Bio-informatika

2015

(magyar fordítás)

A fordítás teljes mértékben Pongor Sándor bioinformatika diasorain alapszik, és egyes helyeken azokból származó, színben és nyelvben módosított ábrákat tartalmaz.

Tartalomjegyzék

1. dia - Bevezetés az adattípusokba 3
1. rész: A tárgy definíciója – A bioinformatika 3
2. rész: A biomolekuláris adatok elmélete, szerkezet/struktúra és funkció (rendszer elmélet), az adat annotációja 4
2/A. rész: Annotáció 6
3. rész: A 4 sztenderd adattípusról részletesen 7
4. rész: Összefoglalás 9
2. dia – Hasonlóságok és csoportosítás 10
Reprezentációk 12
Összehasonlítás 13
Granularity 17
3. dia – Szekvencia illesztés 19
Helyettesítési mátrixok 20
Dot plot 23

1. dia - Bevezetés az adattípusokba

1. rész: A tárgy definíciója – A bioinformatika

Mi az a bioinformatika?

- Szűk definíció szerint:
A biológiai adatok tudománya. Főleg molekuláris biológia. Leírás, kezelés, értelmezés.
- Tágabban értelmezve:
A biológiai ismeretek tudománya. Minden (molekuláris) biológiával kapcsolatos alkalmazás, beleértve a modellezést (viselkedés szimulációja) is.

Megjegyzés: történelmileg kapcsolatban áll a DNS-szekvenálás és a 3D fehérje-szerkezet-meghatározás '80-as évekbeli forradalmával. Ma már kiterjesztették a biológia minden területére.

A számítógépek használata a biológiában:

Adatok kezelése:
- megszerzés
- tárolás, annotáció
- interpretáció, analízis, data-mining

Modellezés, szimuláció

A számítógép használata ma már minden tudományágban nélkülözhetetlen. A biológiában lehetségessé tette a nagy, "ipari" mennyiségű adat kezelését, az egyszerű objektumoktól a **nagyméretű rendszerek** szintjére való lépést:
- a molekuláris biológia / hagyományos bioinformatika egy vagy néhány objektummal foglalkozik
- a modern, kísérleti megközelítések nagy számú tárggyal foglalkoznak egyidőben
  → "**systems biology"** (="rendszerek biológiája")
- biológiai rendszerek: egy sejt, egy szövet, egy szerv, egy szervezet…
- tipikus példa: genomok

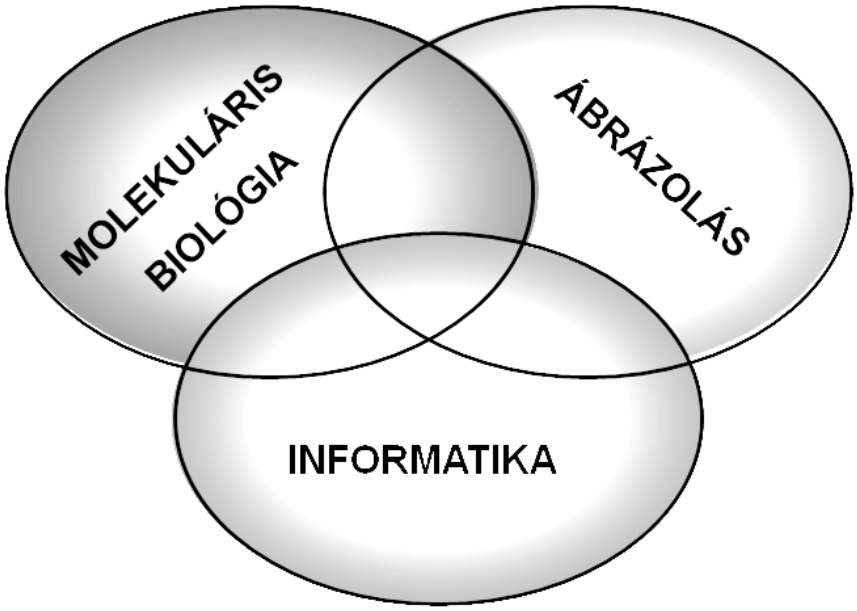
Az egyéni, külön megfigyelésekből egy holisztikus, azaz a rendszert átfogó hálózati modell alakítható ki.

Példák:

Egyedi entitások (molekuláris biológia):
- egy molekula mozgásának modellezés *in vacuo* vagy vízben (molekula modellezés, molekula-dinamika)
- dokkolás (pl. farmakónok receptor fehérjéhez)

Nagy rendszerek (system biology):
- nagy molekulák modellezése
- biológia közösségek modellezése (pl. baktériumok, állatok, emberi tömegek)

A bioinformatika interdiszciplináris:

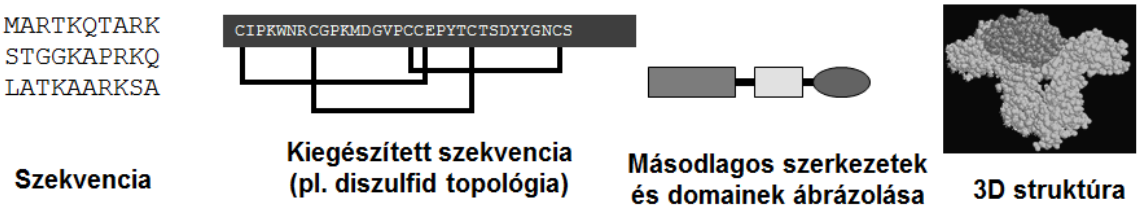


Összességében a bioinformatika:

- tárgya: molekuláris szerkezetek, anyagcsere/metabolikus útvonalak, szabályozó hálózatok ÉS adatbázisaik
- módszere: analízis és hasonlóság kihasználása
- a biológiai ismeretek komplexitása (és kevésbé az adatok mennyisége)

2. rész: A biomolekuláris adatok elmélete, szerkezet/struktúra és funkció (rendszer elmélet), az adat annotációja

Molekuláris szerkezetek különböző reprezentációi:



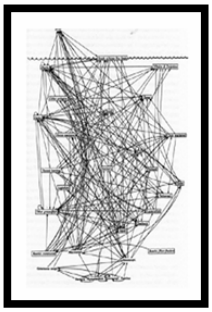
Szekvencia | **Kiegészített szekvencia (pl. diszulfid topológia)** | **Másodlagos szerkezetek és domainek ábrázolása** | **3D struktúra**

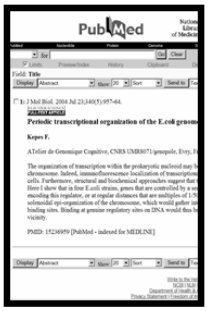
Szimbolikus diagramok (pl. hidrofobicitás plotok, helikális kör diagramok)

Egyszerűsített 3D rajzok

Mindegyik típusnak vannak egyszerűbb, vagy kiegészített (annotációkkal ellátott) fajtái.

- szekvencia
- 3D szerkezet
- hálózat/gráf
- szöveg





Ezen különböző elemek halmazán relációkat lehet definiálni, ezzel kapcsolatokat felépíteni közöttük.

Ontológia:

1. **engedélyezett** elemek és relációk összessége, **szókészlet**
2. **szabályok** egy bizonyos tudás-doménen belül

pl. gén ontológiák, 3D szerkezet ontológiás stb

Például: rendszer: molekula; szókészlet: atomok; relációk: kémiai kötések

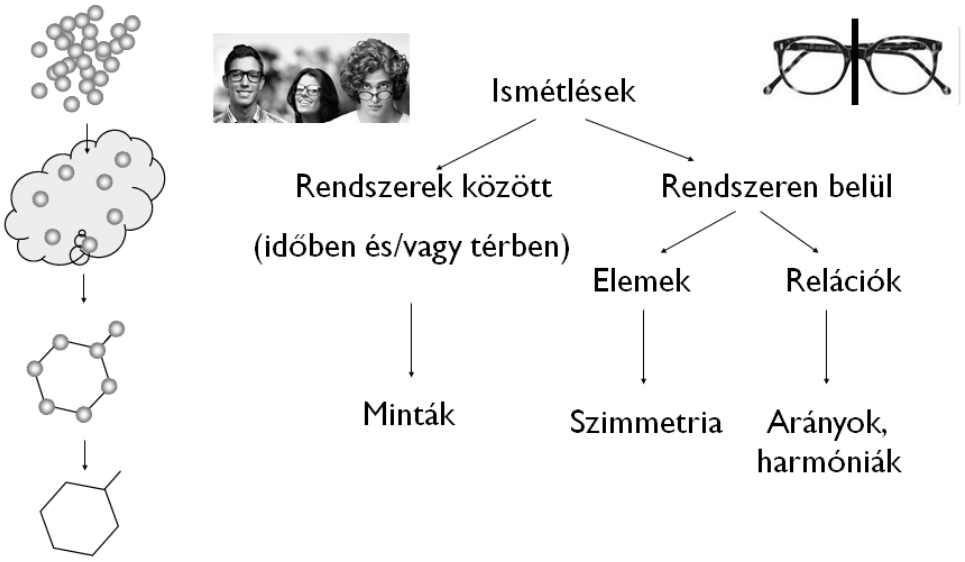
Az adattípus kiválasztásának szabályai:

- a szerkezet leírása hierarchikus
  (atom → aminosav → fehérje → útvonal → sejt → szövet stb.)
- egy adott problémához érdemes egy sztenderd leírást, vagy „legbelső szerkezeti szintet" kiválasztani. pl. a molekuláris biológiai problémák sztendersz szintje a DNS szekvencia.
- a szenderd vagy legbelső leíráshoz mindig van egy logikai szerkezetünk, emellett több hozzáadott, egyszerűsített és annotált nézetünk

Mi az a rendszer?

- a valóság bármely része, ami elhatárolható a környezettől. Egy közösség a környezetben.
- részei egymással interakcióban vannak
- interakcióban van a környezetével (input, output)
- a rendszer modellek a valóságot generalizálják
- van egy rendszerük, amit részek és folyamatok definiálnak
- a részeknek funkciós és strukturális kapcsolatai is vannak egymással

struktúra: egy (térben-időben állandó) elhelyezkedése az elemeknek vagy tulajdonságoknak.
funkció: a rendszerben játszott szerep.



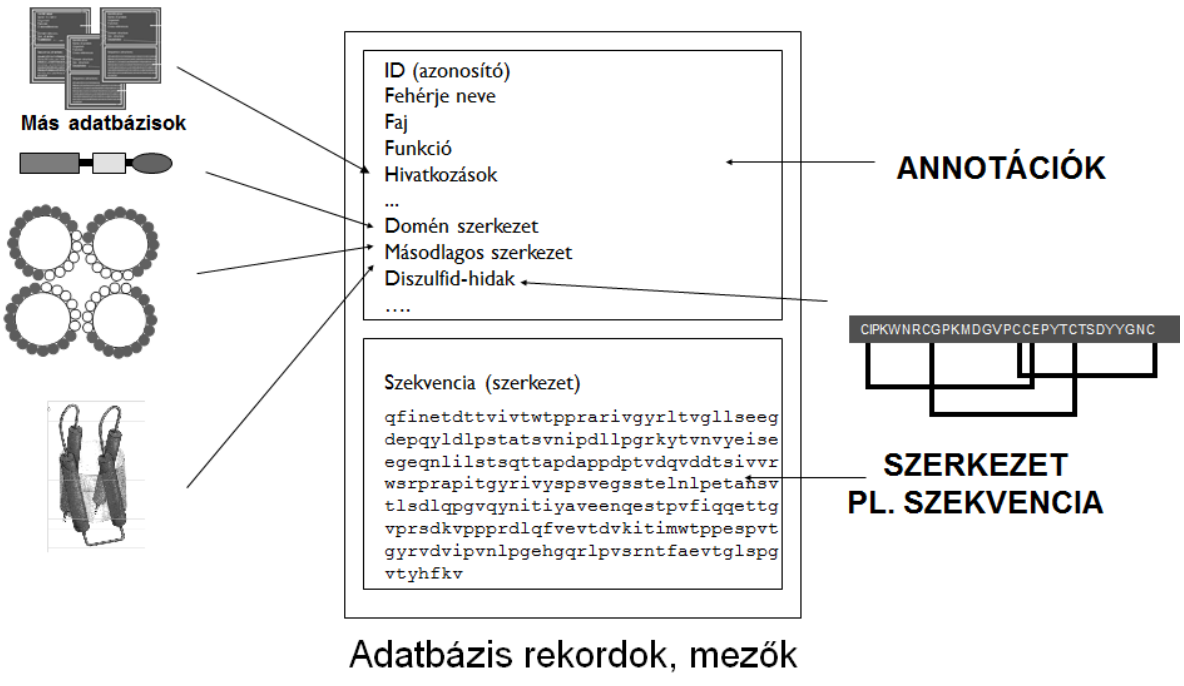
2/A. rész: Annotáció

annotáció: „adding notes" azaz az elem ellátása jegyzetekkel

- adat (pl. egy szekvencia)
- adat az adaton (annotáció, meta-data)
- adat az annotáción (ontológiák, meta-meta-data: az annotáció nyelvének meghatározása)

Adatbázis létrehozása nyers adatból és annotációkból:

- a nyers adat hozzáadása az adatbátishoz
- adatok ellátása alapvető annotációkkal (project neve, dátum stb.)
- annotációk hozzáadása hasonlóság alapján, azaz adatbázis keresés
- további információ hozzáadása emberi tudás alapján (analízis programok, irodalomban keresés)



Annotáció szekvencia esetében pl.:
globális leírók: pl. funkció
lokális (helyi) leírók: pl. domének

Annotáláshoz szükséges az adatbázisban való keresés és a „biológiai" tudás.

Egy annotáció akár struktúra is lehet. A nukleotid- és aminosav-szekvenciák lineáris, láncszerű szerkezetének szakaszaihoz tulajdonságokat rendelhetünk. Ez a korábban említett, hálózat-szerű rendszer elmélettel ellentétben adatbázis-központú nézete az elemeknek (entitásoknak) és relációknak.

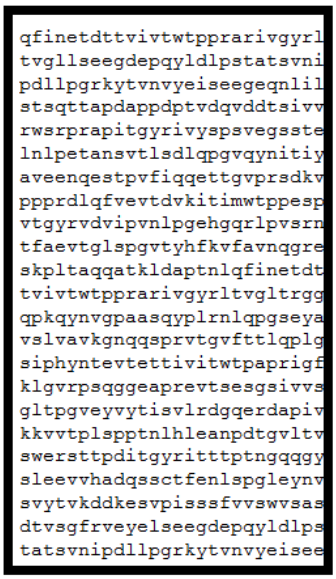
Annotáció és az internet

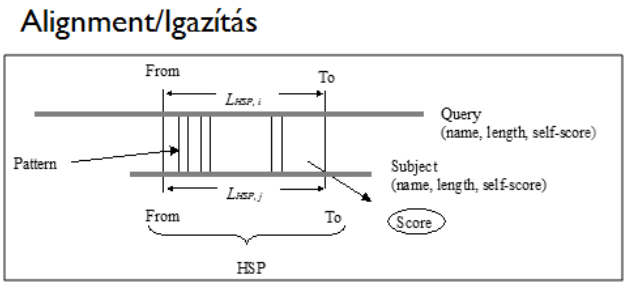
- az annotációkat emberek ellenőrzik, érvényesítik és adják hozzá: gén funkciójának megbecslése, molekuláris adatok strukturális és funkcionális leírása stb.
- az világháló a legnagyobb annotációs rendszer: milliónyi ellenőrizetlen link tartozik az adatokhoz.
  A legfontosabb típusokba adatbázisok is tartoznak (bioinformatikai és bibliográfiai), pl. Wikipédia (közösségen alapuló enciklopédia), speciális wikik, blogok, fórumok stb. A Google keresés az első lépés…
- manapság egy adatbázis annotációja annyit jelent, hogy sztenderd nyelvi leírókat hozunk létre az adatokhoz, melyet internet-linkeken és speciális programokon keresztül érvényesítünk. Emberi beavatkozást igényel.

3. rész: A 4 sztenderd adattípusról részletesen

Szekvenciák (nyelv metafóra)

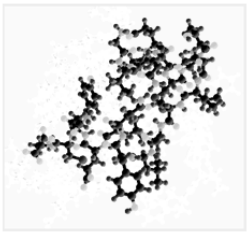
- logikai szerkezet: kémiai képlet (aminosavak vagy nukleotidok sorozata)
- sztenderd leíró: karakterek sorozata
- egyszerűsített vagy kiegészített nézet (pl. funkciókkal, vagy másodlagos szerkezetekkel)



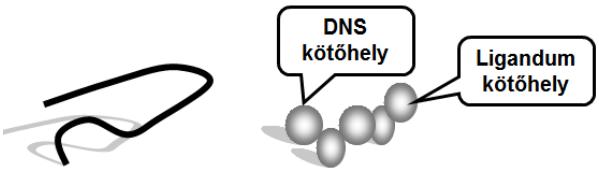


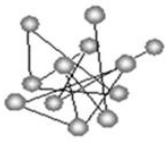
3D szerkezetek (objektum metafóra)

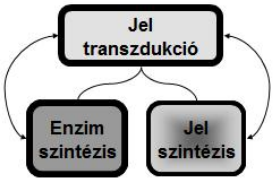
- logikai szerkezet: 3D kémiai szerkezetek
- sztenderd leíró: 3D koordináták + alegység leírók
- (atom, aminosav, nukleotid)
- egyszerűsített és kiegészített (annotált) vizuális megjelenítés

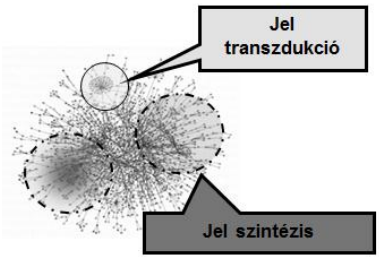


$(x_i, y_i, z_i)_n$









Hálózatok (szociális metafóra)

- logikai szerkezet: entitások, relációk
- sztenderd leíró: gráfok
- egyszerűsített és kiegészített (annotált) vizuális megjelenítés

Szövegek (= tudományos publikációk, kommunikáció metafóra)

- tudományos nyelven írt emberi üzenet ("special English", ~fixed vocabulary).
- ennek is van logikai szerkezete, szenderd, egyszerűsített és kiegészített leírása és adatbázisai
  - egyszerűsített leírók------- > cím (+ szerzők stb.)
    kivonat
  - kérdés---------- > **bevezető**
  - válasz---------- > **eredmények** (mi?), **módszerek** (hogyan?)
  - konlúzió ---------- > **értekezés**
  - annotációk ---------- > referenciák
    kulcsszavak
- DE. az üzenetnek van kibocsátója/emittere (szerző), közönsége (olvasó, kritikus). Másképpen fogalmazva, kontextusfüggőek (nem úgy, mint pl. atomok vagy szekvenciák)
- Laza szerkezetűek (pl. molekulákkal ellentétben). Léteznek ontológiák a nyelvhez, de nem magukhoz a cikkekhez!

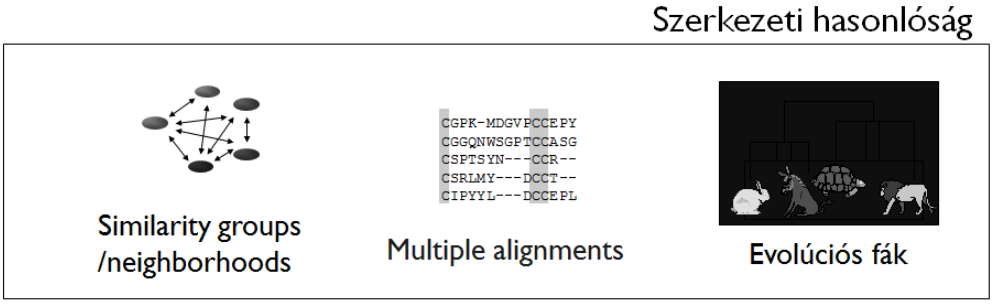
4. rész: Összefoglalás

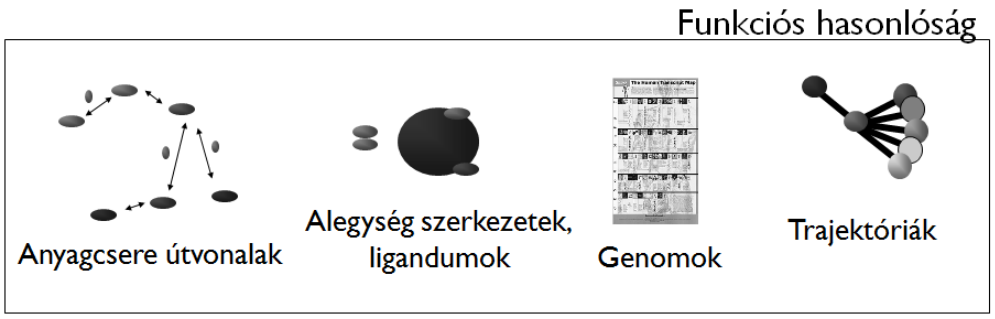
- a számítógépek használata a biológiában magába foglalja a **bioinformatikát** (adatok kezelése, adat-bányászat/data-mining), a **modellezést** és a **szimulációkat**
- a **rendszer**, **struktúra** és **funkció** fogalma.
  struktúra/szerkezet: elemek/entitások és relációk összessége.
  logikai struktúra, egyszerűsített, kiegészített (annotált) leírások.
- a négy alapvető adattípus: **szekvencia, 3D, hálózat és szöveg**
- a modelleket arra szánt adatszerkezetekkel lehet **reprezentálni** képi vagy szöveges formában.
- az adatbázis rekordok számos adattípust tartalmaznak emberi és gépi olvasásra alkalmas formában
- az **annotáció** (hozzáadott emberi ismeret) nélkülözhetetlen, jobb, ha gép számára olvasható.

2. dia - Hasonlóságok és csoportosítás

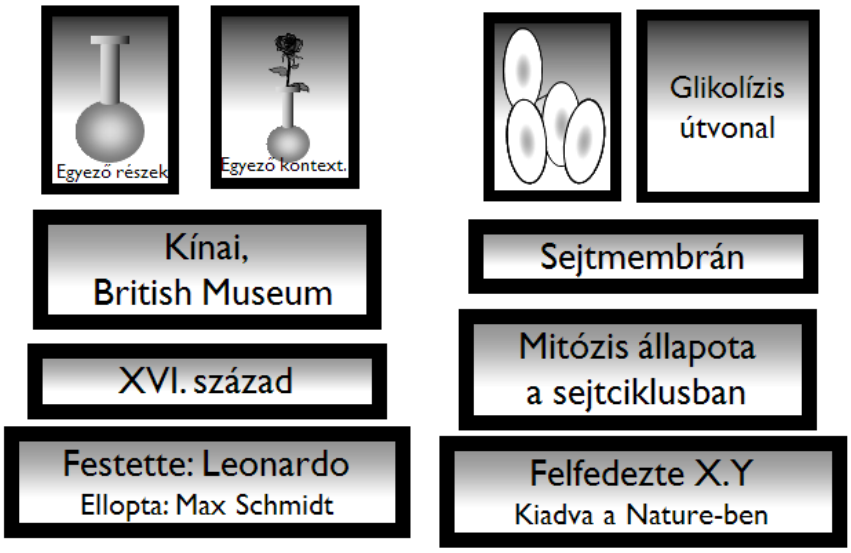
Szekvencia alapok

- ábécé: nukleotidoknál 4-, aminosavaknál 20-betűs
- formátum: FASTA, összefűzött FASTA
- az adat megértése: csoportosítás és besorolás, ismereti egységekbe rendezés, más egységekhez hasonlítás és párosítás
- a emberek ”logikai struktúrákban” gondolkodnak, míg a számítógépek leírás alapján (amit emberek adnak meg nekik).

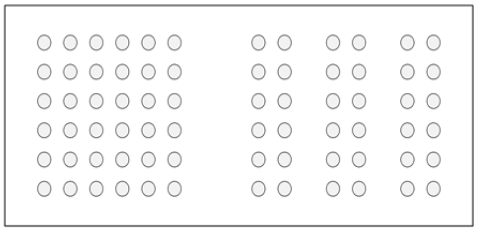




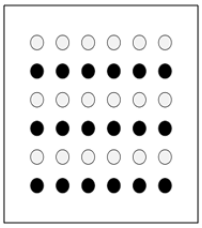
Pl. vázás példa. A két váza alakban (szerkezetben) és funkcióban is hasonló, megosztott a kontextusuk és részeik. Így ezeket egy csoportba sorolhatjuk, és a megkülönböztetés és azonosítás érdekében tér- és időbeli mintákat rendelhetünk hozzá, majd más mintákra is asszociálhatunk:



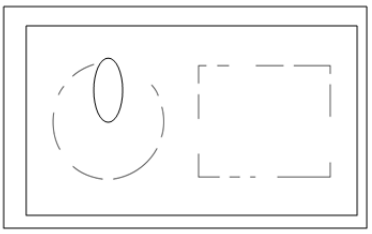
Az ember ösztönösen mintákká aggregálja a látottakat, amit nem lehet hagyományos, szerkezet alapú módszerekkel leírni. Gestalt pszichológiai alapelve ezt próbálja meg:



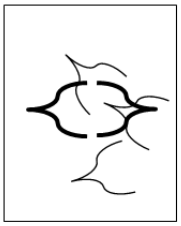
Távolságok alapján



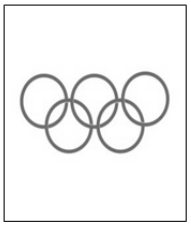
Hasonlóság



Folyamatosság és zártság



Szimmetria



Egyszerűség ("Pregnänz")

Ilyen hasonlóságokat lehet állítani a 4 alap adattípus esetében is, pl.:

- szekvencia: glicin-gazdag régió
- 3D-szerkezet: α-helikális, β-redő stb.
- hálózat: skálafüggetlen
- szöveg: azonos író, téma stb.

és ezeket motívumokkal ábrázolni.

Ember	Gép
Intuitív minták Hasonlóságot motívumok definiálják EZUTÁN esetleg kiértékelés (score)	Leírók (pl. string, vector) Hasonlóság numerikus formában, számítások alapján (score) **Előre definiált** minták keresése
Minták: **ösztönös** vagy **ismereti** alapon	A leírások, numerikus mérők és minták mind **előre definiáltak**
A minták választása és formája **rugalmas**	A score-ok és motívumok ellenőrzése **statisztikával** történik
A konszenzus minták **emlékezetben** tároljuk, és tapasztalattal, ellenőrzéssel frissítjük	Memória: **adatbázisok**
Rugalmas, kvalitatív	**Rigid, kvantitatív**

Reprezentációk

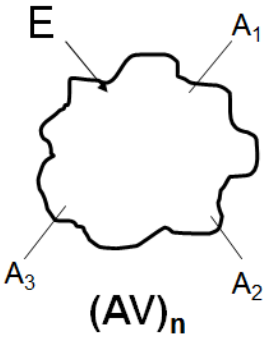
Hogyan választunk reprezentációt?

- általánosságban: modellezni szeretnénk valamit
- prediktív modellek bármilyen tulajdonságot használhatnak (szerkezet nélküli leírók). Pontos predikcióhoz az „információ gazdagság" a fontos, nem a változók pontos értéke.
- interpretációs modelleknél a tartalomnak a lehető legegyszerűbbnek kell lennie, de a jelentés fontos (a lehető legjobban szerkezetbe rendezve)
- **röviden**: megfelelő reprezentáció kell.

Reprezentáció fajtái

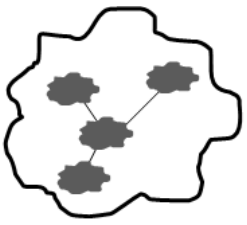
- **strukturált** és **strukturálatlan**
  attól függ, hogy tudjuk, akarjuk-e használni a belső szerkezetet
- **granularity** (=szemcsésség, finomság): a leírás felbontása (pl. 4 nukleotid, 16 dinukleotid)

Strukturálatlan reprezentációk



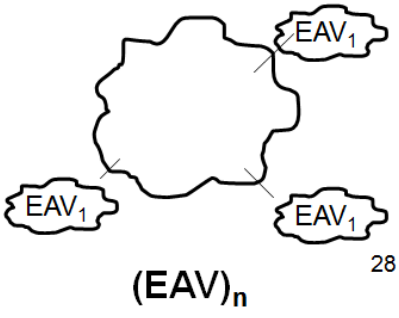
- nem tudunk semmit a belső szerkezetről
- csak a tulajdonságok (globális leírok) ismertek, melyek diszkrétek vagy folytonosak
- legjobban **vektorokkal** írhatóak le
  (minden dimenzió egy attribútum, és a tartalma az érték)
  sok dimenzió lehet
- a vektorműveletek gyorsak
- vektortípusok:
  - **bináris**: 010011000 stb.
    tulajdonság jelenlétét vagy hiányát reprezentálja
  - **nem bináris**: lehet valós vagy természetes szám

Strukturált reprezentációk



- ismerjük a belső szerkezetet entitások és relációk formájában (mind leírható attribútumokkal (A) és értékekkel (V) →EAV and "RAV")
- információ-gazdag, így részletes összehasonlításokat tesz lehetővé
- hasonlításhoz alignment (matching) szükséges…
- például: karakter string (szekvencia), gráfok (a legtöbb molekuláris szerkezet)

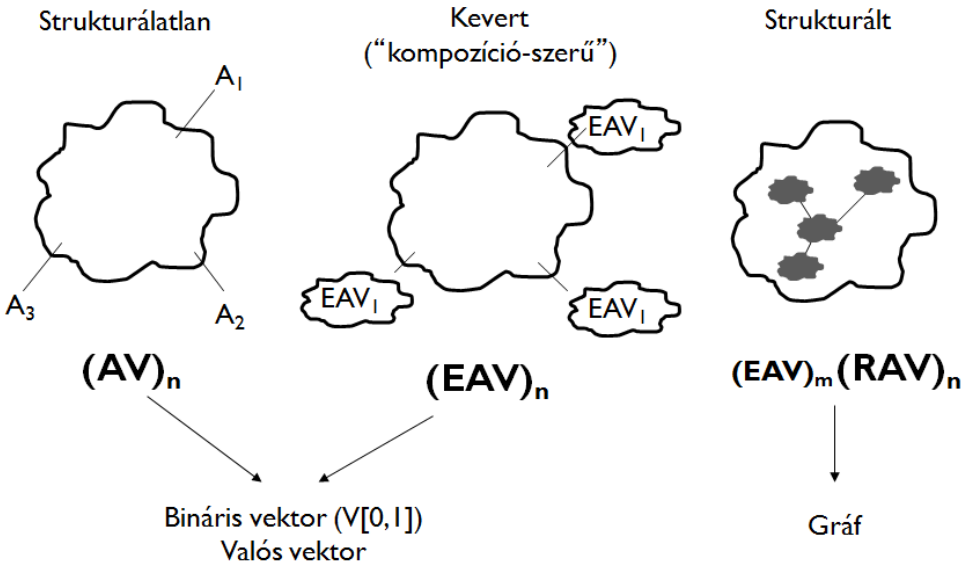
Kevert, kompozíciós leírások



- ismert szerkezetű részekre bontjuk az objektumot, és megszámoljuk a részeket (pl. atom összetétel, H_2O, vagy fehérje aminosav összetétele)
- az eredmény **vektor** → gyors számolás, nem kell alignment
- a vektor információtartalma a részek granularity-én múlik. pl. egy fehérje vagy egy egész ember atom összetétele nem túl informatív.

28

Összességében:



Összehasonlítás

- Input: két leírás
- Output:
  - strukturálatlan: score (hasonlóság, távolság), kötelező
  - strukturált:
    - score (ugyanúgy kötelező) ÉS
    - egy közös minta (az alignment eredménye), tetszőleges

Távolságmérés

- hasonlóság mérése (0 ha különböző, nagy ha hasonló)
- távolságok (nagy ha különböző, 0 ha egyforma)
- vektorokra és szerkezetekre is létezik
- „jól viselkedik": ha korlátolt, pl. [0, 1]
- nem lineáris! („kétszer olyan hasonló" – ennek nincs értelme)
- hasonlóság $S \sim \frac{1}{D}$ vagy $1 - k \cdot D$

pl. vektortávolság (eukledészi metrika)

$$D_{ab} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}$$

Metrika tulajdonságai:

1. $D_{ab} > 0$
2. $D_{aa} = 0$
3. $D_{ab} = D_{ba}$
4. $D_{ab} + D_{bc} > D_{ac}$

Általánosított távolságok

n-dimenzióban:

$$D_{ab} = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2}$$

2-től különböző exponenssel (Minkovszki metrika):

$$D_{ab} = \left(\sum_{i=1}^{n} |a_i - b_i|^k\right)^{\frac{1}{k}}$$

Dot product – skalárszorzat

$$A.B = a_1b_1 + a_2b_2 + ... + a_nb_n \qquad A.B = \sum_{i=1}^{n} a_ib_i$$

Bináris vektoroknál: az ugyanazon helyen lévő nem 0 elemek száma
Egyforma egységvektorok szorzata 1.0.

Asszociációs mérték

Bináris leírók közötti hasonlóság mérésére. Pl. Jaccard állandó:

$$J = \frac{a \cap b}{a \cup b}$$

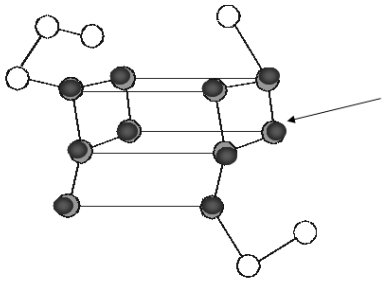
Egyforma halmazokra 1-et, teljesen különbözőkre 0-t ad.
Az ilyen korrelációs koefficienseket sokféle nem-bináris vektorhoz is lehet használni.

Sokféle távolság-mérték létezik, és számuk folyamatosan nő. Könnyű problémákhoz bármelyik működhet, míg bonyolultabbaknál lehet, hogy egyik se. Az ismeretlen távolság mértékektől nem kell félni, de bízni sem szabad bennük ☺

MOSOLYOGJ
A ZHRA KÉSZÜLÉS FUN

Strukturált leírások összehasonlítása

- Input: 2 strukturált leírás (pl. szekvencia)
- Output: 1) távolság mérték (score) and 2) közös minta (motívum).
- a távolság mértéket akkor használhatjuk, ha a leírást sikerült vektorrá változtatni (kompozíciós leírás)
- emellet lehet alignmentet csinálni a szerkezetekre, ami megadja a közös mintát
- a gráfokat a legnagyobb közös részgráf megtalálásával match-elhetjük. Ez egy számítási szempontból nehéz probléma, NP teljes.
- az emberi tudatban a hasonlítás ösztönös

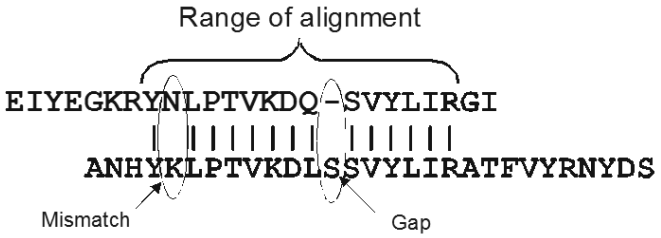


Karakter stringek összehasonlítása

```
A                    B
1: 01010010          1: BIRD
    |||||                 ||
2: 11010001          2: WORD
```

$D_{12}=3$ $D_{12}=2$

- Hamming távolság – egyforma hosszú stringekre alkalmazható, nincs alignment.
- ha lookup table-t társítunk hozzá, súlyozva lesznek a különbségek

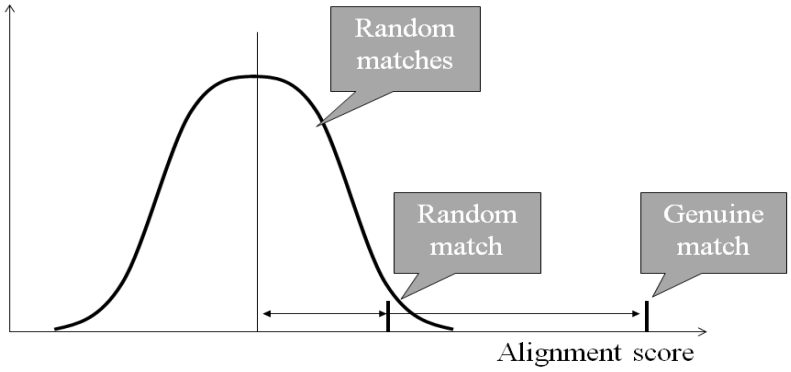


- Levenshtein távolság: a különbségekhez, egyezésekhez és gapekhez (lyuk) rendelt cost-ok összege. A két stringnek nem kell egyforma hosszúnak lennie.
- a hasonlóságmérték az alignment hosszán belül ad meg egy maximumértéket. Ez a maximum függ a scoring system-től, ami magába foglalja:
  - lookup table – cost-ok táblázata, pl. BLOSUM mátrix aminosavakhoz
  - gapek értéke
- gyakran nem metrika, de közel jár hozzá

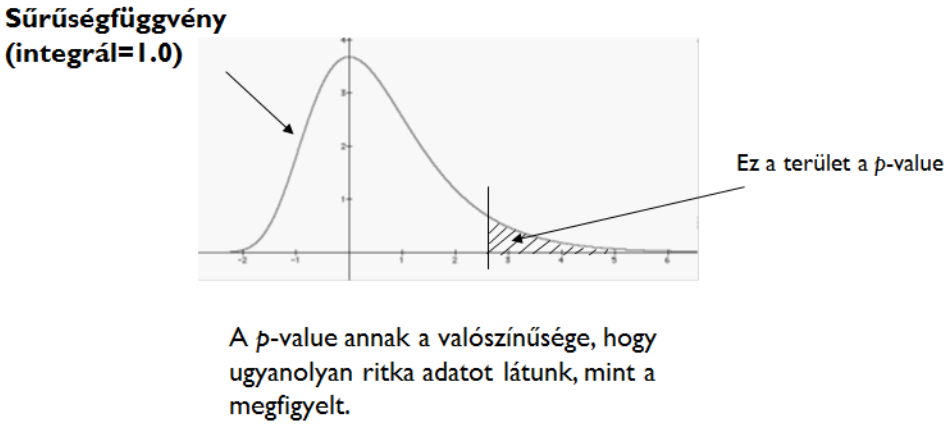
Illesztett (aligned) szekvenciáknál az egyező motívumok evolúciós konzerváltságra utalnak. Ez egy sima szekvenciánál informatívabb. „Amit a szekvencia suttog, az alignment mintázat üvölti”.

A „similarity score” jelentősége

- a lényeges score-ok különbözhetnek, van, amelyik korlátos, míg más nem. Hogy lehet őket ugyanarra a nagyságrendre hozni?
- a természetes nagysárgrend a valószűség, és ezt a valószínűséget nevezzük significance-nek (jelentőség), ezt tudjuk egyszerű statisztikából.
- hogy mondjuk meg, milyen jelentős a score amit találtunk?
- vagy másképpen: mennyire különbözik a score attól, amit véletlenszerűen kaphatnánk?
- ehhez tudnunk kell a random hasonlóságok eloszlását, és szimulálni valamilyen formában…



Ha a kettő közé esik, a határra, akkor mi van? ***p*–value!**



A mérnök útmutatója a szignifikanciához

- Szignifikancia:
  1) annak a valószínűsége, hogy véletlen ezt a score-t kapjuk (*p*-value)
  2) az a szám, ahányszor várhatóan ezt a score-t kapjuk (E-value)
  mindkettőre igaz, hogy minnél kisebb, annál jobb
- a *p*-t becsülni lehet úgy, ha készítünk egy hisztogramot a random score-okról, linearizáljuk, és leolvassuk a *p*-t a lineáris görbéről.

A Z-score: két szekvencia, A és B összehasonlításának significance-e

- először kiszámítjuk a score-t (ez a "genuine score", $S_{genuine}$)
- N-szer megismételjük a következőt (N > 100):
  - randomizáljuk az A szekvenciát az aminosavak összekeverésével (ez lesz a "random adatbázis")
  - align-oljuk a randomizált A szekvenciát a B-hez, kiszámítva egy újabb score-t, ez lesz S_{random}
- kiszámítunk egy átlag $\hat{S}_{random}$-t és egy szórást: σ_{random}
- Ebből számítjuk a Z-score-t:

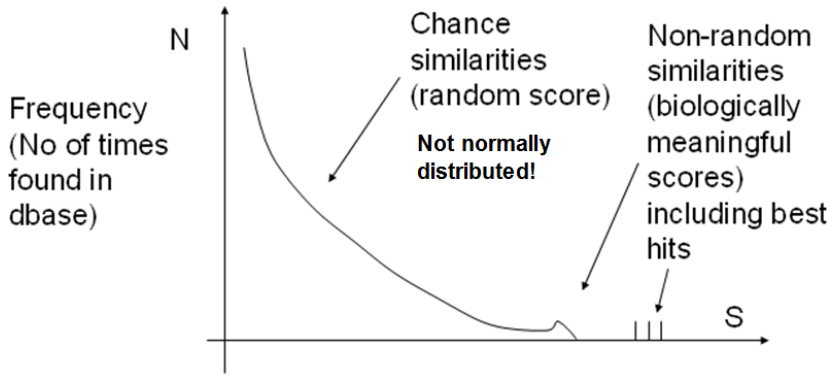
$$Z = (S_{genuine} - \hat{S}_{random}) \; / \; \sigma_{random}$$

Ez a számítás analóg az egymintás t- (student) próbával.

Miért NE használjunk Z-scoret?

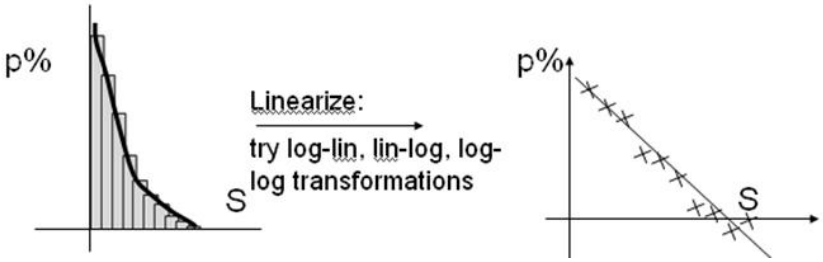
- a Z-score normal-eloszlást feltételez – ez nem igaz!
- a kevert, random A szekvencia sokszor visszaadja magát az A-t
- olyan szekvenciák, mint pl. AAAAAAAAA, változatlanok maradnak keverés után is, szóval σ_{random} nulla közelében lesz…
- a természetes szekvenciák NEM olyanok mint a random kevertek, sok dipeptid, tripeptid (és ~nukleotid) gyakoribb, mint mások
- a Z-score-ok kiszámítása minden alignment párhoz adatbázisban való keresés során (ez egymillió alignmentet is jelenthet) nem praktikus – túl sokáig tart.

Ha a szekvenciát az adatbázissal hasonlítjuk, az ebből kapott hisztogramon a random score-ok dominálnak, a biológiailag jelentős score-ok (a legjobb hit-eket beleértve) erősen kisebbségben vannak.



Szignifikancia számítás ismeretlen disztribúcióból:

1. % hisztogram a random score-okból
2. linearizálás (log-lin, lin-log, log-log transzformációk)
3. egyenes illesztése
4. ekkor x score-hoz tartozó significance value y.



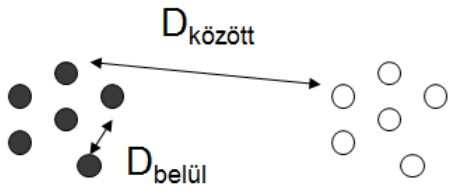
A chance similarities (random score-ok) megkapható létező szekvenciákkal, vagy random generáltakkal való hasonlításokból is. Valójában egyik sem helyes megoldás, de mindkettő jól működik. Általában elég messzire kell extrapolálni, mert a nagy (egyenes „végein" lévő) S-value-k elég ritkák.

Granularity

- a leírás felbontása (pl. vektoré)
- túl magas felbontás:
  minden objektum különbözőnek látszik, nem jelenik meg a hasonlóság azonos csoportok tagjai között…
- túl alacsony:
  minden egyfoma, nincs különbség a különböző csoportok között
- nehéz megtalálni a megfelelő részletességet. Ez része a „dimenzionalitás átkának"

Létezik optimum?

Adott problémához igen, meg lehet találni. Ki kell számítani minden távolságot csoportokon belül és csoportok között, majd ezeket a távolsághalmazokat pl. t-próbával összehasonlítani. Ezt minden felbontáson megismételve plotról leolvashatóbb, melyiknek magasabb a significance-e.

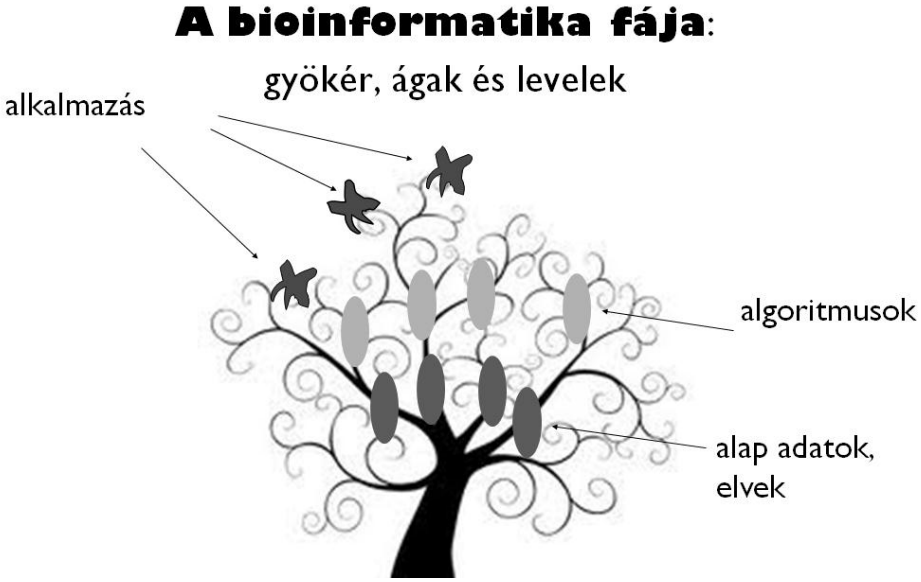


Ez persze csak normáleloszlások esetében igaz, de feltehetjük, hogy a plot minden pontján ugyanannyi az eltolódás.

3. dia – Szekvencia illesztés

Ebben a fejezetben

- szerkesztési távolság (frissítés)
- helyettesítési mátrixok (PAM, BLOSUM, hogyan építsd fel a sajátodat)
- a két alapvető illesztési algoritmus
- algoritmus fajták a végrehatása alapján: kizáró (exhaustive) és heurisztikus, globális és lokális.
- algoritmus fajták a tárgya alapján: két szekvencia, szekv. vs. adatbázis, szekv. vs. genom, sok szekv. vs. genom stb.



Évente új ágak és levelek…

Az illesztési algoritmus bemenetei:

1. 2 szekvencia
2. értékelési (scoring) séma (egy ~ formula ÉS egy ~ (helyettesítési) mátrix)

Lépései:

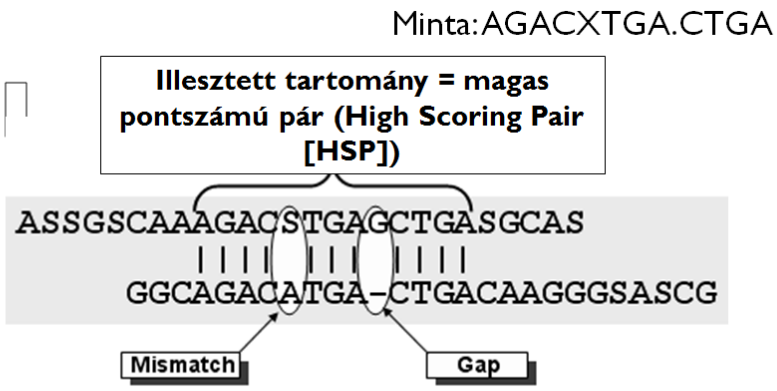
1. megtalálni az összes lehetséges illesztést (lefedést) és egy „gyors” értékelési rendszerrel megtalálni a legjobbakat
2. kiszámítani egy végső kvantitatív score-t a legjobb illesztésre (matching = illeszkedő)

Eredménye:

1. score (hasonlóság vagy távolság)
2. minta (motif)

Az ember is minták és score-ok alapján hasonlítja a dolgokat, de a mintákat az emlékezetében tárolja.

Ismétlés: hasonlósági score



Az S score az egyezésekhez és különbségekhez rendelt értékek összege, mínusz a levonás a lyukak miatt. Ezeket az értékeket helyettesítéasi mátrixban tároljuk. A „gap" általában a lyuk kezdetének és annak kiterjedésének értékének az összege.

Illesztési score

Score =

$$\sum_{\text{eleje}}^{\text{vége}} \text{Hasonlósági súlyok} - \sum_{\text{eleje}}^{\text{vége}} \text{Büntetés}$$

Levonás/bünti a lyukak miatt:

$$\text{Gap}_{\text{eleje}} + \text{Gap}_{\text{hossz}} \times \ell_{\text{gap}}$$

Függvények utóbbira:

- **lineáris**
  a lyuk hosszával egyenesen arányosan nő
- **affin**
  van egy kezdet és egy külön hossz komponense
- **probabilisztikus/valószínűségi**
  a szomszédos elemektől függ
- **egyéb**
  az elején gyorsan növekszik, de egyre hosszabb lyuknál lecsökken

A helyettesítési mátrix

- a helyettesítési (vagy pontozási - scoring) mátrix az aminosav egyezésekhez ls helyettesítésekhez tartozó értékeket tartalmazza.
- aminosavaknál ez egy 20×20-as, szimmetrikus mátrix, amit rokon szekvenciák páronkénti összehasonlításából lehet megszerkeszteni.
- a „rokon" jelentheti:

- evolúciós rokonságok egy „elfogadott” evolúciós fa alapján (ilyen Dayhoff PAM mátrixa)
- bármilyen hasonló szekvenciák a PROSITE adatbázisban (Hennikoff-féle BLOSUM)

- hasonló szekvenciákat tartalmazó csoportokat többszörös illesztésekbe (multiple alignment) lehet rendezni a mátrix elemek kiszámításához.

A mátrix kiszámítása egy multiple alignment alapján:

```
ASDESKLVV
  |
ATDDATLSI
  |     |
ASDSERITV
```

f(S/T)=3

f(S)=5, f(T)=3

Az elemeket a megfigyelt és várható megtalálási gyakoriságból számítjuk ki (log odds elv alapján), pl:

$$M(S/T) = \log\left(\frac{f(S/T)}{f(S)\times f(T)}\right)$$

S/T: S illeszkedik T-hez (vagy fordítva)

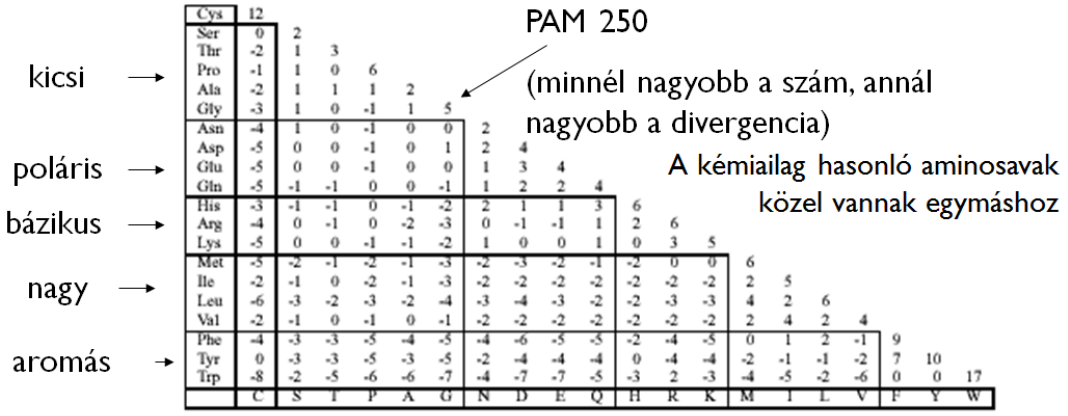
Ezeket sok (nem csak egy) multiple alignmentből kell kiszámítani.

Ezeket a log odd értékeket a végén normalizáljuk egy adott tartományra, attól függően, hogyan alkalmazzuk (pl. -5-től +15 történelmi okokból. De a tartomány nem igazán számít.)

PAM mátrix

- **P**ercent **A**ccepted **M**utation: az evolúciós változás egysége fehérje szekvenciákra. (Margaret Dayhoff ’78)
- “elfogadott” evolúciós fákba rendezett szekvenciák alapján (71 fa, 1572)
  Calculated from related sequences organized into “accepted” evolutionary trees (71 trees, 1572 csere [csak])
- 20x20 mátrix, az oszlopainak összege a megfigyelt esetek számával egyezik

- Pam_1 = az aminosavak 1% mutálódott
- Pam_30 = (Pam_1)^{30} *(mátrix szorzással)*



	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W
Cys	12																			
Ser	0	2																		
Thr	-2	1	3																	
Pro	-1	1	0	6																
Ala	-2	1	1	1	2															
Gly	-3	1	0	-1	1	5														
Asn	-4	1	0	-1	0	0	2													
Asp	-5	0	0	-1	0	1	2	4												
Glu	-5	0	0	-1	0	0	1	3	4											
Gln	-5	-1	-1	0	0	-1	1	2	2	4										
His	-3	-1	-1	0	-1	-2	2	1	1	3	6									
Arg	-4	0	-1	0	-2	-3	0	-1	-1	1	2	6								
Lys	-5	0	0	-1	-1	-2	1	0	0	1	0	3	5							
Met	-5	-2	-1	-2	-1	-3	-2	-3	-2	-1	-2	0	0	6						
Ile	-2	-1	0	-2	-1	-3	-2	-2	-2	-2	-2	-2	-2	2	5					
Leu	-6	-3	-2	-3	-2	-4	-3	-4	-3	-2	-2	-3	-3	4	2	6				
Val	-2	-1	0	-1	0	-1	-2	-2	-2	-2	-2	-2	-2	2	4	2	4			
Phe	-4	-3	-3	-5	-4	-5	-4	-6	-5	-5	-2	-4	-5	0	1	2	-1	9		
Tyr	0	-3	-3	-5	-3	-5	-2	-4	-4	-4	0	-4	-4	-2	-1	-1	-2	7	10	
Trp	-8	-2	-5	-6	-6	-7	-4	-7	-7	-5	-3	2	-3	-4	-5	-2	-6	0	0	17

A ma használt mátrixok

- **BLOSUM (leggyakoribb)**
  - Kifejlesztette: Henikoff & Henikoff (1992)
  - BLOcks SUbstitution Matrix
  - A BLOCKS adatbázisból
- **PAM**
  - Kifejlesztette: Schwarz and Dayhoff (1978)
  - Point Accepted Mutation
  - egymással rokon fehérjék manuális illesztéseiből

PAM

- az első használható mátrix fehérjékhez
- Markov Modellt feltételez az evolúcióra (azaz a szekvencia mindenhol egyformán és függetlenül mutálódhat)
- kicsi, egymással közeli rokonságban álló fehérjékből származtatott, ~15% diverzitással
- **magasabb** PAM szám: ritkább hasonlóságokhoz jó
- **alacsonyabb** PAM szám: nagymértékű hasonlósághoz jó
- 1 PAM ~ 1 millió évnyi divergencia
- a PAM 1 hibáit a PAM 250 a 250-szeresére nagyítja

BLOSUM

- későn lépett be a mátrixok közé
- nem feltételez evolúciós modellt
- a PROSITE-ból szerzett szekvencia-blokkok alapján
- sokkal nagyon és diverzebb szekvenciahalmazból (30%-90% sequence ID)
- **alacsonyabb** BLOSUM szám: ritkább hasonlóságokhoz jó
- **magasabb** BLOSUM szám: nagymértékű hasonlósághoz jó
- érzékeny a szerkezeti és funkcionális helyettesítésekre
- a BLOSUM hibái az illesztés hibáiból erednek

Gyakorlatban használt mátrixok:

- **PAM … :**
  - **40**: rövid, nagyban hasonló illesztésekhez
  - **120**: általános
  - **250**: távoli hasonlóságok kimutatásához
- **BLOSUM … :**
  - **90**: >90%-os sequence ID-val rendelkező BLOCKS szekvenciákból rövid, nagyban hasonló illesztésekhez
  - **62**: általános (alapértelmezett)
  - **30**: gyenge, lokális illesztéshez

	V	D	S	-	C	Y	
	V	E	T	L	C	F	
BLOSUM62	+4	+2	+1	-12	+9	+3	7
PAM30	+7	+2	0	-10	+10	+2	11

Nukleinsav mátrixok

	A	C	G	T
A	10	0	0	0
C	0	10	0	0
G	0	0	10	0
T	0	0	0	10

	A	C	G	T
A	10	-9	-9	-9
C	-9	10	-9	-9
G	-9	-9	10	-9
T	-9	-9	-9	10

- Needleman-Wunsch
- Smith-Waterman

1. az elemek magnitúdója egymáshoz relatív, tehát át lehet skálázni
2. már heurisztikus mátrixokat is könnyen létre lehet hozni, pl. diag(1, 1, …), többi 0, az identitásmátrix. Bizonyos párokra büntetést lehet kiszabni, ha nagy negatív értéket rendelünk hozzá stb.

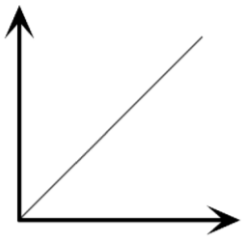
Dot plotok (pont ábra)

1. Az n hosszú (itt nukleotid) szekvenciát két merőleges tengelyre írjuk. Ez egy n×n-res mátrixot definiál, amit „dot matrix" vagy „alignment/illesztés mátrix" néven nevezünk.
2. Ahol az x(i) és y(i) nukleotidok egyeznek, oda teszünk egy pontot.
3. Ha a két szekv. egyezik, középen egy átlót kapunk, x(i)=y(i) teljesül az egész. szekvenciára.

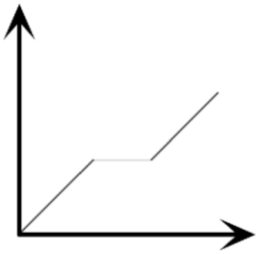
Paraméterek:

- **scoring scheme**: mátrix, pl: egyezésre=1, különbségre=0 (pl. EMBOSS: 4 és -5, szélesebb intervallumokat használva könnyebb bővíteni a mátrixot, ha pl ACGT mellé hozzá akarjuk venni U-t és más pontszámot rendelni neki)
- **word-size**: csak az ennél hosszabb match-ekre ad pöttyöt
- **cut-off score**: egy szavon belül mennyi egyezés kell ahhoz, hogy matchnek minősüljön

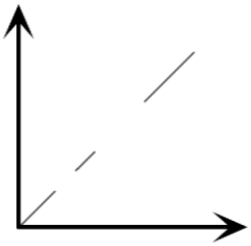
Egyforma:



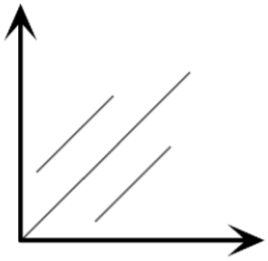
Deléció:



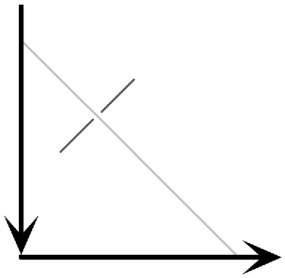
Hasonló régiók:



Ismétlődések:



Stem loops:



Vége a dotplotos résznek. Már csak 8 dia van hátra ☺