Satunnaismuuttujien riippumattomuuden testaamisesta

Taneli Mielikäinen

Kevät 2002
Älykkäiden järjestelmien tutkimusseminaari
Tietojenkäsittelytieteen laitos
Helsingin yliopisto

1 Johdanto

Tässä kirjoitelmassa tarkastellaan kahden satunnaismuuttujan riippumattomuuden arvioimista ja siihen liittyvää otosvaativuutta.

Satunnaismuuttujien riippumattomuuden havaitseminen pienestä otoksesta on erittäin tärkeä kysymys data-analyysitehtävissä: Yhteistodennäköisyysjakauman arviointiin riittää olennaisesti pienempi otos, jos arvioitava todennäköisyysjakauma on tulojakauma. Riippumattomuustestejä voitaneen käyttää myös esimerkiksi salaliittojen ja vihollisten paljastamiseen.

Käytännössä tässä kirjoitelmassa rajoitutaan tarkastelemaan (äärellisiä) multinomijakaumia. Merkitään X:llä satunnaismuuttujaa yli joukon $[n]$, Y:llä satunnaismuuttujaa yli joukon $[m]$, ja kirjaimilla P, Q ja R todennäköisyysjakaumia yli joukon $[n] \times [m]$.

Luku 2 käsittelee satunnaismuuttujien riippumattomuuden testaamisen perusteita. Luvussa 3 käsitellään ominaisuuden testaamista. Luvussa 4 esitetään otostehokas ratkaisu satunnaismuuttujien riippumattomuuden testaamiseksi tilastollisen etäisyyden mielessä.

2 Satunnaismuuttujien riippumattomuuden testaaminen

Satunnaismuuttujien riippumattomuuden testaaminen on osa tilastotieteen yhtä keskeisimmistä osa-alueista: tilastollisia testejä. Tilastollisia testejä on lukematon määrä, mutta satunnaismuuttujien riippumattomuuden tapauksessa monet pelkistyvät empiirisen jakauman etäisyyden arviointiin siitä saatavasta tulojakaumasta [7, 8, 9, 10].

Satunnaismuuttujien X ja Y riippumattomuuden testaamisessa pyri-

tään selvittämään onko $P(X = i, Y = j)$ (lähes) sama kuin

$$P(X = i)P(Y = j) = \left(\sum_{j\in[m]} P(X = i, Y = j)\right)\left(\sum_{i\in[n]} P(X = i, Y = j)\right).$$

Ensimmäinen ja vielä nykyisinkin suosituin tapa riippumattomuuden testaamiseksi on χ^2-jakaumaan perustuva χ^2-testi, jonka keksimistä aiemmin tilastollinen testaus oli kokeiden perusteella piirrettyihin kuviin pohjautuvaa arvailua [7].

Satunnaismuuttujien X ja Y riippumattomuutta voidaan arvioida χ^2-testin avulla seuraavasti [7, 10]:

1. Poimitaan riippumattomasti riittävän suuri otos (tuntemattoman) jakauman P mukaisesti jakautuneita pareja $x_1, \ldots x_k \in [n] \times [m]$.

2. Lasketaan otoksen frekvensseistä $\#(i, j)$ todennäköisyysjakauman P empiirinen arvio

$$\hat{P}(X = i, Y = j) = \frac{\#(i, j)}{k} = \frac{\#(i, j)}{\sum_{(i,j)\in[n]\times[m]} \#(i, j)}.$$

3. Lasketaan testisuureen V arvo

$$V = \sum_{(i,j)\in[n]\times[m]} \frac{\left(\hat{P}(X = i, Y = j) - \hat{P}(X = i)\hat{P}(Y = j)\right)^2}{\hat{P}(X = i)\hat{P}(Y = j)},$$

   joka on jakautunut likimäärin kuten χ^2-jakauma $(n - 1)(m - 1)$:llä vapausasteella, jos satunnaismuuttujat X ja Y ovat riippumattomia.

4. Satunnaismuuttujat ovat riippumattomia, jos χ^2-jakauman määräämä merkitsevyystaso $(n-1)(m-1)$:llä vapausasteella pisteessä V on liian pieni.

χ^2-testin suurin heikkous satunnaismuuttujien riippumattomuuden testaamisessa on siinä, että otoksen tulee olla kokoa $\Omega(nm)$ [7].

Jakaumien etäisyysmittoja voidaan motivoida myös informaatioteoreettisesti [4]. Informaatioteorian keskeinen käsite on *entropia (entropy)*

$$H(X) = -\sum_{i \in [n]} P(X = i) \log P(X = i),$$

joka intuitiivisesti kuvaa satunnaismuuttujan X informaatiosisältöä. (Entropia on pienimmillään, kun satunnaismuuttuja on tasaisesti jakautunut.) Ehdollinen entropia

$$H(X|Y) = -\sum_{(i,j) \in [n] \times [m]} P(X = i, Y = j) \log P(X = i|Y = j)$$

kuvaa satunnaismuuttujan X sisältämää informaatiota, kun Y tunnetaan.

Todennäköisyysjakaumien välistä etäisyyttä mitataan usein *suhteellisen entropian (relative entropy)*

$$D(R||Q) = \sum_{(i,j) \in [n] \times [m]} R(X = i, Y = j) \log \frac{R(X = i, Y = j)}{Q(X = i, Y = j)}$$

avulla. (On syytä huomata, että suhteellinen entropia ei ole symmetrinen.) Kahden satunnaismuuttujan riippumattomuutta kuvaa suhteellisen entropian erikoistapaus: *yhteinen informaatio (mutual information)*

$$\begin{aligned}
I(X;Y) &= \sum_{(i,j) \in [n] \times [m]} P(X = i, Y = j) \log \frac{P(X = i, Y = j)}{P(X = i)P(Y = j)} \\
&= \sum_{(i,j) \in [n] \times [m]} P(X = i, Y = j) \log \frac{P(X = i|Y = j)}{P(X = i)} \\
&= -\sum_{(i,j) \in [n] \times [m]} P(X = i, Y = j) \log P(X = i) + \\
&\quad \sum_{(i,j) \in [n] \times [m]} P(X = i, Y = j) \log P(X = i|Y = j) \\
&= -\sum_{i \in [n]} P(X = i) \log P(X = i) + \\
&\quad \sum_{(i,j) \in [n] \times [m]} P(X = i, Y = j) \log P(X = i|Y = j) \\
&= H(X) - H(X|Y) = H(Y) - H(Y|X).
\end{aligned}$$

Yhteinen informaatio saadaan suhteellisesta entropiasta valitsemalla jakaumaksi R yhteistodennäköisyysjakauma ja jakaumaksi Q tulojakauma.

Kolmannen, algoritmisen lähestymistavan jakaumien samankaltaisuuden tarkastelemiseksi tarjoaa *Kolmogorov-kompleksisuus (Kolmogorov complexity)* [8]. Jakauman P Kolmogorov-kompleksisuus $K(P)$ on pienimmän sen tuottavan ohjelman pituus (jossain kiinnitetyssä koodauksessa). Ehdollinen Kolmogorov-kompleksisuus $K(R|Q)$ on pienimmän ohjelman pituus, joka tuottaa jakauman R annettuna jakauma Q. Yhteistä informaatiota vastaava Kolmogorov-kompleksisuudesta nouseva etäisyysmitta olisi siis yhteistodennäköisyysjakauman ehdollinen Kolmogorov-kompleksisuus ehdolla tulojakauma. Kolmogorov-kompleksisuuden eräs suurimmista heikkouksista on se, ettei se ole yleisessä tapauksessa laskettava.

Tilastollisesti ja informaatioteoreettisesti motivoitujen etäisyysmittojen lisäksi luonnollisen ryhmän muodostavat L_p-etäisyydet

$$L_p(x, y) = \left(\sum_{i \in [n]} |x - y|^p \right)^{1/p},$$

missä x ja y ovat avaruuden $\mathbb{R}^n$ pisteitä. Multinomijakaumathan voidaan ajatella L_1-etäisyyden mielessä origosta yhden päässä oleviksi pisteiksi. Todennäköisyysjakaumien kannalta tärkeää L_1-etäisyyttä kutsutaan myös *tilastolliseksi etäisyydeksi (statistical difference)* [9]. Jakaumien samankaltaisuutta voidaan arvioida siis monin tavoin. Jatkossa rajoitumme tarkastelmaan vain tilastollista etäisyyttä.

3 Ominaisuuden testaaminen

Ominaisuuden testaamisessa (property testing) pyritään selvittämään (suurella todennäköisyydellä), onko tarkasteltava olio kaukana (jonkin etäisyysmitan mielessä) kaikista niistä olioista, joilla on testattava ominaisuus [5]. Hieman tarkemmin ominaisuuden testaus on esitetty määritelmässä 1.

Määritelmä 1 ([5]) *Olkoon A algoritmi, joka saa syötteenään kokoparametrin n, virheparametrin $0 < \epsilon < 1$, luotettavuusparametrin $0 < \delta < 1/2$, sekä mahdollisuuden poimia näytteitä jonkin tuntemattoman jakauman mukaisesti. Algoritmi saa jonon $(x_1, f(x_1)), (x_2, f(x_2)), \ldots$ esimerkkejä, missä alkiot $x_i \in \{0,1\}^n$ on valittu (tuntemattoman) jakauman P mukaisesti.*

Funktio f on ϵ-lähellä funktioluokkaa F_n, jos on olemassa $g \in F_n$ niin, että

$$P(x)[f(x) \neq g(x)] \leq \epsilon,$$

missä $[\cdot]$ on indikaattorifunktio. Muuten funktion f sanotaan olevan ϵ-kaukana funktioluokasta F_n.

Algoritmi A on ominaisuuden testaava algoritmi funktioluokalle F_n, jos seuraavat ehdot pätevät:

- *Jos $f \in F_n$, niin A hyväksyy funktion f vähintään todennäköisyydellä $1-\delta$.*
- *Jos f on ϵ-kaukana funktioluokasta F_n, niin A hylkää funktion f vähintään todennäköisyydellä $1 - \delta$.*

Algoritmin A otosvaativuus on algoritmin tarkastelemien esimerkkien lukumäärä syötteellä (n, ϵ, δ).

Tyypillinen muunnelma edellisestä on tilanne, jossa testausalgoritmi saa kysyä mielivaltaisen alkion x arvoa $f(x)$ sen sijaan, että saisi kysyessään vain satunnaisen alkion x ja sen arvon $f(x)$.

Ominaisuuden testaamisella on läheinen yhteys myös älykkäiden järjestelmien teorialle tärkeään PAC-oppimisen malliin [1, 5]. PAC-oppimisalgoritmi on seuraavia poikkeuksia lukuunottamatta samanlainen kuin testausalgoritmi:

- PAC-oppijalle luvataan, että tuntematon funktio f kuuluu luokkaan F_n, ja
- PAC-oppijan pitää löytää funktio h, joka on ϵ-lähellä funktiota f.

PAC-oppiminen on *oikeaa (proper)*, jos $h \in F_n$. Ominaisuuden testaamisen suhde PAC-oppimiseen on seuraavanlainen:

- ei ole vaikeampaa kuin oikea PAC-oppiminen,
- voi olla helpompaa kuin PAC-oppiminen ja
- voi olla vaikeampaa kuin PAC-oppiminen.

Ominaisuuden testaamista on sovellettu laajasti erilaisten kombinatoristen olioiden, esimerkiksi k-väritettävien verkkojen, tunnistamiseen ja ominaisuuden testaamisella onkin ollut keskeinen rooli esimerkiksi approksimointialgoritmien kehityksessä: Approksimoimattomuuden tarkastelussa keskeisen probabilistisesti tarkistettavien todistusten voidaan nähdä nousevan ominaisuuden testaamisesta [5, 11]. Myös approksimointitehokkaat satunnaisalgoritmit ovat saaneet vaikutteita ominaisuuden testaamisesta. Onpa ominaisuuden testaamisen piirissä syntynyt uusi approksimointialgoritmien ryhmäkin: vakioaikaiset approksimointiskeemat [5].

4 Riippumattomuuden testaaminen tehokkaasti

Artikkelissa [2] esitetään menetelmä kahden satunnaismuuttujan riippumattomuuden testaamiseksi $\tilde{O}\left(n^{2/3}m^{1/3}\mathrm{poly}(\epsilon^{-1})\right)$[1] kokoisen otoksen perusteella. Menetelmä hyväksyy riippumattoman jakauman suurella todennäköisyydellä ja hylkää riipumattomasta jakaumasta 7ϵ:in päässä olevan jakauman suurella todennäköisyydellä. Koska menetelmä on varsin tekninen, sen seikkaperäinen käsittely tässä kirjoitelmassa ei ole suotavaa.

[1] $f = \tilde{O}(g)$, jos $f = O(g\mathrm{poly}(\log g))$.

Lyhyesti menetelmä toimii seuraavasti: Menetelmä arvioi jakaumia

$$\begin{aligned} &P(X = i), \quad i \in [n] \qquad \text{ja} \\ &P(Y = j), \quad j \in [m] \end{aligned}$$

pienten otosten perusteella. Arvioiden avulla menetelmä osittaa joukot $[n]$ ja $[m]$ polylogaritmiseen määrään joukkoja, joiden avulla riippumattomuus testataan jakaumien

$$\begin{aligned} &P(X = i), \quad i \in [n] \qquad \text{ja} \\ &P(Y = j), \quad j \in [m] \end{aligned}$$

arvioiden suosittamin menetelmin.

Artikkelissa [2] edellä kuvantun menetelmän osoitetaan olevan asymptoottisesti (logaritmisia tekijöitä lukuunottamatta) optimaalinen.

Yritetään katsoa menetelmää vielä hieman toisesta suunnasta. Olkoon $\alpha = \left(2 + \frac{\log m}{\log n}\right)/3$. Määritellään joukon $[n]$ todennäköisten alkioiden joukot

$$\begin{aligned} S' &= \left\{i \in [n] : P(X = i) \geq n^{-\alpha}\right\} \quad \text{ja} \\ S'' &= \left\{i \in [n] : P(X = i) \geq \frac{1}{2}n^{-\alpha}\right\}. \end{aligned}$$

Menetelmä on pääpiirteittäin seuraavanlainen:

1. Estimoidaan jakaumaa $P(X = i), i \in S''$ $O(n^\alpha \epsilon^{-2} \log n)$ kokoisen otoksen perusteella virheellä $\epsilon/75$ otoksesta estimoidulla empiirisellä jakaumalla, jota merkitään $\tilde{P}$:llä. Olkoon

   $$\tilde{S} = \left\{i \in [n] : \tilde{P}(X = i) \geq \frac{2}{3}n^{-\alpha}\right\}.$$

   Tällöin $\tilde{S} \supset S'$ ja $\tilde{S} \cap \{i \in [n] : P(X = i) \leq \frac{1}{2}n^{-\alpha}\} = \emptyset$.

2. Estimoidaan onko $P(X \in \tilde{S}) > \epsilon/2$ $\tilde{O}(\epsilon^{-1})$ kokoisen otoksen avulla. Jos $P(\in \tilde{S}) > \epsilon/2$, niin estimoidaan jakaumaa

$$P(X = i, Y = i), \tilde{S} \times [m]$$

$\tilde{O}(n^{2/3}m^{1/3}\text{poly}(\epsilon^{-1}))$ kertaa $O(\epsilon^{-1}\log(nm))$ kokoisen otoksen avulla, suoritetaan raskaiden alkioiden riippumattomuustesti ja hylätään jakauma, jos raskaiden alkioiden testi hylkää sen.

3. Estimoidaan onko $P(X \in [n] \setminus \tilde{S}) > \epsilon/2$ $\tilde{O}(\epsilon^{-1})$ kokoisen otoksen avulla. Jos $P(X \in [n] \setminus \tilde{S}) > \epsilon/2$, niin estimoidaan jakaumaa

$$P(X = i, Y = i), \tilde{S} \times [m]$$

$\tilde{O}(n^{2/3}m^{1/3}\text{poly}(\epsilon^{-1}))$ kertaa $O(\epsilon^{-1}\log(nm))$ kokoisen otoksen avulla, suoritetaan keveiden alkioiden riippumattomuustesti ja hylätään jakauma, jos keveiden alkioiden testi hylkää sen.

4. Jos joko $P(X \in \tilde{S}) > \epsilon/2$ tai $P(X \in [n] \setminus \tilde{S}) > \epsilon/2$, niin testataan päteekö

$$\sum_{j \in [m]} \left| P_{\tilde{S}}(Y = j) - P_{[n]\setminus\tilde{S}}(Y = j) \right| \leq \epsilon,$$

missä

$$P_S(Y = j) = \frac{\sum_{i \in S} P(X = i, Y = j)}{\sum_{i \in S, j \in [m]} P(X = i, Y = j)}.$$

Jakauma hylätään P, jos ja etäisyys on edellistä suurempi.

5. Hyväksytään jakauma, jos sitä ei ole vielä hylätty.

Edellä luonnosteltu testi nojautuu aiemmin julkaistuihin jakaumien samankaltaisuustesteihin [3, 6]. Niitä ei kuitenkaan esitellä tässä kirjoitelmassa, sillä jo riippumattomuustestin korkean tason kuvaus ilman perusteluja osoittautui varsin tekniseksi.

5 Loppusanat

Satunnaismuuttujien riippumattomuutta ja jakaumien samankaltaisuutta voidaan arvioida monin tavoin.

Artikkelissa [2] kuvattu riippumattomuustesti on paras tunnettu tapa riippumattomuuden arvioimiseen otosvaativuuden mielessä, mikä on hieman yllättävää, sillä ominaisuuden testaamiseen läheisesti liittyvää PAC-oppimisen mallia pidetty käytännössä käyttökelvottomana. Miksei helpommista näkökulmista ole saavutettu pienempiä otosvaativuuksia? Tämä ei voi johtua kysymyksen vähäpätöisyydestä. Näiden kysymysten pohtiminen edellyttänee vähintään pääosin artikkelien luonnosteleman riippumattomuustestin toiminnan ymmärtämistä [2, 3, 5, 6]. Valitettavasti tämä kirjoitelma ei pysty tarjoamaan tätä ymmärrystä ainakaan riittävissä määrin.

Yhteenvetona voidaan todeta, että ominaisuuden testaaminen näyttää tarjoavan käyttökelpoinen näkökulman myös tilastolliseen testaamiseen.

Viitteet

[1] MARTIN ANTHONY JA NORMAN BIGGS. *Computational Learning Theory*. Cabridge University Press, 1992.

[2] TUĞKAN BATU, ELDAR FISCHER, LANCE FORTNOW, RAVI KUMAR, RONITT RUBINFELD JA PATRICK WHITE. Testing random variables for independence and identity. *Annual Symposium on Foundations of Computer Science* 42:442–451, 2001.

[3] TUĞKAN BATU, LANCE FORTNOW, RONITT RUBINFELD, WARREN D. SMITH JA PATRICK WHITE. Testing that distributions are close. *Annual Symposium on Foundations of Computer Science* 41:259–269, 2000.

[4] THOMAS COVER JA JOY A. THOMAS. *Elements of Information Theory*. Wiley Interscience, 1991.

[5] ODED GOLDREICH, SHAFI GOLDWASSER JA DANA RON. Property testing and its connection to learning and Approximation. *Journal of the ACM* 45:653–750, 1998.

[6] ODED GOLDREICH JA DANA RON. On testing expansion in bounded-degree graphs. *Electronic Colloquium on Computational Complexity* TR00-20, 2000.

[7] DONALD E. KNUTH. *The Art of Computer Programming: Seminumerical Algorithms*. Addison Wesley, 1998.

[8] MING LI JA PAUL VITÁNYI. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer Verlag, 1997.

[9] AMIT SAHAI JA SALIL VADHAN. A complete problem for statistical zero knowledge. *Electronic Colloquium on Computational Complexity* TR00-84, 2000.

[10] JUN SHAO. *Mathematical Statistics*. Springer Verlag, 1999.

[11] MARIUS ZIMAND. Probabilistically checkable proofs the easy way. *Electronic Colloquium on Computational Complexity* TR01-27, 2001.