Statisztikai minta és becslések, átlag és szórás. Konfidenciaintervallumok. Az u-próba.

A statisztikai minta fogalma

Definíció: Valamely valószínűségi változóra vonatkozó véges számú független kísérlet vagy megfigyelés eredménye: véges sok azonos eloszlású valószínűségi változó.

Jelölés: Tekintsük a valószínűségi változót, ekkor a X-re vonatkozó n elemű minta

$X_1, X_2, ..., X_n$

Az n számú kísérlet elvégzése során a i mintaelem egy-egy konkrét számértéket vesz fel:

$X_1 = x_1, X_2 = x_2, ..., X_n = x_n$

(elméleti vs tényleges)

A statisztikai minta reprezentatív: a mintaelemek eloszlása megegyezik a vizsgált valószínűségi változó eloszlásával, hiszen mindegyik kísérletnél magát a valószínűségi változót figyeljük meg.

A statisztikai minta elemei független valószínűségi változók, mivel a kísérleteket egymástól függetlenül végezzük.

A mintaelemekből tapasztalati jellemzőket, ún. **statisztikát** konstruálunk.

A statisztika a mintaelemek valamely függvénye. A statisztika tehát maga is valószínűségi változó és eloszlásának meghatározása fontos feladat.

A várható érték az eloszlás súlypontjáról, a szórás a változó értékeinek szétszórtságáról ad felvilágosítást. Ezekre az elméleti jellemzőkre a mintaelemekből igyekszünk következtetni úgy, hogy az $X_1, X_2, ..., X_n$ mintából különböző függvényeket képezünk. Valamely függvény minden konkrét minta esetén egyetlen számadatba tömöríti a mintaelemekben rejlő információt.

Mintaközép (átlag)

$$\bar{\xi} = \frac{\xi_1 + \xi_2 + \ldots + \xi_n}{n}$$

Tétel:

Ha a valószínűségi változó várható értéke μ, szórása σ, akkor a mintaközépre

$$E(\bar{\xi}) = \mu \qquad D(\bar{\xi}) = \frac{\sigma}{\sqrt{n}}$$

Rendezett minta:

A véletlen, az észlelés sorrendjében kapott mintaelemeket rendezzük nagyság szerint. Jelölje a nagyság szerint a legkisebbet ξ_1^*, a megmaradók közül a legkisebbet ξ_2^*, stb.

Ekkor

$$\xi_1^* \le \xi_2^* \le \ldots \le \xi_n^*$$

A rendezett mintaelemek már nem függetlenek és nem is azonos eloszlásúak.

Mintaterjedelem:

$$R = \xi_n^* - \xi_1^*$$

Medián:

Ha a mintanagyság páratlan, akkor a középső mintaelem a medián - páros mintanagyság esetén a két középső átlaga.

Tapasztalati (empirikus) szórásnégyzet:

A mintaközéptől vett eltérések négyzetének átlaga:

$$s_n^2 = \frac{\left(\xi_1 - \bar{\xi}\right)^2 + \left(\xi_2 - \bar{\xi}\right)^2 + \ldots + \left(\xi_n - \bar{\xi}\right)^2}{n}$$

Korrigált tapasztalati szórásnégyzet:

$$s_n^{*2} = \frac{\sum_{i=1}^{n}\left(\xi_i - \bar{\xi}\right)^2}{n-1} = \frac{n}{n-1} S_n^2$$

Variációs tényező (relatív szórás):

$$V = \frac{s_n}{\bar{\xi}}$$

A statisztikai becslések jellemzői

Torzítatlan becslés:

Ha a valószínűségi változó elméleti jellemzője az a paraméter, és az $\alpha_n(\xi_1, \xi_2, \ldots \xi_n)$ statisztikai mintából kívánjuk becsülni, akkor elvárjuk, hogy az α_n statisztika értékei az 'a' szám körül ingadozzanak.

$$E(\alpha_n(\xi_1, \xi_2, \ldots \xi_n)) = a$$

Konzisztens becslés:

A minta elemszámának növelésével az α_n statisztika egyre jobban közelítse meg az 'a' paramétert.

Elégséges becslés:

Az α_n statisztika tartalmazza az 'a' paraméterre vonatkozó összes információt.

Efficiens becslés:
A legkisebb szorasu becsles

Konfidenciaintervallum:

Olyan intervallum, amely a paramétert lefedi bizonyos valószínűséggel. Az intervallum végpontjai függnek a véletlentől, a paraméter pedig egy ismeretlen konstans.

Az ismeretlen m várható értékre adunk meg egy intervallumot, amely végpontjai a mintaelemek függvényei lesznek és adott (1-Ɛ) valószínűséggel lefedi az ismeretlen valószínűséget.

Ez -ra szimmetrikus, hiszen az „T1" alsó és „T2" felső érték az alábbi módon számítható ki a várható értékre normális eloszlás és ismert szórás esetén:

$$: T_1 \overset{def}{=} \bar{\xi}_n - \frac{u_\varepsilon \sigma_0}{\sqrt{n}}, T_2 \overset{def}{=} \bar{\xi}_n + \frac{u_\varepsilon \sigma_0}{\sqrt{n}}$$

Itt az u értékét táblázatból kereshetjük ki, konstans, a szórás és a minták száma ismert. (kszi görög betű és x ugyanazt jelölik, forrástól függenek)

Ekkor annak a valószínűsége, hogy az ismeretlen az [a, b] intervallumba esik, 1 - Ɛ.

Pl: Ɛ=0.05

u_ε értékét keressük táblázatból: $\Phi(u_\varepsilon)=1-(\varepsilon/2) = 0.975$, tehát $u_\varepsilon=1.96$

Tehát a mintából ismert átlagot az elolszlásból ismert szórást és a mintavételek számát kell behelyettesíteni a képletbe:

T1 = átlag-1.96*szórás/√mintavételek száma T2 = átlag+1.96*szórás/√mintavételek száma

Statisztikai próbák: Legyen teta egy ismeretlen paraméter, és

H0: teta=teta0 nullhipotézis. Itt teta0 egy adott szám (előírás, szabvány, érték stb., amely kedvező számunkra, az a jó, ha ezt elérjük). Teta nem ismert. H0 egyszerű hipotézis, és 1 teta0 érték van.

H1: teta!= teta0 ellenhipotézis, vagy alternatív hipotézis, ez általában számunkra kedvezőtlen eset. Összetett hipotézis, mert több érték jön számításba:

H1: teta != teta0 kétoldali

teta < teta0 egyoldali (pl: vásárló esetén a csoki súlya <100g rossz)

teta > teta0 egyoldali (pl: eladó esetén a kínált áru súlya > 10kg rossz)

Döntéshozás:

	H0 igaz	H0 hamis
H0 elfogadva	helyes döntés	másodfajú hiba
H0 elvetve	elsőfajú hiba	helyes döntés

Mindkét hibát minimalizálni kellene, de nem lehetséges. Általában ha az egyik hibát csökkentem, a másik nőni fog. Ezért rögzítünk egy alfa>0 kicsi számot (pl 0.01)

Pl: H0: jó az autó, amit meg akarok venni

1.ránézek, azt hiszem, hogy jó és nem veszem meg: 1.fajú hiba

2. rossz az autó, és mégis megveszem: 2.fajú hiba

Rögzített elsőfajú hiba esetén keressük azt a próbát/tesztet, amely a 2.fajú hibát minimalizálja. Ezt a rögzített hibát szokás szabványba foglalni.

U-próba:

A legegyszerűbb próba az u-próba. Legyen $X_1,...,X_n$ minta N(0,1) eloszlásból. Tegyük fel, hogy σ^2 ismert. Az m várható értékre az előírás m_0. Tehát a $H_0: m=m_0$ nullhipotézist kell vizsgálnunk a $H_1: m \neq m_0$ altenatív hipotézissel (ellenhipotézissel) szemben.H_0 fennállása esetén az u statisztika normális eloszlású lesz:

$$u = \frac{\overline{X} - m_0}{\sigma}\sqrt{n}$$

Tehát ha H_0 igaz, akkor u nagy valószínűséggel beleesik egy $[-u_{\alpha/2}, u_{\alpha/2}]$ intervallumba. Ha ez nem áll, akkor az H_1 teljesülésére utal.

Tehát a döntési eljárás a következő. Adott α értékhez meghatározzuk azt az $u_{\alpha/2}$ értéket, melyre

$P(-u_{\alpha/2} \leq N(0,1) \leq u_{\alpha/2}) = \alpha$.

A az elsőfajú hiba nagysága. Ha $u \in [-u_{\alpha/2}, u_{\alpha/2}]$, tehát beleesik az elfogadási tartományba, akkor H_0-at $1-\alpha$ szinten (azaz$(1-\alpha)\cdot 100\%$ szignifikancia szinten) elfogadom. Ha nem esik ebbe a tartományba, akkor a kritikus tartományba esik, tehát H_0-at $1-\alpha$ szinten (azaz$(1-\alpha)\cdot 100\%$ szignifikancia szinten) elvetem. Az α értékét 0.1, 0.05, 0.01-nek szoktuk választani.

Pl: 100g csokit veszek. Tényleg 100g a súlya?

H0: m= 100g (itt m ismeretlen, 100g a feltételezésünk, mivel ez van ráírva)

H1: m != 100g

X: csoki súlya, EX(várható érték) elvileg 100g

Megfigyelések: x1 = X1(omega)=99g, x2=101g, …, x16 = 98g

m0 = 100g, n=16, átlag = 99.5, szigma = 1

u=(99.5-100)/(1/4) = 4*(-0.5) = -2

ha alfa = 0.1, akkor táblázatból 0.95 (=1-(alfa/2))-höz tartozó értéket kell kikeresni, ami 1.64. Tehát elfogadási tartomány: [-1.64, 1.64]. Mivel az u-statisztikánk értéke -2, ez kritikus tartományba esik, tehát H0-t 95% szignifikancia szinten elvetem.

Atolvasni a masik tetel szamtech reszet!!!!!!!!!!!!!!!!!!!

Az informatikai biztonság fogalma, legfontosabb biztonsági célok. Fizikai, emberi, technikai fenyegetések és ellenük való védekezés. Algoritmus védelem eszközei: titkosítás, digitális aláírás, hash függvények. Az AES és RSA algoritmusok.

Az informatikai biztonság fogalma

Az informatikai biztonság az informatikai rendszer olyan – az érintett számára kielégítő mértékű – állapota, amelyben annak védelme az informatikai rendszerben kezelt adatok bizalmassága, sértetlensége és rendelkezésre állása, valamint a rendszer elemeinek sértetlensége és rendelkezésre állása szempontjából zárt, teljes körű, folytonos és a kockázatokkal arányos.

Legfontosabb biztonsági célok

Biztonsági kérdések

- Mik azok az eszközök, illetve erőforrások, amiket meg akarunk védeni?
- Milyen veszélyek fenyegetik az adott erőforrásokat?
- Milyen hatásfokkal kezeli ezeket a kockázatokat a választott biztonsági megoldás?
- A választott megoldás milyen új biztonsági réseket okoz?
- Megéri-e alkalmazni a megoldást?

Fizikai, emberi, technikai fenyegetések és ellenük való védekezés

Fizikai veszélyforrások és az ellenük való védekezés

Kockázati tényezők

- A lehetséges veszélyforrásokat számba kell venni a védekezés miatt
- Konkrét esetben nem mindegyik jelentkezik, s a súlyuk is különböző lehet
- Az informatikai rendszerek üzemeltetőinek:
    - fel kell mérni a kockázati tényezőket
    - elemezni kell azok hatását
    - meg kell tervezniük az ellenük való védelmet
- Változások, gyors fejlődés miatt rendszeresen ismételni kell a kockázatelemzést

Fizikai veszélyforrások

- Víz, tűz, sugárzás, elemi csapás
- Lopás
- Rongálás

- elektromos árammal működnek (tűzveszélyes) -> általános tűz- és vagyonvédelmi szabályok alkalmazása (ne tartsunk a számítógép közelében gyúlékony anyagokat)
- hőtermelés miatt hűtésre van szükség, a nagy teljesítményű szervereket klimatizált helységben kell elhelyezni
- Váratlan áramkimaradás okozta károkat csökkenteni kell – például szünetmentes áramforrás használatával. Ezekben akkumulátorok vannak, ha áramkimaradás van, innen kapnak áramot a berendezések (nagyobb költség: saját áramfejlesztő generátor használata)
- Az elhelyezése az informatikai berendezéseknek nagyon fontos
    - nagyobb épület központjában, mely által nincs a külvilággal érintkező fala. A belépéseket szabályozni, naplózni kell.
    - nedves környezettől távol, azaz ne legyen a környékén vizes helyiség, mosdó, konyha, vízvezeték
- A mágneses térre és az elektromágneses sugárzásra érzékenyek a berendezések, ezektől is óvni kell
    - ez két irányú hatás, ugyanis az erős mágneses tér vagy sugárzás károsíthatja a tárolt adatokat, s mivel a berendezések maguk is sugárforrások, ezek mérésével esetleges támadás is indítható. A tárolt adatok főképp a mágnesszalagokon és a mágneslemezeken eshettek áldozatul, a mai CD-k, DVD-k már nem mágneses elven tárolnak adatokat. Ezeken az adatok azonban nagy sűrűségben helyezkednek el, amely miatt a kozmikus sugárzás által néhány bit változhat, ezek ellen hibajavító kódokkal lehet védekezni.
- a processzorok, katódsugárcsöves monitorok elektromágneses sugárzását (mely az egészségre ártalmatlan) mérve, illetéktelen megfigyeléssel néhány méterről megismerhetővé válnak a munkafolyamatok
- különösen érzékeny adatok kezelése esetén (pl. banki rendszerek, hitelesítő szervezetek) biztosítani kell, az elektromágneses sugárzást leárnyékoló berendezéseket
- smart kártya is bocsát ki sugárzást, amely által a tulajdonos privát kulcsa is kompromittálódhat

Biztonsági intézkedések

- Fizikai védelem
    - elhelyezés
    - energiaellátás

    - kábelezés elhelyezése
- Sugárzás elleni védelem
- Hardverhiba – ma már nem gyakori
- Szoftver biztonsága – a felhasználók csak arra használhassák, amire jogosultságuk van. Nem kívánt szolgáltatások, weboldalak szűrése

Emberi veszélyforrások és az ellenük való védekezés

Emberi veszélyforrások, social engineering

- Tapasztalatlanság
- Adatlopás – bennfentes, külső
- Rendszergazda
- Mérnök

Emberi beavatkozás elleni védelem

- Felhasználó/ügyintéző
    - szűk jogkör
    - adatbevitelt végez
    - a hozzárendelt, korlátozott erőforrásokat használhatja
    - ha gyakorlatlan, akkor például nem érzi az azonosítás fontosságát (egyszerű jelszó stb.)
- Üzemeltető
    - széleskörű ismerete és jogosítványa van a rendszer felhasználásával kapcsolatban
    - titoktartási kötelezettsége van (fontos információk)
    - beállítások végrehajtása, módosítása
    - felhasználók nyilvántartása, jogosultságok beállítása, jogosultságok életciklusának nyomkövetése
    - utasítások alapján dolgozik
- Mérnök
    - magas szintű informatikai végzettsége van
    - bizalmas információk – titoktartási kötelezettsége van
    - rendszer beállítása, módosítása, javítása

    - széles hatáskör
- Programozó
    - rendszer készítése
    - teljes hatáskör
    - komoly minőségbiztosítási rendszerrel felügyelik a programot

Algoritmus védelem eszközei: titkosítás, digitális aláírás, hash függvények

Hash függvények

- Hash függvények: adatbázisok rendszerezésére
- Kriptográfiában: adatok integritásának biztosítására
- Tetszőleges hosszú adatok helyett egy fix hosszúságú, igen kisméretű bitsztringre (kb. 160 bit) koncentrálunk
- A tetszőleges méretű üzenetre egy hash függvényt hajtunk végre, melynek eredményeként egy fix méretű hash értéket (üzenetkivonatot vagy lenyomatot) kapunk
- $H: \{0,1\}^* \rightarrow \{0,1\}^n$
- Integritásvédelem: a nagyméretű, eredeti üzenetünk változásának ellenőrzéséhez: hash függvény az eredeti üzenetre, összehasonlítva a korábbi üzenetkivonattal
- Példa: programkód, smart kártya
- Ne lehessen megadni két olyan üzenetet, amelyeknek a lenyomata megegyezik
- Nehéz legyen olyan üzeneteket találni, amelyek hash értéke megegyezik (ütközésmentes h függvény)
- A lenyomatból az eredeti üzenet kiszámítása nehéz

Hagyományos vs. digitális aláírás

- Hagyományos aláírás
    - fizikai dokumentum részét képezi
    - több oldalas dokumentum minden oldalát alá kell írni
    - ellenőrzése egy hiteles aláírási minta alapján történik
    - aláírt dokumentum fénymásolata megkülönböztethető az eredetitől
- Digitális aláírás

- hozzácsatolódik az elektronikus dokumentumhoz
- hosszabb dokumentumnál elég egyszer aláírni
- nyilvános ellenőrző algoritmus
- az aláírt üzenet könnyen másolható
- sosem ugyanaz

Digitális aláírás jellemzői

- Hitelesítés: a fogadó félnek bizonyítjuk, hogy az információ nem lett megváltoztatva, kicserélve valamely támadó által
  - üzenet adatintegritása (a dokumentum aláírás után nem változtatható meg)
  - üzenet eredetének igazolása (az aláíró kiléte beazonosítható a kulcspárja segítségével – hitelesítés szolgáltató által kiadott tanúsítvány)
- Letagadhatatlan: bárki által ellenőrizhető, hogy ki írta alá
- Hamisíthatatlan
- Az aláírás nem átruházható

Digitális aláírási séma definíciója

- Jelölje P a lehetséges üzenetek halmazát, és A az aláírások halmazát. Az AS = (K, Sign, Ver) digitális aláírási séma három algoritmusból áll:
  - a K kulcsgeneráló algoritmus
  - a Sign aláíró algoritmus
  - a Ver ellenőrző algoritmus
- Ez így már elfogadhatóan gyors, de…
- Aladár nem lehet biztos abban, hogy Kriszta nyilvános kulcsa tényleg hozzá tartozik
- Kell tehát erre egy igazolás, amelyet egy hitelesítő szervezet ad ki
- Erre Krisztának is szüksége van, hogy hamisítás esetén bizonyítani tudja az igazát

Mivel és hol írjuk alá a dokumentumokat?

- Nem tollal! Vagy ha igen, akkor a tollnak legalább annyit kell változnia, mint a lúdtollnak a mai írószerszámokhoz képest
- A digitális tollnak elég nagy számítási teljesítménnyel kell rendelkeznie, de ne legyen mások számára elérhető és vihessük mindig magunkkal.

- Megoldási javaslat: a privát kulcsot az aláíró algoritmussal helyezzük el egy aktív memória kártyán vagy egy pendrive-on

Gyakorlati alkalmazások

- Hivatalos okiratok aláírása (adóbevallás, cégeljárás, ügyvédi ellenjegyzés, közigazgatási hatósági eljárás, elektronikus számlázás, vizsgalejelentések)
- Időbélyegzés
- Vak aláírás (elektronikus szavazások, elektronikus pénz)
- Online nyereményjátékok (Puttó)
- Kód aláírás
- Partner-azonosítás

Az AES és RSA algoritmusok

AES

Advanced Encryption Standard

- A NIST 1997-ben felhívás új szimmetrikus titkosító szabványra
- 2000-ben eredmény: győztes Rijndael, alkotói Vincent Rijmen és Joan Daemen
- 128/192/256 bites blokkokat 128/192/256 bites kulccsal titkosít, minden párosításban

AES 128 kódolása

- A 128 bites input szót 16 bájtra bontja és ezeket egy 4x4-es táblázatba rendezi, amelyet állapotnak nevez
- Az állapotra 9 teljes és egy részleges fordulóban 4 függvényt alkalmaz
- 11 menetkulcsot generál a mesterkulcsból

AES függvényei

- ByteSub(State): az állapot minden bájtját kicseréli egy S-box által meghatározott bájtra. Az S-boxot matematikai függvényként is ki lehet számítani.
- ShiftRow(State): az állapot i-dik sorát i-1 pozícióval balra tolja.
- MixColumn(State): az állapot oszlopait, mint vektorokat megszorozza egy mátrixszal.
- AddRoundKey(State, RoundKey): bitenkénti xor az aktuális állapot és a menetkulcs között.
- Jelölés:

    - ByteSub = B
    - ShiftRow = S
    - MixColumn = M
    - AddRoundKey = A.
- Az algoritmus folyamata: A BSMA BSMA BSMA BSMA BSMA BSMA BSMA BSMA BSA

RSA

- Ronald Rivest, Adi Shamir és Leonard Adleman publikálta 1977-ben.
- Leggyakrabban használt aszimmetrikus vagy nyílt kulcsú titkosító algoritmus.
- A biztonsága azon alapul, hogy nagy számokat nagyon nehéz prímszámok szorzatára bontani. (faktorizáció)

RSA paraméterek

- Legyenek:
    - p, q prímszámok, $n = pq$, $\varphi(n) = (p-1)(q-1)$
    - $1 < e, d < \varphi(n)$ olyanok, hogy $ed \bmod \varphi(n) = 1$
    - n es e a nyilvános kulcsok
    - d a titkos kulcs

RSA titkosítás és visszafejtés

- Legyen $0 \leq x < n$, akkor a titkosítás:
    - $y = RSA(x) := x^e \bmod n$
    - ezt a nyilvános kulcs (n, e) ismeretében bárki ki tudja számítani
- Ha x és n legnagyobb közös osztója 1, ami nagyon valószínű, akkor a visszafejtés:
    - $RSA^{-1}(y) := y^d \bmod n$
    - ezt csak az tudja kiszámítani, aki d-t ismeri.

RSA paraméterek választása

- p, q legalább 512 bit nagyságú prímszámok, amelyek különbsége legalább 500 bites
- $n = pq$ és $\varphi(n) = (p-1)(q-1)$ kiszámítása kézenfekvő
- e-t véletlenszerűen választhatjuk és $(e, \varphi(n)) = 1$
- e és $\varphi(n)$ ismeretében d-t kibővített euklideszi algoritmussal lehet meghatározni

Legyenek p és q különböző prímszámok, azaz olyan természetes számok, amelyeknek 1-en és önmagukon kívül nincs más osztójuk. Végtelen sok prímszám létezik. Annak a valószínűsége, hogy egy véletlenszerűen kiválasztott x-nél kisebb szám prímszám legyen 1/ln x. A Miller-Rabin teszttel gyorsan eldönthető, hogy nagy valószínűséggel prímszám-e.

Nagy prímszámokat tehát könnyű találni, de ha két ilyet összeszorzunk, akkor csak a szorzatot ismerve nagyon nehéz a tényezőket megtalálni. Ezt a faktorizáció problémájának nevezik, ami nehéz algoritmikus probléma.

Legyenek p és q különböző prímszámok és n=qp. Ekkor az n-nél kisebb, n-hez relatív prím természetes számok száma $\varphi(n) = (p-1)(q-1)$. Ezt az értéket p és q ismeretében könnyű kiszámítani. A φ függvényt Euler függvénynek nevezzük. Legyen most e egy olyan φ(n)-hez relatív prím természetes szám, amelyik kisebb φ(n)-nél. Akkor pontosan egy olyan $1 <= d < \varphi(n)$ természetes szám létezik, amelyre $ed \bmod \varphi(n) = 1$. Ezek után a nyilvános kulcs az e,n számpáros, a titkos kulcs pedig a d szám. A kulcsok meghatározása után a p és q értékét is titokban kell tartani vagy ezeket a számokat meg kell semmisíteni.

A kódolás során az üzenetet először számok sorozatává alakítjuk olyan módon, hogy a számok mindegyike kisebb legyen, mint n. Ez könnyen megtehető, hiszen az üzenetet a számítógépben bináris alakban tároljuk és most ezt a bináris sorozatot, mint egy kettes számrendszerben megadott szám számjegyeit értelmezzük. Ezután az egyes m számokat az

$M = me \bmod n$

képlettel kódoljuk előállítva a rejtjelezett M üzenetet. A kódoláshoz csak a nyilvános kulcsot, az e,n számpárt kell ismerni! A titkos M üzenetet az

$m = Md \bmod n$

képlet alapján lehet dekódolni. A visszafejtéshez tehát a titkos d kulcs ismerete kell!