

Satunnaisalgoritmien vaativuusteoriaa

Formalisoidaan hieman täsmällisemmin, millaisia suoritustakuita satunnaisalgoritmeilta voidaan vaatia.

Yksinkertaisuuden vuoksi tarkastellaan päätösongelmia. Useimmat ideat yleistyvät helposti.

Sopiva laskennan malli on esim. Turingin kone, jossa jokaisella (tila, syötemerkki)-parilla on määritelty **kaksi** siirtymäfunktion arvoa ja näistä valitaan aina jompikumpi symmetristä kolikkoa heittämällä. Koneen aikavaativuus annetulla syötteellä on pisimmän mahdollisen laskennan pituus. Yksityiskohdat eivät kuitenkaan ole tässä tärkeitä.

Kun M on tällainen **probabilistinen Turingin kone**, olkoon $P(M, x)$ todennäköisyys että syötteellä x kone M päättyy hyväksyvään tilaan. Koneen M hyväksymä kieli on nyt

$$L(M) = \{x \mid P(M, x) > 1/2\}.$$

Kieli A kuuluu luokkaan PP (Probabilistic Polynomial time), jos $A = L(M)$ jollain polynomisessa ajassa toimivalla M .

Luokka PP ei ole kovin realistinen satunnaisalgoritmin malli. Seuraava algoritmi osoittaa, että $SAT \in PP$.

```
ProbSAT( $f(x_1, \dots, x_n)$ ):  
   $b := \text{random}(\{0, 1\})$   
  if  $b = 0$   
(1) then accept  
(2) else  
    for  $i := 1$  to  $n$  do  $b_i := \text{random}(\{0, 1\})$   
    if  $f(b_1, \dots, b_n) = 1$  then accept  
    else reject
```

Jos $f(x_1, \dots, x_n)$ ei ole toteutuva, niin todennäköisyydellä $1/2$ hyväksytään haarassa (1), muuten aina hylätään.

Jos $f(x_1, \dots, x_n)$ on toteutuva, niin lisäksi haarassa (2) hyväksytään ainakin todennäköisyydellä $1/2^n$, joten hyväksymistodennäköisyys on yli $1/2$.

Siis ProbSAT [tai sen esitys probabilistisena Turingin koneena] hyväksyy kielen SAT. Selvästi ProbSAT toimii polynomisessa ajassa.

Samalla idealla nähdään helposti, että itse asiassa

$$NP \cup \text{co-NP} \subseteq PP.$$

Toisaalta polynomisessa *tilassa* on mahdollista kokeilla kaikkia mahdollisia satunnaisarvausten tuloksia, joten

$$PP \subseteq PSPACE.$$

Tarkempien suhteiden selvittäminen on avoin ongelma. (Tietysti jos esim. $P = PSPACE$ niin satunnaisuudesta ylipäänsä ei tällä tarkastelutasolla ole mitään etua.)

Huomattavasti käytännönläheisempi luokka on BPP (Bounded error Probabilistic Polynomial time). Kone M tunnistaa kielen A virhetodennäköisyydellä ε jos

$$P(M, x) \begin{cases} \leq \varepsilon & \text{jos } x \notin A \\ \geq 1 - \varepsilon & \text{jos } x \in A \end{cases}$$

missä $\varepsilon < 1/2$ on syötteestä riippumaton vakio.

Virheparametrin ε tarkka arvo ei itse asiassa ole kovin tärkeä, kuten seuraavaksi näemme.

Oletetaan, että algoritmi R tunnistaa kielen A virhetodennäköisyydellä $\varepsilon < 1/2$. Tarkastellaan seuraavaa algoritmia, missä m on toistaiseksi määräämätön parametri.

```
 $S(x, m)$ :  
   $p := 0$   
   $q := 0$   
  for  $i := 1$  to  $m$  do  
    if  $R(x)$  hyväksyy then  $p := p + 1$   
    else  $q := q + 1$   
  if  $p > q$  then accept  
  else reject
```

Oletetaan, että m on pariton, $m = 2k + 1$.

Jos S antaa väärän vastauksen, niin R on antanut oikean vastauksen korkeintaan k kertaa m yrityksellä. Tämän todennäköisyys on

$$\sum_{j=0}^k \binom{m}{j} (1 - \alpha)^j \alpha^{m-j}$$

missä $\alpha \leq \varepsilon$ on algoritmin R virhetodennäköisyys syötteellä x .

Koska $\alpha < 1/2$, niin $1 - \alpha > \alpha$, joten

$$\begin{aligned} \sum_{j=0}^k \binom{m}{j} (1 - \alpha)^j \alpha^{m-j} &\leq \sum_{j=0}^k \binom{m}{j} (1 - \alpha)^{m/2} \alpha^{m/2} \\ &\leq 2^m (1 - \alpha)^{m/2} \alpha^{m/2} \\ &= (4\alpha(1 - \alpha))^{m/2} \end{aligned}$$

sillä binomikaavan mukaan

$$\sum_{j=0}^k \binom{m}{j} \leq \sum_{j=0}^m \binom{m}{j} = (1 + 1)^m.$$

Koska $\alpha(1 - \alpha)$ saavuttaa maksiminsa kun $\alpha = 1/2$, pätee $\alpha(1 - \alpha) \leq \varepsilon(1 - \varepsilon) < 1/4$. Olkoon nyt

$$c = (-\log(4\alpha(1 - \alpha)))^{-1}$$

jolloin $c > 0$ ja $(4\alpha(1 - \alpha))^c = 1/2$. Algoritmin S virhetodennäköisyys on korkeintaan

$$(4\alpha(1 - \alpha))^{m/2} \leq \left(\frac{1}{2}\right)^{m/(2c)}$$

eli esim. virhetodennäköisyyden $(1/2)^n$ saavuttamiseen riittää $2cn$ iteraatiota. Virhetodennäköisyys pienenee eksponentiaalisesti.

Probabilistinen Turingin kone tunnistaa kielen A toispuolisella virhetodennäköisyydellä ε jos

$$P(M, x) \begin{cases} \geq 1 - \varepsilon & \text{jos } x \in A \\ = 0 & \text{jos } x \notin A. \end{cases}$$

Monilla käytännössä tärkeillä satunnaisalgoritmeilla on toispuolinen virhe, koska ne ovat tyyppiä

1. Arvaa todiste y .
2. Jos y on validi todiste sille, että $x \in A$, niin hyväksy. Muuten hylkää.

missä $x \in A$ jos ja vain jos tälle seikalle on olemassa jokin todiste y (vrt. luokka NP).

Luokka R (merk. toisinaan RP, Random Polynomial time) koostuu niistä kielistä, jotka voidaan tunnistaa polynomisessa ajassa toispuolisella virheellä $\varepsilon < 1/2$. Kuten luokan BPP tapauksessa, parametrin ε tarkka arvo ei ole oleellinen.

Lisäksi määritellään luokka ZPP (Zero error Probabilistic Polynomial time) koostumaan kielistä, jotka voidaan tunnistaa polynomisessa ajassa satunnaisalgoritmilla, joka ei koskaan vastaa väärin mutta voi vastata "en tiedä" todennäköisyydellä $\varepsilon < 1$.

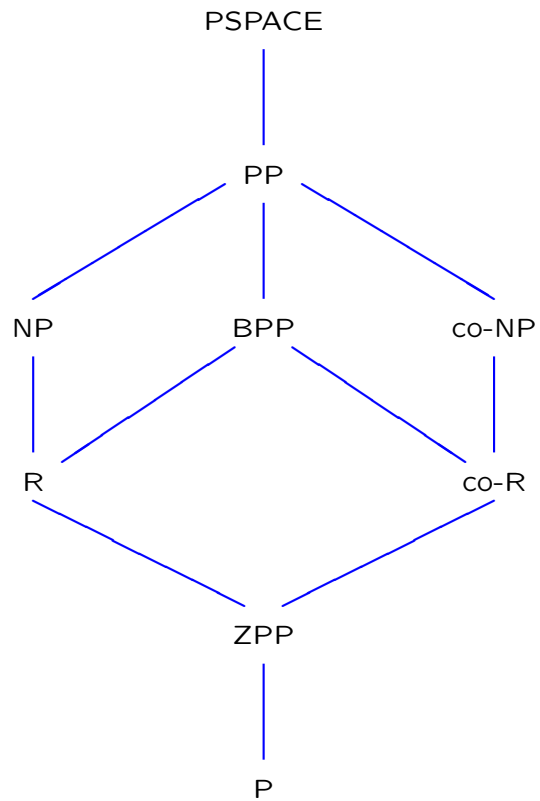
On helppo nähdä, että

$$ZPP = R \cap \text{co-R}.$$

Myös luokkaan ZPP kuuluvilla ongelmilla virhetodennäköisyyttä voidaan iteroimalla pienentää eksponentiaalisesti.

Samoin nähdään, että $A \in ZPP$, jos ja vain jos A voidaan tunnistaa algoritmilla, joka ei koskaan tee virheitä eikä vastaa "en tiedä" ja toimii *odotusarvoisesti* polynomisessa ajassa.

Siis ZPP vastaa Las Vegas -algoritmeja ja BPP Monte Carlo -algoritmeja.



Probabilististen vaativuusluokkien sisältyvyyksiä. Sisältyvyyden aitous kaikissa tapauksissa avoin ongelma.

7.3 Otantaan perustuvia algoritmeja

Ensimmäisenä esimerkkinä tarkastellaan polynomien identtisuuden tarkistamista.

Perusongelmana on määrätä kahdesta n muuttujan polynomifunktiosta r ja s ovatko ne identtiset, ts. päteekö $r(a_1, \dots, a_n) = s(a_1, \dots, a_n)$ kaikilla a_1, \dots, a_n . Tarkastelemme yksinkertaisuuden vuoksi ongelmaa, onko polynomifunktio p identtisesti nolla, ts. onko $p(a_1, \dots, a_n) = 0$ kaikilla a_1, \dots, a_n . Alkuperäinen ongelma palautuu tähän valinnalla $p = r - s$.

Tietysti jos polynomit on annettu eksplisiittisesti, ongelma voidaan triviaalisti ratkaista katsomalla ovatko polynomien kaikki kertoimet samat.

On kuitenkin olemassa polynomifunktioita, joilla kertoimien muodostaminen on turhan työlästä. Esim. funktio

$$p(x_1, \dots, x_n) = \prod_{i=1}^n (1 + x_i)$$

on helppo evaluoida millä tahansa muuttujien arvoilla, mutta esityksessä

$$p(x_1, \dots, x_n) = \sum_{S \subseteq \{1, \dots, n\}} \prod_{i \in S} x_i$$

on eksponentiaalinen määrä termejä.

Selvytyksen vuoksi käytetään merkintää $p(\mathbf{x}) \equiv 0$ tarkoittamaan $p(x_1, \dots, x_n) = 0$ kaikilla x_1, \dots, x_n , ja merkintää $p(\mathbf{x}) \not\equiv 0$ tämän negaatiolle. Polynomien p astetta merkitään $\deg p$.

Lause (Zippel ja Schwartz 1979) Olkoon p astetta d oleva reaalikertoiminen n muuttujan polynomi jolla $p(\mathbf{x}) \not\equiv 0$, ja olkoon $S \subseteq \mathbf{R}$ äärellinen. Nyt polynomilla p on joukossa S^n korkeintaan $d|S|^{n-1}$ nollakohtaa.

Todistus Induktio arvojen n ja d suhteen.

$n = 1$: Tunnettu perustulos.

$d = 1$: Olkoon $p(\mathbf{x}) = a_1x_1 + \dots + a_nx_n - b$ missä $a_j \neq 0$. Jos $p(\mathbf{x}) = 0$ niin

$$x_j = \frac{1}{a_j} \left(b - \sum_{i \neq j} a_i x_i \right),$$

joten $n - 1$ muuttujaa määrää yksikäsitteisesti jäljellejäävän. Siis joukossa S^n voi olla korkeintaan $|S|^{n-1}$ nollakohtaa.

$n > 1, d > 1$: Olkoon p astetta d ja $p(\mathbf{x}) \neq 0$.

Tapaus A: p on jaollinen. Siis $p(\mathbf{x}) = q(\mathbf{x})r(\mathbf{x})$ kaikilla \mathbf{x} , missä $\deg q > 1$ ja $\deg r > 1$. Kun merkitään

$$N_p(S) = \{ \mathbf{x} \in S^n \mid p(\mathbf{x}) = 0 \},$$

saadaan induktio-oletuksen nojalla

$N_q(S) \leq (\deg q)|S|^{n-1}$ ja $N_r(S) \leq (\deg r)|S|^{n-1}$. Siis

$$\begin{aligned} N_p(S) &= |N_q(S) \cup N_r(S)| \\ &\leq |N_q(S)| + |N_r(S)| \\ &\leq (\deg q + \deg r)|S|^{n-1} \\ &= d|S|^{n-1}. \end{aligned}$$

Tapaus B: p jaoton. Kullakin $s \in S$ tarkastellaan $n - 1$ muuttujan polynomia $p_s(\mathbf{x}) = p(x_1, \dots, x_{n-1}, s)$. Jos $p_s(\mathbf{x}) \not\equiv 0$ kaikilla $s \in S$, niin

$$|N_p(S)| = |\cup_{s \in S} N_{p_s}(S)| \leq |S| \cdot d \cdot |S|^{n-2} = d|S|^{n-1}.$$

Olkoon toisaalta $p_s(\mathbf{x}) \equiv 0$ jollain $s \in S$. Ajatellaan p muuttujan x_n polynomiksi, jossa kertoimet ovat muuttujien x_1, \dots, x_{n-1} polynomeja. Jakoyhtälön nojalla

$$p(\mathbf{x}, x_n) = (x_n - s)q(\mathbf{x}, x_n) + r(\mathbf{x})$$

kaikilla $\mathbf{x} \in S^{n-1}$. Kun $x_n = s$, seuraa oletuksesta $p_s(\mathbf{x}) \equiv 0$ nyt $r(\mathbf{x}) \equiv 0$ eli

$$p(\mathbf{x}, x_n) = (x_n - s)q(\mathbf{x}, x_n)$$

vastoin oletusta polynomin p jaottomuudesta. \square

Korollaari Olkoon $p(x) \not\equiv 0$ ja $\deg p = d$. Jos luvut a_1, \dots, a_n valitaan toisistaan riippumatta tasaisen jakauman mukaan joukosta $\{-d, \dots, d\}$ niin

$$P(p(\mathbf{a}) = 0) < \frac{1}{2}.$$

Todistus Valitaan edellisessä lauseessa $S = \{-d, \dots, d\}$. Siis $|S| = 2d + 1$ ja

$$P(p(\mathbf{a}) = 0) \leq \frac{d(2d + 1)^{n-1}}{(2d + 1)^n} = \frac{d}{2d + 1} < \frac{1}{2}.$$

□

Yllä johdettua todennäköisyysrajaa voidaan tiukentaa iteroimalla kuten edellä on todettu.

Jos $p(x) \equiv 0$ niin koskaan ei löydy yhtään $\mathbf{a} \in S^n$ jolla $p(\mathbf{a}) \neq 0$.

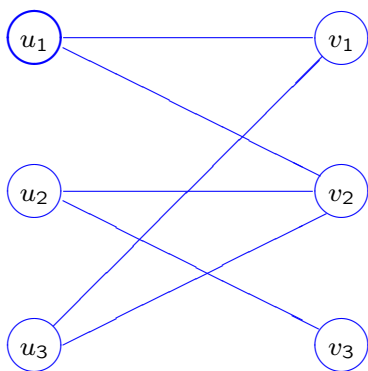
Jos taas $p(x) \not\equiv 0$, niin otamalla m otosta joukosta S^n lyötyy todennäköisyydellä $1 - (1/2)^m$ ainakin yksi \mathbf{a} jolla $p(\mathbf{a}) = 0$.

Sovellus: pariutustesti.

Olkoon $G = (U \cup V, E)$ kaksijakoinen verkko,
 $U = \{u_1, \dots, u_n\}$ ja $V = \{v_1, \dots, v_n\}$. Muodostetaan
 $n \times n$ -matriisi $X_G = (\tilde{x}_{ij})$, missä

$$\tilde{x}_{ij} = \begin{cases} x_{ij} & \text{jos } (u_i, v_j) \in E \\ 0 & \text{muuten} \end{cases}$$

ja x_{ij} ovat muuttujasymboleita.



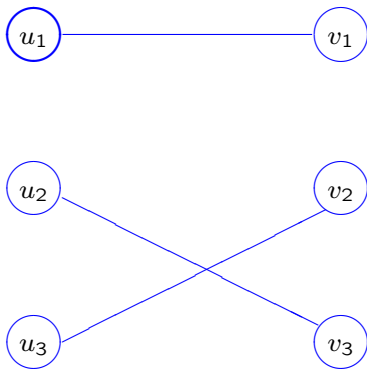
Verkko G

$$\begin{pmatrix} x_{11} & x_{12} & 0 \\ 0 & x_{22} & x_{23} \\ x_{31} & x_{32} & 0 \end{pmatrix}$$

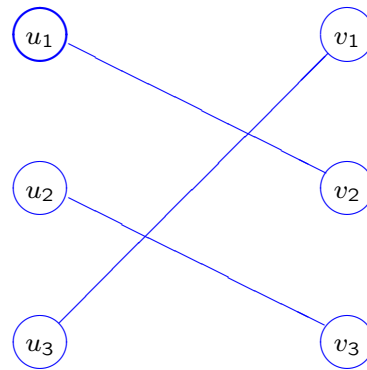
Matriisi X_G

Matriisin X_G determinantti on nyt muuttujien x_{ij} korkeintaan astetta n oleva polynomi. Jokaiseen termiin on valittu tekijäksi tasan yksi muuttuja kullakin riviltä ja kullakin sarakkeelta, joten kukin termi vastaa yhtä täydellistä pariutusta.

$$\det X_G = -x_{11}x_{32}x_{23} + x_{12}x_{23}x_{31}$$



$$-x_{11}x_{32}x_{23}$$



$$x_{12}x_{23}x_{31}$$

Siis täydellinen pariutus on olemassa, jos ja vain jos polynomi $\det X_G$ ei ole identtisesti nolla. Muistetaan, että vaikka determinantissa on $n!$ termiä, se voidaan evaluoida ajassa $O(n^3)$ kun matriisiin sijoitetaan jotkin vakioarvot. Kysymys täydellisen pariutuksen olemassaolosta ratkeaa siis edellisellä otantamenetelmällä.

Menetelmä yleistyy myös ei-kaksijakoisille verkoille (Tutte 1947).

Binomijakauma ja Chernoffin rajat

Kun S on onnistumisten lukumäärä suoritettaessa n riippumatonta toistokoetta, joiden kunkin onnistumistodennäköisyys on p , merkitään $S \sim \text{Bin}(n, p)$. Binomijakautuneen satunnaisuuttujan pistetodennäköisyydet, odotusarvo ja varianssi ovat tunnetusti

$$P(S = k) = \binom{n}{k} p^k q^{n-k}$$

$$E[S] = np$$

$$\text{Var}[S] = npq$$

missä on merkitty $1 - p = q$.

Me olemme erityisesti kiinnostuneita binomijakauman häntätodennäköisyyksistä

$$P(S \geq (1 + \lambda)np) = \sum_{k=(1+\lambda)np}^n \binom{n}{k} p^k q^{n-k}$$

$$P(S \leq (1 - \lambda)np) = \sum_{k=0}^{(1-\lambda)np} \binom{n}{k} p^k q^{n-k}$$

jotka esittävät todennäköisyyttä, että saadaan hyvin epätyypillinen otos.

Jos parametrien arvot on annettu, häntätodennäköisyyksiä voi yrittää laskea suoraan em. kaavoista.

Jos n on hyvin suuri, voidaan käyttää hyväksi keskeistä raja-arvolausetta, jonka nojalla

$$\frac{S - np}{\sqrt{npq}} \sim N(0, 1)$$

eli sopivasti skaalattuna S noudattaa standardinormaalijakaumaa.

Asymptoottisessa algoritmianalysissa on usein kätevintä käyttää ns. Chernoffin rajoja

$$\begin{aligned} P(S > (1 + \lambda)np) &\leq \exp(-\frac{1}{3}\lambda^2 np) \\ P(S < (1 - \lambda)np) &\leq \exp(-\frac{1}{2}\lambda^2 np) \end{aligned}$$

kun $0 \leq \lambda \leq 1$.

Chernoffin rajat eivät kuitenkaan välttämättä ole erityisen tarkkoja, jos halutaan parhaita mahdollisia vakioita; kehittyneempiäkin menetelmiä tunnetaan.

Perusesimerkki: Kuinka monta toistoa tarvitaan, että onnistumistodennäköisyys p saadaan arvioiduksi suhteellisella virheellä ε luotettavuudella δ ?

Tietysti S/n on sopiva estimaatti parametrille p . Siis mikä on oltava n jotta

$$P(|S/n - p| > \varepsilon p) \leq \delta.$$

Chernoffin rajoista saadaan

$$\begin{aligned} P(|S/n - p| > \varepsilon p) &= P(S > (1 + \varepsilon)np) \\ &\quad + P(S < (1 - \varepsilon)np) \\ &\leq 2 \exp\left(-\frac{1}{3}\varepsilon^2 np\right) \\ &\leq \delta \end{aligned}$$

kun valitaan

$$n \geq \frac{3}{\varepsilon^2 p} \ln \frac{2}{\delta}.$$

Siis kiinteällä p riittää $O(\varepsilon^{-2})$ otosta.

Otoskoko kuitenkin kasvaa, kun p lähestyy nollaa, eli pienten todennäköisyyksien arviointi hyvällä *suhteellisella* virheellä on vaikeaa.

Yhdisteen koon arviointi Suurella perusjoukolla X on kokoelma osajoukkoja S_i , $i = 1, \dots, k$, joista kaikilla i

- tiedetään alkioiden lukumäärä $|S_i|$,
- osataan ratkaista päteekö $x \in S_i$ annetulla $x \in S$ ja
- osataan tuottaa joukon S_i alkioita satunnaisesti tasaisella jakaumalla.

Tehtävänä on määrittää joukkojen S_i yhdisteen $S = S_1 \cup \dots \cup S_k$ koko $|S|$. Koska voi olla $|S| \ll |X|$, ongelman ratkaiseminen suoraan edellisen esimerkin avulla ei välttämättä toimi vaikka osattaisiinkin generoida satunnaisia $x \in X$.

Merkitään

$$\begin{aligned} R &= \{ (i, x) \mid x \in S_i \} \\ R' &= \{ (i, x) \mid x \in S_i \text{ ja } x \notin S_j \text{ kun } j < i. \} \end{aligned}$$

Siis $|R| = \sum_i |S_i|$ ja $|R'| = |\cup_i S_i|$. Jos tunnettaisiin $p = |R'|/|R|$, saataisiin

$$|S| = p \sum_i |S_i|.$$

Suhteen p estimoimiseksi halutaan tasaisesti jakautuneita alkioita $(i, x) \in R$. Näitä saadaan seuraavasti:

1. Valitse $i \in \{1, \dots, k\}$ todennäköisyyksin

$$P(i = r) = \frac{|S_r|}{\sum_j |S_j|}.$$

2. Valitse satunnainen $x \in S_i$.

Pääohjelma on siis seuraava:

```

count := 0
for  $n$  times do
    valitse satunnainen  $(i, x) \in R$ 
    if  $x \notin S_j$  kaikilla  $j < i$  then count := count + 1
return  $(\text{count}/n) \cdot \sum_i |S_i|$ 

```

Edellisen esimerkin perusteella estimaatti $\hat{p} = (\text{count}/n)$ on ε -tarkka luotettavuudella δ , jos

$$n \geq \frac{3}{\varepsilon^2 p} \ln \frac{2}{\delta}.$$

Koska $p \geq 1/k$, tämä pätee ainakin jos

$$n \geq \frac{3k}{\varepsilon^2} \ln \frac{2}{\delta}.$$