

hyväksymispäivä arvosana

arvostelija

Koneoppimisen seminaari: Käytännön virherajat

Otso Mäkinen

Helsinki 10.10.2005

HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

Tiedekunta/Osasto — Fakultet/Sektion — Faculty		Laitos — Institution — Department	
Tekijä — Författare — Author Otso Mäkinen			
Työn nimi — Arbetets titel — Title Koneoppimisen seminaari: Käytännön virherajat			
Oppiaine — Läroämne — Subject			
Työn laji — Arbetets art — Level		Aika — Datum — Month and year 10.10.2005	Sivumäärä — Sidoantal — Number of pages 9 sivua + 0 liitesivua
Tiivistelmä — Referat — Abstract			
Avainsanat — Nyckelord — Keywords			
Säilytyspaikka — Förvaringsställe — Where deposited			
Muita tietoja — övriga uppgifter — Additional information			

Sisältö

1 Johdanto	1
2 Luokittelija	1
3 Virherajan määrittely	2
4 Satunnaistetut luokittelijat	3
4.1 Satunnaistetun luokittelijan virheraja	4
5 Puolivalvotut virherajat	4
5.1 Bagging	6
5.2 Cross-Validation	6
5.3 Puolivalvottu testijoukkoa käyttävä raja	7
5.4 Muita menetelmiä	7
6 Tuloksia	7
7 Yhteenveto	8
Lähteet	9

1 Johdanto

Tässä kirjoitelmassa käsitellään luokittelijoiden käytännöllisiä virherajoja. Pohjana on käytetty pääasiassa artikkeleita [Kää05], [Kää] ja [Lan].

Täysin yleisesti ei luokittelijoiden tulevaisuudessa tekemistä virheistä voida päätellä kovinkaan paljoa. Jos kuitenkin voimme tehdä oletuksia alkuperäisestä jakaumasta, jonka tuottamia näytteitä haluamme luokitella, voidaan kysymykseen “Kuinka usein luokittelija c on tulevaisuudessa väärässä?” vastata. Oletamme että data on IID, eli otokset otetaan toisistaan riippumatta samasta jakaumasta. Voidaan osoittaa että tämä on riittävä oletus luokittelijoiden virherajojen määrittelyyn. Koske tutkimme tuntemattomia jakaumia, on virheraja aina määritelty vain tietyn todennäköisyyden puitteissa. Esimerkiksi “Luokittelijan virhe on alle 20.0% todennäköisyydellä 99%”

2 Luokittelija

Tässä luokittelijoilla tarkoitetaan funktioita, jotka jakavat syötejoukon kahteen tulostajoukkoon, esim. $\{+1, -1\}$ tai {“kuuluu luokkaan A”, “ei kuulu luokkaan A”}. Luokittelija saadaan oppimisalgoritmin tuloksena, jonka tavoitteena on löytää funktio joka mallintaa tuntematonta (syöte, tulos) jakaumaa. Jakaumasta oletetaan ainoastaan, että se on IID, eli näytteet otetaan toisistaan riippumatta samasta jakaumasta. Toisin sanoen näytteiden välillä ei ole korrelaatiota. Jakauma ei myöskään välttämättä ole minkään deterministisen funktion määrittämä, vaan voi esimerkiksi sisältää kohinaa. Esitetyissä tuloksissa kaikki luokittelijat ajatellaan mustina laatikoina, eikä niiden varsinaiseen toteutukseen tai oppimisalgoritmiin oteta mitään kantaa. Joitain käytettyjä merkintöjä ([Lan]):

Merkintä	Kuvaus
X	Luokittelijan syöteavaruus
$Y = \{-1, 1\}$	Luokittelun tulosavaruus.
D	Tuntematon jakauma $X \times Y$:ssä
S	Sarja D :stä otettuja riippumattomia näytteitä.
m	$ S $ näytteiden lukumäärä
c	Kuvaus X :ltä Y :lle
x	Syöte $x \in X$
y	Haluttu tulos kun, x ja $y \in X \times Y$

3 Virherajan määrittely

Todellinen virhe

Luokittelijan todellinen virhe määritellään yksinkertaisesti todennäköisyydeksi, että näyte x luokitellaan väärin.

$$e(c, D) = \Pr_{(x,y) \sim D} (c(x) \neq y)$$

Empiirinen virhe

Koska jakauma D on tuntematon, ei todellista virhettä $e(c, D)$ voida suoraan havainnoida. Tarkka todellinen virhe voidaan selvittää vain jos funktio jonka luokittelija yrittää oppia on tarkasti tiedossa. Käytännössä mitataan empiiristä virhettä

$$\hat{e}(c, S) = \frac{1}{|S|} \sum_{(x,y) \in S} I(c(x) \neq y)$$

Empiirinen virhe on väärin luokiteltujen näytteiden ja testattujen näytteiden lukumäärän suhde otoksessa S . Seuraavaksi esitetään kuinka kokeellisesti virheestä voidaan laskea yläraja todelliselle virheelle tietyllä luottamusvälillä.

Yläraja todelliselle virheelle todennäköisyydellä $1 - \delta$

Empiirinen virhe on mitattu testaamalla luokittelijaa m kappaleella nimettyjä näytteitä ts. joiden oikea luokittelu on ennalta tiedossa (labeled sample). Käytettävissä olevat suureet ovat siis havaittujen virheiden määrä k ja käytettyjen näytteiden määrä m . Siis $\hat{e}(c, S) = k/m$. Todetaan, että havaittujen virheiden määrä on binomijakautunut parametrein m ja $e(c, D)$. Koska jakauman keskiarvo $e(c, D)$ on tuntematon, etsimme sen sijaan binomijakauman keskiarvon joka on halutulla todennäköisyydellä yläraja oikealle keskiarvolle. Tämä löydetään etsimällä suurin keskiarvo, jonka kanssa mitattu virhe $\hat{e}(c, S)$ on yhteensopiva.

Määritelmä [Lan]: (Inverse binomial tail) $\overline{\text{Bin}}(k/m, m, \delta)$ on q jolle

$$\sum_{i=0}^k \binom{m}{i} q^i (1-q)^{m-i} = \delta.$$

Todennäköisyys saada k/m :ää pienempi virhe binomijakaumasta, jonka keskiarvo on suurempi kuin $\overline{\text{Bin}}(k/m, m, \delta)$, on pienempi kuin δ . Siten todellisen virheen e on oltava pienempi kuin $\overline{\text{Bin}}(k/m, m, \delta)$ luotettavuudella $1 - \delta$.

Kaikille D, c

$$\Pr_{S \sim D^m} (\overline{\text{Bin}}(\hat{e}(c, S), m, \delta) \geq e(c, D)) \geq 1 - \delta.$$

Eli: todennäköisyys, että virheraja on suurempi kuin todellinen virhe on $1 - \delta$.

Tällä tavalla määritelty virheraja on täydellisen tiukka: jos oikea virhe on riittävän suuri, virheraja rikotaan täsmälleen suhteessa δ . (IID oletuksella)

Riippumattoman virhearvion saamiseksi on helpointa käyttää joukkoa valmiiksi nimettyjä näytteitä, joita ei käytetä oppimiseen. Tällöin virhearvioon käytetyt näytteet eivät pääse vaikuttamaan oppimisen tulokseen ja antavat siten luotettavan arvion todellisesta virheestä. Nimettyjä näytteitä voi olla kuitenkin saatavissa vain rajoitettu määrä, jolloin mahdollisemman moni näyte halutaan käyttää oppimiseen. Tällöin on virheraja laskettava oppimiseen käytetyn datan perusteella ja tarvitaan monimutkaisempia menetelmiä virheenmäärittelyyn, sillä virheen mittauksen ja oppimisen välille syntyy riippuvuus. Perinteisesti oppimisdatasta lasketut luotettavat virherajat ovat olleet liian väljiä ollakseen käyttökelpoisia.

4 Satunnaistetut luokittelijat

Satunnaistetuissa luokittelijoissa lopullinen luokitus määritetään käyttämällä useampaa samalla datalla opetettua luokittelijaa, joiden joukosta yksi valitaan jonkun satunnaisjakauman perusteella.

Tässä esiteltyt satunnaiset luokittelijat toimivat jakamalla opetusdatan k :hon osaan S_i ja opettamalla monta instanssia samasta luokittelijasta näillä eri näytejoukoilla. Kullekin luokittelijalle c_i määritellään virheraja käyttämällä sitä osaa datasta (S'_i), jota ei käytetty kyseisen luokittelijan opettamiseen. Näin koko data saadaan käyttöön sekä opettamiseen että virherajan määrittelyyn. Lopullinen luokittelija valitsee yhden näistä luokittelijoista tasaisesta jakaumasta ja käyttää sitä annetun syötteen luokitteluun. Luokittelija valitaan aina uudestaan jokaiselle syöttelelle. Satunnainen luokittelija antaa siis tilastollisen keskiarvon useammasta luokittelijasta. Myös virhearvio tälle yhdistetylle luokittelijalle voidaan laskea keskiarvona osaluokittelijoiden virheestä. Deterministinen luokittelija, jota käytettiin testijoukkoon perustuvan virheen määrittelyyn, saadaan erikoistapauksena $k = 1$.

4.1 Satunnaistetun luokittelijan virheraja

Satunnaistetun luokittelijan f virheraja $e(f, D)$ saadaan seuraavasti.

Jokaiselle aliluokittelijalle c_i virheraja käyttämällä testidataa S'_i on vähintään todennäköisyydellä $1 - \delta/k$

$$e(c_i, D) \leq \overline{\text{Bin}}(\hat{e}(c_i, S'_i), |S'_i|, \delta/k).$$

Yhdistetyn luokittelijan, joka siis valitsee aina satunnaisesti yhden c_i luokittelua tehdessään, virherajaksi saadaan vähintään todennäköisyydellä $1 - \delta$

$$e(f, D) = \frac{1}{k} \sum_{i=1}^k e(c_i, D) \leq \frac{1}{k} \sum_{i=1}^k \overline{\text{Bin}}(\hat{e}(c_i, S'_i), |S'_i|, \delta/k)$$

eli virherajojen keskiarvo.

5 Puolivalvotut virherajat

Usein parhaat virherajat saadaan juuri satunnaistetuilla luokittelijoilla, mutta niiden ominaisuudet eivät yleensä ole käytännössä toivottuja. Suurimpana haittapuolelta sama syöte x voidaan luokitella eri tavalla kun luokittelijaa käytetään sille monta kertaa. Tämä johtuu yhdistetyn luokittelijan stokastisesta luonteesta.

Satunnaiselle luokittelijalle saatua virherajaa voidaan kuitenkin käyttää hyödyksi laskettaessa virherajaa paremmin käyttäytyvälle deterministiselle luokittelijalle poistamalla satunnaisuus virherajoista.

Merkitsemättömiä näytteitä voidaan käyttää satunnaistetun luokittelijan virherajan turvalliseen muuntamiseen deterministiselle luokittelijalle sopivaksi. Ideana on käyttää merkitsemättömiä näytteitä arvioimaan todennäköisyyttä, että satunnainen luokittelija ja valittu deterministinen luokittelija ovat eri mieltä. Tämä erimielisyys muutetaan sitten luokittelijoiden väliseksi virheeksi, joka voidaan lisätä satunnaisen luokittelijan virherajaan.

Selvennyksenä käytetty deterministinen luokittelija on samaa algoritmia käyttämällä opittettu luokittelija jonka opettamiseen käytetään kaikki nimetyt näytteet.

Menetelmää käytetään seuraavasti:

1. Valitaan oppimisalgoritmi
2. Opetetaan satunnainen luokittelija f , jolle voidaan laskea tiukka virheraja.
3. Opetetaan lopullinen deterministinen luokittelija c_{final} kaikella datalla S .
4. Käytetään nimeämätöntä dataa ja lasketaan luokittelijoiden välinen ns. etäisyys $d(f, c_{\text{final}})$, eli kuinka usein ne ovat eri mieltä.
5. Lopullisen luokittelijan virheraja on satunnaisen luokittelijan virheraja + luokittelijoiden erimielisyyteen perustuva virhe.

(Disagreement probability) luokittelijoiden f ja g erimielisyyden todennäköisyys määritellään suhteena jolla luokittelijat ovat eri mieltä näytteen luokittelusta joukossa D .

$$d(f, g) = d(f, g, D) = \Pr_{(x,y) \sim D} (f(x) \neq g(x))$$

Tätäkin ei voida suoraan mitata, vaan käytetään empiiristä versiota:

$$\hat{d}(f, g, U) = \frac{1}{|U|} \sum_{x \in U} I(f(x) \neq g(x))$$

Jossa U on joukko nimeämättömiä näytteitä. Tärkeää on huomata tämän metrikan yhteys luokittelijan virheen määrittelyyn. Samaa binomijakaumaan perustuvaa virherajan laskentaa voidaan soveltaa myös \hat{d} :n rajojen määrittelyyn.

Luokittelijoiden välinen etäisyys $d(f, c_{\text{final}})$ lasketaan siis käytännössä samalla tavalla kuin satunnaisen luokittelijan virhe verrattuna oikeaan jakaumaan D . Intuitiivisesti siis lasketaan ensin satunnaistetun luokittelijan etäisyys jakaumaan D , sekä deterministisen luokittelijan etäisyys satunnaistettuun luokittelijaan. Etäisyyksien summa on yläraja deterministisen luokittelijan etäisyydelle jakaumaan D (kolmioepäyhtälö).

$$e(c_{\text{final}}, D) \leq e(f, D) + d(f, c_{\text{final}})$$

Ja vähintään todennäköisyydellä $1 - \delta$

$$e(c_{\text{final}}, D) \leq \frac{1}{k} \sum_{i=1}^k \overline{\text{Bin}}(\hat{e}(c_i, S'_i), |S'_i|, \delta/(2k)) \\ + \overline{\text{Bin}}(\hat{d}(f, c_{\text{final}}, U), |U|, \delta/2).$$

Deterministinen luokittelija siis tässä tapauksessa käyttää kaiken datan S oppimiseen, eikä testidataa tarvitse jättää syrjään. Mahdollisuudesta käyttää nimeämättömiä näytteitä voi olla suurta etua, sillä se on yleensä paljon halvempaa tuottaa kuin nimetyt näytteet. Käytännössä tilanteissa joissa tilastollista oppimista käytetään, voi nimeämättömien näytteiden tuottaminen olla ilmaista.

Seuraavaksi esitellään eri menetelmiä satunnaistetun luokittelijan käyttämien joukkojen S_i valitsemiseksi.

5.1 Bagging

Breiman esitteli Bagging-menetelmän ([Bre96]). Siinä kukin S_i tuotetaan valitsemalla satunnaisesti takaisinpanoin n otosta S :stä. Jokaiselle i noin $1 - 1/e \simeq 0.63$ osuus näytteistä valitaan joukkoon S_i loppujen 0.37 jäädessä testauskäyttöön joukkoon S'_i .

Alunperin Bagging-menetelmä esiteltiin äänestävänä luokittelijana, joka vastaa aiemmin esiteltyä satunnaista luokittelua, mutta satunnaisen luokittelijan sijaan lopullinen luokittelu ratkaistaan äänestämällä kaikkien c_i luokittelijoiden kesken. Tässä esiteltyjä menetelmiä voidaan käyttää myös tällaisen äänestävän luokittelijan virherajan laskemiseen.

Äänestävän luokittelijan yhtenä huonona puolena jokaista luokittelua tehdessä on luokittelu tehtävä k kertaa, joka vaatii luonnollisesti enemmän laskentaresursseja. Jokainen luokittelija on myöskin pidettävä yhtä aikaa muistissa.

5.2 Cross-Validation

Cross-Validation -menetelmä toimii seuraavasti.

Näytejoukko S jaetaan k :hon (mahdollisimman) yhtä suureen osaan S'_i . Oppimisalgoritmi ajetaan k kertaa ja kullakin ajolla i oppimiseen käytetään näytteitä S_i , jotka eivät ole joukossa S'_i . Virhearvion tekemiseen käytetään vastaavasti kullakin ajolla joukkoa S'_i . Kun valitaan $k = |S|$ saadaan erikoistapaus, jossa jätetään aina yksi näyte virheen tarkistamiseen. Tätä ei kuitenkaan voi soveltaa tässä esiteltyihin

menetelmiin, sillä yhdellä näytteellä ei saada millään tavalla hyödyllisiä virherajoja yksittäisille luokittelijoille.

Cross-validation -menetelmää käytetään usein käytännössä koko joukolla S opetetun deterministisen luokittelijan virheen approksimointiin. Sen tekemä arvio ei kuitenkaan ole tilastollisesti pätevä ja voi olla hyvinkin kaukana todellisesti virheestä. Esitellyllä virherajan satunnaisuuden poistamisella saadaan kuitenkin luotettavia tuloksia.

Myös Cross-Validation -menetelmästä voi tehdä äänestävän luokittelijan.

5.3 Puolivalvottu testijoukkoa käyttävä raja

Erikoistapauksena valinta $k = 1$ vastaa deterministisen luokittelijan, jonka virheraja on saatu erillistä testijoukkoa käyttämällä, muuntamista puolivalvotuksi opetusjoukkoon perustuvaksi virherajaksi luokittelijalle c_{final} . Satunnaisen luokittelijan sijaan käytetään siis suoraan luvussa 3 esiteltyä virherajan mittausta pohjana lopullisen luokittelijan virherajoille.

Jos testijoukon kooksi $|S'_1|$ valitaan n/k , on saatu virheraja suoraan vertailtavissa Cross-Validation -menetelmän rajaan, sillä niillä on sama odotusarvo luotettavuustermiä lukuun ottamatta.

5.4 Muita menetelmiä

Kääriäinen ja Langford vertailevat myös muita menetelmiä, mukaan lukien online-menetelmät ja PAC-Bayesian virherajoja. Bayesiläisillä menetelmillä on monta kiinnostavaa ominaisuutta, mutta käytännössä PAC-Bayes rajat näyttävät olevan selvästi muita esiteltyjä rajoja huonommat. [Kää]

6 Tuloksia

Kääriäinen ja Langford [Kää05] mittasivat virherajojen käytännön tuloksia laske-
malla esitellyillä menetelmillä virherajat useille tunnetuille vertailujoukoille käyttäen erilaisia yleisessä jaossa olevia oppivia luokittelijoita. Taulukoissa 6 ja 6 on esitetty osa tuloksista, paljon kattavampi taulukko löytyy alkuperäisestä artikkelista. Testeissä nimeämättömät näytteet saatiin unohtamalla luokittelu 10% alkuperäisestä datasta. Kaikissa testeissä on valittu $\delta = 0.01$, eli rajojen luotettavuus on 99%.

Taulukko 1: Tuloksia virherajojen laskemisesta satunnaistetuille virherajoille.

DATA/ALGORITMI	R-TEST	R-BAG	R-CV
<i>AUSTRALIAN/SVMLIGHT</i>	13.04 /31.12	13.77/ 23.29	15.36/32.30
<i>CENSUS – INCOME/c4.5</i>	4.61/ 4.93	4.82/5.03	4.59 /5.01
<i>CRX/c4.5</i>	10.14 /25.40	11.45/ 24.83	10.72/33.48
<i>FOURCLASS/SVMLIGHT</i>	19.54/28.75	15.40 / 22.98	19.77/32.23
<i>SATIMAGE/c4.5</i>	13.20 / 16.50	15.19/17.68	14.13/18.72

Taulukko 2: Tuloksia puolivalvotuille virherajoille.

DATA/ALGORITMI	S-TEST	S-BAG	S-CV
<i>AUSTRALIAN/SVMLIGHT</i>	45.16	38.64	43.27
<i>CENSUS – INCOME/c4.5</i>	5.80	7.12	5.84
<i>CRX/c4.5</i>	36.93	47.03	47.38
<i>FOURCLASS/SVMLIGHT</i>	38.33	41.67	43.46
<i>SATIMAGE/c4.5</i>	32.31	37.95	35.73

Testijoukkoon perustuva virheen laskeminen käytti 10% näytteistä testaukseen ja Bagging- ja Cross-Validation -menetelmät käyttivät arvoa $k = 10$.

R-alkuiset sarakkeet tarkoittavat satunnaistettujen luokittelijoiden virherajoja ja ne sisältävät “mitattu virhe/virheraja” parin. S-alkuiset sarakkeet ovat samoista luokittelijoista muunnettuja deterministisiä luokittelijoita, joiden virheraja on esitetty.

Satunnaisille luokittelijoille parhaat virherajat näytetään saavan Bagging-menetelmällä, mutta se ei kuitenkaan saa parhaita tuloksia mitatusta varsinaisesta virheestä. Tämä on luonnollista, sillä Bagging-menetelmä käyttää noin 37% näytteistä virheen arviointiin, jotka ovat pois oppimisessa käytetyistä näytteistä.

Puolivalvotut oppimisjoukolle lasketut virherajat ovat selvästi satunnaistettuja huonompia, mutta silti Kääriäisen ja Langfordin mukaan vaikuttavia aikaisempiin tuloksiin verrattuna.

7 Yhteenveto

Jos käytössä on ylimääräistä nimettyä dataa, on virherajojen määrittely suoraviivaista ja tuloksena saadaan tiukat tilastolliset rajat. Ongelmana on ollut virherajat jotka saadaan kun kaikki data halutaan käyttää opetukseen. Tämä on erityisen haluttua jos valmiiksi luokiteltua opetusdataa on vain rajoitetusti saatavilla. Kääriäisen ja Langfordin mukaan esitetyt puolivalvotut virherajat alkavat olla jo

käyttökelpoisia verrattuna aikaisempiin menetelmiin. Vaatimuksena on ainoastaan luokittelemattoman datan saatavuus, mikä ei yleensä ole ongelma luokittelijoiden käyttökohteissa.

Parhaan oppimistuloksen saavuttamisen ja opitun tuloksen luotettavuuden välillä on tehtävä kompromissi: parhaan virherajan saamiseksi osa näytteistä on jätettävä pois oppimisesta ja käytettävä virherajan tiukentamiseen. Tämä kuitenkin heikentää luokittelijan todellista luokittelukykyä.

Lähteet

- Bre96 Breiman, L., Bagging predictors. *Machine Learning*, 24,2(1996), sivut 123–140. URL citeseer.ist.psu.edu/breiman96bagging.html.
- Kää Kääriäinen, Matti; Langford, J., A comparison of tight generalization error bounds. *ICML 2005*.
- Kää05 Kääriäinen, M., Generalization error bounds using unlabeled data. *Lecture Notes in Computer Science*, 3559, sivut 127 – 142.
- Lan Langford, J., Tutorial on practical prediction theory for classification.