Predicció de mutacions a partir de la descomposició per residus de l'energia d'unió proteïna-pèptid

Treball Final de Grau

Facultat Informàtica de Barcelona
Universitat Politècnica de Catalunya - BarcelonaTech

Luis Faura Romagosa

Directors: Lluís Belanche Muñoz
Josep Maria Campanera Alsina
Especialitat: Computació

Barcelona
23 de juny de 2015

Resum

En aquest projecte estudiem un cas d'estudi a partir d'unes dades reals per tal de crear un model predictiu de mutacions. En aquestes dades tenim informació sobre la descomposició de l'energia d'unió a nivell de residus de la unió proteïna-pèptid. S'han modelat usant dinàmica molecular per a obtenir un modelatge molecular (informació sobre cada residu per separat). S'intenta que la unió d'aquesta proteïna (anticòs) amb el lligand (el pèptid, antigen) millori. Per fer això hem d'intentar disminuir l'energia d'unió total (ΔG_{bin}). A partir d'aquestes dades, hem creat un model predictiu usant la informació que hem extret usant mètodes de correlació, correlació parcial i regressió PLS i les informacions que tenim sobre els diferents aminoàcids, en concret la hidrofobicitat de cadascun d'ells, per tal de predir mutacions d'aminoàcids (canviar un residu per un altre). Amb aquestes mutacions sobre algun dels pèptids dels quals tenim dades, esperem millorar els pèptids per tal que els anticossos detectin millor, en el nostre cas, els causants de la malaltia de Chagas. A partir de les observacions i resultats obtinguts en aquest estudi, s'ha desenvolupat una aplicació interactiva en `R Shiny` el més general possible per a poder automatitzar la creació d'aquest model predictiu per qualsevol conjunt diferent de dades similars. Aquesta aplicació estarà disponible a la plataforma *online* ScienceNodes.

Abstract

In this project we study a case study based on some real data to create a predictive model of mutations. These data have information on the breakdown of binding energy in the level of residues from the protein-peptide binding. It has been modeled using molecular dynamics to obtain a molecular modeling (with separate information for every residue). The goal is to improve the union of this protein (antibody) with the ligand (peptide, antigen). To do this, we must try to reduce the total binding energy (ΔG_{bin}). From these data, we created a predictive model using the information extracted using methods of correlation, regression and partial correlation PLS and the information we have about the different amino acids, in particular hydrophobicity of each of them, to predict amino acid mutations (change one residue for another). We hope that with these mutations in any of the peptides of which we have data, improve the detection by our antibodies of, in our case, the cause of Chagas disease. Based on observations and results obtained in this study, we have developed a generalized interactive application in `R Shiny` in order to automate the creation of this predictive model in any different set of similar data. This application will be available on the online platform ScienceNodes.

Índex

Índex de taules

Índex de figures

Índex de codis

Capítol 1

Introducció

Aquest projecte es realitza com a Treball Final de Grau (TFG) dels estudis de Grau en Enginyeria Informàtica, de l'especialitat de Computació, a la Facultat d'Informàtica de Barcelona (FIB). S'ha realitzat amb la col·laboració de'n Josep Maria Campanera Alsina del Departament de Fisicoquímica (Facultat de Farmàcia) de la UB.

En el transcurs del projecte s'ha desenvolupat una aplicació interactiva per construir, validar i aplicar models QSAR (*Quantitative structure—activity relationship*) orientats al disseny de millors inhibidors que puguin ser usats com a medicació anticancerosa. S'han utilitzat tècniques de regressió PLS (*Partial Least-Squares*) per trobar grups de residus susceptibles de ser mutats, així com l'exploració de mètodes de modelització no lineals, en concret KPLS (*Kernel PLS*), per tal d'estudiar-ne la seva viabilitat i decidir si és convenient usar-los. L'aplicació s'ha desenvolupat a partir d'un cas d'estudi amb unes dades de mostres proporcionades pel Dr. Campanera. Un cop acabat el desenvolupament de l'aplicació, s'ha publicat a ScienceNodes[1], desenvolupada al departament de Ciències de la Computació de la UPC amb la col·laboració del Dr. Campanera.

1.1 Contextualització

La resposta immunològica al paràsit *Trypanosoma Cruzi* causant de la malaltia de Chagas depèn del reconeixement de la proteïna MHC (*Major Histocompatibility Complex*, un anticòs) envers pèptids de 9 residus (antigens) provinents del processament del paràsit. Així per pacients amb aquesta malaltia, l'administració de pèptids que activin el sistema immunitari envers el paràsit ajuda a la seva millora. La malaltia de Chagas o tripanosomosi americana, coneguda també com "el mal dels pobres", és una malaltia àmpliament distribuïda per Amèrica del Sud que es transmet mitjançant les femtes infectades d'insectes dels gèneres Rhodnius i Triatoma. Infecta les cèl·lules de glàndules del sistema nerviós i pot provocar lesions importants a l'esòfag, al cor i a l'intestí gros. El paràsit entra a la sang i és capaç d'arribar al cor, establir-s'hi i malmetre la fibra del múscul cardíac. Això, a llarg termini, pot provocar greus problemes al bombament de la sang (miocardiopaties) [1, 2].

[1]Plataforma *online* de suport a l'aprenentatge i la investigació actualment allotjada a `http://science.cs.upc.edu/`

Els anticossos (proteïnes MHC) són els encarregats d'identificar i neutralitzar elements estranys al cos (antigens), com ara bacteris, virus o paràsits. Un cop identificats a través del reconeixement molecular els anticossos generen un conjunt de senyals per tal que el sistema immunològic continuï la seva tasca contra aquest paràsit. Per tal d'estimular la resposta immunològica del nostre cos envers el paràsit *Trypanosoma Cruzi* s'administren pèptids semblants als provinents del processament del paràsit. El que s'intenta és que aquests anticossos reconeguin i neutralitzin aquest tipus de paràsit en cas de detectar-ho. Per això és important trobar pèptids que estimulin el sistema immunològic. Aquest projecte vol proposar nous pèptids per tal que siguin reconeguts eficientment per l'anticòs MHC.

1.2 Formulació del problema

En aquest projecte busquem predir petites mutacions, d'un o dos residus, d'uns pèptids a partir d'unes matrius que ens proporcionen informació sobre l'energia d'unió d'aquests amb la proteïna MHC. Aquesta energia d'unió (ΔG_{bin}) està dividida en dues parts, una de polar i una altra d'apolar. Aquesta energia d'unió està íntimament lligada amb les propietats fisicoquímiques dels aminoàcids que formen part del pèptid, especialment cal tenir en compte la hidrofobicitat (la qual té relació directa amb l'aportació d'energia polar o apolar del residu) de cada residu possible (aminoàcids) i els volums d'aquests residus. En total tenim fins a 20 aminoàcids naturals que podrien formar part del pèptid. A més, aquí estudiem també 4 versions neutres d'aminoàcids que normalment estan carregats [3].

Amb aquestes mutacions, es busca augmentar l'afinitat de la proteïna vers el pèptid, és a dir que la proteïna reconegui més activament al pèptid. L'activitat biològica és proporcional a l'energia d'unió (ΔG_{bin}) proteïna-pèptid, i en aquest cas ens interessen energies d'unió baixes (molt negatives), pel qual buscarem disminuir ΔG_{bin}.

A la figura 1.1 podem observar un dibuix de com l'anticòs i l'antigen s'uneixen. A l'antigen (el pèptid), podem observar els nou residus, cadascun amb un color diferent. Per tal d'augmentar l'afinitat d'aquest pèptid amb la proteïna, el que busquem és que la proteïna i el pèptid siguin el més compatibles possible tant en hidrofobicitat (que s'atreguin entre ells) i en volum (que càpiguen bé en la posició on es troben).
El punt de partida del treball és el coneixement de l'efecte de cinc pèptids diferent sobre l'energia d'unió proteïna-lligand. La informació provinent d'aquestes mutacions ens pot ajudar a proposar una altra mutació encara més activa? Primer analitzarem aquestes dades per tal d'entendre el comportament energètic de cada una de les nou posicions dels cinc pèptids. La resolució d'aquest primer cas donarà lloc a la generalització de la solució per altres casos i a una aplicació interactiva web per una resolució més ràpida i informativa.

1.3 Modelatge molecular

Inicialment cada sistema proteïna-lligand va ser simulat mitjançant dinàmica molecular. La dinàmica molecular és una tècnica que permet simular l'evolució temporal (variabilitat conformacional) d'un sistema químic. Posteriorment, el càlcul de l'energia d'unió (ΔG_{bin})

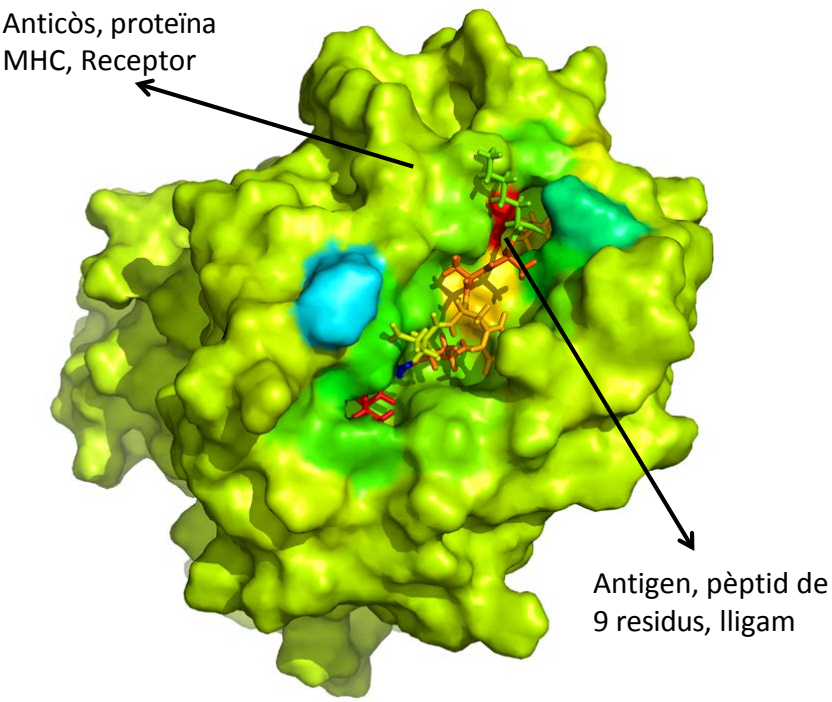

Figura 1.1: Proteïna MHC amb un pèptid

de la proteïna-pèptid s'ha dut a terme utilitzant el mètode MMGBSA [4] com s'implementa a la versió de Perl AMBER12. [5] Per trobar l'energia d'unió cal restar l'energia del receptor (proteïna) i la del lligand (pèptid) a l'energia total del complex receptor-lligand:

$$\Delta G_{bin} = G_{complexe} - G_{receptor} - G_{lligand} \tag{1.1}$$

On ΔG_{bin} és l'energia d'unió, $G_{complex}$ és l'energia lliure del complexe[2], $G_{receptor}$ és l'energia de la proteïna lliure (receptor) i $G_{lligand}$ és l'energia del pèptid (lligand). L'ΔG_{bin} és obtinguda per la mitjana de l'energia lliure d'unió de les molècules implicades. L'energia lliure d'unió MMGBSA (G) és calculada combinant les energies de mecànica molecular energies de solvatació de models continus, els quals, després d'aplicar l'equació 1.1, pot ser expressada de la següent forma:

$$\begin{aligned} \Delta G_{bin} = \Delta E_{MM} + \Delta G_{sol} = \Delta G_{pol} + \Delta G_{apol} = \\ = \Delta E_{int} + \Delta E_{ele} + \Delta E_{vdW} + \Delta G_{sol,pol} + \Delta G_{sol,apol} \end{aligned} \tag{1.2}$$

On ΔE_{MM} és l'energia mecànica molecular expressada com la suma de l'energia interna (ΔE_{int}), l'energia electrostàtica (ΔE_{ele}), el terme de van der Waals (ΔE_{vdW}) i ΔG_{sol} representa l'energia de solvatació que pot ser dividida en una part polar i una altra d'apolar ($\Delta G_{sol,pol}$ i $\Delta G_{sol,apol}$). D'aquestes, l'energia electrostàtica i la part polar de l'energia de solvatació són energies polars, i l'energia interna, el terme de van der Waals i la part apolar de l'energia de solvatació són energies apolars. [6]

[2]El complexe és el conjunt sencer que formen la proteïna i el pèptid.

La descomposició de les contribucions d'aquesta energia d'unió per residus permet desxifrar la xarxa d'interaccions energètiques que estabilitzen la unió proteïna-pèptid, per la qual cosa dóna una idea de la procedència de la unió en un dels residus. Tots els elements d'energia de l'equació 1.2 es poden descompondre amb cert nivell d'aproximació en contribucions per residu d'acord a l'esquema estàndard següent:

$$\Delta G_{bin} = \sum_{i=1}^{n} \Delta G_i = \Delta G_{receptor} + \Delta G_{lligand} \tag{1.3}$$

On n és el nombre total de residus i ΔG_i són les contribucions per residus. Amb aquest esquema, l'energia lliure d'unió també es pot dividir en dos parts, la del receptor ($\Delta G_{receptor}$) i la del lligand ($\Delta G_{lligand}$ sumant les parts corresponents per residu a cada fragment. En aquest treball focalitzarem en l'entesa de l'energia lliure provinent del lligand (el pèptid), és a dir com possibles mutacions poden millorar l'energia d'unió total partint de la informació de mutacions ja estudiades.

1.4 Objectius

Aquests són el objectius que ens hem marcat pel projecte:

- Entendre les dades que tenim i saber analitzar-les
- Predir mutacions a partir de les correlacions amb l'ΔG_{bin}
- Aplicar models de regressió lineal (PLS) sobre les dades per tal de trobar relacions entre posicions
- Estudiar la viabilitat d'aplicar model de regressió no lineal (KPLS) sobre les dades per tal de trobar relacions entre posicions
- Crear una aplicació interactiva per a treballar amb les dades donades
- Generalitzar l'aplicació per a poder treballar amb dades similars

1.5 Motivació

Aquest projecte és interessant ja que permetrà desenvolupar una aplicació web científica per a un cas real en el camp del modelatge molecular No només he hagut de programar l'aplicació, sinó que he hagut d'investigar sobre la part de química i biologia que toca i ha sigut difícil en alguns moments entendre alguns conceptes, però ja tenia nocions de química pel que no ha suposat cap obstacle.

1.6 Actors

Ara detallarem els actors (*stakeholders*) implicats en aquest projecte, és a dir, aquelles persones o organitzacions que poden estar interessades en aquest.

Desenvolupador, dissenyador i *tester*

Les tasques de desenvolupador, dissenyador i *tester* han sigut realitzades per mi mateix, ja que soc la única persona que ha portat a terme aquest projecte.

Directors del projecte

En aquest projecte, hi han dos directors. Un professor del departament de CS de la UPC (Lluís Belanche Muñoz), que s'ha encarregat de supervisar el projecte i ajudar-me en la part dels mètodes de regressió i la part més computacional; i un investigador de la Facultat de Farmàcia de la UB (Josep Maria Campanera Alsina) al que l'interessa el resultat del projecte i m'ha ajudat en l'àrea de la química que implica aquest projecte i a l'anàlisi de les dades.

Investigadors

El projecte pot ser molt interessant per als investigadors que treballin amb gran quantitat de dades que segueixin el model de les que estem estudiant, ja que els permetria treballar amb elles de forma visual i interactiva i els podria estalviar molt de temps quant a processament d'aquestes dades. Com s'espera publicar l'aplicació al web de ScienceNodes, es preveu que l'aplicació sigui accessible per a tots els investigadors interessats. A part, s'espera donar a conèixer l'aplicació per vies comunicatives on puguin estar els investigadors que treballen en aquest camp.

Malalts i persones en risc de la malaltia de Chagas

Les persones que tenen la malaltia de Chagas o estan a un país on es donen casos d'aquesta malaltia, es poden veure beneficiats pels avenços en una millora dels medicaments que necessiten per activar el seu sistema immunitari envers el paràsit causant de la malaltia.

Empreses farmacèutiques

Poden estar interessades en millors medicaments envers aquesta malaltia, resultants dels resultats obtinguts al final del projecte.

1.7 Estat de l'art

En aquest projecte s'ha treballat amb models QSAR que modelen la descomposició per residus de l'energia d'unió proteïna-pèptid. Ens limitem als models amb 9 pèptids per al cas d'estudi. I en el nostre cas, tindrem que preparar els mètodes necessaris per a manegar aquestes dades, ja que actualment no existeix cap paquet d'`R` que ens permeti treballar amb el tipus de dades que tenim [2, 3, 7, 8].

En quant a la part de predicció, s'ha reutilitzat el paquet ja existent sobre la tècnica de regressió PLS. En quant a la regressió no lineal KPLS, el paquet existent, encara que té una

funció que es diu igual, no és exactament el que necessitem en el nostre cas. Pel que, encara que hem vist que no era necessari fer-ho servir, s'ha implementat a partir de les propostes de l'article citat a la bibliografia [9 - 11].

Actualment no existeix cap eina similar que permeti predir mutacions a partir de les dades obtingudes en la investigació. Pel que hem hagut d'implementar el codi necessari per a treballar amb el tipus de dades del que partim, podent reaprofitar algunes funcions com les tècniques de regressió PLS.

1.8 Abast del projecte

1.8.1 Predicció de les mutacions

En aquest projecte ens hem limitat a estudiar un cas d'estudi amb pèptids de 9 residus. A partir d'aquestes dades, hem predit una o dos mutacions dels pèptids amb millor potencial. El que busquem és una millora respecte el que tenim ara, ja que com dèiem abans, no és viable trobar el millor pèptid dins el conjunt de possibles solucions. En aquest moment hi han molt pocs pèptids provats científicament i d'els que es tenen dades, ja que provar una nova mutació pot trigar varies setmanes. Per això, a partir de les dades que disposem, volem triar d'entre els possibles pèptids a provar, els millors candidats.

1.8.2 Aplicació interactiva

Els models QSAR són models de regressió usats en els camps de biologia i enginyeria. En aquest projecte els usem per a modelar la descomposició per residus de l'energia d'unió proteïna-pèptid. Per tal de trobar les possibles mutacions, s'usen els coeficients de correlació entre les posicions i l'energia d'unió i tècniques de regressió lineals. S'ha estudiat la viabilitat d'usar mètodes no lineals. Per a les lineals s'usarà `PLS` i per a les no lineals es s'ha estudiat i implementat el `KPLS`.

En concret hem usat el llenguatge de programació `R`[3], ja que és un llenguatge amb molta potència programadora en estadística computacional i incorpora moltes llibreries que ens poden ser útils. En altres llenguatges s'haurien de programar des de zero. Hem fet ús del paquet `Shiny`[4] d'`R` que permet transformar fàcilment *scripts* d'`R` en aplicacions interactives. Aquesta tecnologia ens permet fer l'anàlisi i la visualització de forma interactiva. Una vegada finalitzada l'aplicació funcional, s'ha incorporat a l'eina ScienceNodes per tal que sigui accessible *online* des de qualsevol navegador modern.

Per a treballar amb els models QSAR, que venen donats en matrius, hem tingut que programar el codi necessari, ja que són d'un tipus específic i el paquet existent per als models QSAR, `QSARdata` [8], no permet manegar aquestes dades.

[3] `http://www.r-project.org`
[4] `http://shiny.rstudio.com`

Per a la implementació de les tècniques de regressió, hem usat d'`R` per el PLS [9] i, encara que no s'ha trobat viable fer servir el KPLS, s'ha programat des de zero basant-nos en l'article de Rosipal [10]. No existeix cap paquet que implementi aquesta aproximació no lineal del KPLS actualment.

1.9 Possibles obstacles

Els possibles obstacles que poden sorgir durant la realització del projecte són els que es detallen a continuació. També es detallen les possibles solucions si és que n'hi ha.

Temps limitat

Com que el projecte disposa d'un temps força limitat, aproximadament 4 mesos i mig, s'haurà de fer una bona planificació per tal de no quedar-se sense temps al final, preveient possibles complicacions que puguin endarrerir el calendari afegint marges de temps adequats.

Errors en el codi

Com sempre que es fa un nou projecte on s'ha de picar codi, és molt probable que apareguin errors en el codi. Normalment es detecten ràpidament i no suposa gaire temps extra, però pot donar-se el cas que hi hagi un error que no es detecti a primera vista. Per tal que això no passi, s'anirà provant el codi a mesura que es programen les diferents parts de l'aplicació.

Errors en les llibreries

Encara que poc probable, sempre és possible que aparegui un error o inconsistència al fer servir els paquets o llibreries de `R` que es faran servir en aquest projecte. Per tal d'evitar això, s'usaran els paquets i/o llibreries més estables que hi hagi en el moment, sempre evitant fer servir versions beta o no massa testejades. Si fos el cas, s'implementarien des de zero els algoritmes i/o funcions necessàries.

Dades insuficients per a entrenar els algoritmes

Es pot donar el cas que es disposi de poques dades reals amb les quals entrenar els algoritmes per a un correcte funcionament. Per tal d'evitar això, s'intentarà disposar de dades de reserva per si es necessiten més de les que en un principi s'havien previst.

1.10 Estructura del document

El document s'estructura en 4 parts principals. Per una part tenim la introducció (capítol 1, on està situada aquesta secció) on es detalla el context i altra informació rellevant que no es específica de la resolució del problema. A continuació, tenim dos parts on es desenvolupa el treball realitzat. En la primera part (capítol 2) tenim la resolució d'un cas concret a partir d'unes dades de les quals disposàvem. Un cop teníem resolt aquest cas, en la següent

part (capítol 3) detallem el procés que hem fer per a generalitzar el cas concret treballat al capítol 2 i crear l'aplicació interactiva on es pugui treballar amb dades semblants però intentant generalitzar al màxim per a possibles diferències entre les dades. A continuació, tenim el capítol 4 on es detalla l'anàlisi de la planificació i del pressupost del projecte així com la sostenibilitat i compromís social d'aquest. Per últim, tenim el capítol 5 amb les conclusions, objectius complerts i treball futur.

1.11 Metodologia i rigor

Com que el temps és limitat, la millor manera de desenvolupar el projecte és fent servir metodologies àgils amb objectius concrets i a curt termini perquè no s'acumuli la feina al final.

És un projecte on el codi que hem programat no és excessivament llarg. Però hem hagut de treballar bé el problema per tal de modelar-ho correctament. Posteriorment hem fet les proves oportunes per tal de veure que funcionava amb el nivell esperat. També s'ha anat calculant el temps per si es detectava qualsevol problema, tenir temps d'arreglar-ho i si calia canviar la planificació inicial establerta.

S'ha tingut *feedback* constant per part dels directors del projecte per tirar endavant el projecte i sortir de possibles aturades en la planificació inicial del projecte.

1.11.1 Eines de seguiment

Per a la part de desenvolupament del codi necessari, s'ha utilitzat un repositori per tal de poder veure com anava evolucionant el projecte i si per qualsevol motiu es fa un canvi no desitjat, poder restaurar a una versió anterior.

Pel que fa a la part de proves, s'han anat guardant tots els resultats de les proves que es van fent per tal de poder comparar-les *a posteriori* per si sorgia algun error.

1.11.2 Mètode de validació

Per fer un seguiment general del projecte, s'han fet reunions presencials periòdiques amb els directors per tal de veure l'estat del projecte i resoldre possibles dubtes. També la comunicació via correu electrònic ha sigut important per a qualsevol dubte o problema que pugui sorgir.

Respecte a la validació del codi, s'ha anat contrastant el resultat de cada part amb les investigacions del Dr. Campanera per a trobat possibles incoherències o errors.

Capítol 2

Anàlisi d'un cas d'estudi

2.1 Format de les dades

En aquest projecte disposarem de dades provinents de cinc mutacions diferents aportades per un dels directors del projecte propietat del seu grup de recerca a la UB. Per a cada mutació tenim tres matrius en fitxers CSV. Un que conté la matriu d'energia polar, un altre que conté la matriu d'energia apolar i un tercer amb la matriu d'energies totals. A les columnes tenim les variables (tots els residus (aminoàcids[1]) de la proteïna MHC i el pèptid que ens interessa ordenats per la seva posició) i a les files la variació de les contribucions a l'energia d'unió en el temps. En el nostre cas, els residus del pèptid són a les nou últimes columnes, i tenim entre dues-centes i cinc-centes files per a cada mutació. A la taula 2.1 podeu veure un exemple simplificat de les matrius. El número que apareix al costat de les sigles de cada aminoàcid o residu és la posició que ocupa, ja que aquests poden aparèixer més d'un cop.

Temps	...	THR379	ALA380	GLU381	LEU382
1	...	−3.4	0.1	−0.4	−1.2
2	...	−1.4	−2.7	−0.2	−1.9
3	...	−2.6	1.0	0.0	−1.3
4	...	−7.5	−2.3	−2.4	−4.6
5	...	−5.8	−1.6	−3.2	−0.3
6	...	−0.4	0.9	−0.8	0.5
⋮	⋱	⋮	⋮	⋮	⋮
N	...	−2.6	−1.1	−1.8	−2.6

Taula 2.1: Matriu de dades d'exemple

També disposem d'un altre fitxer CSV amb la informació de la hidrofobicitat i els volums per a cada residu possible en la composició dels pèptids. Amb aquesta informació podrem

[1]Només existeixen 20 aminoàcids naturals i en el nostre cas tenim també informació sobre quatre versions neutres de d'aminoàcids que normalment estan carregats. Però es poden repetir en posicions diferents.

proposar els canvis a fer així com mirar si té sentit fer canvis en aquella posició.

2.2 Preprocessament de les dades

En el nostre projecte ens interessa només part de les matrius. De les tres matrius per cada mutació, les quals contenen la informació sobre les aportacions polars, apolars i totals a l'energia d'unió respectivament tal com hem comentat a l'apartat anterior, ens interessa:

- La matriu total sencera per obtenir l'ΔG_{bin} per cada instant de temps.
- Les nou últimes columnes de les matrius polar i apolar, on tenim la informació de cada posició del pèptid utilitzat.

L'ΔG_{bin} s'obté de la manera següent:

$$\Delta G_{bin} = \Delta G_{polar} + \Delta G_{apolar} = \sum_{i=1}^{n} \Delta G_{i,polar} + \sum_{i=1}^{n} \Delta G_{i,apolar} \tag{2.1}$$

On ΔG_i és la contribució del residu i i n és el nombre de residus total. Però com disposem de la matriu total, en el nostre cas podem simplificar-ho i obtenir-la a partir de la matriu total:

$$\Delta G_{bin} = \sum_{i=1}^{n} \Delta G_{i,total} \tag{2.2}$$

Ens interessa fer-ho així perquè es pot donar el cas que la suma de les matrius polar i apolar no coincideixi exactament amb la matriu total a causa de les possibles aproximacions que ens trobem a les dades.

Com disposem de més d'una mutació, ens interessa tenir una matriu que contingui tota la informació sobre cada posició del pèptid, independentment del residu que hi hagi en una certa mutació, relacionat amb l'ΔG_{bin}. Pel que construirem a partir de les matrius polars i apolars que tenim, una nova matriu polar i una altra d'apolar seguint l'esquema que podem observar a la figura 2.1.

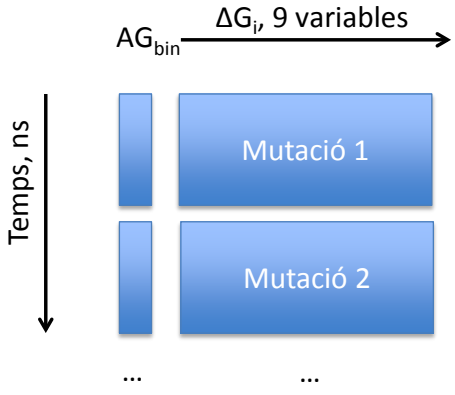

Figura 2.1: Matriu composta

La informació sobre la hidrofobicitat i el volum dels residus ja bé en el format adequat, pel que no necessitem fer cap canvi.

2.3 Anàlisi de les dades

Un cop tenim les matrius compostes amb totes les dades (una matriu amb les aportacions polars i una altra amb les d'apolars) ja podem començar a analitzar aquestes dades per tal de resoldre el problema, que tal com hem definit a la secció 1.2, volem trobar la o les posicions més relacionades amb l'ΔG_{bin} per tal de proposar algunes possibles mutacions que comportin una disminució d'aquesta.

2.3.1 Anàlisi individual a partir de les correlacions

La correlació és una mesura estadística que indica la força i la direcció d'una relació lineal entre dues variables aleatòries, en el nostre cas les variables són ΔG_{bin} i $\Delta G_{i,[polar,apolar]}$. El que busquem amb aquest mètode és trobar els residus que aparentment[2] més influeixen positivament en l'aportació a l'ΔG_{bin} i proposar mutacions sobre aquesta posició tot mirant els residus utilitzats en les mutacions de les quals provenen les dades.

Farem servir el coeficient de correlació d'Spearman (ρ), ja que les nostres dades són contínues. Es calcula utilitzant aquesta fórmula:

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)} \tag{2.3}$$

On n és el nombre de dades que tenim i $d_i = x_i - y_i$ és la diferència entre la posició de cada variable un cop ordenades ascendentment. O sigui, nosaltres tenim dos vectors, ΔG_{bin} i $\Delta G_{i,polar}$. El que fem és ordenar els valors d'aquests vectors i comparem per cada parell, la posició que tenien en el vector sense ordenar i la nova posició en els vectors ordenats. Si hi han valors repetits, es fa la mitjana entre les seves posicions.

Amb això hem construït una taula per a poder visualitzar els resultats, mirar què conclusions podem treure sobre aquests i si tenim suficients dades. En el nostre cas, la taula obtinguda és la que es pot observar a la taula 2.2.

A la taula 2.2 tenim la informació sobre la posició del residu observat, la correlació obtinguda a la matriu polar, el rang de dispersió dels valors polars a la matriu, la correlació obtinguda a la matriu polar, el rang de dispersió dels valors a la matriu apolar i per últim una columna on s'indica si comprovant els valors d'hidrofobicitat dels residus que hi ha a cada posició, sembla que no hi ha contradiccions. El que hem de mirar és que tant la part polar i la part apolar respecte l'ΔG_{bin} ens indiquin la mateixa cosa. No pot ser que, per exemple, la part polar ens digui que una disminució de la hidrofobicitat disminueix l'ΔG_{bin} i la part

[2]Es podria donar el cas que tenim una posició amb molta correlació però que en realitat no influeix gaire en l'ΔG_{bin} i les variacions que observem es deuen a altres canvis, ja que la química no és una ciència exacta.

Total	Posició	Cor. polar	Rang polar	Cor. apolar	Rang apolar	Té sentit?
Total	9	0.463	26.9	−0.102	9.4	Sí
Total	1	0.174	33.6	−0.211	10	Sí
Total	7	0.220	11.9	−0.122	7.3	No
Total	4	0.205	14.9	−0.265	6.3	No
Total	8	0.140	15.7	0.121	6	No
Total	2	0.102	11.1	0.345	6.7	No
Total	6	0.089	8.8	−0.109	5.7	No
Total	5	0.085	7.2	0.068	6	No
Total	3	0.031	36.3	0.236	6.8	No

Taula 2.2: Taula de correlacions

apolar ens digui que un augment en la hidrofobicitat comporta una disminució de l'ΔG_{bin}. En aquest cas de que ens indiquin coses contràries, pot ser que aquestes variacions es donin per altres factors externs i no tingui res a veure amb el que ens interessa que és l'ΔG_{bin}.

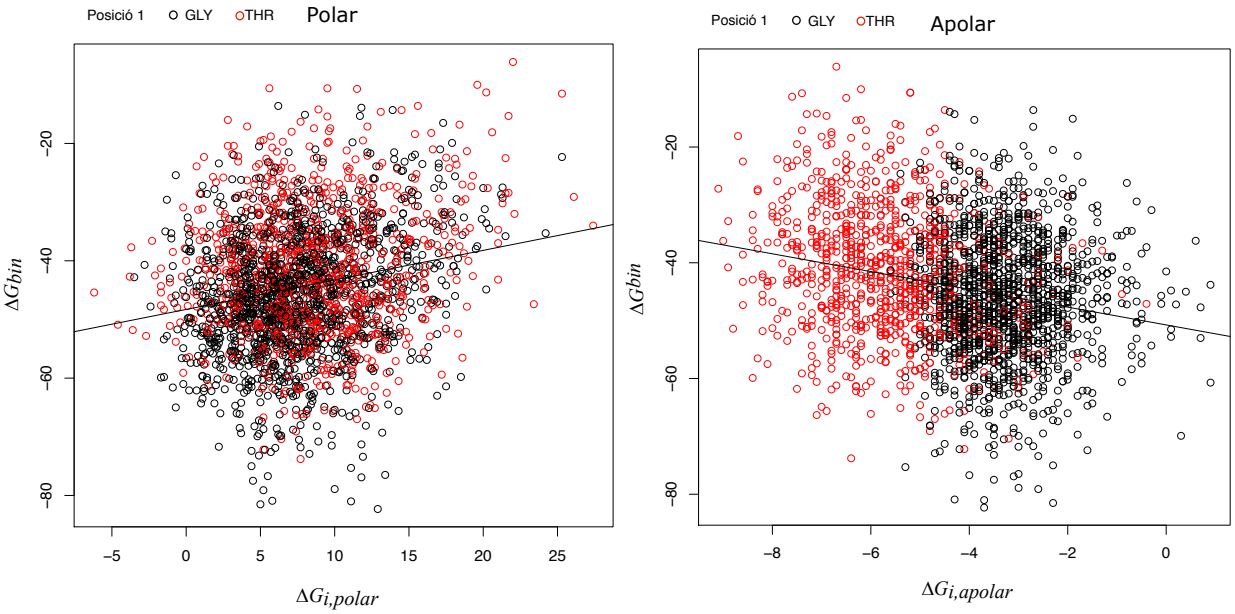

Figura 2.2: Gràfic polar i apolar de la posició 1

Com podem observar a la figura 2.2, tant en la part polar com en la apolar, el residu GLY, amb una hidrofobicitat de -0.77 fa augmentar lleugerament l'ΔG_{bin} respecte al residu THR que té una hidrofobicitat de -0.78. Observem que les pendents són contràries i en tots dos casos, el residu GLY (el pintat en negre), el seu centre se situa per sota del del residu THR. Pel que en aquest cas, buscarem augmentar la hidrofobicitat. Per exemple, podríem proposar una mutació pel residu ALA que té hidrofobicitat -0.47 i un volum que està entre

el GLY i el THR.

2.3.2 Anàlisi individual a partir de les correlacions parcials

El coeficient de la correlació parcial és una mesura de la dependència lineal de dues variables aleatòries en el cas en què la influència de la resta de variables és eliminat. Suposem que les variables aleatòries $X_1, \ldots, X_n$ tenen una distribució conjunta a $\mathbb{R}^n$, definim $X^*_{1;3\ldots n}, X^*_{2;3\ldots n}$ com els millors aproximadors lineals a les variables X_1 i X_2 basat en les variables $X_3, \ldots, X_n$. Llavors, el coeficient de correlació parcial entre X_1 i X_2, denotat com $\rho_{12;3\ldots n}$, és definit com el coeficient de correlació normal entre les variables aleatòries $Y_1 = X_1 - X^*_{1;3\ldots n}$ i $Y_2 = X_2 - X^*_{2;3\ldots n}$ de la següent forma:

$$\rho_{12;3\ldots n} = \frac{\mathrm{E}\{(Y_1 - \mathrm{E}Y_1)(Y_2 - \mathrm{E}Y_2)\}}{\sqrt{\mathrm{D}Y_1 \mathrm{D}Y_2}} \tag{2.4}$$

D'aquesta definició traiem que $-1 \leq \rho_{12;3\ldots n} \leq 1$. El coeficient de la correlació parcial es pot expressar en termes dels elements de la matriu de correlació. Definim $P = \|\rho_{ij}\|$, on ρ_{ij} és el coeficient de correlació entre X_i i X_j, i definim P_{ij} com el cofactor de l'element ρ_{ij} en el determinant $|P|$; llavors

$$\rho_{12;3\ldots n} = -\frac{P_{12}}{\sqrt{P_{11}P_{22}}} \tag{2.5}$$

Per exemple, per a $n = 3$,

$$\rho_{12;3} = -\frac{\rho_{12}\rho_{33} - \rho_{13}\rho_{23}}{\sqrt{(1-\rho^2_{13})(1-\rho^2_{23})}} \tag{2.6}$$

El coeficient de correlació parcial entre dues variables X_i, X_j de $X_1, \ldots, X_n$ es defineix anàlogament. En general, el coeficient de correlació parcial $\rho_{12;3\ldots n}$ és diferent del coeficient de correlació (normal) ρ_{12} de X_1 i X_2. La diferència entre $\rho_{12;3\ldots n}$ i ρ_{12} indica si X_1 i X_2 són linealment dependents, o si aquesta dependència és conseqüència de la dependència d'aquests amb $X_3, \ldots, X_n$. Si les variables X_1 i X_2 són parelles no correlacionades, llavors tots els coeficients de correlació parcial són 0 [12].

Això ens serveix perquè tenim 10 variables, l'ΔG_{bin} i les nou posicions del pèptid, i el que volem és saber la correlació entre l'ΔG_{bin} i cada posició del pèptid, tenint en compte les altres posicions per a treure possibles interferències entre aquestes. També fem servir el coeficient de correlació d'Spearman.

Aquí també hem construït una nova taula de correlacions parcials per a poder visualitzar clarament els resultats. La podem torbar a la taula 2.3.

Podem observar que hi ha un augment generalitzat de les correlacions polars respecte a la taula 2.2 mostrada abans i una disminució de les correlacions apolars. Això es deu a què en el nostre cas, l'$\Delta G_{i,polar}$ és més important que l'$\Delta G_{i,apolar}$.

Total	Posició	Cor. polar	Rang polar	Cor. apolar	Rang apolar	Té sentit?
Total	9	0.517	26.9	−0.059	9.4	Sí
Total	1	0.318	33.6	−0.099	10	Sí
Total	7	0.264	11.9	−0.031	7.3	No
Total	4	0.176	14.9	−0.115	6.3	No
Total	8	0.125	15.7	0.030	6	No
Total	2	0.201	11.1	0.237	6.7	No
Total	6	0.121	8.8	−0.020	5.7	No
Total	5	0.139	7.2	0.018	6	No
Total	3	0.039	36.3	0.113	6.8	No

Taula 2.3: Taula de correlacions parcials

2.3.3 Anàlisi múltiple usant regressió

També ens interessa mirar si hi ha algunes posicions que estan relacionades entre elles, o sigui que si canviem dos pèptids a la vegada, observem una major disminució de l'ΔG_{bin}. Per a fer això, usarem tècniques de regressió, en concret regressió PLS.

La regressió de mínims quadrats parcials (PLS per les seves sigles en anglès *Partial-Least Squares*) és una tècnica per a modelar una relació lineal entre un conjunt de variables de sortida (respostes) $\{y_i\}_{i=1}^n \in R^L$ i un conjunt de variables d'entrada (regressors) $\{x_i\}_{i=1}^n \in R^N$. En el primer pas, el PLS crea variables latents no correlacionades que són combinacions lineals dels regressors originals. El punt clau del procediment és que els pesos usats per a determinar aquestes combinacions lineals dels regressors originals són proporcionals a la covariança entre les variables d'entrada i de sortida. A continuació s'executa una regressió de mínims quadrats sobre aquest subconjunt de variables latents. Això condueix a una estimació de la variància esbiaixada però més baixa dels coeficients dels regressors comparada a la regressió de mínims quadrats ordinària (OLS).

$\mathbf{X}$ representa una matriu $(n \times N)$ d'n entrades i $\mathbf{Y}$ representa una matriu $(n \times L)$ de les corresponents respostes L-dimensionals. Considerem que les variables d'entrada i sortida són centrades, o sigui, que les columnes d'$\mathbf{X}$ i $\mathbf{Y}$ tenen mitjana zero. La forma bàsica és un cas especial de l'algoritme iteratiu no lineal de mínims quadrats parcials NIPALS. NIPALS és un procediment robust per a resoldre problemes de descomposició de valor singular i està estretament relacionat amb el mètode de la potència. Després d'una estimació aleatòria del vector latent t, els dos següents passos es repeteixen fins a la convergència de t i del vector de càrrega p:

1. $p = \mathbf{X}^T t$
2. $t = \mathbf{X}p, t \leftarrow t/||t||$

Després de l'extracció dels vectors t i p, la matriu $\mathbf{X}$ es desinfla[3] per t:

$$\mathbf{X} \leftarrow \mathbf{X} - tt^T\mathbf{X} \tag{2.7}$$

I repetint el procés podem extreure un nou parell de vectors t i p que són ortogonals als anteriors per construcció. Val la pena assenyalar que en el cas que $N < n$ la normalització del vector N-dimensional p després de la primera repetició és computacionalment avantatjosa en comparació amb la normalització del vector n-dimensional t. Però, la normalització de t ens permet adaptar l'algoritme de NIPALS per extreure les variables latents de les matrius kernel[4] $\mathbf{X}\mathbf{X}^T$:

1. $p = \mathbf{X}\mathbf{X}^T t$
2. $t = \mathbf{X}p, t \leftarrow t/||t||$

El desinflatge de la matriu $\mathbf{X}\mathbf{X}^T$ bé donada per:

$$\mathbf{X}\mathbf{X}^T \leftarrow (\mathbf{X} - tt^T\mathbf{X})(\mathbf{X} - tt^T\mathbf{X})^T \tag{2.8}$$

Així, podem aplicar l'algoritme NIPALS a la regressió PLS per tal d'extreure seqüencialment els vectors latents t, u i els vectors de pesos w, c de les matrius $\mathbf{X}$ i $\mathbf{Y}$ en ordre decreixent respecte a els seus valors singulars. El següent és una modificació de l'algoritme "clàsic"NIPALS-PLS en el sentit en que la normalització dels vectors t, u és usada en comptes de la normalització dels vectors w, c:

1. inicialitzar u aleatòriament
2. $w = \mathbf{X}^T u$
3. $t = \mathbf{X}w, t \leftarrow t/||t||$
4. $c = \mathbf{Y}^T t$
5. $u = \mathbf{Y}c, u \leftarrow u/||u||$
6. repetir els passos 2-5 fins la convergència
7. desinflar les matrius $\mathbf{X}$ i $\mathbf{Y}$: $\mathbf{X} \leftarrow \mathbf{X} - tt^T\mathbf{X}$, $\mathbf{Y} \leftarrow \mathbf{Y} - tt^T\mathbf{Y}$

La regressió PLS és un procés iteratiu. Després de l'extracció dels components, es comença de nou usant les matrius desinflades $\mathbf{X}$ i $\mathbf{Y}$ del pas 7. Així, podemarribar a la seqüència dels models fins al punt en què s'aconsegueix el rang d'$\mathbf{X}$. Després de l'extracció dels p components, podem crear les matrius $(n \times p)$ $\mathbf{T}$, $\mathbf{U}$, la matriu $(N \times p)$ $\mathbf{W}$ i la matriu $(L \times p)$ $\mathbf{C}$ consistents de les columnes creades pels vectors $\{t_i\}_{i=1}^p$, $\{u_i\}_{i=1}^p$, $\{w_i\}_{i=1}^p$ i $\{c_i\}_{i=1}^p$, respectivament, extrets en les iteracions individuals.

[3] *Deflate* en anglès, que vol dir que es modifica la matriu per eliminar l'influència d'un vector propi donat.

[4] Encara que s'usan aquestes matrius kernel, no és el mateix que la regressió KPLS que hem descrit anteriorment i de la qual hem estudiat la seva viabilitat, la qual introdueix una transformació no lineal que no hi és aquí.

La regressió PLS es pot escriure's en forma de matriu de la següent forma:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{F} \tag{2.9}$$

On $\mathbf{B}$ és una matriu $(N \times L)$ dels coeficients dels regressors i $\mathbf{F}$ és una matriu $(n \times L)$ de residus. Aquesta matriu és idèntica a les usades en altres models de regressió com la regressió lineal múltiple, la regressió de ridge i la regressió principal per components. Però, en en contrast a aquests models, la matriu $\mathbf{B}$ té la forma:

$$\mathbf{B} = \mathbf{W}(\mathbf{P}^T\mathbf{W})^{-1}\mathbf{C}^T \tag{2.10}$$

On $\mathbf{P}$ és una matriu $(N \times L)$ consistent dels vectors de càrrega $\{p_i = \mathbf{X}^T t_i/(t_i^T t_i)\}_{i=1}^p$. Com que $p_i^T w_j = 0$ per a $i > j$ i, en general, $p_i^T w_j \neq 0$ per a $i < j$ la matriu $\mathbf{P}^T\mathbf{W}$ és triangular superior i invertible. Així, com que $t_i^T t_j = 0$ per $i \neq j$ i $t_i^T u_j = 0$ per $j > i$, ens dona les següents igualtats:

$$\mathbf{W} = \mathbf{X}^T\mathbf{U} \tag{2.11}$$

$$\mathbf{P} = \mathbf{X}^T\mathbf{T}(\mathbf{T}\mathbf{T}^T)^{-1} \tag{2.12}$$

$$\mathbf{C} = \mathbf{Y}^T\mathbf{T}(\mathbf{T}\mathbf{T}^T)^{-1} \tag{2.13}$$

Substituint les equacions 2.11, 2.12 i 2.13 a 2.10 i usant la ortogonalitat de les columnes de la matriu $\mathbf{T}$, podem escriure la matriu $\mathbf{B}$ com:

$$\mathbf{B} = \mathbf{X}^T\mathbf{U}(\mathbf{T}^T\mathbf{X}\mathbf{X}^T\mathbf{U})^{-1}\mathbf{T}^T\mathbf{Y} \tag{2.14}$$

I les diferents escales dels vectors latents individuals $\{t_i\}_{i=1}^p$ i $\{u_i\}_{i=1}^p$ no influeixen en aquesta estimació de la matriu $\mathbf{B}$ [9, 10].

En el nostre cas tenim la matriu de l'ΔG_{bin} i les matrius polars i apolars amb les aportacions de cada posició (ΔG_i), pel que buscarem obtenir dos models, un per a l'energia polar i una altra per a la d'apolar.

Com abans, hem generat una taula per a poder visualitzar els resultats més fàcilment. Però en aquest cas el que ens interessa són les possibles relacions entre els residus en posicions diferents. Farem servir el mètode *Jackknife*, que és una tècnica de mostreig de les dades especialment útil per a la variància i per al biaix d'estimació. Amb això podem mirar quines posicions són les més interessants per a fer mutacions segons les dades que tenim.

En el nostre cas, als codis 2.1 i 2.2 observem que tant per a la matriu polar com l'apolar, les posicions 1 i 9 són les més interessants, tal com ja havíem observat en els casos anteriors. El que ens hem de fixar dels resultats del *Jackknife* és en l'última columna, on busquem que els valors siguin el més propers a zero, i amb els asteriscos ens marca les posicions interessants, que en aquest cas són totes menys la 3. Amb aquestes 3 observacions diferents, podem dir amb bastant certesa que un canvi en aquestes dues posicions, farà que es redueixi l'ΔG_{bin}.

```
Response y (9 comps):
       Estimate Std. Error Df t value  Pr(>|t|)
GLY377 0.686114   0.019827  9 34.6057 6.938e-11 ***
ILE378 1.225006   0.077492  9 15.8081 7.146e-08 ***
LEU379 0.096037   0.048145  9  1.9947 0.0772042 .
GLY380 1.372410   0.108303  9 12.6719 4.835e-07 ***
PHE381 1.770769   0.178706  9  9.9088 3.862e-06 ***
VAL382 1.440325   0.253801  9  5.6750 0.0003038 ***
PHE383 1.698023   0.110322  9 15.3916 9.017e-08 ***
THR384 0.625848   0.062247  9 10.0543 3.421e-06 ***
LEU385 1.371108   0.061132  9 22.4288 3.297e-09 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Codi 2.1: Resultats *Jackknife* sobre la matriu polar

```
Response y (9 comps):
       Estimate Std. Error Df t value  Pr(>|t|)
GLY377 0.686114   0.028226  9 24.3079 1.615e-09 ***
ILE378 1.225006   0.121354  9 10.0945 3.309e-06 ***
LEU379 0.096037   0.037421  9  2.5664 0.0303666 *
GLY380 1.372410   0.111105  9 12.3523 6.015e-07 ***
PHE381 1.770769   0.272306  9  6.5029 0.0001111 ***
VAL382 1.440325   0.241526  9  5.9634 0.0002118 ***
PHE383 1.698023   0.120357  9 14.1083 1.919e-07 ***
THR384 0.625848   0.066267  9  9.4444 5.747e-06 ***
LEU385 1.371108   0.036813  9 37.2453 3.595e-11 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Codi 2.2: Resultats *Jackknife* sobre la matriu no polar

2.4 Programari `R`

En aquest projecte hem decidit utilitzar el llenguatge de programació `R`. Creiem que és la millor opció per a treballar amb el problema que volem resoldre, ja que ens dóna moltes facilitats per a treballar amb matrius i té molts paquets amb molts dels algoritmes que utilitzem per a analitzar les dades.

A continuació detallarem els mètodes que hem utilitzat en la lectura i l'anàlisi de les dades que podem trobar a les seccions anteriors.

2.4.1 Lectura de les dades

El programari `R` ens dóna bastants facilitats a l'hora de llegir certs tipus de fitxers, així, ens evitem haver de programar una funció que ens llegeixi les dades per a poder treballar amb aquestes. Així, disposem d'una funció anomenada `read.csv(file)` que ens permet llegir les matrius que tenim en format CSV, ja preparades per a poder introduir-les fàcilment.

2.4.2 Preprocessament de les dades

Primer necessitem calcular l'ΔG_{bin} a partir de la matriu d'energies totals. Per a fer això, el que hem de fer és sumar tots els valors de cada fila i crear un nou vector amb aquests. Això és fàcil de fer amb `R`, només hem de fer una cosa com `apply(mat.total[, -1], 1, sum)`. La funció `apply` el que fa és aplicar la funció que li passem per paràmetre (en el nostre cas la suma) a cada fila de la matriu `mat.total`. Si ens fixem, estem agafant la matriu sencera menys la primera columna, que conté la informació sobre el temps. Això ho aconseguim en `R` amb la notació `mat.total[,-1]`, que li diu que volem totes les files i totes les columnes menys una (que per defecte treu la primera). Com que tenim cinc matrius totals diferents, el que haurem de fer és calcular l'ΔG_{bin} per cada una d'elles i després crear un vector a partir de la informació de totes elles (unint els vectors que obtenim). Això ho podem fer creant un vector nou a partir dels cinc que tenim fàcilment amb la funció `c(x, ...)` que el que fa és construir un vector a partir de tots els elements que li passem, però si li passem vectors, el que fa és unir-los en comptes d'inserir directament el vector en la posició que posem, ja que és un vector unidimensional i no suporta tenir vectors dins de vectors.

A continuació, necessitem unir les matrius polars i apolars de les diferents mutacions. Per a fer això, el que hem de fer és canviar els noms de les columnes que ens interessen per a poder utilitzar la funció `rbind`, que el que fa és unir matrius per columnes. Només hem de modificar els noms de les columnes que ens interessen, que en el nostre cas són les nou últimes. I a continuació només agafem aquestes nou columnes per a treballar amb menys dades, ja que si no trigaríem més a fer certes coses després.

2.4.3 Correlació

Per a trobar la correlació entre l'ΔG_{bin} i les ΔG_i de cada posició, el que utilitzem és la funció `cor(x, y, method = "spearman")` on `x` és el vector que conté la informació sobre l'ΔG_{bin} en cada instant de tems i `y` és la columna de la matriu que conté la informació d'ΔG_i on i és la posició d'un residu al pèptid. A `method = "spearman"` especifiquem que volem que s'usi el coeficient de correlació d'Spearman (ρ).

2.4.4 Correlacions parcials

Per a fer les correlacions parcials entre ΔG_{bin} i ΔG_i amb $i \in (1..9)$ hem trobat que la millor forma era utilitzar la funció `pcor(x, method = "spearman")` que podem trobar al paquet `ppcor`[5] creat i mantingut per Seongho Kim. La `x` en el nostre cas és una matriu amb la primera columna contenint l'ΔG_{bin} i la resta de columnes amb la informació sobre ΔG_i on el número de la columna és $i + 1$.

2.4.5 Regressió PLS

Després de provar la regressió OLS que trobem implementada per defecte a les llibreries d'`R`, i la regressió PCR i la PLS de la llibreria `pls` [9], hem observat que les tres donaven

[5]Podem trobar més informació sobre aquest paquet a `http://cran.r-project.org/web/packages/ppcor/ppcor.pdf`

resultats semblants. Al final hem decidit utilitzar el PLS perquè, a més de ser la regressió lineal que ens vam marcar utilitzar des de l'inici, creiem que pel nostre cas funciona millor. Això es deu a que el PLS està pensat per a treballar amb models QSAR, entre d'altres, i els altres mètodes de regressió són més generals.

A la llibreria `pls` de `R` s'usa l'algorisme Kernel PLS [13] que no és la mateixa aproximació a la que anomenem aquí com KPLS, ja que és lineal. Aquest algorisme és una millora vers l'algoritme NIPALS, detallat anteriorment, en quant a eficiència, però dona els mateixos resultats.

Per a fer servir aquestes funcions, necessitàvem normalitzar les matrius que tenim. Per fer això, hem fet servir la funció `scale(x)` on `x` és la mateixa matriu que la utilitzada a les correlacions parcials definida a la secció 2.4.4.

Un cop tenim les matrius normalitzades, utilitzem la crida `plsr(y~.,data = mydata, validation = "CV", jackknife = TRUE)`, on `mydata` és un element `data.frame` que conté el vector de l'ΔG_{bin} i la matriu amb els ΔG_i. `y` és el nom del vector de l'ΔG_{bin} i el que indiquem amb `y~.` és que volem que es faci la regressió de totes les columnes de la matriu que conté les ΔG_i respecte a l'ΔG_{bin}. Hem decidit utilitzar el CV (*Cross-Validation*) pel fet que per a utilitzar el mètode *Jackknife* per a mostrejar les dades, necessitem validar-les, i hem vist que el CV, que indica que faci una validació creuada no exhaustiva, és molt més ràpid que l'altra opció que tenim que és utilitzar la validació creuada exhaustiva LOO (*Leave-one-out*). El `jackknife = TRUE` ens prepara els resultats que obtenim per a poder utilitzar la crida `jack.test(resultat)` i obtenir les dades que podem veure als codis 2.1 i 2.2.

Capítol 3

Implementació de l'aplicació

3.1 Introducció al paquet `Shiny` d'`R`

El paquet `Shiny` d'`R` permet crear aplicacions interactives web fàcilment fent servir `R`. `Shiny` combina la potència computacional d'`R` amb la interactivitat de la web moderna. Per això vam decidir usar aquest paquet per a implementar la nostra aplicació. A continuació detallarem com funciona aquest paquet.

3.1.1 Estructura d'una aplicació `Shiny`

Les aplicacions `Shiny` tenen dos parts clarament diferenciades. Per una part tenim el fitxer on va la part de la interfície d'usuari (UI). L'altra part, anomenada servidor, és on va la part del codi relacionada amb tota la part del càlcul dels elements que es mostren a la part de la interfície (on es generen els gràfics, taules, etc.).

La part de la interfície d'usuari consta d'una funció que es crida en inicialitzar l'aplicació i és on s'ha de posar el codi que genera els elements gràfics. Aquesta funció és la següent: `shinyUI(ui)` on `ui` és una definició de la interfície d'usuari. El fitxer que la conté s'anomena sempre **ui.R**.

La part del servidor consta d'un altre fitxer que ha d'incloure la funció `shinyServer(func)` on `func` és la funció d'aquesta l'aplicació i és en aquest fitxer on es posarà el codi que no és pròpiament de la UI. A continuació es detalla el format d'aquest fitxer, que s'ha d'anomenar sempre **server.R**, i les diferents opcions que tenim.

Al codi 3.1 podem observar un exemple de com seria aquest fitxer. Primer de tot, hem de carregar el paquet `Shiny`. A continuació veiem que tenim la crida a la funció esmentada anteriorment. Aquesta funció `shinyServer` conté un paràmetre, que sempre és la definició funció anònima[1] (`function`) que ha de tenir com a mínim dos paràmetres (`input` i `output`) que són objectes `data frame` que contenen tots els elements d'entrada i sortida de dades

[1]Una funció anònima és una o procediment que no està lligat a cap identificador i son susceptibles de ser passades com a valor, com fem en aquest cas. Aquesta és una característica del llenguatge de programació `R` que podem trobar per exemple al llenguatge `JavaScript`.

que s'usen a la UI, o sigui, aquesta és la forma en què es comuniquen les dues parts de l'aplicació i com poden accedir als elements de l'altra part. Opcionalment pot tenir un tercer paràmetre anomenat `session` que és on es guarda la informació de la sessió de l'usuari actual, que és necessari per a fer certes accions com modificar els elements d'entrada de dades (`inputs`) amb les quals l'usuari interacciona amb l'aplicació.

```
# Exemple del server.R

library(shiny)

# Posició 1 on podem posar codi

shinyServer(function(input, output, session) {

  # Posició 2 on podem posar codi

  output$element <- funció({

    # Posició 3 on podem posar codi

  })
})
```

Codi 3.1: Exemple d'un fitxer **server.R**

En aquest arxiu tenim tres llocs on podem posar el nostre codi. Primer tenim la posició 1 tal com es pot veure al codi 3.1. El codi que es troba en aquesta posició s'executa només una vegada, i és quan l'aplicació s'inicia per primer cop i es deixa enllestida per a què els usuaris puguin accedir a ella. En aquesta part normalment posarem tot el codi possible que només s'hagi d'executar un cop, ja que així ens estalviarem executar-ho més cops dels necessaris. És un bon lloc, per exemple, per posar el codi que carrega les dades estàtiques.

A continuació ens trobem la posició 2. El codi que posem aquí, s'executarà cada cop que un usuari obre l'aplicació (quan inicia sessió, i aquesta informació es guarda a la variable `session` abans descrita). Aquest codi s'executarà, llavors, quan l'aplicació es carrega per a aquest usuari (una mateixa aplicació d'aquestes pot ser usada per més d'un usuari a la vegada sense que hi hagi interferències entre ells). Aquí és bona idea, per exemple, posar codi que prepara la UI per l'usuari a partir del treball realitzat a la posició 1, com per exemple canviar els valors dels *inputs*.

Per últim, tenim la posició 3, que fa referència al codi que es pot incloure dins de les funcions que generen els *ouptuts*. Aquest codi, per tant, s'executarà cada cop que es modifiqui algun paràmetre dels *inputs* que es faci servir en aquest element concret. Aquest codi sempre va dins una certa funció de les que ens dóna el paquet `Shiny`, i sempre s'envia a un element dels de sortida. Aquest codi, normalment prepara algun tipus de gràfic, taula, etc. que s'ha de treure en última instància per a què es pugui mostrar bé a la UI. Existeixen funcions específiques per a cada element que podem mostrar, com gràfics, taules, imatges, text o directament codi HTML.

3.2 Diseny

L'aplicació que hem creat consta de dues parts bastant diferenciades, en quant al disseny. Per una part tenim una columna d'entrada (*inputs*) on podem escollir certes opcions que modifiquen els resultats mostrats a l'aplicació. Disposem de tres grups d'entrada. En el primer grup tenim les opcions per a escollir les dades que volem treballar (les diferents mutacions) i indicar quines són les posicions on es troben els residus del pèptid (per defecte l'aplicació agafarà les nou últimes columnes de les matrius, però per tal de generalitzar, hem ficat l'opció d'escollir-ne la posició, ja que es podria donar el cas en que es trobessin en una altra posició o sigués un pèptid amb menys residus). A continuació, tenim un apartat on podem escollir les posicions que ens interessa observar detalladament i certes opcions de visualització. Per últim, tenim un apartat on podem escollir una estructura entre les disponibles per a visualitzar-la fent servir el visualitzador interactiu JSmol, que serveix per a veure estructures de forma gràfica.

Després hem distribuït els elements de sortida en sis pestanyes. Les pestanyes són: **Data description, Data analysis, Correlation, Partial correlation, PLS regression** i **Peptide visualization**, que es detallaran a la secció 3.3. A la figura 3.1 es mostra un esquema de com es veu l'aplicació.

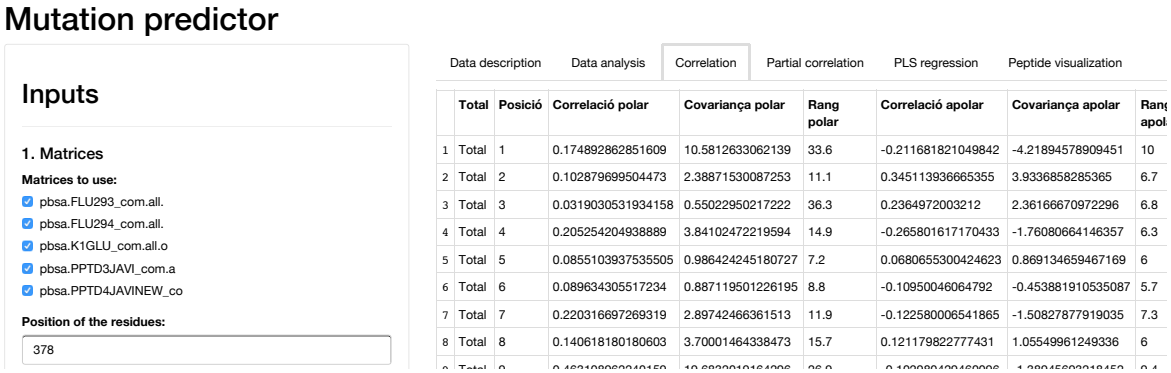

Figura 3.1: Aplicació interactiva

Hem cregut necessari restringir el refresc dels *outputs* per a que no sigui instantani sinó que l'usuari necessiti polsar un botó. Aquesta decisió bé donada pel fet que seria molt ineficient tenir que recalcular cada cop que l'usuari modifica algun paràmetre d'entrada, ja que es possible que necessiti fer més d'un canvi.

3.3 Distribució del codi

A partir del codi que hem fet per a resoldre el cas d'estudi explicat al capítol 2, hem fet un procés per a adaptar-ho i crear l'aplicació. Primer hem creat l'aplicació a partir d'aquest codi, i a continuació, quan ja funcionava bé, hem prosseguit a generalitzar l'aplicació per a poder treballar amb dades similars. A continuació es detalla com hem distribuït el codi per tal d'optimitzar l'aplicació.

3.3.1 Lectura i preprocessament de les dades

Una part important de l'aplicació és disposar de dades amb les quals treballar. Per tal de fer que això fos automàtic, hem decidit que l'aplicació llegiria les matrius del *workspace*[2] on s'executa. Aquestes matrius han de seguir un patró per tal que l'aplicació les reconegui. En concret han d'haver-hi 3 fitxers CSV per a cada mutació diferent, el nom dels quals ha de ser igual i ha de contenir una paraula clau per a cada matriu diferent, que són `pol`, `npol` i `total`. Exemple:

```
pbsa.FLU293_com.all.out.csv.Binding.FLU293.R.TDC.residues.pol.csv
pbsa.FLU293_com.all.out.csv.Binding.FLU293.R.TDC.residues.npol.csv
pbsa.FLU293_com.all.out.csv.Binding.FLU293.R.TDC.residues.total.csv
```

També hi ha d'haver un fitxer CSV amb la informació sobre els diferents residus que ens podem trobar. S'ha de dir `hydromatrix.csv` i s'inclourà per defecte. En cas d'existir més fitxers que no segueixin aquestes nomenclatures, es descartaran. Per a fer això, llistarem els fitxers que acabin amb aquesta extensió i amb expressions regulars trobarem els que ens interessen.

Aquestes matrius es carreguen al inicialitzar l'aplicació, i només un cop, ja que com poden ser grans, seria massa costós llegir-les més d'un cop. Per temes d'optimització de la memòria, l'aplicació només guardarà les matrius polar i apolar, i un cop hagi calculat l'ΔG_{bin} a partir de la matriu total, la destruirà per a què no ocupi memòria, ja que `R` per defecte es guarda totes les variables en memòria local i s'ha de tenir cura amb això, ja que no destrueix mai automàticament cap objecte fins que s'atura l'aplicació i no existeixen variables temporals.

En aquest pas, generem una llista de llistes on cada posició és una mutació diferent i per cada posició guardem les dades que ens interessa per a poder treballar amb les dades independentment del nombre de mutacions que tenim.

A continuació, hem de generar una llista amb les mutacions que tenim, així com mirar on és el pèptid per tal de una vegada l'usuari inicialitzi l'aplicació, poder mostrar aquests valors per que pugui escollir quins usar. Per a poder modificar aquests *inputs* ho haurem de fer cada cop que l'usuari accedeix a l'aplicació, ja que les funcions que modifiquen els *inputs* necessiten la variable *session*. També construirem la matriu descrita anteriorment a la figura 2.1 amb les dades de totes les mutacions per si es dóna el cas de que l'usuari no decideixi descartar cap mutació des de l'inici (que és el que se suposa que passarà habitualment), ens estalviem el cost de generar aquesta matriu composta cada cop que algú obre l'aplicació.

3.3.2 Càlcul dels resultats

Una vegada tenim les dades processades, cada cop que s'obri l'aplicació, l'usuari veurà un resum de les dades (**Data description**) que té i tindrà la possibilitat d'escollir què mutacions vol provar i especificar (si cal) on es troba el pèptid. A continuació, tenim 5 pestanyes

[2]Directori on treballa l'aplicació i on apunta per defecte el sistema d'entrada i sortida de fitxers.

on poder treballar amb aquestes matrius.

Primer de tot trobem la pestanya de **Data analysis** on podrem escollir una posició del pèptid per a poder analitzar quines dades tenim, així com veure gràfics sobre aquestes (com el mostrat a la figura 2.2).

A la següent pestanya, trobem els resultats obtinguts d'aplicar el mètode de la correlació sobre les dades, amb una taula com la mostrada a la taula 2.2, el gràfic de la posició més significativa, així com algunes propostes de mutacions.

A les pestanyes de **Partial correlation** i **PLS regression** trobem els resultats, mostrats de manera similar als de la correlació, d'aplicar aquests mètodes.

Per últim, a la pestanya de **Peptide visualization** podrem veure interactivament l'estructura de la unió proteïna-pèptid.

El codi que necessitem per a mostrar aquestes coses anirà a la part més interior (posició 3), ja que com l'aplicació és interactiva, s'haurà de recalcular cada cop que l'usuari modifiqui els *inputs*.

3.4 Integració amb ScienceNodes

ScienceNodes és una plataforma *online* de suport a l'aprenentatge i la investigació actualment allotjada a `http://science.cs.upc.edu/`. Un cop finalitzada l'aplicació, s'ha treballat per a integrar-la en aquesta plataforma per tal que els investigadors puguin accedir a ella fàcilment. En les pròximes setmanes estarà oberta per a tot investigador que pugui estar interessat a usar-la. La integració en aquesta plataforma ha sigut fàcil, ja que permet pujar els arxius de l'aplicació així com els arxius de les mutacions per tal d'executar-la directament des de la web. No ha calgut fer cap canvi en el codi perquè des d'un principi es va pensar a publicar-la en aquesta plataforma. Tot i això, no és complicat exportar qualsevol aplicació feta en `Shiny`, `R` o d'altres llenguatges, ja que es pot editar el codi directament des del panell d'usuari i dóna moltes facilitats d'ús.

Capítol 4

Planificació i sostenibilitat

4.1 Planificació temporal: estimació inicial

El projecte té una duració estimada d'aproximadament 4 mesos i mig, des de l'inici de GEP (Gestió de Projectes) a mitjans de febrer fins a la defensa del projecte a finals de juny. S'intentarà acabar el projecte a principis de juny per a disposar d'un marge per si alguna tasca dura més de l'esperat.

Com que és una planificació temporal, és probable que hi hagi canvis en la planificació inicial, per això es necessita disposar d'un marge de temps per si s'ha d'allargar alguna de les tasques més enllà del marge ja calculat.

4.1.1 Descripció de les tasques

4.1.1.1 Planificació inicial del projecte (fita inicial)

Aquesta és la part que es realitzarà principalment en seguiment de GEP i inclou la majoria dels lliuraments que s'hauran de realitzar.

És una de les fases més importants, ja que ajuda a l'estudiant a posar en marxa el projecte i encarrilar-ho. En aquesta part es defineixen tots els aspectes del projecte incloent l'abast, una planificació inicial del temps, un estat de l'art inicial i d'altres temes importants.

Aquesta fase dura aproximadament un mes.

4.1.1.2 Familiaritzar-se amb l'entorn

Aquí és on hauré de familiaritzar-me amb les eines que s'usaran per a implementar el codi, en concret tenim els models QSAR, les eines de modelatge (*docking*, *molecular dynamics* i *Molecular Mechanics Poisson-Boltzmann Surface Area (MM-PBSA) decomposition of the binding free energy*) i les tècniques de regressió PLS i KPLS i s'estudiarà la seva viabilitat.

Es calcula que aquesta fase no duri més d'un parell de setmanes.

4.1.1.3 Implementació i proves

En aquesta fase s'implementarà el codi necessari per a fer funcionar l'aplicació. Aquest codi es dividirà en 3 parts clarament diferenciades. Per un costat tenim el codi principal que s'encarregarà de llegir, modelar i mostrar les dades, i després tenim el codi que s'encarregarà d'implementar el PLS i el KPLS si és viable usar-ho i de processar els resultats obtinguts (per separat). A més a més, a mesura que es vagi implementant el codi, s'anirà documentant el que es faci per a després afegir-ho a la memòria del projecte.

Per tal de poder escriure el codi per cada part, serà necessari haver-se familiaritzat amb les eines que s'usaran. En concret per al tros principal serà necessari haver-se familiaritzat amb els models QSAR i les eines de modelatge abans descrites. I per la part de les tècniques de regressió es necessitarà haver-se familiaritzat amb elles prèviament.

Per tal d'assegurar-se que l'aplicació es comporta correctament es faran proves individuals amb cada part del codi un cop acabada cada part. Quan ja estigui tot funcionant correctament, s'acabarà de programar el que sigui necessari per a poder provar l'aplicació sencera. Un cop es tingui, es provarà amb dades reals per veure com respon. Si es detecta alguna anomalia, s'intentarà mirar d'identificar-la i arreglar-la. S'aniran documentant les proves per a afegir-les a la memòria del projecte.

Es calcula que aquesta fase duri aproximadament 2 mesos.

4.1.1.4 Solucionar el problema

En aquesta part és on es prediran les mutacions per a les dades .

Es calcula que aquesta fase duri com a molt 4 setmanes.

4.1.1.5 Integració a ScienceNodes

Un cop tinguem l'aplicació funcional, s'integrarà amb l'eina ScienceNodes, parcialment desenvolupada al departament CS de la UPC, per tal que sigui accessible als investigadors a la que va destinada.

Aquesta fase no hauria de durar més de 2 dies.

4.1.1.6 Finalització del document i preparació de la defensa

En aquesta part es tancarà el document de la memòria del projecte a partir de la documentació obtinguda en les altres parts estructurant-la correctament i afegint el que sigui necessari.

Es començarà a preparar el document definitiu un cop estigui el programa pràcticament enllestit i una vegada iniciades les proves. Un cop enllestit, es prepararà la defensa del projecte, que s'estima que sigui l'última setmana de juny.

4.1.2 Recursos

Per realitzar aquest projecte s'estima que s'usaran els següents recursos:

- Portàtil MacBook Air 13" amb OSX Yosemite (totes les tasques)
- Ordinador de sobretaula iMac de 21" amb OSX Yosemite (totes les tasques)
- Programari `R`[1] per a OSX (per a les tasques d'implementació i proves)
- Paquet `Shiny`[2] (per a les tasques d'implementació i proves)
- Eina ScienceNodes[3] (per a la tasca de la integració amb aquesta eina)
- LaTeX[4] (per a la generació dels documents)
- Git (pel control de versions a les tasques d'implementació i proves)
- Correu electrònic de la FIB (per a les comunicacions amb els directors del projecte)

4.1.3 Diagrama de Gantt

A la figura 4.1 es mostra el diagrama de Gantt per a les tasques definides a la secció 4.1.1, i a la taula 4.2 la part textual per indicar els rols involucrats a cada tasca. Realitzat amb el programari lliure GanttProject[5]. Les tasques de risc son les d'implementació i proves i la de solucionar el problema, en verd i groc respectivament a la figura.

4.1.4 Estimació dels temps per tasca

A la taula 4.1 es detallen les estimacions dels temps que necessitarà cada tasca. Com que el projecte consta de 18 crèdits, el projecte tindria que tenir entre 450h i 540h de treball.

Tasca	Estimació de temps (hores)
GEP	75
Familiaritzar-se amb l'entorn	60
Implementació i proves	190
Solucionar el problema	100
Incorporació a ScienceNodes	5
Finalització del document i preparació de la defensa	50
Total:	480

Taula 4.1: Estimació dels temps per tasca

[1] `http://cran.r-project.org/mirrors.html`
[2] `http://shiny.rstudio.com`
[3] `http://science.lsi.upc.edu`
[4] `http://www.latex-project.org`
[5] `http://www.ganttproject.biz`

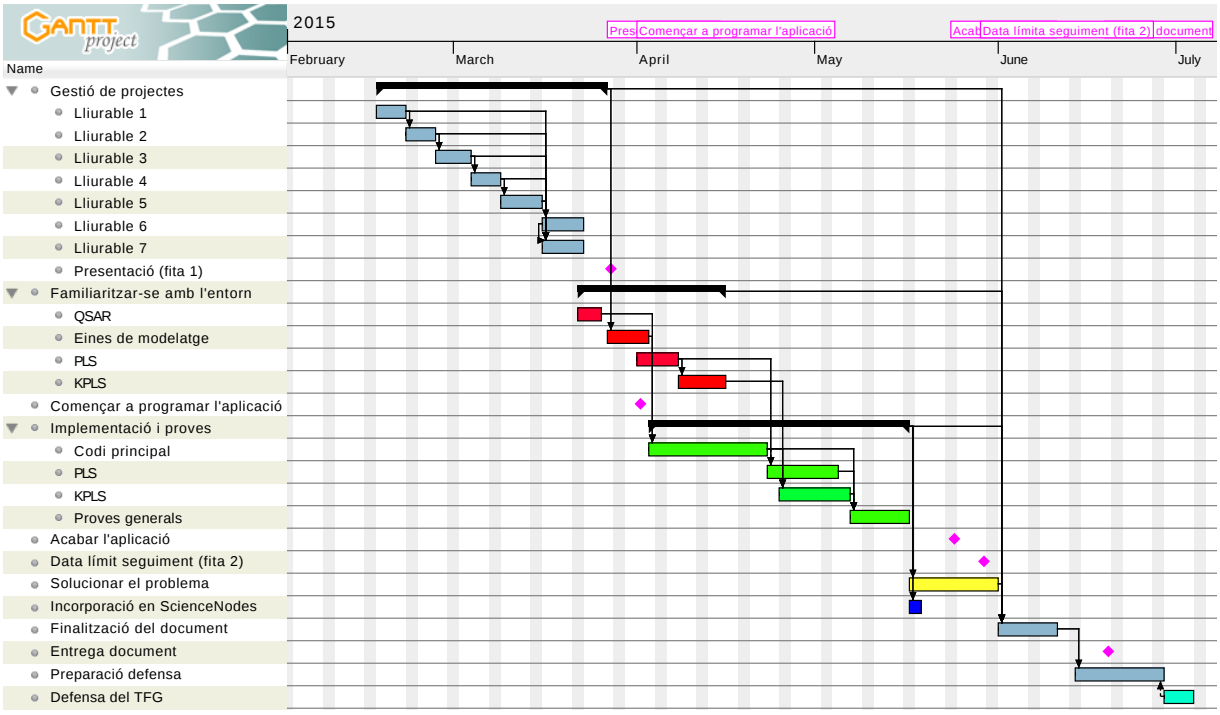

Figura 4.1: Diagrama de Gantt estimació inicial

4.1.5 Valoració d'alternatives i pla d'acció

És probable que es produeixin desviacions quant al temps estimat de cada tasca, ja que és difícil predir segons quin tipus de tasques. Per aquest motiu, s'ha donat temps extra per a cada tasca. En el cas que la tasca s'acabés abans, es continuaria amb la següent sense esperar a la data inicialment calculada per si una altra tasca dura de més. En el cas que una tasca s'allargui, la següent tasca (si és estrictament necessari) començarà més tard també. En el cas inusual que perilli la data de finalització del projecte a causa que una tasca s'allarga molt, es continuarà fent les altres tasques programades per tal de no endarrerir massa el calendari ampliant les hores dedicades al projecte.

Pel que fa al consum de recursos, no es veuran afectats per aquestes possibles modificacions del calendari.

Coordinador	Tasca	Inici	Fi	Rols involucrats
Cap de projecte	Gestió de projectes	2/16/15	3/26/15	Cap de projecte
Cap de projecte	Lliurable 1	2/16/15	2/20/15	Cap de projecte
Cap de projecte	Lliurable 2	2/21/15	2/25/15	Cap de projecte
Cap de projecte	Lliurable 3	2/26/15	3/3/15	Cap de projecte
Cap de projecte	Lliurable 4	3/4/15	3/8/15	Cap de projecte
Cap de projecte	Lliurable 5	3/9/15	3/15/15	Cap de projecte
Cap de projecte	Lliurable 6	3/16/15	3/22/15	Cap de projecte
Cap de projecte	Lliurable 7	3/16/15	3/22/15	Cap de projecte
Cap de projecte	Presentació (fita 1)	3/27/15	3/27/15	Cap de projecte
Programador	Familiaritzar-se amb l'entorn	3/22/15	4/15/15	Programador
Programador	QSAR	3/22/15	3/25/15	Programador
Programador	Eines de modelatge	3/27/15	4/2/15	Programador
Programador	PLS	4/1/15	4/7/15	Programador
Programador	KPLS	4/8/15	4/15/15	Programador
Programador	Començar a programar	4/1/15	4/1/15	Programador
Programador	Implementació i proves	4/3/15	5/16/15	Programador, *Tester*
Programador	Codi principal	4/3/15	4/22/15	Programador
Programador	PLS	4/23/15	5/4/15	Programador
Programador	KPLS	4/25/15	5/6/15	Programador
Programador	Proves generals	5/7/15	5/16/15	*Tester*
Programador	Acabar l'aplicació	5/24/15	5/24/15	Programador
Cap de projecte	Data límit seguiment (fita 2)	5/29/15	5/29/15	Cap de projecte
Investigador	Solucionar el problema	5/17/15	5/31/15	Investigador
Programador	Incorporació en ScienceNodes	5/17/15	5/18/15	Programador
Cap de projecte	Finalització del document	6/1/15	6/10/15	Cap de projecte
Cap de projecte	Entrega document	6/19/15	6/19/15	Cap de projecte
Cap de projecte	Preparació defensa	6/14/15	6/28/15	Cap de projecte
Cap de projecte	Defensa del TFG	6/29/15	7/3/15	Cap de projecte

Taula 4.2: Diagrama de Gantt part textual

4.2 Planificació temporal: resultat final

4.2.1 Diagrama de Gantt

A la figura 4.2 podem veure el diagrama final aproximat de la planificació que hem portat durant el projecte. Hi ha hagut alguns canvis respecte a la planificació inicial comentada a la secció 4.1 que detallarem a la secció 4.2.2.

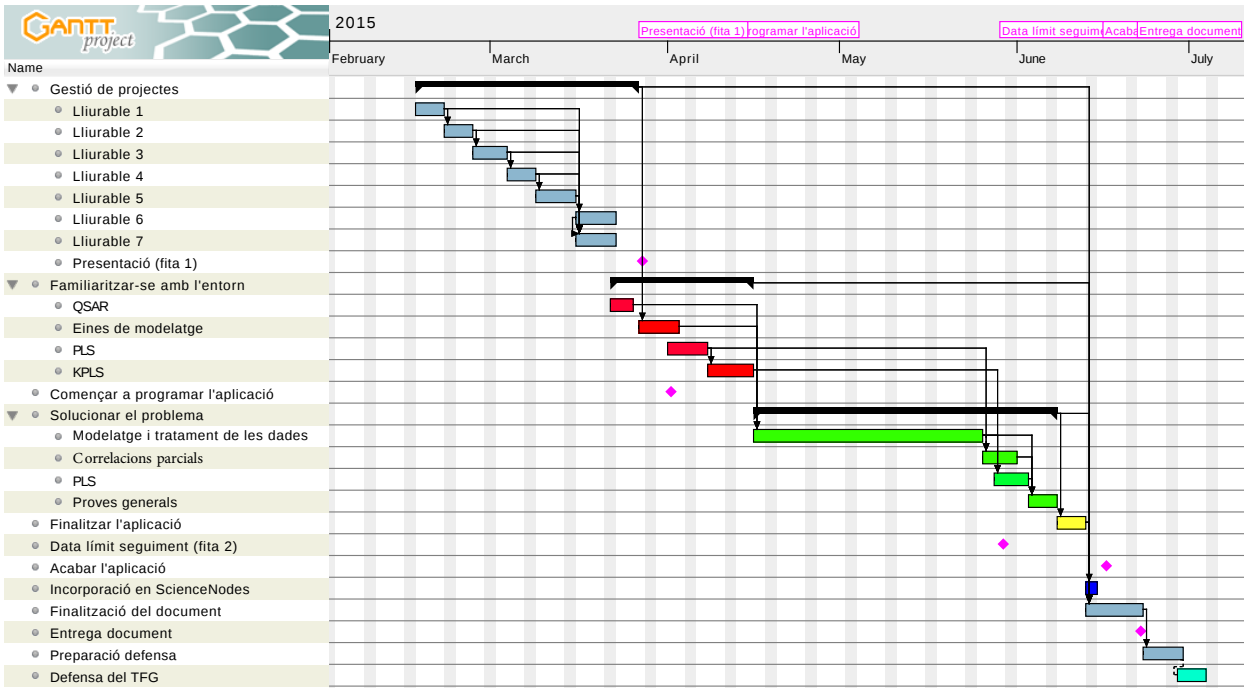

Figura 4.2: Diagrama de Gantt final

4.2.2 Canvis

El principal canvi en la planificació inicial, és que les tasques de la implementació i solucionar el problema, s'han intercanviat. Crec que era necessari aquest canvi, pel fet que, com que no sabíem com volia exactament l'aplicació l'usuari final, hem hagut de tractar molt les dades i intentar solucionar el problema amb diferents mètodes per a veure quin era el més adequat. Si hagués programat una aplicació sense saber exactament com resoldre el problema, probablement hauria hagut d'adaptar molts cops aquesta aplicació, cosa que hauria causat un retràs considerable, impossibilitant que s'acabés el projecte a temps. A partir dels resultats de resoldre el problema, ens serà més fàcil crear l'aplicació final interactiva. Amb aquest canvi, hi ha una petita variació dels costos que es detalla a la secció 4.3.4, encara que es preveu que es mantinguin bastant en els calculats inicialment.

4.2.3 Desviació temporal

Amb els canvis que hem fet en la planificació, hem hagut d'ajustar una mica els temps de cada tasca, pel que al final hi ha hagut un retràs respecte a la planificació inicial d'aproximadament una setmana. Aquest retràs entra dins del marge que havíem deixat a la planificació que vam fer a GEP per a possibles desviacions, pel que no suposa un problema.

4.3 Estimació del pressupost

4.3.1 Identificació dels costos

En aquesta secció es fa una identificació dels costos derivats dels elements considerats en el diagrama de Gantt i les eines necessàries per a cada part, com es pot veure a la taula 4.3.

Tasca	Rol	Hores	Material
Gestió de projectes	Cap de projecte	75	paper, LaTeX, ordinadors, correu electrònic
Familiaritzar-se amb l'entorn	Programador	60	`R`, `Shiny`, ordinadors, correu electrònic
Programació i proves	Programador	145	`R`, `Shiny`, Git, ordinadors, correu electrònic
	Tester	45	ordinador, correu electrònic
Solucionar el problema	Investigador	100	paper, ordinadors, correu electrònic
Incorporació en ScienceNodes	Programador	5	ordinadors, Git, correu electrònic
Finalització del document	Cap de projecte	20	paper, LaTeX, ordinadors, correu electrònic
Preparació defensa	Cap de projecte	30	paper, ordinadors, correu electrònic

Taula 4.3: Identificació dels costos a partir del diagrama de Gantt

4.3.2 Estimació dels costos

4.3.2.1 Pressupost dels recursos humans

A la taula 4.4 es mostren els preus estimats de cada perfil implicat en el projecte, encara que el projecte es dugui a terme per una sola persona.

Rol	Hores	Preu per hora	Preu total
Cap de projecte	125	40,00 €	5.000,00 €
Programador	210	30,00 €	6.300,00 €
Investigador	100	35,00 €	3.500,00 €
Tester	45	20,00 €	900,00 €
Total:	480		15.700,00 €

Taula 4.4: Pressupost dels recursos humans

4.3.2.2 Pressupost de *hardware*

Per tal de fer el projecte cal disposar d'un *hardware*. A la taula 4.5 es detallen els pressupostos del *hardware* necessari detallat a l'apartat de recursos de la planificació temporal.

Producte	Preu	Quantitat	Vida útil	Amortització
MacBook Air 13"	1.029,00 €[6]	1	5 anys	77,175 €
iMac de 21,5"	1.129,00 €[7]	1	5 anys	84,675 €
Total:	2.158,00 €			161,85 €

Taula 4.5: Pressupost de *hardware*

El preu de les reparacions és nul, ja que es disposa de la garantia d'un any que cobreix qualsevol avaria.

4.3.2.3 Pressupostos de *software*

A la taula 4.6 es detallen els costos del *software* que s'usarà, detallat a l'apartat de recursos de la planificació temporal. Com podem observar, el cost es nul, ja que utilitzem majoritàriament programari lliure i el sistema operatiu que usem ve inclòs amb els ordinadors.

Producte	Preu	Quantitat	Vida útil	Amortització
`R`	0,00 €	1		0,00 €
`Shiny`	0,00 €	1		0,00 €
LaTeX	0,00 €	1		0,00 €
Git	0,00 €	1		0,00 €
OSX Yosemite	31,00 €	2	3 anys	0,00 €[8]
Total:	62,00 €			00,00 €

Taula 4.6: Pressupost de *software*

[6] `http://store.apple.com/es/buy-mac/macbook-air` (consultat l'1 de març de 2015)
[7] `http://store.apple.com/es/buy-mac/imac` (consultat l'1 de març de 2015)
[8] Inclòs a ambdós ordinadors

4.3.2.4 Despeses indirectes

En tot projecte informàtic, apareixen unes despeses indirectes com poden ser de l'electricitat, connexió a internet o paper usat. Les detallem a la taula 4.7

Producte/Servei	Preu	Quantitat	Total estimat
Electricitat	0,13 €/kWh[9]	480h x 0.2kW/h[10]	12,48 €
Accés a internet	30,00 €/mes	5 mesos	150,00 €
Paper	5,00 €/paquet	1	5,00 €
Total:			167,48 €

Taula 4.7: Despeses indirectes

4.3.2.5 Pressupost total

A la taula 4.8 es detalla el pressupost total estimat per al projecte en cas que no es fes com a TFG.

Concepte	Cost
Recursos humans	15.700,00 €
Hardware	161,85 €
Software	0,00 €
Altres despeses	167,48 €
Total:	16.029,33 €

Taula 4.8: Pressupost total

4.3.3 Control de gestió

Per tal de controlar les desviacions respecte a el pressupost, s'anirà apuntant diàriament els costos de les tasques realitzades aquell dia. Al final del projecte, es calcularà la desviació total, incloent-hi possibles canvis de la tarifa elèctrica o d'altres desviacions. A la taula 4.9 es mostra un exemple del càlcul de les desviacions.

[9] `http://tarifaluzhora.es` (consultat l'1 de març de 2015)

[10] Dades aproximades tretes de `http://www.appleadictos.com/especiales/consumo-electrico-mac/` (consultat l'1 de març de 2015)

Concepte	Cost estimat	Cost real	Desviació
Recursos humans	15.550,00 €	15.940,00 €	390,00 €
Hardware	161,85 €	1.359,00 €	1.197,15 €
Software	0,00 €	0,00 €	0,00 €
Altres despeses	167,48 €	469,00 €	301,52 €
Desviació total:			1.888,67 €

Taula 4.9: Exemple de control de desviacions

4.3.4 Càlcul de les desviacions sobre el pressupost estimat

Un cop acabat el projecte, hem calculat el cost real segons les hores dedicades i eines usades en cas de que aquest projecte no s'hagués fet com a TFG.

4.3.4.1 Desviacions en els recursos humans

La desviació més important que hem trobat sobre el pressupost estimat al principi és en l'àrea dels recursos humans. Aquest canvi ha sigut motivat pel canvi que hem hagut de fer a la planificació temporal descrit a la secció 4.2.2. Això ha suposat un canvi en les hores preestablertes de cada rol. A la taula 4.11 detallem el nou pressupost segons les noves hores dedicades per cada rol que es detallen a la taula 4.10.

Tasca	Rol	Hores	Material
Gestió de projectes	Cap de projecte	75	paper, LaTeX, ordinadors, correu electrònic
Familiaritzar-se amb l'entorn	Programador	60	`R`, `Shiny`, ordinadors, correu electrònic
Solucionar el problema	Investigador	80	paper, ordinadors, correu electrònic
	Programador	150	`R`, Git, ordinadors, correu electrònic
Programació i proves	Programador	90	`R`, `Shiny`, Git, ordinadors, correu electrònic
	Tester	40	ordinador, correu electrònic
Incorporació en ScienceNodes	Programador	5	ordinadors, Git, correu electrònic
Finalització del document	Cap de projecte	20	paper, LaTeX, ordinadors, correu electrònic
Preparació defensa	Cap de projecte	30	paper, ordinadors, correu electrònic

Taula 4.10: Identificació dels costos a partir del nou diagrama de Gantt

Rol	Hores	Preu per hora	Preu total
Cap de projecte	125	40,00 €	5.000,00 €
Programador	235	30,00 €	7.050,00 €
Investigador	80	35,00 €	2.800,00 €
Tester	40	20,00 €	800,00 €
Total:	480		15.650,00 €

Taula 4.11: Pressupost final dels recursos humans

En general veiem un canvi en les hores dedicades pels rols d'investigador i programador, i això suposa un petit canvi al pressupost final.

4.3.4.2 Desviacions en les despeses indirectes

Com el preu de la llum fluctua cada mes, el preu previst inicialment ha variat una mica, pel que hem de calcular el cost real segons les dades reals.[11] A la taula 4.12 es mostren els costos reals aproximats del projecte.

Producte/Servei	Preu	Quantitat	Total estimat
Electricitat	0,1236 €/kWh[12]	480h x 0.2kW/h[13]	11,87 €
Accés a internet	30,00 €/mes	5 mesos	150,00 €
Paper	5,00 €/paquet	1	5,00 €
Total:			166,87 €

Taula 4.12: Despeses indirectes aproximades

4.3.5 Pressupost real i desviacions

Un cop calculades les desviacions dels recursos humans i les despeses indirecte (no hi han hagut desviacions en quant a *hardware* o *software*) hem calculat el pressupost real i la desviació total segons el que vam establir a la secció 4.3.3. A la taula 4.13 es detalla això. Podem apreciar una lleugera disminució deguda principalment al canvi de les hores que fan l'investigador i el programador, ja que es va calcular que l'investigador té una retribució major.

[11] Dades extretes de `http://tarifaluzhora.es` (consultat el 18 de juny de 2015)

[12] Ibid.

[13] Dades aproximades tretes de `http://www.appleadictos.com/especiales/consumo-electrico-mac/` (consultat l'1 de març de 2015)

Concepte	Cost estimat	Cost real	Desviació
Recursos humans	15.700,00 €	15.650,00 €	-50,00 €
Hardware	161,85 €	161,85 €	0,00 €
Software	0,00 €	0,00 €	0,00 €
Altres despeses	167,48 €	166,87 €	-0,61 €
Total:	16.029,33 €	15.978,72 €	-50,61 €

Taula 4.13: Pressupost real i desviacions

4.4 Sostenibilitat i compromís social

Amb el fi d'identificar i tenir en compte la sostenibilitat del projecte, s'avalua l'impacte en tres aspectes: l'econòmic, el social i l'ambiental. Després d'analitzar cada una de les tres dimensions, s'ha donat una puntuació de 15 sobre 30 a la planificació. Es pot veure amb més detall a la taula 4.14.

Sostenible?	Econòmica	Social	Ambiental
Planificació	Viabilitat econòmica	Millora en la qualitat de vida	Anàlisi de recursos
Valoració	5	8	2
Valoració total:	15		

Taula 4.14: Matriu de sostenibilitat del TFG

4.4.1 Dimensió econòmica

El projecte està previst per a ser usat en la investigació. No està pensat per ser competitiu, sinó per resoldre un problema específic. Pel que és difícil calcular si és viable, ja que la malaltia és més comuna a zones de l'Àfrica o Amèrica del Sud i potser les farmacèutiques no veurien negoci allà.

Pel que fa al temps dedicat a cada tasca és bastant proporcional a la importància d'aquesta, dedicant més hores a la investigació, resoldre el cas d'estudi i desenvolupar l'aplicació interactiva a partir d'aquest que a altres tasques.

Un cop acabat el projecte, es preveu que l'aplicació desenvolupada s'usi per a la investigació dins el camp de la química pel que està dissenyat.

4.4.2 Dimensió Social

La situació política del país on es realitza el projecte és estable, encara que en el sector de l'educació i investigació cada vegada hi ha menys pressupost per part de l'estat espanyol.

La necessitat d'aquest projecte no és elevada, ja que els investigadors poden treballar amb les seves dades manualment, però pot estalviar-los molt de temps. Això pot suposar en una millora en el nivell de vida de moltes persones a països on la malaltia de Chagas es present, sobretot a països de l'Amèrica llatina i l'Àfrica, ja que pot fer que s'avanci més ràpidament en la investigació sobre aquesta malaltia.

4.4.3 Dimensió ambiental

El projecte en si no ha necessitat gaires recursos, ja que és una aplicació que no involucra grans quantitats de màquines per a funcionar o màquines especialment dissenyades per ser usat. Per això l'impacte ambiental és baix. Indirectament pot tenir un impacte una mica més elevat si els investigadors han de mutar els pèptids que el programa proposi.

Capítol 5

Conclusions

5.1 Conclusions

Després d'analitzar el cas d'estudi, hem arribat a la conclusió que les posicions 1 i 9 són les més significatives en el nostre cas, i una mutació en aquestes posicions pot suposar una disminució de l'ΔG_{bin}. Tant usant els mètodes de la correlació com la regressió, hem vist com, encara que a la posició 9 només disposàvem de dades sobre un residu, sembla que una disminució en l'aportació d'energia polar d'aquest residu, comportava una disminució de l'ΔG_{bin}, per tant seria una bona opció provar de mutar aquesta posició. Amb la posició 1 passava una cosa similar, a més, com d'aquesta posició sí que disposàvem d'informació sobre dos residus diferents, ha sigut més fàcil veure la relació entre aquesta posició i l'ΔG_{bin}. Per tant, creiem que seria bo provar de mutar aquestes dues posicions a la vegada.

Pel que fa a l'aplicació interactiva, hem vist que hi ha hagut algun repte a l'hora d'automatitzar certs càlculs i generalitzar-ne alguns, per això, encara que hem pogut generalitzar bastant per a acceptar dades distribuïdes d'altres formes, han de seguir un cert criteri per tal de poder utilitzar-ne l'aplicació. En definitiva creiem que pot ser una bona utilitat que pot ajudar a alguns investigadors en aquesta àrea i que faci servir models QSAR en la seva tasca d'extreure conclusions sobre les dades que tenen.

En el transcurs del projecte hem vist com era el procés de crear una aplicació per a un cert tipus de client específic, i ha sigut una bona experiència per aprendre sobre aquests temes que s'han tocat durant el projecte així com a esbrinar en certs moments què és el que vol exactament un client especialitzat.

5.2 Objectius complerts

Aquests són el objectius complerts durant projecte:

- Entendre les dades que tenim i saber analitzar-les - Complet
- Predir mutacions a partir de les correlacions amb l'ΔG_{bin} - Complet

- Aplicar models de regressió lineal (PLS) sobre les dades per tal de trobar relacions entre posicions - Complet
- Estudiar la viabilitat d'aplicar el model de regressió no lineal (KPLS) sobre les dades per tal de trobar relacions entre posicions - No complet

  Després d'estudiar el model de regressió no lineal (KPLS) [10], hem arribat a la conclusió que no era suficientment viable l'aplicació sobre el tipus de dades de les quals disposem. Això es deu al fet que aquest model funciona bé quan tenim poques observacions per a moltes variables, però en el nostre cas treballem només amb 9 variables, i disposem de centenars d'observacions. Per aquest fet, hem cregut que no seria necessari aplicar-ho i creiem que amb el PLS obtindrem millors resultats, ja que funciona millor amb moltes observacions per a no tantes variables.
- Crear una aplicació interactiva per a treballar amb les dades donades - Complet
- Generalitzar l'aplicació per a poder treballar amb dades similars - Complet

5.3 Treball futur

Un cop acabada l'aplicació, encara es pot treballar sobre aquesta per tal de millorar-la. Un dels possibles objectius futurs és incloure l'opció de treballar amb formats diferents del CSV, encara que avui en dia és fàcil convertir les dades d'un format a un altre.

Una altra millora a tenir en compte pot ser la possibilitat que l'usuari proposi una mutació en una posició i que l'aplicació, a partir de les dades, intenti predir si esdevindria en una disminució de l'ΔG_{bin}, un augment d'aquesta o no té suficients dades per a extreure'n conclusions.

Altres possibles àrees de treball futur poden ser estudiar casos diferents on les dades difereixin suficient per a no ser viable fer servir aquesta aplicació i intentar crear-ne una de més general que accepti altres tipus de models diferents del QSAR.

Glossari

ALA L'alanina és un dels aminoàcids que formen les proteïnes dels éssers vius. 14, 17

CSV Els fitxers CSV (de l'anglès *Comma-Separated Values*) són un tipus de document en format obert senzill per a representar dades en forma de taula, on les columnes van separades per comes (o punt i coma) i les files per salts de línia. Els camps que tenen comes o salts de línia, han d'anar entre cometes dobles. 14, 22, 28, 44

CV La validació creuada (*Cross-Validation*) és una tècnica utilitzada per avaluar els resultats d'una anàlisi estadística i garantir que són independents de la partició entre dades d'entrenament i prova. Consisteix a repetir i calcular la mitjana aritmètica obtinguda de les mesures d'avaluació sobre diferents particions. 24

GLU L'àcid glutàmic és un dels aminoàcids que formen les proteïnes dels éssers vius. 14

GLY La glicina és un dels aminoàcids que formen les proteïnes dels éssers vius. 17, 18

HTML L'HTML (acrònim d'*Hyper Text Markup Language*, en català, "llenguatge de marcat d'hipertext"), és un llenguatge de marcat que deriva de l'SGML dissenyat per estructurar textos i relacionar-los en forma d'hipertext. Gràcies a Internet i als navegadors web, s'ha convertit en un dels formats més populars que existeixen per a la construcció de documents per a la web.. 26

KPLS El Kernel PLS consisteix en kernelitzar un model PLS. O sigui, el que s'intenta és obtenir un model de regressió no lineal a partir d'una transformació no lineal sobre les variables lineals que tenim al PLS. 6, 9–12, 20, 24, 30, 31, 34, 44

LEU La leucina és un dels aminoàcids que formen les proteïnes dels éssers vius. 14

LOO La validació creuada deixant-ne un fora (*leave-one-out*, també coneguda com a LOOCV de *leave-one-out cross-validation*) implica separar les dades de manera que per a cada iteració tinguem una sola mostra per a les dades de prova i tota la resta conformant les dades d'entrenament. L'avaluació ve donada per l'error, i en aquest tipus de validació creuada l'error és molt baix, però en canvi, a escala computacional és molt costós, ja que s'han de realitzar un elevat nombre d'iteracions, tantes com N mostres tinguem i per a cada una analitzar les dades tant d'entrenament com de prova. 24

MHC El complex d'histocompatibilitat principal (*Major Histocompatibility Complex*) és una família de gens ubicats en el braç curt del cromosoma 6 els productes del qual estan implicats en la presentació d'antígens als limfòcits T i en la diferenciació del propi i l'aliè en el sistema immunitari. 5–8, 14

MMGBSA . 8

NIPALS En estadística, l'algoritme iteratiu no lineal de mínims quadrats parcials (*non-linear iterative partial least squares*) permet computar els primers components d'un anàlisi de components principals (PCA) o d'un anàlisi PLS. Per a conjunts de dades molt multi-dimensionals, normalment només és necessari computar una part petita dels primers components principals. Aquest algoritme calcula t_i i p'_1 d'$\mathbf{X}$. El producte exterior $t_i p'_1$ es pot extreure d'$\mathbf{X}$ deixant la matriu residual $\mathbf{E}_1$ fora. Això pot ser usat per a calcular components principals subsegüents. Això redueix substancialment el temps computacional ja que s'evita calcular la matriu de covariances. 19, 20, 24

OLS En estadística, la regressió de mínims quadrats ordinària (*ordinary least squares*) o de mínims quadrats lineal és un mètode per a estimar els paràmetres desconeguts en un model de regressió lineal, amb el repte de minimitzar les diferències entre les respostes observades en un conjunt de dades arbitrari i les respostes predites per la aproximació lineal de les dades. L'estimador resultant es pot expressar com una fórmula simple, especialment en el cas d'un únic regressor en el costat dret. 19, 23

PCR En estadística, la regressió principal per components (*Principal Components Regression*) és una tècnica d'anàlisi de la regressió que es basa en l'anàlisi de components principals (PCA). Consisteix generalment a considerar una regressió dels resultats (també coneguda com la resposta o, la variable dependent) en un conjunt de covariables (també conegudes com a predictors, variables explicatives o variables independents) basat en un model de regressió lineal estàndard, però que utilitza PCA per l'estimació dels coeficients de regressió desconeguts en el model. 23

PLS La Regressió de mínims quadrats parcials (*Partial Least-Squares*) és un mètode estadístic per trobar un model de regressió lineal sobre 2 matrius. Mitjançant la projecció de les variables de projecció i les variables observables sobre un nou pla. 1, 6, 9–12, 19–21, 23, 24, 30, 31, 34, 44–46

QSAR La relació estructura-activitat quantitativa (*Quantitative structure–activity relationship*) és el procés pel qual una estructura química es correlaciona quantitativament (mitjançant mètodes matemàtics) amb un procés ben definit, com pot ser l'activitat biològica (unió d'un fàrmac amb un receptor) o la reactivitat química (afinitat d'una substància a una altra perquè una reacció es produeixi). En tot cas, s'ha de tindre sempre present l'estructura química no només del fàrmac o de la substància química sinó també del receptor o substància *diana*. 6, 10, 11, 24, 30, 31, 34, 43, 44

THR La treonina és un dels aminoàcids que formen les proteïnes dels éssers vius. 14, 17, 18

Bibliografia

[1] P. Lasso, C. Cárdenas, F. Guzmán, F. Rosas, M. del Carmen Thomas, M. C. López, J. M. González, A. Cuéllar, J. M. Campanera, F. J. Luque i C. J. Puerta. "Effect of secondary anchor amino acids substitutions on the immunogenic properties of an HLA-A*0201-restricted T cell epitope derived from the *Trypanosoma cruzi* KMP-11 protein".

[2] R. A. Storino i J. Milei. *Enfermedad de chagas*. Doyma Argentina, Division Mosby, 1994.

[3] R. Pouplana i J. M. Campanera. "Energetic contributions of residues to the formation of early amyloid-β oligomers." eng. A: *Physical chemistry chemical physics : PCCP* 17.4 (2015), pàgines 2823 - 2837. ISSN: 1463-9084. DOI: 10.1039/c4cp04544k. URL: http://www.ncbi.nlm.nih.gov/pubmed/25503571.

[4] H. Cloke, F Pappenberger i J.-P. Renaud. "Multi-method global sensitivity analysis (MMGSA) for modelling floodplain hydrological processes". A: *Hydrological processes* 22.11 (2008), pàgines 1660 - 1674.

[5] D. Case, T. Darden, T. E. Cheatham III, C. Simmerling, J Wang, R. Duke, R Luo, R. Walker, W Zhang, K. Merz et al. "AMBER 12". A: *University of California, San Francisco* 1.3 (2012).

[6] J. M. Campanera i R. Pouplana. "MMPBSA Decomposition of the Binding Energy throughout a Molecular Dynamics Simulation of Amyloid-Beta (Aß10-35) Aggregation". A: *Molecules* 15.4 (2010), pàgina 2730. ISSN: 1420-3049. DOI: 10.3390/molecules15042730. URL: http://www.mdpi.com/1420-3049/15/4/2730.

[7] *Quantitive Structure - Activity Relationship*. URL: http://en.wikipedia.org/wiki/Quantitative_structure%E2%80%93activity_relationship (consultat 15 de març de 2015).

[8] *QSARdata: Quantitative Structure Activity Relationship (QSAR) Data Sets*. 2013. URL: http://cran.r-project.org/web/packages/QSARdata/index.html (consultat 14 de març de 2015).

[9] B.-H. Mevik i R. Wehrens. "The pls package: principal component and partial least squares regression in R". A: *Journal of Statistical Software* 18.2 (2007), pàgines 1 - 24.

[10] R. Rosipal i L. J. Trejo. "Kernel Partial Least Squares Regression in Reproducing Kernel Hilbert Space". A: *J. Mach. Learn. Res.* 2 (març de 2002), pàgines 97 - 123. ISSN: 1532-4435. URL: http://dl.acm.org/citation.cfm?id=944790.944806.

[11] R. Rosipal, L. J. Trejo i B. Matthews. "Kernel PLS-SVC for linear and nonlinear classification". A: *ICML*. 2003, pàgines 640 - 647.

[12] Partial correlation coefficient. *Encyclopedia of Mathematics*. URL: `http://www.encyclopediaofmath.org/index.php?title=Partial_correlation_coefficient&oldid=24254` (consultat 15 de juny de 2015).

[13] B. Dayal, J. F. MacGregor et al. "Improved PLS algorithms". A: *Journal of chemometrics* 11.1 (1997), pàgines 73 - 85.