2. Tétel

Valószínűség fogalma és kiszámításának kombinatorikus módszerei (permutációk, variációk, kombinációk). Feltételes valószínűség, függetlenség, Bayes-formula. Algoritmusok lépésszáma: beszúrásos rendezés, összefésüléses rendezés, keresések lineáris és logaritmikus lépésszámmal. Gyorsrendezés, az összehasonlítások minimális száma. Rendezés lineáris lépésszámmal: radix rendezés, vödör rendezés.

Valószínűség fogalma

Minél többször elvégzünk egy kísérletet, egy esemény relatív gyakorisága annál jobban megközelít egy számot. Ez a szám az esemény valószínűsége. Az A esemény valószínűségét P(A)-val jelöljük.

Kombinatorikus módszer

Faktoriális: $n! = 1 \cdot 2 \cdots n. \quad 0! = 1.$

Szemifaktoriális: $(2n+1)!! = 1 \cdot 3 \cdots (2n+1).$

Binomiális együttható: $\binom{n}{k} = \frac{n!}{k!(n-k)!}. \quad \binom{n}{0} = 1.$

Permutáció: n különböző elem összes lehetséges sorrendjének a száma: n!.

Ismétléses permutáció: n elem összes lehetséges sorrendjének a száma, ha n1,n2,...,nr egyező van közöttük: $\frac{n!}{n_1! \cdot n_2! \cdots n_r!}.$

Variáció: n különböző elemből K darab lehetséges kiválasztásainak a száma, ha nincs visszatevés és a sorrend számít: $\frac{n!}{(n-k)!}.$

Ismétléses variáció: n különböző elemből k darab lehetséges kiválasztásainak a száma, ha van visszatevés és a sorrend számít: $n^k.$

Kombináció: n különböző elemből k darab lehetséges kiválasztásainak a száma, ha nincs visszatevés és a sorrend nem számít: $\binom{n}{k}.$

Ismétléses kombináció: n különböző elemből k darab lehetséges kiválasztásainak a száma, ha van visszatevés és a sorrend nem számít: $\binom{n+k-1}{k}.$

Binomiális tétel:

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^{n-k} b^k = \binom{n}{0} a^n b^0 + \binom{n}{1} a^{n-1} b^1 + \cdots + \binom{n}{n} a^0 b^n .$$

Pascal háromszög képzési szabálya:

$$\binom{n+1}{k+1} = \binom{n}{k+1} + \binom{n}{k}.$$

A feltételes valószínuség fogalma

Tegyük fel, hogy az A esemény valószínuségére vagyunk kíváncsiak, de ismeretes számunkra, hogy a B esemény bekövetkezett. A valószínuség bevezetésekor használt relatív gyakoriságos megközelítést alkalmazzuk most is. Ismételjük meg a kísérletünket n-szer, de csak azokat a végrehajtásokat vegyük figyelembe, amelyekben B bekövetkezett. Ezen részsorozatban az A relatív gyakorisága

$$\frac{k_{AB}}{k_B} = \left(\frac{k_{AB}}{n}\right) : \left(\frac{k_B}{n}\right).$$

Ez utóbbi pedig P(AB) / P(B) körül ingadozik. Így ezt érdemes elfogadni a feltételes valószínűségnek.

Legyen A és B esemény, P(B) > 0. Ekkor az A esemény B-re vonatkozó feltételes valószínűségén a

$$\boxed{P(A \mid B) = P(AB) / P(B)}$$

mennyiséget értjük.

1.3.2. Example. (a) A feltételes valószínűség végeredményben az egész eseménytér egy részére leszűkített valószínűség. Ez leginkább a részsokaságból történő mintavétellel szemléltethető. Tekintsünk egy 10000 fős populációt, ebben 5050 nő és 4950 férfi van. A nők között 100, a férfiak között 900 180 cm-nél magasabb található. Ha véletlenszerűen kiválasztunk egy embert a populációból, akkor annak a valószínűsége, hogy az 180 cm-nél magasabb (a klasszikus képlet alapján) $1000/10000 = 0,1$. Ha a nők közül választunk ki egyet, akkor ugyanez a valószínűség $100/5050 = 0,0198$. A feltételes valószínűség $P(A|B) = P(AB)/P(B)$ képletével számolva:

$$\frac{100}{10000} : \frac{5050}{10000} = 100/5050,$$

tehát a két felfogás azonos eredményre vezet. Klasszikus valószínűségi mező esetén a kétféle számolás mindig csak az „összes esetek számával” történő bővítésben (egyszerűsítésben) különbözik egymástól.

(b) Egy szelvénnyel lottózunk. A lottóhúzást figyeljük; az első négy kihúzott szám szerepel a szelvényünkön. Most következik az ötödik húzás. Mennyi a valószínűsége, hogy ötösünk lesz?

Jelölje A azt az eseményt, hogy ötösünk lesz, B azt, hogy az első 4 kihúzott számot eltaláltuk.

$$P(A|B) = \frac{P(AB)}{P(B)} = \frac{1}{\binom{90}{5}} : \frac{\binom{5}{4}}{\binom{90}{4}} = \frac{1}{86}.$$

Ugyanerre az eredményre jutnánk akkor is, ha úgy okoskodnánk, hogy mivel négyet már eltaláltuk, a maradék 86-ból kell egyet eltalálnunk. Ez utóbbi esélye $1/86$.

Általában is igaz, hogy a feltételes valószínűséget úgy is ki lehet számítani, hogy az eseményteret „leszűkítjük” a feltételben szereplő B eseményre. Ennek hátterét világítja meg a következő állítás.

1.3.3. Theorem. *Legyen $(\Omega, \mathcal{F}, P)$ valószínűségi mező, B egy rögzített esemény, $P(B) > 0$. Jelölje $\mathcal{F}_B$ az AB alakú halmazokat, ahol $A \in \mathcal{F}$. Legyen $P_B(C) = P(C|B)$ minden $C \in \mathcal{F}_B$-re. Ekkor $(B, \mathcal{F}_B, P_B)$ valószínűségi mező.*

Függetlenség: Legyen ξ és η együttes eloszlása a (2.1.3)-ban megadott. ξ és η függetlensége a következőt jelenti: az, hogy ξ felvesz valamilyen x értéket, nem befolyásolja annak az esélyét, hogy η valamely y értéket vegyen fel.

2.1.11. DEFINITION. Azt mondjuk, hogy ξ és η *független*, ha

$$P(\xi = x_i, \eta = y_j) = P(\xi = x_i)P(\eta = y_j), \quad i, j = 1, 2, \ldots.$$

A $\xi_1, \xi_2, \ldots, \xi_n$ valószínűségi változókat *páronként függetleneknek* nevezzük, ha közülük bármely kettő független.

A $\xi_1, \xi_2, \ldots, \xi_n$ valószínűségi változókat *(teljesen) függetleneknek* nevezzük, ha (2.1.4)

$$P(\xi_1 = x_{k_1}, \xi_2 = x_{k_2}, \ldots, \xi_n = x_{k_n}) = P(\xi_1 = x_{k_1})P(\xi_2 = x_{k_2}) \cdots P(\xi_n = x_{k_n})$$

teljesül minden $x_{k_1}, \ldots, x_{k_n}$-re a valószínűségi változók értékészletéből.

2.1.12. EXERCISE. Legyenek $\xi_1, \ldots, \xi_n$ függetlenek. Lássuk be, hogy ekkor $\xi_1, \ldots, \xi_n$ bármely részrendszere is független!

2.1.13. DEFINITION. Valószínűségi változók egy tetszőleges rendszerét *függetlennek* nevezünk, ha bármely véges részrendszere független.

2.1.14. NOTE. A definícióból adódik, hogy valószínűségi változók tetszőleges $\{\xi_i, i \in I\}$ családja akkor és csak akkor független, ha az általuk generált teljes eseményrendszerek $\{\mathcal{E}(\xi_i), i \in I\}$ családja független.

2.1.15. THEOREM. *Ha $\xi_1, \ldots, \xi_n$ független diszkrét valószínűségi változók és $g_1, \ldots, g_n$ valós függvények, akkor az $\eta_1 \models g_1(\xi_1), \ldots, \eta_n = g_n(\xi_n)$ valószínűségi változók is függetlenek.*

Bayes tétel: Ha egy „kétfázisú” kísérletben a második fázis eredményeiből akarunk visszakövetkeztetni az első fázis eredményére, akkor a Bayes-tétel hasznos segédeszköz.

Legyen A és B két, pozitív valószínuségű esemény. A feltételes valószínűség definíciójából

$$\boxed{P(B|A) = P(A|B)P(B)/P(A)}$$

Ez a Bayes-formula.

1.3.9. THEOREM. *Legyen A egy esemény, $B_1, B_2, \ldots$ teljes eseményrendszer, $P(A) > 0, P(B_i) > 0, i = 1, 2 \ldots.$ Ekkor*

$$\boxed{P(B_i|A) = \frac{P(A|B_i)P(B_i)}{\sum_{j=1}^{\infty} P(A|B_j)P(B_j)}}$$

minden j-re.

Algoritmusok lépésszáma: beszúrásos rendezés, összefésüléses rendezés, keresések lineáris és logaritmikus lépésszámmal. Gyorsrendezés, az összehasonlítások minimális száma. Rendezés lineáris lépésszámmal: radix rendezés, vödör rendezés.

Beszúrásos rendezés

A beszúró rendezés sok esetben a leggyorsabb négyzetes rendezés. Működése hasonló ahhoz, mint amikor lapjainkat rendezzük egy kártyajáték során. A rendezés fő lépése az, hogy az asztalon lévő rendezetlen saját pakliból elvesszük a felső lapot és beszúrjuk a kezünkben tartott rendezett lapok közé. Kezdetben a rendezett rész az első felvett lapból áll, majd n-1 beszúrás után lapjainkat már rendezett módon tartjuk a kezünkben.

A beszúró rendezés még nem említett előnye az, hogy nem csak tömbben tárolt elemek rendezésére alkalmas, hanem a láncolt listákra is könnyen alkalmazható.

Adatstruktúra	tömb
Legrosszabb idő bonyolultság	$O(n^2)$
Legjobb idő bonyolultság	$\Omega(n), O(1)$ csere
Átlagos idő bonyolultság	$O(n^2)$ összehasonlítás, csere
Legrosszabb tár bonyolultság	$O(n)$ teljes, $O(1)$ kiegészítő

Összefésüléses rendezés

Az összefésülő rendezés alapötlete, hogy eleve rendezett elemsorozatok aránylag egyszerűen vonhatók össze rendezett sorozattá. Az algoritmus lényege, hogy az elemeket két csoportba osztjuk, a csoportokat rendezzük (akár összefésülő rendezéssel), majd a kapott részeket összefésüljük.

Az összefésülés lényege, hogy a bemeneti sorozatok aktuális elemét nézve a kimeneti sorozatba mindig a kisebb kerül. Így egy összefésülés futásideje a kimeneti sorozat hosszával arányos.

Lineáris keresés

A legegyszerűbb keresési algoritmus, amely rendezetlen tömbön dolgozik. A tömb első elemétől kezdve összehasonlítjuk a tömbelemeket a keresendő elemmel. A keresési ciklus akkor ér véget, ha valamelyik tömbelem megegyezik a keresettel, vagy, ha a tömb végére érünk. Az utóbbi esetben a keresett elem nincs a tömbben. Az algoritmus onnan kapta nevét, hogy a keresések száma, és így a futási idő, lineáris függvénye a tömb elemszámának.

Logaritmikus keresés

A logaritmikus vagy bináris keresési algoritmus rendezett tömbön működik, így az előző módszernél lényegesen gyorsabb keresést tesz lehetővé. A keresendő elemet először a tömb középső eleméhez hasonlítjuk. Ha a két elem egyezik, megtaláltuk a tömbelemet, ha nem, a keresést abban a tömbfélben folytatjuk, amelyet a középső és a keresett elem viszonya kijelöl. Ha a tömb növekvő sorrendbe rendezett és a keresendő elem nagyobb, mint a középső elem, akkor a második tömbrészben, egyébként pedig az elsőben folytatjuk a keresést. A keresést akkor fejezzük be, ha megtaláltuk a keresett tömbelemet, vagy a tömbrész elfogyott. Az összehasonlítások száma, s így a futási idő, az elemszám kettesalapú logaritmusával arányos, ami nagy elemszámok esetén lényegesen kisebb lehet, mint a lineáris keresés esetén.

Gyorsrendezés

- Oszd meg és uralkodj elven alapszik, rekurzív elv:
    - osszuk két részre a rendezendő sorozatot úgy, hogy az egyik rész minden eleme kisebb legyen a másik rész összes eleménél
    - a két részre külön-külön ismételjük meg az előbbi lépést, míg mindkét rész 0 vagy 1 elemű lesz
- A feldolgozási művelet egy szétválogatás, melynek során egy megadott értékhez viszonyítva a tőle kisebb elemeket elé, a nagyobbakat mögé helyezzük.
- Viszonyítási értéknek a gyakorlatban a sorozat középső vagy első elemét választjuk ki.

Adatstruktúra	tömb
Legrosszabb idő bonyolultság	$O(n^2)$
Legjobb idő bonyolultság	$O(n \log n)$
Átlagos idő bonyolultság	$\Theta(n \log n)$
Legrosszabb tár bonyolultság	$O(n)$

Radix rendezés

Másnéven számjegyes rendezés. Ha a kulcsok összetettek, több komponensekből állnak, akkor rendezhetünk az egyes komponensek szerint. Például a dátum az év, hónap, nap rendezett hármasából áll. Ha a kulcsok utolsó komponense szerint rendezünk, majd az eredményt az utolsó előtti komponens szerint, és így tovább, akkor végül rendezett sorozathoz jutunk.

Az egész számok tekinthetőek bitsorozatoknak, s a rendezés minden egyes fordulójában két sorozattá bontjuk a kiinduló, illetve az eredményül kapott sorozatokat, aszerint, hogy a vizsgált bit 0 illetve 1. E két lista egymás után fűzéséből kapható meg a soron következő sorozat. (Természetesen nemcsak kettes, hanem bármilyen más számrendszerbeli számokként is tekinthetnénk a sorozat elemeit, s például négyes számrendszer esetén négy sorozattá bontatánk a sorozatot.) Kettes számrendszer használata esetén, ha a számok 0 és $2^k - 1$ közé esnek, akkor a bonyolultság O(nk).

Vödör rendezés

Az edényrendezés – szokták vödör rendezésnek is hívni – várható futási ideje lineáris, amennyiben a bemenet egyenletes eloszlásból származik.

Éppúgy, mint a leszámláló rendezés, az edényrendezés is azért lehet gyors, mert feltesz valamit a bemenetről.

- Míg a leszámláló rendezés azt feltételezi, hogy a bemenet olyan egészekből áll, amelyek egy kis intervallumba tartoznak,
- addig az edényrendezés azt, hogy a bemenetet egy olyan véletlen folyamat generálja, amelyik egyenletesen osztja el az elemeket a [0, 1) intervallumon.