UNIVERSITAT JAUME I

Grau en Matemàtica Computacional

Estada en Pràctiques i Projecte Final de Grau

Introducció a la modelització matemàtica dels arbres filogenètics

Autora:
Marta Iborra Clari

Supervisor:
Francesc Alted Abad
Tutor acadèmic:
Pablo Gregori Huerta

Data de lectura: juliol de 2021
Curs acadèmic 2020/2021

Resum

En aquest document es detalla tant l'estada en pràctiques amb Francesc Alted, com el treball final de grau.

En la primera part es detalla l'estada en pràctiques. Es parla de les tasques realitzades durant l'estada en pràctiques, els coneixements assolits i les ferraments emprades. La tasca principal fou implementar l'emmagatzematge en disc no seqüencial.

En la segona part d'aquest document, es fa una introducció a un tema actual tant en la biologia com en les matemàtiques: la filogenètica. S'intenta presentar la relació que hi ha entre ambdues mitjançant la descripció de models i altres ferramentes usats en el camp de la filogenètica.

Paraules clau

Arbres filogenètics. Estadística algebraica. Blosc.

Keywords

Phylogenetics trees. Algebraic statistics. Blosc.

Índex

Capítol 1

Introducció

1.1 Context i motivació del projecte

En el grau en Matemàtica Computacional, el projecte final de grau i l'estada en pràctiques corresponen a 18 crèdits que es cursen l'últim curs.

L'estada en pràctiques es va realitzar amb Francesc Alted per tal de completar el compressor *Blosc*. Un dels aspectes que em van agradar és el bon ambient que hi havia: familiar, però respectuós i professional. Un altre aspecte que em va interessar fou el de formar part d'un projecte de software lliure. Perquè aquests projectes solen estar molt complets, ja que els han construït de manera progressiva gent amb diferents maneres de veure el món.

Respecte al tema del TFG, va sorgir de l'interés partit que tenia per dues branques de les matemàtiques: l'estadística i l'àlgebra. Aquest tema va conduir a la filogenètica, i d'aquí és d'on naix aquest TFG.

Tant la compressió de dades (amb el gran resó que té el *Big Data*), com la filogenètica i les matemàtiques, són temes actuals i de gran interés per a la societat en què vivim.

Capítol 2

Estada en pràctiques

2.1 Introducció

Francesc Alted és un assessor i desenvolupador de software amb més de 20 anys d'experiència especialitzat en els llenguatges de programació C i Python. Situat al Grau de Castelló, assessora empreses en la gestió de dades alhora que treballa amb altres projectes de programació.

És el creador, entre d'altres, de *PyTables* (paquet que gestiona conjunts de dades dissenyats per a fer front a grans quantitats de dades) i *Blosc*. *Blosc* és un compressor de dades d'alt rendiment optimitzat per a emmagatzematge binari (i.e. nombres en coma flotant, enters i booleans) que fa de fonaments per a altres projectes de Francesc Alted, com ara *bcolz* o més recentment, *Caterva* (un contenidor de dades multidimensional que es pot comprimir).

2.2 Objectius del projecte formatiu

Durant aquesta estada en pràctiques col·laboraré en el desenvolupament del compressor de dades *Blosc*. A diferència de la majoria de compressors, *Blosc* té com a objectiu no només reduir el tamany de les dades, sinó també guanyar velocitat. Aquest últim objectiu l'aconsegueix dividint els conjunts de dades anomenats *chunks* en conjunts més petits anomenats blocs. El tamany d'aquests blocs ha de ser menor que el tamany de la memòria cau per tal d'evitar comunicar-se amb la memòria central (l'accés a la qual és més lent). A banda, es permet l'execució mutithreading automàtica, característica que s'aprofita cada dia més tenint en compte les característiques dels nous ordinadors.

A més a més, *Blosc* permet escollir entre diferents compressors i filtres (programa que generalment, millora la ràtio de compressió), per la qual cosa se l'hauria d'anomenar més aviat un metacompressor.

Al principi de l'estada, *Blosc* permetia emmagatzematge en memòria (tant de manera seqüencial com distribuïda) i emmagatzematge en disc només de manera seqüencial. La diferència entre emmagatzematge en disc i en memòria és que en el primer, les dades persisteixen en l'ordinador encara que s'apague i després es torne a encendre, mentre que en el segon les dades es perden quan l'ordinador s'apaga.

La meua tasca consistirà a completar les prestacions de *Blosc* per tal que es puguen emmagatzemar dades de manera distribuïda en disc.

Per tal de realitzar aquesta tasca, empraré el llenguatge de programació C, tot millorant els meus coneixements d'aquest. Hauré d'aprendre no només com funciona el compressor *Blosc*, sinó que també hauré d'aprendre a utilitzar-lo per tal de comprovar el correcte funcionament de les meues implementacions.

2.3 Explicació detallada del projecte realitzat a l'empresa

2.3.1 Metodologia i definició de tasques

Definició de l'estructura

En *Blosc* les dades es guarden com un conjunt de chunks anomenat schunk (superchunk). Per tal que aquests chunks estiguen emmagatzemats en disc de manera distribuïda, cada chunk serà en realitat un fitxer binari independent. Tots els chunks d'un mateix schunk estaran emmagatzemats al mateix directori proporcionat per l'usuari.

A més a més, per a poder descomprimir els chunks, necessitarem un header amb la informació necessària per a açò i un fitxer a banda que faça de llista dels chunks que té el schunk. A més d'això, com que l'usuari ha de poder afegir metainformació, també es requerirà d'un trailer que contindrà aquesta metainformació.

Arribats a aquest punt cal descriure l'estructura d'un Contiguous Frame (cframe d'ara endavant) que és la que s'empra per a l'emmagatzematge seqüencial (tant en disc com en memòria). En el cas d'emmagatzematge en disc, el cframe consta d'un únic fitxer binari format per un header, una secció de chunks i un trailer (com s'observa en la Figura 2.1). El header conté informació necessària per a comprimir i descomprimir els chunks. La secció dels chunks està

composta per tots els chunks de dades més un chunk addicional al final d'aquesta secció anomenat index chunk. Aquest chunk conté l'offset de cada chunk (la posició on comença cada chunk dins del fitxer cframe) ordenats segons els seus índexos. Finalment, el trailer conté un chunk amb metainformació que pot afegir l'usuari.

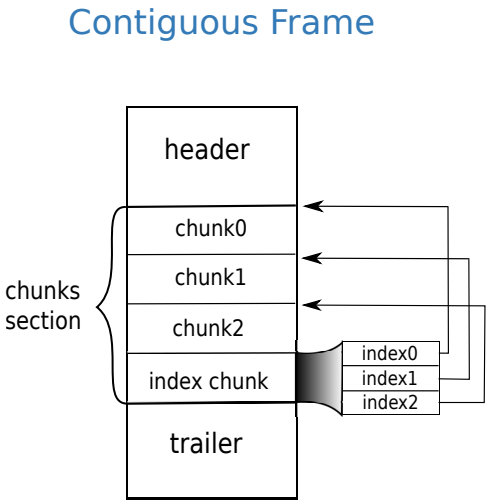


Figura 2.1: Estructura d'un cframe.

Així doncs, si comparem les necessitats de l'estructura per a l'emmagatzematge en disc distribuït amb l'estructura d'un cframe, ens adonem que tot el conjunt d'informació addicional necessària (header, llista dels chunks i trailer) es pot emmagatzemar en realitat com un fitxer cframe on la secció de chunks està composta només per l'index chunk (la llista). Després de plantejar diferents formats, es va decidir que aquesta era la millor perquè s'aprofitava una estructura ja creada que, al seu torn, evitava duplicar codi amb la possibilitat de reproduir bugs i agilitzava tot el treball d'implementació necessari.

A partir d'aquesta decisió, es va batejar l'estructura per a emmagatzematge en disc distribuït de *Blosc* com a Sparse Frame (sframe d'ara endavant), que resumint el que ja s'ha discutit, constaria d'un directori on s'emmagatzemarien tots els chunks de dades com a fitxers binaris individuals i un fitxer anomenat chunks.b2frame que seria el fitxer cframe però sense chunks. Per tal de no confondre aquest fitxer chunks.b2frame amb un cframe, l'estructura comuna a ambdós s'anomenarà frame.

D'altra banda, a fi de fer la longitud del nom de cada fitxer el més constant possible, el nom del fitxer d'un chunk vindria determinat pel seu índex (un número identificador que més endavant es veurà que no té per què coincidir amb la posició que ocupa), al que se li afegirien zeros per l'esquerra fins que aquest nombre tinguera una longitud de 8 caràcters. Seguidament, a aquest número se li afegeix l'extensió ".chunk" per a denotar que el fitxer és un chunk.

A la Figura 2.2 es compara l'estructura d'un Contiguous Frame amb la d'un Sparse Frame de només 3 chunks.

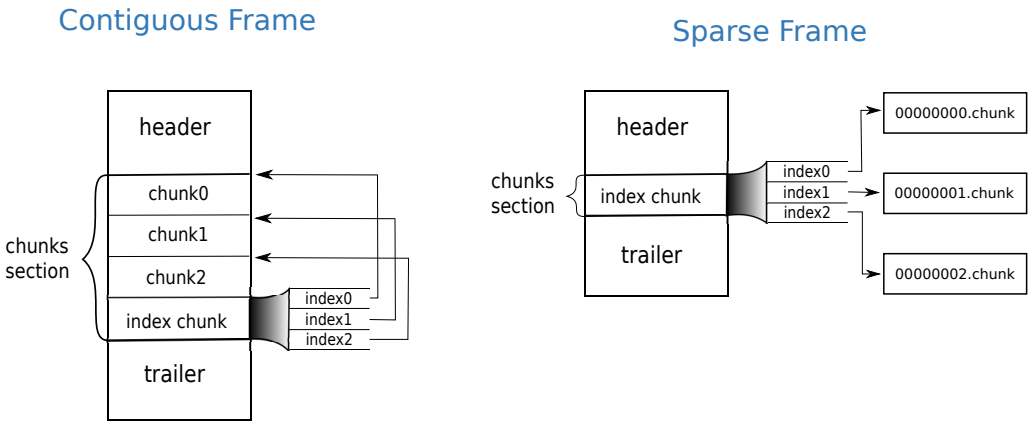


Figura 2.2: Comparació entre el fitxer frame per a emmagatzematge seqüencial (cframe) i distribuït (sframe).

Funcions bàsiques d'un schunk

Un cop definida l'estructura d'un sframe, el següent pas fou implementar la funció `schunk_new` que creava un schunk buit segons els paràmetres especificats per la variable `blosc2_storage` composta pels 4 camps següents:

- `bool contiguous`: variable booleana indicant si l'emmatgatzematge és seqüencial o no.
- `char* urlpath`: en el cas de l'emmagatzematge en disc (per a emmagatzematge en memòria és `NULL`) indica la ruta on emmagatzemar les dades.
- `blosc2_cparams* cparams`: paràmetres per a la compressió.
- `blosc2_dparams* dparams`: paràmetres per a la descompressió.

És interessant observar que amb les dues primeres variables es pot especificar quin dels 4 possibles tipus d'emmagatzematge es desitja. En particular, si contiguous és false i la urlpath és diferent de `NULL`, haurem de crear un sframe, el nom del directori del qual serà la urlpath. Mentre que si contiguous és true i la urlpath és diferent de `NULL` el que indica aquesta variable no és un directori, sinó el fitxer del cframe.

Com que interessava que tant un cframe com un sframe usaren el mateix codi, s'havia d'afegir algun paràmetre a l'estructura del cframe per poder diferenciar un cas de l'altre ja que

un chunk no s'emmagatzemava de la mateixa manera en un cframe que en un sframe. És per això que es va afegir la variable booleana sframe al frame. Si aquesta era true estàvem en un sframe i el fitxer frame s'obtenia afegint a la urlpath la cadena "chunks.b2frame". Si era false, la ruta completa del fitxer cframe era exactament la cadena urlpath.

Després de crear un schunk buit, el que interessa és poder afegir dades. De manera que el següent pas fou impletar la funció `blosc2_schunk_append_buffer`. Aquesta funció rep un buffer, el comprimeix per tal de crear un chunk arran d'aquest i afegeix el chunk al superchunk (crea el fitxer amb el contingut del chunk i actualitza el fitxer frame).

En tercer lloc, calia poder recuperar aquestos chunks mitjançant la funció `blosc2_schunk_decompress_chunk`. Aquesta funció rep l'índex del chunk que es vol llegir, llegeix del fitxer chunks.b2frame el contingut del index chunk per a obtindre el nom del fitxer chunk i retorna el seu contingut ja descomprimit. Després d'implementar les funcions necessàries per a fer un roundtrip, vaig implementar un exemple d'ús (sframe_simple.c) que realitzava un roundtrip alhora que testejava de manera superficial el funcionament de les funcions que havia implementat.

Després de les funcions bàsiques per a treballar amb un sframe, es va implementar la funció `blosc2_schunk_open`. Aquesta funció rep la ruta d'un frame per a obrir-lo i crear un schunk arran del frame. En el cas del sframe rep el nom del directori. Per tal de diferenciar si es tracta d'un cframe o d'un sframe es comprova si la ruta que s'ha passat com a paràmetre és un directori o no.

Implementació dels lazychunks

Per tal d'explicar què són i com funcionen els lazychunks, primer cal explicar l'estructura d'un chunk. L'estructura d'un chunk està formada per un petit header (amb informació sobre el chunk) i una secció de blocks (particions del contingut del chunk). A banda, pot contindre un trailer de manera opcional on s'indica el número de chunk, el seu offset (en el cas d'un cframe) i el tamany en bytes de cada block.

Per al cframe, Francesc Alted va implementar els lazychunks per a quan només es vol llegir un tros d'un chunk. Un lazychunk és un chunk on només es carrega la informació necessària. És molt eficient sobretot quan només es desitja llegir un tros d'un chunk, perquè primer es determinen quins blocks de cada chunk es necessiten i després s'obtenen només eixos blocks.

La meva tasca fou completar aquesta implementació per tal que els lazychunks es pogueren emprar també en un Sparse Frame. Un cop més, aquesta tasca es va simplificar bastant, perquè només s'havien de tindre en compte les diferències entre un sframe i un cframe. És a dir, en sframe s'havia d'escriure al trailer del chunk el número de nchunk al camp de l'offset i evidentment, el chunk s'havia de llegir des del directori del sframe i no des de fitxer (com es fa

a un cframe).

Tests

Per tal de comprovar el bon funcionament de la implementació, es van implementar diferents tests.

A banda, com que no es pot crear un directori que ja existeix però ens interessa poder executar els mateixos tests que creen superchunks amb uns paràmetres predeterminats (i per tant amb el mateix nom de directori en el cas d'un sframe) de manera reiterada, vaig haver d'implementar una funció que esborrara un directori sencer (`blosc2_remove_dir`). Aquesta funció es cridava al final de cada test on s'haguera construït un sframe i fou una de les dificultats més grans que vaig tindre (després d'entendre el funcionament de *Blosc*) durant l'estada. Com que aquesta funció havia de ser portable, vaig tindre alguns problemes a l'hora de trobar funcions en C que em permeteren esborrar el contingut d'un directori en Windows. De fet, vaig haver d'implementar dues funcions: una per a Windows i l'altra per a la resta de sistemes operatius suportats per *Blosc*.

Noves funcionalitats

Com que *Blosc* era encara un projecte en desenvolupament, durant la meva estada s'afegiren tres funcions més al frame: `frame_update_chunk`, `frame_insert_chunk` i `frame_reorder_offsets`.

La funció `frame_update_chunk` actualitzava el contingut d'un chunk. En el cas d'un cframe, si el tamany nou del chunk era major que el del chunk vell, funcionava com el joc del Jenga, on la peça es col·locava al final de la torre (vegeu Figura 2.3). En aquest cas, es deixava l'antic espai que ocupava el chunk buit i es col·locava al final de la secció del chunks (abans del index chunk). Si per contra, el tamany del nou chunk era més menut o igual que l'antic, es col·locava en la mateixa posició deixant només l'espai sobrant buit (tal com mostra la figura de sota).

Tanmateix, en el cas d'un sframe no es deixava cap espai buit perquè el fitxer es sobreescrivia amb el nou contingut del chunk. Podríem dir que un Sparse Frame funciona com un prestatge sense límit d'altura en el qual caben tants llibres (chunks) com els fitxers que permeta el sistema. Així, si volem canviar un llibre (fitxer chunk) per un altre més gran no es deixa cap espai buit, tal i com s'observa en la Figura 2.4), on es substitueix el llibre groc.

En segon lloc, la funció `frame_insert_chunk` inseria un chunk en una posició especificada. Aquesta funció es va implementar més pensant en Caterva.

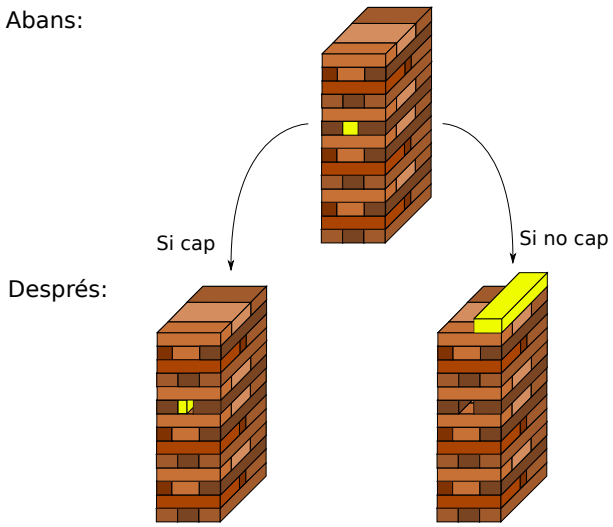


Figura 2.3: Representació de la secció de chunks d'un cframe abans i després de realitzar un `update`

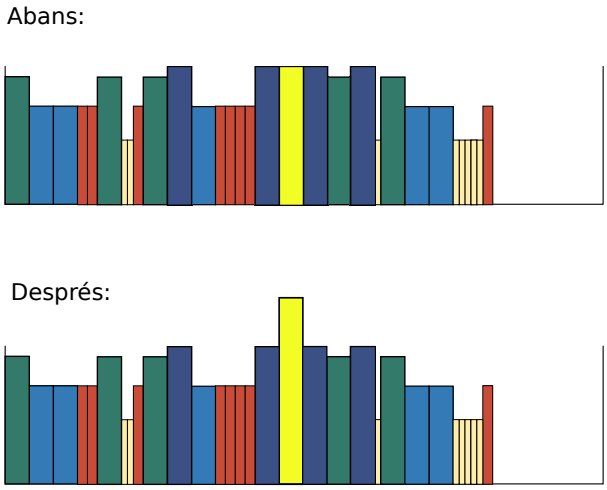


Figura 2.4: Representació de la secció de chunks d'un sframe abans i després de realitzar un `update`

I és ací on cal remarcar que el número que determina el nom del fitxer chunk no té per què coincidir amb la posició en què està el chunk. Si per exemple tenim un sframe amb 4 chunks, de la següent manera:

```
directoriSframe/
|
|- 00000000.chunk
|
|- 00000001.chunk
|
|- 00000002.chunk
|
|- 00000003.chunk
|
|- chunks.b2frame
```

Com que aquestos chunks s'han afegit de manera ordenada, el contingut de l'index chunk estarà també ordenat i serà: [0, 1, 2, 3] (recordem que aquest chunk conté el número de cada chunk en la posició que li pertoca. Si ara volem inserir un chunk en la posició 2, però, a aquest nou chunk se'l anomena 00000004.chunk i no 00000002.chunk i el nou contingut de l'index chunk seria [0, 1, 4, 2, 3]. És a dir, els chunks que ocupaven una posició major o igual a la que es vol inserir, es desplacen una posició. Així, els fitxers chunks que ja hi havien no s'han modificat ni de nom ni de contingut i el contingut del sframe després de realitzar la inserció és:

```
 directoriSframe/
|
|- 00000000.chunk
|
|- 00000001.chunk
|
|- 00000002.chunk
|
|- 00000003.chunk
|
|- 00000004.chunk
|
|- chunks.b2frame
```

Si decidírem haver canviat els noms dels fitxer per tal que coincidiren amb les posicions

que ocupen en l'index chunk la funció seria molt més ineficient, i tampoc ens caldria per a res l'index chunk.

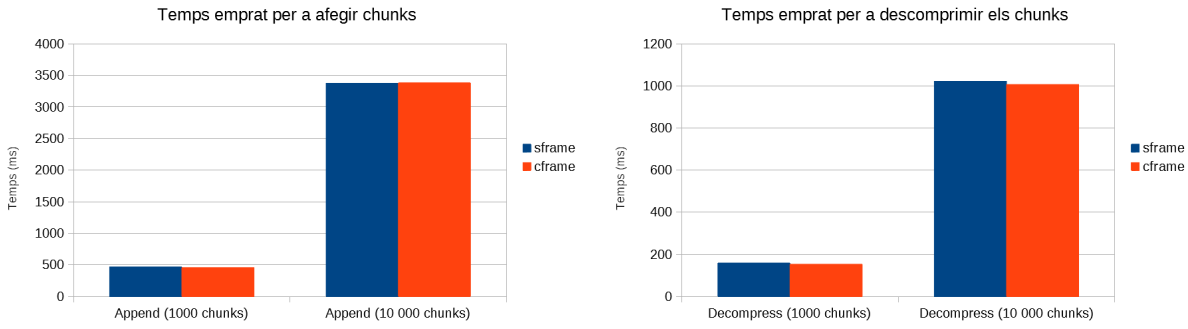
En tercer lloc, la funció `frame_reorder_offsets` reordena la posició dels chunks segons una nova llista passada com a paràmetre. De la mateixa manera que en la funció `frame_insert_chunk`, aquesta funció només modifica el contingut de l'index chunk i no el nom dels chunks. Si per exemple, tenim l'anterior sframe amb 4 chunks [0, 1, 2, 3] i volem reordenar-los per tal que el primer siga el segon, el segon el tercer, el tercer el quart i el quart el primer, la nova llista que li passarem com a paràmetre a aquesta funció serà [3, 0, 1, 2] i aquesta llista serà en realitat el nou contingut de l'index chunk.

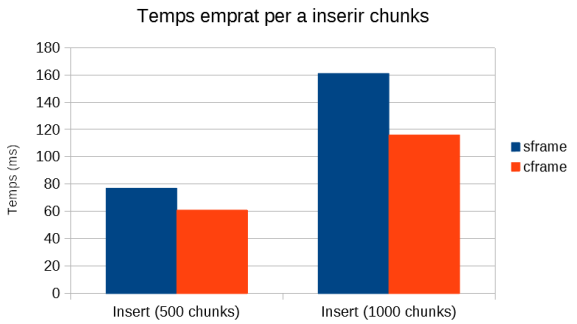
Cal remarcar, però, que si el contingut inicial de l'index chunk fora [1, 2, 3, 0], el nou ordre dels chunks seria en realitat [0, 1, 2, 3].

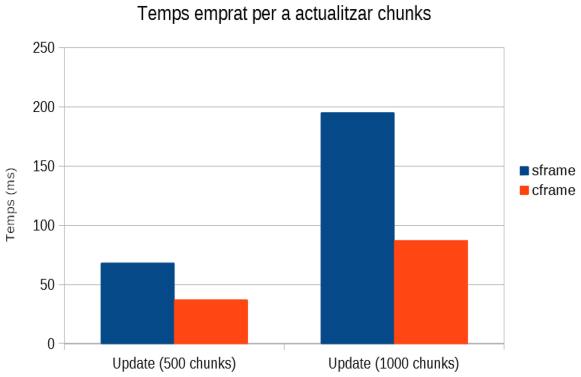
Tests de referència

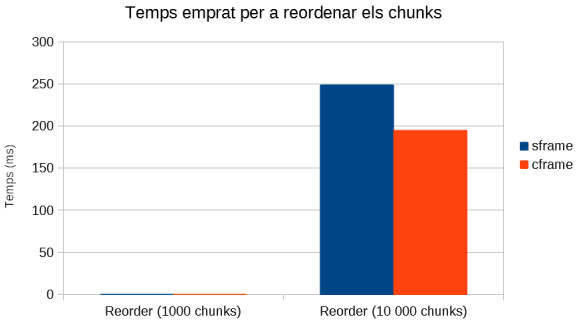
Com que la part de la implementació del sframe es va fer més ràpid del previst gràcies a haver aprofitat ja l'estructura del frame, es va implementar un test de referència per tal de comparar el rendiment d'un sframe amb el d'un cframe.

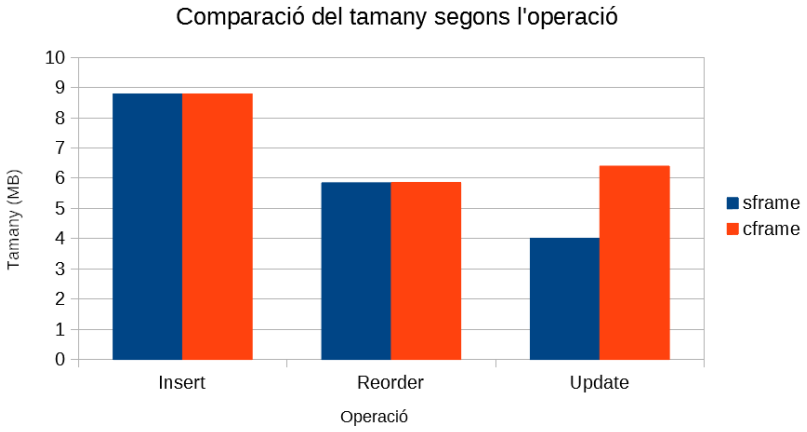
En aquest test, es creaven un sframe i un cframe idèntics i es mesurava el temps que tardaven a realitzar cada operació, i s'obteniren els següents resultats:











D'aquests gràfics s'observa que, tot i que un Sparse Frame és més lent, a l'hora de fer els updates, un Sparse Frame només ocupa el que necessita mentre que un Contiguous Frame pot estar ocupant més espai que el que ocupen en total les seves dades (recordem que en aquests tests el contingut del schunk és el mateix per a ambdós casos). Segons aquest últim gràfic, el sframe contruït en el test de referència ocupa 4MB mentre que el cframe ocupa uns 6,5 MB

malgrat contindre exactament els mateixos chunks.

Documentació

Per acabar, es va redactar un fitxer en format markdown on s'explicava el format d'un sframe amb la seva estructura (`README_SFRAME_FORMAT.rst`). A més a més, també es va redactar una entrada al blog de Blosc, anunciant aquesta nova funcionalitat de Blosc que Francesc Alted va millorar (`https://www.blosc.org/posts/introducing-sparse-frames/`).

2.3.2 Software emprat

Git

És un software de control de versions que permet coordinar els canvis que es fan de manera contínua a un projecte. Si es desitja modificar un projecte, Git permet crear una branca (similar a una còpia del projecte) on hom pot modificar al seu ritme sense modificar el projecte original. Si el projecte original es modifica, Git permet afegir aquestos canvis a la branca per tal de poder seguir treballant amb l'última versió del projecte.

Un cop ja s'han realitzat els canvis desitjats en la branca, hom pot suggerir d'afegir-los al projecte mitjançant una pull request. Llavors, els col·laboradors poden veure els canvis i comentar o suggerir canvis.

Així, moltes persones poden estar treballant en un mateix projecte sense col·lisionar.

CLion

És un IDE (entorn integrat de desenvolupament) de JetBrains que permet executar de manera remota. És a dir, es pot treballar i compilar el projecte tant en l'ordinador local com de manera remota a una màquina (en el meu cas Ubuntu). Açò facilita la tasca de programar codi portable per a diferents sistemes operatius.

2.3.3 Grau de consecució dels objectius proposats

La implementació d'emmagatzematge distribuït en disc s'ha pogut realitzar en la seva totalitat i està disponible per a la seva descàrrega en el repositori de github

`https://github.com/Blosc/c-blosc2` .

Aquesta implementació obre les portes a altres aplicacions. Una de les possibilitats seria aprofitar aquesta implementació per a emmagatzemar dades remotament tot modificant la implementació per a realitzar operacions en la xarxa, de manera que només amb la metainformació (el fitxer frame) es podria accedir als chunks d'un sparse frame que hi ha a una altra màquina. Açò vol dir que si tenim un sparse frame amb 10TB de dades, amb el seu fitxer frame d'uns 10 KB, només amb aquest fitxer de 10KB hom podria accedir al sparse frame sencer de 10TB.

Una altra possible aplicació donaria lloc a bases de dades clau/valor remotes. Aquestes bases de dades funcionen de manera semblant a un conjunt de portes amb les seues claus. Cada clau obre una única porta i el valor és el contingut que amaga cada porta. En aquest cas cada porta seria un chunk diferent i la clau estaria formada per la url des d'on poder accedir el sframe junt amb la posició del chunk que es vol llegir.

Dins dels resultats del test de referència, malgrat la rapidesa superior d'un cframe, cal remarcar la diferència de tamany que es pot arribar a obtindre entre un sframe i un cframe quan s'actualitzen els chunks vells per d'altres que ocupen més tamany.

2.3.4 Conclusions

Durant aquesta estada considere que he millorat molt les meues competències. D'una banda, he augmentat de manera considerable els meus coneixements de programació en C i he aprés a emprar la ferramenta Git. D'atra banda considere que la llibertat que m'ha donat Francesc Alted no només m'ha permés aprendre al meu ritme sinó que també m'ha pujat l'autoestima. A més a més, participar en un projecte com aquest ha fet que em veja més capacitada per a realitzar altres projectes o inserir-me al món laboral.

Capítol 3

Memòria TFG

3.1 Motivació i objectius

Els grans reptes que presenta la biologia en el present estan delimitats per l'estudi del genoma dels organismes. L'estructura i funció dels gens, les relacions entre diferents organismes, etc, són preguntes que es poden respondre amb una barreja d'estadística, probabilitat, i geometria algebraica.

En primer lloc, es realitza una introducció a la genètica per tal que el lector puga seguir els raonaments exposats.

En segon lloc, es comença amb un model estadístic per tal d'inferir informació sobre el codi genètic.

En tercer lloc, s'exposa un model més realista usat arreu en la biologia: el model ocult de Màrkov. Tot relacionat-ho amb l'aritmètica tropical i la programació dinàmica.

En quart lloc, s'exposa el problema de l'alineament de seqüències junt amb el model ocult de Màrkov parell (una extensió del model ocult de Màrkov que com el seu nom indica, s'aplica a parelles). També s'exposen de manera ràpida els polítops per tal de poder realitzar inferències paramètiques en aquest model.

En cinqué lloc, s'exposen els models evolutius. Aquests ja permeten comparar més de dues espècies alhora, i construir el seu l'arbre evolutiu més probable.

Finalment, s'aprofita el treball de M. Drton, N. Eriksson i G. Leung [14] basat en els principis

exposats, per tal d'estimar la probabilitat que un fragment del genoma es conserve per pura casualitat.

3.2 Coneixements previs: el genoma

Tot organisme viu té un genoma, format per àcids desoxiribonucleic (ADN) disposats en una doble hèlix [23], que codifica els ingredients fonamentals de la vida. Els organismes es divideixen en dues classes principals: eucariotes (organismes les cèl·lules dels quals contenen un nucli) i procariotes (organismes les cèl·lules dels quals no tenen nucli com per exemple, un bacteri). En aquest text ens centrarem en genomes d'eucariotes i, en particular, en el genoma humà.

Els genomes eucariotes es divideixen en cromosomes. El genoma humà té dues còpies de cada cromosoma. Hi ha 23 parelles de cromosomes: 22 autosomes (dos còpies de cada, tant en homes com en dones) i dos cromosomes sexuals, que es denoten com a X i Y. Les dones tenen dos cromosomes X, i el homes tenen un X i un Y. Els pares passen un mosaic de les seues parelles de cromosomes als fills.

La seqüència de molècules d'ADN en un genoma es representa típicament com una seqüència de lletres del conjunt $\Omega = \{A, C, G, T\}$, particionades en cromosomes. Aquestes lletres corresponen a les bases de la doble hèlix, que són els nucleòtids adenina, citosina, guanina i timina. Com que cada base està emparellada amb una contrària en l'altra meitat de la doble hèlix (A amb T i C amb G), per tal de descriure un genoma és suficient fer una llista amb les bases en només una part. Tanmateix, és important notar que les dues cadenes tenen una direccionalitat que ve indicada pels nombres $5'$ i $3'$ al final (corresponents als àtoms de carboni en la columna de l'hèlix). La convenció és representar l'ADN en la direcció $5' \to 3'$. El genoma humà està constituït per aproximadament 2.8 bilions de bases, i s'ha obtingut emprant tecnologies de seqüenciació d'alt rendiment que es poden usar per a llegir la seqüència de fragments curts d'ADN de centenars de bases de longitud. Seguidament, s'usen els algoritmes de muntatge de seqüències per a ajuntar aquests fragments. Vegeu [19] per a més informació.

Tot i que els genomes es poden resumir com a cadenes en l'alfabet Ω, no s'ha d'oblidar que estan altament estructurats: per exemple, determinades seqüències dins d'un genoma corresponen a gens. Aquestes subseqüències juguen un paper important en la codificació de les proteïnes. Les proteïnes són polímers formats per vint tipus diferents d'aminoàcids. Dins dels gens, triplets d'ADN, coneguts com a codons, codifiquen els aminoàcids per a les proteïnes. Açò és conegut com el codi genètic. La Taula 3.1 mostra els 64 codons possibles i els 20 aminoàcids que codifiquen. Cada aminoàcid està representat per un identificador de tres lletres ('Phe' = Phenylalanine o fenilalanina, etc). Els tres codons TAA, TAG i TGA són especials: no codifiquen aminoàcids, sinó que s'usen per a indicar on acaba la proteïna.

	T	C	A	G
T	TTT $\mapsto$ Phe	TCT $\mapsto$ Ser	TAT $\mapsto$ Tyr	TGT $\mapsto$ Cys
	TTC $\mapsto$ Phe	TCC $\mapsto$ Ser	TAC $\mapsto$ Tyr	TGC $\mapsto$ Cys
	TTA $\mapsto$ Leu	TCA $\mapsto$ Ser	TAA $\mapsto$ stop	TGA $\mapsto$ stop
	TTG $\mapsto$ Leu	TCG $\mapsto$ Ser	TAG $\mapsto$ stop	TGG $\mapsto$ Trp
C	CTT $\mapsto$ Leu	CCT $\mapsto$ Pro	CAT $\mapsto$ His	CGT $\mapsto$ Arg
	CTC $\mapsto$ Leu	CCC $\mapsto$ Pro	CAC $\mapsto$ His	CGC $\mapsto$ Arg
	CTA $\mapsto$ Leu	CCA $\mapsto$ Pro	CAA $\mapsto$ Gln	CGA $\mapsto$ Arg
	CTG $\mapsto$ Leu	CCG $\mapsto$ Pro	CAG $\mapsto$ Gln	CGG $\mapsto$ Arg
A	ATT $\mapsto$ Ile	ACT $\mapsto$ Thr	AAT $\mapsto$ Asn	AGT$\mapsto$ Ser
	ATC $\mapsto$ Ile	ACC $\mapsto$ Thr	AAC $\mapsto$ Asn	AGC $\mapsto$ Ser
	ATA $\mapsto$ Ile	ACA $\mapsto$ Thr	AAA$\mapsto$ Lys	AGA $\mapsto$ Arg
	ATG $\mapsto$ Met	ACG $\mapsto$ Thr	AAG $\mapsto$ Lys	AGG $\mapsto$ Arg
G	GTT $\mapsto$ Val	GCT $\mapsto$ Ala	GAT $\mapsto$ Asp	GGT $\mapsto$ Gly
	GTC $\mapsto$ Val	GCC $\mapsto$ Ala	GAC $\mapsto$ Asp	GGC $\mapsto$ Gly
	GTA $\mapsto$ Val	GCA $\mapsto$ Ala	GAA $\mapsto$ Glu	GGA $\mapsto$ Gly
	GTG $\mapsto$ Val	GCG $\mapsto$ Ala	GAG $\mapsto$ Glu	GGG $\mapsto$ Gly

Taula 3.1: El codi genètic

Per tal de produir proteïnes, es copia primer l'ADN a una molècula semblant anomenada missatger RNA (abreviada mRNA) en un procés anomenat transcripció. És el RNA el que és convertit en una proteïna. El procés sencer s'anomena expressió. Les proteïnes poden ser elements estructurals o realitzar tasques (com la regulació de l'expressió) interactuant amb moltes molècules i cèl·lules. La comprensió dels gens, les funcions de les seues proteïnes, i els seus patrons d'expressió és fonamental en la biologia.

El genoma humà conté aproximadament 25000 gens, tot i que el nombre exacte encara no s'ha determinat. Malgrat haver-hi mètodes experimentals per a validar i descobrir gens, encara no hi ha cap tecnologia d'alt rendiment per a identificar acuradament tots els gens d'un genoma.

Les diferències entre els genomes d'individus d'una població són menudes i causades principalment per a la recombinació d'esdeveniments (part del procés en què dos còpies dels cromosomes dels pares es combinen en el fill descendent). D'altra banda, els genomes d'espècies diferents (classes d'organismes que no poden produir un descendent junts) tendeixen a divergir més. Les diferències dels genomes entre espècies es poden explicar per molts esdeveniments biològics, inclosos:

- Reorganització del genoma–comparant cromosomes d'espècies relacionades es mostren segments grans que s'han invertit (inversions), segments que s'han mogut (transposicions), fusions de cromosomes, i altres esdeveniments.

- Duplicació i pèrdua–alguns genomes han patit una duplicació del genoma complet. Com per exemple el genoma del llevat [16]. Cromosomes individuals o gens també es poden duplicar. Els esdeveniments de duplicació sovint van acompanyats per pèrdua de gens, on els gens redundants perden o adapten lentament la seua funció amb el pas del temps [6].

- Expansió parasitària–seccions grans del genoma són repetitives, compostes per elements que poden duplicar-se i reintegrar-se en el genoma.

- Mutació, inserció i eliminació puntual–les seqüències d'ADN muten, i en regions no funcionals aquestes mutacions s'acumulen amb el pas del temps. En aquestes regions és probable trobar eliminacions; per exemple, disminució de la cadena durant la replicació que pot comportar una còpia incorrecta del nombre de bases repetides.

Dues bases d'ADN que tenen un ancestre en comú s'anomenen homòlogues. Les bases homòlogues es poden relacionar mitjançant esdeveniments d'especiació i duplicació, i per això es divideixen en dues classes: ortòlogues i paràlogues. Les bases ortòlogues són descendents d'una única base del genoma de l'avantpassat que es relacionen mitjançant la duplicació. Com que no podem obtindre la seqüenciació del genoma de l'avantpassat, mai es pot demostrar formalment que dues bases d'ADN són homòlogues. Tantmateix, arguments estadístics poden mostrar que és extremadament probable que dues bases siguen homòlogues o fins i tot ortòlogues. El problema d'identificar bases homòlogues entre genomes d'espècies relacionades es coneix com el problema d'alineament.

L'alineació de genomes és el primer pas per tal d'identificar seqüències molt conservades que assenyalen el petit fragment del genoma que està en selecció, i per això és probable que siga funcional. Tot i que el problema de l'alineació de seqüències és matemàticament i computacionalment parlant desafiador, seqüències homòlogues proposades es poden validar de manera ràpida i independent (és fàcil determinar si dues seqüències s'alineen un cop s'han identificat), i les regions normalment es poden analitzar en un labolatori de biologia molecular per a determinar la seua funció. Dit d'un altra manera, l'alineació de seqüències revela evidències concretes per a la selecció de l'evolució i sovint resulta en hipòtesis que es poden comprovar en laboratoris.

Com a punt central per a la nostra discussió, presentem una seqüència d'ADN específica de longitud 42. Aquesta seqüència es va descobrir en la tardor del 2003 com a derivat de treball computacional realitzat pel grup de Lior Pachter a Berkeley [2]. Es descobriren i s'analitzaren tots els alineaments del genoma de l'humà (hs), el ximpanzè (pt), el ratolí (mm), la rata (rn), el gos (cf), el gall (gg), el peix zebra (dr), el peix globus (tr), i el peix fugu (tn). Les abreviacions es refereixen al nom en llatí d'aquests organismes. Dels alineaments dels 9 genomes, es va derivar la següent hipòtesi.

Conjectura 1. *La seqüència de 42 bases*

$$TTTAATTGAAAGAAGTTAATTGAATGAAAATGATCAACTAAG \tag{3.1}$$

estava present en el genoma de l'avantpassat de tots els vertebrats i s'ha conservat sencera fins al present (i.e., cap de les bases ha mutat, ni hi ha hagut cap inserció o eliminació).

La identificació de seqüències com aquesta requereix computacions no trivials: l'alineament de 9 genomes (inclosos genomes de mamífers de 3 bilions de bases de longitud) i anàlisis posteriors per a identificar regions ortòlogues conservades dins de l'alineació[25].

Identificar i analitzar seqüències com (3.1) és important perquè estan altament conservades i sovint no són gèniques [10]. Un dels misteris en curs en la biologia és desenredar la funció de les parts del genoma que no són gèniques i que també estan molt ben conservades.

L'any 2003, la seqüència (3.1) semblava ser la seqüència més llarga conservada completament entre els vertebrats. Potser que siga una coincidència que el segment anterior continga dues còpies del motiu TTAATTGAA, però potser que aquest motiu també tinga alguna funció (per exemple, pot estar delimitat per una proteïna). De fet, la identificació d'aquests elements és el primer pas per a entendre el complex codi regulador del genoma.

La conjectura es va formular en la primavera del 2004. En la tardor del mateix any, Drton, Eriksson, i Leung [15] dugueren a terme un nou estudi basat en alineaments millorats. El seu treball, i estudis similars realitzats per altres grups [8], han dut a terme a la identificació de seqüències més llargues amb propietats semblants (d'una longitud de 125 per al cas dels 9 vertebrats en qüestió).

3.3 Codons

Per raó del codi genètic, el conjunt Ω^3 format per totes les paraules de tres lletres de l'alfabet $\Omega = \{A, C, G, T\}$ juga un paper important en la biologia molecular. Tal com es va dir en la secció anterior, aquestes paraules s'anomenen codons, on cada triplet codifica un dels vint aminoàcids (Taula 3.1). La funció que tradueix els 64 codons als 20 aminoàcids no és injectiva, perquè diversos codons codifiquen el mateix aminoàcid (tal i com s'observa en la Taula 3.1). Els codons que codifiquen el mateix aminoàcid s'anomenen sinònims. Vuit aminoàcids tenen la propietat que els codons sinònims que els codifiquen tenen les dues primeres posicions iguals. La tercera posició d'aquest tipus de codó s'anomena posició amb degeneració quàdruple (four-fold degenerate en anglés). La traducció d'una sèrie de codons en un gen (uns centenars normalment) resulta en una proteïna plegada amb estructura tridimensional.

Definició 1. *Un model per a codons és un model estadístic l'espai d'estats del qual és el conjunt de 64 elements* Ω^3

Seleccionar un model significa especificar una família de distribucions de probabilitat $p =$

(p_{IJK}) en Ω^3. Cada distribució de probabilitat p és una taula $4 \times 4 \times 4$ de nombres reals no negatius que sumen en conjunt 1. Geomètricament parlant, una distribució en codons és un punt p en el 63-símplex

$$\Delta_{63} = \left\{ p \in \mathbb{R}^{|\Omega^3|} : \sum_{IJK \in \Omega^3} p_{IJK} = 1 \text{ i } p_{IJK} \geq 0 \ \forall IJK \in \Omega^3 \right\}.$$

Un model per a codons és per tant, un subconjunt $\mathcal{M}$ del símplex Δ_{63}. Models estadísticament significatius es donen de normal en forma paramètrica. Si el nombre de paràmetres és d, aleshores existeix un subconjunt $\mathcal{P} \subset \mathbb{R}^d$ de paràmetres permesos, i el model $\mathcal{M}$ és la imatge de la funció f de $\mathcal{P}$ a Δ_{63}. Per tal d'il·lustrar-ho exposarem un model d'independència senzill.

Considerem una seqüència d'ADN de longitud $3m$ que s'agrupa en m codons consecutius. Siga u_{IJK} el nombre d'ocurrències del codó IJK en particular. Aleshores, les nostres dades estan en la taula $4 \times 4 \times 4$ $u = (u_{IJK})$. Les entrades d'aquesta taula són enters no negatius, i si dividim cada entrada entre m, obtenim una nova taula $\frac{1}{m} \cdot u$ (amb les probabilitats relatives) que és un punt del símplex Δ_{63}. Aquesta taula és la distribució empírica dels codons per a la seqüència donada.

Siga $\mathcal{M}$ un model estadístic que estipula que, per a la seqüència considerada, les dues primeres posicions en un codó són independents de la tercera posició. Desitgem provar si aquest model d'independència encaixa amb les nostres dades u. Aquesta pregunta té sentit en biologia molecular perquè molts dels aminoàcids estan especificats excepcionalment per les dues primeres posicions en cada codó que representa l'aminoàcid en particular (vegeu Taula 3.1). Per això, els nucleòtids de la tercera posició de codons sinònims tendeixen a ser independents dels dos primers.

Així doncs, el nostre model d'independència $\mathcal{M}$ té 18 paràmetres lliures. El conjunt de paràmetres permesos és un polítop convex 18-dimensional (vegeu la secció 3.5.2), concretament, és el producte

$$\mathcal{P} = \Delta_{15} \times \Delta_3.$$

D'una banda hi ha les 16 possibilitats corresponents a les dues primeres posicions denotades pel símplex Δ_{15} de les distribucions de probabilitat $\alpha = (\alpha_{IJ})$ en Ω^2, i d'altra banda tenim les 4 possibilitats corresponents a la tercera posició denotades pel tetrahedre Δ_3 compost per les distribucions de probabilitat $\beta = (\beta_K)$ en Ω. El nostre model $\mathcal{M}$ està parametritzat per la funció

$$f : \mathcal{P} \to \Delta_{63}, \quad f((\alpha, \beta))_{IJK} = \alpha_{IJ} \cdot \beta_K.$$

D'aquí, $\mathcal{M} = Im(f)$ és un subconjunt algebraic 18-dimensional dins del símplex Δ_{63}. Per a saber si una taula $4 \times 4 \times 4$ donada p es troba en $\mathcal{M}$, escrivim la taula com una matriu

bidimensional amb 16 files i 4 columnes (un total de 64 entrades):

$$p' = \begin{pmatrix} p_{AAA} & p_{AAC} & p_{AAG} & p_{AAT} \\ p_{ACA} & p_{ACC} & p_{ACG} & p_{ACT} \\ p_{AGA} & p_{AGC} & p_{AGG} & p_{AGT} \\ p_{ATA} & p_{ATC} & p_{ATG} & p_{ATT} \\ p_{CAA} & p_{CAC} & p_{CAG} & p_{CAT} \\ \vdots & \vdots & \vdots & \vdots \\ p_{TTA} & p_{TTC} & p_{TTG} & p_{TTT} \end{pmatrix}$$

La següent proposició ens estableix les següents caracteritzacions per al nostre model [14].

Proposició 1. *Donat un punt $p \in \Delta_{63}$, les següents condicions són equivalents:*

1. *La distribució p es troba en el model $\mathcal{M}$.*
2. *La matriu 16×4 té rang 1.*
3. *Tots els menors 2×2 de la matriu p' són zero.*
4. *$p_{IJK} \cdot p_{LMN} = p_{IJN} \cdot p_{LMK}$ per a tots els nucleòtids $I, J, K, L, M, N \in \Omega$*

En el llenguatge de la geometria algebraica, el model $\mathcal{M}$ es coneix com a la varietat Segre.

Definició 2. *La varietat Segre és una varietat determinantal (espai de matrius amb una fita superior en el seu rang) en la qual, tots els menors 2×2 d'aquestes matrius són zero.*

Més precisament, $\mathcal{M}$ és el conjunt de punts reals no negatius en la immersió Segre de $\mathbb{P}^{15} \times \mathbb{P}^3$ en $\mathbb{P}^{63}$. En aquest text, el símbol $\mathbb{P}^m$ denota l'espai projectiu complex de dimensió m.

Retornant a l'objectiu inicial, ens topem amb el següent problema d'estadística. La seqüència d'ADN considerada està resumida en les dades u, i desitgem saber si el model $\mathcal{M}$ encaixa amb les dades. La idea geomètrica d'açò és determinar si la distribució empírica $\frac{1}{m} \cdot u$ es troba prop de la varietat Segre $\mathcal{M}$. Els estadístics han elaborat una àmplia gamma de contrasts d'hipòtesis. Açò inclou el test χ^2, entre d'altres. Una ferramenta útil d'àlgebra lineal per a mesurar la distància d'un punt a la varietat Segre és l'autovalor de la descomposició de la matriu p'. Encara més, p' es troba en $\mathcal{M}$ si i només si, el segon autovalor de p' és zero. Els autovalors proporcionen una bona noció de distància entre una matriu donada i diverses varietats determinants tals com $\mathcal{M}$.

Un ingredient clau en els contrasts d'hipòtesis és l'estimador de màxima versemblança. Si considerem totes les seqüències possibles d'un genoma de longitud $3m$, aleshores la probabilitat d'observar les nostres dades en concret (la funció de versemblança) és

$$\gamma \cdot \prod_{IJK \in \Omega^3} p_{IJK}^{u_{IJK}} ,$$

on γ és la constant combinatòria. Desitgem trobar el punt dins del domini del nostre paràmetre $\mathcal{P} = \Delta_{15} \times \Delta_3$ que maximitze aquesta funció que està en funció de (α, β), i.e. l'estimador màxim versemblant $(\hat{\alpha}, \hat{\beta})$ per a les dades u. En el nostre model d'independència, la funció de versemblança és convexa i es pot escriure el màxim global de manera explícita:

$$\hat{\alpha}_{IJ} = \frac{1}{m} \sum_{K \in \Omega} u_{IJK} \quad \text{i} \quad \hat{\beta}_K = \frac{1}{m} \sum_{IJ \in \Omega^2} u_{IJK}$$

En general, la funció de versemblança d'un model estadístic no és convexa, i no hi ha una fòrmula senzilla d'escriure l'estimador màxim versemblant com una funció de les dades. En la pràctica, s'empren mètodes numèrics d'escalada simple per a resoldre aquest problema d'optimització. No hi ha cap garantia però, que el màxim local trobat per aquest tipus de mètodes siga el màxim global.

3.4 Predicció de gens

Per tal de trobar gens en seqüències d'ADN, cal identificar característiques estructurals i característiques de seqüències que distingeixin seqüències gèniques de seqüències no gèniques. Comencem descrivint més en detall l'estructura del gen que és essencial per a desenvolupar models.

Els gens no són seqüències contínues del genoma, més aviat estan dividits en peces anomenades introns i exons. Després de la transcripció, els introns es desempalmen i són només els exons els que es gasten en la traducció (vegeu Figura 3.1). No tot el que hi ha en la seqüència d'exons es tradueix; el primer i últim exó poden estar compostos de regions no traduïdes (indicades en gris en la Figura 3.1). Com que el codi genètic està en triplets (no sobreposats), les longituds de les porcions traduïdes dels exons han de sumar en total 0 mod 3. A més a més, a banda de l'estructura exons-introns dels gens, hi ha seqüències que són senyals. El codó ATG inicia una traducció, i per tant és el primer codó seguit d'una porció no traduïda dels exons inicials. El codó final en un gen ha de ser un dels següents: TAG, TAA, o TGA (tal i com s'indica en la Taula 3.1). Aquests codons indiquen a la maquinària de traducció que pare. També hi ha marques de seqüències en la frontera entre un intró i un exó: GT al final de $5'$ (entre el primer exó i el primer intró d'esquerra a dreta en la Figura 3.1, 1a fila) i AG al final de $3'$ (entre l'últim intró indicat per les fletxes i l'últim exó d'esquerra a dreta en la Figura 3.1).

Per tal de modelitzar simultàniament les bases en una seqüència d'ADN de longitud n i les característiques estructurals associades a aquesta seqüència s'empra el Model Ocult de Màrkov.

3.4.1 El model ocult de Màrkov

Així com les cadenes de Màrkov ens permeten modelitzar els estats que prenen unes variables aleatòries segons les probabilitats de seqüències d'aquestes variables aleatòries (tot assumint que l'estat en el futur d'una variable depèn només de l'estat actual). El model ocult de Màrkov (HMM de l'anglés Hidden Markov Model), ens permet modelitzar seqüències de variables aleatòries quan no podem observar de manera directa esdeveniments (estan ocults). D'aquesta manera, ens permet modelitzar tant els esdeveniments observats (en el nostre cas la seqüències d'ADN), com els esdeveniments ocults (introns-exons) que afecten el resultat final (seqüència d'ADN).

Així doncs, si una cadena de Màrkov ve determinada pel conjunt dels k estats possibles que poden prendre les variables aleatòries $X_1, \ldots, X_n$, la matriu de probabilitats de transició (indica la probabilitat que la variable passe d'un estat a un altre) i la distribució de probabilitats inicials (indica la probabilitat que hi ha que la variable estiga inicialment en cada estat), un HMM ve determinat per aquests tres factors, més una seqüència de n observacions de les variables aleatòries $Y_1, \ldots, Y_n$ cadascuna pertanyent al mateix conjunt que prenen els l possibles estats i una seqüència de probabilitats d'emissió (o observació) que indiquen la probabilitat que una observació s'haja generat des de l'estat i. En el context de la filogenètica, les variables observades Y_i tenen normalment $l = 4$ estats, a saber, $\Omega = \{A, C, G, T\}$. Les variables aleatòries ocultes X_i serveixen per a modelitzar les característiques associades a la seqüència que està generada per $Y_1, Y_2, \ldots, Y_n$. Un escenari simplificat és $k = 2$, amb el conjunt d'estats ocults $\Theta = \{$ exó, intró $\}$.

La propietat característica d'un HMM és que les distribucions de cada Y_i depenen de les de X_i, mentre que les X_i formen una cadena de Màrkov. Açò s'il·lustra en la Figura 3.2, on els cercles blancs representen les variables ocultes X_1, X_2, X_3 i els cercles ombrejats representen les variables observades Y_1, Y_2, Y_3.

Els biòlegs computacionals usen models ocults de Màrkov per a anotar seqüències d'ADN. La idea bàsica és la següent: està postulat que les bases són instàncies de les variables aleatòries $Y_1, \ldots, Y_n$, i el problema és identificar els estats més probables de $X_1, \ldots, X_n$ que es puguen associar a aquestes observacions. En la predicció de gens, s'usen models ocults de Màrkov homogenis. Açò vol dir que totes les probabilitats de transició $X_i \to X_{i+1}$ venen donades per la matriu $k \times k$ $S = (s_{ij})$, i les probabilitats de transició $X_i \to Y_i$ venen donades per una altra matriu $k \times 4$ $T = (t_{ij})$. Aquí, s_{ij} representa la probabilitat que es passe de l'estat i a l'estat j; per exemple, si $k = 2$, aleshores $i, j \in \Theta = \{$ exó, intró $\}$. El paràmetre t_{ij} representa la probabilitat que l'estat $i \in \Theta$ produeixi $j \in \Omega$. En la pràctica, els paràmetres s_{ij} i t_{ij} són els nombres reals que satisfan

$$s_{ij}, t_{ij} \geq 0 \quad \text{i} \quad \sum_{j \in \Theta} s_{1j} = \sum_{j \in \Omega} t_{1j} = 1 \quad \forall i \in \Theta \tag{3.2}$$

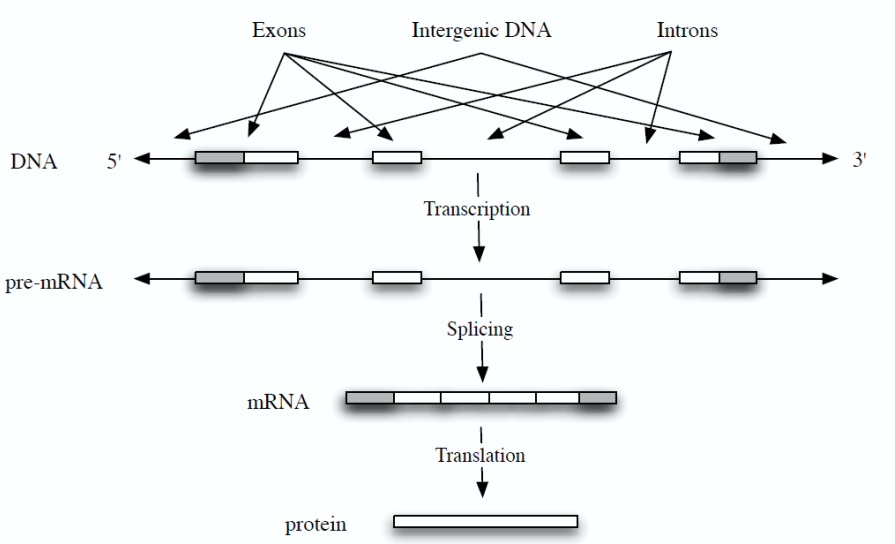


Figura 3.1: Estructura d'un gen.

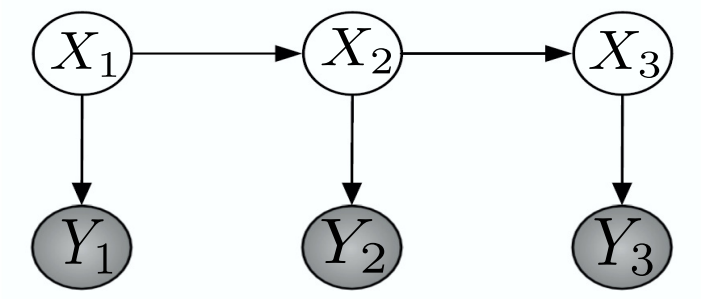


Figura 3.2: El model ocult de Màrkov de longitud tres.

No obstant això, de la mateixa manera que en la varietat Segre, podem relaxar les condicions de (3.2) i permetre que els paràmetres siguen nombres complexos arbitraris. Açò condueix a la següent representació algebraica [18].

Proposició 2. *El model ocult de Màrkov homogeni és la imatge d'una aplicació* $f : \mathbb{C}^{k(k+l)} \to \mathbb{C}^{l^n}$*, on cada coordenada de* f *és un polinomi bihomogeni de grau* $n-1$ *en les probabilitats de transició* s_{ij} *i de grau* n *en el resultat de les probabilitats* t_{ij}*.*

La coordenada f_σ de l'aplicació f indexada per una seqüència d'ADN en particular $\sigma \in \Omega^n$ representa la probabilitat que el HMM genere la seqüència σ. La següent fórmula explícita per a aquesta probabilitat estableix la Proposició 2:

$$f_\sigma = \sum_{i_1 \in \Theta} t_{i_1\sigma_1} \left(\sum_{i_2 \in \Theta} s_{i_1 i_2} t_{i_2\sigma_2} \left(\sum_{i_3 \in \Theta} s_{i_2 i_3} t_{i_3\sigma_3} \left(\sum_{i_4 \in \Theta} s_{i_3 i_4} t_{i_4\sigma_4} (\cdots) \right) \right) \right) \tag{3.3}$$

L'expansió d'aquest polinomi té k^n termes:

$$t_{i_1\sigma_1} s_{i_1 i_2} t_{i_2\sigma_2} s_{i_2 i_3} t_{i_3\sigma_3} \cdots s_{i_{n-1} i_n} t_{i_n\sigma_n} \tag{3.4}$$

Donats uns paràmetres, hom desitja determinar la cadena $\widehat{\mathbf{i}} = (i_1, i_2, \ldots, i_n) \in \Theta^n$ que indexa un terme de (3.4) amb el major valor numèric d'entre tots els k^n termes de f_σ. (Si hi ha més d'una cadena amb el valor màxim, es desempata amb l'ordre lexicogràfic). Anomenem $\widehat{\mathrm{i}}$ l'explicació de la seqüència d'ADN σ. En el nostre exemple ($k = 2, l = 4$), l'explicació $\widehat{\mathrm{i}}$ de la seqüència d'ADN σ és un element de $\Theta^n = \{$ exó, intró $\}^n$. Açò revela la informació crucial de la Figura 3.1: la localització dels exons i introns. En resum, la seqüència d'ADN per a ser anotada per un HMM correspon a l'observació $\sigma \in \Omega^n$, i l'explicació $\widehat{\mathbf{i}}$ és la predicció del gen. Així, la predicció del gen no vol dir res més que computar la sortida $\widehat{\mathrm{i}}$ de l'entrada σ.

En aplicacions del món real, l'enter n pot ser considerablement gran ($n \geq 1,000,000$). El tamany k^n de l'espai de cerca per a trobar l'explicació és enorme (exponencial en n). Afortunadament, la descomposició recursiva en (3.3), reminiscent de la regla de Horner, ens permet avaluar polinomis multivariants amb una quantitat exponencial de termes en temps lineal en n. A continuació expliquem el seu principi.

La regla de Horner s'aplica a polinomis de grau finit. Donat un polinomi $f(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n$ on x denota les variables i a_i els coeficients. Es reescriu el polinomi com a producte de monomis de la següent manera:

$$f(x_0) = a_0 + x_0 (a_1 + x_0 (a_2 + x_0 (a_3 + \ldots + (a_{n-1} + a_n x_0) \ldots .)$$

Així, a l'hora de calcular $f(x_0)$, es redueix el nombre de multiplicacions que l'ordinador ha de realitzar, disminuint al seu torn el temps de càlcul. De manera que el temps exponencial es

pot reduir a lineal en n. En altres paraules, donats els paràmetres s_{ij} i t_{ij}, es pot calcular la probabilitat $f_\sigma(s_{ij}, t_{ij})$ de manera eficient.

De manera similar, l'explicació $\widehat{\mathrm{i}}$ d'una seqüència d'ADN σ també es pot calcular en temps lineal mitjançant l'algoritme de Viterbi.
Aquest algoritme avalua

$$\max_{i_1 \in \Theta} T_{i_1\sigma_1} + \left(\max_{i_2 \in \Theta} S_{i_1 i_2} + T_{i_2\sigma_2} + \left(\max_{i_3 \in \Theta} S_{i_2 i_3} + T_{i_3\sigma_3} + \left(\max_{i_4 \in \Theta} S_{i_3 i_4} + T_{i_4\sigma_4} + (\ldots) \right) \right) \right)$$

on $S_{ij} = \log(s_{ij})$ i $T_{ij} = \log(t_{ij})$. Aquesta funció és una funció convexa lineal a trossos en $\mathbb{R}^{k(k+l)}$, coneguda com a la tropicalització (procés de passar de l'aritmètica clàssica a la tropical) del polinomi f_σ. Avaluar aquesta expressió requereix exactament les mateixes operacions que avaluar f_σ, amb l'única diferència que es reemplaça l'aritmètica ordinària pel semianell tropical.

3.4.2 Aritmètica tropical i programació dinàmica

Definició 3. *Donades dues operacions $\oplus, \otimes$ sobre un conjunt $\mathbb{S}$, es diu que $(\mathbb{S}, \oplus, \otimes)$ és un semianell si es compleixen totes les propietats d'un anell excepte l'existència d'element invers respecte de la primera operació $\oplus$.*

És a dir, el conjunt anterior $(\mathbb{S}, \oplus, \otimes)$, és un semianell si es compleixen les següents propietats:

1. Respecte a la suma $\oplus$, es compleix l'associativitat, la propietat commutativa i l'existència d'element neutre.
2. Respecte al producte $\otimes$, es compleix l'associativa i l'existència d'element neutre.
3. I finalment, es compleix la propietat distributiva.

L'anell tropical és en realitat el semianell màxim-suma, en el qual per a tots els nombres reals $\mathbb{R}$ junt amb $-\infty$, les operacions aritmètiques de la suma i el producte són el màxim (o equivalentment el mínim si en compte de $-\infty$ tenim ∞,) i la suma respectivament.

La programació dinàmica ofereix mètodes eficients per tal de construir progressivament un conjunt de puntuacions o probabilitats per a resoldre un problema. Una estructura algebraica convenient per a establir diversos algoritmes de programació dinàmica és el semianell tropical. Per tal d'il·lustrar açò, considerarem el problema de trobar el camí més curt en un graf dirigit ponderat (considerant el semianell tropical amb el mínim i ∞).

Siga G un graf dirigit ponderat amb n nodes etiquetats per $1, 2, \ldots, n$. Tota aresta dirigida (i, j) en G té una longitud associada d_{ij} que és un nombre real no negatiu. Si (i, j) no és una aresta de G, aleshores fixem $d_{ij} = +\infty$. Representem el graf per la seua matriu d'adjacència $D_G = (d_{ij})$ les entrades de la qual que no formen part de la diagonal són les longituds de les arestes d_{ij}. Les entrades de la diagonal són zero. Si G és un graf no dirigit amb longituds entre els vèrtexs, podem representar-lo com un graf dirigit amb dos arestes entre cada vèrtex (i, j) i (j, i). En aqueix cas en concret, la matriu D_G és simètrica. Per a un graf dirigit general, la matriu d'adjacència no serà simètrica.
Considerem el resultat de multiplicar tropicalment aquesta matriu amb si mateixa $n-1$ vegades:

$$D_G^{\otimes n-1} = D_G \otimes D_G \otimes \cdots \otimes D_G$$

Aquesta és una matriu $n \times n$ amb entrades en $\mathbb{R}_{\geq 0} \cup \{+\infty\}$.

Proposició 3. *Siga G un graf dirigit ponderat amb n nodes, i siga la seua matriu d'adjacència $n \times n$ D_G. Aleshores, l'entrada de la matriu $D_G^{\otimes n-1}$ en la fila i i la columna j és igual a la longitud del camí més curt des del node i fins al j en G.*

Demostració. Siga $d_{ij}^{(r)}$ la longitud mínima de qualsevol camí des del node i fins al j que usa com a molt r arestes en G. Tenim que $d_{ij}^{(1)} = d_{ij}$ per a dos nodes i i j qualssevol.

Com que s'ha assumit que els pesos de les arestes d_{ij} són no negatius, el camí més curt del node i al j visita cada node de G com a molt una vegada, i.e. com a molt visita $n-1$ arestes dirigides. Així, la longitud del camí més curt des de i fins a j és igual a $d_{ij}^{(n-1)}$.

Per a $r \geq 2$ tenim la següent fórmula recursiva per als camins més curts:

$$d_{ij}^{(r)} = \min\left\{d_{ik}^{(r-1)} + d_{kj} : k = 1, 2, \ldots, n\right\} \tag{3.5}$$

Usant aritmètica tropical, aquesta fórmula es pot reescriure de la següent manera:

$$d_{ij}^{(r)} = d_{i1}^{(r-1)} \otimes d_{1j} \oplus d_{i2}^{(r-1)} \otimes d_{2j} \oplus \cdots \oplus d_{in}^{(r-1)} \otimes d_{nj} \tag{3.6}$$

$$= \left(d_{i1}^{(r-1)}, d_{i2}^{(r-1)}, \ldots, d_{in}^{(r-1)}\right) \otimes (d_{1j}, d_{2j}, \ldots, d_{nj})^T \tag{3.7}$$

D'aquí, es pot demostrar per inducció en r que $d_{ij}^{(r)}$ coincideix amb l'entrada en la fila i i la columna j de la matriu $n \times n$ $D_G^{\otimes r}$. De fet, el costat de la dreta de la fórmula recursiva és el producte tropical de la fila i de la matriu $D_G^{\otimes r-1}$ per la columna j de la matriu D_G, que és l'entrada $D_G^{\otimes} r$. En particular, $d_{ij}^{(n-1)}$ coincideix amb l'entrada en la fila i i la columna j de $D_G^{\otimes n-1}$. □

Tornant a les funcions de predicció del gen, cada parella de paràmetres especifica una funció de predicció del gen

$$\Omega^n \to \Theta^n, \quad \sigma \mapsto \widehat{\mathbf{i}}$$

que pren una seqüència σ i obté la seua explicació $\widehat{\text{i}}$. El nombre de totes les funcions de Ω^n a Θ^n és $2^{n \cdot 4^n}$ i per tant, el seu creixement és doblement exponencial en n. La gran majoria d'aquestes funcions però, no són funcions de predicció de gens. El següent teorema fou demostrat per Elizalde [7].

Teorema 1. *El nombre de funcions de predicció de gens creix com a molt de manera polinòmica en n.*

3.4.3 Exemple

Per tal d'il·lustrar açò, considerem $n = 3$ com en la Figura 3.2. En aquest cas, hi ha un total de $8^{64} = 6.277 \cdot 10^{57}$ funcions de $\{A, C, G, T\}^3 \to \{$ exó, intró $\}^3$. Però només una petita fracció d'aquestes són funcions de predicció de gens. De fet, un problema obert és donar una caracterització combinatòria a les funcions de predicció de gens i plantejar fites acurades superiors i inferiors per al seu nombre en funció de n.

Per als models ocults de Màrkov de predicció de gens, sempre es compleix que l és menut i fix (de normal, $l = 4$), i n és gran. De tota manera, el tamany de k o l'estructura de l'espai d'estats per a les variables X_i sol variar molt. Tot i que en la discussió hem usat $k = 2$ per a les funcions de predicció de gens, un model de predicció de gens més significatiu des del punt de vista biològic podria funcionar amb només tres estats ocults: introns, exons, i seqüències intergèniques. No obstant això, perquè es compleixi la restricció de que la suma de les longituds dels exons és 0 mod 3, cal un espai d'estats ocults més complicat. Les solucions a aquest problema es troben en [4, 20].

Respecte als aspectes estadístics d'aquest model, cal recordar que en el cas dels codons (Secció 3.3), l'estimador màxim versemblant és una funció algebraica de les dades. A diferència del que es va realitzar al final de la (Secció 3.3), ara no es pot localitzar el màxim global del polítop (3.2). Establert açò, una tècnica que es pot emprar per a trobar màxims locals d'una funció de versemblança [19] és l'algoritme d'esperança-maximització (EM). Per als models HMM, aquest algoritme també es coneix com l'algoritme Baum-Welch.

3.4.4 Algoritme d'esperança-maximització

Si una dada (gen) és una seqüència de n lletres del conjunt Ω (que identifiquem amb el conjunt $[m] = \{1, 2, \ldots, m\}$), podem fer els recomptes de cada lletra (u_i) i identificar el gen amb el vector de freqüències $u = (u_1, u_2, \ldots, u_m)$. Si suposem un model paramètric per a les dades observades, podem expressar-lo mitjançant el vector $f(\theta) = (f_1(\theta), f_2(\theta), \ldots, f_m(\theta))$, on θ és el paràmetre i $f_i(\theta)$ expressa la probabilitat d'observar, a cada posició de la seqüència, la i-èssima lletra. Així doncs, es té que $\sum_{i=1}^{m} f_i(\theta) = 1$ per a tot valor de θ.

El vector u resumeix doncs una dada observada. Però considerarem que hi ha dades no observades (ocultes), que seran representades mitjançant matrius $U \in \mathbb{R}^{m\times n}$, de m files, de mode que la suma per files dona lloc a la dada observada.

De la mateixa manera, el model paramètric de les dades completes (ocultes), ve representat per una matriu de funcions $F = (f_{ij}(\theta))_{ij}$ de mode que la suma per files dona lloc al vector model paramètric de les dades observades. És a dir, $\sum_{j=1}^{n} f_{ij}(\theta) = f_i(\theta)$ per a tot i.

Amb això, la probabilitat d'observar una dada concreta $u = (u_1, \ldots, u_m)$ amb el model paramètric és $f_1(\theta)^{u_1} \cdot f_2(\theta)^{u_2} \cdots f_m(\theta)^{u_m}$. Aquesta expressió coincideix amb la versemblança (denotada per $L_{obs}(\theta)$) quan s'interpreta com a funció de θ. I normalment es treballa amb la log-versemblança $\ell_{obs}(\theta) := \log L_{obs}(\theta)$. Per al model ocult, les funcions de versemblança i log-versemblança es defineixen de la mateixa forma i es denoten per $L_{hid}(\theta)$ i $\ell_{hid}(\theta) := \log L_{hid}(\theta)$.

La idea de l'algoritme EM és la següent. Es comença amb una aproximació inicial del vector de paràmetres $\theta \in \Theta$, el domini dels paràmetres. Després es fa una estimació de les dades ocultes U. Aquest últim pas es coneix com a estimació (Pas-E). Seguidament, es resol el problema d'optimització en qüestió mitjançant una subrutina que s'assumeix que existeix per al model ocult en qüestió. Aquest pas es coneix com a maximització (Pas-M). Siga θ^* la solució òptima trobada en la maximització. Substituïm l'antiga aproximació inicial del paràmetre per la nova θ^* i iterem aquest procés fins que estem satisfets.
Una definició d'aquest algoritme escrit en pseudocodi [19] és la següent:
Input: Una matriu $m \times n$ de polinomis $f_{ij}(\theta)$ que representen el model ocult F i les dades observades $u = (u_1, \ldots, u_m) \in \mathbb{N}^m$.
Output: Un possible màxim $\widehat{\theta} \in \Theta \subset \mathbb{R}^d$ de la funció logaritme de versemblança $\ell_{obs}(\theta)$ del model observat f.

- Pas 0. Seleccionar un llindar $\epsilon > 0$ i un paràmetre θ inicial que satisfà $f_{ij}(\theta) > 0 \ \forall \ i, j$.
- Pas 1 (estimació). Definir la matriu esperada de dades ocultes $U = (u_{ij}) \in \mathbb{R}^{m\times n}$ amb

$$u_{ij} \quad := u_i \cdot \frac{f_{ij}(\theta)}{\sum_{j=1}^{m} f_{ij}(\theta)} = \frac{u_i}{f_i(\theta)} \cdot f_{ij}(\theta) \tag{3.8}$$

- Pas 2 (maximització). Calcular la solució $\theta^* \in \Theta$ del problema de maximització per al model ocult $F = (f_{ij})$.
- Pas 3. Si $\ell_{\text{obs}}\ (\theta^*) - \ell_{\text{obs}}\ (\theta) > \epsilon$ aleshores $\theta := \theta^*$ i es torna al pas 1 d'estimació .
- Pas 4. Retorna el vector de paràmetres $\widehat{\theta} := \theta^*$ i la corresponent distribució de probabilitats $\widehat{p} = f(\widehat{\theta})$ del conjunt $[m] := \{1, \ldots, m\}$.

La justificació d'aquest algoritme està donada pel següent teorema.

Teorema 2. *El valor de la funció de versemblança augmenta a cada iteració de l'algoritme EM. En particular, si θ es tria en el conjunt obert Θ abans del pas d'estimació i θ^* es calcula amb una iteració del pas d'estimació i una del de maximització, aleshores $L_{obs}(\theta) \leq L_{obs}(\theta^*)$. La igualtat es compleix si θ és un màxim local de la funció de versemblança.*

Demostració. Emprem el següent fet sobre la funció logarítmica d'un nombre positiu x:

$$\log(x) \leq x - 1 \text{ amb la igualtat sii } x = 1 \tag{3.9}$$

Siga $u \in \mathbb{N}^n$ i $\theta \in \Theta$ els paràmetres seleccionats a priori del pas d'estimació, siga $U = (u_{ij})$ la matriu calculada en el pas d'estimació, i siga $\theta^* \in \Theta$ el vector calculat en el subsegüent pas de maximització. Considerem la diferència entre els valors en θ^* i θ de la funció logarítmica de versemblança del model observat:

$$\ell_{obs}(\theta^*) - \ell_{obs}(\theta) = \sum_{i=1}^{m} u_i \cdot [\log(f_i(\theta^*)) - \log(f_i(\theta))] \tag{3.10}$$

$$= \sum_{i=1}^{m}\sum_{j=1}^{n} u_{ij} \cdot [\log(f_{ij}(\theta^*)) - \log(f_{ij}(\theta))] \tag{3.11}$$

$$+ \sum_{i=1}^{m} u_i \cdot \left(\log\left(\frac{f_i(\theta^*)}{f_i(\theta)}\right) - \sum_{j=1}^{n} \frac{u_{ij}}{u_i} \cdot \log\left(\frac{f_{ij}(\theta^*)}{f_{ij}(\theta)}\right)\right) \tag{3.12}$$

El doble sumatori és igual a $\ell_{\text{hid}}(\theta^*) - \ell_{\text{hid}}(\theta)$, i.e. la diferència entre els valors en θ^* i θ de la funció logarítmica de versemblança del model ocult . Aquesta diferència és no negativa perquè el vector de paràmetres θ^* ha sigut triat per a maximitzar la funció logarítmica de versemblança per al model ocult amb les dades (u_{ij}). Seguidament, demostrarem que l'última suma és també no negativa.

Analitzem per a això l'expressió en parèntesis

$$\log\left(\frac{f_i(\theta^*)}{f_i(\theta)}\right) - \sum_{j=1}^{n} \frac{u_{ij}}{u_i} \log\left(\frac{f_{ij}(\theta^*)}{f_{ij}(\theta)}\right) = \log\left(\frac{f_i(\theta^*)}{f_i(\theta)}\right) + \sum_{j=1}^{n} \frac{f_{ij}(\theta)}{f_i(\theta)} \log\left(\frac{f_{ij}(\theta)}{f_{ij}(\theta^*)}\right) \tag{3.13}$$

I reescrivim l'expressió de la següent manera:

$$\sum_{j=1}^{n} \frac{f_{ij}(\theta)}{f_i(\theta)} \cdot \log\left(\frac{f_i(\theta^*)}{f_i(\theta)}\right) + \sum_{j=1}^{n} \frac{f_{ij}(\theta)}{f_i(\theta)} \cdot \log\left(\frac{f_{ij}(\theta)}{f_{ij}(\theta^*)}\right) = \sum_{j=1}^{n} \frac{f_{ij}(\theta)}{f_i(\theta)} \cdot \log\left(\frac{f_i(\theta^*)}{f_{ij}(\theta^*)} \cdot \frac{f_{ij}(\theta)}{f_i(\theta)}\right) \tag{3.14}$$

Aquesta última expressió és no negativa, perquè si considerem les quantitats no negatives

$$\pi_j = \frac{f_{ij}(\theta)}{f_i(\theta)} \quad \text{i} \quad \sigma_j = \frac{f_{ij}(\theta^*)}{f_i(\theta^*)} \quad \text{for } j = 1, 2, \ldots, n$$

Tenim que $\pi_1 + \cdots + \pi_n = \sigma_1 + \cdots + i_n = 1$, de manera que els vectors π i σ es poden considerar distribucions de probabilitat en el conjunt $[n]$. L'expressió (3.14) és igual a la distància de Kullback-Leibler (malgrat no ser una mètrica per no ser simètrica ni complir la desigualtat triangular, sí que és no degenerada [1]) entre dues distribucions de probabilitat:

$$H(\pi\|\sigma) = \sum_{j=1}^{n} (-\pi_j) \cdot \log\left(\frac{\sigma_j}{\pi_j}\right) \geq \sum_{j=1}^{n} (-\pi_j) \cdot \left(1 - \frac{\sigma_j}{\pi_j}\right) = 0 \tag{3.15}$$

La desigualtat s'obté de (3.9). La igualtat en (3.15) es compleix sii $\pi = \sigma$. Aplicant l'expansió de Taylor a la diferència $\ell_{\text{obs}}\ (\theta^*) - \ell_{\text{obs}}\ (\theta)$, es veu que tot màxim local de la funció logarítmica de versemblança és un punt estacionari del logaritme EM, i a més a més, tot punt estacionari de l'algoritme EM ha de ser un punt crític de la funció logarítmica de versemblança [24]. □

En conclusió, aquest algoritme s'aprofita de la descomposició recursiva de (3.3) per tal que el temps de càlcul siga lineal.

3.5 Alineament de seqüències

Tot i que ferramentes com el HMM són importants per a modelitzar i analitzar seqüències de genomes individuals, una tasca fonamental en la filogenòmica és poder comparar seqüències. Puix que a les seqüències funcionals es tendeix a acumular menys mutacions, comparant els genomes és possible identificar i caracteritzar de manera més efectiva aquest tipus de seqüències.

Com que les seqüències biològiques en la pràctica són considerablement llargues, s'han desenvolupat algoritmes molt eficients per a trobar alineaments òptims. Tot i que en alguns casos s'empren algoritmes heurístics per a reduir la complexitat combinatòria, la majoria d'algoritmes estan basats en el principi de programació dinàmica. El nostre objectiu és mesurar la complexitat de transformar la seqüència σ^1 en la seqüència σ^2 canviant caràcters individualment, inserint-hi o eliminant-hi. Aquests canvis s'anomenen edicions.

Així doncs, donades dues seqüències $\sigma^1 = \sigma_1^1\sigma_2^1\cdots\sigma_n^1$ i $\sigma^2 = \sigma_1^2\sigma_2^2\cdots\sigma_m^2$ de l'alfabet $\Omega = \{A, C, G, T\}$. Un alineament grava 'els passos de modificació' de la seqüència σ^1 a la seqüència σ^2. Cada alineament de la parella $\left(\sigma^1, \sigma^2\right)$ es representa per una cadena h sobre l'alfabet $\{M, I, D\}$. Aquestes lletres es refereixen a mutació, inserció, eliminació. Una I representa una inserció en la primera seqüència σ^1, una D una eliminació en la primera seqüència σ^1, i una M, o bé un canvi de caràcter, o bé manca el caràcter. Si denotem $\#M, \#I$ i $\#D$ al nombre de caràcters M, I i D en la cadena d'edicions per a un alineament de la parella $\left(\sigma^1, \sigma^2\right)$, tenim que

$$\#M + \#D = n \quad \text{i} \quad \#M + \#I = m \tag{3.16}$$

Exemple Siga $n = 7$ i $m = 9$ i considerem les seqüències $\sigma^1 =$ ACGTAGC i $\sigma^2 =$ ACCGAGACC. Aleshores, la següent taula mostra un alineament de σ^1 i σ^2 amb $\#M = 6, \#I = 3$ i $\#D = 1$. La primera fila és la cadena d'edicions entre σ^1 i σ^2.

M	M	I	M	I	M	M	I	D	M
A	C	–	G	–	T	A	–	G	C
A	C	C	G	A	G	A	C	–	C

La transformació de σ^1 a σ^2 es representa per 5 passos d'edició que es realitzen d'esquerra a dreta. Aquesta transformació està codificada de manera única per la cadena d'edicions MMIMIMMIDM.

Proposició 4. *Una cadena sobre l'alfabet d'edicions $\{M, I, D\}$ representa un alineament d'una seqüència de n lletres σ^1 i una seqüència de m lletres σ^2 sii es compleix la condició 3.16.*

Demostració. Com que realitzem les edicions d'esquerra a dreta, cada lletra en σ^1 correspon o a una lletra en σ^2, o s'elimina. En el primer cas, s'obté una M en la cadena d'edicions, mentre que el segon cas correspon a una eliminació (obtenim una D). De manera que el total d'eliminacions i mutacions ha de ser igual a la longitud de la primera cadena.

La segona igualtat també es compleix perquè cada lletra en σ^2 o correspon a una lletra en σ^1 (i per tant, anotem una M en la cadena d'edicions), o s'ha inserit (i per tant, anotem una I en la cadena d'edicions).

Així que qualsevol cadena en l'alfabet $\{M, I, D\}$ que compleixi 3.16, produeix una seqüència d'edicions vàlida que transforma σ^1 en σ^2. □

La següent pregunta que ens ve al cap és: "donades dues seqüències, quants alineaments hi ha en total?". El conjunt de totes les cadenes d'edicions que compleixen la condició 3.16 es denota per $\mathcal{A}_{n,m}$ i l'anomenem el conjunt de tots els alineaments de les seqüències σ^1 i σ^2 en Ω amb longitud n i m respectivament. Cada element $h \in \mathcal{A}_{n,m}$ correspon a un parell de seqüències (μ^1, μ^2) en l'alfabet $\Omega \cup \{-\}$ tal que μ^1 és una còpia de σ^1 amb '-' inserits, i de manera similar, μ^2 és una còpia de σ^2 amb '-' inserits. Les cardinalitats dels conjunts $\mathcal{A}_{n,m}$ són els nombres combinatoris de Delannoy [21]. Aquests nombres descriuen el nombre de camins que es poden realitzar des del cantó $(0, 0)$ fins al cantó (n, m) d'un rectangle amb només passos cap al nord, cap a l'est i cap al nord-est. Dins del camp de la bioinformàtica, també compten el nombre d'alineaments de dues seqüències de longitud n i m respectivament (perquè es continuen tenint els tres possibles passos: M, I i D). La seua fórmula és:

$$D(n, m) = \sum_{k=0}^{\min(n,m)} \binom{n}{k} \binom{m}{k} 2^k \tag{3.17}$$

Aquests nombres però, es poden obtindre també arran d'una funció generatriu.

Proposició 5. *La cardinalitat del conjunt $\mathcal{A}_{n.m}$ de tots els alineaments es pot calcular com el coeficient del monomi $x^n y^m$ en la funció generatriu*

$$\frac{1}{1-x-y-xy} = 1 + x + y + x^2 + 3xy + y^2 + \cdots + x^5 + 9x^4y + 25x^3y^2 + \cdots \tag{3.18}$$

Demostració. Considerem l'expansió de la funció donada

$$\frac{1}{1-x-y-xy} = \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} a_{n,m}x^n y^m \tag{3.19}$$

Els coeficients estan caracteritzats per la relació de recurrència lineal $a_{n,m} = a_{n-1,m} + a_{n,m-1} + a_{n-1,m-1}$ amb $a_{0,0} = 1, a_{n,-1} = a_{-1,m} = 0$. Aquesta mateixa recurrència es pot aplicar a la cardinalitat de $\mathcal{A}_{n,m}$, perquè per a $m+n \geq 1$, tenim 3 possibilitats per a tota cadena en $\mathcal{A}_{n,m}$:

1. Pot ser una cadena en $\mathcal{A}_{n-1,m-1}$ seguida d'una M.
2. Una cadena en $\mathcal{A}_{n-1,m}$ seguida d'una I.
3. Una cadena en $\mathcal{A}_{n,m-1}$ seguida d'una D.

A més a més, $\mathcal{A}_{0,0}$ només té un element, que és la cadena buida i $\mathcal{A}_{n,m}$ és el conjunt buit si $m < 0$ o $n < 0$.

Per tant, els nombres $a_{m,n}$ i $\#\mathcal{A}_{n,m}$ compleixen les mateixes condicions inicials i la mateixa recurrència de manera que han de ser iguals. □

Així doncs, si $n = 2$ i $m = 3$ hi ha $D(2,3) = 25 = |\mathcal{A}_{2,3}| = 25$ alineaments possibles de dues seqüències de longitud dos i tres (escrits a la Taula 3.2).

Passem ara a relacionar els alineaments amb els grafs.

Definició 4. *L'alineament del graf $\mathcal{G}_{n,m}$ és el graf dirigit en el conjunt de nodes $\{0,1,\ldots,n\} \times \{0,1,\ldots,m\}$ i conté tres classes d'arestes: I entre les parelles de nodes $(i,j) \to (i,j+1)$, D entre les parelles de nodes $(i,j) \to (i+1,j)$, i M entre les parelles de nodes $(i,j) \to (i+1,j+1)$.*

Definit açò cal remarcar que existeix una bijecció entre el conjunt $\mathcal{A}_{n,m}$ i el conjunt de tots els camins des del node $(0,0)$ al node (n,m) en el graf de l'alineament $\mathcal{G}_{n,m}$.

Hem introduït tres objectes combinatoris equivalents: cadenes sobre l'alfabet $\{M,I,D\}$ que compleixen la condició (3.16), parelles de seqüències $\left(\mu^1,\mu^2\right)$ que són equivalents a σ^1,σ^2 amb la

IIIDD	$(\cdot\cdot\cdot ij, klm\cdot\cdot)$	$\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{II}\theta_{Im}\theta'_{ID}\theta_{Di}\theta'_{DD}\theta_{Dj}$
IIDID	$(\cdot\cdot i\cdot j, kl\cdot m\cdot)$	$\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{ID}\theta_{Di}\theta'_{DI}\theta_{Im}\theta_{ID'}\theta_{Dj}$
IIDDI	$(\cdot\cdot ij\cdot, kl\cdot\cdot m)$	$\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{ID}\theta_{Di}\theta'_{DD}\theta_{Dj}\theta'_{DI}\theta_{Im}$
IDIID	$(\cdot i\cdot\cdot j, k\cdot lm\cdot)$	$\theta_{Ik}\theta'_{ID}\theta_{Di}\theta'_{DI}\theta_{Il}\theta'_{II}\theta_{Im}\theta'_{ID}\theta_{Dj}$
IDIDI	$(\cdot i\cdot j\cdot, k\cdot l\cdot m)$	$\theta_{Ik}\theta'_{ID}\theta_{Di}\theta'_{DI}\theta_{Il}\theta'_{ID}\theta_{Dj}\theta'_{DI}\theta_{Im}$
IDDII	$(\cdot ij\cdot\cdot, k\cdot\cdot lm)$	$\theta_{Ik}\theta'_{ID}\theta_{Di}\theta'_{DD}\theta_{Dj}\theta'_{DI}\theta_{Il}\theta'_{II}\theta_{Im}$
DIIID	$(i\cdot\cdot\cdot j, \cdot klm\cdot)$	$\theta_{Di}\theta'_{DI}\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{II}\theta_{Im}\theta'_{ID}\theta_{Dj}$
DIIDI	$(i\cdot\cdot j\cdot, \cdot kl\cdot m)$	$\theta_{Di}\theta'_{DI}\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{ID}\theta_{Dj}\theta'_{DI}\theta_{Im}$
DIDII	$(i\cdot j\cdot\cdot, \cdot k\cdot lm)$	$\theta_{Di}\theta'_{DI}\theta_{Ik}\theta'_{ID}\theta_{Dj}\theta'_{DI}\theta_{Il}\theta'_{II}\theta_{Im}$
DDIII	$(ij\cdot\cdot, \cdot\cdot klm)$	$\theta_{Di}\theta'_{DD}\theta_{Dj}\theta'_{DI}\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{II}\theta_{Im}$
MIID	$(i\cdot\cdot j, klm\cdot)$	$\theta_{Mik}\theta'_{MI}\theta_{Il}\theta'_{II}\theta_{Im}\theta'_{ID}\theta_{Dj}$
MIDI	$(i\cdot j\cdot, kl\cdot m)$	$\theta_{Mik}\theta'_{MI}\theta_{Il}\theta'_{ID}\theta_{Dj}\theta'_{DI}\theta_{Im}$
MDII	$(ij\cdot\cdot, k\cdot lm)$	$\theta_{Mik}\theta'_{MD}\theta_{Dj}\theta'_{DI}\theta_{Il}\theta'_{II}\theta_{Im}$
IMID	$(\cdot i\cdot j, klm\cdot)$	$\theta_{Ik}\theta'_{IM}\theta_{Mil}\theta'_{MI}\theta_{Im}\theta'_{ID}\theta_{Dj}$
IMDI	$(\cdot ij\cdot, kl\cdot m)$	$\theta_{Ik}\theta'_{IM}\theta_{Mil}\theta'_{MD}\theta_{Dj}\theta'_{DI}\theta_{Im}$
IIMD	$(\cdot\cdot ij, klm\cdot)$	$\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{IM}\theta_{Mim}\theta'_{MD}\theta_{Dj}$
IIDM	$(\cdot\cdot ij, kl\cdot m)$	$\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{ID}\theta_{Di}\theta'_{DM}\theta_{Mjm}$
IDMI	$(\cdot ij\cdot, k\cdot lm)$	$\theta_{Ik}\theta'_{ID}\theta_{Di}\theta'_{DM}\theta_{Mjl}\theta'_{MI}\theta_{Im}$
IDIM	$(\cdot i\cdot j, k\cdot lm)$	$\theta_{Ik}\theta'_{ID}\theta_{Di}\theta'_{DI}\theta_{Il}\theta'_{IM}\theta_{Mjm}$
DMII	$(ij\cdot\cdot, \cdot klm)$	$\theta_{Di}\theta'_{DM}\theta_{Mjk}\theta'_{MI}\theta_{Il}\theta'_{II}\theta_{Im}$
DIMI	$(i\cdot j\cdot, \cdot klm)$	$\theta_{Di}\theta'_{DI}\theta_{Ik}\theta'_{IM}\theta_{Mjl}\theta'_{MI}\theta_{Im}$
DIIM	$(i\cdot\cdot j, \cdot klm)$	$\theta_{Di}\theta'_{DI}\theta_{Ik}\theta'_{II}\theta_{Il}\theta'_{IM}\theta_{Mjm}$
MMI	$(ij\cdot, klm)$	$\theta_{Mik}\theta'_{MM}\theta_{Mjl}\theta'_{MI}\theta_{Im}$
MIM	$(i\cdot j, klm)$	$\theta_{Mik}\theta'_{MI}\theta_{Il}\theta'_{IM}\theta_{Mjm}$
IMM	$(\cdot ij, klm)$	$\theta_{Ik}\theta'_{IM}\theta_{Mil}\theta'_{MM}\theta_{Mjm}$

Taula 3.2: Tots els possibles alineaments d'una parella de seqüències de longitud 2 i 3 respectivament.

possibilitat d'inserir '-', i camins en el graf d'alineament $\mathcal{G}_{n,m}$. Tots tres representen alineaments. El problema de l'alineament de seqüències és trobar la seqüència d'edicions més curta que relacione ambdues seqüències. La seqüència d'edicions més curta entre σ_1 i σ_2 té com a molt $n+m$ edicions, i per tant, el problema d'identificar el millor alineament és finit. Hom pot enumerar totes les seqüències d'edicions possibles i després triar la més curta. Però aquest mètode es pot millorar considerablement. Més endavant presentarem un algoritme de programació dinàmica per a resoldre el problema de l'alineament que és només d'ordre $O(nm)$. Per tal de poder parlar de manera formal sobre algoritmes que resolen el problema de l'alineament de dues seqüències ens calen un parell de definicions.

Definició 5. *Un esquema de puntuació és el parell de les funcions*

$$w : \Omega \cup \{-\} \times \Omega \cup \{-\} \to \mathbb{R} \tag{3.20}$$
$$w' : \{M, I, D\} \times \{M, I, D\} \to \mathbb{R} \tag{3.21}$$

Els esquemes de puntuació indueixen pesos en els alineaments de seqüències de la següent forma. Donades dues seqüències σ^1 i σ^2 en l'alfabet $\Omega = \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$. Cada alineament ve donat per una cadena d'edicions h en $\{M, I, D\}$. Escrivim $|h|$ per a la longitud d'aquesta cadena. La cadena d'edicions h determina les dues seqüències μ^1 i μ^2 de longitud $|h|$ en $\Omega \cup \{-\}$. El pes de l'alineament h està definit com a

$$W(h) \quad := \sum_{i=1}^{|h|} w\left(\mu_i^1, \mu_i^2\right) + \sum_{i=2}^{|h|} w'\left(h_{i-1}, h_i\right) \tag{3.22}$$

Representem un esquema de puntuació (w, w') per una parella de matrius. La primera és

$$w \quad = \begin{pmatrix} w_{\mathrm{A,A}} & w_{\mathrm{A,C}} & w_{\mathrm{A,G}} & w_{\mathrm{A,T}} & w_{\mathrm{A},-} \\ w_{\mathrm{C,A}} & w_{\mathrm{C,C}} & w_{\mathrm{C,G}} & w_{\mathrm{C,T}} & w_{\mathrm{C},-} \\ w_{\mathrm{G,A}} & w_{\mathrm{G,C}} & w_{\mathrm{G,G}} & w_{\mathrm{G,T}} & w_{\mathrm{G},-} \\ w_{\mathrm{T,A}} & w_{\mathrm{T,C}} & w_{\mathrm{T,G}} & w_{\mathrm{T,T}} & w_{\mathrm{T},-} \\ w_{-,\mathrm{A}} & w_{-,\mathrm{C}} & w_{-,\mathrm{G}} & w_{-,\mathrm{T}} & w_{-,-} \end{pmatrix} \tag{3.23}$$

La segona matriu és de dimensió 3×3:

$$w' = \begin{pmatrix} w'_{M,M} & w'_{M,I} & w'_{M,D} \\ w'_{I,M} & w'_{I,I} & w'_{I,D} \\ w'_{D,M} & w'_{D,I} & w'_{D,D} \end{pmatrix} \tag{3.24}$$

Notem que el paràmetre $w_{-,-}$ és zero perquè en un alineament de dues seqüències mai s'observarà la columna formada per dos caràcters '-'. Així, el nombre total de paràmetres en el problema de l'alineament és $24 + 9 = 33$. L'espai dels paràmetres d'aquest problema l'identifiquem amb $\mathbb{R}^{33}$. Cada alineament $h \in \mathcal{A}_{n,m}$ d'un parell de seqüències $\left(\sigma^1, \sigma^2\right)$ dona lloc a una funció lineal $W(h)$ en $\mathbb{R}^{33}$.

Per exemple, el pes de l'alineament $h = MMIMIMMMIDM$ de les seqüències de l'anterior exemple $\sigma^1 = \text{ACGTAGC}$ i $\sigma^2 = \text{ACCGAGACC}$ és la funció lineal

$$W(h) = 2{\cdot}w_{\text{A,A}}+2{\cdot}w_{\text{C,C}}+w_{\text{G,G}}+w_{\text{T,G}}+2{\cdot}w_{-,\text{C}}+w_{-,\text{A}}+w_{\text{G},-}+2{\cdot}w'_{M,M}+3{\cdot}w'_{M,I}+2{\cdot}w'_{I,M}+w'_{I,D}+w'_{D,M} \tag{3.25}$$

Llavors, donades dues seqüències σ^1 i σ^2 de longituds n i m en l'alfabet Ω i fixat un esquema de puntuació (w, w'), el problema de l'alineament es redueix a calcular un alineament $h \in \mathcal{A}_{n,m}$ el pes del qual $W(h)$ siga mínim entre tots els alineaments $\mathcal{A}_{n,m}$. Per tal de simplificar el problema, assumirem que $w' = 0$, de manera que el pes d'un alineament és la funció lineal $W(h) = \sum_{i=1}^{|h|} w\left(\mu_i^1, \mu_i^2\right)$ en $\mathbb{R}^{24}$. La instància del problema $\left(\sigma^1, \sigma^2, w\right)$ indueix uns pesos en les arestes del graf d'alineament $\mathcal{G}_{n,m}$ de la següent manera. El pes de l'aresta $(i,j) \to (i+1,j)$ és $w\left(\sigma_{i+1}^1, -\right)$, el pes de l'aresta $(i,j) \to (i,j+1)$ és $w\left(-, \sigma_{j+1}^2\right)$, i el pes de l'aresta $(i,j) \to (i+1,j+1)$ és $w\left(\sigma_{i+1}^1, \sigma_{j+1}^2\right)$. Açò reformula el problema de l'alineament com un problema de teoria de grafs en què s'ha de trobar el camí més curt de $(0,0)$ a (n,m) en el graf de l'alineament $\mathcal{G}_{n,m}$.

La Proposició 3 va donar una algoritme general per al problema del camí més curt, l'algoritme Floyd-Warshall, que equival a multiplicar matrius en l'aritmètica tropical. Per al graf específic i els corresponents pesos del problema d'alineament, açò es tradueix en un algoritme d'ordre $O(nm)$, anomenat l'algoritme de Needleman-Wunsch.

Algoritme 1. *(Algoritme de Needleman- Wunsch)*
Input: *Dues seqüències $\sigma^1 \in \Omega^n, \sigma^2 \in \Omega^m$ i un esquema de puntuació $w \in \mathbb{R}^{24}$.*
Output: *Un alineament $h \in \mathcal{A}_{n,m}$ el pes del qual $W(h)$ és mínim.*

- **Inicialització:** creem una matriu M de dimensió $(n+1)\times(m+1)$ començant la numeració de les files i columnes en 0 i acabant en n per a les files i m per a les columnes.

$$\text{Siga } M[0,0] = 0 \tag{3.26}$$
$$\text{Siga } M[i,0] := M[i-1,0] + w\left(\sigma_i^1, -\right) \quad \text{per a} \quad i = 1, \ldots, n \tag{3.27}$$
$$\text{Siga } M[0,j] := M[0,j-1] + w\left(-, \sigma_j^2\right) \quad \text{per a} \quad j = 1, \ldots, m \tag{3.28}$$

- **Bucle:** Per a $i = 1, \ldots, n$ i $j = 1, \ldots, m$, siga

$$M[i,j] := \min \begin{cases} M[i-1,j-1] + w\left(\sigma_i^1, \sigma_j^2\right) \\ M[i-1,j] + w\left(-, \sigma_j^2\right) \\ M[i,j-1] + w\left(\sigma_i^1, -\right) \end{cases} \tag{3.29}$$

  Marquem una o més de les tres arestes que són adjacents a $M[i,j]$, en direcció a (i,j) i que assoleixen el mínim.

- **Retorn:** tracem un camí òptim en direcció contrària (de (n,m) a $(0,0)$). Aquest camí s'aconsegueix seguint una seqüència arbitrària d'arestes marcades.

- **Output:** Les etiquetes de les arestes en $\{M,I,D\}$ del camí òptim en la direcció adient (de $(0,0)$ a (n,m)).

El cas més general (quan la matriu 3×3 $w' \neq 0$) es pot modelitzar reemplaçant cada node interior en $\mathcal{G}_{n,m}$ per un graf bipartit $K_{3,3}$ els pesos de les arestes del qual són $w'_{MM}, w'_{M,I}, \ldots, w'_{DD}$. Aquestes $9(m-1)(n-1)$ noves arestes representen transicions entre els diferents estats $\{M,I,D\}$. El graf resultant es denota per $\mathcal{G}'_{n,m}$ i s'anomena el graf d'alineament estès.

Exemple Considerem les seqüències de l'exemple anterior $\sigma^1 =$ ACGTAGC i $\sigma^2 =$ ACCGAGACC. Segons la Proposició 5, el nombre d'alineaments és

$$\#\mathcal{A}_{7,9} = 224.143 \tag{3.30}$$

El graf de l'alineament $\mathcal{G}_{7,9}$ es representa a continuació.

Per a qualsevol esquema de puntuació triat $w \in \mathbb{R}^{24}$, l'algoritme Needleman-Wunsch troba un alineament òptim. Considerem per a l'exemple l'esquema de puntuació en concret

$$w = \begin{pmatrix} -91 & 114 & 31 & 123 & x \\ 114 & -100 & 125 & 31 & x \\ 31 & 125 & -100 & 114 & x \\ 123 & 31 & 114 & -91 & x \\ x & x & x & x & x \end{pmatrix}$$

on la penalització x, perquè hi haja un buit, és un nombre desconegut entre 150 i 200. Per a $x \geq 169.5$ un alineament òptim és

$$\begin{pmatrix} h \\ \mu^1 \\ \mu^2 \end{pmatrix} = \begin{pmatrix} M & D & M & M & D & M & M & M & M \\ A & - & C & G & - & T & A & G & C \\ A & C & C & G & A & G & A & C & C \end{pmatrix} \quad \text{amb } W(h) = 2x - 243$$

Si la penalització per buit x és inferior a 169.5, aleshores l'alineament òptim és

$$\begin{pmatrix} h \\ \mu^1 \\ \mu^2 \end{pmatrix} = \begin{pmatrix} M & D & M & M & I & M & M & D & D & M \\ \text{A} & - & \text{C} & \text{G} & \text{T} & \text{A} & \text{G} & - & - & \text{C} \\ \text{A} & \text{C} & \text{C} & \text{G} & - & \text{A} & \text{G} & \text{A} & \text{C} & \text{C} \end{pmatrix} \quad \text{amb } W(h) = 4x - 582$$

Ara canviem de ferramenta i presentem el model ocult de Màrkov parell per al problema de l'alineament de dues seqüències.

3.5.1 El model HMM parell

Ja hem vist dos casos de models estadístics que es representen com a polinomis: el model dels codons i el HMM. Els models per a l'alineament de parelles de seqüències també estan especificats per polinomis, i de fet, s'assemblen als models HMM. El que diferencia el problema de l'alineament d'una seqüència és una capa extra de complexitat que sorgeix del gran nombre de possibles alineaments entre dues seqüències.

En aquesta secció s'examina un dels models per a l'evolució de seqüències que permet insercions, eliminacions i mutacions entre dues seqüències d'ADN, anomenats models d'alineament de parelles. El model en concret que revisarem serà el model HMM per a parelles (o HMM parell).

El model ocult de Màrkov parell per al problema de l'alineament és un model estadístic algebraic que depèn dels dos enters n i m:

$$\mathbf{f} : \mathbb{R}^{33} \to \mathbb{R}^{4^{n+m}} \tag{3.31}$$

on n i m són les longituds de cada seqüència.

Hi ha 4^{n+m} parelles $\left(\sigma^1, \sigma^2\right)$ de seqüències de longitud n i m. Els $33 = 24 + 9$ paràmetres s'escriuen com una parella de matrius (θ, θ') on

$$\theta = \begin{pmatrix} \theta_{\mathrm{A,A}} & \theta_{\mathrm{A,C}} & \theta_{\mathrm{A,G}} & \theta_{\mathrm{A,T}} & \theta_{\mathrm{A},-} \\ \theta_{\mathrm{C,A}} & \theta_{\mathrm{C,C}} & \theta_{\mathrm{C,G}} & \theta_{\mathrm{C,T}} & \theta_{\mathrm{C},-} \\ \theta_{\mathrm{G,A}} & \theta_{\mathrm{G,C}} & \theta_{\mathrm{G,G}} & \theta_{\mathrm{G,T}} & \theta_{\mathrm{G},-} \\ \theta_{\mathrm{T,A}} & \theta_{\mathrm{T,C}} & \theta_{\mathrm{T,G}} & \theta_{\mathrm{T,T}} & \theta_{\mathrm{T},-} \\ \theta_{-,\mathrm{A}} & \theta_{-,\mathrm{C}} & \theta_{-,\mathrm{G}} & \theta_{-,\mathrm{T}} & \theta_{-,-} \end{pmatrix}, \quad \theta' = \begin{pmatrix} \theta'_{M,M} & \theta'_{M,I} & \theta'_{M,D} \\ \theta'_{I,M} & \theta'_{I,I} & \theta'_{I,D} \\ \theta'_{D,M} & \theta'_{D,I} & \theta'_{D,D} \end{pmatrix} \tag{3.32}$$

on la primera matriu θ representa les probabilitats que les seqüències donades coincideixin en una posició determinada o no, i la matriu θ' conté les probabilitats corresponent als possibles estats de la cadena d'edicions. Per exemple, $\theta_{A,C}$ és la probabilitat que donat un alineament de dues seqüències σ_1 i σ_2, la posició determinada d'aquest alineament de σ_1 siga A i la de σ_2 siga C. D'una manera semblant, $\theta'_{M,M}$ representa la probabilitat que si en una posició donada s'ha observat una M, a la següent posició també s'observe una M. Per tal que aquests paràmetres siguen significatius estadísticament parlant, han de ser no negatius i complir les sis equacions lineals d'independència derivades dels símplexs que delimiten l'espai de paràmetres:

$$\Theta \quad = \Delta_{15} \times \Delta_3 \times \Delta_3 \times \Delta_2 \times \Delta_2 \times \Delta_2 \quad \subset \quad \mathbb{R}^{33} \tag{3.33}$$

L'espai de paràmetres Θ és el producte dels sis símplexs de dimensions $15, 3, 3, 2, 2$ i 2. El símplex Δ_{15} està compost per totes les matrius no negatives $(\theta_{ij})_{i,j\in\Omega}$ de dimensió 4×4 les entrades de les quals sumen en total 1. Els dos tetraedres Δ_3 venen del requeriment que

$$\theta_{-,\mathrm{A}} + \theta_{-,\mathrm{C}} + \theta_{-,\mathrm{G}} + \theta_{-,\mathrm{T}} = \theta_{\mathrm{A},-} + \theta_{\mathrm{C},-} + \theta_{\mathrm{G},-} + \theta_{\mathrm{T},-} = 1$$

Observem que no s'hi ha inclòs $\theta_{-,-}$ perquè un alineament no tindrà mai buits '-' en la mateixa posició d'ambdues seqüències, pel que $\theta_{-,-} = 0$. Finalment, els tres triangles Δ_2 venen de requerir

$$\theta'_{M,M} + \theta'_{M,I} + \theta'_{M,D} = \theta'_{I,M} + \theta'_{I,I} + \theta'_{I,D} = \theta'_{D,M} + \theta'_{D,I} + \theta'_{D,D} = 1$$

Les coordenades f_{σ^1,σ^2} del HMM parell f representen les probabilitats d'observar la parella de seqüències $\left(\sigma^1, \sigma^2\right)$. Cada coordenada ve representada pel polinomi

$$f_{\sigma^1,\sigma^2} = \sum_{h \in \mathcal{A}_{n,m}} \prod_{i=1}^{|h|} \theta_{\mu_i^1,\mu_i^2} \cdot \prod_{i=2}^{|h|} \theta'_{h_{i-1},h_i} \tag{3.34}$$

Aquí, $\left(\mu^1, \mu^2\right)$ és la parella de seqüències en $\Omega \cup \{-\}$ que correspon a l'alineament h. La següent proposició ens torna a relacionar l'aritmètica tropical amb el nostre problema.

Proposició 6. *La funció objectiu del problema de l'alineament de seqüències és la tropicalització del polinomi coordenada f_{σ^1,σ^2} del HMM parell.*

Demostració. La tropicalització del polinomi (3.34) s'obté reemplaçant el sumatori per la suma tropical $\oplus$ (en aquest cas en lloc de considerar l'operació max considerem el min) i els productoris interiors pels productes tropicals $\otimes$. Reemplacem cada θ desconeguda pels corresponents w desconeguts, que els considerarem un logaritme dels θ. El resultat d'açò és el polinomi tropical

$$\operatorname{trop}\left(f_{\sigma^1,\sigma^2}\right) = \bigoplus_{h \in \mathcal{A}_{n,m}} \bigotimes_{i=1}^{|h|} w_{\mu_i^1,\mu_i^2} \cdot \bigotimes_{i=2}^{|h|} w'_{h_{i-1},h_i} \tag{3.35}$$

El producte tropical dins la suma tropical és precisament el pes $W(h)$ de l'alineament h o $\left(\mu^1, \mu^2\right)$ tal i com s'ha definit en (3.22). Per tant, (3.35) és equivalent a

$$\operatorname{trop}\left(f_{\sigma^1,\sigma^2}\right) = \min_{h \in \mathcal{A}_{n,m}} W(h) \tag{3.36}$$

Avaluar la part de la dreta de la igualtat és, per tant, equivalent a trobar l'alineament òptim de les dues seqüències σ^1 i σ^2. □

Cal destacar que, com que el logaritme d'una probabilitat (en aquest cas) és sempre negatiu, la correspondència en la Proposició 6 només serveix per a esquemes de puntuació en què els pesos tenen el mateix signe. Els esquemes de puntuació, els pesos dels quals tenen signes barrejats (com en l'exemple 2.14) són resultat d'associar w amb els logaritmes del ràtio de les probabilitats $\log\left(\theta/\tilde{\theta}\right)$ on els $\tilde{\theta}$ són nous paràmetres addicionals.

Exemple

Considerem dues seqüències $\sigma^1 = ij$ i $\sigma^2 = klm$ de longitud $n = 2$ i $m = 3$ sobre l'alfabet $\Omega = \{A, C, G, T\}$. El nombre total de tots els alineaments amb aquestes característiques és $|\mathcal{A}_{2,3}| = 25$, i figuren en la Taula 3.2. Per exemple, l'alineament $MIID$, aquí escrit com $(i \cdot \cdot j, klm\cdot)$, correspon a $\begin{smallmatrix} \texttt{i--j} \\ \texttt{klm-} \end{smallmatrix}$ en la notació genòmica estàndard.

El polinomi f_{σ^1,σ^2} és la suma dels 25 monomis (de grau $9, 7, 5$) en la columna de la dreta. Així, el model HMM parell presentat en la Taula 3.2 no és més que una aplicació polinòmica

$$f : \mathbb{R}^{33} \to \mathbb{R}^{1024} \tag{3.37}$$

Per tal de poder passar a la inferència paramètrica, necessitem presentar els polítops.

3.5.2 Polítops

Tot polinomi i tota funció polinòmica té un polítop associat anomenat polítop de Newton (polítop convex els vèrtexs del qual tenen tots coordenades cartesianes). Açò ens permet substituir l'aritmètica tropical per l'àlgebra de polítops, que és útil a l'hora de resoldre problemes d'inferències paramètriques. A continuació presentarem els polítops de Newton procedents del HMM parell per a l'alineament de seqüències.

Exemple Si considerem

$$w' = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{3.38}$$

Considerem la següent possible especificació:

$$\theta' = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \tag{3.39}$$

Escrivim g_{σ^1,σ^2} per al polinomi en les 24 incògnites $\theta \ldots$ obtingudes de f_{σ^1,σ^2} definint cadascun de les 9 incògnites $\theta'_{\ldots}$ a 1. Si calculem el polinomi $g_{\mathrm{s1,s2}}$ per a $s1 := [A, C, G]$ i $s2 := [A, C, C]$ mitjançant l'Algoritme 1 es produeix la descomposició del polinomi $g_{\mathrm{ACG,ACC}}$:
$((\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{C,C}$
$+ (\theta_{A,\_} \cdot \theta_{A,C} + \theta_{A,\_} \cdot \theta_{C,\_} \cdot \theta_{\_,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{C,\_}) \cdot \theta_{\_,C}$
$+ (\theta_{\_,A} \cdot \theta_{C,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{\_,C} + \theta_{\_,A} \cdot \theta_{\_,C} \cdot \theta_{A,\_}) \cdot \theta_{C,\_}) \cdot \theta_{G,C}$
$+ ((\theta_{A,\_} \cdot \theta_{A,C} + \theta_{A,\_} \cdot \theta_{C,\_} \cdot \theta_{\_,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{C,\_}) \cdot \theta_{C,C}$

$+ (\theta_{A,\_} \cdot \theta_{C,\_} \cdot \theta_{A,C} + \theta_{A,\_} \cdot \theta_{C,\_}^2 \cdot \theta_{\_,A}$
$+ (\theta_{A,\_} \cdot \theta_{A,C} + \theta_{A,\_} \cdot \theta_{C,\_} \cdot \theta_{\_,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{C,\_}) \cdot \theta_{C,\_}) \cdot \theta_{\_,C}$
$+ ((\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{C,C} + (\theta_{A,\_} \cdot \theta_{A,C} + \theta_{A,\_} \cdot \theta_{C,\_} \cdot \theta_{\_,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{C,\_}) \cdot \theta_{\_,C}$
$+ (\theta_{\_,A} \cdot \theta_{C,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{\_,C} + \theta_{\_,A} \cdot \theta_{\_,C} \cdot \theta_{A,\_}) \cdot \theta_{C,\_}) \cdot \theta_{\_,G}$
$+ ((\theta_{\_,A} \cdot \theta_{C,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{\_,C} + \theta_{\_,A} \cdot \theta_{\_,C} \cdot \theta_{A,\_}) \cdot \theta_{G,C}$
$+ ((\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{C,C} + (\theta_{A,\_} \cdot \theta_{A,C} + \theta_{A,\_} \cdot \theta_{C,\_} \cdot \theta_{\_,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{C,\_}) \cdot \theta_{\_,C} +$
$(\theta_{\_,A} \cdot \theta_{C,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{\_,C} + \theta_{\_,A} \cdot \theta_{\_,C} \cdot \theta_{A,\_}) \cdot \theta_{C,\_}) \cdot \theta_{\_,G}$
$+ (\theta_{\_,A} \cdot \theta_{\_,C} \cdot \theta_{G,A} + (\theta_{\_,A} \cdot \theta_{C,A} + (\theta_{A,A} + 2 \cdot \theta_{A,\_} \cdot \theta_{\_,A}) \cdot \theta_{\_,C} + \theta_{\_,A} \cdot \theta_{\_,C} \cdot \theta_{A,\_}) \cdot \theta_{\_,G} + \theta_{\_,A} \cdot \theta_{\_,C} \cdot$
$\theta_{\_,G} \cdot \theta_{A,\_}) \cdot \theta_{C,\_}) \cdot \theta_{C,\_}$

Notem que els paràmetres corresponents al nucleòtid T no apareixen perquè no hi ha cap observació d'aquest en cap de les dues cadenes. L'expansió d'aquest polinomi té 14 monomis i la suma dels seus coeficients és $\#\mathcal{A}_{3,3} = 63$.

Llavors, podem considerar els 14 punts següents v_i en l'espai 11-dimensional:

$$
\begin{array}{rc}
v_1 = (0,0,1,0,0,2,0,0,1,1,1) & 20\theta_{\mathrm{A}-}\theta_{-\mathrm{A}}\theta_{\mathrm{C}-}^2\theta_{-\mathrm{C}}\theta_{-\mathrm{G}} \\
v_2 = (1,0,0,0,0,2,0,0,0,1,1) & 6\theta_{\mathrm{AA}}\theta_{\mathrm{C}-}^2\theta_{-\mathrm{C}}\theta_{-\mathrm{G}} \\
v_3 = (0,0,1,0,1,1,0,0,1,0,1) & 7\theta_{\mathrm{A}-}\theta_{-\mathrm{A}}\theta_{\mathrm{CC}}\theta_{\mathrm{C}-}\theta_{-\mathrm{G}} \\
v_4 = (0,0,1,0,0,1,0,1,1,1,0) & 9\theta_{\mathrm{A}-}\theta_{-\mathrm{A}}\theta_{\mathrm{C}} - \theta_{-\mathrm{C}}\theta_{\mathrm{GC}} \\
v_5 = (0,1,1,0,0,1,0,0,0,1,1) & 4\theta_{\mathrm{A}-}\theta_{\mathrm{AC}}\theta_{\mathrm{C}-}\theta_{-\mathrm{C}}\theta_{-\mathrm{G}} \\
v_6 = (0,0,0,0,0,2,1,0,1,1,0) & \theta_{-\mathrm{A}}\theta_{\mathrm{C}-}^2\theta_{-\mathrm{C}}\theta_{\mathrm{GA}} \\
v_7 = (0,0,0,1,0,2,0,0,1,0,1) & 3\theta_{-\mathrm{A}}\theta_{\mathrm{C}-}^2\theta_{\mathrm{CA}}\theta_{-\mathrm{G}} \\
v_8 = (1,0,0,0,1,1,0,0,0,0,1) & 3\theta_{\mathrm{AA}}\theta_{\mathrm{CC}}\theta_{\mathrm{C}-}\theta_{-\mathrm{G}} \\
v_9 = (1,0,0,0,0,1,0,1,0,1,0) & 3\theta_{\mathrm{AA}}\theta_{\mathrm{C}-}\theta_{-\mathrm{C}}\theta_{\mathrm{GC}} \\
v_{10} = (0,0,1,0,1,0,0,1,1,0,0) & 2\theta_{\mathrm{A}-}\theta_{-\mathrm{A}}\theta_{\mathrm{CC}}\theta_{\mathrm{GC}} \\
v_{11} = (0,1,1,0,1,0,0,0,0,0,1) & \theta_{\mathrm{A}-}\theta_{\mathrm{CC}}\theta_{\mathrm{AC}}\theta_{-\mathrm{G}} \\
v_{12} = (0,1,1,0,0,0,0,1,0,1,0) & \theta_{\mathrm{A}-}\theta_{\mathrm{AC}}\theta_{-\mathrm{C}}\theta_{\mathrm{GC}} \\
v_{13} = (0,0,0,1,0,1,0,1,1,0,0) & 2\theta_{-\mathrm{A}}\theta_{\mathrm{C}} - \theta_{\mathrm{CA}}\theta_{\mathrm{GC}} \\
v_{14} = (1,0,0,0,1,0,0,1,0,0,0) & \theta_{\mathrm{AA}}\theta_{\mathrm{CC}}\theta_{\mathrm{GC}}
\end{array}
\tag{3.40}
$$

A la dreta de cada punt v_i està el monomi corresponent obtingut mitjançant l'Algoritme 1 en el conjunt de paràmetres desconeguts $(\theta_{\mathrm{AA}}, \theta_{\mathrm{AC}}, \theta_{\mathrm{A}-}, \theta_{\mathrm{CA}}, \theta_{\mathrm{CC}}, \theta_{\mathrm{C}-}, \theta_{\mathrm{GA}}, \theta_{\mathrm{GC}}, \theta_{-\mathrm{A}}, \theta_{-\mathrm{C}}, \theta_{-\mathrm{G}})$. La j-èssima coordenada en v_i és igual a l'exponent del paràmetre j-èssim desconegut.

Els 14 punts v_i abracen un espai lineal hexadimensional en $\mathbb{R}^{11}$, i és la seua localització en aquest espai la que determina quin alineament és l'òptim. Per exemple, l'alineament sense buits (M, M, M) que es correspon a l'últim monomi $\theta_{\mathrm{AA}}\theta_{\mathrm{CC}}\theta_{\mathrm{GC}}$ és òptim sii l'esquema de puntuació

w satisfà:

$$w_{\mathrm{C}-} + w_{-\mathrm{G}} \geq w_{\mathrm{GC}}, \quad w_{\mathrm{A}-} + w_{\mathrm{AC}} + w_{-\mathrm{G}} \geq w_{\mathrm{AA}} + w_{\mathrm{GC}} \tag{3.41}$$

$$w_{\mathrm{C}-} + w_{-\mathrm{C}} \geq w_{\mathrm{CC}}, \quad w_{\mathrm{A}-} + w_{\mathrm{AC}} + w_{-\mathrm{C}} \geq w_{\mathrm{AA}} + w_{\mathrm{CC}} \tag{3.42}$$

$$w_{\mathrm{A}-} + w_{-\mathrm{A}} \geq w_{\mathrm{AA}}, \quad w_{-\mathrm{A}} + w_{\mathrm{C}-} + w_{\mathrm{CA}} \geq w_{\mathrm{AA}} + w_{\mathrm{CC}} \tag{3.43}$$

$$\text{i } w_{-\mathrm{A}} + 2w_{\mathrm{C}-} + w_{-\mathrm{C}} + w_{\mathrm{GA}} \geq w_{\mathrm{AA}} + w_{\mathrm{CC}} + w_{\mathrm{GC}} \tag{3.44}$$

L'objectiu d'aquesta secció és el d'introduir la geometria que hi ha darrere aquestes derivacions. Donats n punts qualsevols $v_1, \ldots, v_n$ en $\mathbb{R}^d$, la seua envolupant convexa (intersecció de tots els conjunts convexos que contenen els punts donats) és el conjunt

$$P = \left\{ \sum_{i=1}^{n} \lambda_i v_i \in \mathbb{R}^d : \lambda_1, \ldots, \lambda_n \geq 0 \text{ i } \sum_{i=1}^{n} \lambda_i = 1 \right\} \tag{3.45}$$

Qualsevol subconjunt de $\mathbb{R}^d$ d'aquesta forma s'anomena polítop convex (o senzillament polítop). La dimensió del polítop P és la dimensió de l'espai vectorial afí generat $\left\{ \sum_{i=1}^{b} \lambda_i v_i \in \mathbb{R}^d : \sum_{i=1}^{n} \lambda_i = 1 \right\}$. També podem representar un polítop com una intersecció finita de semiespais (qualsevol de les dues parts en les quals un hiperplà divideix un espai afí) tancats. Siga A una matriu real de dimensió $d \times m$ i siga $b \in \mathbb{R}^m$. Cada fila d'A i les corresponents entrades de d defineix un semiespai en $\mathbb{R}^d$. La seua intersecció és el següent conjunt que pot ser fitat o no:

$$P = \left\{ x \in \mathbb{R}^d : A \cdot x \geq b \right\} \tag{3.46}$$

Qualsevol subconjunt de $\mathbb{R}^d$ d'aquesta forma s'anomena poliedre convex.

Teorema 3. *(Teorema de Weyl-Minkowski) Els polítops convexos són precisament els poliedres convexos fitats.*

La demostració d'aquest teorema es pot trobar als llibres [11] i [26].

Així, tot polítop es pot representar de la forma (3.45) o de la forma (3.46). Aquestes representacions es coneixen com a V-polítops i H-polítops. Una tasca algorítmica fonamental en la geometria és transformar-ne un en l'altre.

Exemple 2.20 Siga P el cub estàndard de dimensió $d = 3$. Com a un H-polítop, el cub és la solució a $m = 6$ desigualtats lineals:

$$P = \left\{ (x, y, z) \in \mathbb{R}^3 : 0 \leq x \leq 1, 0 \leq y \leq 1, 0 \leq z \leq 1 \right\}$$

i com a un V-polítop el cub és l'envolupant convexa de $n = 8$ punts
$P = \mathrm{conv}\{(0,0,0), (0,0,1), (0,1,0), (0,1,1), (1,0,0), (1,0,1), (1,1,0), (1,1,1)\}$.

Altres tasques d'aquesta branca són realitzar la V-representació (3.45) irredundant esborrant punts v_i, i fer H-representació (3.46) irredundant eliminant semiespais, ambdues tasques conservant el conjunt P intacte. Per tal d'entendre la geometria que hi ha subjacent, però, necessitem definir les cares dels polítops.

Donat un polítop $P \subset \mathbb{R}^d$ i un vector $w \in \mathbb{R}^d$, considerem el conjunt de tots els punts de P en els quals la funció lineal $x \mapsto x \cdot w$ assoleix el seu mínim. Es denota per

$$\text{face}_w(P) = \{x \in P : x \cdot w \leq y \cdot w \ \ \forall y \in P\} \tag{3.47}$$

Siga $w^* = \min\{x \cdot w : x \in P\}$. Aleshores podem escriure (3.47) de manera equivalent com

$$\text{face}_w(P) = \{x \in P : x \cdot w \leq w^*\} \tag{3.48}$$

Açò mostra que $\text{face}_w(P)$ és un poliedre fitat, i pel Teorema 3 és un polítop. Tot polítop d'aquesta forma s'anomena una cara de P. En particular, P és una cara de si mateixa, obtinguda si prenem $w = 0$. Una cara de dimensió 0 consisteix en un únic punt i s'anomena vèrtex de P. Una cara de dimesió 1 s'anomena aresta, una cara de dimensió $\dim(P) - 1$ s'anomena faceta, i una cara de dimensió $\dim(P) - 2$ s'anomena cresta. L'exemple del cub té 27 cares: 8 vèrtexs, 12 arestes, 6 crestes i el mateix cub.

Escrivim $f_i(P)$ per al nombre de cares i-dimensionals d'un polítop P. El vector $f(P) = (f_0(P), f_1(P), f_2(P), \ldots, f_{d-1}(P))$ s'anomena el f-vector de P. Així doncs, el cub tridimensional P té el f-vector $f(P) = (8, 12, 6)$. El seu polítop dual (els vèrtexs d'un polítop corresponen a les cares del seu polítop dual i les arestes del mateix polítop corresponen a les arestes del seu polítop dual) P^*, que és un octaedre, té el f-vector $f(P^*) = (6, 12, 8)$. Siga P un polítop i F una cara de P. Definim el con normal de P en F com

$$N_P(F) \quad = \quad \left\{w \in \mathbb{R}^d : \ \text{face}_{\ w}(P) = F\right\} \tag{3.49}$$

La seua dimensió compleix que

$$\dim N_P(F) = d - \dim(F) \tag{3.50}$$

En particular, si $F = \{v\}$ és un vèrtex de P, aleshores el seu con normal $N_P(v)$ és d-dimensional i està compost per totes les funcions lineals w que assoleixen el seu mínim en v.

Exemple Siga P l'envolupant convexa dels punts $v_1, \ldots, v_{14}$ de l'exemple anterior. El con normal $N_P(v_{14})$ consisteix en tots els pesos per als quals l'alineament sense buits (M, M, M) és òptim.

La col·lecció de tots els cons $N_P(F)$ a mesura que F es mou per totes les cares de P es denota per $\mathcal{N}(P)$ i s'anomena ventall normal de P. Així, el ventall normal $\mathcal{N}(P)$ és una partició de $\mathbb{R}^d$ en cons. A més a més, existeix una bijecció entre els cons en $\mathcal{N}(P)$ i les cares de P.

El següent resultat lliga les cares d'un polítop P amb les seues representacions irredundants. Siga a_i un vector fila de la matriu A en (3.46) i siga b_i la corresponent entrada del vector b. Açò defineix la cara

$$\text{face}_{a_i}(P) = \{x \in P : a_i \cdot x = b_i\} \tag{3.51}$$

Proposició 7. *La V-representació (3.45) del polítop P és irredundant sii v_i és un vèrtex de P per a $i = 1, \ldots, n$.*
La H-representació (3.46) és irredundant sii la cara $\text{face}_{a_i}(P)$ *és una faceta de P per a $i = 1, \ldots, m$.*

Siga $\mathcal{P}_d$ el conjunt de tots els polítops en $\mathbb{R}^d$. Hi ha dues operacions naturals (l'addició $\oplus$ i la multiplicació $\otimes$) definides al conjunt $\mathcal{P}_d$. L'estructura resultant és l'algebra polítop $(\mathcal{P}_d, \oplus, \otimes)$. En particular, si $P, Q \in \mathcal{P}_d$ són polítops, la seua suma $P \oplus Q$ és l'envolupant convexa de la unió de P i Q :

$$\begin{aligned} P \oplus Q &:= \text{conv}(P \cup Q) && (3.52)\\ &= \Big\{\lambda p + (1-\lambda)q \in \mathbb{R}^d : p \in P, q \in Q, 0 \leq \lambda \leq 1\Big\} && (3.53)\end{aligned}$$

El producte en l'àlgebra del polítop està definit com la suma de Minkowski:

$$\begin{aligned} P \otimes Q &:= \quad P + Q && (3.54)\\ &= \Big\{p + q \in \mathbb{R}^d : p \in P, q \in Q\Big\} && (3.55)\end{aligned}$$

Del Teorema 3 es dedueix que tant $P \oplus Q$ com $P \otimes Q$ són polítops en $\mathbb{R}^d$. L'àlgebra polítop $(\mathcal{P}_d, \oplus, \otimes)$ satisfà molts axiomes de la família de l'aritmètica. No només la suma i la multiplicació són commutatives, sinó que també es compleix la propietat distributiva:

Proposició 8. *Si P, Q, R són polítops en $\mathbb{R}^d$, aleshores*

$$(P \oplus Q) \otimes R \quad = (P \otimes R) \oplus (Q \otimes R) \tag{3.56}$$

Demostració. Considerem els punts $p \in P, q \in Q$ i $r \in R$. Per a $0 \leq \lambda \leq 1$ es compleix que

$$(\lambda p + (1-\lambda)q) + r = \lambda(p + r) + (1-\lambda)(q + r)$$

La part de l'esquerra representa un punt arbitrari en la part de l'esquerra de l'equació (3.56) i la part de la dreta representa un punt arbitrari en la part de la dreta de l'equació (3.56) . □

Exemple (Revisió del semianell tropical) Considerem l'àlgebra $(\mathcal{P}_1, \oplus, \otimes)$ de tots els polítops en la línia real ($d = 1$). Cada element de $\mathcal{P}_1$ és un segment $[a, b]$ on $a < b$ són nombres reals. Les operacions aritmètiques són

$$[a, b] \oplus [c, d] = [\min(a, c), \max(b, d)], \tag{3.57}$$

$$[a, b] \otimes [c, d] = [a + c, b + d] \tag{3.58}$$

Així, l'àlgebra polítop unidimensional és essencialment el mateix que el semianell tropical $(\mathbb{R}, \oplus, \otimes)$.

Una de les principals connexions entre polítops i estadística algebraica són els polítops de Newton dels polinomis que parametritzen un model. Cosiderem el polinomi

$$f = \sum_{i=1}^{n} c_i \cdot \theta_1^{v_{i1}} \theta_2^{v_{i2}} \cdots \theta_d^{v_{id}} \tag{3.59}$$

on c_i és un nombre real diferent de zero i $v_i = (v_{i1}, v_{i2}, \ldots, v_{id}) \in \mathbb{N}^d$ per a $i = 1, 2, \ldots, n$. Definim el polítop de Newton del polinomi f com l'envolupant convexa de tots els vectors exponents que apareixen en l'expansió (3.59) de f:

$$\mathrm{NP}(f) := \mathrm{conv}\,\{v_1, v_2, \ldots, v_n\} \quad \subset \quad \mathbb{R}^d \tag{3.60}$$

Per tant, el polítop de Newton $\mathrm{NP}(f)$ és el V-polítop de (3.45). L'operació de calcular els polítops de Newton respecta les operacions aritmètiques del Teorema 4.

Teorema 4. *Siguen f i g polinomis en $\mathbb{R}[\theta_1, \ldots, \theta_d]$. Aleshores*

$$\mathrm{NP}(f \cdot g) = \mathrm{NP}(f) \otimes \mathrm{NP}(g) \quad i \quad \mathrm{NP}(f + g) \subseteq \mathrm{NP}(f) \oplus \mathrm{NP}(g) \tag{3.61}$$

Si tots els coeficients de f i g són positius, aleshores $\mathrm{NP}(f + g) = \mathrm{NP}(f) \oplus \mathrm{NP}(g)$.

Exemple 2.26 Considerem els polinomis $f = (x + 1)(y + 1)(z + 1)$ i $g = (x + y + z)^2$. Aleshores $\mathrm{NP}(f)$ és un cub i $\mathrm{NP}(g)$ és un triangle.
El polítop de Newton $\mathrm{NP}(f+g)$ de la seua suma és una bipiràmide amb vèrtexs $(0, 0, 0), (2, 0, 0), (0, 2, 0), (0, 0, 2), (1, 1, 1)$. El polítop de Newton $\mathrm{NP}(f \cdot g)$ del seu producte és la suma de Minkowski del cub més el triangle.

Els polítops de Newton ens permeten transformar construccions del conjunt de l'àlgebra de polinomis al conjunt de la geometria de polítops. Per tal d'il·lustrar açò, mostrarem el següent exemple.

Suposem que tenim una matriu de polinomis de dimensió 4×4

$$A(x, y, z) = \begin{pmatrix} a_{11}(x, y, z) & a_{12}(x, y, z) & a_{13}(x, y, z) & a_{14}(x, y, z) \\ a_{21}(x, y, z) & a_{22}(x, y, z) & a_{23}(x, y, z) & a_{24}(x, y, z) \\ a_{31}(x, y, z) & a_{32}(x, y, z) & a_{33}(x, y, z) & a_{34}(x, y, z) \\ a_{41}(x, y, z) & a_{42}(x, y, z) & a_{43}(x, y, z) & a_{44}(x, y, z) \end{pmatrix}$$

i suposem que estem interessats en el polítop de Newton del seu determinant $\det(A(x,y,z))$. Una possible manera de calcular el polítop de Newton és avaluar-ne el determinant, enumerar tots els termes que hi ha en el polinomi, i després calcular l'envolupant convexa. Tanmateix, assumint que els coeficients de $a_{ij}(x,y,z)$ no es cancel·len, és més eficient fer l'aritmètica als polítops de Newton. De manera que substituïm cada entrada de la matriu pel seu polítop de Newton $P_{ij} = \mathrm{NP}(a_{ij})$, considerem la matriu 4×4 de polítops (P_{ij}), i calculem el seu determinant en l'àlgebra polítop. Igual que si ho férem en el semianell tropical, el determinant no canvia:

$$\det \begin{pmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P_{41} & P_{42} & P_{43} & P_{44} \end{pmatrix} = \bigoplus_{\sigma \in S_4} P_{1\sigma(1)} \otimes P_{2\sigma(2)} \otimes P_{3\sigma(3)} \otimes P_{4\sigma(4)} \tag{3.62}$$

Aquest determinant de polítops representa una família de problemes d'alineament parametritzats.

El problema de l'assignació paramètrica es resol calculant el determinant de la matriu dels polítops (P_{ij}) en l'àlgebra del polítop. Seguidament, explicarem com es fan inferències estadístiques en el model HMM per a parelles.

3.5.3 Inferències estadístiques

Donats els paràmetres θ i θ', volem determinar l'alineament $\widehat{i} \in \mathcal{A}_{n,m}$ que indexa el terme amb el major valor numèric entre els molts termes (vegeu Proposició 5) del polinomi f_{σ_1,σ_2} (3.34). Si hi ha més d'un alineament amb aquest valor màxim es desempata amb l'ordre lexicogràfic. Anomenem $\widehat{i}$ a l'explicació de l'observació $\left(\sigma^1, \sigma^2\right)$.

L'explicació de la parella de seqüències d'ADN es pot calcular en temps polinòmic (en les seues longituds n i m). De la mateixa manera que en l'anterior secció, la idea principal està en tropicalitzar les coordenades del polinomi (3.34) del model estadístic en qüestió. Concretament, es calcula

$$\max_{\mathbf{h} \in \mathcal{A}_{n,m}} \theta_{\mu_1^1,\mu_1^2} + \sum_{i=2}^{|\mathbf{h}|} \theta'_{\mu_{i-1}^1,\mu_{i-1}^2} + \theta_{\mu_i^1,\mu_i^2} \tag{3.63}$$

o equivalentment, (3.36) si tenim els pesos.

L'argument que maximitza aquesta funció lineal a trossos convexa és l'alineament òptim $\widehat{i}$. Inferir en HMM parell vol dir calcular l'alineament òptim de les dues seqüències d'ADN observades. Dit d'una altra manera, inferir vol dir avaluar la funció d'alineament

$$\Omega^n \times \Omega^m \to \mathcal{A}_{n,m}, \quad \left(\sigma^1, \sigma^2\right) \mapsto \widehat{\mathbf{i}} \tag{3.64}$$

La quantitat de funcions d'aquestes és doble exponencial (exponencial en n i exponencial en m) i varia de $\Omega^n \times \Omega^m$ a $\mathcal{A}_{n,m}$. Pel teorema d'Elizalde (1), però, com a molt una quantitat polinòmica d'aquestes són funcions d'alineament. Per sort o per desgràcia, de la mateixa manera que per a les funcions de predicció de gens, caracteritzar les funcions d'alineament és un problema obert.

La funció $\mathbb{R}^{33} \to \mathbb{R}$ donada en (3.63) és la funció de suport del polítop convex en $\mathbb{R}^{33}$, concretament és el polítop de Newton del polinomi f_{σ^1,σ^2}. Els vèrtexs d'aquest polítop corresponen a tots els alineaments òptims de les seqüències σ^1, σ^2 respecte de tots els possibles valors triats per als paràmetres, i el ventall normal del polítop divideix l'espai paramètric logarítmic en regions que donen el mateix alineament òptim. Açò es pot usar per tal d'analitzar la susceptibilitat dels alineaments als paràmetres, i per al posterior càlcul de les probabilitats dels alineaments òptims. El procés de calcular aquest polítop es coneix com alineament paramètric o inferència paramètrica. Per [18], es sap que aquesta inferència paramètrica es pot fer en temps polinòmic (en m i n).

Una nota important és que la formulació de l'alineament de seqüències amb models HMM parells és equivalent als esquemes de puntuació combinatoris o a les distàncies d'edició generalitzades que es poden emprar per tal d'assignar pesos als alineaments [3]. L'esquema de puntuació més senzill consta de dos paràmetres: un paràmetre per a les mutacions mis (de l'anglés 'mismatch'), i un paràmetre per a les insercions o eliminacions gap [12]. El pes d'un alineament és la suma de les puntuacions mis i gap per a totes les posicions de l'alineament on, si coincideix, es puntua com a 1. Si mis i gap són no negatius, açò és equivalent a especialitzar els 33 paràmetres logarítmics $\theta'.. = \log(\theta'..)$ i $\theta.. = \log(\theta..)$ del HMM parell de la següent manera:

$$\theta'_{ij} = 0, \quad \theta_{Ij} = \theta_{Di} = -gap \quad \forall\ i, j \tag{3.65}$$

$$\theta_{Mij} = -1 \quad \text{si } i = j, \quad \text{i} \quad \theta_{Mij} = -mis \quad \text{si } i \neq j \tag{3.66}$$

En el cas que l'esquema de puntuació consisteixi en paràmetres positius i negatius, es correspon amb un model HMM parell normalitzat [17]. Aquesta especialització dels paràmetres es correspon a projectar el polítop de Newton de f_{σ^1,σ^2} en dos dimensions. L'alineament paramètric vol dir parametritzar el polígon bidimensional resultant. Per a dues seqüències de longitud n, una fita superior del nombre de vèrtexs en el polígon és $O\left(n^{2/3}\right)$. S'ha observat que per a seqüències biològiques, el nombre pot ser molt més petit.

En un sentit estrictament tècnic, la formulació del polinomi (3.34) no té per què derivar o analitzar algoritmes combinatoris per a l'alineament de seqüències. No obstant això, la traducció de la geometria algebraica (3.34) a l'optimització discreta (3.63) ofereix molt. En [14] es planteja que la geometria algebraica (tropical) és un marc conceptual per a desenvolupar nous models i dissenyar nous algoritmes pràctics per a la filogenòmica.

3.6 Models evolutius

Un cop explicats els models d'alineament per a dues seqüències d'ADN, expliquem ara els models filogenètics per a més de dues seqüències d'ADN.

Com que els organismes de diferents espècies no poden produir una cria junts, les mutacions i els canvis al genoma que es produeixen dins d'una mateixa espècie són independents dels que ocorren en una altra espècie. Tot i que hi ha alguna excepció al món dels bacteris, en aquest text les ignorarem. Així, podem representar l'evolució de les espècies amb una estructura d'arbre. L'estudi de les estructures d'arbre en l'evolució del genoma es coneix com filogenètica. Un arbre filogenètic en X és un arbre T amb tots els seus vèrtexs de com a mínim grau 3, i amb les fulles etiquetades per un conjunt X que correspon a diferents espècies. En aquesta secció, assumirem que T és conegut i que els vèrtexs en T corresponen a esdeveniments d'especiació coneguts. Comencem descrivint els models estadístics d'evolució que s'usen per a identificar regions entre genomes que estan sota selecció.

Els models evolutius intenten capturar 3 aspectes importants de l'evolució de seqüències: longitud de la branca, substitució, i mutació. Considerem un únic ancestre base b a l'arrel r d'un arbre filogenètic T, i assumim que no hi ha insercions ni eliminacions en el temps. Com que l'ancestre base canvia, és possible que en dues fulles $x, y \in X$ observem les bases $c_1 \neq c_2$ diferents. I diem que ha ocorregut una substitució entre x i y. En un model probabilístic evolutiu, ens agradaria calcular la possibilitat d'un canvi en les arestes internes d'un arbre, amb la possibilitat de substitucions. Per exemple, és possible que $b \to c_1 \to b \to c_1$ al llarg del camí de r a x. Per tal de calcular açò, introduïm les matrius de velocitat de transició.

Definició 6. *Una matriu de velocitat de transició és una matriu quadrada $Q = (q_{ij})_{i,j\in\Omega}$ (amb les columnes i les files indexades pels nucleòtids) que compleix les següents propietats:*

$$\begin{aligned} q_{ij} &\geq 0 \quad \textit{per a} \quad i \neq j, \\ \sum_{j\in\Omega} q_{ij} &= 0 \quad \textit{per a tot } i \in \Omega, \\ q_{ii} &< 0 \quad \textit{per a tot } i \in \Omega. \end{aligned}$$

Les matrius de velocitat de transició ens proporcionen una mesura de la velocitat instantània de mutació. Arran d'una matriu de velocitat de transició Q es poden calcular les matrius de substitució $P(t)$ mitjançant l'exponenciació de matrius. L'entrada de $p(t)$ en la fila b i la columna c és igual a la probabilitat que la substitució $b \to \cdots \to c$ ocórrega en un interval de temps de longitud t. Recordem el següent resultat conegut sobre models de Màrkov en temps continu.

Proposició 9. *Siga Q una matriu de velocitat de transició qualsevol i $P(t) = e^{Qt} = \sum_{i=0}^{\infty} \frac{1}{i!} Q^i t^i$. Aleshores*

1. $P(s+t) = P(s) + P(t)$
2. $P(t)$ *és l'única solució de* $P'(t) = P(t) \cdot Q, P(0) = 1$ *per a* $t \geq 0$
3. $P(t)$ *és l'única solució de* $P'(t) = Q \cdot P(t), P(0) = 1$ *per a* $t \geq 0$.

A més a més, una matriu Q és una matriu de velocitat de transició sii la matriu $P(t) = e^{Qt}$ és una matriu estocàstica (no negativa amb la suma total de cada fila igual a 1) per a tot t.

El model evolutiu més simple (i l'únic que veurem en aquest text) és el model d'ADN de Jukes-Cantor, la matriu de velocitat de transició del qual és

$$Q = \begin{pmatrix} -3\alpha & \alpha & \alpha & \alpha \\ \alpha & -3\alpha & \alpha & \alpha \\ \alpha & \alpha & -3\alpha & \alpha \\ \alpha & \alpha & \alpha & -3\alpha \end{pmatrix} \tag{3.67}$$

on $\alpha \geq 0$ és un paràmetre. La seua matriu de substitució corresponent és

$$P(t) = \frac{1}{4} \begin{pmatrix} 1+3e^{-4\alpha t} & 1-e^{-4\alpha t} & 1-e^{-4\alpha t} & 1-e^{-4\alpha t} \\ 1-e^{-4\alpha t} & 1+3e^{-4\alpha t} & 1-e^{-4\alpha t} & 1-e^{-4\alpha t} \\ 1-e^{-4\alpha t} & 1-e^{-4\alpha t} & 1+3e^{-4\alpha t} & 1-e^{-4\alpha t} \\ 1-e^{-4\alpha t} & 1-e^{-4\alpha t} & 1-e^{-4\alpha t} & 1+3e^{-4\alpha t} \end{pmatrix} \tag{3.68}$$

El nombre esperat de substitucions al llarg del temps t és la quantitat

$$3\alpha t = -\frac{1}{4} \cdot \operatorname{tr}(Q) \cdot t = -\frac{1}{4} \cdot \log\det(P(t)) \tag{3.69}$$

Aquest nombre s'anomena la longitud de branca. Es pot calcular de la matriu de substitució $P(t)$ i s'usa per a assignar pesos a les arestes de l'arbre filogenètic T.

Una manera d'especificar un model evolutiu és donar un arbre filogenètic T junt amb una matriu de velocitat de transició Q i una distribució inicial per a l'arrel de T (que aquí assumim que és la distribució estacionària en Ω). Les longituds de les branques de les arestes són paràmetres desconeguts, i l'objectiu és estimar-les arran de les dades. Així, si l'arbre T té r arestes, aleshores un model evolutiu té r paràmetres lliures, i, ens agradaria considerar-lo com una varietat algebraica r-dimensional.

Tal representació algebraica de fet existeix [19]. L'explicarem per al model de Jukes-Cantor en un arbre T. Suposem que T té r arestes i $|X| = n$ fulles. Siga $P_i(t)$ la matriu de substitució associada a l'aresta i-èssima de l'arbre. Escrivim $3\alpha_i t_i = -\frac{1}{4}\log\det\left(P_i(t)\right)$ per a la longitud de la branca de l'aresta i-èssima, i fixem $\pi_i = \frac{1}{4}\left(1 - e^{-4\alpha_i t_i}\right)$ i $\theta_i = 1 - 3\pi_i$. Així,

$$P_i(t) = \begin{pmatrix} \theta_i & \pi_i & \pi_i & \pi_i \\ \pi_i & \theta_i & \pi_i & \pi_i \\ \pi_i & \pi_i & \theta_i & \pi_i \\ \pi_i & \pi_i & \pi_i & \theta_i \end{pmatrix} \tag{3.70}$$

En filogenòmica limitarem l'atenció al segment real especificat per $\theta_i \geq 0, \pi_i \geq 0$, i $\theta_i + 3\pi_i = 1$.

Siga Δ_{4^n-1} el conjunt de totes les distribucions de probabilitat en Ω^n. Com que Ω^n té 4^n elements (totes les seqüències d'ADN de longitud n), el conjunt Δ_{4^n-1} és un símplex de dimensió $4^n - 1$. Identifiquem la fulla j-èssima del nostre arbre amb la coordenada j-èssima de la seqüència d'ADN $(u_1, \ldots, u_n) \in \Omega^n$, i introduïm $p_{u_1 u_2 \cdots u_n}$ per tal de representar la probabilitat (desconeguda) d'observar els nucleòtids $u_1, u_2, \ldots, u_n$ als nivells $1, 2, \ldots, n$. Les 4^n quantitats $p_{u_1 u_2 \cdots u_n}$ són les funcions coordenades en el símplex Δ_{4^n-1}, o en el conjunt de la geometria algebraica, l'espai projectiu $\mathbb{P}^{4^n-1}$ obtingut passant a complex Δ_{4^n-1}.

Proposició 10. *En el model de Jukes-Cantor en un arbre T amb r arestes, la probabilitat $p_{u_1 u_2 \cdots u_n}$ d'observar $(u_1, u_2, \ldots, u_n) \in \Omega^n$ a les fulles s'expressa com un polinomi que és multilineal de grau r en el model de paràmetres $(\theta_1, \pi_1), (\theta_2, \pi_2), \ldots, (\theta_n, \pi_n)$. Equivalentment en termes geomètrics, el model Jukes-Cantor en T és la imatge de la funció multilinear*

$$f : \left(\mathbb{P}^1\right)^r \longrightarrow \mathbb{P}^{4^n-1} \tag{3.71}$$

Les coordenades del mapa f es deriven fàcilment de la suposició que els processos de substitució al llarg de diferents arestes de T són independents. Resulta que les 4^n coordenades de f no són totes diferents. Veurem açò amb un exemple d'un arbre de tres fulles.

Exemple 3.6 Siga $n = r = 3$, i siga T l'arbre amb tres fulles, etiquetades per $X = \{1, 2, 3\}$ que surten directament de l'arrel de T. Considerem el model de Jukes-Cantor per a ADN amb una distribució uniforme de l'arrel en T. Aquest model és una varietat tridimensional, donada com a la imatge d'una funció trilineal

$$f : \mathbb{P}^1 \times \mathbb{P}^1 \times \mathbb{P}^1 \to \mathbb{P}^{63} \tag{3.72}$$

El nombre d'estats en Ω^3 és $4^3 = 64$ però només hi ha 5 polinomis diferents que ocorren en les coordenades de la funció f. Siga p_{123} la probabilitat d'observar la mateixa lletra a totes les tres lletres, p_{ij} la probabilitat d'observar la mateixa lletra a les fulles i, j, i una diferent en la 3a fulla, i p_{dis} la probabilitat d'observar 3 lletres diferents. Aleshores

$$p_{123} = \theta_1\theta_2\theta_3 + 3\pi_1\pi_2\pi_3 \tag{3.73}$$
$$p_{dis} = 6\theta_1\pi_2\pi_3 + 6\pi_1\theta_2\pi_3 + 6\pi_1\pi_2\theta_3 + 6\pi_1\pi_2\pi_3 \tag{3.74}$$
$$p_{12} = 3\theta_1\theta_2\pi_3 + 3\pi_1\pi_2\theta_3 + 6\pi_1\pi_2\pi_3 \tag{3.75}$$
$$p_{13} = 3\theta_1\pi_2\theta_3 + 3\pi_1\theta_2\pi_3 + 6\pi_1\pi_2\pi_3 \tag{3.76}$$
$$p_{23} = 3\pi_1\theta_2\theta_3 + 3\theta_1\pi_2\pi_3 + 6\pi_1\pi_2\pi_3 \tag{3.77}$$

Totes les 64 coordenades de f venen donades per aquests 5 polinomis trilineals:

$$p_{AAA} = p_{CCC} = p_{GGG} = p_{TTT} = \frac{1}{4} \cdot p_{123} \tag{3.78}$$

$$p_{ACG} = p_{ACT} = \cdots = p_{GTC} = \frac{1}{24} \cdot p_{dis} \tag{3.79}$$

$$p_{AAC} = p_{AAT} = \cdots = p_{TTG} = \frac{1}{12} \cdot p_{12} \tag{3.80}$$

$$p_{ACA} = p_{ATA} = \cdots = p_{TGT} = \frac{1}{12} \cdot p_{13} \tag{3.81}$$

$$p_{CAA} = p_{TAA} = \cdots = p_{GTT} = \frac{1}{12} \cdot p_{23} \tag{3.82}$$

Açò vol dir que el nostre model de Jukes-Cantor és la imatge de la funció simplificada

$$f' : \mathbb{P}^1 \times \mathbb{P}^1 \times \mathbb{P}^1 \to \mathbb{P}^4, ((\theta_1, \pi_1), (\theta_2, \pi_2), (\theta_3, \pi_3)) \mapsto (p_{123}, p_{dis}, p_{12}, p_{13}, p_{23}) \tag{3.83}$$

Per tal de caracteritzar la imatge de f' algebraicament, realitzem el següent canvi lineal de coordenades:

$$q_{111} = p_{123} + \frac{1}{3}p_{dis} - \frac{1}{3}p_{12} - \frac{1}{3}p_{13} - \frac{1}{3}p_{23} = (\theta_1 - \pi_1)(\theta_2 - \pi_2)(\theta_3 - \pi_3) \tag{3.84}$$

$$q_{110} = p_{123} - \frac{1}{3}p_{dis} + p_{12} - \frac{1}{3}p_{13} - \frac{1}{3}p_{23} = (\theta_1 - \pi_1)(\theta_2 - \pi_2)(\theta_3 + 3\pi_3) \tag{3.85}$$

$$q_{101} = p_{123} - \frac{1}{3}p_{dis} - \frac{1}{3}p_{12} + p_{13} - \frac{1}{3}p_{23} = (\theta_1 - \pi_1)(\theta_2 + 3\pi_2)(\theta_3 - \pi_3) \tag{3.86}$$

$$q_{011} = p_{123} - \frac{1}{3}p_{dis} - \frac{1}{3}p_{12} - \frac{1}{3}p_{13} + p_{23} = (\theta_1 + 3\pi_1)(\theta_2 - \pi_2)(\theta_3 - \pi_3) \tag{3.87}$$

$$q_{000} = p_{123} + p_{dis} + p_{12} + p_{13} + p_{23} = (\theta_1 + 3\pi_1)(\theta_2 + 3\pi_2)(\theta_3 + 3\pi_3) \tag{3.88}$$

Açò revela que el nostre model és la hipersuperfície en $\mathbb{P}^4$ Si fixem $\theta_i = 1 - 3\pi_i$, aleshores obtenim la restricció addicional $q_{000} = 1$. La construcció en aquest exemple es pot generalitzar a qualsevol arbre T. Existeix un canvi de coordenades, simultàniament en l'espai de paràmetres $\left(\mathbb{P}^1\right)^r$ i en l'espai de probabilitat $\mathbb{P}^{4^n-1}$, tal que la funció f en (3.71) és una funció monomial en les noves coordenades. Aquest canvi de coordenades es coneix com a la transformada de Fourier o la conjugació de Hadamard (vegeu [22]).

Considerem el model de Jukes-Cantor en un arbre T amb n fulles i r arestes com una varietat algebraica de dimensió r en $\mathbb{P}^n - 1$, en particular, és la imatge de la funció (3.71).

Un problema important en la filogenòmica és el d'identificar les longituds de branca de màxima versemblança, donat un arbre filogenètic T, una matriu de velocitat de transició Q, i un alineament de seqüències. Per al model de Jukes-Cantor de l'ADN en els 3 taxons (grup d'éssers vius de categoria determinada reconegut pels codis internacionals de nomenclatura botànica, zoològica i bacteriològica), descrits en l'Exemple 3.6, la solució analítica exacta d'aquest problema d'optimització duu a una equació algebraica de grau 23. Vegeu [13] per a més detalls.

Considerem per contra, el problema d'aquesta estimació de màxima versemblança en el cas més simple del model d'ADN de Jukes-Cantor per a dos taxons. Aquí, l'arbre T té només dues fulles, etiquetades per $X = \{1, 2\}$, que surten directament de l'arrel de T. El model ve donat per una aplicació

$$f : \mathbb{P}^1 \times \mathbb{P}^1 \to \mathbb{P}^1, \quad ((\theta_1, \pi_1), (\theta_2, \pi_2)) \mapsto (p_{12}, p_{dis}) \tag{3.89}$$

Les coordenades de la funció f són

$$p_{12} = \theta_1\theta_2 + 3\pi_1\pi_2 \tag{3.90}$$

$$p_{dis} = 3\theta_1\pi_2 + 3\theta_2\pi_1 + 6\pi_1\pi_2. \tag{3.91}$$

Igual que abans, passem a coordenades afins fixant $\theta_i = 1 - 3\pi_i$ per a $i = 1, 2$. Una diferència crucial entre el model (3.89) i l'Exemple 3.6 és que els paràmetres en (3.89) no són identificables. De fet, la imatge de la inversa de qualsevol punt en $\mathbb{P}^1$ sota l'aplicació f és una corba en $\mathbb{P}^1 \times \mathbb{P}^1$. Suposem que ens donen unes dades compostes de dues seqüències d'ADN alineades de longitud n on k bases són diferents. El punt corresponent en $\mathbb{P}^1$ és $u = (n-k, k)$. La imatge de la inversa de u sota l'aplicació f és la corba del pla afí amb l'equació

$$12n\pi_1\pi_2 - 3n\pi_1 - 3n\pi_2 + k = 0. \tag{3.92}$$

Cada punt (π_1, π_2) en aquesta corba encaixa exactament amb les dades $u = (n-k, k)$. Per tant, aquesta corba és igual al conjunt de tots els paràmetres de màxima versemblança per a aquest model i les dades donades. Reescrivim l'equació de la corba de la següent manera:

$$(1 - 4\pi_1)(1 - 4\pi_2) = 1 - \frac{4k}{3n} \tag{3.93}$$

Recordem de (3.69) que la longitud de la branca de l'arrel a la fulla i és

$$3\alpha_i t_i = -\frac{1}{4} \cdot \log\det(P_i(t)) = -\frac{3}{4} \cdot \log(1 - 4\pi_i) \tag{3.94}$$

Prenent logaritmes a ambdós costats de (3.93), veiem que la corba de tots els paràmetres de màxima versemblança es transforma en una línia en les coordenades de la longitud de la branca:

$$3\alpha_1 t_1 + 3\alpha_2 t_2 = -\frac{3}{4} \cdot \log\left(1 - \frac{4k}{3n}\right) \tag{3.95}$$

La suma a la part de l'esquerra de la igualtat és la distància de la fulla 1 a la 2 en l'arbre T. La nostra discussió del model de dos taxons condueix a la següent fòrmula, que es coneix en la biologia evolutiva [9] sota el nom de la correcció de Jukes-Cantor.

Proposició 11. *Donat un alineament de dues seqüències de longitud n, amb k diferències entre les seues bases, l'estimador de màxima versemblança de la longitud de branca és igual a*

$$\delta_{12} = -\frac{3}{4} \cdot \log\left(1 - \frac{4k}{3n}\right). \tag{3.96}$$

Hi ha hagut progressos a l'hora de resoldre equacions d'estimació de manera exacta per a arbres menuts. Pot donar-se el cas que T siga desconegut, aleshores el problema no és seleccionar un punt de la varietat, sinó que el problema és seleccionar un punt d'una quantitat exponencial de varietats.

Els models evolutius discutits adés no permeten esdeveniments d'inserció ni eliminació. També assumixen que els punts evolucionen de manera independent. Malgrat que molts dels models més usats es basen en aquestes suposicions, la realitat biològica demana models que incloguen esdeveniments d'inserció i eliminació, i la flexibilitat de permetre dinàmiques del genoma com transposicions. Aquesta necessitat duu a les esquenes una caterva de problemes d'investigació encara per resoldre i refinar.

3.7 Resultats

Per acabar, exposem els resultats obtinguts en [19] per tal de determinar quina és la probabilitat de trobar en 9 genomes d'espècies diferents el mateix fragment de cadena d'ADN de longitud ℓ de casualitat. En particular, estudiarem els genomes exposats al principi d'aquest treball: el peix zebra (Danio rerio), el peix fugu (Takifugu rubripes), el peix globus (Tetraodon nigroviridis), el gos (Canis familiaris), l'humà (Homo sapiens), el ximpanzé (Pan troglogytes), el ratolí (Mus musculus), la rata (Rattus norvegicus) i el gall (Gallus gallus).

En primer lloc, ho calcularem per a només una posició (és a dir, el fragment és de longitud 1) i assumim que els nucleòtids en diferents posicions són independents entre si. Sota aquesta suposició, calculem la probabilitat d'observar un element ultra conservat d'una longitud determinada per als 9 vertebrats i els seus alineaments. Per al càlcul de la probabilitat en qüestió usem el model d'un arbre filogenètic.

Abans de poder calcular aquesta probabilitat, hem de construir un arbre filogenètic i estimar els paràmetres del model associat. L'arbre per a l'alineament dels 9 vertebrats es mostra en la Figura 3.3. Mitjançant el paquet `PAML` es poden estimar els paràmetres del model per màxima versemblança.

Existeixen molts models d'arbres filogenètics però ens concentrem en el que hem presentat anteriorment: el model de Jukes-Cantor. Amb l'estimació dels paràmetres podem calcular la probabilitat p_{cons} d'observar una posició conservada en l'alineament. Recordem que la probabilitat $p_{i_1 \dots i_s}$ d'observar els vectors de nucleòtids $(i_1, \dots, i_s) \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}^s$ en una columna de l'alineament de s espècies ve donada per un polinomi en l'entrada de les matrius de transició $P_e(t)$, que s'obtenen com a $P_e(t) = \exp(Qt_e)$ on t_e és la longitud de l'aresta e en l'arbre filogenètic i Q és la matriu de velocitat de transició que depèn del model escollit.

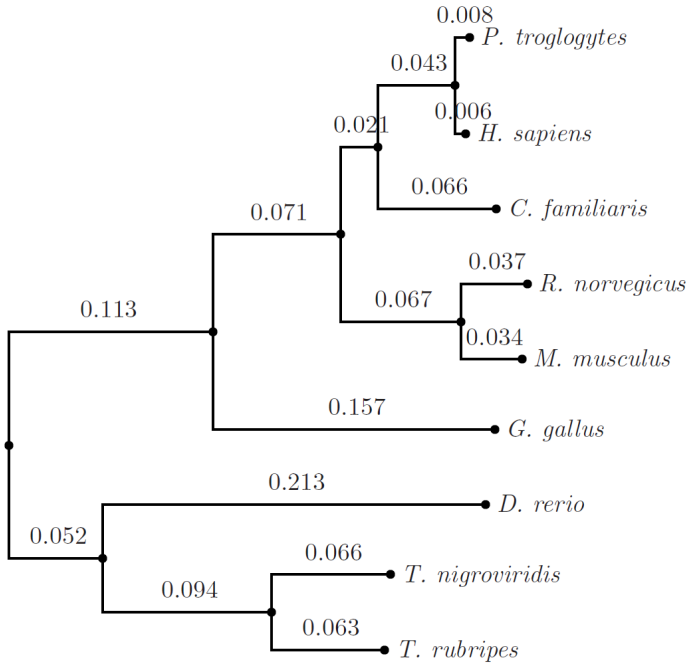


Figura 3.3: Arbre filogenètic per als genomes sencers dels 9 vertebrats en qüestió.

Sota el model de Jukes-Cantor per a l'alineament dels nou vertebrats, les longituds de branca obtingudes per màxima versemblança es mostren en la Figura 3.3 i donen les probabilitats

$$p_{\text{AAAAAAAAA}} = \cdots = p_{\text{TTTTTTTTT}} = 0.0455666\ldots \tag{3.97}$$

Així, la probabilitat d'una columna conservada sota aquest model és $p_{\text{cons}} = 0.1823$.

Els indicadors binaris de que es conserve una posició són independents i idèntics i segueixen una distribució de Bernoulli amb probabilitat d'èxit (es conserva) p_{cons} . La probabilitat que es conserven com a mínim ℓ posicions consecutives començant des d'una posició de l'alineament és p_{cons}^{ℓ} .

longitud	probabilitat
15	0.023
20	$4.60 \cdot 10^{-6}$
125	$1.11 \cdot 10^{-83}$

Taula 3.3: Probabilitat d'observar elements ultra conservats d'una determinada longitud amb probabilitat d'èxit $p_{cons} = 0.1823$ derivat del model d'arbre filogenètic Jukes-Cantor.

La Taula 3.3 avalua la probabilitat per a determinats valors de ℓ.

Tanmateix, 46% de les columnes sense cap buit (-) en l'alineament dels 9 vertebrats tenen un únic element (i per tant és el mateix per a les 9 espècies). Comparant-ho amb amb el 18% que esperaríem amb el model de Jukes-Cantor, ens suggereix que la suposició que hem fet al principi de que les posicions de l'alineament eren independents és massa simple. Si reduïm l'alineament a una seqüència d'indicadors binaris que indique les posicions que es conserven, aleshores un model molt senzill dependent per a aquesta seqüència binària és la cadena de Màrkov.

En una cadena de Màrkov, la longitud d'elements ultra conservats està distribuïda geomètricament, i.e., la probabilitat que un element ultra conservat siga de longitud ℓ és $\theta^{\ell-1}(1-\theta)$, on θ és la probabilitat de transicionar d'una posició ultra-conservada a una altra. El valor esperat de la longitud d'un element ultra conservat és $1/(1-\theta)$. La probabilitat que aquest element siga de longitud ℓ o superior és igual a

$$\sum_{k=\ell}^{\infty} \theta^{k-1}(1-\theta) = \theta^{\ell-1} \tag{3.98}$$

Per això, la probabilitat que com a mínim un dels U elements ultra conservats trobats en l'alineament múltiple siga com a mínim de longitud ℓ és $1-\left(1-\theta^{\ell-1}\right)^U \approx U\cdot\theta^{\ell-1}$ per a ℓ gran.

Amb la mitjana de la longitud d'elements ultra conservats, es pot estimar la probabilitat de transició θ a 0.4785. Aleshores, la probabilitat que com a mínim un dels elements ultra conservats en l'alineament dels 9 vertebrats siga de longitud 25 o superior és d'un 3%. per a $\ell \geq 30$ la probabilitat és menor de 1/1000.

Açò suggereix que el model de la cadena de Màrkov no capta bé l'estructura de dependència en la seqüència binària.

Capítol 4

Conclusions i valoració personal

Respecte a l'estada en pràctiques que vaig realitzar d'octubre a febrer, s'ha aconseguit que el sframe tinga totes les funcionalitats que té el cframe. Malgrat no haver obtingut bons resultats de prestacions en comparació al cframe, cal destacar el tamany que s'ocupa però no s'usa quan s'actualitzen els chunks vells per d'altres que ocupen més tamany en un cframe. A més a més, aquesta implementació obre les portes a noves possibilitats d'emmagatzematge remot. Amb una mica de feina addicional, es podria emmagatzemar un schunk de manera remota. Així, amb només el fitxer chunks.b2frame i l'index del chunk que es vulga obtindre, es podria accedir a una base de dades remota i s'aconsegueix estalviar molt d'espai local. Una altra opció molt interessant també és la contrucció de bases de dades remotes clau/valor. Cada valor seria un chunk identificat pel seu index. De la mateixa manera que en l'anterior opció, amb només el fitxer chunks.b2frame i l'index del chunk es podria accedir al chunk (valor) remot.

L'estada en pràctiques m'ha servit per a conéixer en més profunditat les eixides laborals del Grau en Matemàtica Computacional, conéixer diferents maneres de treballar (tant individualment com en equip) i aprendre i millorar temes tractats en algunes assignatures del grau.

Respecte al TFG, el que es buscava era parlar d'un tema que involucrara l'àlgebra i l'estadística. Aquest tema fou consesuat amb el tutor Pablo Gregori, qui va sugerir parlar de la filogenètica arran de l'article divulgatiu de Marta Casanellas [5]. En aquest article es va poder apreciar l'estreta relació que hi havia entre les matemàtiques i la biologia, així com els nombrosos problemes que encara estaven per resoldre. Cal recordar que aquest tema és encara una branca d'investigació molt recent i activa, que cerca crear i optimitzar solucions i entendre millor les necessitats dels problemes. Això ha fet que em trobara amb una complexitat molt elevada principalment pel tipus de problemes que s'intenten arribar a resoldre. Aquesta complexitat ha sigut un entrebanc per a poder exposar aquest tema amb més deteniment, claredat i sobretot,

justificació. Tanmateix, un dels llibres que més m'ha ajudat a entendre el tema és "Algebraic Statistics for Computational Biology"a càrrec de Bernd Sturmfels i Lior Pachter [14]. Tot i així, hi ha molts models que no s'han exposat en aquest treball que són molt útils per a la filogenètica, com per exemple el model Strand Symmetric. També cal destacar l'important paper que juga el Neighbor-Joining a l'hora de reconstruir arbres filogenètics. Després d'haver realitzar aquest treball, em sembla fascinant tot el joc i l'interés que poden despertar unes cadenes formades per només 4 lletres diferents.

Bibliografia

[1] Kullback-leibler divergence.

[2] N. Bray and L. Pachter. Mavid: Constrained ancestral alignment of multiple sequences. *Genome Res.*, (14):693–699, 2004.

[3] P. Bucher and K. Hofmann. A sequence similarity search algorithm based on a probabilistic interpretation of an alignment scoring system. *Conference on Intelligent Systems for Molecular Biology (ISMB '96)*, pages 44–51, 1996.

[4] C. Burge and S. Karlin. Prediction of complete gene structures in human genomic dna. *J. Mol. Biol.*, (268):78–94, 1997.

[5] Marta Casanellas. Técnicas algebraicas para la evolución de las especies.

[6] E. E. Eichler and D. Sankoff. Structural dynamics of eukaryotic chromosome evolution. *Science*, (301):793–797, 2003.

[7] S. Elizalde. *Algebraic Statistics for Computational Biology*, chapter Inference functions, pages 215–225. Cambridge University Press, 2005.

[8] A. Siepel et al. Evolutionarily conserved elements in vertebrate, insect, worm and yeast genomes. *Genome Res.*, (15):1034–1050, 2005.

[9] J. Felsenstein. *Inferring Phylogenies*. Sinauer Associates, 2003.

[10] G. Bejerano, M. Pheasant, I. Makunin, S. Stephen, W. J. Kent, J. S. Mattick, and D. Haussler. Ultraconserved elements in the human genome. *Science*, (304):1321—-1325, 2004.

[11] Branko Grünbaum. *Convex polytopes*, volume 221. Springer-Verlag, 2003.

[12] D. Gusfield. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, 1997.

[13] S. Hosten A. Khetan and B. Sturmfels. Solving the likelihood equations. *Found. Comput. Math.*, (5):389–407, 2005.

[14] Bernd Sturmfels Lior Pachter. The mathematics of phylogenomics, 2004.

[15] M. Drton, N. Eriksson, and G. Leung. *Algebraic Statistics for Computational Biology*, chapter Ultra-conserved elements in vertebrate and fly genomes, pages 387–402. Cambridge University Press, 2005.

[16] M. Kellis, B. Birren, and E. Lander. Proof and evolutionary analysis of ancient genome duplication in the yeast saccharomyces cerevisiae. *Nature*, (8):617–624, 2004.

[17] R. Durbin S. R. Eddy A. Korgh G. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1999.

[18] L. Pachter and B. Sturmfels. Tropical geometry of statistical models. *Proc. Natl. Acad. Sci.*, (101):16132–16143, 2004.

[19] L. Pachter and B. Sturmfels. *Algebraic Statistics for Computational Biology*. Cambridge University Press, 2005.

[20] D. Kulp D. Haussler M. G. Reese and F. H. Eeckman. *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology (ISMB '96)*, chapter A generalized hidden Markov model for the recognition of human genes in DNA, pages 134–142. AAAI Press, 1996.

[21] R. P. Stanley. *Enumerative Combinatorics,*, volume 1. Cambridge University Press, 1997.

[22] B. Sturmfels and S. Sullivant. Toric ideals of phylogenetic invariants. *J. Comput. Biol.*, (12):204–228, 2005.

[23] J. Watson and F. Crick. A structure for deoxyribose nucleic acid. *Nature*, (171):964–967, 1953.

[24] C. F. Jeff Wu. On the convergence properties of the em algorithm, 1983.

[25] V. B. Yap and L. Pachter. Identification of evolutionary hotspots in the rodent genomes. *Genome Res.*, (14):574–579, 2004.

[26] GM Ziegler. *Lectures on polytopes*, volume 152. Springer-Verlag, 1995.