcoxon, 1945). Uz njega, koristi se i test s predznacima Dixon i Mood (1946).

U idućim pododjeljcima opisat će se različiti načini mjerenja statističke značajnosti viđeni u znanstvenim publikacijama.

Računanje prosjeka

Mjerenja performansi klasifikatora nad različitim skupovima podataka moguće je jednostavno zbrojiti i uzeti srednju vrijednost:

$$\hat{X} = \frac{1}{N} \sum_{i=1}^{N} f_i^j. \tag{5.9}$$

gdje je $\hat{X}$ je mjera dobrote klasifikacije, N broj skupova podataka, a f_i^j izvedba klasifikatora f^j (ovdje $j \in (1, 2)$) nad skupom podataka i.

(Webb, 2000) napominje kako ovakav način često nije smislen, jer usporedba grešaka između različitih skupova podataka iz različitih domena nije nešto što bi trebalo uspoređivati. Ukoliko nije smisleno raditi usporedbe rezultata u različitim domenama, zbrajanje tih vrijednosti također nije valjana operacija. Naravno, postoje slučajevi u kojima je ova metoda ispravna, stoga je ne valja u potpunosti odbaciti.

Upareni t-test

Upareni t-test (Sprinthall i Fisk, 1990) uzima u obzir prosječnu razliku u performansama nad različitim skupovima podataka i provjerava ima li statistički značajne razlike.

Neka su c_i^1 i c_i^2 rezultati dva klasifikatora na skupu podataka i, od ukupno N skupova podataka. Razlika d_i se definira kao $c_i^1 - c_i^2$. Tada se *t* vrijednost se dobije $\hat{d}/\sigma_{\hat{d}}$, gdje je $\sigma_{\hat{d}}$ varijanca varijable d_i raspoređenoj prema Studentovoj raspodjeli (engl. *Student distribution*) s $N - 1$ stupnjeva slobode.

Tri su temeljne zamjerke korištenja t-testa pri evaluaciji klasifikatora. Prva zamjerka spomenuta je prilikom mjerenja značajnosti računanjem prosjeka. U oba slučaja rade se usporedbe performansi između različitih skupova podataka, ali u ovom slučaju uzima se u obzir varijanca između skupova podataka, što donekle umanjuje standardnu pogrešku σ_d, no i dalje se radi dvojbena usporedba različitih skupova podataka. U slučaju da je uzorak manji od (približno) 30 jedinki, korištenje t-testa zahtjeva da uzorak zadovoljava uvjete normalne razdiobe. Dakle, problematično je korištenje t-testa sa malim uzorcima, jer nije moguće pouzdano ispitati normalnost nad malim skupom podataka (Razali i Wah, 2011). Treći problem korištenja t-testa je problem ekstremnih vrijednosti. Ekstremne vrijednosti (jako dobri ili jako loši rezultati) mogu pomaknuti distribuciju podataka i tako umanjiti snagu t-testa.

Wilcoxonov test rangova s predznacima

Wilcoxonov test rangova s predznacima (Wilcoxon, 1945) je neparametarska verzija uparenog t-testa. Razlike u performansama klasifikatora se rangiraju za svaki skup podataka. Rangovi se formiraju prema apsolutnim vrijednostima razlika, a potom se uspoređuju rangovi pozitivnih i negativnih razlika. U slučaju jednakih vrijednosti, u rang se upisuje prosječna vrijednost. Kao što smo opisali kod uparenog t-testa, d_i će predstavljati razliku performansi dva klasifikatora na i-tom skupu podataka (od ukupno N). Ako razlikujemo klasifikatore 1 i 2, potrebno je zbrojiti rangove R^1 i R^2, prema formulama:

$$R^1 = \sum_{d_i>0} rank(d_i) + \frac{1}{2}\sum_{d_i=0} rank(d_i)$$

$$R^2 = \sum_{d_i<0} rank(d_i) + \frac{1}{2}\sum_{d_i=0} rank(d_i)$$

.

Neriješeni slučajevi se ravnomjerno raspoređuju u svaku sumu. Ukoliko postoji neparan broj neriješenih slučajeva, jedan neriješeni slučaj se zanemaruje, kako bi brojka uvijek bila jednaka. Rezultat je statistički značajan ukoliko je je dobivena vrijednost veća od granične T vrijednosti. T vrijednost se dobiva kao $min(R^1, R^2)$ Za uzorke

broj uzoraka	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
$\alpha = 0.10$	0	0	0	1	1	1	2	2	3	3	3	4	4	5	5	5	6	6	7	7	7

Tablica 5.1: Kritične vrijednosti za dvostrani test znakova. Ukoliko je broj pobjeda manji ili jednak onome iz drugog retka, razlika je statistički značajna.

manje od 25 moguće je pronaći točnih graničnih T vrijednosti (Wilcoxon et al., 1973), a za ostale potrebno je izračunati z statistiku:

$$z = \frac{T - \frac{1}{4}N(N+1)}{\sqrt{\frac{1}{24}N(N+1)(2N+1)}}$$

Wilcoxonov test također radi usporedbe različitih skupova podataka, ali ne količinski, zbrajanjem iznosa razlika, već kvalitativno, zanemarujući apsolutne iznose razlika. Pretpostavka da su podatci normalno distribuirani nije potrebna, jer je Wilcoxonov test neparametarski. Utjecaj graničnih vrijednosti je smanjen jer se sada promatraju rangovi, umjesto apsolutnih vrijednosti.

Wilcoxonov test ima manju snagu od uparenog t-testa kada su podatci normalno distribuirani. U suprotnome, Wilcoxonov test može, ali i ne mora prema (Demšar, 2006) biti jači od t-testa.

Test znakova

Zanemarivanjem vrijednosti razlika u performansama klasifikatora isptivanje značajnosti svodi se na bilježenje rezultata kroz pobjede, poraze i izjednačene rezultate. Ako se uspoređuju dva algoritma na ovaj način, nulta hipoteza pretpostavlja kako će svaki algoritam imati približno $N/2$ pobjeda u skupu od N podataka. Pošto razlikujemo dva ishoda, moguće je modelirati podatke binomnom distribucijom (Miller et al., 1965) i provesti binomni test (Dixon i Mood, 1946), test znakova (engl. *sign test*). Postoje izračunate vrijednosti za test znakova vidljive u literaturi (Wilcoxon et al., 1973) prikazane i objašenjene u tablici 5.1.

U testu znakova ne rade se usporedbe između različitih skupova podataka, dakle nije potrebno zadovoljiti pretpostavku da se podatci mogu/smiju uspoređivati, niti je potrebno da podliježu normalnoj razdiobi. Mana testa znakova jest da neće odbaciti nultu hipotezu ukoliko rezultati jednog algoritma nisu konstatno bolji od drugog. Prema tome, snaga testa znakova je manja od Wilcoxonovog testa.

Prilikom nabrajanja korištenih testova statističke značajnosti u pododjeljku 5.2.5 još je spomenuto i prebrojavanje statistički značajnih pobjeda/nerješenih rezultata/po-

raza, što je ekvivalentno ovom načinu, prethodno filtrirajući rezultate nad kojima je utvrđena statistička značajnost. Prema (Demšar, 2006) ovakav postupak je neispravan, jer pretpostavlja kako je moguće da statistički testovi razlikuju prave od slučajnih razlika. Statistički testovi mjere vjerojatnost dobivenog rezultata pod uvjetom da je nulta hipoteza zadovoljena, što nije ekvivalentno vjerojatnosti nulte hipoteze.

5.2.6. Usporedba više klasifikatora na različitom skupu podataka

Usporedba više od dva klasifikatora podrazumijeva kako više nije moguće provoditi testove zasnovane na uparenom t-testu iz istog razloga zašto nije moguće provoditi uzastopne t-testove – gomilanje pogreške prvog tipa. (Salzberg, 1997) je primjetio i dokazao kako je moguće primjeniti *Bonferonnijevu metodu* (engl. *Bonferroni correction*) i tako omogućiti višestruko testiranje. Bonferronijeva korekcija dozvoljava višestruke usporedbe kontroliranjem razine pogreške. (Bland i Altman, 1995). Primjenom korektivnog faktora Bonferonni korekcija umanjuje p smanjujući mogućnost pogreške tipa I.

Opće statističke metode koje bi se mogle koristiti za usporedbu više klasifikatora na različitim skupovima su ANOVA i Friedmanov test (engl. *Friedman test*).

Analiza varijance

ANOVA je često korištena metoda za testiranje značajnosti razlika između više od dvije srednje vrijednosti. Opisana je u (Fisher, 1956). Nulta hipoteza koju ANOVA pretpostavlja je da nema statistički značajne razlike između vrijednosti koje se uspoređuju. Odbacivanje nulte hipoteze nam govori samo da postoji značajna razlika, a ne gdje se nalazi. U tu svrhu osmišljeni su različiti *post-hoc* testovi, primjerice Tukeyev (engl. *Tukey's test*) ili Dunnetov test (engl. *Dunnett test*)(Wallenstein et al., 1980). Tukeyev test uparuje i uspoređuje sve kombinacije klasifikatora, dok Dunnetov test radi usporedbe svih klasifikatora s jednim referentim. Navedeni *post-hoc* testovi u suštini se ne razlikuju od višestruke primjene t-testa, s time što *post-hoc* testovi kontroliraju mogućnost pogreške tipa I.

Korištenje ANOVA postupka zahtjeva dva uvjeta. ANOVA pretpostavlja kako analizirani uzorci podliježu normalnoj raspodjeli, što često nije slučaj prilikom analize postupaka strojnog učenja. Druga pretpostavka ANOVE je homogenost varijanci (engl. *homogeneity of variance*). Svojstvo homogenosti varijanci je ispunjeno ukoliko sve mjere imaju jednake varijance, što se često ne može primjeniti na rezultate klasifikatora. Primjena *post-hoc* testova se ne preporučuje ukoliko pretpostavke ANOVE

nisu zadovoljene (Zar, 1974).

Friedmanov test

Kao što je Wilcoxonov test znakova neparametarska verzija uparenog t-testa, tako je Friedmanov test (Friedman, 1937) neparametarska verzija višestruke ANOVE. U Friedmanovom testu se rangiraju algoritmi za svaki skup podataka zasebno. U slučaju jednakih vrijednosti, dodjeljuju se prosječni rangovi.

Ako je r_i^j rang j-tog klasifikatora na i-tom skupu podataka (ukupno N). Friedmanov test uspoređuje prosječne rangove klasifikatora:

$$R_j = \frac{1}{N} \sum_i^N r_i^j$$

Nulta hipoteza u Friedmanovom testu tvrdi da nema razlike između srednjih vrijednosti rangova klasifikatora. Iz ranga R_j, stupnjeva slobode $k - 1$ Friedmanova vrijednost računa se:

$$\chi_F^2 = \frac{12N}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right]$$

.

U praksi se češće Friedmanova vrijednost svodi na F-vrijednost koja podliježe F-distribuciji s $k - 1$ i $(k - 1)(N - 1)$ stupnjeva slobode prema formuli:

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1) - \chi_F^2}$$

Friedman je usporedio svoj test s ANOVA-om kroz eksperiment (Friedman, 1940) i zaključio kako se testovi uglavnom slažu oko rezultata.

Kada Friedmanov test opovrgne nultu hipotezu, potrebno je nastaviti s *post-hoc testovima*. Nemenyijev test (engl. *Nemenyi test*) (Nemenyi, 1963) neparametarska je verzija Tukeyevog testa za ANOVA-u, dakle uspoređuje rezultate svih klasifikatora međusobno. Performanse dvaju klasfikatora se značajno razlikuju ukoliko se prosjeci njihovih rangova razlikuju za barem:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}}$$

U slučaju da je potrebno uspoređivati sve klasifikatore s jednim referentnim, moguće je koristiti proceduru koja primjenom Bonferonnijevog korektivnog faktora radi međusobne usporedbe. U ovom slučaju radi se manje usporedbi ($k - 1$), naspram

$k(k-1)/2$ prilikom izvedbe Nemenyjevog testa. Što se radi manje usporedbi, to je test jači, zato se i prilikom usporedbi s referentnim klasifikatorom preferira ovakav način usporedbe. Usporedba i-tog i j-tog klasifikatora se računa:

$$z = \frac{R_i - R_j}{\sqrt{\frac{k(k+1)}{6N}}}.$$

Dobivena z vrijednost se uspoređuje s odgovarajućim vrijednostima normalne distribucije za utvrđivanje značajnosti.

U radu (Demšar, 2006) analizirani su Nemenyijev test s testom koji koristi Bonferonni korekciju. Zaključeno je kako je snaga testa veća ukoliko se uspoređuju svi klasifikatori s jednim referentnim. Međusobna usporedba svih klasifikatora se jedino preporučuje u slučaju kada je potrebno dokazati da su novootkrivene tehnike bolje od svih postojećih.

5.3. Metode procjene p-vrijednosti

U dosadašnjem dijelu odgovaranja na statistička pitanja iz strojnog učenja uvijek je bilo moguće analitički izračunati relevantnu statističku mjeru ili željenu p-vrijednost. U slučajevima kada nije moguće ili nije ispravno zbog razloga opisanih u odjeljku 2.2 kao alternativna opcija nameće se procjena p-vrijednosti. U narednom pododjeljku opisat će se takva tehnika nazvana *bootstrap*.

5.3.1. Bootstrap

Bootstrap tehnikom moguće je procijeniti p-vrijednost (Thompson, 1993). Prema Thompsonu, ispravnije je koristiti bootstrap metodu od analtičkih metoda za dokazivanje statističke značajnosti. Ovdje ćemo opisati primjenu boostrap metode u strojnom učenju. To je samo jedan način provedbe boostrap metode, generalno opisane u (Efron i Tibshirani, 1993).

U boostrap metodi (Berg-Kirkpatrick et al., 2012) uzimaju se uzorci x_i iz populacije i mjere f_A^i i f_B^i: performanse klasifikatora C_A i C_B. $d(x_i) = f_A^i - f_B^i$ je očekivana razlika u performansama između klasifikatora A i B. Zbog ograničenog skupa podataka podaci iz S uzorkuju se sa zamjenom (engl. *sampling with replacement*). Tako dobiveni uzorci nazivaju se *bootstrap* uzorcima.

Za dobivene uzorke x_i, prema nultoj hipotezi, bi trebala vrijediti jednakost:

$$\frac{1}{k}\sum_{i=0}^{k} d(x_i) = d(x) \tag{5.10}$$

gdje je *k* broj uzoraka. Ako se želi provjeriti može li se odbaciti nulta hipoteza, potrebno je provjeriti *koliko često klasifikator C_A daje rezultate koji su bolji od očekivanih*. Očekivani rezultat je da je klasifikator C_A bolji od klasifikatora C_B za $d(x)$. Prema tome, potrebno je prebrojati u koliko je slučajeva *A* bio bolji od C_B za $2d(x)$. Pseudokod bootstrap postupka je prikazan je (Berg-Kirkpatrick et al., 2012) bootstrap postupka prikazan je algoritmom 1. U završnom koraku dolazi se do procjenjene p-vrijednosti na temelju koje se donosi odluka o odbacivanju nulte hipoteze. Najveća prednost bootstrap metode je mogućnost računanja $d(x)$ za bilo koju metriku. Bootstrap metoda temelji se na Bayesovom načinu ispitivanja značajnosti i predstavlja alternativu klasičnom načinu ispitivanja hipoteza.

Algoritam 1 Pseudokod bootstrap postupka

Generiranje M bootstrap uzoraka x_i veličine n nasumičnim izborom s ponavljanjem iz populacije S
$s = 0$
repeat
    if $d(x_i) > 2d(x)$ **then**
        $s = s + 1$
    end if
until $i > n$
$p = \frac{s}{M}$

6. Ispitivanje statističke značajnosti između dva algoritma učenja

O ovom poglavlju predstavit će se praktični dio rada. Napravljena su ispitivanja razlike performansi dvaju algoritama učenja. Cilj praktičnog dijela je pokazati različite rezultate različitih testova.

6.1. Opis eksperimenta

Algoritmi *random forest, RF* (Breiman, 2001) i K-najbližih susjeda (engl. *k nearest neighbours, kNN*) (Fukunaga i Narendra, 1975) korišteni su prilikom konstrukcije klasifikatora za prepoznavanje rukom pisanih znamenki. Skup podataka dobiven je iz *Kaggle*[1] baze podataka[2]. Radi se o stvarnom skupu podataka. Testiranje se odvijalo pomoću programskog jezika *R* u razvojnom okruženju *RStudio*[3]. Zbog efikasnosti algoritama u svim eksperimentima se koristio podskup podataka, točnije 2000 jedinki.

Prema informacijama sa izvorne stranice skupa podataka dobiveni podatci su prikupljeni iz više izvora, ali podatci odabrani u eksperimentu pripadaju su prikupljeni iz istih izvora. S obzirom da se koristi podskup od 2000 jedinki od mogućih 60000, korišteni skup podataka je mali. Dakle, eksperimenti se odvijaju nad malim, homogenim skupom podataka. Razina značajnosti u svim eksperimentima je postavljena na fiksnu vrijednost $\alpha = 0.05$.

Primjeri rezultata provođenja algoritama random forest i K-najbližih susjeda vidljivi su u tablici 6.1. Rezultati su iskazani u preciznosti.

[1] Stranica na kojoj se održavaju natjecanja u izradi klasifikatora.

[2] Više na stranici `http://yann.lecun.com/exdb/mnist/index.html`.

[3] Alat besplatno dostupan na `http://www.rstudio.com/`.

Random Forest	K-najbližih susjeda
0.886	0.832
0.867	0.8425
0.8835	0.8445
0.8805	0.8515
0.871	0.845

Tablica 6.1: Primjeri preciznosti algoritama učenja Random Forest i K-najbližih susjeda na skupu podataka rukom pisanih znamenki

6.2. McNemarov test

Za McNemarov test potrebno je izračunati jedan oblik tablice kontigencije nakon provođenja treniranja i testiranja algoritama. Prema formuli, računa se t_{obs} statistika koja bi, prema nultoj hipotezi, trebala imati χ^2 distribuciju s jednim stupnjem slobode. R kod za McNemarov test prikazan je algoritmom 2.

6.3. Test razlika

Test razlika modeliran je prema binomnoj, odnosno normalnoj distribuciji. Također koristi vrijednosti tablica kontigencije, ali samo za izračun parametara vjerojatnosti pogreške. Implementacija testa razlika prikazana je programskim kodom u algoritmu 3.

6.4. T-test s resempliranjem

T-test s resempliranjem je najčešće korišteni test. Njegova velika razlika u odnosu na prethodne testove je što ne temelji rezultate na jednom rezultatu. Umjesto toga, t-test s resempliranjem nastoji promatrati razlike u performansama nad više uzoraka. Algoritam 4 prikazuje korišteni upareni t-test.

6.5. Upareni t-test s k-strukom unakrsnom validacijom

Prethodno korišteni t-test se nije brinuo za korelaciju između uzoraka, što upareni t-test s k-strukom unakrsnom validacijom nastoji donekle ispraviti podjelom u disjunktne

Algoritam 2 R kod za McNemarov test

```
#odabir skupa za treniranje i testiranje
idx_train <- sample(nrow(train), 1000, replace = FALSE)

xtrain <- train[idx_train, ]
xtest <- train[-(idx_train),]

labels <- as.factor(xtrain[,1])
xtrain <- xtrain[,-1]
correct_labels <- xtest[,1]
xtest <- xtest[,-1]

#treniranje i testiranje algoritama
rf <- randomForest(xtrain, labels, xtest, ntree=50)
knn.results <- (0:9)[knn(xtrain, xtest, labels, k = 10, algorithm="cover_tree")]
rf.predictions <- as.numeric(levels(labels)[rf$test$predicted])

#vrijednosti za tablicu kontigencije
n00 <- sum(rf.predictions != correct_labels & knn.results!= correct_labels )
n01 <- sum(rf.predictions != correct_labels & knn.results== correct_labels )
n10 <- sum(rf.predictions == correct_labels & knn.results!= correct_labels )
n11 <- sum(rf.predictions == correct_labels & knn.results== correct_labels )
n<-n00 + n01 + n10 + n11
contigencies <- matrix(c(n00, n01,   n10, n11), nrow = 2, ncol = 2)

#mcnemarov test
mcnemar.result <- mcnemar.test(contigencies)
odbaci_h0.mcnemar  <- mcnemar.result$p.value < 0.05
```

Algoritam 3 R kod za Test razlika

```
#odabir skupa za treniranje i testiranje
idx_train <- sample(nrow(train), 1000, replace = FALSE)

xtrain <- train[idx_train, ]
xtest <- train[-(idx_train),]

labels <- as.factor(xtrain[,1])
xtrain <- xtrain[,-1]
correct_labels <- xtest[,1]
xtest <- xtest[,-1]

#treniranje i testiranje algoritama
rf <- randomForest(xtrain, labels, xtest, ntree=50)
knn.results <- (0:9)[knn(xtrain, xtest, labels, k = 10, algorithm="cover_tree")]
rf.predictions <- as.numeric(levels(labels)[rf$test$predicted])

#vrijednosti za tablicu kontigencije
n00 <- sum(rf.predictions != correct_labels & knn.results!= correct_labels )
n01 <- sum(rf.predictions != correct_labels & knn.results== correct_labels )
n10 <- sum(rf.predictions == correct_labels & knn.results!= correct_labels )
n11 <- sum(rf.predictions == correct_labels & knn.results== correct_labels )
n<-n00 + n01 + n10 + n11
contigencies <- matrix(c(n00, n01,   n10, n11), nrow = 2, ncol = 2)

#Test razlika
p_A <- (n00 + n01)/n
p_B <- (n00 + n10)/n
p <- (p_A + p_B)/2
se <- sqrt(2*p*(1-p)/n)
z = (p_A-p_B)/se
odbaci_h0.test_razlika <- abs(z)> 1.96
```

Algoritam 4 R kod za t-test s resempliranjem

```
#odabir broja ponavljanja uzorkovanja
ponavljanje <- 30
p_A_arr <- list(rep(0, ponavljanje))
p_B_arr <- list(rep(0, ponavljanje))

for (i in 1:ponavljanje){
  idx_train <- sample(nrow(train), 1000, replace = FALSE)

  xtrain <- train[idx_train, ]
  xtest <- train[-(idx_train),]

  labels <- as.factor(xtrain[,1])
  xtrain <- xtrain[,-1]
  correct_labels <- xtest[,1]
  xtest <- xtest[,-1]

  rf <- randomForest(xtrain, labels, xtest, ntree=50)
  knn.results <- (0:9)[knn(xtrain, xtest, labels, k = 10, algorithm="cover_tree")]
  rf.predictions <- as.numeric(levels(labels)[rf$test$predicted])
  n00 <- sum(rf.predictions != correct_labels & knn.results!= correct_labels )
  n01 <- sum(rf.predictions != correct_labels & knn.results== correct_labels )
  n10 <- sum(rf.predictions == correct_labels & knn.results!= correct_labels )
  n11 <- sum(rf.predictions == correct_labels & knn.results== correct_labels )
  n<-n00 + n01 + n10 + n11
  p_A_arr[i] <- (n00 + n01)/n
  p_B_arr[i] <- (n00 + n10)/n
}

#ispitivanje znacajnosti
paired_t_test_resample <- t.test(unlist(p_A_arr), unlist(p_B_arr), paired=TRUE)

odbaci_h0.upareni_t_test <- abs(paired_t_test_resample$statistic) > 2.05
```

trening i testne skupove. Implementacija testa prikazana je algoritmom 5. Odabran je broj 10 za k, broj preklopnih slojeva (engl. *fold*).

Algoritam 5 R kod za upareni t-test s k-strukom unakrsnom validacijom

```
#odabir parametra k
k <- 10
p_A_arr <- list(rep(0, k))
p_B_arr <- list(rep(0, k))


subset.size <- nrow(train) / k
cross_val.idx <- sample(nrow(train), 2000, replace = FALSE)
i<-1
for (i in 1:k){
  #izgradnja skupova za test i trening
  cross_val.idx_test <- cross_val.idx[c(i:(i+subset.size-1))]
  cross_val.test <- train[cross_val.idx_test,]
  cross_val.train <- train[-cross_val.idx_test,]

  labels <- as.factor(cross_val.train[,1])
  cross_val.train <- cross_val.train[,-1]
  correct_labels <- cross_val.test[,1]
  cross_val.test <- cross_val.test[,-1]

  rf <- randomForest(cross_val.train, labels, cross_val.test, ntree=50)
  knn.results <- (0:9)[knn(cross_val.train, cross_val.test, labels, k = 10, algorithm
      ="cover_tree")]
  rf.predictions <- as.numeric(levels(labels)[rf$test$predicted])

  n00 <- sum(rf.predictions != correct_labels & knn.results!= correct_labels )
  n01 <- sum(rf.predictions != correct_labels & knn.results== correct_labels )
  n10 <- sum(rf.predictions == correct_labels & knn.results!= correct_labels )
  n11 <- sum(rf.predictions == correct_labels & knn.results== correct_labels )
  n<-n00 + n01 + n10 + n11

  p_A_arr[i] <- (n00 + n01)/n
  p_B_arr[i] <- (n00 + n10)/n

}

#ispitivanje znacajnosti
cross_val_t_test<-t.test(unlist(p_A_arr), unlist(p_B_arr), paired=TRUE)
odbaci_h0.cross_val_t_test <- (abs(cross_val_t_test$statistic) > 2.05)
```

6.6. 5xcv upareni t-test

Dodatno smanjenje korelacije između skupova za treniranje i testiranje donosi 5xcv upareni t-test, prikazan algoritmom 6. Ovdje je testiranje provedeno eksplicitno, umjesto

putem funkcije, zbog formata dobivenih podataka. Preskočeni su dijelovi programa gdje se radi treniranje i testiranje nad podatcima, zbog preglednosti. Ti dijelovi napravljeni su na sukladan način kao i u ostalim prikazanim algoritmima.

Algoritam 6 R kod za 5xCV upareni t-test

```
#inicijalizacija listi za spremanje vjerojatnosti
p_A_1_arr <- list(rep(0, 5))
p_B_1_arr <- list(rep(0, 5))

p_A_2_arr <- list(rep(0, 5))
p_B_2_arr <- list(rep(0, 5))

#5 puta uzorkovanje
for (i in 1:5){
    #izrada disjunktnih skupova S_1 i S_2
    cv5x2.idx <- sample(nrow(train), 2000, replace = FALSE)
    n <- length(cv5x2.idx) / 2
    S_1.idx <- cv5x2.idx[c(1:n)]
    S_2.idx <- cv5x2.idx[c((n+1):(2*n))]

    # treniramo s S_1, testiramo s S_2
    # ...
    p_A_1_arr[i] <- (n00 + n01)/n
    p_B_1_arr[i] <- (n00 + n10)/n

    # treniramo s S_2, testiramo s S_1
    # ...
    p_A_2_arr[i] <- (n00 + n01)/n
    p_B_2_arr[i] <- (n00 + n10)/n

  }
  #izracun prema Diettrichovoj formuli
  p_1 <- unlist(p_A_1_arr) - unlist(p_B_1_arr)
  p_2 <- unlist(p_A_2_arr) - unlist(p_B_2_arr)
  p.hat <- (p_1 + p_2) / 2
  s.var <- (p_1 - p.hat)^2 + (p_2 - p.hat)^2
  #ovdje je moguce dodati poboljsanje prema Alpaydinu
  T_obs.5xcv <- p_1[1] / sqrt(sum(s.var)/5)
  odbaci_h0.5xcvtest <- (abs(T_obs.5xcv) > 2.05)
```

6.7. Boostrap

Alternativa klasičnom načinu testiranja prikazana je algoritmom *boostrap* čija implementacija je vidljiva u algoritmu 7. Ovdje smo pretpostavili (iz dosadašnjih ispitivanja) kako je očekivana razlika u preciznosti dva algoritma 0.1, što smo postavili kao apriornu vrijednosti razlike.

Algoritam 7 R kod boostrap algoritam

```
#varijabla kojom se biljezi pojava dvostruke razlike u performansama
S = 0
#broj uzoraka
M = 20
#apriori pretpostavljena razlika u performansama
d_1 = 0.1

  for (i in 1:M){
    idx_train <- sample(nrow(train), 1000, replace = TRUE)

    xtrain <- train[idx_train, ]
    xtest <- train[-(idx_train),]

    labels <- as.factor(xtrain[,1])
    xtrain <- xtrain[,-1]
    correct_labels <- xtest[,1]
    xtest <- xtest[,-1]

    rf <- randomForest(xtrain, labels, xtest, ntree=50)
    knn.results <- (0:9)[knn(xtrain, xtest, labels, k = 10, algorithm="cover_tree")]
    rf.predictions <- as.numeric(levels(labels)[rf$test$predicted])

    rf.result <- sum(rf.predictions == correct_labels) / length(correct_labels)
    knn.result <- sum(knn.results== correct_labels ) / length(correct_labels)

    if (rf.result - knn.result > 2*d_1){
      S <- S + 1
    }
  }
#konacna vrijednost boostrapa
boostrap.p <- S / M
```

6.8. Ispitivanje različitih testova značajnosti

U prethodnim odjeljcima opisani su različiti načini ispitivanja statističke značajnosti. Konačan korak je usporedba rezultata testiranja između svih 6 navedenih metoda. Tablica 6.2 prikazuje matricu podudaranja testiranja različitih metoda na istom skupu podataka. Izvorni skup podataka u svim eksperimentima je bio isti. Matrica pokazuje koliko su korelirani rezultati različitih statističkih testova. Testovi su ponovljeni 50 puta. Broj x u retku i i stupcu j znači da se metode i i j x puta slažu oko zaključka testiranja (maksimalan broj slaganja je 50). Ishod testiranja može biti jedino prihvaćanje ili odbacivanje nulte hipoteze.

	McNemar	Test razlika	T-test (resempl).	T-test k-struka valid.	5xcv upareni t-test	Bootstrap
McNemar		38	41	45	33	45
Test razlika			33	31	31	33
T-test (resempl.)				46	34	50
T-test k-struka valid.					32	46
5xcv upareni t-test						34

Tablica 6.2: Matrica podudaranja rezultata testiranja

McNemar	Test razlika	T-test (resempl).	T-test k-struka valid.	5xcv upareni t-test	Bootstrap
100	66	100	92	68	100

Tablica 6.3: Tablica jačine testova – koliko puta (%) je nulta hipoteza odbačena

Tablicom 6.3 se može vidjeti u kolikoj mjeri je test odbacio nultu hipotezu. Prema tablici vidimo da su McNemarov, T-test s resempliranjem i Bootstrap test najviše puta odbacili nultu hipotezu. Bootstrap test uvelike ovisi o *apriori* vrijednosti te je moguće kako je broj ponavljanja testa bio premali kako bi se dobili kvalitetni rezultati. McNemarov test pokazao se, po dobivenoj jačini, sličnim T-testu, što je suprotno početnom očekivanju, jer McNemarov test uzima samo jedan nasumični uzorak. Preporučeni 5xcv se pokazao relativno slabim u ovom slučaju. Općenito, rezultati pokazuju relativno visoku varijancu, što upućuje da je odabir statističkog testa iznimno bitna stavka prilikom ispitivanja statističke značajnosti rezultata.

7. Zaključak

U seminarskom radu pružen je pregled metoda ispitivanja statističke značajnosti u nekim problemima iz bogatog područja strojnog učenja. Rad ima svrhu pružanja referentnog mjesta za provođenje statističkih provjera istraživačima u području strojnog učenja, ali i demonstrira koliko izbor testa može biti rezultirati različitim rezultatom ispitivanja u istim okolnostima eksperimenta. Prepoznavanje uvjeta eksperimenata je zadaća istraživača, što je preduvjet za ispravan odabir odgovarajućeg statističkog testa.

Daljnje istraživanje vezano uz statističko ispitivanje značajnosti moglo bi dovesti do usuglašavanja standarda za provođenje. Izgradnjom standarda, otvara se i put automatizaciji samog procesa statističkog testiranja. Krajnji proizvod takve automatizacije mogao bi biti softver koji automatski prepoznaje uvjete eksperimenta te provodi ispitivanje statističke značajnosti nad ulaznim podatcima (primjerice, rezultatima algoritma učenja). Takav softver vjerojatno bi trebao biti zasnovan na modelima strojnog učenja upravo iz razloga što nije moguće popisati i predvidjeti sve moguće uvjete viđene u eksperimentima, stoga bi automatski proces morao imati ugrađene mehanizme na temelju kojih može kategorizirati neviđene kombinacije uvjeta eksperimenta, u čemu se strojno učenje pokazalo vrlo dobrim. Ovo je jedan mogući cilj daljnjeg istraživanja statističke značajnosti.

Ispravno korištenje metoda ispitivanja statističke značajnosti pomaže znanstvenoj zajednici pri ispravnom tumačenju statističkih podataka. Zablude u znanstvenim publikacija nerijetko se objave u znanstvenim krugovima, pa čak i dospiju do senzacionalističkih naslova u medijima. Do takvih pogrešaka uglavnom dolazi zbog nedovoljno pažljive validacije rezultata. Bolja, stroža i standardizirana kontrola rezultata eksperimenata trebala bi dovesti do povećane kvalitete znanstvenih radova te uspješnijem i bržem znanstvenom napretku.

8. Literatura

Ethem Alpaydm. Combined 5× 2 cv f test for comparing supervised classification learning algorithms. *Neural computation*, 11(8):1885–1892, 1999.

John Robert Anderson, Ryszard Spencer Michalski, Ryszard Stanisław Michalski, Thomas Michael Mitchell, et al. *Machine learning: An artificial intelligence approach*, svezak 2. Morgan Kaufmann, 1986.

Pierre Baldi et al. *Bioinformatics: the machine learning approach*. The MIT Press, 2001.

Taylor Berg-Kirkpatrick, David Burkett, i Dan Klein. An empirical investigation of statistical significance in nlp. U *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, stranice 995–1005. Association for Computational Linguistics, 2012.

Maximilian Bisani i Hermann Ney. Bootstrap estimates for confidence intervals in asr performance evaluation. U *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, svezak 1, stranice I–409. IEEE, 2004.

Christopher M Bishop i Nasser M Nasrabadi. *Pattern recognition and machine learning*, svezak 1. springer New York, 2006.

J Martin Bland i Douglas G Altman. Multiple significance tests: the bonferroni method. *BMJ: British Medical Journal*, 310(6973):170, 1995.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Nancy Chinchor. The statistical significance of the muc-4 results. U *Proceedings of the 4th conference on Message understanding*, stranice 30–50. Association for Computational Linguistics, 1992.

Michael Collins. Head-driven statistical models for natural language parsing. *Computational linguistics*, 29(4):589–637, 2003.

Reuven Dar, Ronald C Serlin, i Haim Omer. Misuse of statistical tests in three decades of psychotherapy research. *Journal of consulting and clinical psychology*, 62(1):75, 1994.

Cameron Davidson-Pilon. *Probablistic Programming & Bayesian Methods for Hackers*. 2013. URL `http://camdavidsonpilon.github.io/Probabilistic-Programming-and-Bayesian-Methods-for-Hackers/`.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

Thomas G Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923, 1998.

Wilfrid J Dixon i Alexander M Mood. The statistical sign test. *Journal of the American Statistical Association*, 41(236):557–566, 1946.

Bradley Efron i Robert Tibshirani. *An introduction to the bootstrap*, svezak 57. CRC press, 1993.

EJ Eisen. 225. note: The quasi-f test for an unnested fixed factor in an unbalanced hierarchal design with a mixed model. *Biometrics*, stranice 937–942, 1966.

Brian S Everitt. *The analysis of contingency tables*, svezak 45. CRC Press, 1992.

Ronald A Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 222:309–368, 1922.

Ronald A Fisher. Statistical methods and scientific inference. 1956.

Ronald Aylmer Fisher. The design of experiments. 1935.

Milton Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 32(200): 675–701, 1937.

Milton Friedman. A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics*, 11(1):86–92, 1940.

Keinosuke Fukunaga i Patrenahalli M. Narendra. A branch and bound algorithm for computing k-nearest neighbors. *Computers, IEEE Transactions on*, 100(7):750–753, 1975.

Andrew Gelman i Cosma Rohilla Shalizi. Philosophy and the practice of bayesian statistics. *British Journal of Mathematical and Statistical Psychology*, 2012.

Mark A Hall. *Correlation-based feature selection for machine learning*. Doktorska disertacija, The University of Waikato, 1999.

John Michael Hammersley, David Christopher Handscomb, i George Weiss. Monte carlo methods. *Physics today*, 18:55, 1965.

Larry V Hedges, Ingram Olkin, Mathematischer Statistiker, Ingram Olkin, i Ingram Olkin. Statistical methods for meta-analysis, 1985.

G Hinton, R Neal, i R Tibshirani. Delve team members (1995). assessing learning procedures using delve. Technical report, Technical report, University of Toronto, Department of Computer Science.

Darrell Huff. *How to lie with statistics*. WW Norton & Company, 2010.

Philipp Koehn. Statistical significance tests for machine translation evaluation. U *EMNLP*, stranice 388–395, 2004.

Irwin Miller, John E Freund, i Richard Arnold Johnson. *Probability and statistics for engineers*, svezak 4. Prentice-Hall Englewood Cliffs, NJ, 1965.

Peter Nemenyi. *Distribution-free multiple comparisons*. Doktorska disertacija, Princeton University, 1963.

Kishore Papineni, Salim Roukos, Todd Ward, i Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. U *Proceedings of the 40th annual meeting on association for computational linguistics*, stranice 311–318. Association for Computational Linguistics, 2002.

Zdzisław Pawlak. Rough sets, decision algorithms and bayes' theorem. *European Journal of Operational Research*, 136(1):181–189, 2002.

John D Potter. The lady tasting tea: How statistics revolutionized science in the twentieth century. *Nature Medicine*, 7(8):885–886, 2001.

DMW Powers. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

Mary Louise Pratt. *Toward a speech act theory of literary discourse*, svezak 266. Indiana University Press Bloomington, 1977.

Nornadiah Mohd Razali i Yap Bee Wah. Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *Journal of Statistical Modeling and Analytics*, 2(1):21–33, 2011.

Steven L Salzberg. On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and knowledge discovery*, 1(3):317–328, 1997.

George W Snedecor i WG Cochran. Statistical methods 7th edition. *The Iowa State University*, 1980.

Richard C Sprinthall i Stephen T Fisk. *Basic statistical analysis*. Prentice Hall Englewood Cliffs, NJ, 1990.

Bruce Thompson. The use of statistical significance tests in research: Bootstrap and other alternatives. *The Journal of Experimental Educational*, stranice 361–377, 1993.

SYLVAN Wallenstein, Christine L Zucker, i JOSEPH L Fleiss. Some statistical methods useful in circulation research. *Circulation Research*, 47(1):1–9, 1980.

Geoffrey I Webb. Multiboosting: A technique for combining boosting and wagging. *Machine learning*, 40(2):159–196, 2000.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics bulletin*, 1 (6):80–83, 1945.

Frank Wilcoxon, SK Katti, i RA Wilcox. Critical values and probability levels for the wilcoxon rank sum test and the wilcoxon signed rank test. *Selected tables in mathematical statistics*, 1:171–259, 1973.

Alexander Yeh. More accurate tests for the statistical significance of result differences. U *Proceedings of the 18th conference on Computational linguistics-Volume 2*, stranice 947–953. Association for Computational Linguistics, 2000.

Nigel G Yoccoz. Use, overuse, and misuse of significance tests in evolutionary biology and ecology. *Bulletin of the Ecological Society of America*, 72(2):106–111, 1991.

Jerrold H Zar. Multiple comparisons. *Biostatistical analysis*, 1:185–205, 1974.

Ying Zhang, Stephan Vogel, i Alex Waibel. Interpreting bleu/nist scores: How much improvement do we need to have a better system? U *LREC*, 2004.