

## INACTIVACIÓN DE GENES EN MIELOMA MÚLTIPLE

En este trabajo se va a realizar la búsqueda de aquellos genes, que tras su inactivación, provocan un bloqueo en el crecimiento de células de mieloma múltiple, sin afectar a las células sanas.

Para facilitar su consulta, todos los ficheros de este estudio se encuentran publicados en GitHub: <https://github.com/fjredondo/mm-geneKnockout> (Redondo Aniorte, 2024).

El proceso que consigue encontrar los genes diana se ha dividido en tres más simples. Estos se han de ejecutar de manera secuencial. Son los siguientes:

1. Transformar los ficheros originales en formato “xml” a formato binario, ya que se acelera su carga en memoria para poder trabajar con ellos.
2. Cargar los ficheros binarios de los modelos y encontrar los genes específicos que afectan al crecimiento de cada tipo de célula. El término “genetic Minimal Cut Sets” o “gMCSs” hace referencia a este algoritmo (Apaolaza et al., 2019).
3. Por último, es necesario encontrar dentro de ese conjunto de genes (gMCS) aquellos, que de manera diferencial, al bloquearse afectan, exclusivamente, a las células de mieloma múltiple y no a las de tejido sano.

Con el fin de abordar cada problema por separado se han creado tres scripts en Python (“.py”) con la ayuda de los cuadernos de Jupyter (“.ipynb”) correspondientes. Estos son:

1. [modelToBinary.py](#)
2. [gMCSForBiomass.py](#)
3. [gKnockoutTargets.py](#)

La ejecución ordenada de los anteriores scripts se realiza lanzando un trabajo en el gestor Slurm. El comando es el siguiente: `sbatch submit\_sb\_ampere-mm.sh`

Con el fin de automatizar la creación de ficheros con formato binario, se ha creado la estructura de carpetas mostrada en la Figura 1. Las referencias a mieloma múltiple se encuentran bajo “/models/mm”, los tejidos en “/models/tissue”, y el modelo de humano general se encuentra en “/models/generic”. La ejecución del script `submit_sb_ampere-mm.sh` almacena su salida [slurm-32295.out](#) en “/SBATCH\_output”.

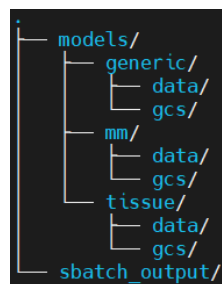


FIGURA 1: ESTRUCTURA DEL DIRECTORIO DE TRABAJO

Por último, el script `gKnockoutTargets.py` genera un informe final que muestra cómo, el knockout de los genes diana encontrados, afecta a cada uno de los tejidos. Este informe se almacena en [biomassAfterGeneKnockout.csv](#), en el raíz del directorio de trabajo.

## 1. ELECCIÓN DE MODELOS

Se han seleccionado tres líneas de mieloma múltiple y cuatro modelos de tejido sano. No obstante, los automatismos implantados permiten probar todas las líneas de mieloma múltiple y tejidos sanos que se deseen, basta con colocar los modelos en formato “xml” en las carpetas correspondientes (“/models/mm” o “/models/tissue”).

Mieloma Múltiple	Genérico	Tejido
KMM1.xml	Human_1_17.mat	colon.xml
MMIS.xml		kidney.xml
SKMM2.xml		liver.xml
		pancreas.xml

TABLA 1. MODELOS ELEGIDOS

Para acelerar la carga de los modelos en memoria, estos ficheros se han transformado a formato binario (extensión .dat), mediante la librería “pickle”. Este proceso se ha automatizado mediante el uso de un script de *sbatch* y el gestor de trabajos *Slurm*. Una vez lanzado, se genera una estructura de carpetas adecuada para alojar los ficheros (mostrada en Figura 1).

El script [modelToBinary.py](#) se encarga de esta tarea. Los archivos del modelo en formato “.xml” y “.dat” se pueden consultar en GitHub, dentro del siguiente archivo comprimido: [cobra\\_core\\_model\\_Models.tar.gz](#)

## 2. TOXICIDAD GENÉTICA ESPECÍFICA DE LOS TEJIDOS SANOS

El listado de genes para cada tejido cuyo bloqueo le produce toxicidad (impiden su crecimiento), se ha guardado en archivos con formato binario mediante la librería “pickle”.

Archivo	Genes
colon.gcs	131
kidney.gcs	127
liver.gcs	130
pancreas.gcs	129

TABLA 2. GENES DE TEJIDO QUE AFECTAN AL CRECIMIENTO

El script [gMCSForBiomass.py](#) se encarga de esta tarea. Los archivos “.gcs” se pueden consultar en GitHub, dentro del siguiente archivo comprimido: [cobra\\_core\\_model\\_Models.tar.gz](#)

## 3. TARGET GENÉTICOS DE LOS MODELOS DE MIELOMA MÚLTIPLE.

Para cada modelo de mieloma múltiple, se han calculado sus targets genéticos (qué genes, al bloquearse, impiden su crecimiento).

Archivo	Genes
MMIS.gcs	127
SKMM2.gcs	147
KMM1.gcs	124

TABLA 3. GENES MM QUE AFECTAN AL CRECIMIENTO

Los estudios siguientes cargan en memoria todos los ficheros .dat y .gcs. Para la carga automatizada se han diseñado dos funciones: *load\_pickle(ruta)* y *load\_files(ext,path)*. Su programación puede consultarse en [gKnockoutTargets.py](#).

El script [gMCSForBiomass.py](#) se encarga de esta tarea. Los archivos se pueden consultar en GitHub, dentro del siguiente archivo comprimido: [cobra\\_core\\_model\\_Models.tar.gz](#)

Como ejemplo de uso se tiene este caso: *mm\_gcs\_dict=load\_files("gcs","./models/mm/gcs")*. Vuelca en un diccionario una lista de genes vinculado al nombre del modelo.

## 4. ELIMINAR GENES QUE SEAN TOXICIDADES GENÉRICAS.

Se han eliminado de los conjuntos modelos de mieloma múltiple aquellos genes que generan toxicidades genéricas (las calculadas en clase para el modelo genérico). El script [gKnockoutTargets.py](#) se encarga de esta tarea.



Para conseguir este punto, se ha creado la función: *recursive\_operation\_over\_sets(d,o)*, que permite realizar operaciones de intersección o diferencia de manera recursiva con todos los conjuntos almacenados en el diccionario pasado como parámetro.

En este estudio se han de bloquear tres líneas de mieloma, por ello, lo primero que se realiza es una intersección entre ellos, de esta manera: *mm\_gcs = recursive\_operation\_over\_sets(mm\_gcs\_dict.copy(), 'intersection')*. Como se puede observar en el fichero de salida [slurm-32295.out](#), se realizan dos intersecciones:

```
Operation: ['MMIS', 127] intersection ['SKMM2', 147]. Result: 122
Operation: ['intersectionTarget', 122] intersection ['KMM1', 124]. Result: 120
```

Posteriormente, el conjunto con los 120 genes coincidentes se cruzan mediante diferencia con el conjunto de genes genéricos obtenidos en *Human\_1\_17.gcs*. Se lanzan estos comandos:

```
mm_gcs_without_generics = {**mm_gcs,**generic_gcs_dict}
mm_gcs_without_generics = recursive_operation_over_sets(mm_gcs_without_generics,'difference')
```

El resultado (29) es la diferencia. Se muestra la salida obtenida:

```
Operation: ['intersectionTarget', 120] difference ['Human_1_17', 92]. Result: 29
```

## 5. ESTRATEGIAS PARA ELIMINAR LÍNEAS DE MIELOMA MÚLTIPLE.

Los 29 genes obtenidos del apartado anterior afectan a cada una de las tres líneas de mieloma y, en cambio, no afectan al crecimiento del tejido sano, en el modelo general.

Para conseguir más precisión, es necesario cruzar esos 29 genes con los genes “gMCS” obtenidos en cada uno de los tejidos escogidos (Tabla 2. Genes de tejido que afectan al crecimientoTabla 2). El script [gKnockoutTargets.py](#) se encarga de esta tarea.

Los comandos lanzados ha sido estos:

```
mm_gcs_without_gTissue = {**mm_gcs_without_generics,**tissue_gcs_dict}
mm_gcs_without_gTissue = recursive_operation_over_sets(mm_gcs_without_gTissue,'difference')
```

El resultado (3) es la diferencia. Se muestra la salida obtenida:

```
Operation: ['differenceTarget', 29] difference ['colon', 131]. Result: 4
Operation: ['differenceTarget', 4] difference ['liver', 130]. Result: 3
Operation: ['differenceTarget', 3] difference ['pancreas', 129]. Result: 3
Operation: ['differenceTarget', 3] difference ['kidney', 127]. Result: 3
```

Por último, se ha realizado de manera expresa el knockout de los genes sobre cada uno de los modelos y se ha visto que afectaba a la producción de biomasa, exclusivamente, en las líneas de mieloma.

Gene	MMIS	SKMM2	KMM1	Human_1_17	colon	pancreas	liver	kidney
ENSG00000137700	0.0	0.0	0.0	-1	-1	-1	-1	-1
ENSG00000173614	0.0	0.0	0.0	124.7608	76.7285	81.0611	78.603	76.7256
ENSG00000135437	0.0	0.0	0.0	124.7608	76.7285	81.0611	78.603	76.7256

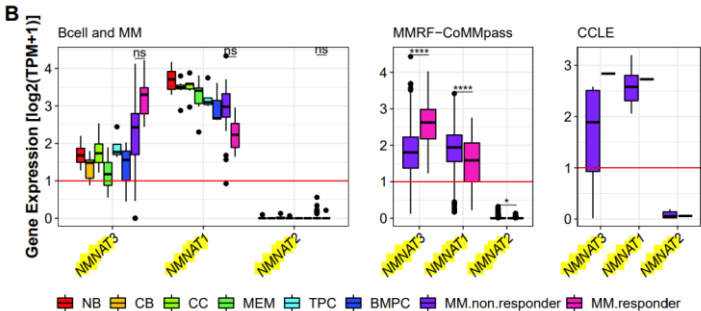
El primer gen no está registrado en el modelo genérico, ni en los específicos de cada tejido por eso da un -1. En cambio, los otros dos sí están registrados y muestran un claro patrón diferencial.

El proceso completo ha tardado en el servidor “ampere” 430 segundos.

6. ANÁLISIS DE LAS MEJORES ESTRATEGIAS.

Con el fin de aprovechar las ventajas de la automatización desarrollada se ha lanzado el proceso añadiendo 30 tejidos. El resultado ([30 tissues biomassAfterGeneKnockout.csv](#)) muestra que permanecen ENSG00000137700 y ENSG00000173614, en cambio, el gen ENSG00000135437 ha sido excluido tras la evaluación diferencial de sangre, músculo y esófago, como se puede comprobar en el fichero de salida: [30 tissues\\_slurm-32302.out](#) (ver líneas 314-316).

Se ha realizado una búsqueda por internet de estos genes. A continuación se muestra que se ha encontrado en la literatura.

Ubicación	ENSG00000137700 - SLC37A4 - Membrana Mitochondrial
Función	This gene regulates glucose-6-phosphate transport from the cytoplasm to the lumen of the endoplasmic reticulum, in order to maintain glucose homeostasis. It also plays a role in ATP-mediated calcium sequestration in the lumen of the endoplasmic reticulum. Mutations in this gene have been associated with various forms of glycogen storage disease. Alternative splicing in this gene results in multiple transcript variants.
Literatura	Genetics of enzymatic dysfunctions in metabolic disorders and cancer. The “Warburg effect” or aerobic glycolysis (Mahé et al., 2023).
Ubicación	ENSG00000173614 - NMNAT1 - Nucleoplasm, Nuclear bodies
Función	Human Nicotinamide Mononucleotide Adenylyl transferase (NMNAT) A nuclear enzyme essential for NAD synthesis
Literatura	Novel systems biology tools for the identification of biomarkers and drug targets in cancer research (Valcárcel-García, 2022) . <div><p><b>B</b></p></div>

TRABAJOS CITADOS

Apaolaza, I., Valcarcel, L. V., & Planes, F. J. (2019). GMCS: Fast computation of genetic minimal cut sets in large networks. *Bioinformatics*, 35(3). <https://doi.org/10.1093/bioinformatics/bty656>

Mahé, M., Rios-Fuller, T. J., Karolin, A., & Schneider, R. J. (2023). Genetics of enzymatic dysfunctions in metabolic disorders and cancer. *Frontiers in Oncology*, 13, 1230934. <https://doi.org/10.3389/fonc.2023.1230934>

Redondo Aniorte, F. J. (2024, January). *mm-geneKnockout*. <https://Github.Com/Fjredondo/Mm-GeneKnockout>.

Valcárcel-García, L. V. (2022). *Novel systems biology tools for the identification of biomarkers and drug targets in cancer research*. [Universidad de Navarra]. <https://doi.org/10.15581/10171/64531>