MEMORIA TRABAJO SQL

Realizada por Miguel Ramón Alonso para el módulo de SQL del Máster de Bioinformática

Organizado por el ISCIII en colaboración con el BSC y otras instituciones

INTRODUCCIÓN

El objetivo de este trabajo consistió en aprender a trabajar con bases de datos relacionales SQL, utilizando la libreria OSS SQLite. Para la creación y modificación de dichas bases de datos, se propuso utilizar diversos scripts escritos en Python, los cuales se encargarían de relacionar la información contenida en los diversos ficheros almacenados en los históricos de ClinVar y CIViC requeridos para la realización de la práctica. Específicamente, se requirió el procesado de la base de datos de variantes de ClinVar, en sus versiones congeladas de junio y diciembre de 2022, mediante el script *clinvar_parser.py*; a continuación, debíamos escribir dos scripts más, basados en éste, los cuales se encargaran de añadir a la BBDD creada por dicho script la información hallada en los ficheros de citaciones y de estadísticas específicas de cada gen. Paralelamente, debíamos realizar un script similar al modelo utilizado en clase para el parseo de la base de datos de variantes contenida en CIViC.

PRIMERA PARTE

1) Carga de fichero de referencias de ClinVar en la BBDD generada por *clinvar_parser.py*

Misma lógica programática seguida en el script *clinvar_parser.py*, creando una nueva tabla específica para la carga de las citaciones contenidas en el fichero *var_cditations.txt*, y simplificando el código.

La misma estructura fue utilizada tanto para la creación de las tablas como para la función de apertura de fichero clinvar, la conexión con la base de datos de SQLite y la declaración de las diferentes tablas. Algunas de las tablas declaradas por el autor son redundantes o creadas con el propósito de una funcionalidad posterior ("à la" gene).

```
### From clinvar_parser.py ###

def open_clinvar_db(db_file):
    """
        This method creates a SQLITE3 database with the needed
        tables to store clinvar data, or opens it if it already
        exists
    """

    db = sqlite3.connect(db_file)

    cur = db.cursor()
    try:
        # Let's enable the foreign keys integrity checks
        cur.execute("PRAGMA FOREIGN_KEYS=ON")

        # And create the tables, in case they were not previously
        # created in a previous use
        for tableDecl in CLINVAR_TABLE_DEFS:
            cur.execute(tableDecl)
    except sqlite3.Error as e:
        print("An error occurred: {}".format(str(e)), file=sys.stderr)
    finally:
        cur.close()

    return db
```

Resto de puntos comentados in-script por el autor.

2) Carga de fichero de estadísticas por gen de ClinVar en la BBDD generada por *clinvar_parser.py*

Creación de fichero *clinvar_gene_stats_parser.py*. Misma estructura que en el fichero anterior, introduciendo el siguiente bloque de código (ln 63-67) para procesar la primera línea del fichero:

next(sf)

```
### From clinvar_gene_stats_parser.py ###

with gzip.open(stats_file, "rt", encoding="utf-8") as sf:
       headerMapping = None
       # remove first line of the file
       next(sf)
       cur = db.cursor()
```

En este caso, el fichero utilizado fue *gene_specific_summary_2022-(mm).txt.gz*, en sus versiones congeladas de junio y de septiembre de 2022.

3) Elaboración de BBDD de CIViC a partir del fichero *01-(MMM)-2022-VariantSummaries.tsv*

En este tercer caso, también se tomo como referencia el script anteriormente referido. Debido a la estructura del fichero *...VariantSummaries.tsv*, se decidió purgar gran cantidad de código no útil en el procesamiento de este fichero. Así mismo, se crearon tres tablas: una *(gene)* con información sobre el gen específico y sus respectivas entradas modelo **entrez_id**; en otra tabla *(variant)* se introdujo el grueso de la información contenida en la base de datos de CIViC; y una tercera en la que se incluyeron las expresiones HGVS de manera más ordenada. También se indexaron varias claves de las tablas para optimizar su acceso durante la realización de la segunda parte del ejercicio.

```
"""
CREATE INDEX IF NOT EXISTS assembly_variant ON variant(ensemble, ref_build)
"""

,
"""
CREATE INDEX IF NOT EXISTS coords_variant ON
variant(chr_start,chr_stop,chr_1,representative_transcript)
"""

,
"""
CREATE INDEX IF NOT EXISTS coords_2_variant ON
variant(chr_2_start,chr_2_stop,chr_2,representative_transcript_2)
"""

,
"""
CREATE INDEX IF NOT EXISTS gene_symbol_variant ON variant(gene_symbol)
"""

,
"""
CREATE TABLE IF NOT EXISTS hgvs_expressions (
    ventry_id INTEGER PRIMARY KEY AUTOINCREMENT,
    variant_id INTEGER NULL,
    hgvs_expression VARCHAR(64) NULL,
    FOREIGN KEY (variant_id) REFERENCES gene(variant_id)
        ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (variant_id) REFERENCES variant(variant_id)
        ON DELETE CASCADE ON UPDATE CASCADE
)
"""
```

A diferencia de los datos de ClinVar, en CIViC, las celdas que no tienen información vienen asignadas con N/A; esto debe ser reflejado en el código para realizar correctamente el cambio a NULL a la hora de insertar los datos en la BBDD.

```
else:
                    columnValues = re.split(r"\t",wline)

                    # As these values can contain "nulls", which are
                    # designed as 'N/A', substitute them for None
                    for iCol, vCol in enumerate(columnValues):
                        if len(vCol) == 0 or vCol == "N/A":
                            columnValues[iCol] = None
```

3.5) Elaboración de BBDD de CIViC a partir del fichero *01-(MMM)-2022-ClinicalEvidenceSummaries.tsv*

Para una mayor integridad de los datos, se decidió realizar un script adicional conteniendo la información de referencias y evidencia, "à la" ClinVar. De esta manera, mantuve una mayor lógica estructural en mi cabeza a la hora de entender qué estaba haciendo. El script se calcó al de CIViC original, modificando los datos requeridos. También lo hice para forzarme a escribir más código y a seguir entendiendo qué es lo que estabamos intentando hacer.

```
"""
CREATE TABLE IF NOT EXISTS evidence (
    evidence_id INTEGER PRIMARY KEY,
    variant_id INTEGER NOT NULL,
    gene_symbol VARCHAR(16) NOT NULL,
    disease VARCHAR(64) NULL,
    doid INTEGER NULL,
    phenotypes VARCHAR(32) NULL,
    evidence_type VARCHAR(16) NOT NULL,
    evidence_direction VARCHAR(32) NULL,
    evidence_level VARCHAR(1) NOT NULL,
    clinical_significance VARCHAR(32) NULL,
    evidence_statement VARCHAR(512) NULL,
    rating INTEGER NULL
)
"""
,
"""
CREATE TABLE IF NOT EXISTS drugs (
    evidence_id INTEGER PRIMARY KEY,
    variant_id INTEGER NOT NULL,
    drugs VARCHAR(64) NULL,
    drug_interaction_type VARCHAR(32) NULL,
    FOREIGN KEY (evidence_id) REFERENCES evidence(evidence_id)
        ON DELETE CASCADE ON UPDATE CASCADE
)"""
,
"""
CREATE TABLE IF NOT EXISTS citations (
    evidence_id INTEGER PRIMARY KEY,
    variant_id INTEGER NOT NULL,
    citation_id INTEGER NOT NULL,
    source varchar(16) NOT NULL,
    asco_id INTEGER NULL,
    citation VARCHAR(128) NOT NULL,
    nct_ids VARCHAR(32) NULL,
    FOREIGN KEY (evidence_id) REFERENCES evidence(evidence_id)
        ON DELETE CASCADE ON UPDATE CASCADE
)"""
```

SEGUNDA PARTE

Consultas SQL

1) ¿Cuántas variantes están relacionadas con el gen P53 tomando como referencia el ensamblaje GRCh38 en ClinVar y en CIViC?

INPUT

```
/* CLINVAR */
SELECT gene_symbol, assembly, COUNT(*) as count
FROM "variant"
WHERE gene_symbol LIKE '%TP53%'
AND assembly LIKE '%38%';

/* CIViC */
SELECT *, COUNT(*) as count
FROM "variant"
WHERE gene_symbol LIKE '%TP53%'
AND ref_build LIKE '%38%';
```

OUTPUT

```
/* ##### JUNE ##### */
/* CLINVAR */
gene_symbol,assembly,count
TP53,GRCh38,2547

/*  CIViC */
gene_symbol,ref_build,count
NULL,NULL,0

/* ##### DECEMBER ##### */
/* CLINVAR */
gene_symbol,assembly,count
TP53,GRCh38,2761

/* CIViC */
gene_symbol,ref_build,count
NULL,NULL,0
```

2) ¿Qué cambio del tipo “single nucleotide variant” es más frecuente, el de una Guanina por una Adenina, o el una Guanina por una Timina? Usad las anotaciones basadas en el ensamblaje GRCh37 para cuantificar y proporcionar los números totales, tanto para ClinVar como para CIViC.

INPUT

```
/* CLINVAR */
SELECT type,ref_allele,alt_allele,assembly, COUNT(*) as count
FROM variant
WHERE assembly LIKE "%37%"
AND type LIKE "%single%"
AND (ref_allele IS "G" AND alt_allele IS "A")
OR (ref_allele IS "G" AND alt_allele IS "T")
GROUP BY ref_allele, alt_allele;

/* CIViC */
SELECT var_types,ref_bases,var_bases,ref_build, COUNT(*) as count
FROM variant
WHERE ref_build LIKE "%37%"
AND (ref_bases IS "G" AND var_bases IS "A")
OR (ref_bases IS "G" AND var_bases IS "T")
GROUP BY ref_bases, var_bases;
```

OUTPUT

```
/* ##### JUNE ##### */

/* CLINVAR */
type,ref_allele,alt_allele,assembly,count
single nucleotide variant,G,A,GRCh37,299595
single nucleotide variant,G,T,GRCh37,132780

/* CIViC */
var_types,ref_bases,var_bases,ref_build,count
"missense_variant,transcript_fusion",G,A,GRCh37,95
"missense_variant,transcript_fusion",G,T,GRCh37,49

/* ##### DECEMBER ##### */
/* CLINVAR */
type,ref_allele,alt_allele,assembly,count
single nucleotide variant,G,A,GRCh37,316717
single nucleotide variant,G,T,GRCh37,145358

/* CIViC */
var_types,ref_bases,var_bases,ref_build,count
"missense_variant,transcript_fusion",G,A,GRCh37,96
"missense_variant,transcript_fusion",G,T,GRCh37,50
```

3) ¿Cuáles son los tres genes de ClinVar con un mayor número de inserciones y deleciones? Usa el ensamblaje GRCh37 para cuantificar y

proporcionar los números totales.

INPUT

```
/* CLINVAR */
SELECT DISTINCT gene_symbol,type,assembly, COUNT(*) as count
FROM variant
WHERE assembly LIKE "%37%"
AND type LIKE "%insertion%" OR type LIKE "%deletion%"
GROUP BY gene_symbol ORDER BY count DESC;
```

OUTPUT

```
/* ##### JUNE ##### */
/* CLINVAR */
gene_symbol,type,assembly,count
BRCA2,Deletion,GRCh37,4698
BRCA1,Deletion,GRCh37,4056
NF1,Deletion,na,2802

/* ##### DECEMBER ##### */
/* CLINVAR */
gene_symbol,type,assembly,count
BRCA2,Deletion,GRCh37,5037
BRCA1,Deletion,GRCh37,4232
NF1,Deletion,GRCh37,3017
```

4) ¿Cual es la deleción más común en el cáncer hereditario de mama en CIViC? ¿Y en ClinVar? Por favor, incluye en la respuesta además en qué genoma de referencia, el número de veces que ocurre, el alelo de referencia y el observado.

INPUT

```
/* CLINVAR */
SELECT gene_symbol,ref_allele,alt_allele,assembly,phenotype_list, COUNT(*) as ocurrence
FROM variant
WHERE phenotype_list LIKE "%breast%cancer%"
AND type LIKE "%del%"
GROUP BY gene_symbol
ORDER BY count(*) DESC
LIMIT 1;

/* CIViC */
SELECT evidence.gene_symbol,disease,ref_bases,var_bases,ref_build, count(*) AS ocurrence
FROM evidence
JOIN variant ON variant.variant_id=evidence.variant_id
WHERE disease LIKE "%breast%"
AND ref_bases IS NOT NULL AND var_bases IS NULL
ORDER BY ocurrence DESC
LIMIT 1;
```

OUTPUT

```
##### JUNE #####
/* CLINVAR */
gene_symbol,ref_allele,alt_allele,assembly,phenotype_list,ocurrance
BRCA2,CTTTCGG,C,GRCh37,"Breast-ovarian cancer (...)",4321

/* CIViC */
evidence.gene_symbol,disease,ref_bases,var_bases,ref_build,ocurrance
ERBB2,Breast Cancer,TTGAGGGAAAACACA,NULL,GRCh37,2

/* ##### DECEMBER ##### */
/* CLINVAR */
gene_symbol,ref_allele,alt_allele,assembly,phenotype_list,ocurrance
BRCA2,CTTTCGG,C,GRCh37,"Breast-ovarian cancer (...)",4345

/* CIViC */
evidence.gene_symbol,disease,ref_bases,var_bases,ref_build,ocurrance
ERBB2,Breast Cancer,TTGAGGGAAAACACA,NULL,GRCh37,2
```

5) Ver el identificador de gen y las coordenadas de las variantes de ClinVar del ensamblaje GRCh38 relacionadas con el fenotipo del Acute infantile liver failure due to synthesis defect of mtDNA-encoded proteins.

INPUT

```
/* CLINVAR */

SELECT DISTINCT gene_symbol,chro,chro_start,chro_stop,assembly,phenotype_list
FROM variant
WHERE phenotype_list LIKE "%infantile%liver%mtDNA%"
AND assembly LIKE "%38%";
```

OUTPUT: Too large to be shown

6) Para aquellas variantes de ClinVar con significancia clínica “Pathogenic” o “Likely pathogenic”, recuperar las coordenadas, el alelo de referencia y el alelo alterado para la hemoglobina (HBB) en el assembly GRCh37.

INPUT

```
/* CLINVAR */
SELECT DISTINCT
gene_symbol,chro,chro_start,chro_stop,ref_allele,alt_allele,assembly,significance
FROM variant
JOIN clinical_sig ON clinical_sig.ventry_id=variant.ventry_id
WHERE gene_symbol LIKE "%HBB%"
AND assembly LIKE "%37%"
AND (significance IS "Pathogenic" OR significance IS "Likely pathogenic")
ORDER BY significance;
```

OUTPUT: Too large to be shown

7) Calcular el número de variantes del ensamblaje GRCh38 que se encuentren en el cromosoma 13, entre las coordenadas 10,000,000 y 20,000,000 , tanto para ClinVar como para CIViC.

INPUT

```
/* CLINVAR */
SELECT chro, COUNT(*) AS ocurrance
FROM variant
WHERE assembly LIKE "%38%"
AND chro IS "13"
AND (chro_start > 10000000 AND chro_stop < 20000000);

/* CIViC */
SELECT chr_1,chr_2, COUNT(*) AS ocurrance
FROM variant
WHERE ref_build LIKE "%38%"
AND (chr_1 IS 13 AND chr_start > 10000000 AND chr_stop < 20000000)
OR (chr_2 IS 13 AND chr_2_start > 10000000 AND chr_2_stop < 20000000);
```

OUTPUT

```
/* ##### JUNE ##### */
/* CLINVAR */
chro,ocurrance
13,18

/* CIViC */
chr_1,chr_2,ocurrance
NULL,NULL,0

/* ##### DECEMBER ##### */
/* CLINVAR */
chro,ocurrance
13,20

/* CIViC */
chr_1,chr_2,ocurrance
NULL,NULL,0
```

8) Calcular el número de variantes de ClinVar para los cuáles se haya provisto entradas de significancia clínica que no sean inciertas (“Uncertain significance”), del ensamblaje GRCh37, en aquellas variantes relacionadas con BRCA2.

INPUT

```
/* CLINVAR */
SELECT DISTINCT assembly, COUNT(*) as ocurrance
FROM variant
LEFT JOIN clinical_sig ON clinical_sig.ventry_id=variant.ventry_id
WHERE gene_symbol LIKE "%BRCA2%"
AND assembly LIKE "%37%"
AND significance NOT LIKE "%uncertain%"
```

OUTPUT

```
/* ##### JUNE ##### */
/* CLINVAR */
assembly,ocurrance
GRCh37,8994

/* ##### DECEMBER ##### */
/* CLINVAR */
assembly,ocurrance
GRCh37,9837
```

9) Obtener el listado de pubmed_ids de ClinVar relacionados con las variantes del ensamblaje GRCh38 relacionadas con el fenotipo del glioblastoma.

INPUT

```
/* CLINVAR */
SELECT DISTINCT variation_id,citation_source,citation_id,assembly,phenotype_list
FROM variant
LEFT JOIN reference ON reference.ventry_id = variant.ventry_id
WHERE assembly LIKE "%38%"
AND phenotype_list LIKE "%glioblastoma%"
```

OUTPUT: Too large to be shown

10) Obtener el número de variantes del cromosoma 1 y calcular la frecuencia de mutaciones de este cromosoma, tanto para GRCh37 como para GRCh38. ¿Es esta frecuencia mayor que la del cromosoma 22? ¿Y si lo comparamos con el cromosoma X? Tomad para los cálculos los tamaños cromosómicos disponibles tanto en

https://www.ncbi.nlm.nih.gov/grc/human/data?asm=GRCh37.p13 como en https://www.ncbi.nlm.nih.gov/grc/human/data?asm=GRCh38.p13.

INPUT

```sql
/* CLINVAR */
SELECT chro,assembly, COUNT(*) as ocurrance,
CASE
    WHEN assembly LIKE '%37%' AND chro = 1 THEN 249250621.0
    WHEN assembly LIKE '%37%' AND chro = 22 THEN 51304566.0
    WHEN assembly LIKE '%37%' AND chro = 'X' THEN 155270560.0
    WHEN assembly LIKE '%38%' AND chro = 1 THEN 248956422.0
    WHEN assembly LIKE '%38%' AND chro = 22 THEN 50818468.0
    WHEN assembly LIKE '%38%' AND chro = 'X' THEN 156040895.0
END AS chr_length,
COUNT(*) / CASE
    WHEN assembly LIKE '%37%' AND chro = 1 THEN 249250621.0
    WHEN assembly LIKE '%37%' AND chro = 22 THEN 51304566.0
    WHEN assembly LIKE '%37%' AND chro = 'X' THEN 155270560.0
    WHEN assembly LIKE '%38%' AND chro = 1 THEN 248956422.0
    WHEN assembly LIKE '%38%' AND chro = 22 THEN 50818468.0
    WHEN assembly LIKE '%38%' AND chro = 'X' THEN 156040895.0
END * 100 AS mut_frequency
FROM variant
WHERE (assembly LIKE "%37%" OR assembly LIKE "%38%")
AND (chro IS 22 OR chro IS 1 OR chro IS "X")
GROUP BY chro,assembly ORDER BY mut_frequency DESC;

/* CIViC */
SELECT chr_1,ref_build, COUNT(*) as ocurrance,
CASE
    WHEN ref_build LIKE '%37%' AND chr_1 = 1 THEN 249250621.0
    WHEN ref_build LIKE '%37%' AND chr_1 = 22 THEN 51304566.0
    WHEN ref_build LIKE '%37%' AND chr_1 = 'X' THEN 155270560.0
    WHEN ref_build LIKE '%38%' AND chr_1 = 1 THEN 248956422.0
    WHEN ref_build LIKE '%38%' AND chr_1 = 22 THEN 50818468.0
    WHEN ref_build LIKE '%38%' AND chr_1 = 'X' THEN 156040895.0
END AS chr_length,
COUNT(*) / CASE
    WHEN ref_build LIKE '%37%' AND chr_1 = 1 THEN 249250621.0
    WHEN ref_build LIKE '%37%' AND chr_1 = 22 THEN 51304566.0
    WHEN ref_build LIKE '%37%' AND chr_1 = 'X' THEN 155270560.0
    WHEN ref_build LIKE '%38%' AND chr_1 = 1 THEN 248956422.0
    WHEN ref_build LIKE '%38%' AND chr_1 = 22 THEN 50818468.0
    WHEN ref_build LIKE '%38%' AND chr_1 = 'X' THEN 156040895.0
END * 100 AS mut_frequency
FROM variant
WHERE (ref_build LIKE "%37%" OR ref_build LIKE "%38%")
AND (chr_1 IS 22 OR chr_1 IS 1 OR chr_1 IS "X")
GROUP BY chr_1,ref_build ORDER BY mut_frequency DESC;
```

OUTPUT

```
/* ##### JUNE ##### */
/* CLINVAR */
chro,assembly,ocurrance,chr_length,mut_frequency
22,GRCh37,32726,51304566,0.06378769484181973
22,GRCh38,31280,50818468,0.06155242617703469
1,GRCh37,126477,249250621,0.05074290266261764
1,GRCh38,122700,248956422,0.04928573403099439
X,GRCh37,64346,155270560,0.04144121074851537
X,GRCh38,60455,156040895,0.03874304873731979

/* CIViC */
chr_1,ref_build,ocurrance,chr_length,mut_frequency
22,GRCh37,19,51304566,0.000037033740817532694
1,GRCh37,63,249250621,0.00002527576450852653 8
X,GRCh37,22,155270560,0.00001416881603312308 4
1,GRCh38,1,248956422,4.0167672396898447e-7

/* ##### DECEMBER ##### */
/* CLINVAR */
chro,assembly,ocurrance,chr_length,mut_frequency
22,GRCh37,35343,51304566,0.06888860535337146
22,GRCh38,33866,50818468,0.06664112739486755
1,GRCh37,136176,249250621,0.05463416678909699
1,GRCh38,132361,248956422,0.05316633286125875
X,GRCh37,67744,155270560,0.04362964878854047
X,GRCh38,63782,156040895,0.04087518211171501

/* CIViC */
chr_1,ref_build,ocurrance,chr_length,mut_frequency
22,GRCh37,18,51304566,0.000035084596563978336
1,GRCh37,63,249250621,0.00002527576450852653 8
X,GRCh37,22,155270560,0.00001416881603312308 4
1,GRCh38,1,248956422,4.0167672396898447e-7
```