PÉCSI TUDOMÁNYEGYETEM

Biológia Doktori Iskola

Motivum keresés a humán promóterekben

Ph.D. értekezés

Nagy Tibor

Témavezető:
Dr. Barta Endre
tudományos főmunkatárs

Pécs, 2009

Tartalomjegyzék

Táblázatok jegyzéke

Ábrák jegyzéke

Rövidítések

API	(Application programmable interface) Alkalmazás fejlesztői felület
BLAST	(Basic Local Alignment Search Tool) Szekvenciakereső program
BRE	A TFIIB felismerőhelye
CAGE	Transzkripciós kezdőpont adatbázis
CGI	(Common gateway interface) Protokollszabvány a szerveren található programok futt
CPAN	Perl modulok gyűjtőhelye
DMSO	dimetil-szulfoxid
DPE	A TFIID felismerőhelye
DRA	Az általunk talált motívum munkaneve
DoOP	Ortológ promóter adatbázis
EMBOSS	Bioinformatikai programcsomag
ENCODE	A genom funkcionális elemeinek enciklopédiája
EnsEMBL	Genom annotációs adatbázis
GNU	A szabad szoftverek licencelésének neve.
GO	Gén ontológia
HSP	(High score pairs) Magas pontszámú szegmens párok
MCMC	Markov lánc Monte Carlo algoritmus
MEME	(Multiple EM for Motif Elicitation) Motívumkereső program
PIC	Preiniciációs komplex
PSWM	Pozícióspecifikus súlymátrix
RSG	Roziglitazon
TFBS	Transzkripciós faktor kötőhely
TSS	Transzkripciós kezdő pont
URS/UAS	Transzkripciót gátló illetve aktiváló elemek

1. fejezet

Bevezetés

A szekvenálási technológiák fejlődésével egyre több nyers adat került a kutatók kezébe. Egyfelől sok kérdésre kaptak választ, de ahogy mind jobban feltérképezték a különböző élőlények genomjait, úgy nőtt a megválaszolatlan kérdések száma is. Hány génje van az adott szervezetnek? Milyen folyamatok szabályozzák ezeket a géneket? A szekvenciában mely elemek felelősek a szabályozásért?

A számítástechnikai kapacitás növekedésének hála, a nyers adatok összegyűjtésével a korábban költséges laboratóriumi vizsgálatok egy részét olcsó és gyors programok futtatásával lehet szimulálni. Sajnos az algoritmusok még nem adnak olyan pontos válaszokat, mint a ,,nedves biológia" eszközei, de segíthetnek az erőforrások gazdaságosabb felhasználásában.

Új, nagy számú vizsgálat egyidejű lefolytatására alkalmas módszerek jelentek meg, tovább növelve a feldolgozásra váró adatok mennyiségét. A teljesség igénye nélkül ilyen módszerek például a különböző pipettázó robotok és a DNS chip technológia. Ez utóbbiak például lehetővé teszik, hogy egy bizonyos hatásra a genom valamennyi aktivitást mutató génjét megtaláljuk.

Az egyes gének vizsgálata háttérbe szorult és egyre nagyobb figyelem fordul a génszabályozás megismerésére. A génszabályozás kulcsa a promóterükben előforduló elemekben rejlik. Ezen motívumok feltérképezéséhez nagy számú promóter szekvenciáját kell átvizsgálni, valamint az egymáshoz fűződő viszonyaikból az áttekintést megkönnyítendő éredemes hálózatokat építeni.

Ahhoz, hogy ezeket az eredményeket gyorsan ki lehessen értékelni, szintén a számítástechnika és a statisztika nyújt segítséget. Ha ezekkel az eszközökkel felfegyverkezünk, csatába indulhatunk, hogy megfejthessük a genomok titkait.

A Mezőgazdasági Biotechnológiai Kutatóközpont Bioinformatika csoportja számos ko-

operációs partnerrel próbálta kideríteni, hogy egy gén promóterében található motívumok milyen kapcsolatban állnak az adott gén élettani szerepével. A kérdés általános volta miatt nem volt elég egyetlen kísérletsorozat, hogy érdemi következtetéseket lehessen levonni. Dolgozatomban több megközelítést is bemutatok, melyek mindegyike kicsit közelebb vitt a válaszhoz.

2. fejezet

Irodalmi áttekintés

2.1. Eukarióta szabályozó régió felépítése és működése

A transzkripció során a DNS bázissorrendje alapján RNS szintetizálódik. Az RNS molekula típusa szerint lehet hírvivő mRNS, riboszómális rRNS, aminosav szállító tRNS. A fehérjekódoló gének mRNS-el adják tovább információ tartalmukat. Az mRNS szintézis során az RNS polimeráz II nevű enzim a DNS-hez kötődik. Azt a pozíciót, ahonnan az mRNS szintézis elkezdődik transzkripciós start helynek (TSS) nevezzük. Eukarióta sejtekben ez a pozíció nem korlátozódik egy abszolút pontra.

A transzkripció kezdőpontjától 5' irányban (upstream) helyezkedik el a szabályozó régió, idegen szóval promóter. Az itt található transzkripciós faktor kötőhelyek befolyásolják a génexpressziót. A szabályozó régió pontos határai nem ismertek, de a TSS-t megelőző és követő első 50 bázispárt mag- (core promóter), a távolabbiakat proximális (1-2 kbp) illetve disztális (ha 2 kbp-nál nagyobb távolságra található) szabályozó régiónak nevezik. Általánosságban elmondhatjuk, hogy a magpromóterben és sokszor a proximális promóterben található elemek felelősek az alap transzkripciós szerkezet működéséért, például a polimeráz II enzimet is tartalmazó preiniciációs komplex (Preinitiation complex, PIC) a DNS-hez kötődve megközelítőleg a magpromóter régiót fedi le. A távolabbi, akár 100 kilobázis nagyságrendű távolságban lévő transzkripciós faktor kötőhelyek (transcription factor binding site, TFBS) pedig az egyedfejlődéssel kapcsolatos és szövetspecifikus finomszabályozásért felelősek inkább. A TSS-től 3' (downstream) irányba eső pozíciókat pozitív, míg az 5' (upstream), tehát promóter régióba eső elemeket negatív előjellel számozzuk.

A transzkripciót, és így az RNS polimeráz kötődését számos molekula kapcsolódása

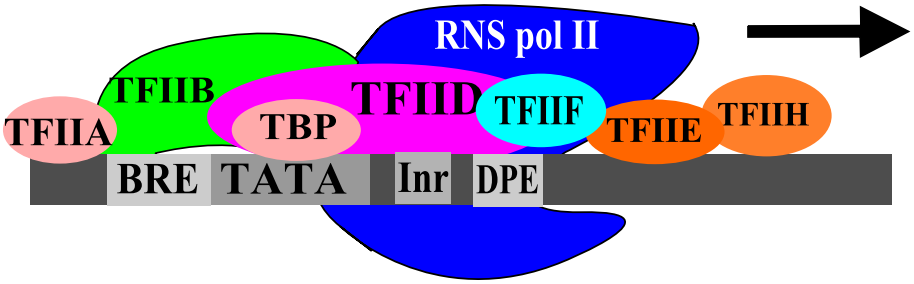
előzi meg. Ezeket közös néven transzkripciós faktoroknak nevezik. Két nagyobb csoportba sorolhatóak: az általános transzkripciós faktorok minden gén átírásához szükségesek, de nem az összes. Ezek a TFIIA, TFIIB, TFIID, TFIIE, TFIIF, TFIIH és az Srb/Mediátor komplex. A speciális transzkripciós faktorok csak kis számú gén átírásában segédkeznek. Azokat a szekvenciákat, melyek a transzkripciós faktorok DNS-hez kötődését segítik, transzkripciós faktor kötőhelynek nevezik.

Az első azonosított eukarióta transzkripciós faktor kötőhely a TATA-box. Többsejtű élőlényekben a kötőhely a TSS-től 25-30 nukeotid távolságra van, de *Saccharomyces cerevisiae* esetében a pozíció változékonyabb. Habár a prokarióták is rendelkeznek egy hasonló konszenzus szekvenciájú promóter elemmel, az ott található Pribnow-box a legújabb kutatások szerint nem homológja az eukarióta TATA-boxnak (Butler and Kadonaga, 2002). A szakirodalom kezdetben 25 százalékra tette a TATA-box-al rendelkező szabályozó régiók arányát (Wray et al., 2003), de az újabb publikációk ennél alacsonyabb számot állapítottak meg.

A TATA-box, elsősorban a RNS polimeráz II által átírt gének szabályozó régiójában fordul elő. Nevét rendkívül konzervált konszenzus szekvenciájáról kapta (TATAAA). Későbbi vizsgálatok azonban feltárták, hogy *in vivo* körülmények között több más szekvencia is működhet TATA-boxként (Singer et al., 1990). Fő feladata a transzkripció kezdőpontjának kijelölése. Ezt támasztja alá, hogy ezen gének transzkripciója a kísérletek tanúsága szerint egy meghatározott pontból indul.

Erre a szekvenciára kötődik a TATA kötő fehérje (TATA Binding Protein - TBP), a legősibb transzkripciós faktor. A kötés hatására a DNS két szála eltávolodik egymástól, lehetővé téve, hogy az RNS polimeráz könnyebben kapcsolódjon. Míg prokariótákban egyedül látja el feladatát, addig az eukarióta sejtekben a TFIID (transzkripciós faktor II D) komplex részeként. A létrejött DNS-TBP kapcsolatot a TFIIA (Transzkripciós faktor II A) stabilizálja. A TFIID komplex tartalmaz számos TBP kapcsolt faktort (TBP-associated factors - TAFIIS). A TAFIIS-ok is felismernek számos elemet a szabályozó régióban, akár TBP hiányában is. Ennek következtében TATA-box nélkül is képesek a TFIID komplex feladatának ellátására. Elmondható tehát, hogy a TAFIIS a szabályozó régió felismerésében játszik szerepet, nem a transzkripció szabályozásában.

A TFIID az úgynevezett DPE (downstream core promoter element) helyet ismeri fel. Ez a hely a később ismertetésre kerülő Inr-től 3' irányban található a +28 és +33 pozíciók között. Konzervált szekvenciája a *Drosophila*tól az emberig megtalálható. Érdekes módon *Saccharomyces cerevisiae*-ben nem ismert (Juven-Gershon and Kadonaga, 2010). *Drosophila*ban több kísérletes vizsgálat célpontja, de emberben mindössze egy génben ta-



2.1. ábra. A preiniciációs komplex felépítése

nulmányozták (Gershenzon and Ioshikhes, 2005).

A TFIIB szerepe, hogy kijelölje a transzkripció kezdőpontját. A TFIIF stabilizálja a preiniciációs komplexet. A TFIIE és TFIIH foszfát csoportot hasít le az RNS polimeráz CTD (karboxy-terminál ismétlődő domain) alegységéről, amitől annak térszerkezete megváltozik és az enzim megkezdi a transzkripciót (Lee and Young, 2000).

Az általános transzkripciós faktorok és az RNS polimeráz alkotják a preiniciációs komplexet (PIC) (2.1 ábra). Az Srb/Mediátor komplex képes stimulálni a TFIIH foszforilációját *in vivo* körülmények között, így valószínűleg szerepe van az RNS polimeráz iniciációs és elongációs formájának átalakulásában. Habár az Srb/Mediátor komplex lényeges része a preiniciációs komplexnek, egyes vizsgálatok találtak olyan enzim formát is, melynél a Mediátor komplex az alegységek nagy részét nem tartalmazza (Lee and Young, 2000).

Az Srb/Mediátor komplex a transzkripcióban hasonlóan jelentős szereppel bír, mint az RNS polimeráz II. Hidat képez a polimeráz és az aktivátor proteinek között, így elmondhatjuk, hogy a szabályozás üzenete az enhanszerektől a polimerázig ezen komplexen keresztül halad. Nem csak a pozitív, de a negatív regulációban is szerepe van. Nevével ellentétben tehát nem csak koaktivátor, de korepresszor és bazálist transzkripciós faktor is egyben (Kornberg, 2007).

A TATA-box mellett a szabályozó régióban megtalálható még egy úgynevezett iniciátor elem (Inr). Az Inr jelöli ki a transzkripció kezdőpontját. Konszenzus szekvenciája ember esetében Py-Py-A-N-T/A-Py-Py, ahol a Py tetszőleges pirimidin bázist jelölhet, míg az N bármely nukleotidot. Elhelyezkedése a -2 - +4 pozícióban van. Elsősorban a TFIID kötődik ide, de *in vitro* körülmények között az RNS polimeráz II önmagában is képes felismerni (Butler and Kadonaga, 2002).

A TFIIB-nek is ismert a kötőhelye. BRE-nek nevezik, elhelyezkedése a TATA-boxtól 5' irányba található, szinte egybeolvad azzal. Konszenzus szekvenciája G/C-G/C-G/A-C-

G-C-C. Szerepe nem teljesen tisztázott. Egyes vizsgálatok szerint segít a működő transzkripciós iniciációs komplex kialakulásában, míg más kutatók szerint a bazális transzkripcióra negatív hatással van (Evans et al., 2001).

A TATA-boxot és Inr elemet nélkülöző szabályozó régiók magas GC-tartalmú helyekkel rendelkeznek, amelyeket CpG szigeteknek neveznek. A CpG szigetek is befolyásolják a transzkripció helyét, de ezek a TATA-boxal ellentétben nem egy jól meghatározott helyről indulnak (Gustincich et al., 2006). Méretük 0,5-től 2 kbp-ig terjedhet, és több, gyenge magpromótert tartalmazhatnak. Nagy számú Sp1 kötőhely fordul elő bennük. Az Sp1 egy általános sejtfolyamat szabályozó fehérje, ami szerepet játszik a sejt növekedésében, differenciációjában és az apoptózisban. Az Sp1 és homológjai tartalmaznak egy C_2H_2-típusú cink ujjat, amely lehetővé teszi, hogy a GC dús szekvenciákhoz kapcsolódjanak. Az Sp1-ről amiről kiderítették, hogy TATA-box hiányában és Inr jelenlétében képes aktiválni a transzkripciót. Egereknél a normális embriogenezis nélkülözhetetlen szereplője Shen et al..

A promóteren kívül is találhatóak szabályozó elemek, melyek befolyásolják a gének expressziós szintjét. Ezek az upstream aktivátor szekvenciák (UAS), enhanszerek, upstream represszáló szekvenciák (URS) és a gén csendesítők (silencer).

Az upstream aktivátor szekvenciák transzkripciós aktivátorokat kötnek meg a transzkripciós kezdőpont közelében. Az enhanszerek olyan DNS kötő régiók, melyek 85 kbp-nál nagyobb távolságra találhatóak a kezdőponttól és orientációjuk független az általuk szabályozott transzkripció irányától. Az ide kötődő fehérjék hatására a DNS térszerkezet változást szenved, aminek következtében az iniciációs komplexhez nagyobb távolságra található faktorok is kapcsolódhatnak. A további faktorok hatására a gén expressziója a bazális expressziós szint többszörösére növekedhet.

Az upstream represszáló szekvenciákhoz kötődő fehérjefaktorok a transzkripciót többféle módon gátolhatják. Módosíthatják a kromatin struktúrát, megakadályozzák az aktivátorok kötődését, esetleg gátolják a transzkripciós apparátus létrejöttét.

A gén csendesítők elnyomják a promóter aktivitást orientációtól és távolságtól függetlenül. Hatásukra vagy proteinek kötődnek a szabályozó régióra, megelőzve a transzkripciót aktiváló elemek kötődését vagy a hiszton burkot módosítják ugyanezen cél érdekében. Magasabb rendű eukariótákban a DNS metilációhoz kötött csendesítésben a CpG dinukleotidok jutnak szerephez. Az enhanszerek és gén csendesítők egyszerre több gén expresszióját is képesek befolyásolni. A határoló elemek (inszulátorok) szerepe az, hogy adott esetben megakadályozzák, hogy a gén ezen elemek hatása alá kerüljön. A határoló elem csak abban az esetben tudja befolyásolni a gén expresszióját, ha pozíciója

az enhanszer és a szabályozandó gén közé esik. Feladatukat a hiszton fehérje konformáció módosításán keresztül látják el. Ezt támasztja alá, hogy ecetmuslincán végzett vizsgálatok azt mutatják, hogy a határoló elemek DNáz I hiperszenzitív helyeket tartalmaznak (Gerasimova and Corces, 2001). Ugyanakkor egyes források azt feltételezik, hogy az inszulátorok a szabályozó régiókhoz hasonló struktúrák. Ez esetben az enhanszer nem a promóterhez kötődik, hanem a határoló elemhez, ezzel mintegy lefoglalva azt és megakadályozva, hogy a szabályozó régióra fejtse ki hatását (Raab and Kamakaka, 2010).

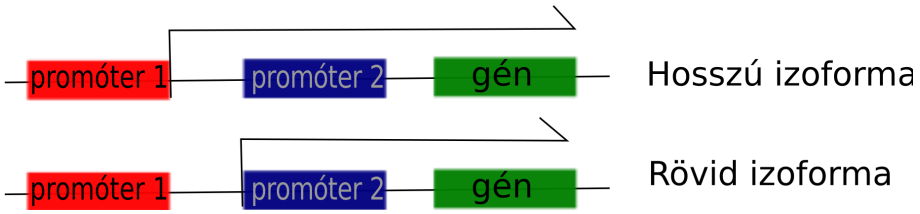
Az eddigi példák is szemléltetik, hogy a hiszton fehérjék fontos szerepet játszanak a transzkripció szabályozásában. A hiszton burok módosítására két folyamat alakult ki. Az első a kromatin átrendező faktor (chromatin remodeling factor), amely a hiszton fehérjéket képes mozgatni a DNS körül. Két nagyobb családba sorolhatjuk őket, az SWI/SNF-be és az ISWI-be. A másik enzimcsoport a hiszton fehérjék oldalláncaihoz kapcsol különböző (acetil, metil, foszfát, stb.) oldalláncokkal, melyek befolyásolják a hiszton-DNS vagy hiszton-hiszton kapcsolatokat és ezen keresztül a transzkripciós faktorok DNS-hez kötődését is.

Emlősökben az SWI/SNF-nek nincs szekvencia specifikus DNS kötő aktivitása, a komplex mégis képes kapcsolódni a nukleoszómához, sőt *in vitro* kísérletek alapján azt is tudjuk, hogy akár a csupasz DNS-hez is. Több szteroid receptor is képes hatást gyakorolni az SWI/SNF egységeire. A hiszton módosításához szükséges energiát ATP-áz aktivitása segítségével teremti elő.

Az eukarióta génszabályozás eddig bemutatott, cseppet sem egyszerű képét az alternatív promóterek tovább bonyolítják. Genom szintű vizsgálatok azt mutatják, hogy az emlős gének 20-30%-a rendelkezik alternatív szabályozó régióval (Davuluri et al., 2008b). Ezek a szabályozó régiók alternatív első exonok előtt találhatóak, segítségükkel a génszabályozás szöveti és időbeli dimenziókat kaphat. Például a HBG1 gén két szabályozó régiója közül az egyik tartalmaz TATA-boxot, míg a másikból ez hiányzik. Ez lehetővé teszi, hogy az embrionális fejlődés alatt és után eltérő szabályozás alá essen a gén.

Az alternatív szabályozó régiók megléte nem jelenti feltétlenül eltérő szerkezetű fehérjék transzlációját. Az *OTX2* gén esetében például csak az mRNS 5' UTR-ben van eltérés, ami a szövetspecifikus kifejeződést befolyásolja. Ha az alternatív promóter intronba esik, az introntól 5' irányba található exonok nem íródnak át, ezáltal a keletkezett géntermék eltérő funkcióval fog rendelkezni. Például az *LEF1* gén két szabályozó régióval rendelkezik. Az egyik egy hosszabb, teljes értékű fehérjét ír át, ami képes aktiválni más géneket, míg a rövidebb nem, ezáltal gátolja azok expresszióját (2.2 ábra).

Egyes betegségek kapcsolatot mutatnak az alternatív szabályozó régiók rendellenes



2.2. ábra. LEF1 gén szabályozó régiói

működésével. A korábban említett *LEF1* esetében például megfigyelték, hogy tumorsejtekben egyedül az 5' szabályozó régió mutat aktivitást. Normál sejtben vagy csak a 3' promóter aktív, vagy mindkettő. Ez utóbbi esetben a 3' promóter gátló hatást fejt ki az 5' géntermékre.

Az alternatív szabályozó régió kapcsolásában valószínűleg a hiszton acetiláció és promóter metiláció játszik szerepet. Erre utal, hogy a *TGFB3* gén proximális szabályozó régiójában a metiláció hiánya összefüggést mutat az emlőrák sejtekben mérhető aktivitásával (Davuluri et al., 2008a).

Az eddig bemutatottak alapján a génszabályozás egy rendkívül összetett folyamat. Nehéz egy általános képet lefesteni róla, mert bármilyen szabályszerűséget is mutatnak ki a vizsgálatok a folyamat egyes szereplőiről, szinte azonnal akad rá ellenpélda.

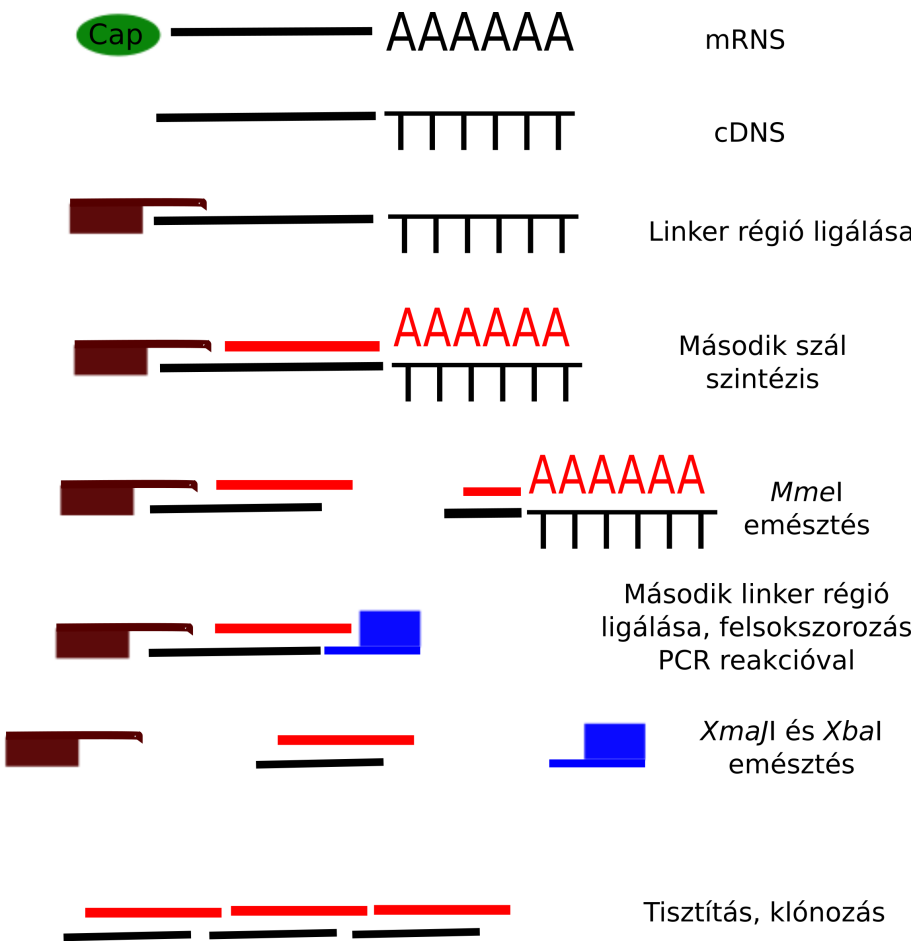
2.2. A szabályozó régió vizsgálati módszerei

2.2.1. Transzkripciós kezdőpont meghatározása

A transzkripció kezdőpontjának helyét kísérletesen a gén expressziós cap analízissel (CAGE) határozzák meg. A szöveti sejtmintákból izolált mRNS szálakról cDNS-t szintetizálnak reverz transzkriptáz enzim segítségével. Kiválasztják a cap szekvenciával rendelkezőket, amelyekhez egy linker régiót kapcsolnak. A cap szekvenciát főként egy módosult guanin nukleotid alkotja, ami a többek között a riboszóma kötésben is szerepet játszik, tehát csak a fehérjét kódoló mRNS-ek tartalmazzák. A linker régió *MmeI*, *XmaJI* felismerő helyeket tartalmaz, valamint biotint a nem ligálandó végén.

Az *MmeI* hasítás nem a felismerőhelyen, hanem attól 20-22 bázispár távolságra történik. A hasított végre egy második linker régiót ligálnak, ami ugyancsak tartalmaz *XmaJI* hasítóhelyet és biotin véget.

Ezután az *MmeI* felismerőhelyet tartalmazó régiót PCR segítségével megsokszorozzák. Az *XmaJI* enzimmel eltávolítják a linker régiókat. Az így nyert különböző CAGE darabo-



2.3. ábra. A CAGE darabok szintézisének folyamata

kat ligáz enzim felhasználásával összeszerelik,vektorba építik, végül meghatározzák a szekvencia sorrendjét (2.3 ábra). A meghatározott szekvenciákból bioinformatikai eljárásokkal a genomra térképezik az egyes darabokat (Shiraki et al., 2003).

2.2.2. Aktív szabályozó régiók felderítése

A legegyszerűbb módszer, hogy megtudjuk, a vizsgált szabályozó régió aktív-e, ha azonosítjuk a génről átíródó RNS-ket. A korábbi Southern- és Northern-blot eljárást a chip technikák váltották fel, míg napjainkban az új generációs szekvenálásokon alapuló módszerek terjednek.

A microarray vagy chip technikák alapja, hogy a vizsgált szövetből izolálják az összes mRNS-t, majd fluoreszcens festékkel vagy radioaktív módszerrel jelölt cDNS-sé írják át. Ezt hívják mintának. A mintákat hibridizálják a próbához, ami egy felületre rögzített egyszálú cDNS, ismert a bázissorrenddel.

A felület többek között lehet üveg, nylon, szilikon, nitrocellulóz. A próbák egy része olyan nukleotidokból áll, melyekkel a kiértékeléseket segítik. Mivel a hibridizáció során hibák léphetnek fel, ezért ugyanazon próbák a felület több, különböző pontján is előfordulnak. Ezáltal egy chipen belül megvalósul bizonyos számú ismétlés is. Ennek száma a chip gyártójától függ.

A minták száma alapján két nagy csoportba sorolhatóak a chipek. Az első csoportba tartozó chipekre csak egy minta helyezhető el, ezeket oliginukleotid chipeknek nevezik és az Affimetrix tervezi, illetve árusítja. Megközelítőleg 1.5 x 10^5 darab 25 bázispár hosszú nukleotidot helyeznek el a felületen. A gyártás során egy maszkkal letakarják a felületet, így a kívánt bázispárok csak a maszk által szabadon hagyott területre épülhetnek. Az eljárás előnye, hogy az elhelyezett próbák sűrűsége nagyobb, mint a robottal előállított chipeknél. A nukleotidokat fotolitografikus eljárással rögzítik. Az elsőt az üres felületre, majd a további nukleotidokat az előző tetejére. Az oligonukleotidok ily módon történő szintézise erősen párhuzamosított. A vizsgált és kontroll minták ugyanazon festékkel vannak jelölve, ezért egy chip csak egy mérésre szolgál. A kiértékelésnél ezért szükség van chipek közötti normalizálásra is (Kohane et al., 2003).

Mivel 25 bázispár nem elég hosszú, hogy egyértelműen azonosítson egy gént, ezért egy trükköt alkalmaznak. Nem csak tökéletesen egyező oligonukleotidok vannak a chipen, hanem úgynevezett mismatch próbák is, amelyek a 13. nukleotidban eltérnek. A statisztikai elemzéshez felhasználják ezeket is. Egyéni chipek előállítása körülményes ezzel a módszerrel.

A robot által előállított chipeknél ezzel szemben előre szintetizálják az oligonukleotidokat és azokat egy robot helyezi el a felszínen. A módszer időigényesebb, viszont lehetővé teszi, hogy a kísérlet szükségleteihez igazítsák a chip paramétereit.

A hibridizáció során felhasznált jelölő festék száma alapján elkülöníthetünk egy vagy két csatornás chipeket. A két csatornás chipeknél Cy3 és Cy5 festéket használnak, lehetővé téve, hogy nem csak a vizsgált mintákat, hanem a kontrollt is ugyanazon a chipen vizsgálják. Ahol mindkét jelölő festékkel ellátott minta hibridizál, ott a gép a két szín keverékét olvassa le (Kohane et al., 2003).

2.2.3. Fehérje kötőhelyek azonosítása kísérletes módszerekkel

A kötőhelyek vizsgálatának legegyszerűbb módszere a mutációs analízis. A szekvencia különböző pontjain mutációkat indukálnak, és vizsgálják ennek hatását a transzkripcióra. Ha a transzkripció elindul a mutáció ellenére is, az expressziós szintben nem áll be változás, akkor az adott pozícióra nem kötődik a transzkripcióért felelős fehérje, vagy

a vizsgált kötés nem szekvencia specifikus.

Kromatin immunprecipitációval egybekötött chip kísérlettel (chip-on-chip) a specifikus fehérje kötés vizsgálata genom szintre terjeszthető ki. A módszer során a vizsgálni kívánt fehérje ellen ellenanyagot termelnek, amit ezen felül ellátnak egy markerrel a későbbi detektálhatóság végett. Miután a fehérje kapcsolódott az örökítő anyaghoz, a kötést formaldehiddel stabilizálják. Ezután a DNS-t ultrahang segítségével feldarabolják 0,2-1 kbp méretű darabokra. A fehérje-DNS kapcsolatokat az ellenanyag segítségével elválasztják a nem jelölt DNS-től, majd leválasztják a fehérjéket is. Végezetül az így kinyert szekvenciákat a korábban bemutatott chip módszerrel tovább vizsgálják. A különbség annyi, hogy a próbák nem csak géneket tartalmazhatnak, hanem kromoszómákat, CpG szigeteket, vagy az ENCODE régiót (Carter and Vetrie, 2004). (Az ENCODE régióról a 2.4.4 fejezetben teszek említést.)

Az adatokat is másképp kell értelmezni kromatin immunprecipitáció esetén. Az eredményül kapott intenzitásokból kell meghatározni, hogy az adott genomi pozícióra kötődött-e fehérje.

Újabban a különböző chip kísérleteket felváltották az új generációs szekvenálási eljárások (chip-seq, rna-seq). Ezek az előnye, hogy a vizsgálatok immár az egész genomra kiterjeszthetőek.

2.2.4. Fehérjekötő motívumok azonosítása bioinformatikai módszerekkel

A motívumok biológiai szempontból olyan rövid DNS vagy fehérje szekvencia elemek, melyek biológiai szereppel bírnak. Ez a szerep lehet például a fehérjekötő képesség. Bioinformatikai szempontból a motívumok az ortológ vagy paralóg szekvenciák hasonló szakaszai. Bármelyik definíciót is használjuk, szükségünk van olyan jelölésekre, melyek segítségével érzékelhetővé tehetjük az egyes pozíciókban előforduló alternatív nukleotidokat.

Egyik jelölési módszer a konszenzus szekvencia. Ha úgy képzeljük el a motívumokat, mint egy többszörös illesztést, akkor az egyes konszenzus szekvencia pozíciókba írhatjuk azokat a bázisokat, melyek mindegyik szekvenciában megtalálhatóak, amelyek részt vesznek az illesztésben. Amennyiben több bázis is előfordulhat egy pozícióban, akkor az egyezményes IUPAC jelöléssel tudatjuk, hogy milyen nukleotid kombinációk fordulhatnak elő egy pozícióban (2.1. táblázat). A konszenzus szekvencia alkalmazásának előnye, hogy az ember számára könnyebben értelmezhető. Hátránya, hogy nem ad elegendő információt az adott pozícióban milyen arányban fordulhatnak elő alternatív bázisok.

R	adenin vagy guanin
Y	citozin vagy timin
S	guanin vagy citozin
W	adenin vagy timin
K	guanin vagy timin
M	adenin vagy citozin
B	citozin, guanin vagy timin
D	adenin, guanin vagy timin
H	adenin, citozin vagy timin
V	adenin, citozin vagy guanin
N	bármilyen nukleotid

2.1. táblázat. IUPAC jelölések nem egyértelmű nukleotidok esetén

(Schneider, 2002).

Ennél fejlettebb a mátrixos megközelítés, ahol számszerűen megadható, hogy az egyes pozíciókban milyen a bázisok gyakorisága. Szokás pozíció specifikus súlymátrixnak (PSWM) is nevezni ezt a fajta leírást. Ennek a módszernek a hátránya, hogy két különböző motívum összehasonlítása csak komplex matematikai formulákkal lehetséges (Stormo, 2000).

A súlymátrixok lényeges jellemzője az információs tartalom (information content), ami azt mutatja meg, mennyire tér el a mátrix az egyenletes eloszlástól. A számításnál figyelembe kell venni a célszekvencia (esetünkben a szabályozó régió) báziseloszlását, ugyanis egy magas GC tartalommal bíró szekvenciában az AT tartalmú motívumoknak nagyobb lesz az információs tartalma, mint egy AT túlsúlyt mutató szekvencia esetében. Az információs tartalom grafikus megjelenítése az úgynevezett szekvencia logo. Az egyes bázisok túlsúlya informatívabb a kutatók számára ezzel a módszerrel (Schneider and Stephens, 1990).

Elsősorban fehérjék esetében létezik még a motívumoknak egy matematikai leírása is, a rejtett Markov-modell. A modell lényege, hogy van több megfigyelésünk (nukleotid szekvencia sorrend) és néhány rejtett állapotunk (az adott oligonukleotid transzkripciós kötőhely vagy nem, úgynevezett háttér). Azért rejtett, mert nem tudjuk eldönteni, hogy mi a szerepe. Ha ismerjük annak a valószínűségét, hogy az egyes rejtett állapotok milyen valószínűséggel váltakoznak egy szekvencián bellül, valamint, hogy a megfigyeléseink milyen valószínűséggel feleltethetőek meg a rejtett állapotoknak, akkor kiszámíthatjuk,

hogy a megfigyelt nukleotid sorrend transzkripciós faktor kötő helye-e.

Amennyiben az összes rejtett állapot összes átmenetének valószínűsége ismert, teljes Markov láncról beszélünk. A modell megalkotói azt remélik, hogy ennek segítségével képesek lesznek megfejteni a szabályozó régió nyelvezetét (Won et al., 2008).

A motívumok bioinformatikai módszerekkel történő felderítésének három módszere ismert. Az első, ismert motívumok azonosítása a szekvenciákban. A második módszer felülreprezentált mintázatokat különít el a megadott szekvencia részletekben. Ezt statisztikai vagy kombinatorikus számítások segítségével érik el. Az utolsó módszerben egyéb információk felhasználásával találnak motívumokat.

Ismert motívumok felderítésére fejlesztett programok egy adott motívum készlet alapján megkeresik annak összes előfordulását a megadott szekvenciákban. A módszer hátránya, hogy új motívum leírására nem alkalmas.

A statisztikai megközelítést használó motívum felderítő alkalmazások Gibbs mintavételezést, elvárás maximalizálást vagy rejtett Markov láncot használnak.

A Markov lánc Monte Carlo (MCMC) algoritmus család alapja, hogy véletlenszerű valószínűségi változók eltérését vizsgálja ismert eloszláshoz viszonyítva. A módszer előnye, hogy komplex, több dimenziós integrálokat képes közelíteni, megspórolva a kiszámításukat.

A Gibbs mintavételező algoritmus egy iteratív mintavételezésen alapuló statisztikai módszer (Liu et al., 1995).Ennek az algoritmusnak DNS szekvencia illesztésekre optimalizált változatát használja az `AlignACE` (Roth et al., 1998). A megközelítés lényege, hogy mindegyik beadott szekvenciákban egyetlen motívumot azonosít, míg a szekvencia többi része úgynevezett háttér adat. A szekvenciákban fellelhető mintázatok (oligonukleotidok) közül kiválasztja azt, amelyik pontszáma maximálisan meghaladja a háttér pontszámát. Ez a maximalizálás NP-nehéz probléma, ami azt jelenti, hogy a megoldásához szükséges gépidő nem csökkenthető jelentősen erősebb hardver alkalmazásával. Ezért a Gibbs mintavételezés véletlenszerűen választja ki a vizsgált mintázatokat. A gyakorlatban ez azt jelenti, hogy a program minden futtatás során más eredményt fog adni (Lawrence et al., 1993).

A `MEME` algoritmusa ezzel ellentétben egy elvárás maximalizáló módszert használ, hogy behatároljon egy vagy több ismétlődő motívumot. A `MEME` minden motívumhoz rendel egy diszkrét valószínűségi eloszlásokat tartalmazó mátrixot, majd a rendelkezésre álló adatok alapján maximalizálja ezen modellek poszterior valószínűségét. A maximalizálási lépés során itt is háttérnek tekinti a motívumot nem tartalmazó szekvencia részeket.

A motívumok felderítésére számos számítógépes algoritmus létezik. Az egyik legis-

mertebb a `MEME`, ami hasonló DNS szekvenciák csoportjából, egy véges kevert modell segítségével képes megtalálni a feldúsuló szekvencia mintázatokat (Bailey and Elkan, 1994).

A `MEME`-hez hasonló, ám annál jóval hatékonyabb program a `NestedMica`. Ez a Java nyelven írt alkalmazás nem csak a motívumokat találja meg, hanem képes figyelembe venni a beadott szekvenciák evolúciós távolságát is, tehát a konzerválódott elemeket súlyozza. Algoritmusa sokkal érzékenyebb, mint a `MEME` (Down and Hubbard, 2005).

A kombinatorikus megközelítés elsősorban konszenzus szekvenciák esetében használhatóak, de megengedik a báziscserét. Az algoritmus itt is NP-nehéz, ezért több módszert is kidolgoztak, amivel közelítik az ideális esetet. Az egyik ilyen pattern alapú algoritmusok, ahol egy előre legenerált mintázatot keres, amely az összes megadott szekvenciában megtalálható a legkevesebb báziscserével. Ez a módszer hosszú motívumok esetén lassú. A minta alapú megközelítés ezzel szemben megadott hosszúságú mintákra bont egy szekvenciát a készletből és megvizsgálja, hogy a készlet többi szeknveciájában megtalálható-e ez a minta a megadott maximális báziscser mellett. A módszer hátránya, hogy ha a kiindulási szekvencia nem tartalmazza a motívumot vagy a motívum ,,gyenge", akkor a keresés eredménytelen lesz (Thijs et al., 2001).

A motívumok felderítésének másik módja az evolúciósan konzerválódott szakaszok azonosítása a promóterben. A módszer nehézségét az adja, hogy minél nagyobb az evolúciós távolság, annál kisebb a régiók hasonlósága. A másik problémát a genomi átrendeződések adják. A génduplikációkkal létrejövő paralóg szekvenciák megkülönböztetése az igazi ortológoktól kizárólag bioinformatikai módszerekkel közel lehetetlen.

2.3. Szekvencia összehasonlítás bioinformatikai módszerekkel

2.3.1. Fasta

A FASTA egy gyors szekvenciaillesztő alkalmazás. Gyorsasága abban rejlik, hogy ahelyett, hogy a teljes szekvenciát összehasonlítaná, részszekvenciákat keres, amelyeket ,,k-tuple-nek" vagy ,,szavaknak" neveznek. A közös k-tuple-ek segítségével próbál meg lokális illesztést végezni. Az algoritmus kevésbé érzékeny, mint a Smith-Waterman algoritmus, de annál jóval gyorsabb. Sebessége mégsem éri el a következő fejezetben bemutatásra kerülő Blastot.

Gyorsaságának egyik kulcsa egy hasító tábla, mely tartalmazza minden k-tuple-nek az

előfordulását. A k-tuple-ök relatív távolságából határozza meg az algoritmus az illesztést. A hasítótábla segítségével az összehasonlítás sebessége lineáris a szekvenciák hosszával, szemben a Smith-Waterman négyzetes arányával.

Az algoritmus igen népszerű volt adatbázissal szembeni keresések esetén, mert a hasítótáblát előre el lehetett készíteni. Később a BLAST elterjedésével használata visszaszorult.

2.3.2. Blast

A BLAST algoritmus gyorsabb a FastA-nál, miközben az érzékenyéséből veszít hozzá képest. Manapság ez a legelterjedtebb szekvencia kereső alkalmazás. Első (BLAST1) verziója még hézagmentes illesztésre volt képes. A második verziója - melyet két intézet egymástól függetlenül fejlesztett ki - már képes a hézagos illesztésre, miközben a keresés sebességét tovább növelték. A két verzió közül az NCBI-BLAST az elterjedtebb, míg a WU-BLAST használata háttérbe szorult.

A működés megértéséhez vezessünk be néhány új fogalmat. A kereső, és az adatbázis szekvenciát azonos méretű darabokra bontja a program, amelyet szegmens pároknak neveznek. A szegmens párok hasonlóságát az őket felépítő nukleotidok hasonlósági számainak összege adja. Aminosavaknál a PAM vagy BLOSUM mátrixot használják, míg DNS esetén a BLAST mérőszámot. Ha a szegmens párok hosszúságát kiterjesztjük és további egyezést kapunk, a hasonlóság mérőszáma növekszik. A terminológia ezt HSP-nek nevezi. Amennyiben nem tudjuk tovább növelni a szegmens párok méretét, hogy magasabb mérőszámot kapjunk, maximális szegmens pároknak nevezzük őket (MSP). Ha az MSP mérőszáma magasabb egy küszöb értéknél, a BLAST hasonlónak fogadja el a kereső szekvenciát és az adatbázisban talált szekvenciát.

Első lépésként a program készít egy kereső táblát, ahol a beadott szekvenciát felbontja kisebb darabokra és elkészíti a lehetséges eltéréseket is ettől a szekvenciától, amit még megenged a felhasználó. A kisebb darabok mérete szintén állítható.

Ezekt után az algoritmus végignézi az adatbázist, egyező párok után kutatva. Amennyiben talál ilyet, tárolja a párok helyzetét a szekvenciákon belül. A találatokat ezek után kiterjeszti mindkét irányba addig, amíg a pontszám nem csökken, más szavakkal meghatározza a HSP-t. Több HSP felhasználásával tovább növeli a kiterjesztést, miközben folyamatosan újra számolja az illesztés szignifikanciáját. Végezetül egy módosított Smith-Waterman algoritmus segítségével meghatározza a hézagok helyzetét (Sung, 2010).

A találatok értékeléséhez két statisztikai mérőszámot bocsát a BLAST rendelkezésünkre. Az első az E-érték, ami azon illesztések számának várható értéke, melyek

pontszáma magasabb egy megadott küszöbértéknél. A gyakorlatban a 10-nél magasabb E-értéket a program ki sem írja (Altschul et al., 1997).

A második mérőszám a bit pontszám (bit score). Bevezetését az tette szükségessé, hogy az illesztés mérőszáma erőteljesen függ a szekvenciák hosszúságától és a felhasznált mátrixoktól. A bit pontszám egy normalizált mérőszám, ezért független a korábban említett hatásoktól.

A BLAST algoritmusnak több formája is létezik. Nukleotid szekvenciák összehasonlítására a BlastN programot használják. BlastP a fehérje szekvenciák keresésére való. Mivel a fehérje szekvenciák keresése jóval pontosabb találatot tesz lehetővé, mint a nukleotid összehasonlítás, ezért megalkották a BlastX-et, ami nukleotid szekvenciát keres fehérje adatbázisban úgy, hogy lefordítja a kereső szekvenciát mind a hat lehetséges leolvasási keretbe. A Tblastn a BlastX fordítottjának tekinthető. Fehérjével keres nukleotid adatbázisban. Ebben az esetben az adatbázist fordítja fehérjévé. A Tblastx abban különbözik a BlastN-től, hogy mind a kereső nukleotid szekvenciát, mind a nukleotid adatbázist fehérjévé fordítja.

A PSI-BLAST egy pozícióspecifikus mátrix segítségével hajtja végre a keresést, ami nagyobb érzékenységet tesz lehetővé biológiailag releváns, de alacsony homológiát mutató szekvenciáknál a pusztán nukleotid alapú kereséshez képest. A mátrixot közvetlenül is beadhatjuk a programnak, de a program egy előzőleg futtatott BLAST kimenetéből is elő tudja állítani, ha szükséges (Altschul et al., 1997).

Érdemes még megemlíteni a MegaBLAST-ot. Abban az esetben lehet használni, ha az adatbázis mérete miatt a ,,közönséges" BLAST túl lassú lenne. A MegaBLAST hosszabb szegmens párokkal dolgozik, ami az érzékenységét csökkenti. Nem egyetlen szekvenciával keres, hanem többel, amiket összefűz egybe, majd az eredmények kiírásánál szétdarabolja (Sung, 2010).

2.3.3. Egyéb algoritmusok

A BLAST mellett számos más algoritmus létezik, melyek igyekeznek egy-egy potenciális niche-t kiragadni maguknak. A BLAT (BLAST-like alignment tool) a BLAST-hoz hasonlóan működik, de kereső táblát hoz létre az adatbázishoz, amivel megnöveli a keresés sebességét. A visszaadott eredményeknél pedig összevon több találatot, amennyiben azok azonos szekvencián vannak.

A PatternHunter más megközelítést használ. A szekvenciák hasonlóságát egy úgynevezett spaced-seed segítségével állapítja meg. A szakirodalom szerint kisebb lesz a találatok száma, a fals pozitív találatok rovására.

Végezetül érdemes még említést tenni a BWT-SW algoritmusról. A heurisztikus algoritmusok, mint amilyen a BLAST, nem garantálják, hogy megtalálják az összes lehetséges optimális illesztést. A Smith-Waterman képes erre, de sebessége nem elfogadható. A BWT-SW célja, hogy gyorsítsa a Smith-Waterman algoritmust különböző indexelési eljárások alkalmazásával.

2.4. Bioinformatikai adatbázisok

2.4.1. Elsődleges és másodlagos adatbázisok

A DNS szekvenálással egy időben szükségessé vált a szekvenciák tárolása és gyors keresése. A számítástechnika nyújtotta eszközökkel mindez lehetővé vált. Három nagy elsődleges szekvencia adatbázis van, a GenBank (Benson et al., 2010), az EMBL-Bank (Kulikova et al., 2007) és a DDBJ (Sugawara et al., 2008). A kutatók bármelyikbe is küldjék a bázissorrendet, az adatbázisok együttműködésének hála az mindháromban meg fog jelenni.

Ahogy nőtt a szekvenciák száma, úgy jelentek meg a speciális adatbázisok, melyek egy jól körülhatárolt szempont szerint gyűjtötték össze az elsődleges adatbázisokból származó elemzett adatokat. Ilyen szempont lehet például a faj, transzkripciós start hely (Wakaguri et al., 2008), vagy több genom homológ szakaszai. A később bemutatásra kerülő GO, Jaspar és Transfac adatbázisok is ide tartoznak. Ezek után nem meglepő, hogy a CAGE adatoknak is van adatbázisuk, ahol fajokra lebontva megtalálhatóak a génekhez tartozó TSS-ek (Kawaji et al., 2006).

2.4.2. Motívum adatbázisok

A motívumokat két nagyobb adatbázis gyűjti. Ezek neve JASPAR (Sandelin et al., 2004) és a TRANSFAC (Matys et al., 2006). A TRANSFAC transzkripciós faktor kötőhelyeket tartalmaz súlymátrixok formájában. Hozzáférése nem ingyenes, a benne található motívumok redundánsak. A Patch nevű programjuk segítségével lehet keresni konszenzus szekvenciával is az adatbázisban. Előnye, hogy tartalmaz növényi TFBS-eket is.

A JASPAR kisebb mennyiségű adatot tartalmaz, mint a TRANSFAC, de a tartalma ingyen elérhető bárki számára és a benne található valamennyi TFBS kurátor által ellenőrzött, minimális redundanciát tartalmaz. Az adatbázisban a TFBS-ek mátrixok formájában tároltak, de felépítésük eltér a korábban bemutatott PSWM-től, itt ugyanis

az elemek azon szekvenciák számai, melyekben az adott pozícióban a megadott bázist tartalmazza a motívum (Sandelin et al., 2004).

2.4.3. Gén ontológiai adatbázisok

Ha motívumainkhoz biológiai funkciót kívánunk rendelni, szükségünk van egy olyan adatbázisra, mely egyértelműen beazonosítja és kereshetővé teszi ezen jellemzőket. Jelenleg ilyen adatbázis csak fehérjékhez és génekhez létezik.

A gén funkciók rendkívül sokrétűek. Az egyes géneket célszerű funkció alapján csoportosítani, hogy több gént egy kategóriába lehessen sorolni. Ha a kategóriákat is csoportosítani kell, akkor több probléma is felmerülhet.

Az egyik ilyen probléma, hogy megfelelő kategóriákat kell találni. A másik, hogy egy gén így több kategóriában is előfordulhat. A kategóriák kijelölése csak önkényes alapon lehetséges, mert a csoportosítás csak egyféle szempont szerint mehet. Ha egy másik szempontot választunk, akkor az addig összetartozó csoportok szétesnek.

Jelenleg az egyik legelterjedtebb és legmegfelelőbb erre a gén ontológiai adatbázis (GO). Itt egy úgynevezett aciklikus irányított gráf csomópontjaiként jelennek meg a kategóriák. (Gene Ontology Consortium, 2006)

A gráf gyökeréből kiindulva három nagy csoportot találunk. Az első a sejt alkotók, amelyek vagy önmagukban vagy egy nagyobb kompartment részeként a sejt anatómiai felépítését végzik. A második csoport a biológiai folyamatok, melyek több lépéses események során alakítanak ki egy terméket. Tipikusan ilyenek a különböző metabolikus reakciók és a jelátvitel. Fontos megemlíteni, hogy nem szabad összekeverni ezt a kategóriát az anyagcsere útvonalakkal, mert a GO a leírásában nem utal sem a folyamatok dinamikájára, sem a különböző függőségekre. Gyakran nehéz elválasztani őket a harmadik GO csoporttól, a molekuláris funkcióktól. Ez utóbbiak molekuláris szintű eseményeket írnak le, mint amilyenek az egyes molekulák megkötése vagy a katalitikus aktivitás.

Aciklikus volta miatt az egyes kategóriák a gráf több, különböző szintjén is megjelenhetnek, viszont a fa bejárása során soha nem találkozunk hurkokkal (tehát egy irányba haladva soha nem juthatunk vissza olyan csomópontba, amit már egyszer érintettünk), ami az algoritmikus feldolgozást könnyíti. Ezzel a lépéssel viszont a statisztikai értékelés jut nehéz helyzetbe.

Kiértékelésnél a legfontosabb annak megállapítása, hogy mely kategóriák dúsultak fel egy génlistában. Erre a következő statisztikai eljárásokat használhatjuk: hipergeometrikus teszt, Fisher-próba, khi-négyzet próba, binomiális teszt. Jelenleg a legop-

timálisabb eredményt a hipergeometrikus eloszláson alapuló módszerek adják, mert ezek nem érzékenyek a minták számára (Rivals et al., 2007).

Ha felépült a rendszer, akkor azt adatokkal is fel kell tölteni. A gének ontológiai adatbázisokba sorolásának két szélsőséges módszere szerint történhet kísérletek alapján, ami a legpontosabb, de leglassabb osztályozási eljárás, vagy szekvencia homológia alapján, ami könnyen automatizálható, ellenben kevésbé megbízható. A két szélsőséges eset között nagyszámú átmenet található. Ezeket a GO adatbázisban úgynevezett bizonyítékként tárolják. A bizonyítékok rövidítései és leírásuk a 2.2 táblázatban látható. Habár a GO konzorcium szerint a bizonyítékok nem minőségi mutatók, a gyakorlatban a kutatók jobban megbíznak egy kísérletes bizonyítékban, mint egy nem számon kérhető kurátor véleményében.

Az elemzés nehéz voltát bizonyítja, hogy egyre másra jelennek meg a különböző módszerek.

2.4.4. ENCODE tervezet

A gének mellett egyéb funkcionális elemek is találhatóak a genomban. Ezen elemek feltérképezését a humán genomban az ENCODE (Encyclopedia of DNA elements, DNS elemek enciklopédiája) konzorcium tűzte ki célul. Ez a több nemzetközi kutatócsoportot magába foglaló szervezet az összes fellelhető funkcionális elemet fel kívánja térképezni, akár fehérje vagy RNS kódoló szekvenciáról, akár a szabályozásban részt vevő elemről van szó.

A grandiózus célkitűzések megvalósulása érdekében először csak bizonyos régiókat választottak ki, melyek együttes hosszúsága 30 Mb, ami közelítőleg a genom 1%-nak felelt meg.

A régiók fele manuálisan került kiválasztásra, ahol a feltétel az volt, hogy a régió tartalmazzon olyan géneket, melyekről sok irodalmi adat áll rendelkezésre, valamint jelentős mennyiségű összehasonlító szekvencia van hozzá. A régiók másik felét véletlenszerűen választották ki. A harminc darab 500 kb-os szekvencia kiválasztásánál ügyeltek, hogy gén denzitás és a konzerváltság mértéke különböző legyen, hogy megfelelő áttekintő képet kapjanak az emberi genomról (ENCODE Project Consortium, 2007).

A felhasznált módszerek között a kvantitatív PCR és a kromatin immunprecipitáció is megtalálható. A kapott eredményeket nyilvános adatbázisokban lehet megtekinteni, mint amilyen például az UCSC (UCSC weboldal).

Miután a tervezet bevezető szakasza sikeresen lezárult, a vizsgálatokat kiterjesztették a teljes humán genomra.

EXP	Kísérletből
IDA	Direkt Assayből
IPI	Fizikai interakcióból (például 2 hibrid, ion kötés vizsgálat)
IMP	Mutáns fenotípusból
IGI	Genetikai kísérletből
IEP	Expressziós mintázatból (Northern blot, chip)
ISS	Szekvencia vagy strukturális hasonlóság alapján
ISO	Ortológ szekvencia alapján
ISA	Szekvencia alapján
ISM	Szekvencia modell alapján (pl. Rejtett Markov modell)
IGC	Genomi környezet alapján (pl. operon struktúra)
RCA	Számítógépes analízis
TAS	Visszakereshető szerző által
NAS	Nem visszakereshető szerző által
IC	Kurátor alapján
ND	Nincs bizonyíték

2.2. táblázat. A GO adatbázisban tárolt leírás és gén közti kapcsolat bizonyítékainak jegyzéke

2.4.5. Gén expressziós adatbázisok

A génexpressziós adatokat három nagyobb adatbázis gyűjti, az ArrayExpress, aminek az EBI ad otthont és az NCBI-nál található GEO (Gene Expression Omnibus) adatbázis, valamint a japán CIBEX. Rajtuk kívül még léteznek kisebb adatbázisok, melyek egy faj vagy modellállat expressziós adatait gyűjtik össze, mint amilyen a GXD, mely a laboratóriumi egérre specializálódott és a FlyEx, ami *Drosophila melanogasterre*.

Az ArrayExpress több, mint 200 faj expressziós adatát tartalmazza. Összetett keresések segítségével megtalálható a kívánt kísérlet, sőt weben keresztül elemezni is lehet az adatokat. Mivel az egyes funkciók SOAP-kérésként jutnak el a szerverhez, ezért más programokból is el lehet érni azokat (Parkinson et al., 2007). A SOAP egy szabványos felépítésű, szöveg alapú üzenet az interneten elküldvel, amit a kiszolgáló értelmez és a kéréshez hasonló szabványos választ ad vissza.

A GEO hozzáállása más, ők elsősorban a mennyiségre helyezik a hangsúlyt. Az adatok beküldése olyan egyszerű és rugalmas, hogy az adatok 15%-a nem is expressziós adat! Itt is lehetőség van elemzésére böngészőprogramon keresztül, de harmadik fél által írt programokkal csak úgy dolgozhatunk, ha letöltjük az adatokat (Barrett et al., 2009).

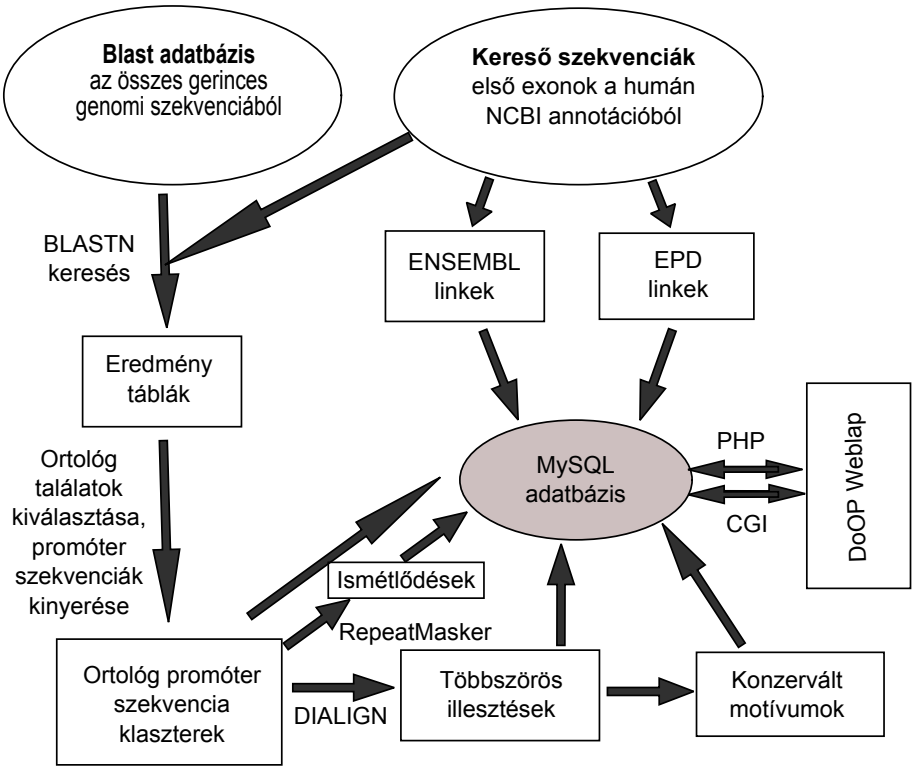
Mindhárom adatbázis a standardizálás jegyében a MIAME (minimálisan szükséges információ egy microarray kísérletről) ajánlását követi. Célja, hogy az adatok egyértelműek, a belőlük levont következtetések megismételhetőek legyenek. Nem köt ki formai követelményeket, de előírja, hogy minden kísérletnek tartalmaznia kell többek között a nyers adatfájlokat, a feldolgozott adatokat, a laboratóriumi és elemzési módszereket és a chip gyári adatait.

2.4.6. DoOP

Csoportunk is készített egy speciális adatbázist, a DoOP-ot (Barta et al., 2005). Célja, hogy könnyen kereshető formába gyűjtse össze a növényi és gerinces fajok genomjaiból származó szabályozó régiókat és képet adjon az ott található feltételezett transzkripciós szabályozó elemekről.

Két referencia faj, az ember és az *Arabidopsis thaliana* genomját felhasználva a BLAST program segítségével megkerestük más fajok homológ első kódoló exonját. A referencia fajok kiválasztásának szempontja a jól annotált genom volt. Az első exonoktól 5' irányban található szekvenciát tekintettük az adott kódoló szakasz szabályozó régiójának.

Mivel a promóterek pontos határát jelenlegi ismereteink szerint nem lehet megmondani, ezért az adatbázis 500, 1000, 3000 bázispár hosszú szakaszokat tartalmaz. Egy



2.4. ábra. A DoOP adatbázis elkészítésének folyamata

referencia szekvenciához tartozó összes ortológ szekvenciák együtt úgynevezett klasztert alkotnak. A klaszterek bevezetésével az annotáció jó közelítéssel kiterjeszthető az ismeretlen promóterekre is (2.4. ábra).

Az ortológ promóterek birtokában már sokkal érdekesebb kérdésekre kereshetjük a válaszokat. Például mely elemek konzerválódtak? Az adatbázisban szereplő konzerválódott elemek a következő módszerrel kerültek megállapításra: A DIALIGN2 (Morgenstern, 1999) program segítségével az egy klaszterbe eső promóterekből többszörös illesztést készítettünk. A program által számított információs tartalmat felhasználva egy saját fejlesztésű programmal kiválasztottuk a többszörös illesztés azon részeit, melyeknél az információs tartalom a legmagasabb volt. Természetesen az ismétlődő szakaszok (repeatek) eltávolításra kerültek. Ezen elemek bázissorrendjének ismeretében kísérleteket lehet tervezni a szabályozás pontosabb felderítésére akár olyan fajokban is, ahol a genom annotálása még gyerekcipőben jár.

3. fejezet

Célkitűzések

Munkánk célja az volt, hogy kapcsolatot találjunk a gerinces állatok génjeinek biológiai szerepe és a promóterükben található konzerválódott motívumok konszenzus szekvenciája között. A problémát több irányból, több módszerrel közelítettük meg, hogy minél több lehetőséget lefedő eredményeket kapjunk.

Első lépésben szükség volt egy olyan bioinformatikai eszközre, ami a már meglévő, DoOP-ban elérhető motívum adatbázisunkból képes kikeresni a hasonló szekvenciával bírókat. Fontos szempont volt, hogy ez az eszköz más kutatók számára is hozzáférhető legyen egy weboldal formájában. A hasonlóságot felhasználva a motívumok klaszterezhetőek, az eredményül kapott csoportok pedig GO annotációs vizsgálatnak vethetőek alá. Ha ezzel az eszközzel egy csoporton belül szignifikáns feldúsulást találnánk egyes GO kategóriákban, akkor kijelenthetjük, hogy megtaláltuk a választ a kérdésünkre.

Ha a klaszterezés bármilyen okból kifolyólag nem hozná az elvárt eredményt, akkor azonos biológiai szereppel bíró géneket tervezzük kísérletesen kiválasztani, és bioinformatikai módszerekkel közös motívumokat találni a promóter régiójukban. A génexpressziós vizsgálatok segítségével megtalálhatóak egy biológiai reakcióra expressziós szint változást mutató gének. Ezen gének promóterében *de novo* motívum kereséssel vagy a szakirodalomban fellelhető konzervált szakaszok felkutatásával ugyancsak bizonyítható lenne a kiindulási feltevés.

Nagy valószínűséggel ezek a motívumok a transzkripciós szabályozáson keresztül fejtik ki hatásukat, ezért célként tűztük ki, hogy megvizsgáljuk ennek lehetőségét is. Számos publikáció utal rá, hogy a transzkripciós start hely környékén a szabályozásban szerepet játszó elemek száma nagyobb, mint attól távolabb. Ha azt találnánk, hogy a fent vázolt vizsgálatokból származó motívumok eloszlása nem független a TSS-től, akkor nem csak azt mondhatnánk el, hogy megtaláltuk a kapcsolatot a biológiai szereppel, de a kapcsolat

miként jére is fényt deríthetnénk.

Összefoglalva a következő feladatokat tűztük ki:

- Motívum elemzésre alkalmas bioinformatikai eszközök készítése.
- Konzerválódott motívumok csoportosítása.
- Kapcsolatot találni a motívum-csoportok és a biológiai elemek között.
- Motívumokat keresni azonos funkciójú gének promóterében
- Kapcsolatot találni a motívum-csoportok és a transzkripció kezdőpontja között.

4. fejezet

Anyagok és módszererek

4.1. Felhasznált számítógépek

Az elemzések, ahol külön nincs jelezve, 1,8 GHz-es Pentium 4 számítógépen futottak 1GB memóriával, GNU/Linux Slackware 11, később Slackware 12 operációs rendszeren. A `mofext` programokat következetesen egy 4 processzoros SUN Sparc gépen futtattuk Solaris 9 operációs rendszer alatt 16 GB memóriával. A GeneSpring DNS chip elemző program Microsoft Windowsos verzióját használtuk egy HP OmniBook laptopon 256 MB memóriával.

4.2. Felhasznált adatbázisok

A DNS szekvenciák elsődleges forrása az EnsEMBL adatbázis volt (Flicek et al., 2008). A munka ideje alatt mindig a legfrissebb verzió volt telepítve a 35-től a 42-ig. Az adatbázis verziószámával megegyező Perl API felhasználásával készültek azok a programok, melyek segítették az igényeknek megfelelő szekvencia kinyerését az adatbázisból.

Homológia vizsgálatokhoz az EnsEMBL Compara adatbázis szolgáltatta a forrást. A munka ideje alatt az adatbázisban elérhető fajok száma folyamatosan növekedett, de tárolóhely szűke miatt csak a következő fajokat tartalmazta a helyi rendszer: *Homo sapiens*, *Pan troglodytes*, *Gallus domesticus*, *Bos taurus*, *Canis familiaris*, *Fugu rubripens*, *Tetraodon nigroviridis*, *Mus musculus*, *Rattus norvegicus*.

A GO analízishez a GO adatbázis 2007 februári kiadását használtuk. A szignifikáns GO kategóriákat a GeneMerge 1.2 program módosított változatával határoztuk meg (Castillo-Davis and Hartl, 2003). A forráskódja a CD mellékletben található.

4.3. A szekvencia adatok feldolgozása

Az adatbázisból kinyert szekvenciák feldolgozására az Emboss programcsomag 3.1-es verzióját használtuk (Rice et al., 2000). Egyszerűbb motívumok keresésére a `fuzznuc` program megfelelt. A vizsgálatok során 2 bázispárnál több cserét nem engedtünk a kereső szekvenciában, mert nagyon sok irreleváns találatot kaptunk volna vissza.

A vizsgálatok során gyakran éltünk az összehasonlítás azon formájával, hogy a szekvenciát összekevertük és megismételtük a kísérletet. A bázissorrend megváltozott, de a nukleotid összetétel nem. A véletlenszerű összekeverést a `shuffleseq` programmal végeztük, *-shuffle* paraméterének a 2-t választottuk, vagyis két ciklusban ment végbe a keverés.

Egy oligonukleotid előfordulásának valószínűségét a `compseq` programmal határoztuk meg. Ez összeszámolja a szekvencián belül az összes oligonukleotid előfordulást, majd a bázisok előfordulásának valószínűségéből kiszámolja az oligonukleotid várható értékét. A két érték hányadosa alapján, amit szintén megad a program, akkor fogadtuk el a véletlennél nagyobbnak az oligonukleotid előfordulását, ha meghaladta a 2-t.

A repetitív szekvenciák eltávolítását a `censor` 4.1 verziója végezte. Humán szekvenciák esetében a *-hum* kapcsolót használtuk, mely a faj specifikus ismétlődések mellett az ALU szekvenciákat is kivágja (Jurka et al., 1996).

4.4. Motívum keresési módszerek

Motivumok keresésére a NestedMica 0.7.3-as verzióját használtuk (Down and Hubbard, 2005). A program igen érzékeny az *-ensembleSize* paraméterre, ezért végeztünk néhány teszt futtatást, hogy megállapíthassuk ennek optimális értékét. Minél nagyobbra állítjuk, annál pontosabb lesz az eredmény, de a futási időt is megnöveli. Gyakorlati tapasztalatok alapján ezért 400-ra állítottuk. A nagyszámú adat miatt 8 db Sun Fire X2100-as gépen futott, melyek 2,2 GHz-es AMD Opteron processzorokkal voltak felvértezve. A gépeket a debreceni egyetem bocsátotta a rendelkezésünkre. Mindegyik gépen két szálon futtattuk a programot (*-threads* paraméter). A java-s futtatókörnyezetnek pedig az alapértelmezettnél nagyobb memóriát állítottunk, 300 MB-ot. A NestedMica ezen felül megköveteli, hogy beállítsuk neki a megtalálandó motívum nagyságát is, amit 13-ben határoztunk meg (*-targetLength* paraméter). Ennél rövidebb szekvencia túl általános lett volna, a hosszabbak pedig a feleslegesen növelnék a futási időt. A program ugyanis annyi motívumot fog találni, amennyit paraméterként megadunk neki, még akkor is, ha azok információs tartalma alacsony. Ezen megfontolások alapján négy motívum megtalálását

tűztük ki célul (*-numMotifs* paraméter).

Később megpróbálkoztunk egy másik keresési módszerrel is, ami a program azon képességén alapult, hogy különböző fajok ortológ szekvenciáiból is képes eredményeket kinyerni. Az ortológ szekvenciákat a kutya, szarvasmarha, egér és patkány genomokból választottuk.

A programnak van még egy kényelmi szolgáltatása. Az épp aktuális számítási eredményeket bizonyos futási ciklus után (*-checkpointInterval* paraméter) egy fájlba írja (*-checkpoint* paraméter). Váratlan hiba esetén az utolsó állapottól képes folytatni a működését. Ebben az esetben a *-restartFromCheckpoint* paraméterrel kell újraindítani a programot.

Összefoglalva a következő parancsot futtattuk:

```
makemosaicbg -seqs szekvencianév -mosaicClasses 1 \
             -mosaicOrder 1 -out mosaicfájl
motiffinder -numMotifs 4 -targetLength 13 -seqs szekvencianév \
            -backgroundModel mosaicfájl -outFile eredményfájl \
            -sampleFile mintafájl -ensemblSize 400 -cluster \
            -checkpoint állapotfájl -checkpointInterval 2000 \
            -threads 2
```

Az eddigieken kívül alkalmaztunk olyan keresési módszert is, ahol a promóter méretét úgy definiáltuk, hogy a következő gén határáig tartson. Ebben az esetben a szekvenciákat 30 ezer bázispárnál elvágtuk a `seqret` programmal, mert a NestedMica ennél nagyobb szekvenciáknál lefagyott.

4.5. Chip és kromatin immunprecipitációs vizsgálatok kiértékelése

A chip kísérleteket Szatmári István végezte Affimetrix HG-U133A típusú plate-n. A nyers chip adatok kiértékelését GeneSpring 7.3.1-al és Bioconductor 2.0-val hajtottuk végre (GeneSpring weboldal) (Reimers and Carey, 2006). Minden esetben GC RMA (Millenaar et al., 2006) előkészítést alkalmaztunk. Amelyik expressziós adat 0,01-nél kisebb volt, azt a számolások egységesítése végett 0,01-nek vettük. A chipenkénti expressziós értékeket 50%-hoz, míg génenként a mediánhoz normalizáltunk. Azon géneket, melyek nyers szignál értéke mindegyik vizsgálatban 20 alatt volt, eltávolítottuk, ellenben meg-

tartottuk azokat, ahol a szignál érték 2-szeres változást mutatott a kezeletlen mintákhoz képest, és ez a változás szignifikáns volt (`p < 0,01`).

A vizsgálat tárgyai a roziglitazonnal (RSG) kezelt monociták voltak. RSG kezelés hatására a monociták dendritikus sejtté érnek. Az RSG egy mesterséges ligandja a peroxiszóma proliferátor-aktivátor receptor gammának (PPARγ), ami a lipid anyagcserében szerepet játszó transzkripciós faktor. Ez a transzkripciós faktor közvetlenül szabályozza számos zsírsav felvételben és lipid raktározásban szerepet játszó gén kifejeződését. A monocita differenciációja során ezen receptor által szabályozott gének fokozott aktivitást mutatnak. Ilyen gén például az FABP4 és az ABCG2.

A kezelést követően hat, huszonnégy óra elteltével, valamint 5 nappal később mintát vettünk és megmértük az egyes gének RNS szintjét. Ha a kezelést követően az RNS mennyisége meghaladta a kezelés nélküli RNS mennyiségének kétszeresét, fokozott, ha kevesebb, mint fele mennyiségű volt, csökkent aktivitást mutatónak tekintettük. A későbbi időpillanatban vett mintákból nyert gének között előfordulhatnak korábban aktiválódott gének, melyek expressziós szintje nem csökkent le, ezért a több időpillanatban is előforduló gének csak a legkorábbi listában lelhetőek fel (Szatmari et al., 2007).

A chip elemző programok által kinyert génlistákhoz tartozó génszekvenciákat a bioinformatikai vizsgálatokhoz az EnsEMBL-ből töltöttük le saját fejlesztésű programokkal. A `t_affy_genseq.pl` az Affymetrix saját azonosítói alapján kiszedi az adatbázisból a gén szekvenciáját. Ehhez a programhoz hasonló a `t_affy_promo2.pl`, ami az adott génhez tartozó szabályozó régió szekvenciáját adja vissza.

A kromatin immunprecipitáció szintén Affimetrix márkájú chippel készült. A laboratóriumi munkát Bálint B. László végezte. A vizsgálatok célja a kromatin struktúra változásának felderítése retinsav kezelés hatására, mieloid leukémia sejtekben. A HL-60/CDM-1 sejteket először egy 16 órás DMSO kezelésnek vetettük alá, ami az érést elindította, de a sejtdifferenciációt nem. Ezeket a sejteket a vizsgálatok során naív sejteknek jelöltük. A retinoid kezelés hatására megindult a differenciáció. Az immunprecipitációs lépés során ellenanyag segítségével megjelöltük a H3 hisztont, amikor a K4 oldallánca metilált állapotban volt. Egy másik kísérlet során az ellenanyag a H4 hiszton acetilált R3 végét jelölte. A hisztonhoz kötődött DNS-t az elválasztás után az Affimetrix Encode chip-en hibridizáltuk, hogy megtudjuk mely genomi pozíciókban találhatóak.

A nyers adatok feldolgozását az Affymetrix Tiling Analysis SDK 2 parancssoros programjával végeztük a következő beállítások felhasználásával: -type 0 -band 25 -pval_scale 0 -sig_scale 2. Az eredményül kapott genomi pozíciókhoz tartozó szekvenciákat az NCBI 35-ös verziójú emberi genomból nyertük ki.

4.6. Statisztikai elemzések

A különböző hipotézisek statisztikai ellenőrzését az R csomag 2.4.0-ás verziójának felhasználásával végeztük. A teszteket 0.05 százalékos konfidencia intervallumon számoltuk. Különböző átlagok vizsgálatához a `t.test` programot használtuk, míg a korrelációkhoz a `cor`-t.

5. fejezet

Eredmények

5.1. DoOP modul fejlesztés

Az új DoOP honlap tervezésénél fontos szempont volt, hogy a kiszolgáló oldali programok egységes programozói felülettel (API) rendelkezzenek. Ez nem csak a CGI szkriptek írását, tehát a weboldal fejlesztését könnyíti meg, hanem a későbbiek során segítséget nyújt a parancssoros programok elkészítéséhez is.

Mivel a legtöbb bioinformatikus a Perl programozási nyelvet használja, és a nyelv alkalmas CGI szkriptek létrehozására is, ezért rá esett a választás. A nyelv másik nagy erőssége a programozói könyvtárak nagy száma. Ezek az interneten a CPAN-on találhatóak meg (CPAN weboldal). Azért, hogy ezek a könyvtárak a fejlesztők számára könnyen áttekinthetőek legyenek, valamint ne írjon két ember ugyan arra a problémára két különböző modult, szigorú osztályozást és névkonvenciót vezettek be. A lényege, hogy egy adott feladatra létrehozott modulok közös névterekbe kerüljenek. A biológiai munkákhoz a `Bio` névtérben találhatóak programozói könyvtárak. Mivel a DoOP, mint adatbázis nem illeszthető be egyik alkategóriába sem a `Bio` névtér alatt, ezért külön, egy `DOOP` nevű névtérbe került. A tényleges osztályok ezen belül találhatóak.

Az adatbázis kapcsolatot a DBSQL osztályon keresztül lehet létrehozni. Jelen formájában, ezen modullal írt programok csak MySQL adatbázishoz tudnak kapcsolódni, mivel a DoOP weboldala is ezt használja. A biológiai tartalmat a `Cluster`, `ClusterSubset`, `Sequence`, `SequenceFeature`, `Motif` osztályok reprezentálják. Az osztályok hierarchikusan épülnek fel, tükrözve az alatta található adatstruktúra alárendeltségi viszonyait. A szülő osztályból elérhetőek annak gyermek osztályai. Például az egy klaszterbe tartozó alcsoportok mindegyikéről (ClusterSubset) információt kaphatunk a `Cluster` osztály segítségével.

A tervezésnél fontos szempont volt a rendszer gyorsasága is. A `Sequence` osztály ezért nem Bioperl kompatibilis módon adja vissza a szekvenciát, mert ez további adatmozgatási műveletekkel járt volna, ami a program hatékonyságát csökkenti. Ez talán kényelmetlenséget okoz, de megakadályozza a modulok esetleges összeférhetetlenségét a programozói könyvtár további fejlesztése esetén. Ha ugyanis a Bioperl szekvencia osztályai változnak, a DoOP osztályokat is okvetlenül át kell írni, hogy összhangban legyenek.

Az egyes motívumok szekvencián belüli elhelyezkedésének grafikus ábrázolására született meg a `Graphics` osztály. Segítségével szemléletessé lehet tenni, hogyan viszonyulnak egymáshoz a motívumok a szabályozó szekvencián belül és mely régiókban dúsulnak fel azok. Kimenete szabványos PNG állomány, melyet az összes fejlett böngésző és képnézegető alkalmazás képes megjeleníteni.

Végül érdemes még említést tenni a `Util` osztályról. Kicsit kilóg a `DOOP` modul többi osztályából, mert inkább egy eljárásgyűjteménynek tekinthető, mint valódi objektumorientált osztálynak. Célja, hogy a gyakran felmerülő problémákra elegáns megoldást nyújtson anélkül, hogy több fejlesztőnek is egymástól függetlenül létre kelljen hoznia ugyan azokat a kódrészleteket. Ilyen feladat lehet például az összes olyan klaszter kikeresése, mely tartalmaz egy adott gént, vagy egy bizonyos motívumot tartalmazó szekvenciák visszaadása. Az osztálynak van még egy gyermekosztálya, ami lehetővé teszi, hogy `fuzznuc` és `mofext` programokat futtassunk az adatbázisból kinyert szekvenciákon. A `mofext`ről bővebben az 5.2 fejezet ír. A teljes csomag forráskódja megtalálható a CD-n és a CPAN-on. A licencelése lehetővé teszi, hogy bárki ingyenesen letöltse és használja.

Az alábbiakban a modulok használatának rövid bemutatása következik. További segítség a programozói felület súgójában található, amit telepítés után a következő módon lehet elérni GNU/Linux parancssorban:

```
perldoc Bio::DOOP::DOOP
```

A programozói könyvtár használatának első lépése a csomagok betöltése. A korábban bemutatott osztályok betöltéséhez elegendő csupán a DOOP modul betöltése.

```
use Bio::DOOP::DOOP;
```

A második az adatbázishoz való kapcsolódás. A megadott adatok az adatbázis telepítésétől függenek.

```
$db = Bio::DOOP::DOOP::DBSQL->connect("felhasznaloi_nev",
                                      "jelszo",
                                      "adatbazis_nev",
                                      "szamitogep_url");
```

A `$db` változón keresztül tudjuk a kívánt adatokat elérni. Példának okáért keresük meg az összes olyan 500 bázispár hosszúságú promótert, aminek a leírásában szerepel a cink (zinc) kulcsszó.

```
@clusters =
@{Bio::DOOP::Util::Search::get_all_cluster_by_keyword($db,
                                                      "zinc",
                                                      500)};
```

Az eredményül kapott klaszterekkel ezután bármilyen műveletet végre lehet hajtani. Például kiírathatjuk az azonosítójukat.

```
for(@clusters){
   print $_->get_cluster_id,"\n";
}
```

Az itt bemutatott példákból is jól látszik, hogy minimális programozói ismeretekkel is könnyen lehet használni az adatbázisban fellelhető adatokat. A bioinformatikusok több időt tölthetnek a kapott adatok elemzésével, nem kell ismerniük a háttérben meghúzódó bonyolult rendszereket.

5.2. Motívum összehasonlítás

Ha a modulok segítségével sikeresen hozzájutottunk a munkánkhoz szükséges motívumokhoz, a következő lépés, hogy más, hozzájuk hasonló szekvenciájú motívumokat találjunk. Az összehasonlítására a közönséges szöveg alapú keresés, amit például a népszerű szövegszerkesztő programok is használnak, nem alkalmas, mert nem engedélyezett a kereső motívum ,,lötyögése", tehát az adott pozícióban előforduló alternatív bázisok jelenléte. Az Emboss programcsomagban található `fuzznuc` program már lehetővé teszi a báziscserét az összehasonlításkor, de nem teszi lehetővé, hogy az egyes alternatív nukleotidokat súlyozzuk. Tehát a lehetséges bázisok egyforma valószínűséggel vesznek részt a keresésben. Ez nagy számú fals pozitív találathoz vezethet olyan esetekben, ha tisztában vagyunk vele, hogy egy pozícióban melyek azok a bázisok, melyek nagyobb valószínűséggel vesznek részt a motívum felépítésében, és melyek azok, amelyek nem.

A másik gyakori probléma a motívum összehasonlító programok esetén, hogy a keresőszekvencia csak adott hosszúságú lehet, míg a mi motívumaink különböző

hosszúságúak. Ez olyan esetben lehet probléma, ha a keresőszekvencia hosszabb, mint az a szekvencia, amivel összehasonlítjuk. Azért is nehéz meghatározni, hogy egy hosszabb motívum mikor tekinthető hasonlónak egy rövidebbhez, hiszen egy hosszabb statisztikailag nagyobb valószínűséggel tartalmazza a kisebbet.

A Blast algoritmus mentes lenne ettől a hibától és használható lenne olyan rövid szekvenciák esetében is, mint amilyenek a motívumok, de ennek a programnak nem lehet konszenzus szekvenciát megadni. A pozíció specifikus súlymátrixok alkalmazása az adatbázis szerkezete miatt nem volt alkalmas (mert konszenzus szekvenciákat tárol), ráadásul az ezen alapuló keresési algoritmusok igen erőforrás igényesek. Szükség volt egy olyan megoldásra, mely rendelkezik a súlymátrixok előnyeivel, mégis megmarad a konszenzus szekvenciák használhatóságának egyszerűsége.

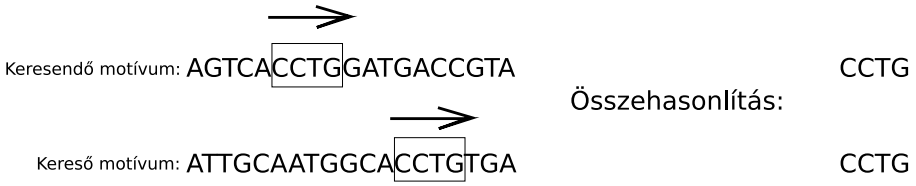
A megoldást a `mofext` algoritmus jelentette. Ez egy hasonlósági mátrixot használ annak megállapítására, hogy az adott bázisok mennyire hasonlóak az összehasonlítani kívánt motívumokban. Mindkét motívumot egységnyi darabokra bontja, méretüket a program *-w* opciójával állíthatjuk be. A darabolás során keletkező részek egyenlő hosszúságúak, ezért a hasonlósági mátrix felhasználásával a két szekvencia részletben a nukleotidok hasonlósági pontszámait összeadja. A program továbbá kiszámítja, hogy milyen összeget kapnánk, ha az összehasonlítandó szekvencia megegyezne a kereső szekvenciával. Ez a legtökéletesebb egyezés értéke. A két összeg hányadosa megadja, hogy milyen százalékban egyezik meg a két szekvencia részlet (5.1 ábra).

Ha ez a hányados nagyobb a *-c* opcióban beállított hasonlósági küszöb értéknél, a program megpróbálja kiterjeszteni az összehasonlítást egy hosszabb szakaszra, ellenkező esetben továbblép a következő összehasonlítandó darabokra.

A kiterjesztés során a darabokhoz további nukleotidokat illeszt, és folyamatosan számolja, hogyan változik a hasonlósági pontszámok összegének hányadosa. Ha az új nukleotid hozzáadása csökkenti ezt a hányadost, a program nem folytatja a kiterjesztést, de találatként értékeli az egyezést.

Eredményül a kiterjesztett szekvenciákat kapjuk. Az eredmények között feltűnteti a hasonlósági mátrix alapján számított összeget és a szekvenciák azonosságának arányát is. Parancssori opciókkal szabályozható a megjelenítendő információ mennyisége.

A program parancssori paraméterei a következőek: *-d* kapcsolóval adható meg az a fájl (vagy fájlok, mert szóközzel elválasztva többet is felsorolunk), amiben keresni kell a motívumokat. A keresőmotívumok konszenzus szekvenciáját a *-q* opcióval adhatjuk meg, szóközzel elválasztva. A felhasítandó darabok méretét a *-w* opcióval lehet beállítani. Az *-m* kapcsolóval megadott mátrix állományt használja a program az összehasonlítás



5.1. ábra. A `mofext` program működési elve

ch-13-500-80100003-1	agGctGgGct	10	151	160
ch-13-500-80100003-2	AgGAcaATYGTTR	13	465	477
ch-13-500-80100003-3	CTtggcTgGATTGTTACMta	20	479	498
ch-13-500-80100003-4	AaRAgGCctc	10	566	575
ch-13-500-80100003-5	GaggAtg	7	650	656
ch-13-500-80100003-6	TGCTAGCc	8	684	691
ch-13-500-80100004-1	GKCTRACTCT	10	374	383
ch-13-500-80100004-2	GCCCaa	6	610	615
ch-13-500-80100004-3	TtctgTCTaCTgt	13	617	629
ch-13-500-80100004-4	GCCWCTGYCT	10	646	655

5.1. táblázat. A mofext program bemeneti állománya

pontozásához. Az eredmények közé csak azok a találatok kerülnek, melyek a *-c* opcióval megadott hasonlósági küszöbérték feletti értéket kapnak. Itt egy 0 és 1 közötti számot adhatunk meg. Az eredmények a képernyőre kerülnek kiírásra. A kimeneti formátumot az *-o* kapcsolóval szabályozhatjuk, itt adhatjuk meg, mely oszlopok jelenjenek meg az eredmények leírásánál. A program futtatására álljon itt egy rövid példa:

```
> mofext -d mypatterns1.list mypatterns2.list -q GGATCC
TTGANTGA -m matrix.dat -w 4 -c 0.5 -o ied
```

A program ennek hatására a mypatterns1.list és mypatterns2.list állományokban fogja keresni a GGATCC és TTGANTGA motívumokat. Minimum négy bázisos egyezést keres, ahol a hasonlósági érték 0,5-nél nagyobb. Eredményül visszaadja az azonosítót, a kiterjesztett hasonlósági pontszámot és a megtalált motívum darabot.

A bemeneti motívum adatbázis egy szöveges állomány, aminek minden egyes sora egy azonosítót és egy motívum szekvenciát tartalmaz szóköz karakterrel elválasztva. További oszlopok megadhatóak, de a program nem fogja figyelembe venni a tartalmukat. Itt adhatunk meg egyéb járulékos adatokat, amivel a motívumok az ember számára is könnyen értelmezhetőek lesznek. Egy lehetséges bemeneti állományra mutat példát az 5.1 táblázat.

Az azonosító teszi lehetővé, hogy az eredményfájlból meghatározzuk, mely motívumok feleltek meg a keresési kritériumoknak. Egy lehetséges kimenet részlete látható az 5.2 táblázatban. Az adatok oszlopai szóközzel vannak elválasztva.

Fontos megjegyezni, hogy a `mofext` kizárólag hézag mentes motívumokkal dolgozik. Ez megkülönbözteti a lokális illesztést alkalmazó programoktól, mint amilyen a Blast

ch-13-500-82400906-10	120.00	GGATCC
ch-13-500-82400906-10	120.00	GGATCC
ch-13-500-82400919-21	100.00	ggATcc
ch-13-500-82400927-12	100.00	GGAKCY
ch-13-500-82400953-1	105.00	RGATcC
ch-13-500-82400966-2	120.00	GGATCC
ch-13-500-82500080-10	120.00	GGATCC
ch-13-500-82500080-10	120.00	GGATCC
ch-13-500-82500101-17	120.00	GGATCC

5.2. táblázat. A mofext program lehetséges kimeneti állománya

is. Az algoritmus által használt feldarabolás ilyetén történő megvalósításának egyik oka éppen az, hogy így a program megtalálhatja a motívumokon belül az azonos részeket, még akkor is, ha a teljes hosszában a két motívum nem hasonlít egymásra.

Az algoritmus megvalósítása C programozási nyelven történt, mert az így fordított program megfelelő teljesítményt nyújt, kevesebb erőforrást igényel és könnyebbé teszi más operációs rendszer alá történő telepítést. Épp ezért a forráskód mentes a rendszerfüggő könyvtárak használatától. A próbák során gond nélkül fordítható volt Linux, Solaris és MacOSX rendszerre is. A program forráskódja megtalálható a CD-n.

Habár a program csak egy feldolgozó egységet használ, a motívum adatok elosztásával az összehasonlítás folyamata párhuzamosítható. Ezt a vizsgálatok során többször kihasználtuk, sőt a később bemutatásra kerülő klaszterezés futásidejét is így csökkentettük.

A program felhasználásra került a DoOP adatbázis kiegészítésének szánt DoOPSearch oldalon is. A keresést a hátterében egy `mofext` program valósítja meg. Ennek érdekében a DoOP programozói felület tartalmaz egy olyan Perl nyelven írt osztályt, ami a felhasználó elől elrejtve meghívja a `mofext`et és visszaadja annak eredményét egy Perl objektumban, hogy a programozó tetszőleges további műveletet végezhessen el az eredményen. A DoOPSearch keresőoldalon például a `mofext` által kapott eredményt linkekkel kiegészítve láthatja a felhasználó. A motívum klaszterező alkalmazás az összehasonlítás lépését szintén ezzel a programmal végzi.

A program legérzékenyebb pontja a hasonlósági mátrix. A megfigyeléseink azt mutatják, hogy a legoptimálisabb eredményt akkor kapjuk, ha a mátrix alapjául az Emboss programcsomag ednaful mátrixát választjuk (lásd az 5.3 táblázatban). Az összehasonlító mátrix transzponálható, ezért elég csak felét feltölteni. A jobb összehasonlítás érdekében kiegészítettük az IUPAC kódokat kis betükkel is, amivel azt akartuk jelezni, hogy a motívum azon pozíciójában nem egyeduralkodó az adott nukleotid, de döntő többségében előfordul.

A program tudásának demonstrálásához a Transfac 9.2 adatbázisból kiválasztottunk négy olyan motívumot, amely jól elkülöníthető biológiai szereppel rendelkezik,

15	a	c	g	t	A	C	G	T	M	R	W	S	Y	K	N
a	10														
c	-30	10													
g	-30	-30	10												
t	-30	-30	-30	10											
A	15	-30	-30	-30	20										
C	-30	15	-30	-30	-30	20									
G	-30	-30	15	-30	-30	-30	20								
T	-30	-30	-30	15	-30	-30	-30	20							
M	5	5	-30	-30	10	10	-30	-30	10						
R	5	-30	5	-30	10	-30	10	-30	-30	10					
W	5	-30	-30	5	10	-30	-30	10	-30	-30	10				
S	-30	5	5	-30	-30	10	10	-30	-30	-30	-30	10			
Y	-30	5	-30	5	-30	10	-30	10	-30	-30	-30	-30	10		
K	-30	-30	5	5	-30	-30	10	10	-30	-30	-30	-30	-30	10	
N	-30	-30	-30	-30	-30	-30	-30	-30	-30	-30	-30	-30	-30	-30	10

5.3. táblázat. A mofext program EDNAFul mátrix alapján képzett összehasonlító mátrixa

motívumaik pedig jellegzetesek. Ez a négy kategória: sejtciklus szabályozás, homeosztázis fenntartás, neuron, illetve izom fejlődés.

A `mofext` futtatása az 1.4-es DoOP adatbázis konzervált motívumaival szembe történt, mivel a munka idején ez volt a legátfogóbb és legjobban annotált promóter adatbázis. A hasonlósági szint paramétere 0,9 volt mindegyik futtatás esetén. Az eredményül kapott motívumok GO annotációját átvizsgálva összeszámoltuk, hogy hány esetben kaptunk a kereső motívummal egyező annotációt (5.4 táblázat). Az oszlopok jelentése a következő: a *Szóméret* jelöli, hogy a program mekkora elemi darabokra vágta fel a motívumokat. Az *Összesen* elnevezésű oszlop mutatja, hogy a DoOP-os adatokból összesen mennyi hasonló motívumot talált a program. *Nincs GO*-val jelöltem azon megtalált motívumok számát, melyekhez az adatbázis nem tartalmazott GO annotációt. Ezekről nem lehet egyértelműen eldönteni a gén annotációs adatbázis alapján, hogy valódi találatok vagy fals pozitívok. A *Találat* és *Nincs találat* oszlopok pedig azt jelzik, hogy ahol a motívumokhoz tartozott GO annotáció, azok közül hány annotációjában fordult elő (illetve hiányzik) az első oszlopban feltüntetett kulcsszó. A *Találat* és az *Összesen* oszlopok hányadosa az *Arány*, ami megmutatja a `mofext` által talált motívumok hány százalékát tudtuk GO annotációval igazolni.

Az eredmények jól jelzik a GO annotációval való munka nehézségeit. Ugyanis a kulcsszó hiánya nem jelenti feltétlenül azt, hogy az adott motívumnak nincs szerepe a feltételezett biológiai folyamatban. Ennek eldöntésére figyelembe kellene venni a GO hierarchiában betöltött helyzetét is. Ha a megtalált GO meghatározás valamelyik szülőjében fordul elő a kulcsszó, akkor az egyértelműen amellett szól, hogy a program megfelelő motívumokat talált. Egy ilyen teszt ugyanakkor rendkívül idő és erőforrás igényes, ezért nem végeztük el. Az itt tapasztaltak vezettek el végül az 5.3.3 fejezetben leírtakhoz.

Az eredmények ennek ellenére remekül mutatják a program paramétereinek és a meg-

Kulcsszó	Szóméret	Találat	Nincs találat	Nincs GO	Összesen	Arány
Neuro	9	293	4617	3927	8837	0,033
Muscle	10	179	9453	6137	15769	0,011
Muscle	12	33	1697	1582	3582	0,009
homeo develop	10	525	3653	3202	7380	0,071
homeo develop	14	26	113	74	213	0,122
homeo develop	17	3	12	4	19	0,157
homeo develop growth	10	582	3569	3202	7380	0,078
homeo develop growth	14	29	110	74	213	0,136
homeo develop growth	17	4	11	4	19	0,210

5.4. táblázat. Mofext program tesztelése.

talált eredmények összefüggését. A szóméret növelésével növekszik a program pontossága. Ez legszemléletesebben a 17-es szóméretnél látható. A találati arány itt a legmagasabb.

5.3. Motívum klaszterezés

A `mofext` program motívum összehasonlító képessége lehetővé teszi, hogy az eredményül kapott hasonlósági érték alapján a motívumokat csoportosítsuk. Feltételezések szerint a motívum szekvenciája és a genomban betöltött szerepe között kapcsolat van. Ha ez a feltételezés igaz, akkor a `mofext` által kapott motívum csoportok egy adott biológiai szerep köré fognak tömörülni.

A `mofext` program tervezése folytán alkalmatlan arra, hogy motívumok csoportján páronkénti összehasonlítást végezzen, mert bemenő paraméterei csak kis számú motívumot keresnek egy nagyobb adatbázisban. Szükség van egy burkoló programra, amely a rendelkezésre álló adatokat úgy alakítja, hogy az megfeleljen a `mofext`-nek. Ezt a szerepet a `klaszterezo.pl` látja el (forráskódja a CD-n található).

A program egy motívumokat tartalmazó fájlt kér bemenetnek, és egy adatbázist, amin lefuttatja a kéréseket. Ha ez a két állomány ugyanaz, akkor a motívumok páronként összehasonlításra kerülnek. Meg kell jegyezni, hogy a program védve van attól, hogy egynél többször hasonlítson össze két motívumot. Ezt úgy védtük ki, hogy felhasználtuk a `mofext` összehasonlító mátrixának transzponálhatóságát. Ha A motívumot összehasonlítjuk B motívummal, ugyan azt az eredményt kapjuk, mintha B motívumot hasonlítanánk össze A-val. Ennek segítségével elkerüljük a szükségtelen iterációs lépéseket.

Nagy mennyiségű adat feldolgozásánál felmerülhet az igény, hogy a bonyolult folya-

82000688_1000_E_motif017	AAYAGGGTGT	ARCAGGgtgT	42.00
82400937_1000_E_motif023	CAGGGTGTN	CAGGgtgTA	38.00
82401444_1000_E_motif540	AGCAGGGTGTGG	ARCAGGgtgTAG	47.00
80801051_1000_H_motif153	RGSAGGGTGT	ARCAGGgtgT	38.00
80100596_1000_P_motif026	GGGTGTAGG	GGgtgTAGG	45.00
80100613_1000_P_motif578	GCAGGGTGTA	RCAGGgtgTA	46.00
80100673_1000_P_motif546	ACCAGGGTG	ARCAGGgtg	36.00
80100675_1000_P_motif281	GGGTGTAGG	GGgtgTAGG	45.00
80100768_1000_P_motif469	ACAGGGTGT	RCAGGgtgT	41.00

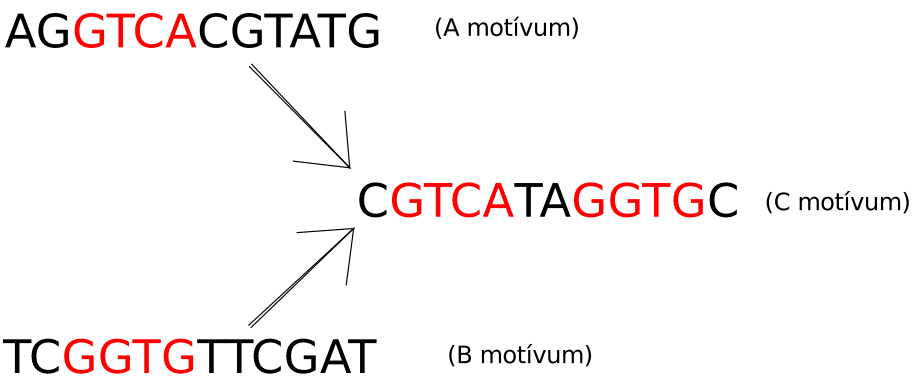
5.5. táblázat. Klaszterezés eredményfájljának részlete. Az első oszlop az egyedi azonosító, ami a vizsgálatainkban a klaszter azonosítójából, a promóter méretéből és az itt található motívum sorszámából áll. A második oszlop a kereső motívum konszenzus szekvenciája. A harmadik oszlop a megtalált motívum konszenzus szekvenciája. Az utolsó oszlop az összehasonlítás pontszáma.

mat több feldolgozó egység között legyen elosztva. A programnak ezt is megadhatjuk paraméterként, így egy többmagos rendszeren a futási idő lerövidül. A program a kereső szekvenciákat tartalmazó listát automatikusan egyenlő darabokra osztja, majd minden egyes szeletre párhuzamosan elindul a keresés. Az eredmény fájlok nevében egyértelmű számozás van, aminek segítségével vissza lehet keresni, hogy melyik folyamatban hányadik motívum volt a kereső szekvencia (lásd. 5.5 táblázat).

Az egyértelmű számozás ellenére az emberi szemnek kaotikusnak tűnik a fenti táblázat, ezért ezt a látszólagos kavalkádot hivatott letisztázni a `mofext_res.pl` program. A nyers eredményeken futtatva, átláthatóbb összképet kapunk. A mozgó ablakos keresési módszer következtében ugyanis különböző keresőmotívumoknak is lehet azonos szekvencia részlete. A másik probléma, hogy a klaszterezésünk során sérül a háromszög egyenlőtlenség. Ez azt jelenti, hogy ha A motívum megtalálja C-t, de B-t nem, viszont B is megtalálja C-t, akkor A, B és C egy klaszterbe tartozik (5.2). (Ha gondatlanul választjuk meg a paramétereket, akkor könnyen az összes motívum egy klaszterbe kerülhet!)

Szigorúan véve a klaszterezési eljárásunk nem a motívumokat rendezi csoportokba, hanem azokat a szekvenciákat, melyek a `mofext` darabolásából származnak. A szóméret megválasztása ezért döntő fontosságú a végeredmény kimenetelére.

Vizsgálataink során a klaszterezés első célpontjai kromatin immunprecipitációs vizsgálatokból származó szekvenciák voltak. Azért esett rájuk a választás, mivel ezek kísérletes adatok, így mentesek a GO annotáció mellékhatásaitól.



5.2. ábra. A háromszög egyenlőtlenség sérülése a motívumok klaszterezésekor

5.3.1. Kromatin immunprecipitáció

Kromatin immunprecipitációval meghatároztuk az ENCODE régióba eső azon kromoszómapozíciókat, melyek hiszton acetilációval vagy metilációval vannak kapcsolatban. A kromoszóma pozíciók alapján az USCS Genome Browser által közreadott többszörös szekvencia illesztések segítségével meg lehet állapítani, hogy más fajokban mely homológ kromoszóma pozíciók felelnek meg az adott helynek.

Az ember (*Homo sapiens*) és más fajok homológ szekvenciáinak kinyerése után, a `dialign` program segítségével megtalálhatóak az evolúciósan konzerválódott szakaszok. Az itt bemutatott eljárás hasonló a DoOP adatbázis készítésénél alkalmazott módszerhez, azzal a különbséggel, hogy a homológ szakaszokat nem BLAST segítségével, hanem kísérleti adatok alapján határoztuk meg.

Ezek a szakaszok képezték az első klaszterezés bemeneti adatait. A klaszterezést elvégeztük több érzékenységi beállítás mellett is a korábban leírt paraméter függés miatt. Ha például a szóméretet 13 bázisra állítottuk, 4283 klasztert kaptunk. Ebből 3462 klaszter csak két elemet tartalmazott. A klaszterezésnél tehát a túl nagy érzékenység nem kedvezett a csoportképződésnek.

Számuk 4708 volt. Miután lefutott a klaszterező algoritmus, eredményül 21 csoport jött létre. Az első csoport 45692 elemet tartalmazott. Ez azért nincs ellentmondásban a 4708 kiindulási számmal, mert a mofext feldarabolja a motívumokat, hogy megtalálja az elemi feldúsulásokat. A `mofext` paraméter érzékenysége az egy klaszterbe került motívumok konszenzus szekvenciáján is érződött. A nagyszámú elemeket tartalmazó csoport például ezeket az elemeket is tartalmazta: CccccTCa, ttTtTTTT. Ez a csoport felfogható egy olyan kategóriának is, ahol azok az elemek találhatóak, melyek egyik klaszterbe sem kerültek. A kis és nagy betűk jelentése megegyezik a mofext mátrix bemutatásánál

leírtakkal.

Miután megkaptuk a csoportokat, a szekvencia hasonlóságán túl szükséges megtudni, hogy mi a közös biológiai szerepük. Ennek megállapításához a GO adatbázist hívtuk segítségül, amit az 5.3.3 fejezetben fogok bemutatni.

5.3.2. DoOP adatbázis motívumai

A második klaszterezési munka már nagyobb szabású volt. Nem kevesebb volt a kitűzött cél, mint a DoOP adatbázisban található valamennyi motívum klaszterezése a saját fejleszésű eljárással, hogy funkcionális csoportokba sorolhassuk őket. Ez komoly technikai problémákat vetett fel, hiszen az adatbázis több, mint 4 millió motívumot tartalmazott. A program jelenlegi formájában, négy számítási egység felhasználásával előzetes számítások szerint 113 évig futott volna.

A megoldásnak mindenképp az adatok indexelése látszott, ugyanis felesleges gépidőt fordítani olyan keresésekre, melyek biztosan nem hoznak eredményt. Ezért a teljes motívum készletet öt bázispáros átfedő darabokra bontva tároltuk minden egyes oligonukleotid helyzetét. A `mofext`nek ezek után csak azokat a motívumokat kellett összehasonlítania, melyek közös oligonukleotidokat tartalmaztak. Ezzel szükségtelenné vált a `mofext` csúszó ablakos összehasonlító lépése, és csak azt kellett vizsgálnia, hogy a hasonlóság kiterjeszthető-e.

A programnak, mely ezt a feladatot elvégezte, `moind` lett a neve. Az indexet a számítógép memóriájában tárolta, ami 4 millió motívumnál meghaladta a szerverünk 16 gigabájtos kapacitását.

Ez volt az oka, hogy a klaszterezés nem a teljes gerinces csoporton hajtottuk végre, hanem kisebb taxonokon. Ezek az evolúciós csoportok úgy lettek kiválasztva, hogy lehetőleg azonos evolúciós távolságra legyenek egymástól.

A felosztás megtekinthető az 5.6 táblázatban. A motívumok száma így is százezres nagyságrendre duzzadt, ahogy a vizsgált evolúciós csoport az emberhez egyre közelebb és közelebb került. A *Primates*, *Euarchontoglires*, *Eutheria* csoportok nem kerültek összehasonlításra, mert motívumszámuk egyedül is olyan nagy, hogy a rendelkezésre álló számítási kapacitás az analízisüket nem tette lehetővé.

Releváns következtetéseket a többi csoport analíziséből is le lehet vonni. A motívumok minél nagyobb evolúciós távolságot ölelnek át, méretük annál rövidebb, és konszenzus szekvenciájuk annál degradáltabb.

A motívumok száma is kevesebb lesz a taxon csoportokban, ahogy távolodunk a *Primates* osztálytól. A motívum generálási eljárásunk egyik következménye, hogy minél ki-

Csoport rövidítése	Taxon	Homológ illesztések száma	Átlagos motívum szám egy homológ illesztésben	Átlagos motívum hosszúság
C	*Chordata*	9	7,67	6,0290
V	*Vertebrata*	2	10	8,1
F	*Teleostomi*	610	14,79	10,0556
T	*Tetrapoda*	580	15,06	10,2737
N	*Amniota*	1122	15,35	10,0960
M	*Mammalia*	37	26,89	11,6784
H	*Theria*	3841	26,31	9,8717
E	*Eutheria*	14310	51,17	9,4140
R	*Euarchontoglires*	13871	62,96	9,2999
P	*Primates*	21051	112,96	14,8324

5.6. táblázat. Gerinces csoportok és a bennük található motívumok statisztikai jellemzői

sebb evolúciós távolságból álló élőlényekből készül el a szekvencia illesztés, annál több motívum található a végeredményben. Ez legszembetűnőbben a *Primates* osztálynál figyelhető meg, ahol átlagosan 113 konzervált szakasz jön létre a szabályozó régiók illesztéséből.

Habár a `mofext` a motívum összehasonlítás terén jól teljesít, az eredmények azt támasztják alá, hogy klaszterezésre alkalmatlan, aminek oka, hogy az eredményül adott mérőszámot nem lehet a hasonlóság mérőszámának tekinteni. Ez és a háromszög egyenlőtlenség következményeként diszkrét csoportok helyett egyfajta motívum grádiens jött létre. Azonban nincs kizárva, hogy további fejlesztések segítségével ezek a problémák kiküszöbölhetőek.

A motívum klaszterezés nehézségét mutatja, hogy az EnsEMBL részét képező cisRed adatbázis is komoly számítástechnikai erőforrásokat vonultatott fel a probléma megoldására. Az eredményül kapott motívumoknál náluk is megfigyelhető a különböző hosszúságú elemek összehasonlításának hibája (Robertson et al., 2006).

A cisRed ugyanis egy módosított Levenstein távolságon alapuló klaszterezési eljárást használ, ami nem engedélyezi a réseket (gap) a motívumok konszenzus szekvenciájában. A módszer remekül teljesít, ha az összehasonlítandó motívumok azonos vagy közel azo-

nos hosszúságúak, ellenben rossz hasonlósági értéket ad, ha a motívumok hosszúsága jelentősen eltér.

5.3.3. Gén ontológiai analízis

Akármelyik kísérletből is kaptuk meg a motívum csoportokat, azokhoz hozzá kell rendelni a megfelelő biológiai szerepet. A dolgozat írásának időpontjában - hibái ellenére - is a legjobban használható adatbázisnak a gén ontológiai adatbázist találtuk.

A motívumok biológiai szerepei úgy kerültek megállapításra, hogy annak a génnek a funkcióit rendeltük hozzájuk, melynek a promóter szekvenciájában előfordulnak. A DoOP adatbázis tartalmazza a gének GO azonosítóit, ezért ezt a műveletet is egy programra lehetett bízni.

A klaszterezés befejezése után az egy klaszterbe kerülő motívumokhoz könnyen tudtunk rendelni biológiai szerepet. A nagyszámú, különböző GO azonosító közül ki kell választanunk azokat, amelyek szignifikánsan gyakrabban fordulnak elő a csoportban. A szakirodalom egyetért abban, hogy a hipergeometrikus eloszláson alapuló statisztika a legmegfelelőbb erre a célra.

A rendelkezésre álló nagyszámú program közül a GeneMerge-re esett a választás, mert képes hipergeometrikus eloszlást számolni nagyszámú GO azonosító felhasználásával. Emellett parancssoros, ezért könnyű felhasználni a folyamatot automatizáló szkriptekben.

Sajnos a hatékonysága alacsony volt, mivel feltételezhetően kis memóriával ellátott gépekre tervezték, és szükségtelenül sok fájlműveletet végzett. A forráskód ismerete és a GNU licenc viszont lehetővé tette, hogy a céljainknak megfelelően átírjuk. A fent vázolt igényekhez igazított GeneMerge program forráskódja megtalálható a CD-n.

A program négy fájlt vár bemenetnek. Az első az asszociációs fájl, ami a motívum és a gén ontológiai azonosító közötti kapcsolatot írja le. Pontosvesszővel elválasztva több gén ontológiát is megadhatunk. A második a gén ontológiai azonosító leírását tartalmazza. A harmadik fájl a vizsgált adatsort teljes készletét tartalmazza, jelen esetben a klaszterezés bemeneti adatsorát. Az utolsó fájl a vizsgálni kívánt klaszter motívumai. Eredményül egy GO azonosítókat tartalmazó listát kapunk.

A klaszterezésünk eredményeként képződött csoportok egyikében sem találtunk olyan szignifikánsan feldúsuló GO funkciót, ami csak egyetlen klaszterre lett volna jellemző. Ennek lehet az is az oka, hogy a klaszterező módszerünk túl sok fals pozitívot adott eredményül.

Némi reménnyel kecsegtetnek a dolgozat alapját képező vizsgálatok lefolytatása után

napvilágot látott publikációk. Egyre erősebb bizonyítékok támasztják alá, hogy egy génhez több alternatív promóter is tartozhat, melyek mind-mind befolyásolják a gén kifejeződését. Az alternatív promóterek akár száz kilóbázis távolságra is előfordulhatnak a transzkripció start helyétől. Szerepük a génkifejeződés térbeli és időbeli elkülönülésének biztosítása.

A DoOP adatbázis nem tartalmazza az alternatív promótereket. Ez egy későbbi fejlesztés részét képezi. Amennyiben a vizsgálatokat ki lehet terjeszteni erre az eddig ismeretlen területre, jobban meg lehet érteni a motívumok és a génszabályozás kapcsolatát. A feltételezhetően itt található motívumok segítségével növelhető lenne a vizsgálatok statisztikai szignifikanciája.

5.4. Motívum keresés

A funkcionális motívumok felkutatásának másik módszere, a *de novo* motívum keresés. Nagy számban elérhetőek olyan algoritmusok, melyek a beadott szekvenciák között túlreprezentált oligonukleotidokat keresnek. Mi a `NestedMica`-t használtuk.

Munkánk egyik célja az volt, hogy olyan motívumokat találjunk, melyek meghatározott biológiai szereppel jellemezhető gének szabályozó régiójában fordulnak elő. Vizsgálatainkhoz tehát szükség volt olyan génekre, melyek biztosan azonos funkciókat látnak el. Ehhez a chip kísérletek adnak megfelelő alapanyagot.

Szatmári István chip kísérletei a monocita-dendritikus sejt átalakulást vizsgálták roziglitazon hatására. Az RSG kezelés után 6 óra, 24 óra, 5 nap elteltével mintákat vettek, majd expressziós vizsgálatnak vetették alá.

A géneket az expressziós változások alapján csoportokba lehet rendezni aszerint, hogy az RSG kezelés hatására növekedett vagy csökkent az expressziójuk. A minimum kétszeres expressziós változást figyelve elkülöníthető minden időintervallumra egy olyan csoport, melynek az expressziója erősödik, és egy olyan, melynél az expresszió szintje csökken.

Abból kiindulva, hogy ezeknek a géneknek a szabályozásában közös mechanizmusok játszanak szerepet, el lehet kezdeni a promóterükben közös motívumokat keresni. A GeneSpring eredményfájljai alapján az EnsEMBL segítségével egy adott gén transzkripciós starthelyétől 5' irányban 10 kbp hosszú szekvenciákat szedtünk ki, valamint az első intront, mivel szabályozó elemek itt is előfordulhatnak.

Korábbi publikációk arról számoltak be, hogy a retinoid receptor a monocita érés szabályozásában érintett, ezért az első keresések célpontja ennek a konszenzus szekvenciája volt. A retinoid receptor heterodimer formában fordul elő, a két dimer között változó

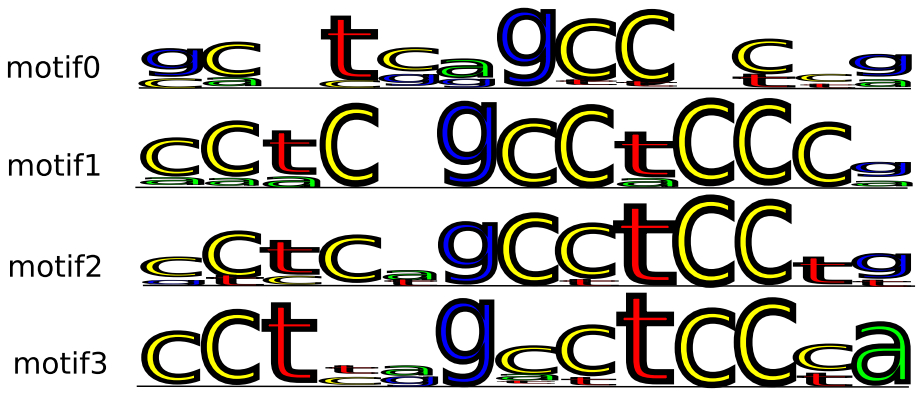
hosszúságú linker régió található. A vizsgálatok során az AGSTCMN(1,7)AGSTCM konszenzus szekvenciákat kerestük. Jelölésük a linker régió hossza alapján DR1-DR7 volt. Célunk az volt ezzel a jól definiált elemmel, hogy egyfajta kontrollja legyen a motívum kereső módszerünknek.

A kereséshez a legmegfelelőbb programnak az `EMBOSS` programcsomag részét képező `fuzznuc` bizonyult. A programnak 0-2 mismatch lett engedélyezve. Az 5' és a 3' szálon egyaránt történt keresés. Kontrollnak további két adatsort készítettünk. Az első a chip kísérletekből kapott szekvenciákat tartalmazta a `shuffleseq` programmal megkeverve. Ez a program is az `EMBOSS` programcsomag része. Mivel az eredeti adatsorból származtak, nukleotid arányuk megegyezett azzal. A második összehasonlító adatsor véletlenszerűen kiválasztott gének 5' régióját tartalmazta 10 kbp hosszúságban. A vizsgált adatsorral nem volt átfedése ezen készletnek. Számuk megegyezett a microarray kísérletből kapott adatok számával, hogy a statisztikai kiértékelést könnyebben elvégezhessük.

A `fuzznuc` program az összes DR elemet megkereste mindegyik adatsorban, valamennyi mismatch értékkel. A szabályozó régió szekvenciáját felbontottuk 200 bázispár hosszú ablakokra, 50 bázispár hosszú átfedéssel. Mindegyik ablakban összeszámoltuk a megtalált DR elemek számát, majd a számítást megismételtük a véletlenszerűen kiválasztott és a kevert nukleotidokat tartalmazó adatsoron is. Arra kerestük a választ, hogy az ablakokban összeszámolt átlagos motívum előfordulás mutat-e eltérést. Az eredmények kiértékelését t-próbával végezve a következő eredmények jöttek ki (α `<= 0,05`): Az összekevert szekvenciáktól minden esetben tapasztalható volt eltérés, de a véletlenszerűen kiválasztott szekvenciákhoz képest nem. Ez alapján levonható az a következtetés, hogy a DR elemnek van szabályozó szerepe, de az valószínűleg sokkal általánosabb és nem köthető kizárólag a chip kísérletekből kinyert génekhez (Szatmari et al., 2007). A fent vázolt módszerrel ellenben gyorsan ítéletet mondhatunk minden újonnan megtalált transzkripciós faktor kötőhelyről.

Következő kérdésünk tehát, hogy van-e olyan elem a vizsgált szabályozó régiókban, melyeket még nem írtak le? A kérdés megválaszolásához szükség van a `NestedMica`-ra. A program bemenetét a chip kísérletekben meghatározott gének szabályozó régiói képezték. Az ott leírt módszerekhez képest csak annyi változtatás történt, hogy 30 ezer bázispárnál hosszabb szekvenciákat csonkoltuk, mert a program memóriakezelése nem tette lehetővé ilyen méretű bemeneti adat vizsgálatát. A szekveciákból a `censor` program segítségével eltávolításra kerültek a repetitív elemek és az alacsony komplexitású régiók.

Az első futtatások nem szolgáltak használható eredménnyel. A `NestedMica` nem talált értékelhető motívumot. A második lépés során a szekvenciák homológ régióit is kinyertük.



5.3. ábra. A DRA motívum szekvencia logója

Az EnsEMBL-ben minden szekvencia esetén letárolták a hozzájuk tartozó ortológ és paralóg régiókat. Egy saját fejlesztésű programmal kinyertük a vizsgálatban felhasznált emberi referencia szekvenciához illesztett ortológ szekvenciák közül azokat, melyek más fajban csak egyetlen régió szekvenciájával feleltethetőek meg. (Tehát nem voltak paralógjai) Az adatbázisban ezek 1-1 ortológ néven szerepelnek. A program neve `homo_11_megf.pl`, forráskódja a CD-n található.

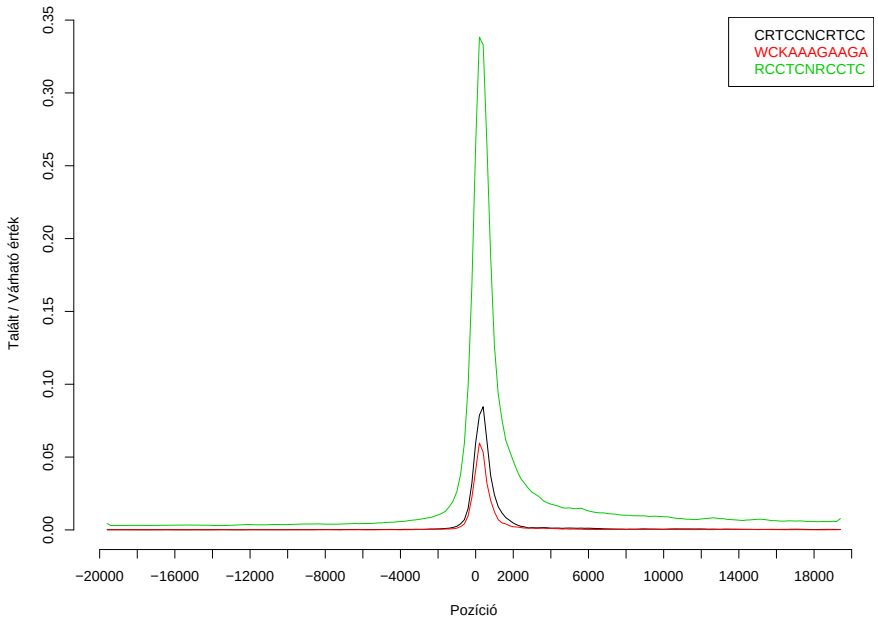
A `NestedMica` ennek segítségével már nem csak a humán, hanem a szarvasmarha (*Bos taurus*), kutya (*Canis familiaris*), egér (*Mus musculus*), patkány (*Rattus norvegicus*) szekvenciákat is megkapta bemeneti adatként, és így ki lehetett használni igazi erősségét, az evolúciós távolságon alapuló motívum keresést. Az ortológ szekvenciák között szándékosan nem szerepel egyetlen főemlős sem. Ezen fajok szekvenciájái rendkívül hasonlóak az ember genomjához, ezért torzítják az eredményeket.

Természetesen itt is eltávolításra kerültek a repetitív elemek és az alacsony komplexitású régiók. A `censor` program adatbázisában több faj ALU ismétlődése is szerepel, ezért nem fordulhatott elő az az eset, hogy nem megfelelő szekvenciákat eltávolítottunk.

Az így kapott eredmények közül, a 6 órás indukálódó gének listáján kitűnt egy igen érdekes szekvencia, mely a DR-hez hasonlóan heterodimer formát mutatott. Konszenzus szekvenciája: RCCTCNRCCTC. A munka ezen fázisában a DRA jelölést kapta, a linker régió méretét itt is egy szám reprezentálta. A motívum szekvencia logója az 5.3 ábrán látható.

Ezen kívül még két motívum tűnt érdekesnek. Külön nevet nem kaptak, konszenzus szekvenciájuk: WCKAAAGAAGA, illetve CRTCCNCRTCC. A NestedMica gyengébbnek találta ezt a két szekvenciát, csak az összehasonlítás kedvéért tartottuk meg őket.

A DRA-val meg lehetett ismételni a `fuzznuc`-os keresést a kevert és véletlenszerűen



5.4. ábra. A DRA előfordulásának valószínűsége a TSS-hez képest

kiválasztott gének szabályozó régiójában. Legnagyobb meglepetésre a DR-el egyező eredmény jött ki. Vagyis csak a kevert szekvenciájú adatsorhoz képest tudtunk szignifikáns eltérést kimutatni. A Transfac és Jaspar adatbázisokban nem találtunk ilyen motívumot.

A motívumok TSS-hez viszonyított elhelyezkedését vizsgálva érdekes összefüggést vettünk észre (5.4 ábra). A vizsgált szekvenciarészletek nagysága miatt, az ábrázolást könnyítendő átfedő ablakos módszerrel táblázatot készítettünk a motívumok előfordulásának számáról. A tapasztalat alapján az ablakok méretét 1000 bázispárra állítottuk, az átfedés pedig 200 bázispár. Minden egyes ablakra kiszámítottuk a DRA elem várható előfordulásának nagyságát is. A tényleges előfordulást osztva ezzel az értékkel megkaptuk az előfordulás valószínűségét. Az előfordulás maximuma a TSS-el megegyezik, míg attól távolodva ez az érték lecsökken. Ez azért is meglepő, mert több 30 kbp méretű szekvenciát vizsgáltunk. Ha ez egy új, ismétlődő szekvencia lenne, az előfordulások diffúz képet mutatnának.

Mivel a random listákhoz képest a motívumok száma nem változik, valamint a TSS környékén feldúsul a mennyisége, elképzelhető, hogy a motívum szerepet játszik a TSS kijelölésében is. Az eredményeket némiképp árnyalja a tény, hogy ha a szekvenciák ismértlődéseit a RepeatMaskerrel távolítjuk el, a TSS-hez viszonyítva nem találunk ilyen kiugró eredményt. Az annotáció szerint az általunk talált motívum feltűnő hasonlóságot

mutat egy SINE családba tartozó nem virális transzpozon darabjához. Amennyiben a kísérletes vizsgálatok ezt az eredményt támasztanák alá, ez nem az alkalmazott módszer hibája, hanem a felhasznált adatoké. Csupán egy újabb ismétlődéseket tartalmazó adatbázist kell felhasználni, hogy a NestedMica ne futhasson tévútra.

5.5. Kromatin immunprecipitáció

Egy másik módszerrel is összegyűjtöttünk olyan szabályozó régiókat, melyek azonos biológiai szereppel bírtak. Kromatin immunprecipitációval a kinyerhető kromoszóma szakaszok nem korlátozódnak a génkódoló régiókra, mint a chip kísérleteknél, tehát lehetőségünk van közvetlen, a szabályozó régióban elhelyezkedő célpontokat találni.

A laboratóriumi vizsgálatok célja az volt, hogy feltérképezzük az ENCODE régióba eső hiszton acetiláció és metiláció mértékét a differenciálódó HL60 sejtvonalban. A hisztonvég kovalens módosításáért a szöveti transzglutamináz felel, aminek proximális promótere tartalmaz egy retinsav receptort kötő elemet. Az enzim ennek hiányában olyan alacsony expressziós szintet mutat, ami az érzékelhetőség határa alatt van. A HL60 mieloid sejtek differenciációját DMSO kezelés segítségével indították el. A DMSO kezelés nélküli minták a naiv jelölést kapták.

A kísérletek által relevánsnak nyilvánított régiók szekvenciáját az EnsEMBL genom adatbázisból ki lehet nyerni a kromoszóma pozíciók segítségével. Az így kapott régiókat szintén alá lehet vetni a korábban leírt motívum kereső módszereknek. Mivel az ENCODE régió a vizsgálatok ideje alatt még csak az emberi genom egy százalékát tette ki, ezért a kapott motívumok klaszterezése nem adott volna releváns információt.

`Fuzznuc` segítségével ezekben a régiókban is megkerestük az ismert retinoid receptor kötő szekvenciát legfeljebb 2 bázis csere engedélyezésével. Kontrollként kiszedtünk olyan kromoszóma szakaszokat is, melyek hossza megegyezett a vizsgált régiókkal, de nem volt átfedése azokkal. Kiszámoltuk a 100 bázispárra eső átlagos találatok számát, majd a t-próbával szignifikáns eltéréseket kerestünk (`p <= 0,05`) a kontroll adatsorhoz képest. Ilyen eredményt egyik DR elem esetében sem találtunk. Ez egybevág a korábbi feltételezésünkkel, amely szerint a DR elem általánosabb szabályozó feladatot lát el.

További lehetőség az acetilációs és metilációs régiók és a transzkripcós kezdő pont közötti távolság vizsgálata. A TSS adatbázis tartalmazza a transzkripciós kezdő pontok genomi pozícióit. Ezért a TSS-ektől +- 5 kilóbázis távolságon belül található összes acetilációs illetve metilációs pont helyzetét összegyűjtöttük. Ezt az összesen 10 kbp nagyságú régiót egy ún. csúszó ablakos módszerrel 200 bázispáros szakaszokra osztottuk

50 bázispáros átfedéssel. Összeszámoltuk, hogy egy ablakba hány acetilációs és metilációs pont jut, majd ezt ábrázoltuk. Az ábrák érdekes eredményeket sejtetnek: A TSS-től való távolság nem véletlenszerű, hanem szisztematikus.

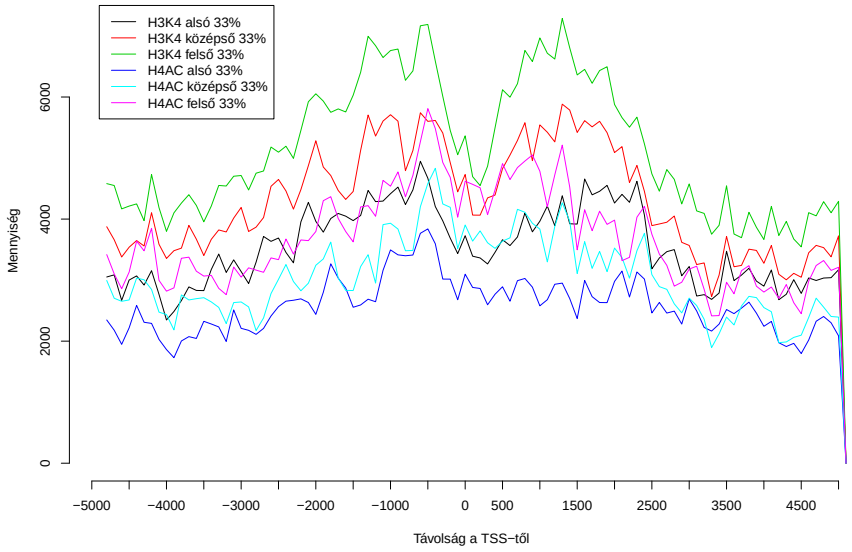
A kromatin immunprecipitációs vizsgálatok szűrése vált szükségessé, hogy az eredmények még árnyaltabbak legyenek. Reménykedtünk benne, hogy a szűrés hatására találunk egy olyan folyamatot, mely hasonló karakterisztikával bír, mint a nyers adatok, mégis kevesebb zajt tartalmaz. Ennek érdekében megkerestünk minden acetilációs és metilációs szakaszhoz tartozó gént. Ha a kérdéses szakaszt egy gén exonjában vagy intronjában találtuk, könnyű volt a dolgunk, ha viszont intergenikus régióban bukkant fel, akkor a tőle 3' és 5' irányba található gének orientációjától függően azt a gént rendeltük hozzá, amelynek a TSS-étől 5' irányban volt. Ha két gén is megfelelt a kitételnek, akkor a közelebbit választottuk. Ha ez a távolság több volt, mint 10 kbp, akkor a szakaszt kihagytuk a vizsgálatból, mert nem lehetünk teljesen biztosak benne, hogy melyik génhez tartozhat. A saját fejlesztésű program, amivel a leírtakat megvalósítottuk, a `affygenecage.pl` volt.

Rendelkezésre álltak olyan vizsgálati eredmények is, amelyek ugyancsak a retinsav kezelés hatását vizsgálták a HL60-as sejteken, de génexpressziós chip-el. A kísérlet felépítése megegyezett az 5.4 fejezetben leírtakkal, az egyetlen különbséget a sejtdifferenciáció elindításához felhasznált anyag adta. A vizsgálat eredménye ebben az esetben is egy gén lista volt, ami összevethető a kromatin immunprecipitációs eredményekkel. A két kísérlet eredményeinek összevonása lehetővé tette, hogy a kromatin immunprecipitációs eredményeket a génexpresszió alapján szűrjük.

A két különböző jellegű kísérletes vizsgálat bioinformatikai összekapcsolása újszerű megközelítés.

A jelintenzitás értékeket a normalizált expressziós szint nagysága szerint sorba rendeztük, majd egyenlő mértékben elosztottuk oly módon, hogy a felső 33 százalék fokozott, míg az alsó 33 százalék csökkent aktivitást mutató, a maradék a nem változó jelölést kapta. A felosztás hátterében az állt, hogy a korábbi kísérletben alkalmazott 10-80-10 százalékos felosztással a fokozott és csökkent aktivitást mutató gének száma olyan alacsony lett, hogy a további statisztikai elemzések nem adtak volna szignifikáns eredményt. Ezzel a lépéssel igaz, megnőtt a fals pozitív gének száma, de mivel a chip kísérleteket nem önmagukban, hanem kromatin immunprecipitációval együtt használtuk, ezért az eredő fals pozitív hibaarányt nem növeltük.

A szűrés segítségével mind az acetilációs, mind a metilációs lista három részre szakadt annak függvényében, hogy az expressziós vizsgálatok szerint milyen volt az mRNS szintje.



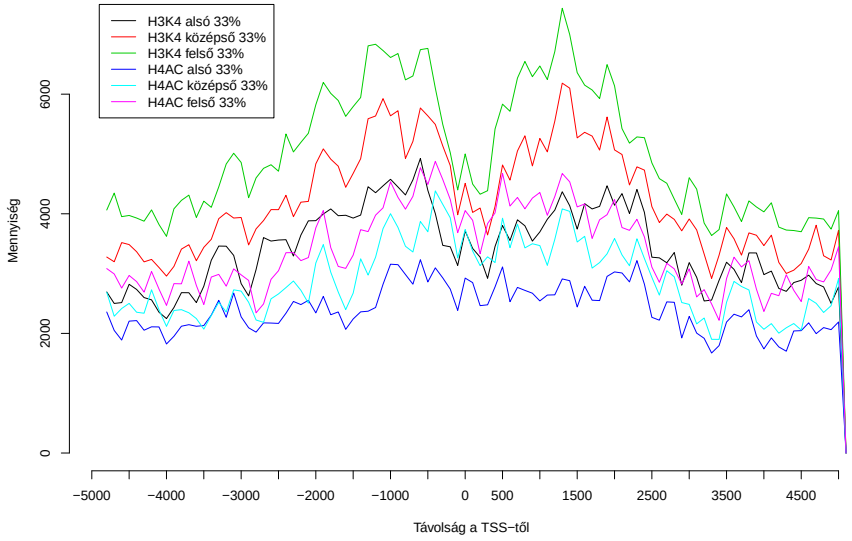
5.5. ábra. A naiv kromoszómapozíciók a TSS-hez viszonyítva, expressziós szint szűrés után.

A szűrés után nem láttunk különbséget az egyes listákban, a TSS-függő változás mindegyikben megjelent, szabad szemmel észrevehetetlen a különbség (5.5 és 5.6 ábrák). A naiv sejtek itt is a retinsav kezelés nélküli kontroll adatsornak tekinthető.

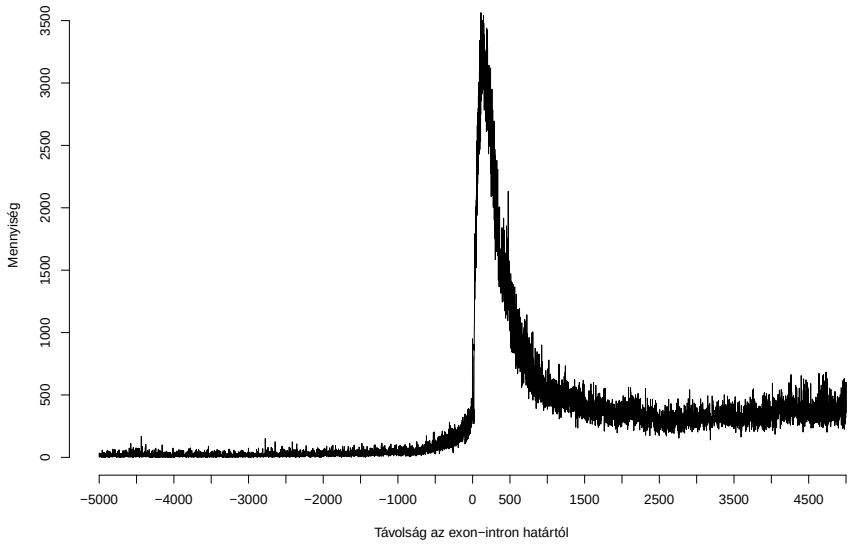
Ezért a kérdés további vizsgálatához más módszerek alkalmazását terveztük. Több publikáció is beszámolt róla, hogy az exon-intron határok szerepe fontos a génszabályozásban. A következő vizsgálatok annak kiderítésére irányultak, hogy az acetilációs és metilációs pontok előfordulása független-e az exon-intron határok kromoszóma pozícióitól.

Az EnsEMBL adatbázis segítségével megállapítottuk a korábban meghatározott gének összes exon-intron határának kromoszóma pozícióját, majd - akárcsak korábban a TSS-ek vizsgálatánál - kiszámítottuk a távolságukat az acetilációs és metilációs pontoktól. A fent említett ablakos módszerrel az előfordulások számát ábrázolva az exon-intron határoktól való távolság függvényében, az tűnhet fel, hogy az elemek a határok pozíciójában fordulnak elő legnagyobb mennyiségben, és számuk az intronban magasabb, mint az exonban (5.7 ábra).

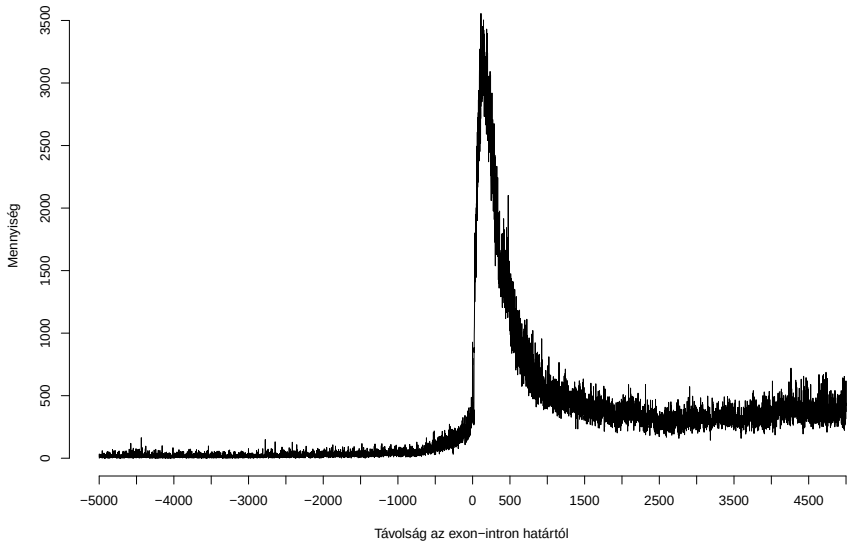
Az eredményeket a naiv sejtek vizsgálatával validáltuk (5.8 ábra). Elvégeztük az adatok szűrését az expressziós chip segítségével, de a grafikon alakját nem változtatta meg, csak az adatok számát csökkentette. Az exon-intron határok szerepe tehát nem lehet folyamat specifikus, hiszen a kontroll adatsorban is ugyan olyan változást láthatunk, mint



5.6. ábra. A retinoid kezelés kromoszóma pozíciói a TSS-hez viszonyítva, expressziós szint szűrés után.



5.7. ábra. A retinoid kezelt sejtekben a metilációs pontok távolsága az exon-intron határokhoz képest



5.8. ábra. A naív sejtekben a metilációs pontok távolsága az exon-intron határokhoz képest

a retinoid kezelésesben.

6. fejezet

Összefoglalás

A gének funkciója és promóter régiójukban található transzkripciós kötőhelyek konszenzus szekvenciája között kapcsolatot találni nem könnyű feladat. Első lépésként a bioinformatikai hátteret kell felépíteni, amit csoportunk a DoOP adatbázissal, `mofext` programmal és a kettő ötvözetének tekinthető DoOPSearch weboldalla valósított meg. Ezek az eszközök nem kutatás specifikusak. Bármilyen más promóter és konzerválódott motívum vizsgálat eszköztárába felvehetőek.

Második lépésként hasonló motívumokat gyűjtöttünk össze, majd közös funkciókat kerestünk a hozzájuk tartozó gének között. A közös biológiai szerepet a gén ontológiai adatbázis segítségével állapítottuk meg, és hipergeometrikus eloszláson alapuló módszerrel határoztuk meg annak mértékét. Az így keletkezett csoportok klaszterezése meghaladta a rendelkezésünkre álló erőforrásokat, ezért kisebb evolúciós csoportokon alkalmaztuk csak.

A cél érdekében megfordítottuk a módszereket. Nem a motívumok felől jutottunk el a gének felé, hanem a hasonló biológiai funkcióval rendelkező gének szabályozó régiójában kerestünk közös elemeket. A géneket chip kísérletekből kaptuk meg. Közös tulajdonságuk, hogy PPARγ-t tartalmaznak, ami bizonyítottan a lipid metabolizmusban szerepet játszó transzkripciós kötőhely. Kidolgoztunk egy módszert, amivel a chip kísérletekben pozitív géneket lehet vizsgálni. Bizonyítottuk, hogy a fokozott intenzitást mutató gének szabályozó régióiban nem fordul elő statisztikailag feldúsulva a DR1 a véletlenszerűen kiválasztott génekhez képest. A génlistán végrehajtott *de novo* motívum keresések segítségével sikerült azonosítani új elemeket, de biológiai szerepük nem tisztázott.

Sokkal jobb eredményt ad a motívumok és a transzkripciós kezdőpont távolságának elemzése. Itt arra az eredményre jutottunk, hogy a TSS közelében nagyobb a motívumok előfordulásának valószínűsége. Ezek alapján a szabályozást nemcsak a motívum jelenléte vagy hiánya szabja meg, hanem a szabályozó régióban betöltött pozíciója is. Ezt más pub-

likációk is alátámasztják (Moses et al., 2003), (Berendzen et al., 2006), (Vardhanabhuti et al., 2007).

Ugyancsak a pozíció specifikus jelenlét egyik közvetett bizonyítékaként tekinthetünk az exon-intron határok és az acetilációs illetve metilációs pontok távolságának összefüggésére is. Ugyanis a határpontok maximuma az exon-intron határtól 154 bázispár távolságra található. Tehát bármilyen kapcsolat is legyen az intronok kezdőpontja és az acetilációs, illetve metilációs pontok között, az nem független kettejük távolságától.

A transzkripciós szabályozásról kialakult kép még közel sem teljes. Nem tudjuk meddig terjednek a szabályozó régiók határai, esetleg nincs-e olyan genomi struktúra, ami szintén befolyásolhatja a génregulációt.

Bizonyított, hogy a térszerkezet is konzerválódhat. Különböző, egymáshoz nem hasonló szekvenciák képesek felépíteni hasonló térbeli szerkezetet, amit a szabályozó elemek hasonlóként ismernek fel(Parker et al., 2009). A bioinformatikai vizsgálatokhoz ezen ismereteket is fel kell használni, hogy az eredmények egyértelműbbek legyenek.

7. fejezet

Summary

Finding connection between gene function and consensus sequence of TFBS of its promoter region is not an easy task. First of all, a new bioinformatic background was needed. It was achived by the creation of the DoOP database, the `mofext` program and by the combination of these two in the DoOPSearch webpage. These tools are not specific for a certain investigation, they can be used for any other promoter and conserved motif researches.

As a second step, we collected motifs that were similar to each other and tried to search for common functions in their genes. We have used a gene ontology database to identify the common biological functions of the genes and a method based on hypergeometric distribution was used to determine the percentage of similarity. Clustering of these groups was beyond our strength, so we have used it only in groups of animals that are in a small evolutionary distance from each other.

For the sake of the cause we reversed the methods. Instead of collecting genes of similar motifs, we tried to search common control elements in the genes with similar biological function. Gene collections were obtained from chip experiments. All the collected genes contained PPARγ, a transcription factor binding site that proved to participate in lipid metabolism. A new method was developed to analyze the genes with positive responses in chip experiments. We proved that the promoters of overrepresented genes did not contain statistically more DR1 than the randomly selected promoters. With the help of *de novo* motif searching, new elements were identified from the gene list. However, their biological role is still not clear.

Analysis of the distance between the transcription start site and the motifs was more succesfull. It was shown that the probability of occurence of the motifs is increased nearby the TS-Sites. These results suggests that the regulation depends not only on the presence

or absence of the motif, but also on its position in the promoter. Other publications also supports this suggestion (Moses et al., 2003), (Berendzen et al., 2006), (Vardhanabhuti et al., 2007).

Another indirect evidence of the position specific presence can be the relationship between the positions of the exon-intron junctions and the acetilation or metilation points. This distance between the maximal number of junctions and the acetilation or metilation points proved to be 154 basepairs. Without understanding the nature of the relation between the first base of the intron and the acetilation or metilation points, it can be concluded that the linkage dependends on their distance from each other.

The complex picture of the transcription regulation is far from completely understood. For example it is still unclear how far the boundaries of the promoters extend or whether there is any genomic structure that can also influences the gene regulation.

It is proved that the topology of DNA can also be conserved. Different sequences can produce similar three-dimension topologies that can be recognized by elements of the regulation (Parker et al., 2009). These peaces of information shoud also be integrated in the bioinformatic analysis to get more appropriate results.

Irodalomjegyzék

Stephen F. Altschul, Thomas L. Madden, Alejandro A. Schaffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J. Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, 1997.

Timothy L. Bailey and Charles Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 28–36. AAAI Press, 1994.

Tanya Barrett, Dennis B. Troup, Stephen E. Wilhite, Pierre Ledoux, Dmitry Rudnev, Carlos Evangelista, Irene F. Kim, Alexandra Soboleva, Maxim Tomashevsky, Kimberly A. Marshall, Katherine H. Phillippy, Patti M. Sherman, Rolf N. Muertter, and Ron Edgar. Ncbi geo: archive for high-throughput functional genomic data. *Nucleic Acids Research*, 37(10):885–890, 2009.

Endre Barta, Endre Sebestyen, Tamas Balint Palfy, Gabor Toth, Csaba Ortutay, and Laszlo Patthy. Doop: Databases of orthologous promoters, collections of clusters of orthologous upstream sequences from chordates and plants. *Nucleic Acid Researcher*, 33(1):86–90, Jan 2005.

D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and E. W. Sayers. Genbank. *Nucleic Acids Research*, 38(1):46–51, 2010.

K. W. Berendzen, K. Stüber, K. Harter, and D. Wanke. Cis-motifs upstream of the transcription and translation initiation sites are effectively revealed by their positional disequilibrium in eukaryote genomes using frequency distribution curves. *BMC Bioinformatics*, 30(7):522, 2006.

Jennifer E. F. Butler and James T. Kadonaga. The rna polymerase ii core promoter: a

key component in the regulation of gene expression. *Genes Development*, 16:2583–2592, 2002.

Nigel P. Carter and David Vetrie. Applications of genomic microarrays to explore human chromosome structure and function. *Human Molecular Genetics*, 13(13):297–302, 2004.

C. I. Castillo-Davis and D. L. Hartl. Genemerge–post-genomic analysis, data mining, and hypothesis testing. *Bioinformatics*, 19:891–892, 2003.

CPAN weboldal. http://cpan.org/ tibi/doop-1.02.tar.gz.

Raman V. Davuluri, Yutaka Suzuku, Sumio Sugano, Christoph Plass, and Tim H.-M. Huang. The functional consequences of alternative promoter use in mammalian genomes. *Trends in Genetics*, 24(4):167–177, 2008a.

Ramana V. Davuluri, Yutaka Suzuki, Sumio Sugano, Christoph Plass, and Tim H.-M. Huang. The functional consequences of alternative promoter use in mammalian genomes. *Cell*, 24(4):167–177, 2008b.

T. A. Down and T. J. Hubbard. Nestedmica: sensitive inference of over-represented motifs in nucleic acid sequence. *Nucleic Acids Res.*, 33(5):1445–53, Mar 2005.

ENCODE Project Consortium. Identification and analysis of functional elements in 1% of the human genome by the encode pilot project. *Nature*, 7146(447):799–816, 2007.

R. Evans, J. A. Fairley, and S. G. E. Roberts. Activator-mediated disruption of sequence-specific dna contacts by the general transcription factor tfiib. *Genes & Development*, 15(1):2945–2949, 2001.

P. Flicek, B. L. Aken, K. Beal, B. Ballester, M. Caccamo, Y. Chen, and L. Clarke. Ensembl. *Nucleic Acids Res.*, 36:707–714, 2008.

Gene Ontology Consortium. The gene ontology (go) project in 2006. *Nucleir Acids Res.*, 34(1):322–6, Jan 2006.

GeneSpring weboldal. www.agilent.com/chem/genespring.

Tatiana I. Gerasimova and Victor G. Corces. Chromatin insulators and boundaries: effects on transcription and nuclear organization. *Annual Reviews of Genetics*, 35:193–208, 2001.

Naum I. Gershenzon and IIya P. Ioshikhes. Synergy of human pol ii core promoter elements revealed by statistical sequence analysis. *Bioinformatics*, 21(8):1295–1300, 2005.

S. Gustincich, A. Sandelin, C. Plessy, S. Katayama, R. Simone, D. Lazarevic, Y. Hayashizaki, and P. Carninci. The complexity of the mammalian transcriptome. *J Physiol*, 575(9):321–32, 2006.

J. Jurka, P. Klonowski, V. Dagman, and P. Pelton. Censor - a program for identification and elimination of repetitive elements from dna sequences. *Comput Chem*, 1(20):119–121, 1996.

Tamar Juven-Gershon and James T. Kadonaga. Regulation of gene expression via the core promoter and the basal transcriptional machinery. *Developmental Biology*, 339(2): 225–229, 2010.

H. Kawaji, T. Kasukawa, S. Fukuda, S. Katayama, C. Kai, J. Kawai, P. Carninci, and Y. Hayashizaki. Cage basic/analysis databases: the cage resource for comprehensive promoter analysis. *Nucleic Acids Research*, 34(1):632–6, 2006.

Isaac S. Kohane, Alvin T. Kho, and Atul J. Buttle. *Microarrays for an integrative genomics*. The MIT Press, 2003.

Roger D. Kornberg. The molecular basis of eukaryotic transcription. *PNAS*, 104(32): 12955–12961, 2007.

T. Kulikova, R. Akhtar, P. Aldebert, N. Althorpe, M. Andersson, A. Baldwin, K. Bates, S. Bhattacharyya, L. Bower, P. Browne, M. Castro, G. Cochrane, K. Duggan, R. Eberhardt, N. Faruque, G. Hoad, C. Kanz, R. Leinonen, Q. Lin, V. Lombard, R. Lopez, D. Lorenc, H. McWilliam, G. Mukherjee, F. Nardone, M. P. Pastor, S. Plaister, S. Sobhany, P. Stoehr, R. Vaughan, D. Wu, W. Zhu, and R. Apweiler. Embl nucleotide sequence database in 2006. *Nucleic Acids Research*, 35(1):16–20, 2007.

C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wootton. Detecting subtle sequence signals: A gibbs sampling strategy for multiple alignment. *Science*, 262(1):208–214, 1993.

Thong Ihn Lee and Richard A. Young. Transcription of eukaryotic protein-coding genes. *Annual Reviews of Genetics*, 34:77–137, 2000.

S. J. Liu, A. F. Neuwald, and C. E. Lawrence. Bayesian models for multiple local sequence alignment and gibbs sampling strategies. *Journal of American Statistic Association*, 90 (1):1156–1170, 1995.

V. Matys, O. V. Kel-Margoulis, E. Fricke, I. Liebich, S. Land, A. Barre-Dirrie, I. Reuter, D. Chekmenev, M. Krull, K. Hornischer, N. Voss, P. Stegmaier, B. Lewicki-Potapov, H. Saxel, A. E. Kel, and E. Wingender. Transfac and its module transcompel: transcriptional gene regulation in eukaryotes. *Nucleic Acids Research*, 34(1):108–10, 2006.

Frank F. Millenaar, John Okyere, Sean T. May, Martijn van Zanten, Laurentius A. C. J. Voesenek, and Anton J. M. Peeters. How to decide? different methods of calculating gene expression from short oligonucleotide array data will give different results. *BMC Bioinformatics*, 7(137):1–16, 2006.

B. Morgenstern. Dialign2: Improvement of the segment-to-segment approach to multiple sequence alignment. *Bioinformatics*, 15(3):211–8, 1999.

A. M. Moses, D. Y. Chiang, M. Kellis, E. S. Lander, and M. B. Eisen. Position specific variation in the rate of evolution in transcription factor binding sites. *BMC Evol Biol*, 28(3):19, 2003.

C. J. S. Parker, L. Hansen, H. O. Abaan, T. D. Tullius, and E. H. Margulies. Local dna topology correlates with functional noncoding regions of the human genome. *Science*, 324(4):389–392, 2009.

H. Parkinson, M. Kapushesky, M. Shojatalab, N. Abeygunawardena, R. Coulson, A. Farne, E. Holloway, N. Kolesnykov, P. Lilja, M. Lukk, R. Mani, T. Rayner, A. Sharma, E. William, U. Sarkans, and A. Brazma. Arrayexpress - a public database of microarray experiments and gene expression profiles. *Nucleic Acid Research*, 35 (1):747–750, 2007.

Jesse R. Raab and Rohinton T. Kamakaka. Insulators and promoters: closer than we think. *Nature Review of Genetics*, 11(6):439–446, 2010.

M. Reimers and V. J. Carey. Bioconductor: an open source framework for bioinformatics and computational biology. *Methods Enzymology*, 411(1):119–134, 2006.

P. Rice, I. Longden, and A. Bleasby. Emboss: the european molecular biology open software suite. *Trends Genet.*, 16(6):276–7, Jun 2000.

I. Rivals, L. Personnaz, L. Taing, and M. C. Potier. Enrichment or depletion of a go category within a class of genes: which test? *Bioinformatics*, 23(4):401–407, Febr 2007.

G. Robertson, M. Bilenky, K. Lin, A. He, W. Yuen, M. Dagpinar, R. Varhol, K. Teague, O. L. Griffith, X. Zhang, Y. Pan, M. Hassel, M. C. Sleumer, W. Pan, E. D. Pleasance, M. Chuang, H. Hao, Y. Y. Li, N. Robertson, C. Fjell, B. Li, S. B. Montgomery, T. Astakhova, J. Zhou, J. Sander, A. S. Siddiqui, and S. J. Jones. cisred: a database system for genome-scale computational discovery of regulatory elements. *Nucleic Acid Research*, 34(1):68–73, 2006.

F. P. Roth, J. D. Hughes, P. W. Estep, and G. M. Church. Finding dna regulatory motifs within unaligned noncoding sequences clustered by whole-genome mrna quantitaion. *National Biotechnology*, 16(1):939–945, 1998.

A. Sandelin, W. Alkema, and B. Lenhard P. Engstrom, W. W. Wasserman. Jaspar: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acid Research*, 32(1):91–4, 2004.

T. D. Schneider. Consensus sequence zen. *Appl Bioinformatics*, 3(1):111–119, Jan 2002.

T. D. Schneider and R. M. Stephens. Sequence logos: a new way to display consensus sequences. *Nucleic Acid Research*, 18(20):6097–100, 1990.

Xiang Shen, Jeong-Seok Park, Ye Qui, Joel Sugar, and Beatrice Y J T Yue. Effects of sp1 overexpression on cultured human corneal stromal cells.

T. Shiraki, S. Kondo, S. Katayama, K. Waki, T. Kasukawa, H. Kawaji, R. Kodzius, A. Watahiki, M. Nakamura, T. Arakawa, S. Fukuda, D. Sasaki, A. Podhajska, M. Harbers, J. Kawai, P. Carninci, and Y. Hayashizaki. Cap analysis gene expression for high-throughput analysis of transcriptional starting point and identification of promoter usage. *Procl Natl Acad Sci*, 100(26):15776–81, 2003.

V. L. Singer, C. R. Wobbe, and K. Struhl. A wide variety of dna sequences can functionally replace a yeast tata element for transcriptional activation. *Genes & Development*, 4(1): 636–645, 1990.

Gary D. Stormo. Dna binding sites: representation and discovery. *Bioinformatics*, 16(1): 16–23, 2000.

H. Sugawara, O. Ogasawara, K. Okubo, T. Gojobori, and Y. Tateno. Ddbj with new system and face. *Nucleic Acids Research*, 36(1):22–24, 2008.

Wing-Kin Sung. *Algorithms in bioinformatics*. CRC Press, Taylor and Francis Group, 2010.

Istvan Szatmari, Daniel Torocsik, Maura Agostini, Tibor Nagy, Mark Gurnell, Endre Barta, Krishna Chatterjee, and Laszlo Nagy. Pparγ regulates the function of human dendritic cells primarily by altering lipid metabolism. *Blood*, 110(9):3271–80, 2007.

Gert Thijs, Magali Lescot, Kathleen Marchal, Stephane Rombauts, Bart De Moor, Pierre Rouzé, and Yves Moreau. A higher-order background model improves the detection of promoter regulatory elements by gibbs sampling. *Bioinformatics*, 17(12):1113–1122, 2001.

UCSC weboldal. http://genome.ucsc.edu/encode/.

S. Vardhanabhuti, J. Wang, and S. Hannenhalli. Position and distance specificity are important determinants of cis-regulatory motifs in addition to evolutionary conservation. *Nucleic Acids Res*, 35(10):3203–13, 2007.

H. Wakaguri, R. Yamashita, Y. Suzuki, S. Sugano, and K. Nakai. Dbtss: database of transcription start sites, progress report 2008. *Nucleic Acids Res.*, 36:97–101, Jan 2008.

Kyoung-Jae Won, Albin Sandelin, Troels Torben Marstrand, and Anders Krogh. Modelling promoter grammars with evolving hidden markov models. *Bioinformatics*, 24(15): 1669–1675, 2008.

G. A. Wray, M. W. Hahn, E. Abouheif, J. P. Balhoff, M. Pizer, M. V. Rockman, and L. A. Romano. The evolution of transcriptional regulation in eukaryots. *Mol Biol Evol*, 20(9):1377–419, Sep 2003.

8. fejezet

Publikációk

8.1. A disszertáció alapjául szolgáló tudományos közlemények

I. Szatmari, D. Torocsik, M. Agostini, *T. Nagy*, M. Gurnell, E. Barta, K. Chatterjee, L. Nagy. PPARgamma regulates the function of human dendritic cells primarily by altering lipid metabolism. Blood, 110(9):3271-80, 2007. jul. Cikk (IF:10.55)

Sebestyén E, *Nagy T*, Suhai S, Barta E. DoOPSearch: a web-based tool for finding and analysing common conserverd motifs in the promoter regions of different chordate and plant genes. BMC Bioinformatics, 10(6):S6, 2009. Cikk (IF:3.43)

Összes impakt faktor: 13.98

8.2. A disszertáció témakörében készült konferencia előadások és poszterek

Endre Sebestyén, *Tibor Nagy*, Tamás Pálfy, Gábor Tóth, Endre Barta: Identifying common conserved promoter motifs between genes, using the taxonomic group-based motif collections of the DoOP database. 15th Annual International Conference on Intelligent Systems for Molecular Biology and 6th European Conference on Computational Biology. Bécs, 2007 július 21-25. Poszter

Sebestyén Endre, *Nagy Tibor*, Pálfy Tamás, Szenes Áron, Molnár János, Tóth Gábor és Barta Endre: A transzkripció szabályozásának vizsgálata bioinformatikai módszerekkel: A DoOP adatbázis és a DoOPSearch keresőoldal. 2006. Magyar Biokémiai Egyesület 2006. évi vándorgyűlése. Poszter

Pálfy Tamás, Sebestyén Endre, *Nagy Tibor*, Tóth Gábor és Barta Endre: A transzkripciós kezdőpont körüli szekvenciák nukleotid eloszlás és motivum mintázat vizsgálat különböző emlős és rizs promóterekben. 2006. évi vándorgyűlése.

Pálfy Tamás, Sebestyén Endre, *Nagy Tibor*, Tóth Gábor és Barta Endre: A transzkripciós kezdopont körüli szekvenciák nukleotid eloszlás és motivum mintázat vizsgálat különbözo emlos és rizs promóterekben. 2006. évi Magyar Bioinformatikai Társaság alakuló ülése. 2006 június 12-13. Előadás

Sebestyén Endre, *Nagy Tibor*, Pálfy Tamás, Szenes Aron, Molnár János, Tóth Gábor és Barta Endre: A transzkripció szabályozásának vizsgálata bioinformatikai módszerekkel - DoOP adatbázis és a DoOPSearch keresőoldal. 2006. évi Magyar Bioinformatikai Társaság alakuló ülése. 2006 június 12-13. Előadás

9. fejezet

Köszönetnyilvánítás

Munkám során sokat köszönhetek Barta Endrének, aki egyengette bioinformatikai karrieremet. Egy kutatócsoport tagjának lenni azt jelenti, hogy eredményeink sokban függenek kollégáink munkájától is. Ezért köszönetet kell mondanom Sebestyén Endrének, Pálfy Tamásnak és Tóth Gábornak. A kísérleti hátteret a DEOEC, ÁOK, Biokémiai és Molekuláris Biológiai Intézete végezte, akik közül közvetlenül Nagy László, Bálint B. László és Szatmári Istvánt említeném meg.

Köszönöm továbbá Putnoky Péternek, aki támogatott annak ellenére is, hogy kikerültem a PTE védőszárnyai alól.

Végezetül szeretném megköszönni édesanyámnak és feleségemnek, akik mellettem álltak mindvégig.