Desenvolupament d'un aplicatiu web per la correcta classificació de variants genètiques en els gens causants de les RASopaties.

Elisabeth Castellanos Pérez
Màster en Bioinformàtica i Bioestadística
Àrea 3 - Subàrea 5: Epigenòmica y càncer

Directora TFM: Dra. Izaskun Mayona
Professor/a responsable: Ferran Prados

24/06/2020

FITXA DEL TREBALL FINAL

Títol del treball:	*Desenvolupament d'un aplicatiu web per la correcta classificació de variants genètiques en els gens causants de les RASopaties*
Nom de l'autor:	*Elisabeth Castellanos Pérez*
Nom del consultor/a:	*Dra. Izaskun Mayona González*
Nom del PRA:	*Dr. Ferran Prados Carrasco*
Data de lliurament (mm/aaaa):	*06/2020*
Titulació o programa:	*Màster en Bioinformàtica i Bioestadística*
Àrea del Treball Final:	*Àrea 3 - Subàrea 5: Epigenòmica y càncer*
Idioma del treball:	*Català*
Paraules clau	*RASopathies, variant classification, Shiny*

Resum del Treball (màxim 250 paraules): *Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball*

Finalitat: El diagnòstic genètic té com a finalitat determinar la causant genètica del desenvolupament de malalties hereditàries, com les RASopaties. L'estudi d'alteracions genètiques es realitza mitjançant NGS i aquestes es classifiquen segons el seu efecte deleteri. En aquest projecte ens proposem generar un aplicatiu per automatitzar aquest procés de classificació en els gens de les Rasopaties seguint el sistema de puntuació establert a nivell internacional.

Metodologia: Mitjançant R-Shiny, s'ha generat un GUI per automatitzar la classificació de les variants genètiques. La informació necessària s'extreu d'una base de dades pròpia (Pandora) i de la literatura.

Resultats: s'ha evolucionat un R-script desenvolupat prèviament en el grup per tal de que no contingui fragments de codi específics de cada gen ni les credencials per accedir a Pandora. Posteriorment, aquest s'ha modificat per tal de ser compatible amb R-shiny i s'ha generat un GUI a nivell local. Aquest GUI demana que s'identifiqui la variant a classificar i que s'introdueixin les dades que no es poden extraure de Pandora. Com a resultats, l'app genera una taula resum de la variant a classificar, els criteris que compleix la variant per tal de ser classificada i, la classificació final. En total, van ser avaluades 20 variants prèviament classificades manualment. També s'ha intentat transformar aquest GUI per un GUI localitzat en un servidor del centre.

Conclusió: El GUI a nivell local és de gran utilitat per classificar les variants en els gens de les RASopaties. Aquest GUI ha estat validat comparant la classificació manual i la automàtica però encara no és funcional en format server.

Abstract (in English, 250 words or less):

Aim: The purpose of genetic diagnosis is to determine the genetic cause of the development of hereditary diseases, such as RASopathies. The study of genetic alterations is performed using NGS and these are classified according to their deleterious effect. In this project we propose to generate an application to automate this classification process in Rasopathies-related genes following the guidelines established internationally.

Methodology: Using R-Shiny, a GUI has been generated to automate the classification of genetic variants. The necessary information is extracted from our own database (Pandora) and from the literature.

Results: An R-script previously developed in the group has evolved so that it does not contain snippets of gene specific code or the credentials to access Pandora. This was later modified to be R-shiny compatible and a GUI was generated locally. This GUI requires theu user indicates the variant to be classified and include all data that cannot be extracted from Pandora. As a result, the app generates a summary table of the variant to be classified, the criteria that the variant meets in order to be classified and the final classification. In total, 20 previously manually classified variants were evaluated. Attempts have also been made to transform this GUI into a server GUI.

Conclusion: The local GUI is very useful for classifying variants in the RASopathies-related genes. This GUI has been validated comparing manual and automatic classification but is not yet functional in server format.

Índex

Llista de figures

1. Introducció

1.1 Context i justificació del Treball

1.1.1. La ultraseqüenciació (NGS) i el diagnòstic genètic

El diagnòstic genètic és un camp en expansió des de principis de segle, i sobretot en la darrera dècada degut al desenvolupament de noves eines de seqüenciació, també conegudes com a *Next Generation Sequencing* (NGS) o ultraseqüènciació. El diagnòstic genètic és molt rellevant quan parlem de malalties hereditàries. Aquest grup de malalties són causades per mutacions en gens concrets, les quals es poden transmetre de pares a fills. Per tant, la detecció d'aquesta mutació causant d'una malaltia mitjançant un diagnòstic genètic és clau per, en la gran majoria de casos, confirmar la sospita clínica del pacient, determinar el maneig i tractament del pacient, detectar altres membres de la família portadors de la mutació i poder establir un seguiment adequat per ells, i per últim, poder oferir a la família una planificació familiar per tal d'evitar la transmissió d'aquesta mutació en futures generacions[1].

A partir dels anys 90, que es quan es van començar a identificar els gens causants de moltes de les malalties hereditàries monogèniques, aquests diagnòstics es feien mitjançant seqüenciació Sanger on s'analitzaven hot spots o exons dels gens causants de manera seqüencial, és a dir, s'analitzava una regió o exó i si el resultat era indeterminat perquè no s'havia trobat cap alteració, s'analitzava un altre fragment codificant del gen. Quan la malaltia presentava heterogeneïtat genètica, és a dir, que podia estar causada per més d'un gen alterat, primer s'analitzava un gen candidat i, si no es trobava cap alteració, s'analitzava un altre[2]. Però amb el desenvolupament de la NGS, les aproximacions per realitzar els diagnòstics genètics ha canviat radicalment. Actualment la NGS permet seqüenciar diversos gens per múltiples pacients mitjançant el marcatge o *barcoding* del àcid desoxiribonucleic (ADN) procedent de cada pacient i mapant el resultat contra un genoma de referència que contingui, com a mínim, tots els gens seqüenciats alhora, per un preu molt ajustat[3].

1.1.2. Anàlisi i interpretació de les dades obtingudes mitjançant NGS

La implementació de la NGS en el diagnòstic genètic ha comportat moltes millores com hem comentat en el paràgraf anterior, però, precisament degut a la reducció del cost de seqüenciació, l'increment de la demanda de la societat en realitzar aquests diagnòstics i el coneixement de nous gens implicats en determinades malalties ha induït a un augment de diagnòstics en pacients amb clíniques incompletes i la anàlisis de gens poc coneguts fins ara. Aquest fet ha comportat un augment en la detecció de variants de significat incert (VSI, o VUS en anglès (*Variant of Unknown Significance*)), on la seva relació amb una patologia concreta no es pot confirmar ni descartar [4]. És per això que la comunitat científica (The American College of

Medical Genetics and Genomics and the Association for Molecular Pathology (ACMG-AMP)) ha fet un esforç per consensuar uns criteris i establir unes guies clares i sistemàtiques per interpretar la patogenicitat de totes les variants detectades a partir de (1) la freqüència poblacional de la variant detectada, (2) del tipus de mutació, (3) del resultat dels predictors *in silico* sobre la seva patogenicitat, (4) del domini on es troba la variant, (5) si ha estat reportada anteriorment en pacients amb la mateixa patologia, (6) si és una variant detectada co-segrega amb la malaltia en la família i (7) del resultat d'estudis funcionals sistemàtics[5,6] (Figura 1).

	Benign		Pathogenic			
	Strong	**Supporting**	**Supporting**	**Moderate**	**Strong**	**Very strong**
Population data	MAF is too high for disorder BA1/BS1 **OR** observation in controls inconsistent with disease penetrance BS2			Absent in population databases PM2	Prevalence in affecteds statistically increased over controls PS4	
Computational and predictive data		Multiple lines of computational evidence suggest no impact on gene /gene product BP4 Missense in gene where only truncating cause disease BP1 Silent variant with non predicted splice impact BP7 In-frame indels in repeat w/out known function BP3	Multiple lines of computational evidence support a deleterious effect on the gene /gene product PP3	Novel missense change at an amino acid residue where a different pathogenic missense change has been seen before PM5 Protein length changing variant PM4	Same amino acid change as an established pathogenic variant PS1	Predicted null variant in a gene where LOF is a known mechanism of disease PVS1
Functional data	Well-established functional studies show no deleterious effect BS3		Missense in gene with low rate of benign missense variants and path. missenses common PP2	Mutational hot spot or well-studied functional domain without benign variation PM1	Well-established functional studies show a deleterious effect PS3	
Segregation data	Nonsegregation with disease BS4		Cosegregation with disease in multiple affected family members PP1	Increased segregation data →		
De novo data				De novo (without paternity & maternity confirmed) PM6	De novo (paternity and maternity confirmed) PS2	
Allelic data		Observed in *trans* with a dominant variant BP2 Observed in *cis* with a pathogenic variant BP2		For recessive disorders, detected in trans with a pathogenic variant PM3		
Other database		Reputable source w/out shared data = benign BP6	Reputable source = pathogenic PP5			
Other data		Found in case with an alternate cause BP5	Patient's phenotype or FH highly specific for gene PP4			

Figura 1: Marc d'evidències per classificar les variants detectades. Aquest gràfic organitza cadascun dels criteris segons el tipus d'evidència, així com la força dels criteris per a una afirmació benigna (costat esquerre) o patogènica (costat dret)[6].

En funció de quines evidències presenti cada variant, aquestes guies estableixen un sistema de puntuació per classificar les variants en 1) Variants Benignes 2) Variants Probablement Benignes, 3) Variants de Significat Incert, 4) Variants Probablement Patogèniques i 5) Variants Patogèniques (Taula 1). Tot i així, aquestes guies tenen un caràcter general i el que s'està desenvolupant actualment són adaptacions d'aquestes guies per cadascun dels grups de malalties hereditàries on es contemplen les particularitats dels gens afectats, els tipus de

mutacions, els estudis funcionals acceptats i la freqüència poblacional i de pacients amb la mateixa presentació clínica acceptada.

Pathogenic	(i) 1 Very strong (PVS1) *AND* (a) ≥1 Strong (PS1–PS4) *OR* (b) ≥2 Moderate (PM1–PM6) *OR* (c) 1 Moderate (PM1–PM6) and 1 supporting (PP1–PP5) *OR* (d) ≥2 Supporting (PP1–PP5) (ii) ≥2 Strong (PS1–PS4) *OR* (iii) 1 Strong (PS1–PS4) *AND* (a)≥3 Moderate (PM1–PM6) *OR* (b)2 Moderate (PM1–PM6) *AND* ≥2 Supporting (PP1–PP5) *OR* (c)1 Moderate (PM1–PM6) *AND* ≥4 supporting (PP1–PP5)
Uncertain significance	(i) Other criteria shown above are not met OR (ii) the criteria for benign and pathogenic are contradictory
Likely pathogenic	(i) 1 Very strong (PVS1) *AND* 1 moderate (PM1–PM6) *OR* (ii) 1 Strong (PS1–PS4) *AND* 1–2 moderate (PM1–PM6) *OR* (iii) 1 Strong (PS1–PS4) *AND* ≥2 supporting (PP1–PP5) *OR* (iv) ≥3 Moderate (PM1–PM6) *OR* (v) 2 Moderate (PM1–PM6) *AND* ≥2 supporting (PP1–PP5) *OR* (vi) 1 Moderate (PM1–PM6) *AND* ≥4 supporting (PP1–PP5)
Benign	(i) 1 Stand-alone (BA1) *OR* (ii) ≥2 Strong (BS1–BS4)
Likely benign	(i) 1 Strong (BS1–BS4) and 1 supporting (BP1–BP7) *OR* (ii) ≥2 Supporting (BP1–BP7)

Taula 1: Criteri de puntuació per classificar les variants detectades en funció de les evidències de patogenicitat o neutralitat [6].

1.1.3. Les RASopaties

Un exemple de la necessitat d'aquestes guies més específiques són les RASopaties. Aquest grup de malalties són relativament freqüents (1/2000 nounats) i estan causades per la desregulació dels gens de la via de les Ras/mitogen-activated protein kinasa (RAS/MAPK). Les RASopaties engloben diferents síndromes, com la Neurofibromatosis tipus 1 (NF1), la síndrome de Noonan (NS), la síndrome de Noonan amb lèntigs (NSML), la síndrome de Costello (CS), la síndrome de Legius (LS), la síndrome cardio-facio-cutani (CFC) i malformacions capil·lars i arteriovenoses (CM-AVM). Malgrat que cada síndrome té les seves peculiaritats fenotípiques, totes elles presenten un cert solapament clínic, sobretot en edat pediàtriques, segurament degut a què afecten la mateixa via de senyalització [7].

La via RAS/MAPK s'activa per factors de creixement extracel·lulars i per tant, un correcte funcionament és transcendental pel correcte desenvolupament. Degut al solapament molecular, mutacions en diferents gens poden causar un síndrome determinat, però també certs gens poden ser causant de més d'un síndrome (Figura 2). Un clar exemple és la síndrome de Noonan (NS) que pot estar causat per més de 15 gens, entre ells *PTPN11* (SHP2 en la Figura 2), mentre que aquest mateix gen pot ser causant de la síndrome de Noonan amb lèntigs (NSML)[8]. Per tant, aquest grup de malalties es caracteritzen per presentar heterogeneïtat genètica i, degut a la funció reguladora de molts d'aquest gens, les mutacions que presenten la majoria dels gens són canvis a nivell d'aminoàcid (missense) que indueixen un de guany de funció (Gain of Function, GOF) i no pas mutacions truncants, la qual cosa dificulta la

interpretació de la seva patogenicitat [9].

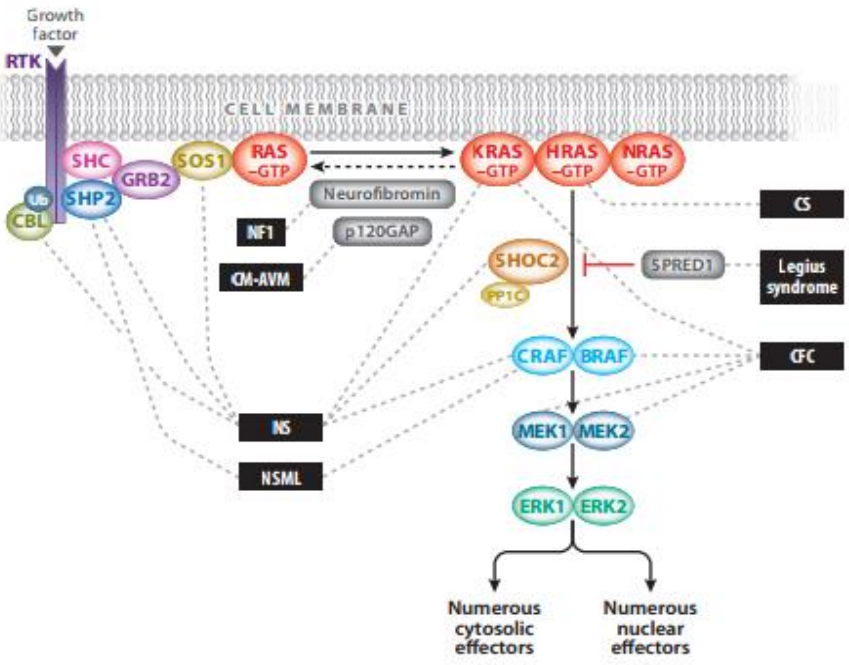


Figura 2: Esquema simplificat de la via de RAS/MAPK i patologies hereditàries associades.

1.1.4. Interpretació de les variants genètiques en els gens de les RASopaties

Per aquesta raó, recentment el *ClinGen RASopathies Expert Panel* ha desenvolupat les guies específiques per la classificació de variants en aquests grups de gens[10,11], la qual cosa ha sigut un avenç molt significatiu per aquest grup de patologies. En concret, a diferència de les guies generals, especifica:

- Una puntuació concreta en funció del nombre de pacients reportats amb la mateixa variant, és a dir, no es considera evidències de patogenicitat equivalents si una variant ha estat reportada 2 cops o 5 en pacients diagnosticats d'una RASopatia.

- De la mateixa manera, estableix una puntuació diferent en funció del nombre de casos *de novo* reportats o de famílies estudiades on la variant co-segrega amb el fenotip.

- Estableix clarament quines regions o dominis funcionals per cada gen es poden considerar alhora de realitzar els anàlisis, així com *hot spots* per cada un dels gens.

- Determina els criteris per establir quins estudis funcionals són vàlids alhora d'acceptar una evidència de patogenicitat o neutralitat en funció del gen i del domini de cadascun d'ells.

L'implementació d'aquestes guies de manera manual, però, és força tediós i costós en quant a temps es refereix i requereix d'una persona amb uns nivells determinats per poder fer un

correcte ús de les guies i fer una classificació correcta. En concret, per cada variant es necessita (1) que es determini on es localitza aquesta dins de la proteïna i comprovar si es troba en un domini o *hot spot* conegut, (2) realitzar un anàlisi de predictors in silico per determinar si l'aminoàcid alterat és conservat, si podria alterar el plegament de la proteïna, si podria alterar l'splicing, etc [...], (3) buscar les freqüències poblacionals en persones sanes i en pacients amb clínica compatible amb una RASopatia, (4) cercar si aquesta variant ja ha estat descrita anteriorment en alguna família, si aquesta co-segrega amb la malaltia o si s'ha determinat si la variant és de novo en els casos esporàdics, i finalment (5) si s'ha analitzat l'efecte de la variant concreta sobre la funció del gen alterat i si aquest estudi funcional és d'una qualitat determinada per acceptar els resultats obtinguts com a evidència de patogenicitat o neutralitat. És per aquesta raó que es necessita d'algun sistema semi-automatitzat per facilitar l'ús d'aquestes guies i evitar errors humans en la seva implementació.

1.1.5. Sistemes semi-automatitzats per la Interpretació de variants genètiques.

Com s'ha comentat, la creixent quantitat de dades genètiques generades dificulta molt la interpretació clínica de les variants. Per poder classificar una variant s'han de revisar tots els criteris manualment. Això comporta una gran quantitat de temps i dedicació. Si bé és cert que alguns criteris són impossibles d'automatitzar com per exemple la determinació de si la mutació es *de novo* o el resultat d'un estudi funcional, d'altres, s'extreuen de diferents bases de dades disponibles i podrien ser obtinguts de manera automàtica. De fet, existeixen programes, com Intervar[12] o Sherloc[13] que ja han abordat aquesta problemàtica, però la seva classificació no és específica per les RASopaties.

1.1.6. El diagnòstic genètic a la Unitat de Genòmica Clínica de l'Hospital Germans Trias i Pujol:

La Unitat de Genòmica Clínica de l'Hospital Germans Trias i Pujol està especialitzada en l'anàlisi de les RASopaties, entre altres patologies. Aquesta unitat comparteix un sistema d'anàlisi i base de dades relacional anomenada "Pandora" amb la Unitat de Diagnòstic Genètic de Càncer Hereditari de l'Institut Català d'Oncologia. Aquest sistema s'utilitza per a la gestió de les mostres, experiments de seqüenciació (runs), execucions d'anàlisis bioinformàtics prèviament validats dins la unitat, filtratge i anotació, i finalment registre de la classificació de cada variant detectada.

En aquesta Unitat ja s'ha realitzat una primera aproximació per semi-automatitzar la classificació de les variants detectades en els diferents anàlisis mitjançant un script en R. Aquest script extreu les dades de Pandora i permet puntuar automàticament, aquells criteris que es basen en freqüència poblacional, dominis on es troba la variant i el resultat de predictors *in silico*, per tal d'interrogar cadascuna de les variants individualment. Tot i així, aquesta aproximació presenta algunes dificultats pel personal de laboratori no avesat a treballar mitjançant terminal i presenta algunes mancances que s'haurien de millorar per optimitzar el

seu ús, com seria la incorporació dels criteris no automatitzables (*de novo*, co-segregació, nombre de pacients ja descrits amb la mateixa variant, etc [...]) a partir de preguntes clares. Per aquest motiu, ens proposem de millorar un *script* inicial ja desenvolupat a la Unitat de Genòmica clínica el qual per millorar la incorporació de criteris no-automatitzables, i de crear un GUI mitjançant *Shiny* per tal de facilitar l´ús d'aquest sistema de classificació de variants per les RASopaties a tot el personal de la Unitat de Genòmica Clínica.

1.2 Objectius del Treball

1. Implementació d'un sistema de classificació de variants per als gens de les RASopaties
    1.1. Anàlisi exhaustiu de l'script disponible i determinació de processos millorables o absents.
    1.2. Desenvolupament i validació de les millores detectades

2. Desenvolupament d'un aplicatiu web per facilitar l'ús del sistema de classificació pels diferents membres del laboratori
    2.1. Disseny del GUI mitjançant Shiny a nivell local
    2.2. Validació del sistema de classificació via web (local) amb un set de variants prèviament conegudes.
    2.3. Implementació del GUI en el servidor Shiny del centre

1.3 Enfocament i mètode seguit

El treball es podria enfocar des de dues vessants. D'una banda, existeixen programes i aplicatius web, com Intervar o Sherloc que ja han abordat aquesta problemàtica i per tant, una opció hauria pogut ser modificar els seus algoritmes per tal d'adaptar aquests criteris. Ara bé, la seva classificació no és específica per les RASopaties i presenta algunes dificultats com per exemple, introduir la informació de si la variant es troba en un domini rellevant en un gen homòleg, i també presenta divergències respecte les diferents bases de dades d'on s'extreu la informació poblacional, fet que fa que no sigui idònia pels objectius que es volen assolir.

L'altra opció és la de construir aquest aplicatiu web (GUI) a partir de l'script R disponible al laboratori i que, en general, permet classificar automàticament les variants amb la informació extreta de la base de dades relacional que disposa la Unitat de Genòmica Clínica de l'Hospital Germans Trias i Pujol. En concret, creiem que el desenvolupament d'aquest GUI mitjançant R-shiny és una bona aproximació i assequible durant el desenvolupament del TFM. Creiem que aquesta és la opció més adient per les peculiaritats dels gens a analitzar i la millora que

suposaria tenir un aplicatiu fet a mida i que recollís les necessitats de les persones que el faran servir.

1.4 Planificació del Treball

A continuació es detalla la llista de tasques planificades a l'iniciar el TFM:

- Lectura de bibliografia relacionada amb la classificació de variants, com desenvolupar un script ben estructurat i sense "hardcode", com dissenyar i implementar un GUI Shiny a nivell local i en servidor.
- Establir les diferents conneccions amb les bases de dades a consultar (UCSC, Pandora, …)
- Establir un criteri per millorar l'script
- Familiarització amb les taules PostGreSQL de Pandora
- Millorar les consultes a Pandora amb RPostGreSQL per extreure totes les dades que necessitem d'una manera endreçada i només una consulta per mostra (del 23/03 al 22/04)
- Extreure dades d'altres bases de dades disponibles. (del 23/03 al 22/04)
- Incorporar les noves funcions a l'algoritme pre-establert i millorar el seu format i estructura. (del 23/03 al 22/04)
- Incloure tot l'script en un repositori públic com pot ser GitHub (del 23/03 al 22/04)
- Disseny i desenvolupament del servidor Shiny a nivell local per classificar les variants a partir de l'script millorat. (del 23/04 al 18/05)
- Implementació del GUI en el servidor Shiny del centre.(del 23/04 al 18/05)
- Escriure la memòria (del 19/05 al 10/06)
- Elaboració de la presentació (11/06 – 14/06)

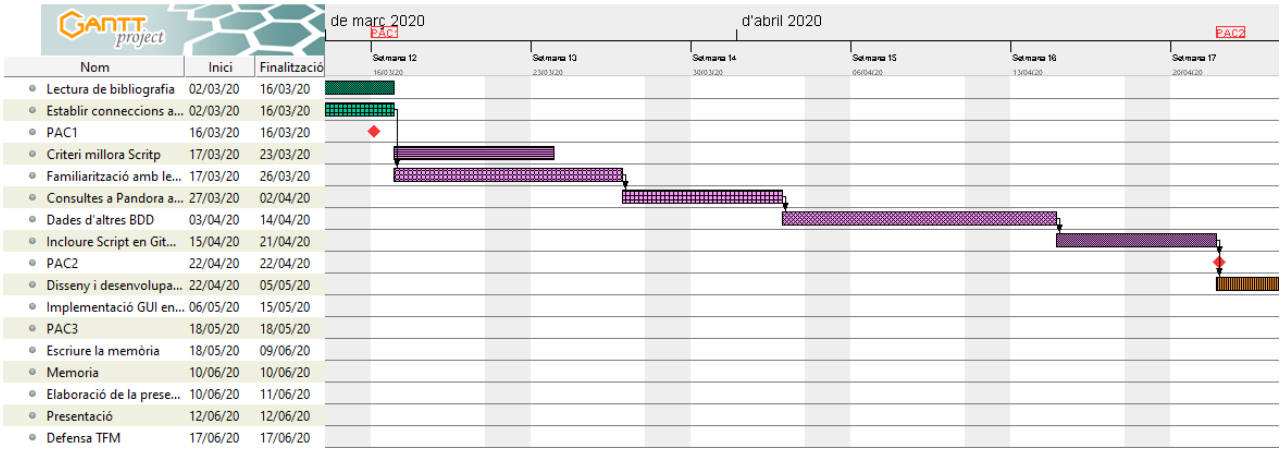


Figura 3A: Diagrama de Gantt esquematitzant la planificació inicials del TFM.

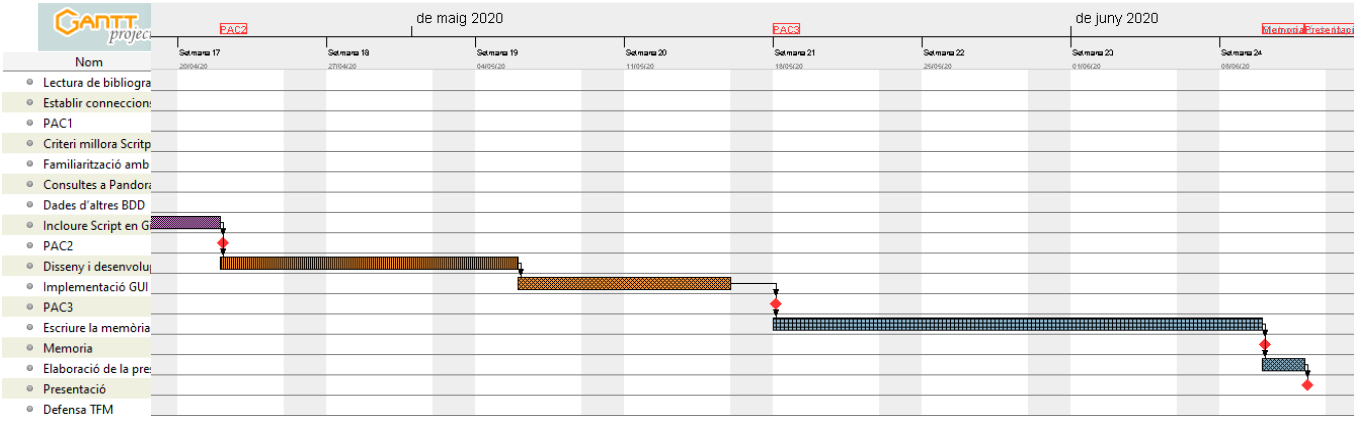


Figura 3B: Diagrama de Gantt esquematitzant la planificació inicials del TFM.

1.5 Breu sumari de productes obtinguts

En aquest treball de final de Màster hem desenvolupat un aplicatiu web utilitzant R-Shiny a nivell local per tal de classificar les variants detectades en els gens causants de les Rasopaties en funció del seu efecte deleteri. Aquest aplicatiu, o GUI, s'ha basat en un script previ disponible en el laboratori, el qual s'ha hagut de modificar, primer per eliminar part del codi específic de cada gen analitzat, fragments de hard-code, així com la part de codi per entrar a la base de dades privada del laboratori.

Posteriorment s'ha hagut de modificar l'script per fer-lo compatible amb R-shiny, endreçant les funcions, fent un disseny agradable i validant el funcionament del GUI a nivell local comparant el seu funcionament en vint variants prèviament classificades manualment.

Finalment, s'ha intentat fer unes lleugeres modificacions per tal d'incloure aquest GUI en el server del laboratori on es farà servir aquest GUI però aquest funciona parcialment, ja que no s'ha pogut trobar la causa de perquè dues de les funcions han deixat de funcionar. Segurament aquest punt s'hagués pogut millorar si s'hagués fet un *testing* dels diferents scripts, ja que hauríem detectat en quin punt apareix l'error.

1.6 Breu descripció dels altres capítols de la memòria

En els següents capítols es descriu la base de dades privada d'on el GUI dissenyat extreu les dades, la diferent metodologia utilitzada, així com els resultats obtinguts en les diferents etapes del treball: millora de l'script existent, desenvolupament del GUI local, validació d'aquest i desenvolupament del GUI per incloure'l en el servidor del laboratori.

2. Resta de capítols

2.1. Materials i mètodes

2.1.1. Base de dades relacional (Pandora)

La majoria de la informació obtinguda de cada variant per la seva correcta classificació s'extreu de la base de dades relacional compartida entre la Unitat de Genòmica i l'ICO (Pandora). Aquesta base de dades s'utilitza per a la gestió de les mostres, runs, execucions, filtratge i classificació de variants. Aquest sistema s'utilitza per a la gestió de les mostres, experiments de seqüenciació (runs), execucions d'anàlisis bioinformàtics prèviament validats dins la unitat, filtratge i anotació, i finalment registre de la classificació de cada variant detectada. Pandora emmagatzema informació genètica (coordenades genòmiques on es troba cadascuna de les variants detectades), informació poblacional (freqüència màxima, freqüència descrita en ExAc, 1000G, esp6500, i la freqüència in house calculada a partir d'estudis previs analitzats a Pandora), informació de diversos predictors in silico, entre altres.

Pandora és una base de dades relacional feta amb PostgreSQL [14], que conté diferents taules amb informació sobre les variants que es troben en els estudis realitzats en els dos centres. Els objectius principals de la base són automatitzar i independitzar així com facilitar la feina als professionals. Al ser una base pròpia, està adaptada a les necessitats de les unitats que la usen i això en propicia la millora continua. A més a més, permet que tota la informació es registri de forma centralitzada.

Alguns exemples de taules són: "VA_VariantsInTranscripts" , "VA_Frequencies", i "VA_InSilicoPathogenicity". És a dir, Pandora emmagatzema informació genètica (com en quin cromosoma es troba la variant i les coordenades), poblacional (per exemple ens determina la màxima freqüència i la freqüència trobada en els estudis penjats a Pandora), *in silico* (mostra la puntuació d'alguns predictors com Polyphen-2), d'entre d'altra. En l'annex 1 del treball es pot trobar una llista detallada de totes les taules que té i quines variables hi ha en cadascuna d'elles.

A més a més, també s'utilitza de la base de dades de UCSC [15] Genome Browser la taula *rmsk*, del *RepeatMasker*. La taula conté informació (coordenades, cadena..) de seqüènies d'ADN de repeticions intercalades i de baixa complexitat, que seran requerides per elaborar l'automatització d'alguns dels criteris.

2.1.2. Modificació de l'spcript R disponible

En aquesta Unitat ja s'ha realitzat una primera aproximació per semi-automatitzar la classificació de les variants detectades mitjançant un script en R. Aquest script extreu les dades de Pandora i permet puntuar automàticament, aquells criteris que es basen en freqüència poblacional i el resultat de predictors *in silico*, per tal d'interrogar cadascuna de les variants individualment. Tot i així, presenta algunes mancances que s'havien de millorar per optimitzar el seu ús, com són la incorporació dels criteris no automatitzables (*de novo*, co-segregació, nombre de pacients ja descrits amb la mateixa variant, etc [...]) a partir de preguntes clares.

Per dur a terme el treball, s'empra el llenguatge R que és un entorn de programari lliure per a l'estadística informàtica[16,17]. Les consultes a la base de dades Pandora es fan des de *RPostgreSQL*[18]. També s'utilitza el paquet *stringr* per manipular cadenes de text[19] i el paquet *gmodels* per fer taules creuades [20].

Les modificacions de l'script han sigut principalment estètiques per endreçar-lo i fer-lo més easy-reading en llenguatge R. S'han documentat els inputs i outputs, i s'ha eliminat el codi repetitiu, per tal de fer-lo més endreçat visualment. També s'ha extret les parts de codi que eren gen-depenent i aquestes s'han incorporat en el script mitjançant la lectura d'un fitxer extern.

A més a més, s'han incorporat noves funcions mitjançant R per fer les cerques *in house* de manera que no tingui en compte mostres analitzades més d'una vegada, l'avaluació conjunta de variants que tinguin tant evidències neutrals com patogèniques, així com la incorporació de funcions per tal de poder puntuar evidències no automatizables, com la co-segregació o casos *de novo*, casos reportats prèviament i estudis funcionals. Cadascuna de les modificacions realitzades s'ha testat individualment per determinar si realitzava la funció esperada i si alterava l'script en global.

2.1.3. Desenvolupament GUI

El desenvolupament del GUI s'ha realitzat mitjançant R/shiny[21] per tal de poder cridar l'script i incloure els paràmetres manuals mitjançant un sistema més visual com és una pàgina web i no pas via terminal.

2.1.4. Validació GUI a nivell local

Independentment dels diversos tests realitzats durant la modificació de l'scprit R inicial i el desenvolupament del GUI a nivell local i en servidor, s'ha comprovat que 20 variants prèviament classificades manualment seguint les recomanacions del *ClinGen RASopathies Expert Panel* són classificades de la mateixa manera amb l'script i GUI finals.

Per tant, no només s'ha validat la classificació final, sinó també que els paràmetres utilitzats per la classificació eren els mateixos i es puntuaven de la mateixa manera.

2.1.5. Documentació del GUI desenvolupat

Tota la documentació generada durant el treball, així com l'script original i posteriors versions s'han dipositat en GitHub [22] (https://github.com/ecastellanos82/RASo_variantsClassification. A més a més, també s'ha creat un arxiu README per tal d'indicar quin és l'objectiu de l'script.

2.2. Resultats obtinguts

2.2.1. R-script modificat

Degut a que la intenció era dipositar l'script inicial en un repositori públic per poder-lo compartir amb altres usuaris o desenvolupadors, així com per poder fer un registre de totes les modificacions efectuades sobre l'script inicial, primer de tot vam modificar l'script per evitar indicar els passwords i credencials de Pandora en el codi font. Aquestes credencials s'inclouen mitjançant uns fitxers externs que només disposa el personal que n'ha de fer ús, anomenats params.txt. En l'script del GUI es criden de la següent manera:

```
## connection to Pandora DB
get_db_parameters <- function(db_conf) {
  params <- read.table("~/GitHub/RASo_variantsClassification/params.csv",
                       sep = ",", stringsAsFactors = FALSE)
  return(list(user = params$V2[1],
              password = params$V2[2],
              dbname = params$V2[3],
              host = params$V2[4],
              port = params$V2[5]))
}

## connects to the NGS BD using a config file
## param db_conf: a full path to a config file (CSV)

db_connect_postgres <- function(db_conf) {
  drv <- dbDriver("PostgreSQL")

  db_conf <- get_db_parameters(db_conf)

  con <- dbConnect(drv,
                   user = db_conf[['user']],
                   password = db_conf[['password']],
                   dbname = db_conf[['dbname']],
                   host = db_conf[['host']],
                   port = db_conf[['port']])

  return(con)
}
```

Posteriorment, ho vam dipositar en GitHub on la tutora hi té accés, i tots els canvis de l'script els hem anat gravant i anotant per tal de que quedés constància mitjançant Git GUI de Windows. A més a més, també hem creat un arxiu README per tal d'indicar quin és l'objectiu de l'script.

Al revisar l'script original ens hem adonat que aquest no estava en un llenguatge universal, i el vam traduir a l'anglès. També vam modificar la informació gen-específica que estava introduïda dins del codi per tal de que aquesta es pogués introduir, i modificar si calia, mitjançant uns fitxers externs, també dipositats a GitHub.

```
##### Calling RASopathies gene information
domain_groupRAF <- read.csv("~/GitHub/RASo_variantsClassification/domini_grupRAF.csv")
domain_groupRAS <- read.csv("~/GitHub/RASo_variantsClassification/domini_grupRAS.csv")
domain_groupSOS <- read.csv("~/GitHub/RASo_variantsClassification/domini_grupSOS.csv")
domain_groupMAPK <- read.csv("~/GitHub/RASo_variantsClassification/domini_grupMAPK.csv")
Transcripts_RASos <- read.csv("~/GitHub/RASo_variantsClassification/Transcripts_RASos.csv")
```

Posteriorment, vam observar que no determinava correctament els criteris PM6_strong vs PS2 corresponents als casos de novo descrits en la literatura, ni tampoc incorporava el criteri PS4, que correspon a la presència d'assajos funcionals per a la variant a classificar. Aquests s'han incorporat de la següent manera:

```
###PS2 - de novo cases reported
criteria[c("PS2_veryStrong", "PS2"),1][denovo_confirmed>=2]<-c(1,0)
criteria[c("PS2_veryStrong", "PS2"),1][denovo_confirmed==1 & denovo_noconfirmed>=2]<-c(1,0)
criteria[c("PS2_veryStrong", "PS2"),1][denovo_confirmed==1 & denovo_noconfirmed<2]<-c(0,1)
criteria[c("PS2_veryStrong", "PS2"),1][denovo_confirmed==0]<-c(0,0)

###PM6  - de novo cases reported no confirmed

criteria[c("PM6_veryStrong","PM6_strong","PM6"),1][denovo_confirmed==0 & denovo_noconfirmed>3]<-c(1,0,0)
criteria[c("PM6_veryStrong","PM6_strong","PM6"),1][denovo_confirmed==0 & denovo_noconfirmed==3 | denovo_noconfirmed==2]<-c(0,1,0)
criteria[c("PM6_veryStrong","PM6_strong","PM6"),1][denovo_confirmed==0 & denovo_noconfirmed==1]<-c(0,0,1)
criteria[c("PM6_veryStrong","PM6_strong","PM6"),1][denovo_confirmed==0 & denovo_noconfirmed==0]<-c(0,0,0)

###PP1 - cosegregation cases reported

criteria[c("PP1_strong", "PP1_moderate", "PP1_supporting"),1][cosegregation>=7]<-c(1,0,0)
criteria[c("PP1_strong", "PP1_moderate", "PP1_supporting"),1][cosegregation==5|cosegregation==6]<-c(0,1,0)
criteria[c("PP1_strong", "PP1_moderate", "PP1_supporting"),1][cosegregation==3|cosegregation==4]<-c(0,0,1)
criteria[c("PP1_strong", "PP1_moderate", "PP1_supporting"),1][cosegregation<3]<-c(0,0,0)

###PS3 - Functional studies
criteria[c("PS3"),1]<-c(Functional_evidence[1])
```

2.2.2. GUI Local

Per tal de desenvolupar el GUI a nivell local, hem hagut de modificar el script per ser compatible amb una shinyApp. Per tant, hem hagut d'incloure un objecte d'interfície d'usuari i una funció de servidor.

Per altra banda, hem hagut de modificar la incorporació de les dades que l'script no pot incorporar des de Pandora, com els casos de novo, estudis de co-segregació, estudis funcionals, i altres evidències que puguin estar publicades via diferents inputs en Shiny, ja que en l'script en R inicial, aquesta informació s'havia d'incloure via terminal. Per últim, també hem hagut d'incloure un *gobutton* per tal d'indicar quan fer l'anàlisi i evitar que l'aplicatiu calculi constantment totes les funcions cada vegada que l'usuari fa una modificació a cadascun dels paràmetres.

```
# Sidebar with a slider input for number of bins
# Numeric Input with variant identifier in Pandora
sidebarLayout(
  sidebarPanel(
    helpText("Please, indicate the variant's identifier at Pandora
             [ctl + shift + j]"),

    textInput(inputId = "id", label = "Specify variant ID in Pandora"),
    numericInput(inputId = "denovo_noconfirmed", label = "Number of the novo cases reported, paternity non-confirmed", value = 0, min = 0),
    numericInput(inputId = "denovo_confirmed", label = "Number of the novo cases reported, paternity confirmed", value = 0, min = 0),
    numericInput(inputId = "cosegregation", label = "Number of the cosegregated families reported", value = 0, min = 0),
    selectInput(inputId = "Functional_evidence", label = "Are functional studies demostrating variant pathogenicity?",
                choices = list("There is no evidence" = 0,
                               "There is Functional_evidence" = 1), selected = 0),
    selectInput(inputId = "PPAT_evidence", label = "Are relevant references demostrating variant pathogenicity?",
                      choices = list("There is no evidence" = 0,
                                     "There is PPAT_evidence" = 1), selected = 0),
    selectInput(inputId = "PPOL_evidence", label = "Are relevant references demostrating variant neutrality?",
                    choices = list("There is no evidence" = 0,
                                   "There is PPOL_evidence" = 1), selected = 0),
    actionButton("go", "Search")
),
```

Posteriorment, hem fet un disseny senzill i endreçat per tal d'afavorir que el GUI tingui un aspecte força agradable i *user-friendly*, i hem reorganitzat les funcions i hem dividit l'App en 3 parts.

La primera, on l'usuari indica la variant que vol classificar. Com aquesta s'indica a través d'un codi de la base de dades Pandora, i no la variant en concret, la primera funció, anomenada "VARIANT CONFIRMATION", mostra en pantalla quina variant classificarà perquè l'usuari pugui detectar si hi ha hagut algun error i s'està classificant una variant diferent a la que ell considera.

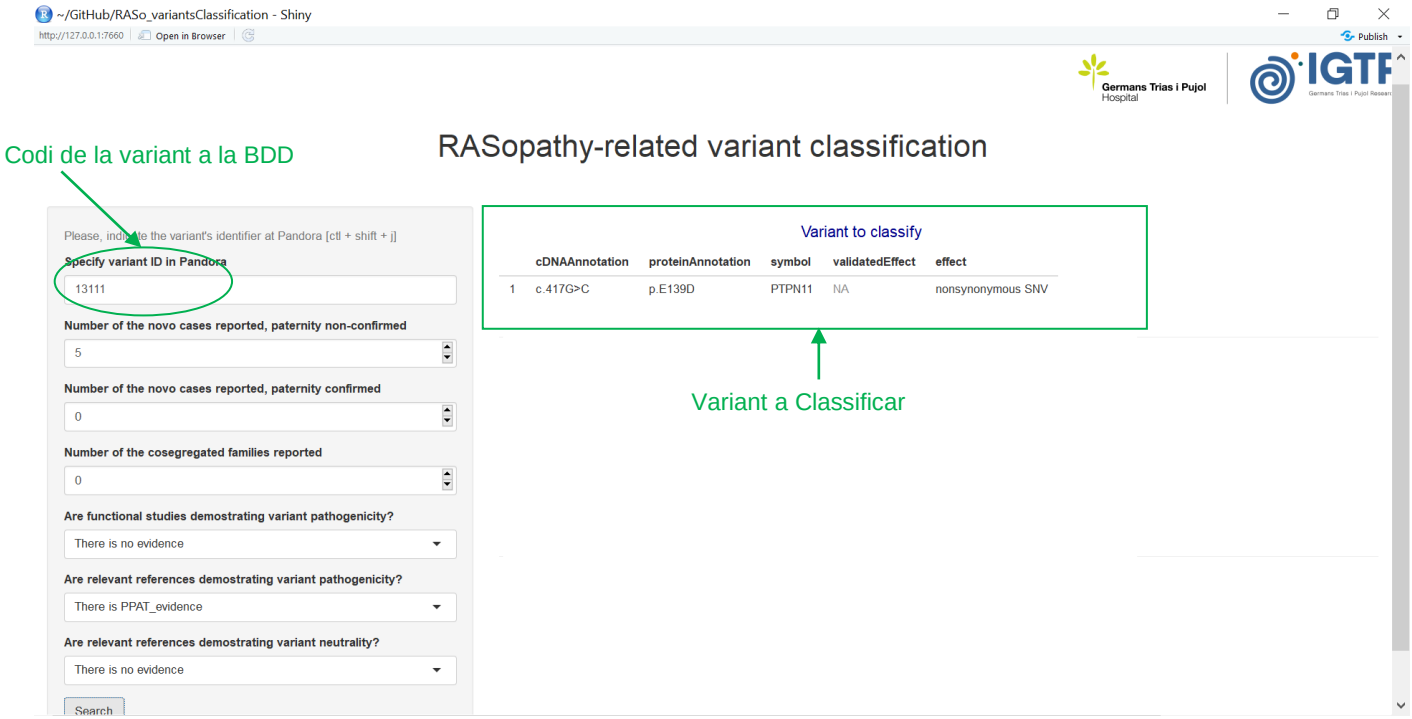


Figura 4: Visualització del GUI a nivell local un cop introduït el codi de la variant a analitzar.

La segona part consisteix en determinar de les diferents evidencies disponibles, quins criteris compleix la variant a classificar. Aquestes evidencies es determinen en funció de les dades que la segona funció desenvolupada pot determinar per ella mateixa a partir de la informació de la BDD Pandora i de UCSC, així com de les dades que l'usuari pot introduir en la mateixa pantalla: casos de novo reportats, estudis de co-segregació, evidència d'assajos funcionals, i si hi altres grups reconeguts l'han descrit anteriorment com a benigna o com a patogènica. Aquesta funció s'ha anomenat "AUTOMATIC CRITERIA FUNCTION". Els criteris que caracteritzen la variant a classificar també es poden veure en el GUI perquè l'usuari pugui contrastar-los.

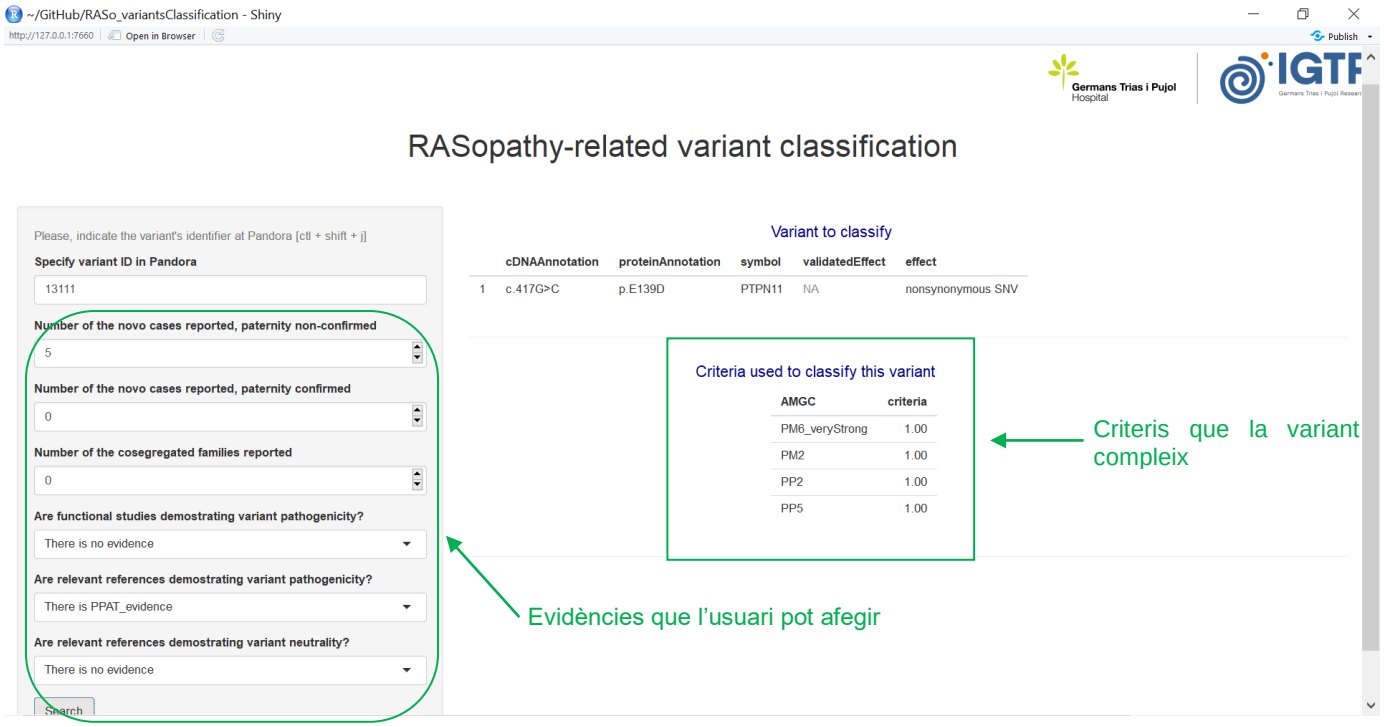


Figura 5: Visualització de la segona funció de la App, on un cop introduït el codi de la variant a analitzar, s'imprimeixen en pantalla només els criteris que la variant compleix.

I la darrera part realitzada per la funció "FINAL CLASSIFICATION" és la taula amb el resum de la classificació de la variant a partir dels criteris obtinguts en la part anterior.

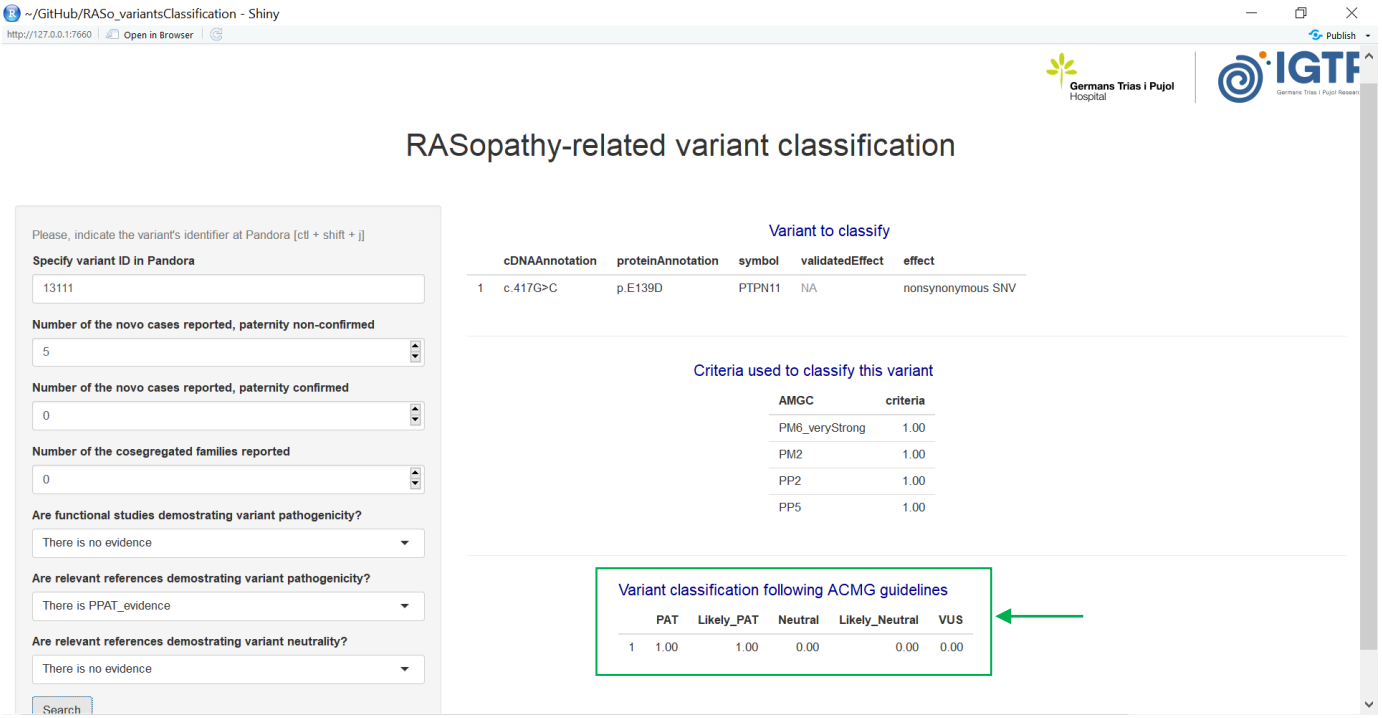


Figura 6: Visualització de la tercera funció de la App, on un cop introduït el codi de la variant a analitzar, s'imprimeixen en pantalla la classificació final de la variant. En aquest cas, probablement patogènica.

En l'annexe s'inclou les diferents etapes que van incloure el seu desenvolupament.

2.2.3. Validació GUI a nivell local

Aquest script i el posterior GUI s'ha testat en un conjunt de 20 variants ja classificades i s'ha pogut comprovar que la classificació coincideix amb la realitzada manualment.

Pandora ID	Gen	Variant (cDNA)	Variant (proteïna)	Classificació Manual	Classificació GUI Local
66756	NF1	c.5471T>C	p.I1824T	Prob.Pat	Prob.Pat
16898	NF1	c.4972A>G	p.I1658V	Benigna	Benigna
30606	NF1	c.2543G>A	p.G848E	Prob.Pat	Prob.Pat
22394	SPRED1	c.88G>C	p.G30R	Prob.Pat	Prob.Pat
23536	SPRED1	c.1149_1152delAGAG	p.S383fs	VSI	VSI
8445	SPRED1	c.1044T>C	p.V348V	Benigna	Benigna
26217	RAF1	c.775T>A	p.S259T	VSI	VSI
22186	BRAF	c.1783T>C	p.F595L	Prob.Patog.	Prob.Patog.
21230	BRAF	c.1219C>A	p.P407T	VSI	VSI
12379	KRAS	c.535G>A	p.G179S	Benigna	Benigna
23884	KRAS	c.179G>T	p.G60V	Prob.Patog.	Prob.Patog.
12379	KRAS	c.535G>A	p.G179S	Benigna	Benigna
21365	HRAS	c.34G>A	p.G12S	Prob.Patog.	Prob.Patog.
19020	HRAS	c.357C>T	p.D119D	Benigna	Benigna
8247	HRAS	c.81T>C	p.H27H	Benigna	Benigna

25786	NRAS	c.101C>T	p.P34L	VSI	VSI
17473	MAP2K2	c.410T>C	p.F137S	Prob.Patog.	Prob.Patog.
26285	MAP2K1	c.389A>G	p.Y130C	Prob.Patog.	Prob.Patog.
25310	SOS2	c.3813G>A	p.P1271P	Benigna	Benigna
24025	SOS1	c.2536G>A	p.E846K	VSI	VSI

Taula 2: Llistat de variants utilitzades per testar la fiabilitat de la App per classificar vairants dels gens associats a les RASopaties. En blau indiquem les que també incloem de manera visual.

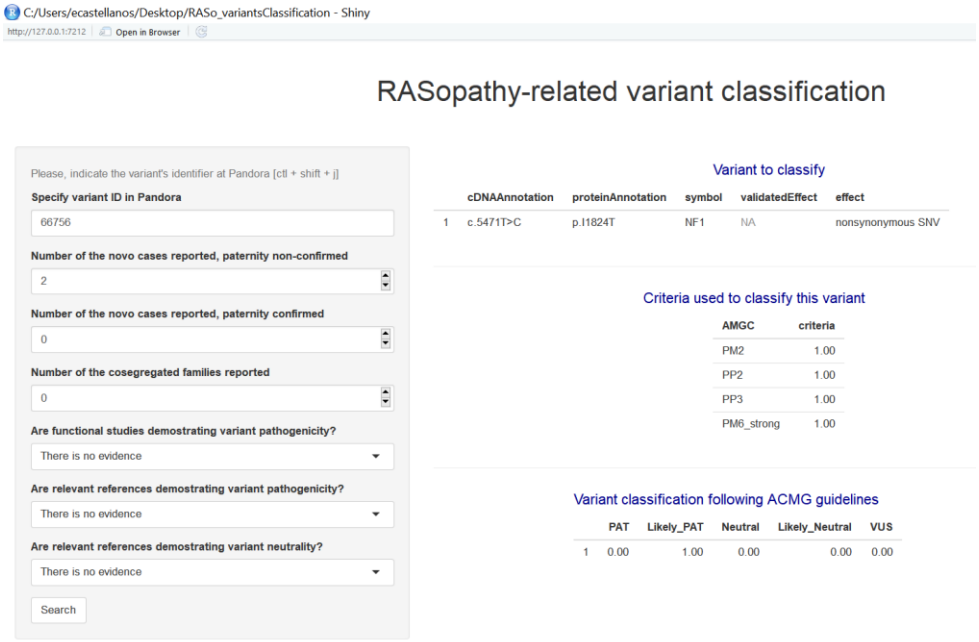


Figura 7: Visualització de la classificació automàtica de la variant NF1: c.5471T>C.

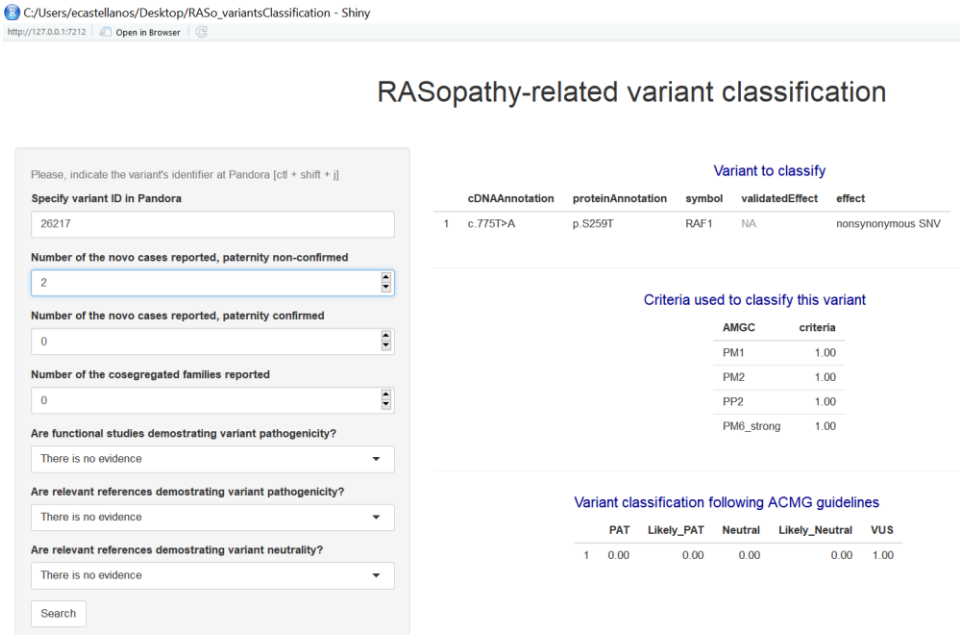


Figura 8: Visualització de la classificació automàtica de la variant RAF1: c.775T>A.

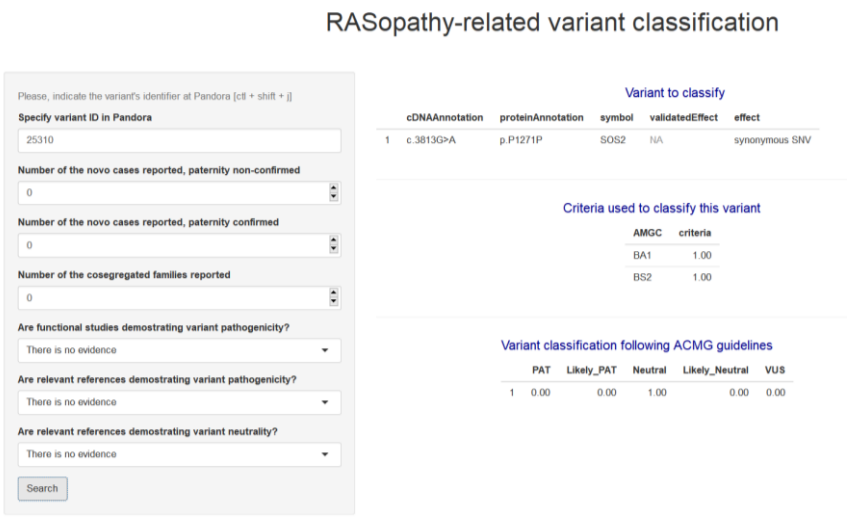


Figura 9: Visualització de la classificació automàtica de la variant SOS2: c.3823G>A.

2.2.4. GUI en servidor Shiny

Per tal d'implementar el GUI en el servidor Shiny del centre on es troba la Unitat de Diagnòstic, per tal de fer accessible l'aplicació desenvolupada no només a nivell local sinó també *on-line* per altres membres del grup, vam instal·lar-los en el sistema local on s'ha desenvolupat la app (portàtil Windows) el sistema *MobaXterm* per tal de poder-nos connectar al servidor Linux del centre de manera remota. També vam demanar permisos a l'administrador del centre per tal de poder copiar els nostres arxius així com poder llegir els diferents logs de la nostra aplicació. Posteriorment vam demanar que l'administrador ens instal·lés els paquets de *postgres client* per connectar-nos a la base de dades i també *RMySQL* per tal de poder les cerques dins la base de dades. Tot seguit vam clonar la carpeta de l'app amb els arxius necessaris perquè fos funcional així com les credencials per conectar-se a la base de dades Pandora des de GitHub.

El primer problema que ens vam trobar és que tal i com estava dissenyada l'app, aquesta no trobava el fitxer de les credencials per connectar-se a la base de dades, ja que el *path* indicat era el local i es va modificar per ser llegit en el servidor. Posteriorment ens vam adonar que la connexió a la nostra base de dades Padora es bloquejava només obrir-la, ja que al ser una app en un servidor, constantment estava interaccionant amb la base de dades i les sessions anteriors no es tancaven, a diferència de l'app a nivell local. Per tal de corregir aquest error, vam modificar l'app per incloure primerament un *submit button*, i posteriorment un *action button*, així com una petita funció per tancar la sessió i desconnectar-se de la base de dades un cop finalitzada cada cerca.

```
  ## session end clean up
  cancel.onSessionEnded <- session$onSessionEnded(function() {
      RPostgreSQL::dbDisconnect(con)
      RMySQL::dbDisconnect(my_connection)
})
```

També vam modificar les diferents funcions per tal de que aquestes estiguessin lligades a la introducció de les dades per part de l'usuari i que només fes la cerca de la variant a classificar un cop l'usuari premés el botó *Search*, tant a la part on descrivim la funció pròpiament com quan la cridem mitjançant un *eventReactive*.

- En la part de la funció pròpiament

```
Automatic_criteria_AMCG<- function(id = input$id, con = con, denovo_noconfirmed = input$denovo_noconfirmed,
                                   denovo_confirmed = input$denovo_confirmed, cosegregation = input$cosegregation,
                                   PPAT_evidence = as.numeric(as.character(input$PPAT_evidence)),
                                   PPOL_evidence = as.numeric(as.character(input$PPOL_evidence)),
                                Functional_evidence = as.numeric(as.character(input$Functional_evidence))){

  if (is.null(input$id) | is.null (input$denovo_noconfirmed) |
      is.null(input$cosegregation) | is.null (input$PPAT_evidence) |
      is.null (input$PPOL_evidence) | is.null(input$Functional_evidence) | input$go == 0) {

    ## nothing to do here
    return(NULL)
  }
```

- En *l'eventReactive* de l'app:

```
#### AUTOMATIC VARIANT CLASSIFICATION

    AutomClass_reactive <-eventReactive(c(input$go, input$id, input$denovo_noconfirmed, input$denovo_confirmed, input$cosegregation,
                                          input$PPAT_evidence,input$PPOL_evidence, input$Functional_evidence),
                                        if (input$go == 0){
                                          return(NULL)
                                        }
                                        else
                                        {Automatic_criteria_AMCG(id = input$id, con = con, denovo_noconfirmed = input$denovo_noconfirmed,
                                                                 denovo_confirmed = input$denovo_confirmed, cosegregation = input$cosegregation,
                                                                 PPAT_evidence = as.numeric(as.character(input$PPAT_evidence)),
                                                                 PPOL_evidence = as.numeric(as.character(input$PPOL_evidence)),
                                                                 Functional_evidence = as.numeric(as.character(input$Functional_evidence)))})

    criteria <- renderTable(expr = AutomClass_reactive(),rownames = FALSE, bordered = FALSE)
    output$AutoClass <- criteria
```

Un cop solucionat aquest problema de connexió, una de les tres funcions va ser funcional. Aquesta primera funció, a diferència de les altres dues, només crea una cerca a la base de dades i permet la visualització de la variant a classificar, però no realitza cap càlcul en el seu interior. Les altres dues funcions presentaven un error.

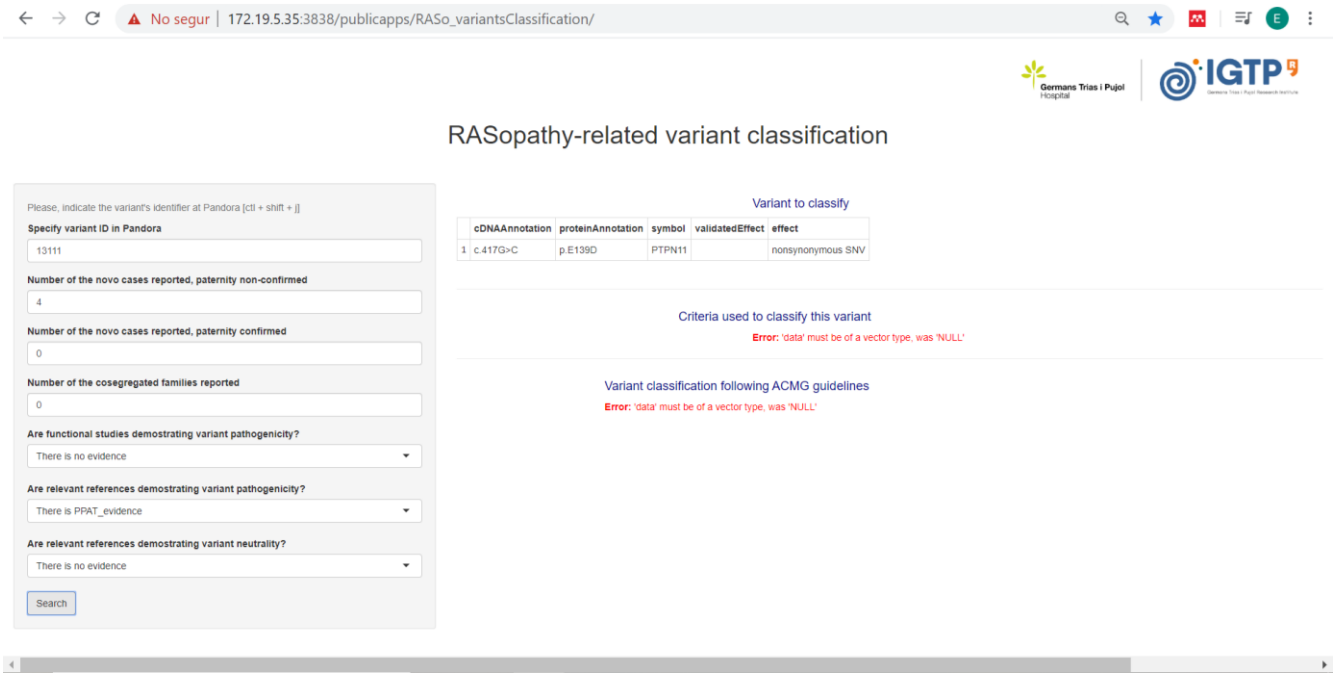


Figura 10: Visualització de l'aplicatiu en el servidor del centre.

Per tal de sol·lucionar aquest error, vam modificar les funcions per tal de que les funcions fossin depenents de l'entrada de l'identificador de la variant a partir del action button (if(!is.null(input$gobutton))) així com de les altres dades necessàries. La introducció d'aquesta dependència per iniciar les funcions no va millorar el problema i vam modificar el codi de les funcions pròpiament per crear aquestes dependències en cadascun dels més de 10 paràmetres que es cerquen i comptabilitzen per classificar les variants, però tampoc hem millorat la funcionalitat de la app en el servidor.

Totes les modificacions en l'app s'han anat probant a nivell local, pujades al GitHub amb un *commit* per cada modificació i clonades en la carpeta del servidor *shiny* del centre. Actualment, l'app és visible per altres usuaris via on-line i hem siguit capaços de fer funcionar una de les tres funcions. El fet de que en local no ens generi cap d'aquests errors i que només poguem comprovar si les modificacions realitzades són útils connectant-nos de manera remota al servidor, copiant el codi a GitHub i d'aquí al servidor, i aleshores poder testar els canvis, no agilitza la cerca de la sol·lució.

La implementació de l'app en el servidor estava prevista realiztar-la durant el desenvolupament de la PEC3, assumint que l'app local funcionaria en el servidor amb algunes petites modificacions. Aquest no ha sigut el cas, ja que seguim sense fer funcionals dues de les tres accions presents en la app. Segurament, si haguéssim fet un testing de l'script inicial o bé del sript del GUI local, podríem haver detectat l'error que impedeix que funcioni correctament en el servidor.

3. Conclusions

En aquest treball s'ha pogut desenvolupar i validar un aplicatiu web o GUI a nivell local per tal de classificar de manera automàtica, amb dades obtingudes d'una base de dades pròpia (Pandora) i de la literatura, les variants en els gens causant de les RASopaties en funció del seu efecte deleteri. Aquesta classificació és d'alta rellevància pel estudis genètics, els quals determinen el maneig dels pacients així com la seva planificació familiar. Aquest aplicatiu s'ha pogut validar comparant vint variants prèviament classificades a mà, on s'ha comprovat que no només les classifica de la mateixa manera, sinó que els criteris per realitzar aquesta classificació són els correctes.

En concret, l'usuari només ha d'introduir l'identificador de la variant dins de la base de dades pròpia i tota la informació referent a la variant que s'obté dels diferents processos d'anotació amb Annovar, de bases de dades públiques com l'USCS, i dels fitxers que s'han creat amb informació específica de cada gen (dominis, *hotspots*, ...) és calculada automàticament. La resta d'informació més específica, com casos reportats en la literatura, resultats d'estudis funcionals publicats o si la variant a classificar co-segrega amb la malaltia o bé és *de novo*, s'ha d'introduir manualment mitjançant unes caixetes user-friendly en l'aplicatiu. Com a resultat, l'usuari visualitza 3 taules: una resum amb la variant a classificar, una segona amb els criteris utilitzats per classificar la variant, i la darrera amb la classificació final de la variant analiztada. Aquest GUI és molt rellevant per automatitzar aquest procés, ja que de manera manual és un procés tediós i molt farragós.

Dels objectius proposats, s'han pogut realitzar en la seva totalitat tots ells en la planificació establerta, excepte el desenvolupament del GUI en el servidor del laboratori, que funciona parcialment. En concret el GUI en el servidor és capaç de cridar la base de dades pròpia i cercar la variant, però durant el procés de classificació, les dues funcions creades per tal efecte donen un error que no hem sabut detectar. Per tal de fer aquestes comprovacions, hauríem d'haver realitzar un *testing* de cadascun dels passos de totes les funcions per determinar on hi ha l'error i perquè, però aquest darrer pas no s'ha pogut realitzar ni en el GUI local ni en el servidor. La metodologia utilitzada ha sigut l'adequada i els problemes obtinguts s'han pogut solucionar amb la cerca d'informació i amb l'ajut de la consultora del TFM, la dra. Izaskun Mayona..

De cara a un futur immediat, caldria realitzar un *testing* de l'script del GUI local per determinar si hi ha cap paràmetre o funció que tot i no alterar el resultat, genera algun tipus de *warning*. També caldria adaptar l'script per incloure missatges d'error en cada apartat per detectar, si mai hi hagués un problema, quin tipus és i on es dóna dins de l'script, com podria ser errors de

connex ió a la base de dades, canvis en les taules SQL que utilitza la base de dades, canvi en les versions dels paquets de R utilitzats, etc [...]. Però de manera general i per un futur més llunyà, valdria la pena considerar la re-escriptura de les funcions utilitzades, ja que aquestes són molt llargues i es podrien simplificar per tal de poder controlar més els paràmetres en cada pas així com els resultats parcials obtinguts.

4. Glossari

ACMG-AMP: American College of Medical Genetics Assocation for Molecular Pathology

ADN: àcid desoxiribonucleic

cDNA: coding DNA

Clin-Gen: **Clin**ical-**Gen**ome

ICO: Institut Català d'Oncologia

GOF: Gain of Function

NGS: Next Generation Sequencing

Patog.: Patogènic, és una de les possibles classificacions que pot rebre una variant segons la ACMG-AMP

Prob.Benigna: Variant probablement benigne/ probablement polimorfisme, és una de les possibles classificacions que pot rebre una variant segons la ACMG-AMP

Prob.Patog.: Probablement patogènic, és una de les possibles classificacions que pot rebre una variant segons la ACMG-AMP

UCSC: University of California Santa Cruz

VSI: Variant de significat clínic Indeterminat, és una de les possibles classificacions que pot rebre una variant segons la ACMG-AMP. VUS en anglès.

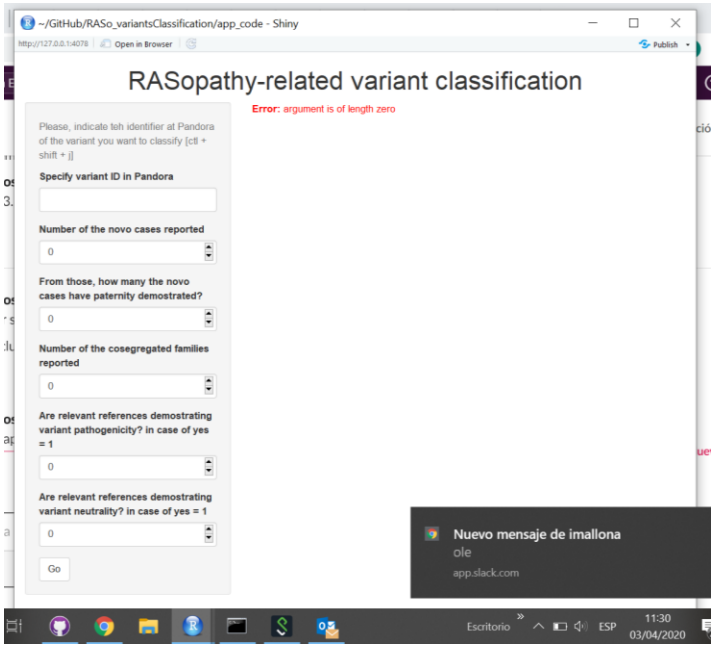
5. Bibliografia

1. Perge P, Igaz P. Family Screening and Genetic Counseling. *Exp Suppl*. 2019. doi:10.1007/978-3-030-25905-1_3
2. Claussnitzer M, Cho JH, Collins R, et al. A brief history of human disease genetics. *Nature*. 2020. doi:10.1038/s41586-019-1879-7
3. Society E, Genetics H, Journal E. Guidelines for diagnostic next generation sequencing. 2014;(December):1-59.
4. Biesecker LG, Burke W, Kohane I, Plon SE, Zimmern R. Next-generation sequencing in the clinic: are we ready? *Nat Rev Genet*. 2012;13(11):818-824. doi:10.1038/nrg3357
5. Rehm HL, Bale SJ, Bayrak-Toydemir P, et al. ACMG clinical laboratory standards for next-generation sequencing. *Genet Med*. 2013;15(9):733-747. doi:10.1038/gim.2013.92
6. Richards S, Aziz N, Bale S, et al. Standards and guidelines for the interpretation of sequence variants: a joint consensus recommendation of the American College of Medical Genetics and Genomics and the Association for Molecular Pathology. *Genet Med*. 2015;17(5). doi:10.1038/gim.2015.30
7. Rauen KA. The RASopathies. *Annu Rev Genomics Hum Genet*. 2013;14(1):355-369. doi:10.1146/annurev-genom-091212-153523
8. Tidyman WE, Rauen K a. Pathogenetics of the RASopathies. *Hum Mol Genet*. 2016;25(July):ddw191. doi:10.1093/hmg/ddw191
9. Tang H, Thomas PD. Tools for predicting the functional impact of nonsynonymous genetic variation. *Genetics*. 2016;203(2):635-647. doi:10.1534/genetics.116.190033
10. Gelb BD, Cavé H, Dillon MW, et al. ClinGen's RASopathy Expert Panel consensus methods for variant interpretation. *Genet Med*. 2018;00(August 2017). doi:10.1038/gim.2018.3
11. Grant AR, Cushman BJ, Cavé H, et al. Assessing the gene-disease association of 19 genes with the RASopathies using the ClinGen gene curation framework. *Hum Mutat*. 2018;39(11):1485-1493. doi:10.1002/humu.23624
12. Li Q, Wang K. InterVar: Clinical Interpretation of Genetic Variants by the 2015 ACMG-AMP Guidelines. *Am J Hum Genet*. 2017;100(2):267-280. doi:10.1016/j.ajhg.2017.01.004
13. Nykamp K, Anderson M, Powers M, et al. Sherloc: a comprehensive refinement of the ACMG–AMP variant classification criteria. *Genet Med*. 2017;00(November 2016):1-13. doi:10.1038/gim.2017.37
14. Postgresql. https://www.postgresql.org/.
15. UCSC. https://genome.ucsc.edu/goldenPath/help/mysql.html.
16. R-Project. https://www.r-project.org/about.html .
17. R documentation. https://www.rdocumentation.org/ .
18. RPostgreSQL. https://cran.r-project.org/web/packages/RPostgreSQL/index.html.
19. String. https://cran.r-project.org/web/packages/stringr/vignettes/stringr.html 17/12/2019.
20. gmodels. https://cran.r-project.org/web/packages/gmodels/index.html.
21. R-shiny. https://shiny.rstudio.com/.
22. GitHub. https://github.com.

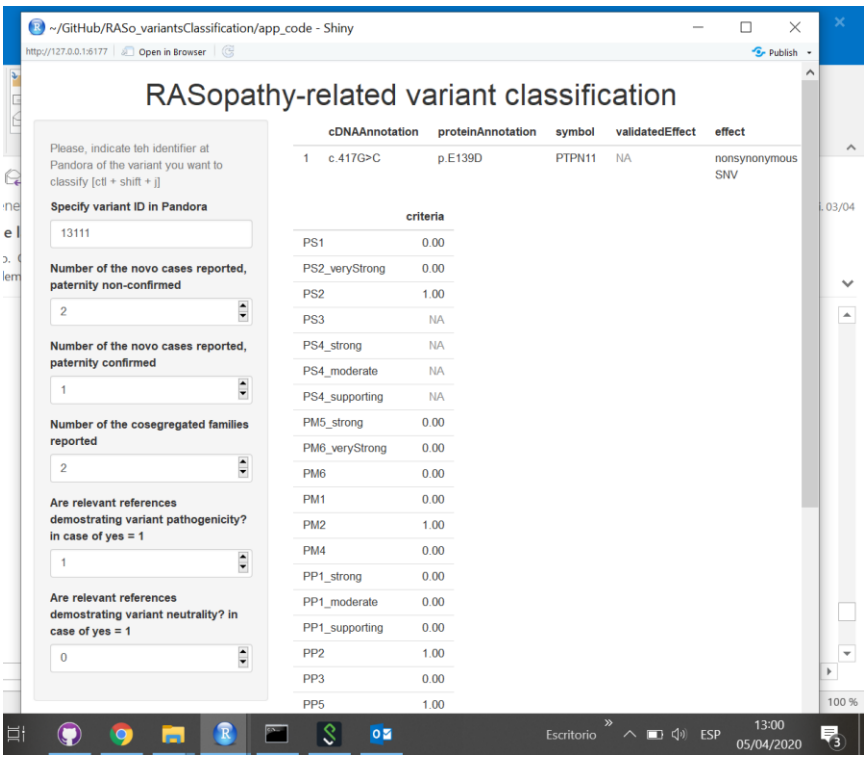
6. Annexos

En aquest aparta tus mostrem algunes captures de pantalla per veure l'evolució en el disseny del GUI a nivell local:

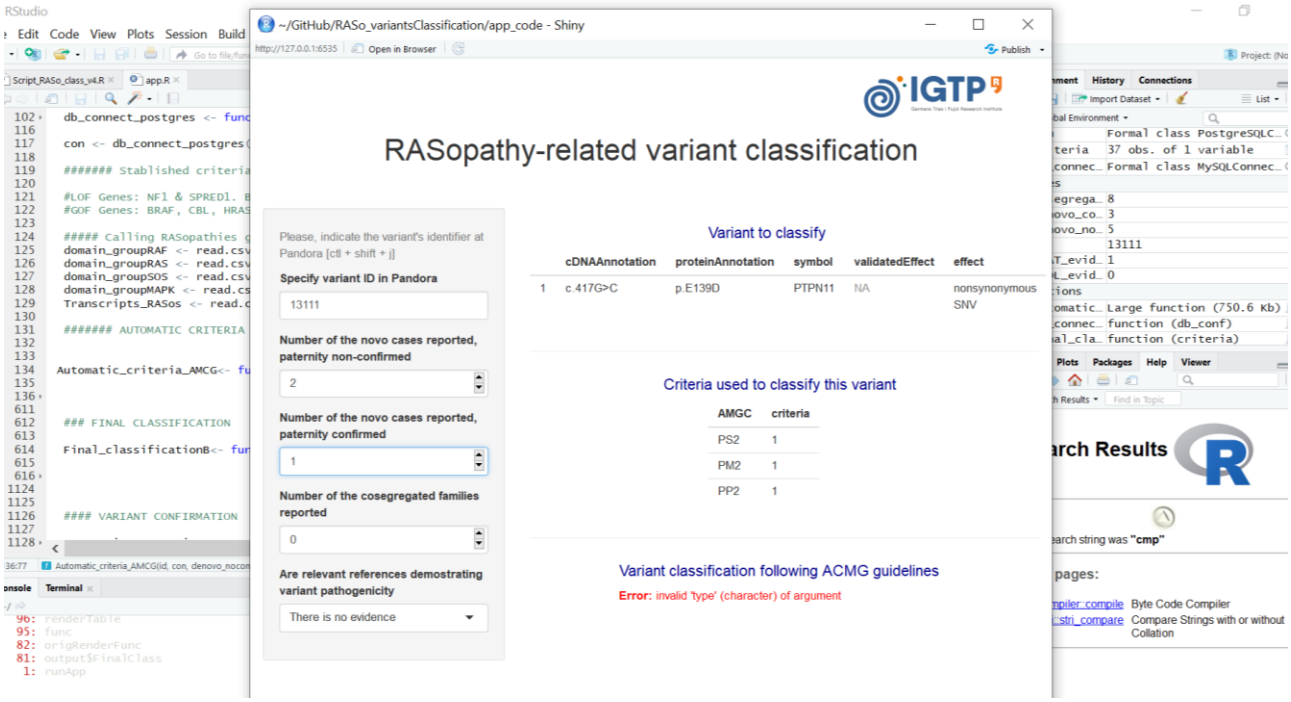
- Primers resultats:



- Posada a punt de la segona funció per fer visibles quins criteris compleix la variant analitzada. En aqeust punt, es veien tots, els que complia (valor 1) i els que no (valor 0 o NA)



- Visualització de la variant a clasificar i dels criteris que compleix la variant, aquesta vegada sí que de manera específica.



- GUI local definitiu amb la taula de classificació final: