Universidad Autónoma de Madrid
Escuela politécnica superior

Trabajo Fin de Máster

Análisis de expresión diferencial para datos de Next Generation Sequencing (NGS) con múltiples condiciones experimentales

Máster Universitario en Investigación e Innovación en Tecnologías de la Información y las Comunicaciones (i2-TIC)

Autor: Daniel Giménez Llorente
Tutora: Irene Rodríguez Luján
Ponente: Ana María González Marcos

SEPTIEMBRE 2017

ANÁLISIS DE EXPRESIÓN DIFERENCIAL PARA DATOS DE NEXT GENERATION SEQUENCING (NGS) CON MÚLTIPLES CONDICIONES EXPERIMENTALES

Autor: Daniel Giménez Llorente
Tutora: Irene Rodríguez Luján
Ponente: Ana María González Marcos

Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
SEPTIEMBRE 2017

Agradecimientos

Quisiera agradecer en primer lugar a Irene Rodríguez Luján, por haberse querido arriesgar otro año siendo mi tutora, esta vez del Trabajo de Fin de Máster. Gracias por haberme llevado de vuelta al camino de la biología. También quiero agradecer a Alberto Torres, por su inconmesurable aportación a este trabajo.

Por otra parte también quiero agradecer a mis padres, a mi hermana y a todos mis amigos. Si estoy aquí hoy, es por todos vosotros. Por último, Ares, gracias por ayudarme a conocerme a mi mismo y por todo el apoyo que siempre me das.

Resumen

Resumen

La expresión génica diferencial es el proceso mediante el cual las células son capaces de decodificar la información contenida dentro del material genético (ácidos nucleicos) para la elaboración del producto génico necesario para el buen funcionamiento del organismo. El proceso más utilizado para revelar este producto génico es el RNA-Seq, a partir del cual se puede realizar el análisis de la expresión diferencial que nos permite detectar aquellos genes que son diferencialmente expresados con respecto a una condición de control. Este tipo de análisis se hace mucho más complejo cuando involucra múltiples condiciones experimentales, ya que no hay un método estándar con el que llevar a cabo su resolución. En concreto, el problema de encontrar genes diferencialmente expresados en una única condición experimental mientras que en el resto permanecen sin expresar es una línea de investigación que permanece abierta actualmente.

El objetivo de este Trabajo de Fin de Master es el análisis teórico e implementación de un método que sea capaz de solucionar este problema, poder seleccionar los genes que se expresan sólo en una condición experimental. Para ello, en este TFM se ofrece la descripción de un nuevo algoritmo, el QPFS-LASSO, que mediante la combinación del método de selección de variables QPFS y regularizadores LASSO permite detectar para cada condición experimental aquellos genes que, teniendo expresión diferencial con respecto a su condición de control, no se expresan en el resto de condiciones. QPFS-LASSO tiene como entrada la matriz con los datos de conteo de las secuencias provenientes de tecnologías NGS y los vectores de condiciones experimentales y control. Su salida es un vector de pesos con un valor para cada gen-condición. El método se ha implementado en R, pero las subrutinas más importantes se han implementado en C para obtener una mayor velocidad.

Para llevar a cabo la verificación de este método se han llevado a cabo dos acciones diferentes. Gracias al software Polyester se han generado simulaciones controladas de RNA-Seq que permiten observar el correcto funcionamiento del algoritmo dado que se conoce la realidad subyacente. Por otra parte, se ha analizado el funcionamiento del algoritmo al aplicarlo sobre una base de datos real procedente con el fin de determinar su rendimiento en escenarios reales no controlados. En ambos casos los resultados obtenidos han sido prometedores, siendo los genes más relevantes para el algoritmo aquellos específicos para una sola condición experimental. Finalmente, el algoritmo ha sido también evaluado sobre una base de datos de clasificación de dígitos manuscritos, obteniendo buenos resultados y demostrando su aplicabilidad a dominios más generales.

Palabras Clave

Expresión Diferencial, NGS, Múltiples Condiciones, RNS-Seq, QPFS, LASSO, Genes Específicos

Abstract

Gene expression is the process through which cells are able to decode information contained in genetic material (nucleic acids) in order to elaborate gene product necessary for the proper functioning of the organism. The most used technologies to release this gene product is RNA-Seq, from which differential gene expression analysis can be done to detect those genes that are differentially expressed with respect to a control condition. This type of analysis gets much more complex under multiple experimental conditions, as there is no standard method to resolve it. In particular, finding genes differentially expressed in a sole experimental condition while the rest stay unexpressed remains as an open line of research.

This Masters Thesis's objective is the theoretical analysis and implementation of a method capable of selecting genes expressed in just one experimental condition. In order to achieve this goal, this Master's Thesis proposes a new algorithm names QPFS-LASSO that, by combining the QPFS feature selection algorithm and Exclusive Group Lasso regularization, is able to detect those genes that are differentially expressed in a sole experimental condition. QPFS-LASSO has three inputs, the table of read counts and experimental and control conditions vectors. The outcome of the algorithm is a weight vector in which each entry represents the importance given to each pair gen-condition. The algorithm has been implemented in R, but the most important routines are implemented in C to achieve better performance.

Two different approaches have been used to verify the usefulness of the proposed method. On the one hand, multiple RNA-seq simulations have been generated by means of the Polyester software, and they allow us to confirm the proper functioning of the algorithm as the ground truth is known. On the other hand, a RNA-Seq database from the TCGA project has been tested in order to determine QPFS-LASSO's performance when facing real-world and non-controlled situations. The results obtained by the proposed algorithm are very promising as the most relevant genes for the algorithm are condition-specific genes in both cases. Finally, the algorithm has been also successfully tested over a handwritten digits classification problem to show its applicability to a wide range of domains.

Key words

Gene expression, NGS, Multiple Conditions, RNA-Seq, QPFS, QPFS-LASSO, Specific Genes

Índice general

Índice de Figuras

Índice de Tablas

1 Introducción

La tecnología RNA-sequencing (RNA-Seq) ha adquirido especial relevancia en los últimos años en diversas áreas de la biología. Este aumento significativo en su uso se debe a que ofrece varias ventajas sobre el resto de técnicas basadas en microarray.

En el proceso de análisis de los datos RNA-Seq, deben de llevarse a cabo diferentes pasos y la correcta realización de cada uno de ellos es crucial para la obtención de resultados significativos desde el punto de vista biológico. Uno de los pasos críticos de este análisis es la detección de genes diferencialmente expresados bajo diferentes condiciones experimentales. Gracias a esto se pueden observar si existen distintos comportamientos en la producción de proteínas en las células cuando éstas se encuentran sometidas a estímulos. Estos estímulos pueden ser de dos tipos: internos, como los que derivan en la diferenciación celular, o externos debidos al entorno. Muchas de las herramientas más utilizadas para llevar a cabo este análisis realizan comparaciones dos a dos entre las condiciones experimentales (típicamente, condición de control frente a condición experimental) [4–6] y no permiten identificar de manera sencilla aquellos genes que se expresan diferencialmente en una sola condición experimental [7]. Sin embargo, cada día es más común tratar de identificar estos genes específicos que nos pueden indicar por ejemplo alteraciones celulares debidas a algún cáncer [8] o incluso la presencia de diferentes toxinas en el ambiente.

Actualmente hay pocas técnicas destinadas a la detección de genes diferencialmente expresados bajo varias condiciones experimentales [8–12], y además ninguna se centra en la especificidad de ese gen con respecto al resto de condiciones, por lo que es un problema abierto dentro del campo de la bioinformática que precisa de soluciones eficaces. Además el abaratamiento del la tecnología de secuenciación de la última década está dando lugar a estudios experimentales cada vez más complejos y que precisan soluciones más específicas. Un ejemplo de aplicación es la detección e identificación de diferentes sustancias tóxicas en el agua mediante la bacteria E. Coli [13].

El problema de identificar genes específicos por condición experimental está además altamente relacionado con el campo de la selección de variables en aprendizaje automático. Sin embargo, los algoritmos de selección de variables para problemas multiclase (múltiples condiciones experimentales), típicamente tratan de encontrar atributos que sean buenos para el problema de clasificación multiclase en general, no focalizándose en encontrar variables particularmente buenas para cada una de las clases. En la literatura podemos encontrar alguna solución al problema multiclase, que finalmente consiste en comparar cada condición contra el resto [14]. En este sentido, la combinación de técnicas de selección de atributos [15] y técnicas de esparcidad

que imponen cierta estructura en las soluciones [16] pueden dar lugar a soluciones al problema de selección de variables específicas por clase.

Finalmente, cabe destacar que uno de los problemas que presenta el análisis de datos biológicos es la falta de un conjunto de datos del que se conozca totalmente y verídicamente la realidad subyacente y, que por tanto, permita comprobar la validez de los resultados obtenidos. Es por ello, que el análisis y desarrollo de algoritmos de expresión diferencial requiere también del conocimiento de técnicas de simulación de datos de RNA-Seq [17].

1.1. Objetivos

El objetivo final de este Trabajo de Fin de Máster es el diseño, implementación y validación de un método de selección de genes que permita detectar genes específicos bajo múltiples condiciones experimentales, es decir, genes que idealmente, se encuentran sobreexpresados para una condición individual frente a la condición de control, mientras que en el resto de condiciones el nivel de expresión génica no es diferencial. Para ello se plantearon los siguientes objetivos parciales:

1. Estudio, análisis y uso de las técnicas actuales de expresión diferencial para pares de condiciones experimentales así como su uso en casos de múltiples condiciones. Se centrará principalmente en el estudio del funcionamiento de los paquetes principales para análisis de expresión diferencial; el edgeR y DESeq.

2. Estudio de las aproximaciones existentes en la literatura para abordar el problema de varias condiciones experimentales y análisis crítico de su adecuación al problema de selección de genes para condiciones específicas.

3. Implementación de un método de selección de genes específicos por condición experimental. La idea principal consiste en partir del método de selección de variables global QPFS (*Quadratic Programming Feature Selection*) [15] y adaptarlo al problema de la expresión diferencial de genes. Para ello, se van a añadir regularizadores, en concreto el *Exclusive Group LASSO* [16], que aporta cierta estructura a la selección de genes. La combinación de ambos métodos nos aporta un selector de genes que da por válidos solo aquellos que se expresen para una condición experimental única.

4. Análisis crítico de los resultados del método de selección propuesto. Por una parte, el hecho de que los datos sean biológicos nos impide conocer la realidad subyacente de los mismos, lo cual implica una dificultad en el análisis posterior de los resultados. Sin embargo, en este trabajo también se ha usado el software Polyester para la simulación de datos provenientes de tecnologías NGS como RNA-Seq [17], lo que nos permite validar el método de selección.

1.2. Metodología y plan de trabajo

Esta memoria se divide en 6 capítulos incluyendo este capítulo introductorio. El segundo capítulo aborda una descripción del estado del arte del análisis de la expresión diferencial. Aquí se incluye la descripción de la tecnología más usada actualmente(RNA-Seq) para poder realizar este tipo de análisis (RNA-Seq), así como la teoría que subyace dentro de las herramientas estadísticas que lo llevan a cabo.

El capítulo 3 se corresponde con el método que se propone en este TFM para solventar el problema de encontrar genes específicos a una única condición experimental. Primeramente, se

describen los métodos a partir de los cuales se desarrolla la herramienta de selección de genes propuesta en este trabajo, seguidamente se presenta y explica el modelo propuesto.

Los capítulos 4 y 5 se corresponden con la explicación de las bases de datos utilizadas en este trabajo y los resultados obtenidos sobre las mismas, respectivamente. Por último, el capítulo 6 recopila las conclusiones derivadas de este trabajo y plantea algunas líneas de trabajo futuro.

2

Estado del arte

A lo largo de este capítulo se exponen los conceptos para entender el problema del análisis diferencial. Primero se centra en la parte teórica de la expresión diferencial para luego explicar el método más usado para llevar a cabo su análisis, el RNA-Seq.

2.1. Expresión diferencial

La expresión génica diferencial es el proceso mediante el cual las células son capaces de decodificar la información contenida dentro del material genético (ácidos nucleicos) para la elaboración del producto génico necesario para el buen funcionamiento del organismo [1]. Este producto génico suelen ser proteínas, aunque también podría ser ARN funcional.

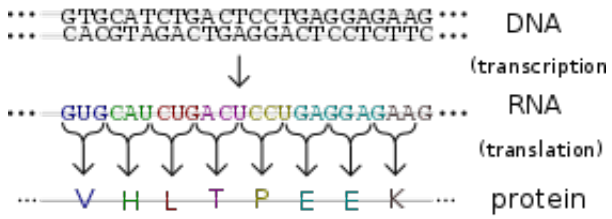

Figura 2.1: Proceso de formación de las proteínas [1].

El genotipo de cada célula en un organismo pluricelular es constante en todas ellas, es decir, todas contienen el mismo material genético. Si se analiza la información de una neurona del sistema nervioso, se puede observar que no contiene solo el código referente a esta funcionalidad, sino que tiene la suficiente información como para llevar a cabo cualquier otra funcionalidad del organismo. Esto se debe a que hay diferentes partes del genoma que se encuentran silenciadas en cada célula y por lo tanto no se generan las proteínas correspondientes a esas partes, ya sea debido al contexto local de las células o por causas externas. Por ejemplo, en el caso de la neurona, solo se expresan las proteínas necesarias para llevar a cabo las funciones nerviosas. Este proceso de inhibición y expresión de los genes viene dado por los intrones, que son los fragmentos en el genoma que no codifican proteínas pero tienen un papel muy importante en la regulación celular. Por otra parte al resto del genoma se le considera el exoma, y está formado por todas aquellas zonas encargadas de codificar proteínas.

Debido a esta diferencia de expresión de las células se produce el proceso denominado diferenciación celular. Todas las células de un organismo pluricelular provienen de una primera célula cigoto, a partir de la cual se desarrollan el resto de células de los tejidos gracias a las proteínas que codifica cada una.

2.2. RNA-Seq y la expresión diferencial

Gracias al surgimiento de las tecnologías de secuenciación de alto rendimiento o *Next Generation Sequencing* (NGS), el precio de la secuenciación de los genomas decayó significativamente a partir de 2007 (véase Figura 2.2) y produjo que RNA-Seq sustituyera a los microarrays como método de análisis para evaluar la expresión diferencial. A partir de la secuenciación RNA-Seq se puede contabilizar el número de secuencias que alinean con las zonas del genoma que interesen, generando una matriz con el número de secuencias generadas por gen y experimento.

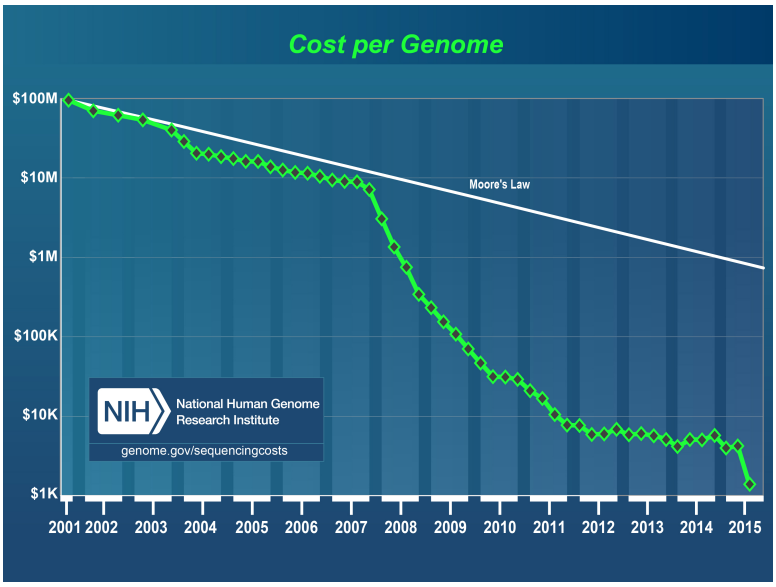

Figura 2.2: Coste de secuenciación del genoma humano [2].

El proceso general del RNA-Seq [3] se puede observar en la Figura 2.3. El punto de partida es el ARN mensajero, el cual contiene una cola de nucleótidos formada por bases de adenina (poliadenilación poly(A)). Este ARNm se ve fragmentado y convertido en una genoteca (*library* en inglés) de ADN complementario al cual se le añaden adaptadores de secuencia que servirán en el proceso de secuenciación. A partir de este momento se utilizan las tecnologías NGS (como por ejemplo Illumina [18] o SOLiD sequencing [19]) para secuenciar los fragmentos de cada ADNc y obtener las secuencias o *reads*, que se alinean con el genoma de referencia para clasificarlos en exones, secuencias de unión y secuencias poly(A). Por último se utilizan las posiciones de los fragmentos con respecto al genoma de referencia y su clasificación para obtener la expresión diferencial por cada uno de los genes que se quieran estudiar. Este proceso se repite para cada experimento, generando una columna asociada a él. Por ello al final se obtiene una matriz que tiene por filas los genes y en cada columna el conteo asociado a cada experimento.

Si se denota como G el conjunto formado por todos los genes que se quieren analizar, y se toma n como el número de experimentos realizados, la matriz de conteo de RNA-Seq será una matriz de dimensiones $G \times n$ que contiene el conteo de secuencias por gen y experimento (véase Figura 2.1). Dado un gen g, vamos a denotar y_{gi} como el número de secuencias asociadas al gen g en el experimento i.

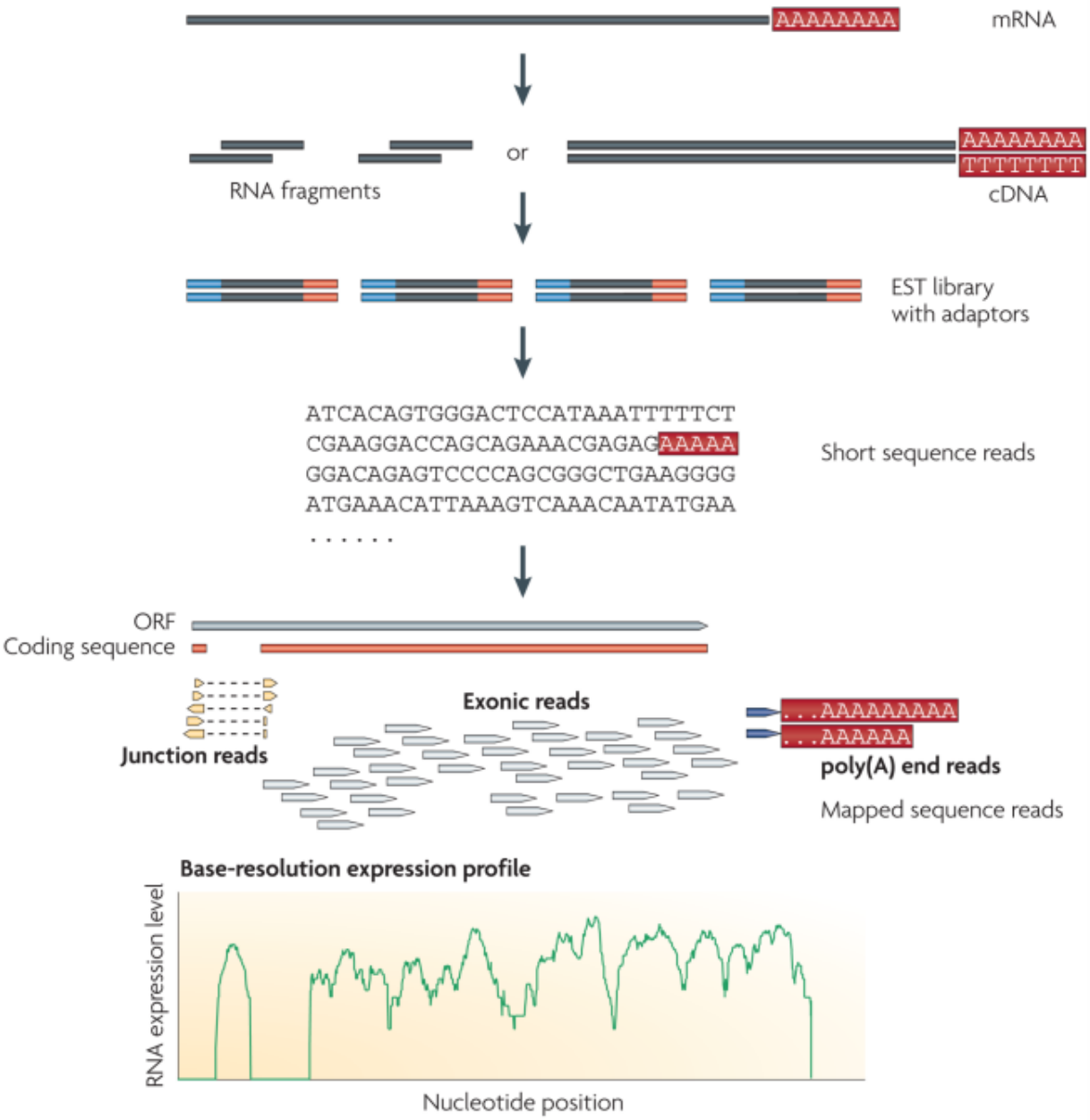

Figura 2.3: Experimento típico de RNA-Seq [3]. Como se explica en el texto, el RNAm se divide en diferentes fragmentos de ADNc los cuales tras el proceso de secuenciación y alineación se obtiene el número de secuencias por cada gen.

Tabla 2.1: Matriz de conteo de las secuencias

	Condición 1		**Condición 2**	
	Experimento 1	Experimento 2	Experimento 3	Experimento 4
Gen 1	205	207	365	324
Gen 2	0	1	0	0
Gen 3	567	551	578	540
...	...	...	...	...
Tamaño genoteca (*library depth*)	235689	235489	237154	229847

2.2.1. Normalización de la matriz de conteo

El conteo del número de secuencias que se alinean con cada gen no nos aporta directamente el nivel de expresión del mismo, ya que el número de secuencias también depende de más factores como podría ser la longitud del gen o las diferencias de tamaños en las genotecas. Por lo tanto es necesario un proceso de normalización para poder llevar a cabo la comparación entre los distintos experimentos. El método de normalización más típico es el llamado RPKM (*Reads Per Kilobase per Million*):

$$RPKM = y_{gi}/(l_g \times N_i/10^6), \tag{2.1}$$

con y_{gi} el número de secuencias del gen g en el experimento i, l_g la longitud del gen g y N_i el tamaño de la genoteca del experimento i.

Sin embargo es probable que en un experimento no se pueda acceder a las longitudes de los genes, para lo cual se utiliza otro proceso de normalización denominado CPM (*Counts Per Million*):

$$CPM = y_{gi} \times 10^6/N_i \tag{2.2}$$

Sin embargo, aunque estos métodos nos permitan comparar distintas genotecas pueden dar lugar a interpretaciones erróneas, ya que el valor normalizado de un gen depende de la proporción que ocupe con respecto a su genoteca. Por ejemplo, partiendo de una genoteca con dos genes y dos experimentos, supondremos que el primer gen no se expresa diferencialmente y tiene un conteo en los dos experimentos de x. Por otra parte el segundo gen tiene un conteo de y en el primer experimento y $2y$ en el segundo. Aunque en los dos experimentos el conteo del primer gen es el mismo, al normalizar, la proporción del primer gen en el segundo experimento va a ser menor que la proporción de ese mismo gen en el primer experimento. Esto se podría percibir como una inhibición del primer gen en el segundo experimento, aunque sería una conclusión errónea ya que partíamos del hecho de que no se expresaba diferencialmente.

Aunque los métodos descritos anteriormente son las más habituales, en la literatura podemos encontrar muchas más formas de llevar a cabo la normalización de los datos RNA-Seq [20–22].

2.2.2. Modelo de la binomial negativa

En un experimento de RNA-Seq podemos distinguir dos tipos de variabilidad. Por una parte la variabilidad biológica, derivada de la diferencia de expresión diferencial de un gen entre varios experimentos (aunque no haya expresión diferencial), y por otra parte la variabilidad técnica derivada de los errores de medida.

Asumiendo que conocemos π_{gi}, que es igual a la proporción de fragmentos de ADNc en la i-ésimo experiemento originados por el gen g y denotando como N_i al tamaño total de la genoteca del experimento i, el valor esperado del número de secuencias que expresarían el gen g conocido π_{gi}, sería $E[y_{gi}|\pi_{gi}] = \pi_{gi}N_i$. Se suele asumir que esta variable sigue una distribución de Poisson [23] por lo que $var(y_{gi}|\pi_{gi}) = \pi_{gi}N_i$. Esto representa la variabilidad técnica derivada de posibles errores en la medida.

Además, si consideramos que π_{gi} puede variar entre las muestras biológicas pero que su Cociente de Variación (CV) permanece constante tenemos que $E[\pi_{gi}] = \mu_{gi}$ y $var(\pi_{gi}) = \phi_g\mu_{gi}^2$ donde ϕ_g representa el cuadrado del CV y μ_{gi} representa la proporción media para un gen g y un experimento i. Por lo tanto la varianza incondicional de y_{gi} pasa a ser:

$$var(y_{gi}) = \mu_{gi} + \phi_g\mu_{gi}^2, \tag{2.3}$$

Ambos paquetes edgeR y DESeq utilizan el modelo de la binomial negativa, pero difieren en cómo calcular el factor de dispersión que se corresponde con el Cociente de Variación ($\sqrt{\phi_g}$). Por una parte edgeR lo calcula como una combinación lineal de pesos entre la dispersión específica de un gen y la dispersión total de todos los genes. En el caso del paquete DESeq la varianza la divide en dos: una estimación de Poisson del valor esperado del gen y un segundo término que modela la varianza biológica entre factores.

2.3. Expresión diferencial entre múltiples condiciones

El problema que se aborda en el TFM consiste en tratar de encontrar un método a partir del cual se puedan detectar aquellos genes que se expresan diferencialmente frente a una sola condición experimental, mientras que en el resto permanece constante.

El estudio de los métodos del estado del arte para la detección de genes diferencialmente expresados de forma específica por cada condición experimental muestra que este es aún un campo de investigación muy abierto. Se ha encontrado alguna publicación de soluciónes parecidas, pero no hay información de cómo se lleva a cabo el análisis ni la selección [13].

En general, el análisis de expresión diferencial con múltiples condiciones experimentales, es bastante más complejo que la comparación entre condición y control. Las paquetes principales como el EdgeR y el DESeq, están preparados para realizar análisis frente a múltiples condiciones gracias a la combinación de la binomial negativa y los modelos lineales generalizados, sin embargo, hay pocos estudios sobre sus resultados [6, 12, 24, 25]. Otros métodos combinan la binomial negativa para encontrar los genes diferencialmente expresados y luego estimaciones bayesianas para llevar a cabo la comparación entre pares [10, 26–28]. En concreto, DegPack es una solución web que te permite el análisis de expresión diferencial frente a múltiples condiciones a partir de la información mutua de las variables [29] con muy buenos resultados en la caracterización de los fenotipos.

3

Método propuesto: QPFS-LASSO

El objetivo de este TFM consiste en elaborar una herramienta que permita obtener los genes que se expresan diferencialmente en una sola condición experimental frente a la condición de control. Parte de la combinación del método QPFS modificado al problema genómico y del uso de regularizadores LASSO.

3.1. Selector de variables QPFS

El método QPFS [15] o *Quadratic Programming Feature Selection* es un método de selección de variables global que tiene en cuenta la dependencia que tienen las variables con respecto a la clase, y su independencia con respecto al resto de variables. QPFS es un método de selección de variables definido como un problema cuadrático sujeto a una serie de restricciones lineales en las variables a optimizar.

Suponiendo un problema de clasificación con C clases y N patrones que son vectores en $\mathbb{R}^{M+1}$ formados por el valor de M variables y su clase $c \in C$ correspondiente, se define F como el vector de entradas positivas que representa la dependencia o similitud de cada variable con respecto a la clase. Análogamente se define Q como la matriz simétrica de valores reales y positivos cuyas entradas se corresponden con la similitud entre pares de variables. El problema que intenta minimizar QPFS es el siguiente:

$$\min_{\mathbf{w}} \{\frac{1}{2}(1-\alpha)\mathbf{w}^T Q\mathbf{w} - \alpha\mathbf{w}^T F\}, \tag{3.1}$$

con las siguientes restricciones:

$$\begin{gathered} w_i > 0, \forall i = 1, ..., M \\ \sum_{i=1}^{M} w_i = 1 \end{gathered} \tag{3.2}$$

La Ecuación 3.1 trata por una parte de minimizar la similitud de variables (minimizar redundancias) y por la otra maximizar la dependencia de las variables con la clase (maximizar la relevancia).

El vector **w** es un vector de M coordenadas que representa el peso de cada variable en el problema, mientras que el parámetro α sirve para regular la importancia que se desea dar a la dependencia de las variables con la clase frente a la independencia de variables entre sí. En concreto si $\alpha = 0$, el segundo término de la Ecuación 3.1 se anula y los pesos solo se calculan teniendo en cuenta que se desea minimizar la similitud entre las variables. Por otro lado, si $\alpha = 1$ el coeficiente de la matriz del término cuadrático se cancela y, por tanto, solo se tiene en cuenta la dependencia de las variables con la clase en el problema de optimización.

El algoritmo original propone usar dos medidas de similitud diferentes. Como primera opción se puede usar el valor absoluto de la correlación de Pearson para medir las correlaciones lineales entre pares de variables (matriz Q) o cada variable con la clase (vector F). Como segunda alternativa se puede recurrir a la información mutua entre variables aleatorias que, aunque necesita de más datos para obtener resultados robustos, es capaz de encontrar dependencias no lineales entre pares de variables.

Cada conjunto de datos tiene un α óptimo que establecerá el balance adecuado entre el vector de relevancia F o la matriz de redundancia Q de cara a obtener los mejores aciertos en clasificación. Sin embargo, la obtención del valor óptimo del α es computacionalmente costosa en la práctica puesto que requiere entrenar y validar un clasificador (a elegir) para diferentes valores de α y distinto número de variables. Por ello, los autores de QPFS proponen un valor heurístico para α que evite esta sobrecarga computacional y que provea resultados competitivos. Este valor heurístico de α, $\hat{\alpha}$, equilibra el peso del término cuadrático y del término lineal en la Ecuación 3.1. Para ello, primero se estima la media $\bar{q}$ de todos los valores de Q y la media $\bar{f}$ de todos los valores de F de la siguiente manera:

$$\bar{q} = \frac{1}{M^2} \sum_{i=1}^{M} \sum_{j=1}^{M} q_{ij} \tag{3.3}$$

$$\bar{f} = \frac{1}{M} \sum_{i=1}^{M} f_i, \tag{3.4}$$

siendo q_{ij} el elemento de la matriz Q en la fila i y columna j y f_i el elemento del vector F en la posición i.

Y como se desea que los términos de la relevancia y la redundancia estén equilibrados, es razonable imponer $(1 - \hat{\alpha})\bar{q} = \hat{\alpha}\bar{f}$. Por tanto, la estimación propuesta para α viene dada por la siguiente ecuación:

$$\hat{\alpha} = \frac{\bar{q}}{\bar{q} + \bar{f}} \tag{3.5}$$

Para la resolución del problema de QPFS los autores sugieren su reformulación en un espacio de menor dimensión asumiendo que en problemas de alta dimensionalidad es probable que la matriz del término cuadrático Q no sea de rango máximo. Sin embargo, la proyección del problema de QPFS en un espacio de menor dimensión requiere de la diagonalización de la matriz Q, lo cual implica un coste cúbico respecto a la dimensión de Q. Para aliviar este problema, los autores sugieren el uso del método Nyström que permite obtener una estimación de los autovalores y autovectores de una matriz submuestreando un conjunto de filas de la misma y reduciendo el coste computacional de la diagonalización [30]. Sin embargo, no se va a emplear esta reformulación ya que, como se verá en la sección 3.3.4, la matriz Q del método propuesto es diagonal a bloques y se podrán paralelizar los cálculos.

3.2. Regularizadores LASSO

LASSO [31] (*Least Absolute Shrinkage and Selection Operator*) es un método de regresión que incorpora en su función objetivo tanto regularización como selección de variables. El proceso de regularización en el campo del aprendizaje automático hace referencia a la introducción de información adicional al modelo para impedir el *overfitting* o sobreajuste y mejorar así su capacidad de generalización. El principal objetivo de LASSO es el de mejorar las predicciones y la interpretabilidad de los modelos de regresión al seleccionar solo un subconjunto de las variables originales del problema.

Este método fue introducido por primera vez en el contexto de mínimos cuadrados, aunque se puede adaptar a otros métodos como por ejemplo los modelos lineales generalizados [32]. Sea y_i la salida asociada al vector de covarianzas $\mathbf{x_i} := (x_1, x_2, ..., x_M)$ para el caso i, se define la función objetivo de LASSO:

$$\begin{aligned} &\min_{w_0,\mathbf{w}} \{\frac{1}{N}\sum_{i=1}^{N}(y_i - w_0 - \mathbf{x_i}^T\mathbf{w})^2\} \\ &\text{sujeto a} \sum_{j=1}^{M}|w_j| \leq t \end{aligned} \tag{3.6}$$

El valor t se corresponde con el parámetro libre que determina el nivel de regularización. Se toma X como la matriz de covarianzas de tal forma que $X_{ij} = (\mathbf{x_i})_j$, es decir que el elemento X_{ij} se corresponde con la coordenada j del vector de covarianzas de la muestra i. Además, se suele trabajar asumiendo que las variables están centradas y las covarianzas normalizadas lo cual nos permite reescribir el problema de forma más compacta eliminando el w_0 dando lugar a:

$$\begin{aligned} &\min_{\mathbf{w}\in\mathbb{R}^M} \{\frac{1}{N}||\mathbf{y} - X\mathbf{w}||_2^2\} \\ &\text{sujeto a } ||\mathbf{w}||_1 \leq t, \end{aligned} \tag{3.7}$$

donde $||\mathbf{Z}||_p = (\sum_{i=1}^{N}|Z_i|^p)^{1/p}$ es la norma l_p [31], e $\mathbf{y}$ es un vector que contiene las clases.

Se puede probar que una forma equivalente de escribir el problema de optimización de la Ecuación 3.7 es usando su forma Lagrangiana [33]:

$$\min_{\mathbf{w}\in\mathbb{R}^M} \{\frac{1}{N}||\mathbf{y} - X\mathbf{w}||_2^2 + \lambda||\mathbf{w}||_1\}, \tag{3.8}$$

donde la relación entre los parámetros λ y t es dependiente del conjunto de datos.

LASSO introduce la norma l_1 dentro de la regresión, de tal forma que se produce esparcidad en las variables llevando a 0 el peso de un gran número de ellas. Cuanto mayor sea el parámetro λ más valores se irán a 0. Por eso es considerado no solo un método de regresión, si no también un método de selección de variables, ya que basta con considerar aquellas variables con peso positivo.

A partir de esta primera versión, se han ido creando diferentes variaciones del método para adaptarlo a problemáticas particulares. Solo se incluyen en este trabajo aquellas variaciones que se utilizan en el método propuesto.

3.2.1. Elastic Net

Elastic Net [34] es un método a partir del cual se intentan solventar algunos problemas de LASSO. Cuando el número de variables es mucho mayor que el número de muestras (M > N), la matriz de covarianzas $X^T X$ tiene a lo sumo rango N y es por tanto, singular. Debido a esto, LASSO puede seleccionar como máximo N variables. Además, LASSO tiende a seleccionar de cada subconjunto de variables muy correlacionadas solo una por grupo [31]. Para solucionar este tipo de problemas, Elastic Net añade una penalización con la norma l_2, de tal forma que que la matriz pueda dejar de ser singular y el problema tenga una solución mas óptima. La fórmula es la siguiente:

$$\min_{\mathbf{w}\in\mathbb{R}^M} \{\frac{1}{N}||\mathbf{y} - X\mathbf{w}||_2^2 + \lambda_1||\mathbf{w}||_1 + \lambda_2||\mathbf{w}||_2\} \tag{3.9}$$

Gracias a la norma l_2 se obtiene que el problema sea convexo y por tanto que la solución sea única debido a la existencia de un mínimo único. El método Elastic Net ha sido aplicado en un numerosos ejemplos: en SVM [35], en aprendizaje de métricas [36] e incluso en la selección y clasificación de genes y cánceres [37–39].

3.2.2. Group LASSO

En determinados problemas es necesario añadir cierta estructura dentro del espacio de variables como por ejemplo definir grupos de variables de acuerdo a cierto criterio propio del dominio y conocido a priori. Por ello, en el año 2006, se introdujo el método Group LASSO [40], el cual permitía que grupos predefinidos de las variables fueran seleccionadas todas juntas. Si una variable del grupo es incluida en la selección de variables, el resto del grupo también se verían incluidas. Uno de los problemas donde mejor se observa su eficacia es un problema con variables categóricas. Las variables categóricas no se pueden introducir directamente en todos los modelos, si no que en la mayoría de los casos tienen que ser convertidas a un grupo de variables binarias (*dummies*), de tal forma que cada variable categórica origina un grupo de variables binarias para representarla (una por cada posible valor). Si el modelo usa la matriz de covarianzas, no tendría sentido tener en cuenta solo las covarianzas de algunas de las variables *dummy*, se querría usar o todas las del grupo o ninguna. Dado un conjunto G de grupos, Group LASSO define el siguiente problema de optimización:

$$\min_{\mathbf{w}\in\mathbb{R}^M} \{||\mathbf{y} - \sum_{g=1}^{G} X_g\mathbf{w_g}||_2^2 + \lambda \sum_{g=1}^{G} ||\mathbf{w_g}||_{k_g}\} \tag{3.10}$$

donde se ha dividido el problema original según los diferentes grupos de G, X_g es la matriz de covarianzas asociada al grupo g y lo mismo con el vector $\mathbf{w_g}$. La norma l_2 en los grupos es la encargada de que se activen todos los pesos a la vez en cada grupo. Este tipo de algoritmos es muy útil, y también se han usado en el contexto de la expresión génica, teniendo como grupos aquellos genes con comportamiento similares [41–43].

3.2.3. Exclusive Group LASSO

Exclusive Group LASSO es un método basado en Group LASSO, pero que, al contrario que Group LASSO, busca la esparcidad dentro de los grupos gracias a la norma l_1. Por tanto, Exclusive Group LASSO aporta esparcidad intra-grupal de tal forma que solo se seleccionan

aquellas variables más relevantes dentro de cada grupo. Además, sobre los grupos se calcula la norma l_2 para evitar que grupos enteros se vayan a 0.

Sea G el conjunto de los grupos de variables predefinidos, el regularizador definido para Exclusive Group LASSO es el siguiente:

$$\forall \mathbf{w} \in \mathbb{R}^M, \Omega_{Eg}^G(\mathbf{w}) = \sum_{g \in G} ||\mathbf{w_g}||_1^2, \tag{3.11}$$

donde g son los distintos grupos de la partición y Ω_{Eg}^G es la norma l_1/l_2, que es igual a aplicar la norma l_1 sobre los elementos de cada grupo y la norma l_2 a la suma de cada grupo. La función objetivo del método Exclusive Group LASSO quedaría definida de la siguiente manera:

$$\min_{\mathbf{w} \in \mathbb{R}^M} f(\mathbf{w}) + \lambda \Omega_{Eg}^G(\mathbf{w}) \tag{3.12}$$

siendo $f(\mathbf{w})$ una función de coste convexa. En la figura 3.1 se ha utilizado como función de coste la función de mínimos cuadrados $f(w) = ||y - X^T||_2^2$ y se muestra la comparativa de los pesos $\mathbf{w}$ obtenidos por Group LASSO y Exclusive Group LASSO.

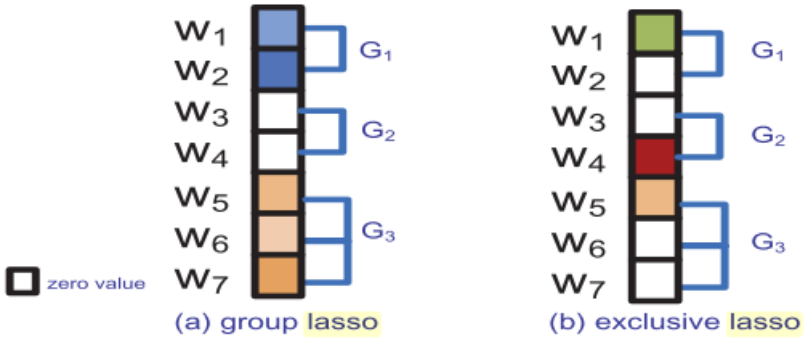

Figura 3.1: Diferencias entre los vectores solución de Group LASSO y Exclusive Group LASSO [44]. Como se puede observar en la imagen, en Group LASSO todos los elementos dentro de un mismo grupo con valor distinto a cero tienen un valor parecido para sus pesos, representados por colores similares. En cambio, en Exclusive Group LASSO solo se activa un peso por grupo, mientras que el resto de pesos del mismo grupo van a cero [44].

Supongamos un problema con cuatro variables ($M = 4$) agrupadas en sendos grupos de dos variables cada uno, perteneciendo las dos primeras variables al primer grupo g_1 y formando la tercera y cuarta variables el grupo g_2. Entonces, la regularización aplicada por Exclusive Group LASSO sería la siguiente:

$$\lambda \Omega_{Eg}^G(\mathbf{w}) = \lambda((|w_1| + |w_2|)^2 + (|w_3| + |w_4|)^2) \tag{3.13}$$

Exclusive Group Lasso es el regularizador que mejor se adapta al problema de selección de genes específicos. En este caso, cada grupo representaría un gen y estaría compuesto por los pesos asociados a ese gen para cada una de las condiciones experimentales. De esta forma, gracias a la esparcidad que aporta la norma l_1, se seleccionarían para cada gen las mejores condiciones experimentales mientras que el resto de condiciones tomarían el valor 0.

3.3. QPFS-LASSO

En esta sección se presenta el método propuesto para la selección de genes específicos cuando existen múltiples condiciones experimentales. La entrada del algoritmo será una matriz de conteo como la descrita en la Sección 2.2 en la que cada fila se corresponde con un determinado gen y las columnas representan las secuencias de varios experimentos (réplicas) por cada una de las condiciones experimentales. Por ejemplo, imaginemos que tenemos datos de diferentes tejidos humanos y para cada uno de ellos se dispone de una serie de secuenciaciones correspondientes a un tejido de control (sin tumor) y varias réplicas correspondientes a un tejido del mismo tipo con cáncer. Entonces, nuestro objetivo es encontrar aquellos genes que se expresen diferencialmente solo en una condición experimental (en este caso un tipo de tejido), y que en el resto de condiciones experimentales no haya diferencias con respecto a su control. Para ello vamos a utilizar el Exclusive Group LASSO y como función de coste una versión modificada del QPFS.

3.3.1. Implementación de la función de coste de QPFS

El algoritmo QPFS admite cualquier medida de similitud simétrica y positiva. En la versión original del algoritmo los autores sugieren utilizar la información mutua o la correlación de Pearson como medidas de dependencia entre variables; sin embargo, existen otras medidas de similitud más comúnmente utilizadas para medir la expresión diferencial de genes. En particular, se va a utilizar el estadístico S2N (signal-to-noise) como medida de similitud entre la expresión de distintos genes. S2N mide las distancias entre dos vectores $\mathbf{a}$ y $\mathbf{b}$ con medias μ_a y μ_b y desviaciones σ_a y σ_b respectivamente:

$$D_{ab} = \frac{\mu_a + \mu_b}{\sigma_a + \sigma_b} \tag{3.14}$$

Esta distancia se usa ampliamente dentro de la bioinformática para localizar genes que son diferencialmente expresados [8, 19, 45, 46].

Como se ha explicado anteriormente, QPFS trata de optimizar la siguiente función objetivo:

$$\min_{\mathbf{w}} \{\frac{1}{2}(1-\alpha)\mathbf{w}^T Q\mathbf{w} - \alpha\mathbf{w}^T F\}, \tag{3.15}$$

donde la matriz Q representa la dependencia entre variables y el vector F la relevancia de cada variable con la clase. Primero hay que calcular las medias y desviaciones de cada gen, agrupados en base a la condición experimental distinguiendo entre control y no control. En nuestro problema original tendremos las medias y las desviaciones para cada par tejido-cáncer o tejido-control. Además, debido a la falta de robustez de la media y la desviación típica cuando hay pocos datos se decidió calcular las medianas y las distancia a la mediana en su lugar.

Sea G el conjunto de genes a analizar y C el conjunto de tejidos o condiciones experimentales, entonces el vector $\mathbf{w}$ será un vector de longitud $G \times C$ en el que cada una de sus entradas representará el peso de un gen dentro de un tejido. Sin embargo hay que realizar algunas modificaciones al estadístico S2N:

1. **Cálculo del valor absoluto del S2N**. El cálculo de valor absoluto viene motivado por el interés de detectar tanto genes sobreexpresados como infraexpresados respecto a la condición de control. Además, tanto Q como F necesitan tener entradas positivas de acuerdo a la definición del método QPFS [15].

2. **Uso de una función sigmoide para escalar los valores**. En QPFS nos interesaba maximizar la relevancia y minimizar la redundancia. Sin embargo, en este nuevo problema seguimos queriendo maximizar la relevancia (valor de S2N entre la condición experimental y su condición de control), pero a la vez queremos maximizar el valor S2N entre las diferentes condiciones experimentales sin incluir las condiciones de control. Esto se debe a que se desea dar mayor peso a aquellos genes en los que la expresión tumoral de un determinado tejido, por ejemplo, sea significativamente distinta no solo a la condición de control si no también al resto de condiciones tumorales. Es así como se alcanzará la especificidad. Por lo tanto, como QPFS minimiza el valor de la matriz Q hay que aplicar primero una función que invierta los valores del S2N, convirtiendo valores altos en bajos y viceversa para que, minimizando Q se esté maximizando el S2N entre condiciones experimentales.

   Además, el estadístico S2N puede llegar a tomar valores demasiado altos que desvirtúan los resultados del algoritmo. Un valor mayor de 2 en el S2N es razonable para asumir un expresión diferencial y, a partir de este valor, ya no es tan relevante su magnitud como el hecho de que se haya producido la expresión diferencial. Por último, si aplicáramos una función únicamente a las entradas de la matriz Q, F y Q no escalarían de igual forma y su comparación en la formulación de QPFS (Ecuación 3.1) carecería de sentido. Por tanto, el posible cambio de escala que vayamos a realizar a Q también lo tiene que recibir F.

   Para solucionar todos los problemas anteriores se ha escogido entre las funciones sigmoides (véase la Figura 3.2) la tangente hiperbólica (tanh). La imagen de la función tanh para valores positivos es el intervalo $I = [0, 1]$. Si aplicamos la función tanh al vector F y la función $1 - \tanh$ a los valores de Q podríamos maximizar el vector F y minimizar la matriz Q garantizando a su vez un escalado comparable para los dos términos de la función objetivo de QPFS. Además la tangente hipérbolica converge a 1 a partir de valores cercanos al dos, luego nos sirve para escalar los valores muy altos.

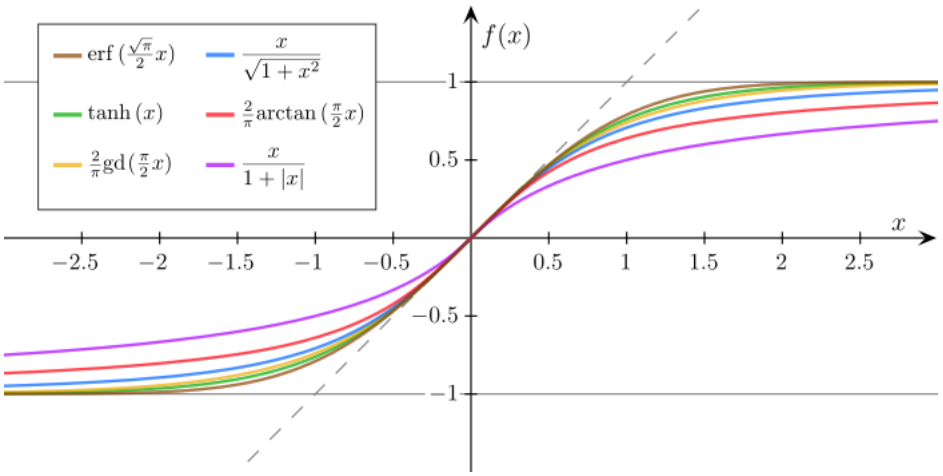

Figura 3.2: Algunas funciones sigmoides normalizadas para que estén acotadas entre el -1 y 1 [47].

A partir de las consideraciones anteriores y tomando μ' y σ' como la mediana y la desviación a la mediana, respectivamente, se definen F y Q como sigue:

$$F_{gc} = \tanh\left(\frac{|\mu'_{gc_{control}} - \mu'_{gc_{cancer}}|}{\sigma'_{gc_{control}} + \sigma'_{gc_{cancer}}}\right), g \in G, c \in C \tag{3.16}$$

$$Q_{g_1c_1,g_2c_2} = \begin{cases} 1 - \tanh\left(\frac{|\mu'_{g_1c_1{cancer}} - \mu'_{g_2c_2{cancer}}|}{\sigma'_{g_1c_1{cancer}} + \sigma'_{g_2c_2{cancer}}}\right) & \text{si } g_1 = g_2 \\ 0 & \text{si } g_1 \neq g_2 \end{cases} \tag{3.17}$$

La matriz Q es una matriz diagonal a bloques, ya que sus entradas solo toman valores no nulos cuando se compara un gen consigo mismo en las diferentes condiciones experimentales.

Finalmente, hay que tener en cuenta que se trata de resolver un problema convexo, por lo cual la matriz Q tiene que ser semidefinida positiva para asegurar la convergencia del método y que haya una solución óptima. Sin embargo Q no tiene por qué ser semidefinida positiva. No obstante, cabe destacar que todas las entradas de Q se encuentran en el intervalo $[0, 1]$ y los elementos de la diagonal son iguales a 1. Esta última propiedad se debe a que en la diagonal se calcula el valor del S2N entre un par gen-tejido consigo mismo, por tanto S2N=0 dado que la mediana del gen es idéntica para la misma condición experimental y entonces el elemento de la matriz se obtiene como 1-tanh(0)=1.

Además, como la matriz Q es simétrica por definición y todos sus valores son reales, la matriz Q es hermítica y se puede aplicar el siguiente teorema:

Teorema 1 *Sea A una matriz hermítica diagonalmente dominante, con todos los valores de la diagonal mayores o iguales a 0 entonces A es semidefinida positiva.*

La demostración del teorema puede consultarse en [48].

Solo tenemos que hacer que Q sea diagonalmente dominante, es decir que para todo elemento q_{ii} en la diagonal de Q se cumpla lo siguiente:

$$|q_{ii}| \geq \sum_j |q_{ij}|, \tag{3.18}$$

es decir, se ha de verificar que los valores de la diagonal sean mayores o iguales a la suma del resto de elementos de la fila. Como todos los valores de la fila Q_{gc} son 0 excepto las columnas asociadas al gen g y están acotadas por el valor 1, basta con poner como diagonal de Q el número de condiciones experimentales C para que Q sea semidefinida positiva. De esta forma, la entrada g_1c_1, g_2c_2 de la matriz Q correspondiente al gen 1-condición 1 frente al gen 2 - condición 2, queda definida como:

$$Q_{g_1c_1,g_2c_2} = \begin{cases} C & \text{si } g_1 = g_2 \wedge c_1 = c_2 \\ 1 - \tanh\left(\frac{|\mu'_{g_1c_1{cancer}} - \mu'_{g_2c_2{cancer}}|}{\sigma'_{g_1c_1{cancer}} + \sigma'_{g_2c_2{cancer}}}\right) & \text{si } g_1 = g_2 \wedge c_1 \neq c_2 \\ 0 & \text{si } g_1 \neq g_2 \end{cases} \tag{3.19}$$

La Figura 3.3 muestra una representación de la matriz Q y el vector F del algoritmo QPFS-LASSO.

3.3.2. Incorporación del término regularizador regularizador

Tal y como se describió en la Sección 3.2.3 (Ecuación 3.12), la función objetivo del modelo Exclusive Group LASSO se define como sigue:

$$\min_{\mathbf{w}\in\mathbb{R}^M} f(\mathbf{w}) + \lambda\Omega_{Eg}^G(\mathbf{w}), \tag{3.12}$$

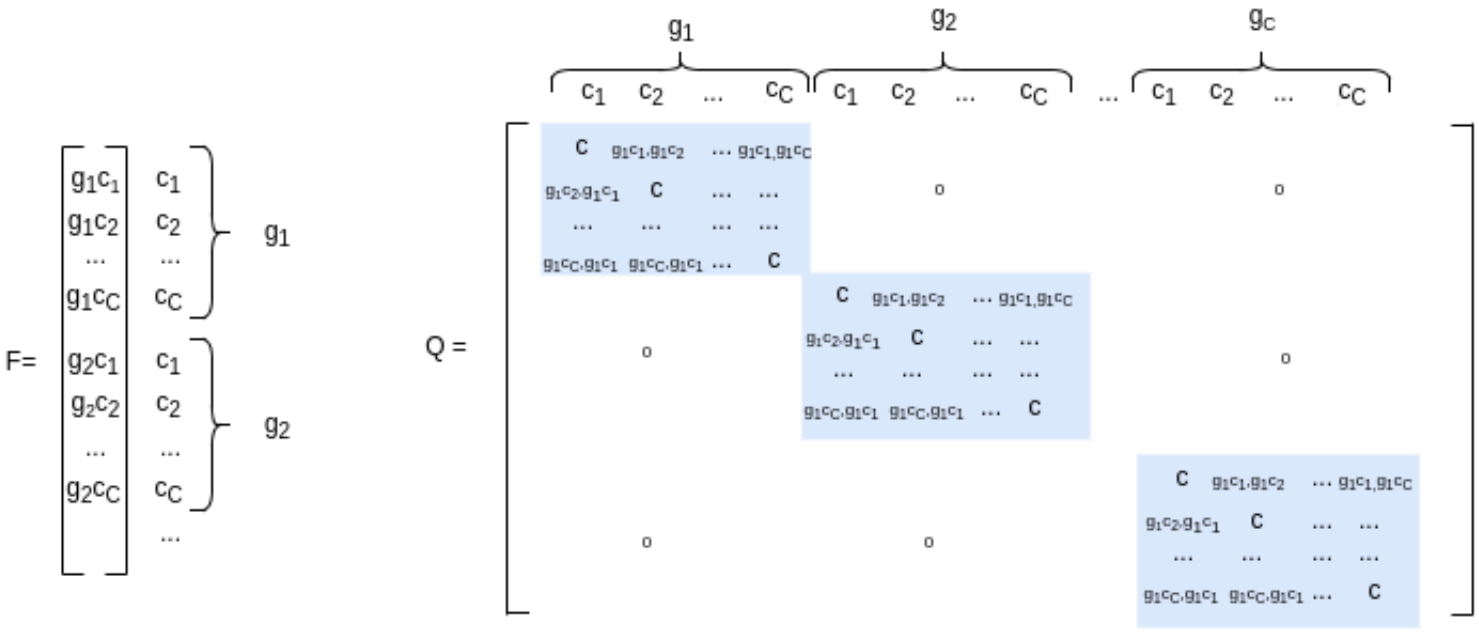

Figura 3.3: Matriz Q y vector F.

En QPFS-LASSO cada uno de los grupos de variables va a representar un gen, cada uno de los cuales, a su vez, tiene asociadas C condiciones experimentales. Es directo comprobar que el número de grupos es $N = \#(G)$, (uno por cada gen) y cada uno de éstos tendrá tamaño C. Así, el término regularizador para QPFS-LASSO pasa a ser:

$$\forall \mathbf{w} \in \mathbb{R}^{NC}, \Omega_{Eg}^{G}(\mathbf{w}) = \sum_{g \in G} ||\mathbf{w_g}||_1^2 \tag{3.20}$$

De esta forma nuestro problema queda definido por:

$$\min_{\mathbf{w} \in \mathbb{R}^{NC}} \{\frac{1}{2}(1-\alpha)\mathbf{w}^T Q \mathbf{w} - \alpha \mathbf{w}^T F + \lambda \sum_{g \in G} ||\mathbf{w_g}||_1^2\} \tag{3.21}$$

con las siguiente restricción:

$$w_i > 0, \forall i = 1, ..., NC \tag{3.22}$$

Se ha omitido la restricción que obligaba a que la suma de los pesos sea igual a 1, ya que la salida con los pesos del algoritmo se puede normalizar para que su suma sea igual a 1.

3.3.3. Penalización de la entropía

Al realizar pruebas sobre una versión inicial del algoritmo (Ecuación 3.21), se observó que había casos en los que el regularizador propuesto en Exclusive Group LASSO no era suficiente para potenciar la especificidad. Por ejemplo, el algoritmo asignaba pesos altos en casos en los que un gen se expresaba diferencialmente en varias condiciones experimentales con respecto a la condición de control, pero a su vez existía también diferencia de expresión entre las dos condiciones no control. Como ejemplo de este escenario, véase el panel de la izquierda de la Figura 3.4 en la que se representa la distribución y entropía del S2N de 10 condiciones experimentales respecto a la condición de control para tres genes diferentes. La asignación de pesos altos en

estos casos se debe, por un lado, a que como las condiciones experimentales (no control) tienen niveles de expresión significativamente diferentes, su valor de S2N es alto y, por tanto, la entrada correspondiente de la matriz Q es baja y, por tanto, favorable para la minimización del término cuadrático. Por otro lado, al haber expresión diferencial respecto a la condición de control en dos condiciones experimentales, la evaluación del vector de relevancia F en el término lineal de QPFS es mayor que en casos de una única condición experimental relevante. Para resolver este problema y potenciar más la identificación de genes específicos, se decidió incorporar información sobre el valor de la entropía de la distribución de S2N entre las condiciones experimentales y la condición de control (Figura 3.4) a la función objetivo de QPFS-LASSO. De acuerdo al objetivo inicial de este TFM, es preferible la selección del tercer gen (panel derecho) de la Figura 3.4 a la elección del primer gen (panel izquierdo) de la Figura 3.4.

La entropía mide la incertidumbre de una fuente de información. Matemáticamente, dado un mensaje X con C posibles estados cada uno con una probabilidad p_i, se define la entropía del mensaje $H(X)$ de la siguiente manera:

$$H(X) = -\sum_{i=1}^{C} p_i \log_2(p_i) \tag{3.23}$$

Para el problema de QPFS-LASSO, se define F'_g como un vector asociado al gen g que tiene como elementos el S2N de g por cada una de las C condiciones experimentales con respecto a la control. La entropía es máxima cuando todos los estados posibles son equiprobables. En el caso de los genes que se expresan solo en una condición, el valor de la entropía de F'_g es baja ya que todos los valores serían parecidos entre sí menos 1. Por otra parte según se exprese un gen en más condiciones, irá aumentando la incertidumbre del sistema provocando un aumento de la entropía. La introducción de la entropía en el algoritmo se explicará en la siguiente sección.

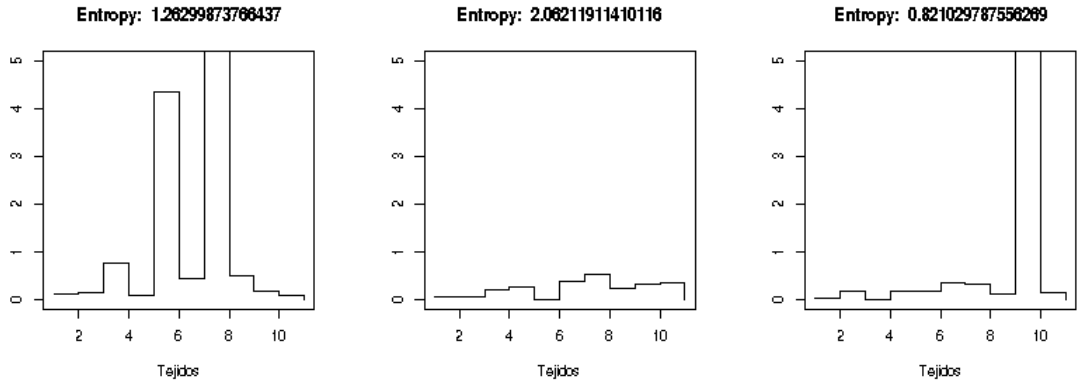

Figura 3.4: S2N de 10 condiciones experimentales (no control) respecto a la condición de control para tres genes diferentes. El tercer gen (derecha), solo se expresa en una condición experimental y es el que tiene mínima la entropía. Por otra parte, el segundo gen es uno de los casos peores, ya que casi todos los estados son iguales y hay mucha incertidumbre.

3.3.4. Relajación del modelo

La resolución del problema de QPFS-LASSO definido en las Ecuaciones 3.21 y 3.22:

$$\min_{\mathbf{w}\in\mathbb{R}^{NC}} \{\frac{1}{2}(1-\alpha)\mathbf{w}^T Q\mathbf{w} - \alpha\mathbf{w}^T F + \lambda \sum_{g\in G} ||\mathbf{w_g}||_1^2\} \tag{3.21}$$

con las siguiente restricción:

$$w_i > 0, \forall i = 1, ..., NC \tag{3.22}$$

requiere de la matriz Q que, para un problema de N genes y C condiciones experimentales, implica el cálculo de $(N \times C)^2$ entradas. El ser humano tiene más de 20.000 genes expresables y si por ejemplo se quisieran analizar 10 condiciones experimentales correspondientes a 10 tipos diferentes de cáncer, esto conllevaría el cálculo de $40 \cdot 10^9$ entradas de Q. Además, si asumimos que cada una de estas entradas ocupa en memoria 32 bits (número en coma flotante), se tendría un tamaño de matriz Q de aproximadamente 160 gigabytes, lo cual es inviable en la práctica en la mayoría de los casos.

Sin embargo, la matriz Q tiene la particularidad de estar definida a bloques de tamaño $C \times C$ debido a que cada gen solo se relaciona consigo mismo a lo largo de las condiciones experimentales y de control. Se define Q_g a la submatriz asociada al gen g. Del mismo modo se definen F_g y w_g como los subvectores de F y de $\mathbf{w}$ asociados al mismo gen g. De esta forma podemos convertir el problema anterior en uno más sencillo dependiente del número de genes:

$$\sum_{g \in G} \min_{\mathbf{w_g} \in \mathbb{R}^C} \{\frac{1}{2}(1-\alpha)\mathbf{w_g}^T Q_g \mathbf{w_g} - \alpha \mathbf{w_g}^T F_g + \lambda ||\mathbf{w_g}||_1^2\}, \tag{3.24}$$

Por tanto, se ha pasado de resolver un problema cuadrático de dimensión $N \times C$, a la resolución de N problemas cuadráticos de dimensión $C \times C$. Además, esta conversión del problema permite la paralelización del método y aumentar la velocidad considerablemente.

Por último, queda pendiente introducir un término de penalización de la entropía asociada a un gen en la función objetivo. Gracias a esta nueva definición del problema, la incorporación de la entropía en el método es mucho más clara. Habíamos definido F'_g al vector del valor absoluto del S2N que comparaba las condiciones experimentales con el control en el gen g.

Se decidió crear una matriz Q_{H_g} definida de la siguiente manera:

$$Q_{H_g} = \begin{cases} C & \text{si } g_1 = g_2 \wedge c_1 = c_2 \\ H\left(\tanh\left(F'_g\right)\right) & \text{en otro caso} \end{cases} \tag{3.25}$$

El uso de la tangente vuelve a ser por la estandarización de la medida. Esta matriz se suma directamente a la matriz Q_g de la Ecuación 3.24 penalizándola si el el gen g se expresa en múltiples condiciones. De esta forma, el algoritmo QPFS-LASSO queda definido como sigue:

$$\sum_{g \in G} \min_{\mathbf{w_g} \in \mathbb{R}^C} \{\frac{1}{2}(1-\alpha)\mathbf{w_g}^T \left(Q_g + Q_{H_g}\right) \mathbf{w_g} - \alpha \mathbf{w_g}^T F_g + \lambda ||\mathbf{w_g}||_1^2\}, \tag{3.26}$$

con las siguiente restricción:

$$w_{g_i} > 0, \forall i = 1, ..., C \tag{3.27}$$

No es objeto de este trabajo de Fin de Máster, el estudio de métodos de optimización para el problema de la Ecuación 3.26. Para su resolución, se utilizó el código facilitado por un miembro del Grupo de Aprendizaje Automático de la EPS-UAM que resolvía el problema de Elastic Net, al cual se le añadieron funcionalidades como QPFS y la estimación del parámetro α así como su adecuación a la ecuación 3.26.

4

Conjuntos de datos

En este capítulo se explican las diferentes bases de datos que se han usado en la validación del modelo. Se utilizaron simulaciones para poder controlar la expresión diferencial de cada uno de los genes y y así tener una base sobre la que validar los algoritmos desarrollados. Por otra parte, también se evaluó el rendimiento de QPFS-LASSO sobre datos reales utilizados en trabajos anteriores con el fin de determinar la aplicabilidad de la técnica propuesta en escenarios reales.

4.1. Datos simulados

La mayor parte de los datos utilizados en este trabajo son simulaciones de experimentos de RNA-seq ya que de esta forma podemos controlar directamente las matrices de expresión diferencial y por tanto disponer de unos datos validados para comprobar la efectividad de nuestros algoritmos. En concreto se van a llevar a cabo simulaciones de diferentes tejidos cancerosos así como las condiciones de control asociados a ellos.

4.1.1. Proceso de generación de las simulaciones

Para la elaboración de las simulaciones se ha seguido el proceso completo de un experimento de RNA-Seq, exceptuando la parte biológica de la secuenciación de las cadenas de nucleótidos:

1. **Creación de la matriz de expresión diferencial**: Para ello primero hay que diseñar todos los experimentos. Todas las simulaciones creadas parten del genoma de la bacteria E. Coli, de la cual se tiene un fichero con todo su genoma así como varios archivos GTF con la información de sus genes. El formato GTF o *Gene Transfer Format*, es un formato de datos que nos brinda información sobre la estructura de los genes, como puede ser su posición o su longitud.

   Un aspecto importante en los experimentos es ver si se van a generar muestras de control que sean aplicables a todas las condiciones experimentales o si cada condición experimental va a tener sus propios experimentos de control. Ya sabemos que la expresión diferencial de una célula en un tejido puede ser diferente de la que tiene otra en otro tejido. Por ejemplo, si tenemos experimentos secuenciados de una célula cancerígena de un tejido hay

que compararlos con experimentos de control de ese mismo tejido y no de otro. De esta forma se han llevado a cabo dos tipos de simulaciones: la primera asume que solo hay un tejido y por lo tanto solo es necesario un tipo de experimentos de control, mientras que en la segunda cada condición experimental necesitará de su propia condición de control.

Una vez definido el tipo de experimento a llevar a cabo se calcula la matriz de *fold-changes* donde por cada gen y tipo de experimento se incluye el ratio entre la media de la expresión diferencial de la condición en cuestión respecto a la media de la condición de control. En la tabla 4.1 se muestran ejemplos de matrices de expresión simuladas. En las simulaciones es necesario especificar la probabilidad de que un gen se exprese diferencialmente, es decir que deje de tomar el valor de fold-change de 1 y pase a tomar cualquiera de los siguientes valores $\{1/5, 1/3, 1/2, 1, 2, 3, 5\}$.

a)	**Control**	**Condición 1**	**Condición 2**	**Condición 3**
Gen 1	1	1	2	3
Gen 2	1	4	1	1
Gen 3	1	1	0.3	1
...	...	...	...	...

b)	**Condición 1**		**Condición 2**		**Condición 3**	
	Control	Estímulo	Control	Estímulo	Control	Estímulo
Gen 1	1	1	2	2	1	3
Gen 2	1	4	1	1	1	1
Gen 3	1	1	1	0.3	1	1
...	...	...	...	...	...	...

Tabla 4.1: Ejemplos de los dos posibles casos de simulación: los recuadros en rojo indican sobreexpresión mientras que los azules indican infraexpresión. En el caso a) todas las condiciones de experimentales tienen la misma condición de control mientras que en el caso b) existe una condición de control distinta para cada condición experimental.

2. **Generar las secuencias a partir de la matriz de expresión**: Para generar las secuencias de los experimentos hemos usado el paquete *polyester* [17] de R que nos permite emular un secuenciador NGS para simular experimentos de RNA-Seq usando la binomial negativa. Esta función recibe el fichero GTF que contiene la información de los genes así como las matrices de expresión descritas en el punto anterior. Además se le puede indicar el tipo de secuencias que se desea generar (*single-end* o *paired-end*) y el número de experimentos o réplicas se quiere simular por cada columna de la matriz (condición experimental). En concreto se han decidido los siguientes parámetros: secuencias de tipo *single-end* (solo se generan secuencias de una sola hebra de la cadena nucleótida), longitud de las secuencia igual a 100 y una cobertura de 20 veces el genoma. Si aumentáramos el parámetro de la cobertura tendríamos un mayor número de secuencias final, pero el tiempo de ejecución se vería incrementado bastante.

   Polyester genera un fichero FASTA por cada experimento que contiene todas las secuencias asociadas a él. En el formato FASTA cada secuencia está formada por dos líneas: una línea de cabecera y otra con las cadenas de nucleótidos simuladas. Si quisiéramos realizar una simulación con C condiciones experimentales, cada una con su propio control, y quisiéramos n réplicas por cada una de ellas, se generarían $2nC$ ficheros FASTA.

3. **Añadir el *quality score***: El *quality score* es la medida de la calidad de la identificación de cada nucleótido en el proceso de secuenciación. Se define a partir de la probabilidad P

de error de identificación de una base o nucleótido:

$$Q = -10 \log_{10}(P) \tag{4.1}$$

Estos valores son necesarios para muchos algoritmos de alineación de secuencias, por lo que son simulados para cada una de las secuencias obtenida en el paso 2 y, tanto la secuencia como sus *quality scores* son incorporados a ficheros del tipo FASTQ[1].

4. **Alinear las secuencias con respecto a un genoma de referencia**: Cada secuencia generada tiene una longitud de 100 nucleótidos, y es necesario determinar de qué parte del genoma procede para así determinar qué genes se están expresando. Para ello hemos usado el programa Bowtie [50] , dado que es uno de los alineadores más utilizados en la actualidad [51]. A partir del indexado del genoma con la transformada de Burrows-Wheler, el alineador Bowtie permite una búsqueda eficiente en el genoma de cada secuencia y devuelve un fichero SAM en el que muestra información sobre las posiciones de alineación de cada secuencia junto con otro tipo de información.

5. **Realizar el conteo de los genes**: Una vez tenemos los ficheros SAM es necesario recurrir a una herramienta que en base a las posiciones que ocupa cada gen en el genoma, genere la matriz de conteo de secuencias para llevar a cabo el análisis diferencial (Figura 2.3). Este proceso se suele llevar a cabo mediante alguna herramienta públicamente disponible como HTSeq. Sin embargo, se ha utilizado una versión alternativa que soluciona algunos problemas que tiene el HTSeq para realizar los conteos [52].

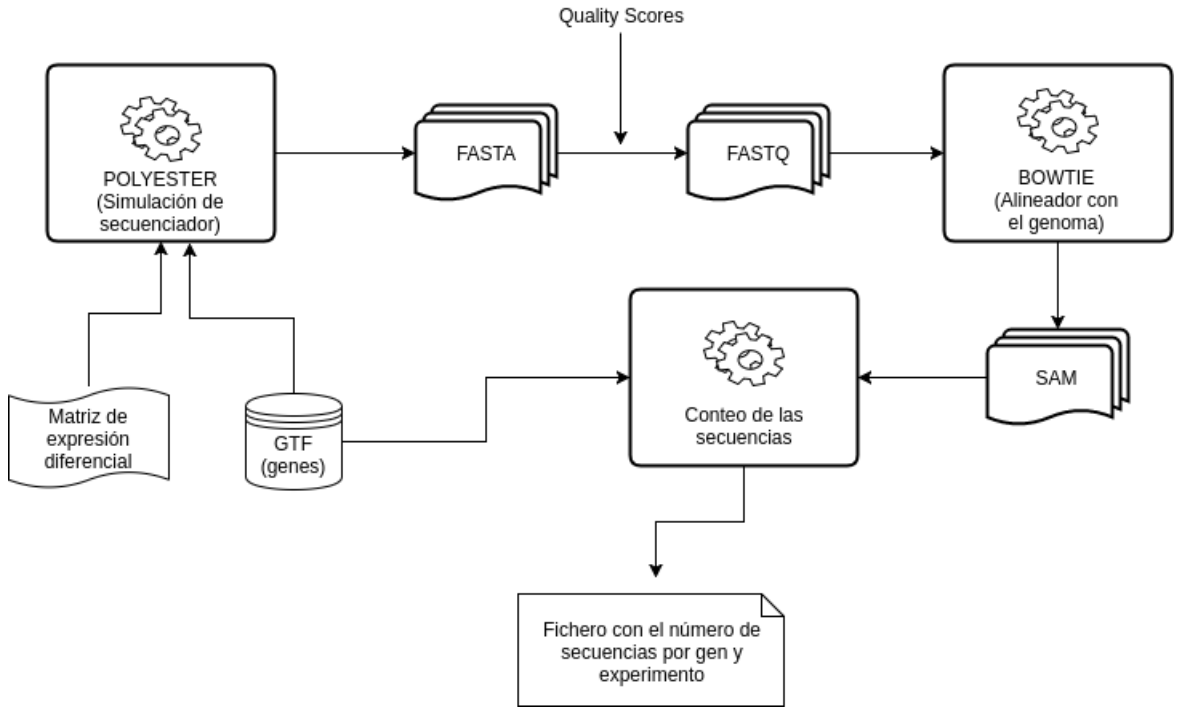

Figura 4.1: *Pipeline* para la simulación de experimentos RNA-Seq.

4.1.2. Simulaciones

Se han generado dos simulaciones diferentes, ambas con los mismos 1000 genes de los 4321 genes que tiene la bacteria E.colli.

[1]No es objeto de este TFM entrar en los detalles del funcionamiento de estos alineadores y de los formatos de los ficheros. Para más detalle, consúltese [49].

- **Simulación 1**: En esta simulación todas las condiciones experimentales compartían los mismos datos de control. Se simularon 10 condiciones experimentales y una de control, y por cada una se realizaron 10 réplicas. En la Tabla 4.2 se contabiliza el número de genes de los 1000 genes considerados en función del número de condiciones experimentales (tejidos) en los que se expresan diferencialmente. Como son datos simulados, esta información se obtiene directamente de la matriz de *fold-changes*. Nuestro objetivo es que el algoritmo implementado sea capaz de detectar los genes que solo se expresan en una condición experimental. La Tabla 4.3 contabiliza el número de genes que solo se expresan diferencialmente en una condición experimental en base a los distintos niveles de sobreexpresión/infraexpresión considerados en la Simulación 1.

Número tejidos donde se produce expresión	**Número de genes**	**Porcentaje**
$c = 0$	137	13.7 %
$c = 1$	266	26.6 %
$c = 2$	275	27.5 %
$c = 3$	172	17.2 %
$c = 4$	100	10 %
$c = 5$	39	3.9 %
$c = 6$	7	0.7 %
$c = 7$	2	0.2 %
$c = 8$	1	0.1 %
$c = 9$	1	0.1 %

Tabla 4.2: Simulación 1: Conteo de genes según el número de tejidos donde se expresan. En total se tienen 1000 genes de los cuales la mayor parte se expresa en un solo tejido o dos.

$n = 1/5$	$n = 1/3$	$n = 1/2$	$n = 2$	$n = 3$	$n = 5$	**Total**
42	50	57	40	40	37	266

Tabla 4.3: Número de genes por cada nivel/ratio de expresión diferencial dentro del subconjunto de los genes que se expresan en un solo tejido en la simulación 1.

- **Simulación 2**: En esta simulación cada condición experimental tiene sus propios experimentos de control. Además se han escogido los mismos genes que en el apartado anterior, pero cambiando las matrices de expresión. La información correspondiente a esta simulación se encuentra en las tablas 4.4 y 4.5.

4.2. Datos reales

4.2.1. TCGA

El Atlas del Genoma del Cáncer o TCGA (The Cancer Genome Atlas) es un proyecto iniciado en 2005 con el principal objetivo de catalogar los cambios moleculares resposables de la aparición de cáncer haciendo uso de la secuenciación genómica y la bioinformática [11]. Una de las mayores aportaciones de este proyecto consiste en la creación de un portal de datos donde la mayor parte son abiertos y accesibles para el público general.

Dentro de esta plataforma se pueden encontrar datos de secuenciación procedentes de distintos tipos de tecnologías: RNA-Seq, WSX, miRNA-Seq... Como se he explicado a lo largo del

Número tejidos donde se produce expresión	**Número de genes**	**Porcentaje**
$c = 0$	98	9.8 %
$c = 1$	231	23.1 %
$c = 2$	262	26.2 %
$c = 3$	198	19.8 %
$c = 4$	125	12.5 %
$c = 5$	58	5.8 %
$c = 6$	24	2.4 %
$c = 7$	4	0.4 %

Tabla 4.4: Simulación 2: Conteo de genes según el número de tejidos donde se expresan. En total se tienen 1000 genes de los cuales la mayor parte se expresa en un solo tejido o dos.

$n = 1/5$	$n = 1/3$	$n = 1/2$	$n = 2$	$n = 3$	$n = 5$	**Total**
35	32	41	32	45	46	266

Tabla 4.5: Número de genes por cada nivel/ratio de expresión diferencial dentro del subconjunto de los genes que se expresan en un solo tejido en la simulación 2.

trabajo, el algoritmo está planteado para recibir datos procedentes de RNA-Seq y en la base de datos de TCGA se pueden encontrar datos da casi 40 tipos de cáncer diferentes. Se han escogido aquellos datos que, además de tener experimentos de muestras etiquetadas como cáncer, también tengan experimentos de control. La base de datos final estudia 12 tipos de cánceres y 20.533 genes. La información correspondiente a la base de datos TCGA se muestra en la tabla 4.6.

4.2.2. MNIST

Debido a que la aplicabilidad del algoritmo QPFS-LASSO no se limita únicamente al campo de la bioinformática y puede catalogarse dentro del área de selección de variables en aprendizaje automático, se decidió incluir alguna base de datos característica este campo [53]. La base de datos MNIST contiene las imágenes en blanco y negro de aproximadamente 42.000 números escritos a mano por diferentes personas. Las imágenes son de tamaño 28×28 píxeles y el valor del pixel puede variar entre el 0, que sería correspondiente al color blanco, y el 255 asociado al color negro. Además, cada imagen está etiquetada con el valor del número al que representa (0-9). Cabe destacar que en esta base de datos, la distribución del número de muestras por cada clase es cercana a la uniforme ya que todos los números tienen aproximadamente el mismo número de muestras. En la figura 4.2 se observa una de las imágenes asociada al número 7.

La transformación del problema de la expresión diferencial a la selección de píxeles se realizaría de la siguiente manera. Las condiciones experimentales en el problema génico pasan a ser las etiquetas de los datos de MNIST. Por otra parte, el algoritmo querría detectar aquellos pixeles que son representativos para sólo un número mientras que en el resto carecen de importancia. Además, habría que incorporar a la base de datos unas muestras que representen el control frente las condiciones experimentales. Se han realizado pruebas o bien considerando la media de todas las muestras como muestra de control, o considerando una imagen en blanco como condición de control (todos los valores a 0).

Nombre	**Abreviación**	**Muestras cancerosas**	**Muestras de control**
Carcinoma urotelial de la vejiga	BLCA	408	19
Cáncer mamario invasivo	BRCA	1097	114
Cáncer de colon	COAD	286	41
Carcinoma de células escamosas de cabeza y cuello	HNSC	520	44
Cromófobo renal	KICH	66	25
Carcinoma renal de células claras	KIRC	533	72
Carcinoma renal de células capilares	KIRP	290	32
Carcinoma hepatocelular	LIHC	371	50
Adenocarcinoma pulmonar	LUAD	515	59
Carcinoma pulmonar con células escamosas	LUSC	503	51
Adenocarcinoma de próstata	PRAD	497	52
Carcinoma en tiroides	THCA	505	59

Tabla 4.6: Número de muestras de cáncer y control en la base de datos TCGA.

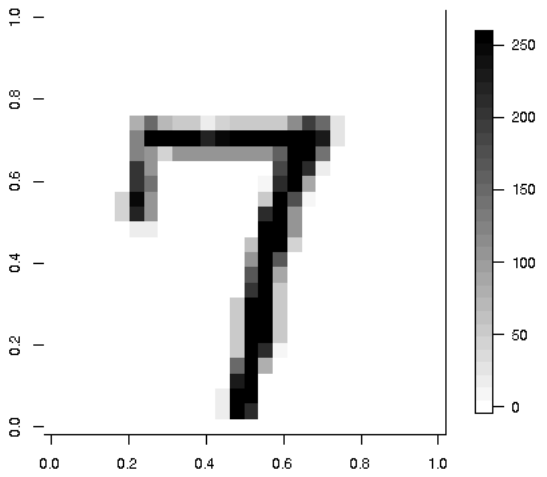

Figura 4.2: Ejemplo de imagen del número 7.

5 Resultados

En este capítulo se muestran los resultados obtenidos por el algoritmo QPFS-LASSO al aplicarlo sobre los conjuntos de datos descritos en el Capítulo 4.

5.1. Datos simulados

Las pruebas se realizaron primero con los datos simulados. En la primera simulación se estudiaba el comportamiento de 1000 genes teniendo una única condición de control. Como en total se analizaban 10 condiciones experimentales, QPFS-LASSO calculó entre las 10000 posibles combinaciones gen-condición, aquellas que maximizaran su expresión diferencial con respecto al control y el número de tejidos en los que se expresa cada gen. En la figura 5.1 se observa la salida del algoritmo para $\lambda = 0{,}01$, y en ella se visualizan los 10000 pares gen-condición (ordenados por tejido (condición experimental) y luego por el peso asignado por el algoritmo QPFS-LASSO). El valor de α se determinó de manera empírica. El rango óptimo de α si situó en el intervalo $I = [0{,}01, 0{,}08]$ donde el orden de los genes que devolvía el algoritmo permanecía constante, a partir de ese valor se producía *underfitting* llevando genes correctos a 0 y por debajo de 0.01 apenas se regulariza, lo cual no es deseable.

Cabe recordar que en esta simulación había 266 genes que se expresaban solo en una única condición y que, por lo tanto, se les considera correctos y se denotan como DE-1. Para la evaluación del rendimiento del algoritmo se utilizarán tres métricas. El *recall* mide la proporción de genes correctos que escogería el algoritmo dado un número de genes o un umbral dado. La *precisión* por su parte, consiste en la proporción de genes que son correctamente descritos como diferencialmente expresados por el algoritmo entre el número total de genes que el algoritmo considera correctos; y por último el tasa de acierto. nos indica el acierto total del modelo.

En la Figura 5.2 se representan las tres medidas descritas anteriormente en función del número de genes seleccionados por el algoritmo. La precisión es del 100 % para los 114 primeros genes, y a partir de este punto empieza a decaer. Este resultado es bueno, ya que indica que los 114 genes con más peso del algoritmo se corresponden con genes que solo se expresan en una sola condición experimental (DE-1). A partir de ese punto, el modelo empieza a incluir genes que no cumplen esta restricción. La curva de *recall* por su parte, crece linealmente en los 114 primeros genes para luego tomar un crecimiento más lento. El *recall* es máximo en la recta ya que es

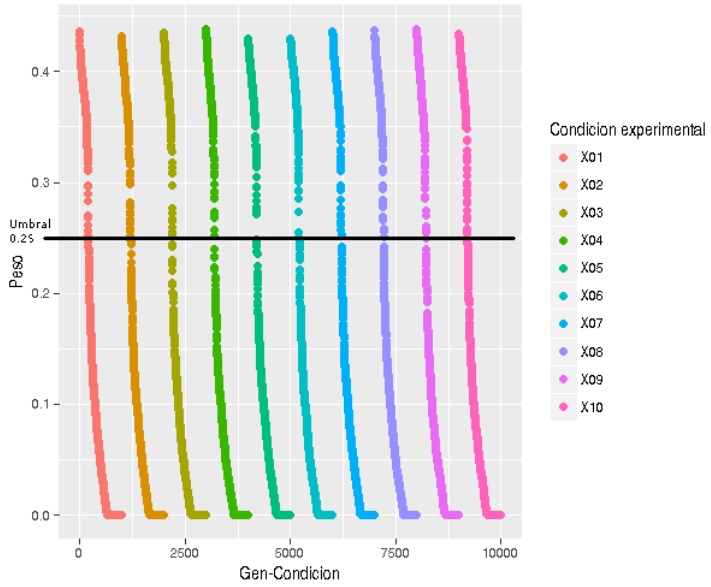

Figura 5.1: Valor de los pesos para la simulación con una única condición de control y $\lambda = 0{,}01$. Los valores se encuentran agrupados por condición experimental y luego de mayor a menor.

imposible que valga más que la identidad al compararlo con el número de genes escogidos. Por último, el valor máximo de acierto del modelo se encuentra en el máximo de la *accuracy*. donde se equilibra el *recall* y la *precision*. Por último, el valor máximo de acierto del modelo es del 82 % y se alcanza con 110 genes seleccionados. Sin embargo, los resultados de precisión decrecientes a partir de los primeros 114 genes indican que se dan casos de genes que se expresan en varias condiciones experimentales y que tienen un peso mayor a otros genes que solo se expresan en una. Esto se puede deber a diferentes razones, como una desviación a la mediana muy alta que implica un valor bajo en el S2N o genes cuya expresión es muy baja siempre (menos de 10 secuencias por gen).

En la mayor parte de estos casos, estos genes que se expresan en varias condiciones tienen un fold-change muy alto (5) en alguna condición y valores de 2 o 0.5 en el resto, lo cual produce que este gen sea considerado como muy relevante por el término lineal del algoritmo QPFS-LASSO y, a su vez, se considere que las expresiones entre distintas condiciones experimentales también son significativamente distintas. Estos resultados no son desfaborables, ya que en la realidad nunca se va a dar el caso idílico de un gen que no se exprese nada en el resto de las condiciones,por lo tanto es importante que el algoritmo de un peso alto a este tipo de genes.

La Tabla 5.1 muestra el número de genes seleccionados entre los 114 primeros en base a su nivel de expresión diferencial. Se puede observar como la mayor parte de los genes seleccionados tienen un nivel de expresión de 5, es decir, se expresan 5 veces más que su condición de control. El nivel de expresión que le es más difícil caracterizar al algoritmo es el nivel $n = 1/2$, debido a que hay mucha menos diferencia entre condición y control.

	$n = 1/5$	$n = 1/3$	$n = 1/2$	$n = 2$	$n = 3$	$n = 5$	**Total**
DE-1	42	50	57	40	40	37	266
114 mejores genes	23	14	9	16	23	29	114
Porcentaje(%)	55 %	28 %	16 %	40 %	58 %	78 %	43 %

Tabla 5.1: Correspondencia entre todos los genes DE-1 y los 114 genes primeros del algoritmo.

Además, en la figura 5.3 se observa la evolución el número de genes según el tipo de expresión.

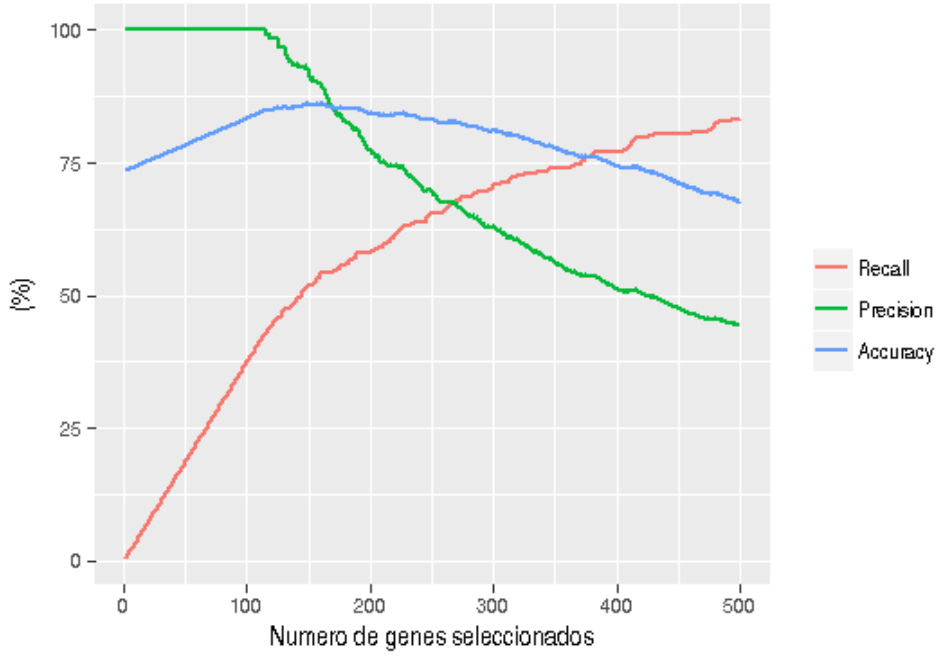

Figura 5.2: Valor de los pesos para la simulación con una única condición de control y $\lambda = 0{,}01$. Los valores se encuentran agrupados por condición experimental.

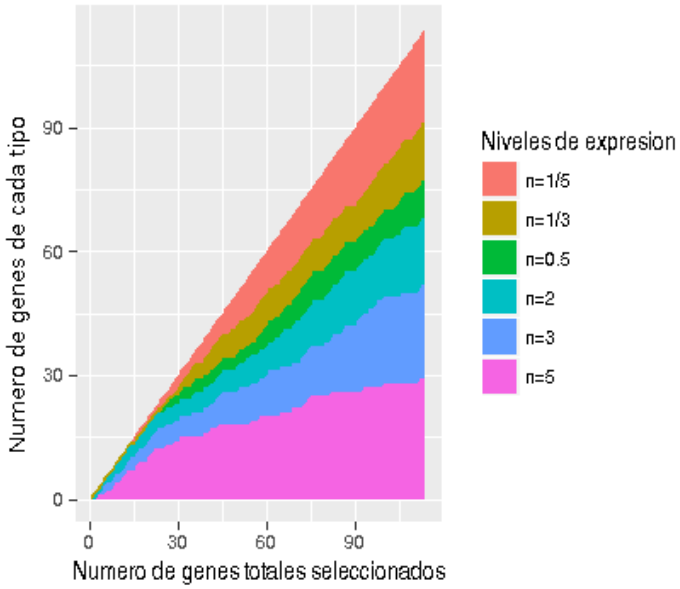

Figura 5.3: Evolución del número de genes por nivel de expresión. En el eje de abscisas representa el número de genes que selecciona el algoritmo, mientras que el eje de ordenadas se representa el número de genes según el nivel de expresión diferencial.

También se han analizado los datos de la segunda simulación, donde cada condición de experimental tenía su propia muestra de control. Sin embargo, los resultados han sido muy parecidos a la primera simulación: no ha afectado al algoritmo que las condiciones de control fueran diferentes entre sí, ya que ninguno de esos genes se seleccionaba por el algoritmo.

5.2. Datos reales

Los experimentos con datos reales, no nos aportan un análisis tan directo como es el caso de las simulaciones, donde partimos del hecho que sabemos qué genes se expresan y en qué proporción. No obstante, para asegurar la aplicabilidad del algoritmo es necesario evaluar el rendimiento del modelo en casos reales.

5.2.1. TCGA

Una vez obtenidos la base de datos TCGA se le aplicó el QPFS-LASSO. A partir de ahí se analizaron los pesos y se escogieron 30 genes por cada condición experimental (los 30 mejores genes de cada una, Figura 5.4). El objetivo era demostrar que el comportamiento de esos genes en su condición experimental era radicalmente distinto con respecto al resto de condiciones. Como se observa en la figura, los valores de los pesos asociados a las condiciones BLCA, BRCA, HNSC y PRAD son relativamente más bajos que el resto de condiciones, por lo que se puede intuir que esas condiciones experimentales tendrán peores resultados que el resto.

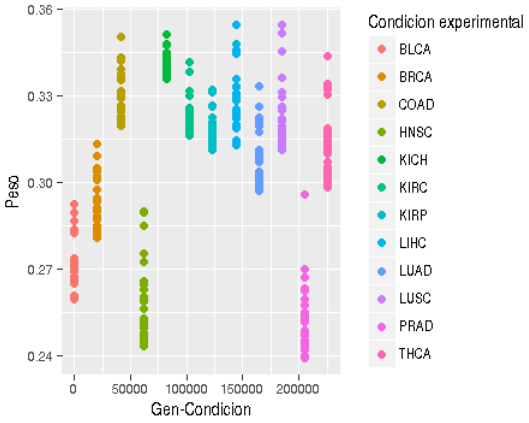

Figura 5.4: Pesos de las variables gen-condición obtenidos por el algoritmo QPFS-LASSO para el conjunto de datos TCGA con lambda=0.06

La figura 5.5 representa la expresión diferencial de los genes escogidos anteriormente en muestras aleatorias de los datos. En el eje de las abscisas se representan 240 experimentos con cáncer (20 de cada condición experimental) escogidos aleatoriamente. Los experimentos se encuentran agrupados por condición experimental. Por otra parte, en el eje de las ordenadas se representa por cada condición experimental, los 30 mejores genes seleccionados por el algoritmo para dicha condición. El código de colores nos permite asociar el tipo de condición experimental con sus mejores genes. Para representar la expresión diferencial, se ha decidido mostrar los valores normalizados de conteo de las secuencias en cada experimento. Para llevar a cabo esta normalización, se ha creado una muestra por cada condición experimental formada por 15 condiciones de control y 15 cánceres aleatorios. Seguidamente se calcularon las medias y desviaciones de cada gen y con ellas se normalizaron los valores de la matriz. Cabría esperar como resultado una matriz diagonal a bloques en la que la diagonal está compuesta por submatrices 20 (muestras) x 30 (genes) cuyos valores en cada fila (muestra) sean diferentes que para el resto de condiciones experimentales. Eso significaría que ese gen se comporta diferente en esa condición frente al resto e indicaría un buen funcionamiento del algoritmo. Un valor oscuro en la matriz indicaría un valor más bajo de lo normal, lo que se interpreta como una inhibición del gen. Al contrario, un valor rojizo querría significar una sobreexpresión del gen.

En la Figura 5.5 se observa que, efectivamente, al considerar los 30 primeros genes seleccionados por QPFS-LASSO para cada condición experimental, aparecen submatrices diferenciadas en la diagonal de la matriz principal. Estas submatrices son de color azul oscuro, lo que indica

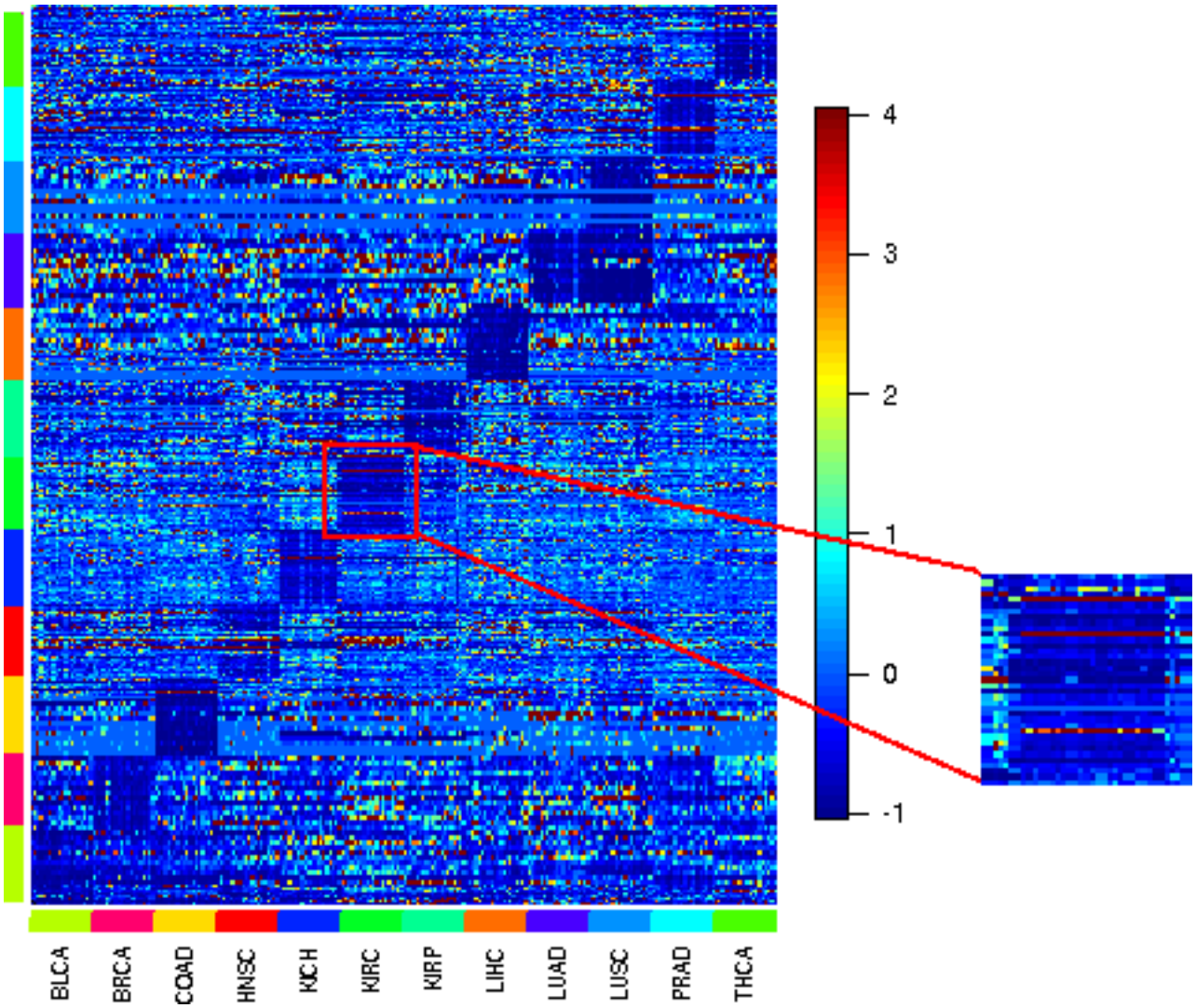

Figura 5.5: Matriz con el valor de expresión de los mejores genes por cada condición. En el eje de las abscisas tenemos 20 muestras por cada tejido de forma secuencial. Por otra parte en el eje de las ordenadas tenemos los mejores 30 genes de cada tejido. Se ha ampliado la zona correspondiente a los experimentos de KIRC, y se puede apreciar que de los 30 genes escogidos, 27 se infraexpresan y 3 se sobreexpresan (SAP30, EGLN3 y PHKA2).

que la mayor parte de los genes detectados por el algoritmo son genes que se infraexpresan en esa condición experimental. Además dentro de las submatrices de la diagonal, se pueden visualizar filas totalmente rojas, asociadas a un gen que se sobreexpresa para esa condición. Es el caso de los genes CLNK en KICH o los genes SAP30, EGLN3 y del PHKA2 en KIRC. Si observamos las condiciones LUAD y LUSC se visualiza una zona de infraexpresión pronunciada cuando nos restringimos a los niveles de expresión de los genes seleccionados para LUAD sobre los datos de LUSC. Ambas condiciones experimentales hacen referencia a cánceres pulmonares, lo que podría indicar que los genes que se expresan en LUSC son mucho más específicos que los de LUAD de acuerdo a la función objetivo de QPFS-LASSO. También se verifica lo esperado al ver la distribución de los pesos en la Figura 5.4. Las filas correspondientes a los genes de las condiciones BLCA y BRCA son mucho más difusas que el resto. En concreto la submatriz asociada a BLCA casi no se distingue del resto de condiciones experimentales.

Para comprobar la validez de los resultados, se ha recurrido a la búsqueda manual de información sobre los genes resultantes de la salida del algoritmo. A través de páginas como

http://www.proteinatlas.org que contienen información de todo el genoma humano se ha observado que efectivamente existen diferencias en estos genes con respecto al resto de condiciones. Más concretamente, en la Figura 5.6 se observa la información de expresión asociada al gen EGLN3 y como su valor en el grupo KIRC es mucho más elevado que en el resto. Por último hay que tener en cuenta que sólo se han utilizado 12 condiciones experimentales, por lo que es probable que algún gen detectado por el algoritmo que solo se expresa en una de las 12 condiciones experimentales consideradas, se exprese también en algún tipo de cáncer no incluido en este estudio.

En el Anexo A, se muestran los 30 mejores genes de cada condición experimental, así como su ratio de expresión calculado con S2N. En él se puede observar como el ratio de expresión de los genes de BLCA es muy bajo en comparación con el resto, lo que coincide con el valor bajo de sus pesos y las filas de la matriz de expresión. Por otra parte KICH es el que tiene ratios de expresión más altos. Además, gracias a estas gráficas, podemos observar que hay genes que son totalmente característicos de algunos tejidos, ya que la expresión diferencial en el resto de tejidos es 0, como por ejemplo el CLCA1 en el cáncer de colon (COAD) o el F9 en el cáncer de hígado (LIHC).

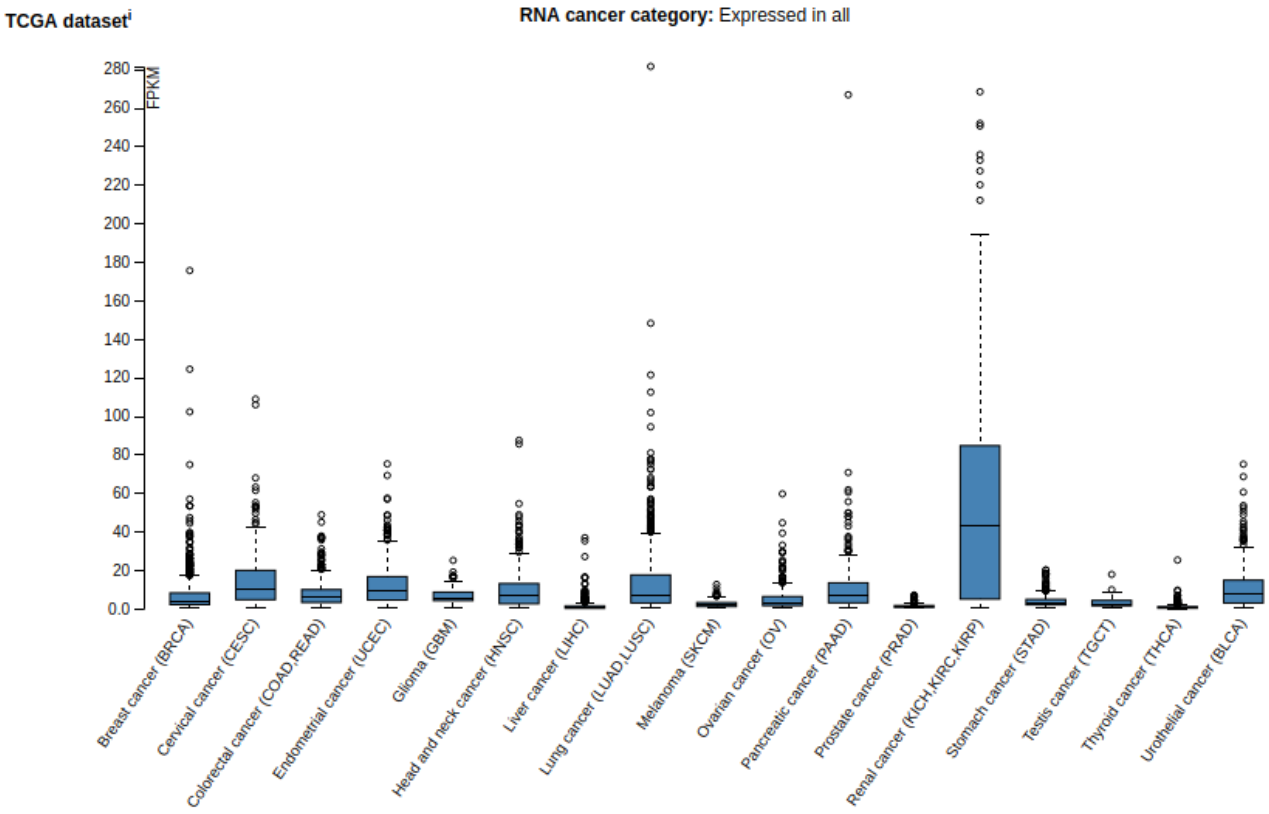

Figura 5.6: Análisis de expresión RNA para el gen EGLN3 [54].

5.2.2. MNIST

Como se indicó en el capítulo anterior, se decidió incorporar una base de datos no biológicos para observar si el algoritmo podría ser usado en otros campos diferentes de la bioinformática. El problema se convertiría en encontrar aquellos píxeles que son muy característicos en la caligrafía de un determinado dígito, mientras que son poco específicos para el resto de números. El resultado del algoritmo es muy visual ya que al irse seleccionando los píxeles más característicos de cada número, se puede obtener una huella del mismo.

En las Figura 5.7 y 5.8 se pueden observar respectivamente los 20 y 50 mejores pixeles escogidos por cada número (valores en negro). En concreto, si se analiza el caso de los 20 mejores píxeles, se ve que la mayor parte de ellos no se repiten; es decir, se detectan aquellos que no son comunes pero sí que son característicos del número. En la Figura 5.9 se encuentran solapados los píxeles de la Figura 5.7. En total se muestran 118 píxeles de los cuales 90(76 %) no se repiten. Además hay 24 pixeles que se repiten en dos números distintos y 4 píxeles que se repiten en 3 números. Está claro el éxito del algoritmo en este problema. Teniendo en cuenta solo 50 o incluso 20 píxeles por número, se pueden intuir visualmente los números mientras que se ha minimizado la redundancia de los pixeles entre ellos.

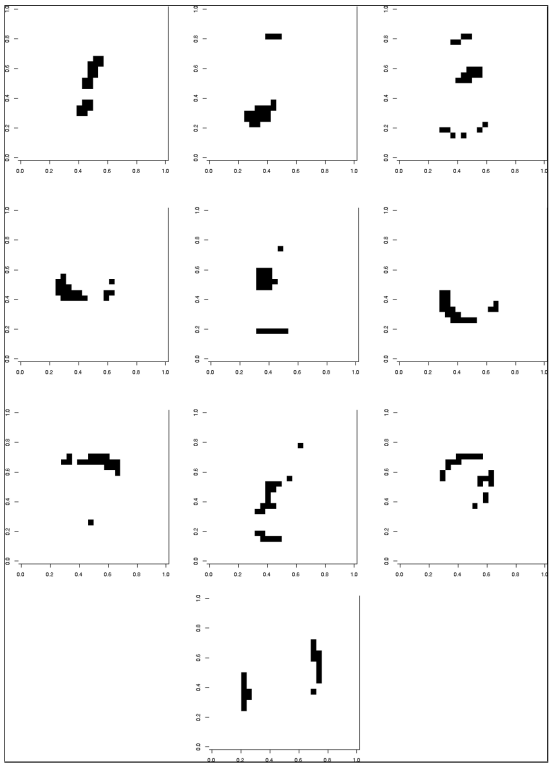

Figura 5.7: Representación de los 20 mejores píxeles para cada dígito del dataset MNIST

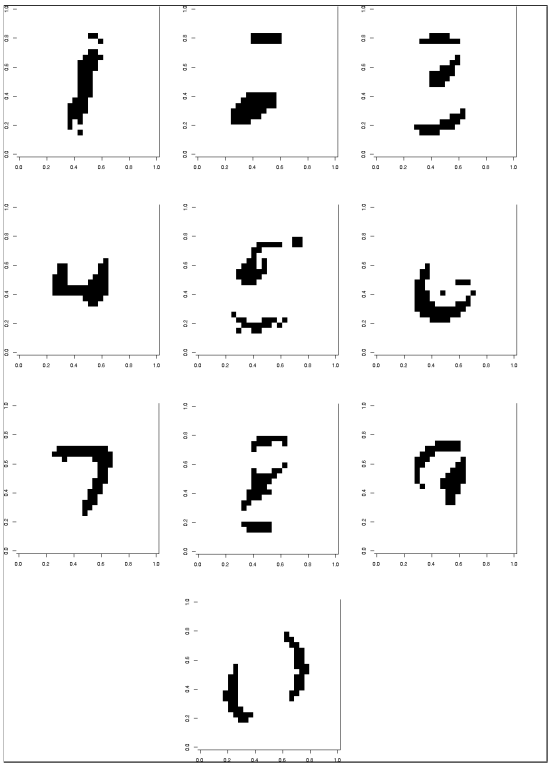

Figura 5.8: Representación de los 50 mejores píxeles para cada dígito del dataset MNIST

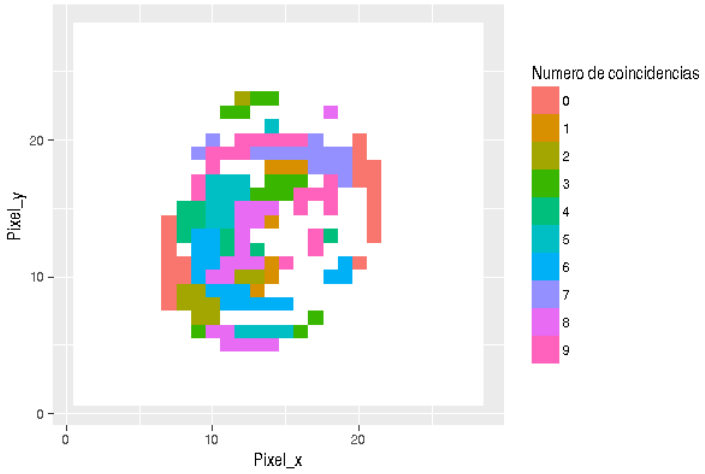

Figura 5.9: Imagen con los 20 mejores píxeles de cada número solapados.

6

Conclusiones y trabajo futuro

La selección de variables específicas por clase y su adecuación al problema de genes diferencialmente expresados en una única condición son problemas de investigación abiertos en las área del aprendizaje automático y la bioinformática. Por ello, el objetivo de este Trabajo de Fin de Máster ha sido la propuesta e implementación de un nuevo método de selección de atributos que permite, a partir de datos RNA-Seq, detectar genes diferencialmente expresados específicos para una determinada condición experimental. Este modelo, no solo es aplicable a datos de origen biológico, si no que es válido para cualquier otro problema de selección de variables que tenga el mismo objetivo de detectar variables especialistas para una determinada clase/condición.

El algoritmo surge de la unión del selector de variables QPFS y el regularizador Exclusive Group LASSO, y recibe el nombre de QPFS-LASSO. Este algoritmo se centra en buscar los genes específicos maximizando el estadístico S2N entre las condiciones experimentales y control a la vez que maximiza el S2N de las condiciones experimentales entre sí. El programa principal del algoritmo se ha desarrollado en el lenguaje R debido a que la mayor parte de las herramientas de análisis diferencial más utilizadas en la actualidad se encuentran implementadas en este lenguaje. Por otra parte, dado que se quería crear un algoritmo que fuese relativamente rápido, la resolución de los problemas cuadráticos se ha implementado en C. Además se ha incorporado la posibilidad de que gran parte del código se ejecute en paralelo, aumentando aún más este factor.

Los resultados obtenidos, tanto en datos RNA-Seq simulados como en datos reales, nos han permitido validar el buen funcionamiento del método. Gracias a los datos simulados se observa que el algoritmo tiene una gran precisión, sobre todo en los genes identificados como más relevantes, alcanzando tasas de precisión del 100 %. Además se han podido extraer una serie de conclusiones a partir de los datos reales. La aplicación del algoritmo a experimentos de expresión diferencial para tumores malignos obtenidos de la base de datos TCGA (*The Cancer Genome Atlas*), ha permitido detectar que la mayor parte de los genes que han resultado ser específicos en cada tipo de cáncer se infraexpresan frente a su condición de control. Además se han observado relaciones entre algunas condiciones experimentales como en el caso de los tipos de cáncer LUAD (Adenocarcinoma pulmonar) y LUSC (Carcinoma pulmonar con células escamosas). Finalmente, también se ha comprobado la efectividad del algoritmo en un conjunto de datos del campo del procesamiento de imágenes consistente en la clasificación de dígitos manuscritos. La aplicación del algoritmo QPFS-LASSO sobre estos datos ha permitido identificar píxeles de imágenes que son específicos para cada uno de los dígitos del cero al nueve.

6.1. Trabajo futuro

Este proyecto deja abiertas diferentes vías de trabajo futuro. La primera línea consistiría en modificar la medida de similitud utilizada (S2N) de forma que no exista un sesgo que haga que el algoritmo seleccione con mayor preferencia los genes sobreexpresados respecto a la condición de control frente a los genes infraexpresados, hecho que se ha observado en los resultados obtenidos.

Además hay varios aspectos que se podrían mejorar en el algoritmo QPFS-LASSO. Por ejemplo, se podría llevar a cabo la detección de genes "generalistas" que se expresen en todas las condiciones experimentales. Para ello, habría que cambiar la función objetivo del algoritmo propuesto, actualmente centrada en la detección de genes "especialistas".

Finalmente, cabe destacar que se trabaja en la creación de un paquete en R que agrupe todo el código desarrollado y permita la distribución del algoritmo de una manera fácil y sencilla.

Glosario de acrónimos

- **NGS**: Next Generation Sequencing
- **RNA-Seq**: Técnica de secuenciación
- **ARNm**: ARN mensajero
- **ADNc**: ADN complementario, sintetizado a partir del ARNm
- **CV**: Cociente da variación
- **QPFS**: Quadratic Programming Feature Selection
- **LASSO**: Least Absolute Shrinkage and Selection Operator
- **S2N**: Signal to noise
- **DE-1**: Genes diferencialmente expresados en una sola condición

Bibliografía

[1] Wikipedia. Expresion genica, 2017.

[2] National Human Genome Research Institute. `https://www.genome.gov/27565109/the-cost-of-sequencing-a-human-genome/`.

[3] Zhong Wang, Mark Gerstein, and Michael Snyder. Rna-seq: a revolutionary tool for transcriptomics. *Nat Rev Genet*, 10(1):57–63, Jan 2009. 19015660[pmid].

[4] Simon Anders and Wolfgang Huber. Differential expression analysis for sequence count data. *Genome Biology*, 11(10):R106, Oct 2010.

[5] Simon Anders, Davis J. McCarthy, Yunshun Chen, Michal Okoniewski, Gordon K. Smyth, Wolfgang Huber, and Mark D. Robinson. Count-based differential expression analysis of rna sequencing data using r and bioconductor. *Nat. Protocols*, 8(9):1765–1786, Sep 2013. Protocol.

[6] Yunshun Chen, Aaron T. L. Lun, and Gordon K. Smyth. *Differential Expression Analysis of Complex RNA-seq Experiments Using edgeR*, pages 51–74. Springer International Publishing, Cham, 2014.

[7] C. Evans, J. Hardin, M. Huber, D. Stoebel, and G. Wong. Differential expression analysis for multiple conditions. *ArXiv e-prints*, October 2014.

[8] Sridhar Ramaswamy, Pablo Tamayo, Ryan Rifkin, Sayan Mukherjee, Chen-Hsiang Yeang, Michael Angelo, Christine Ladd, Michael Reich, Eva Latulippe, Jill P. Mesirov, Tomaso Poggio, William Gerald, Massimo Loda, Eric S. Lander, and Todd R. Golub. Multiclass cancer diagnosis using tumor gene expression signatures. *Proceedings of the National Academy of Sciences*, 98(26):15149–15154, 2001.

[9] Andrew D. Fernandes, Jean M. Macklaim, Thomas G. Linn, Gregor Reid, and Gregory B. Gloor. Anova-like differential expression (aldex) analysis for mixed population rna-seq. *PLOS ONE*, 8(7):1–15, 07 2013.

[10] Davis J. McCarthy, Yunshun Chen, and Gordon K. Smyth. Differential expression analysis of multifactor rna-seq experiments with respect to biological variation. *Nucleic Acids Research*, 40(10):4288–4297, 2012.

[11] Li Peng, Xiu Wu Bian, Di Kang Li, Chuan Xu, Guang Ming Wang, Qing You Xia, and Qing Xiong. Large-scale rna-seq transcriptome analysis of 4043 cancers and 548 normal tissue controls across 12 tcga cancer types. 5:13413 EP –, Aug 2015. Article.

[12] Min Tang, Jianqiang Sun, Kentaro Shimizu, and Koji Kadota. Evaluation of methods for differential expression analysis on multi-group rna-seq count data. *BMC Bioinformatics*, 16(1):360, Nov 2015.

[13] UC San Diego. `http://ucsdnews.ucsd.edu/pressrelease/new_uc_san_diego_biosensor_will_guard_water_supplies_from_toxic_threats`.

[14] Barbara B. Pineda-Bautista, J.A. Carrasco-Ochoa, and J. Fco. MartÄśÌĄnez-Trinidad. General framework for class-specific feature selection. *Expert Systems with Applications*, 38(8):10018 – 10024, 2011.

[15] Irene Rodriguez-Lujan, Ramon Huerta, Charles Elkan, and Carlos Santa Cruz. Quadratic programming feature selection. *J. Mach. Learn. Res.*, 11:1491–1516, August 2010.

[16] Yang Zhou, Rong Jin, and Steven ChuâĂŞHong Hoi. Exclusive lasso for multi-task feature selection. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 988–995, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.

[17] Alyssa C. Frazee, Andrew E. Jaffe, Ben Langmead, and Jeffrey T. Leek. Polyester: simulating rna-seq datasets with differential transcript expression. *Bioinformatics*, 31(17):2778–2784, 2015.

[18] Michael A Quail, Iwanka Kozarewa, Frances Smith, Aylwyn Scally, Philip J Stephens, Richard Durbin, Harold Swerdlow, and Daniel J Turner. A large genome center's improvements to the illumina sequencing system. *Nature methods*, 5(12):1005–1010, 2008.

[19] Vicki Pandey, Robert C. Nutter, and Ellen Prediger. Applied Biosystems SOLiD??? System: Ligation-Based Sequencing. In *Next Generation Genome Sequencing: Towards Personalized Medicine*, pages 29–42. 2008.

[20] Marie-Agnes Dillies, Rau Andrea, Aubert Julie, Hennequet-Antier Christelle, Jeanmougin Marine, Servant Nicolas, Keime Celine, Marot Guillemette, Castel David, Estelle Jordi, Guernec Gregory, Jagla Bernd, Jouneau Luc, Laloe Denis, Le Gall Caroline, Schaeffer Brigitte, Le Crom Stephane, Guedj Mickael, and Jaffrezic Florence. A comprehensive evaluation of normalization methods for illumina high-throughput rna sequencing data analysis. *Briefings in Bioinformatics*, 14(6):671–683, 2013.

[21] Soohyun Lee, Chae Hwa Seo, Byungho Lim, Jin Ok Yang, Jeongsu Oh, Minjin Kim, Sooncheol Lee, Byungwook Lee, Changwon Kang, and Sanghyuk Lee. Accurate quantification of transcriptome from rna-seq data by effective length normalization. *Nucleic Acids Research*, 39(2):e9, 2011.

[22] Mark D. Robinson and Alicia Oshlack. A scaling normalization method for differential expression analysis of rna-seq data. *Genome Biology*, 11(3):R25, Mar 2010.

[23] John C. Marioni, Christopher E. Mason, Shrikant M. Mane, Matthew Stephens, and Yoav Gilad. Rna-seq: An assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res*, 18(9):1509–1517, Sep 2008. 18550803[pmid].

[24] Travers Ching, Sijia Huang, and Lana X. Garmire. Power analysis and sample size estimation for rna-seq differential expression. *RNA*, 20(11):1684–1696, Nov 2014. RA[PII].

[25] Michael I. Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome Biology*, 15(12):550, Dec 2014.

[26] Thomas J. Hardcastle. Generalized empirical bayesian methods for discovery of differential data in high-throughput biology. *Bioinformatics*, 32(2):195–202, 2016.

[27] Thomas J. Hardcastle and Krystyna A. Kelly. bayseq: Empirical bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics*, 11:422–422, Aug 2010. 1471-2105-11-422[PII].

[28] Xiaobei Zhou, Helen Lindsay, and Mark D. Robinson. Robustly detecting differential expression in rna sequencing data using observation weights. *Nucleic Acids Research*, 42(11):e91, 2014.

[29] Jaehyun An, Kwangsoo Kim, Heejoon Chae, and Sun Kim. Degpack: A web package using a non-parametric and information theoretic algorithm to identify differentially expressed genes in multiclass rna-seq samples. *Methods*, 69(3):306 – 314, 2014. Recent development in bioinformatics for utilizing omics data.

[30] C. Fowlkes, S. Belongie, and J. Malik. Efficient spatiotemporal grouping using the nystrom method. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–231–I–238 vol.1, 2001.

[31] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[32] Sara A. van de Geer. High-dimensional generalized linear models and the lasso. 36, 05 2008.

[33] E. M. Lifshitz Lev Davidovich Landau. *Teoria clasica de los campos.* 1973.

[34] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67:301–320, 2005.

[35] Li Wang, Ji Zhu, and Hui Zou. The doubly regularized support vector machine. *Statistica Sinica*, pages 589–616, 2006.

[36] Meizhu Liu and Baba C. Vemuri. A robust and efficient doubly regularized metric learning approach. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part IV*, ECCV'12, pages 646–659, Berlin, Heidelberg, 2012. Springer-Verlag.

[37] Jacob J. Hughey and Atul J. Butte. Robust meta-analysis of gene expression using the elastic net. *Nucleic Acids Research*, 43(12):e79, 2015.

[38] Joseph O. Ogutu, Torben Schulz-Streeck, and Hans-Peter Piepho. Genomic selection using regularized linear regression models: ridge regression, lasso, elastic net and their extensions. *BMC Proc*, 6(Suppl 2):S10–S10, May 2012. 1753-6561-6-S2-S10[PII].

[39] Jun-Tao LI and Ying-Min JIA. An improved elastic net for cancer classification and gene selection. *Acta Automatica Sinica*, 36(7):976 – 981, 2010.

[40] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, 68:49–67, 2006.

[41] J. Li, W. Dong, D. Meng, and H. Xiao. Gene selection for cancer classification using improved group lasso. In *2016 Chinese Control and Decision Conference (CCDC)*, pages 4221–4225, May 2016.

[42] Shuangge Ma, Xiao Song, and Jian Huang. Supervised group lasso with applications to microarray data analysis. *BMC Bioinformatics*, 8:60–60, Feb 2007. 1471-2105-8-60[PII].

[43] Li-Zhi Liu, Fang-Xiang Wu, and Wen-Jun Zhang. A group lasso-based method for robustly inferring gene regulatory networks from multiple time-course datasets. *BMC Syst Biol*, 8(Suppl 3):S1–S1, Oct 2014. 1752-0509-8-S3-S1[PII].

[44] Deguang Kong, Ryohei Fujimaki, Ji Liu, Feiping Nie, and Chris Ding. Exclusive feature learning on arbitrary structures via $l_1, 2$-norm. In Z. Ghahramani, M. Welling, C. Cortes, N.d. Lawrence, and K.q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1655–1663. Curran Associates, Inc., 2014.

[45] Scott L. Pomeroy, Pablo Tamayo, Michelle Gaasenbeek, Lisa M. Sturla, Michael Angelo, Margaret E. McLaughlin, John Y. H. Kim, Liliana C. Goumnerova, Peter M. Black, Ching Lau, Jeffrey C. Allen, David Zagzag, James M. Olson, Tom Curran, Cynthia Wetmore, Jaclyn A. Biegel, Tomaso Poggio, Shayan Mukherjee, Ryan Rifkin, Andrea Califano, Gustavo Stolovitzky, David N. Louis, Jill P. Mesirov, Eric S. Lander, and Todd R. Golub. Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, 415(6870):436–442, Jan 2002.

[46] Hiro Takahashi and Hiroyuki Honda. Modified signal-to-noise: a new simple and practical gene filtering approach based on the concept of projective adaptive resonance theory (part) filtering method. *Bioinformatics*, 22(13):1662–1664, 2006.

[47] Wikipedia. Funciones sigmoides, 2017.

[48] Roger A. Horn. 1985.

[49] Daniel Gimenez. Implementacin y analisis de algoritmos de alineacion para datos de next generation sequencing, 2016.

[50] Ben Langmead, Cole Trapnell, Mihai Pop, and Steven L Salzberg. Ultrafast and memory-efficient alignment of short dna sequences to the human genome. *Genome biology*, 10(3):R25, 2009.

[51] EurocanPlatform. Hts mappers, 2017.

[52] Susana Hernandez. Desarrollo de una gui para el analisis de datos de secuenciacion genomica, 2016.

[53] kaggle. Digit recognizer, 2017.

[54] The Human Protein Atlas. Egln3.

A

Genes específicos (TCGA)

En las siguientes figuras se representa el ratio de expresión diferencial calculado con la fórmula del S2N con respecto a las muestras de cáncer y control de los 30 mejores genes por cada condición experimental.

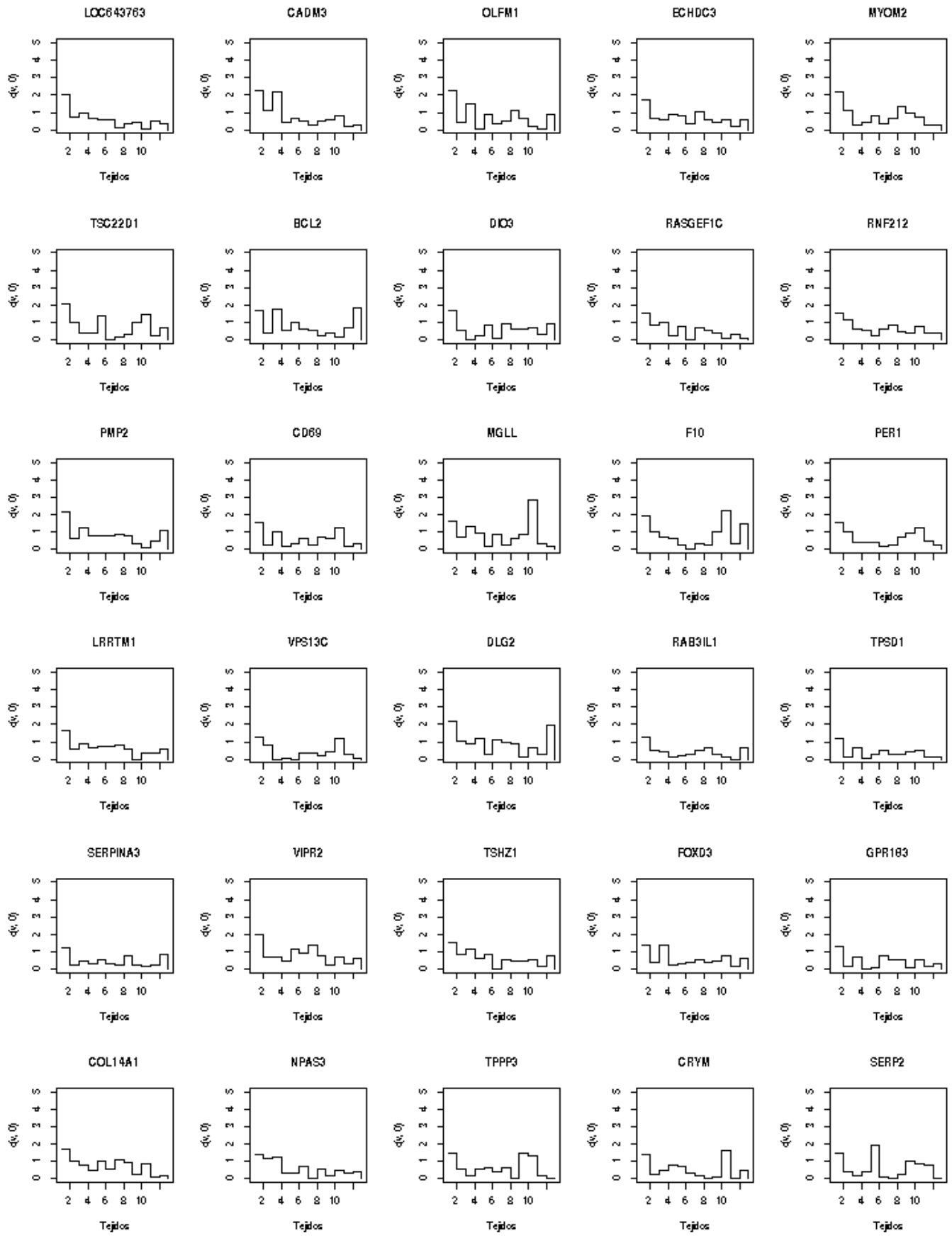

Figura A.1: Ratio de los mejores 30 genes en el tejido **BLCA**

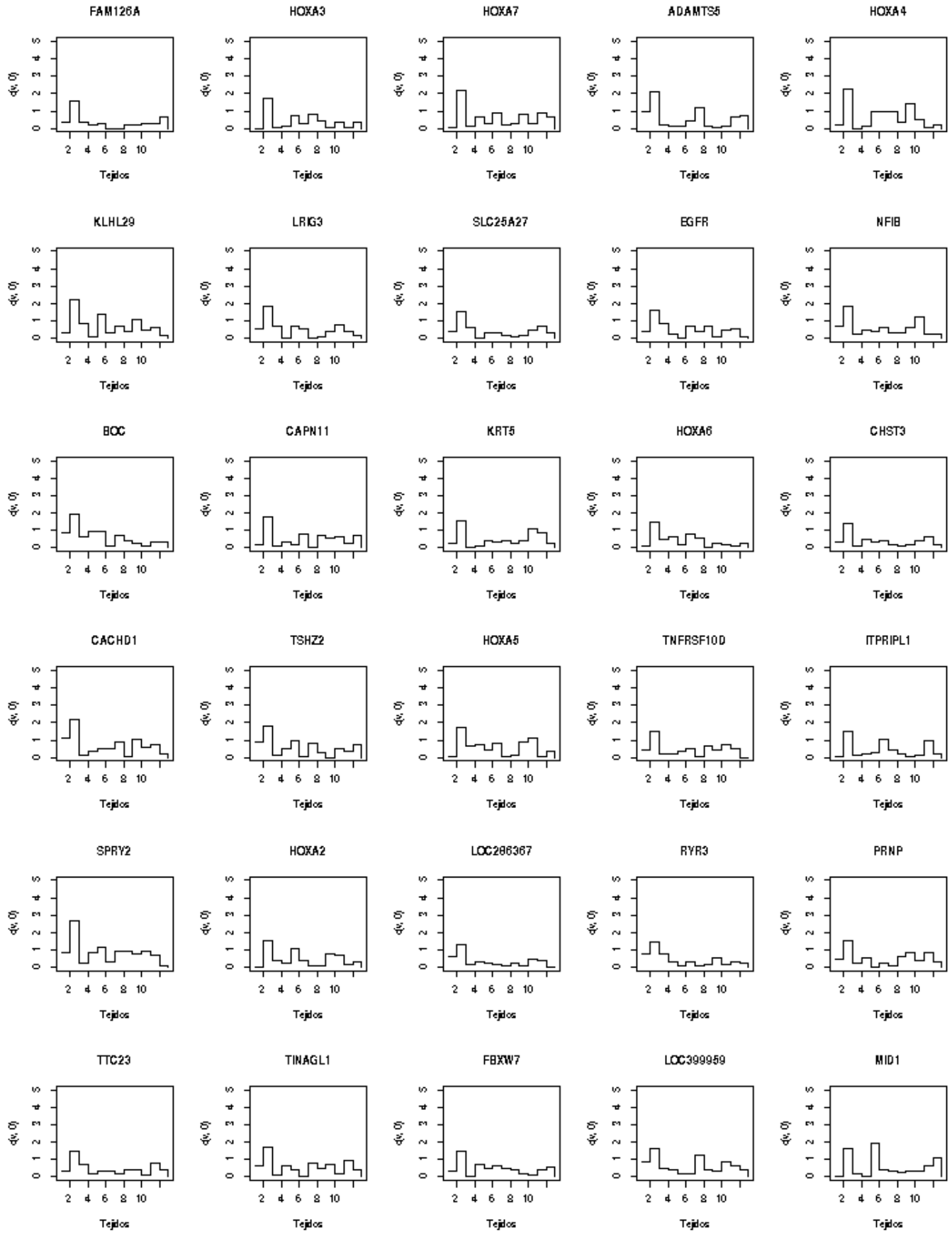

Figura A.2: Ratio de los mejores 30 genes en el tejido **BRCA**

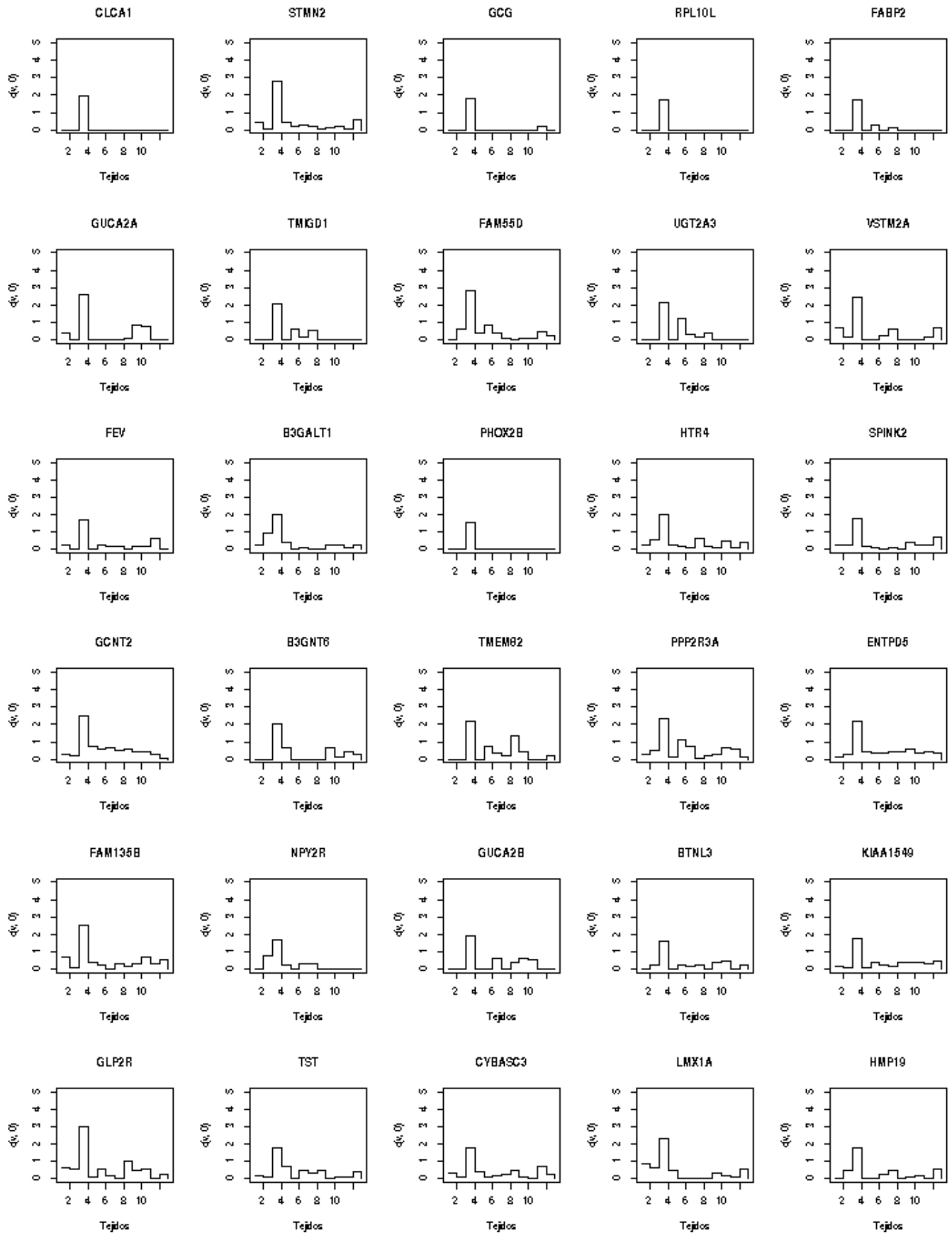

Figura A.3: Ratio de los mejores 30 genes en el tejido **COAD**

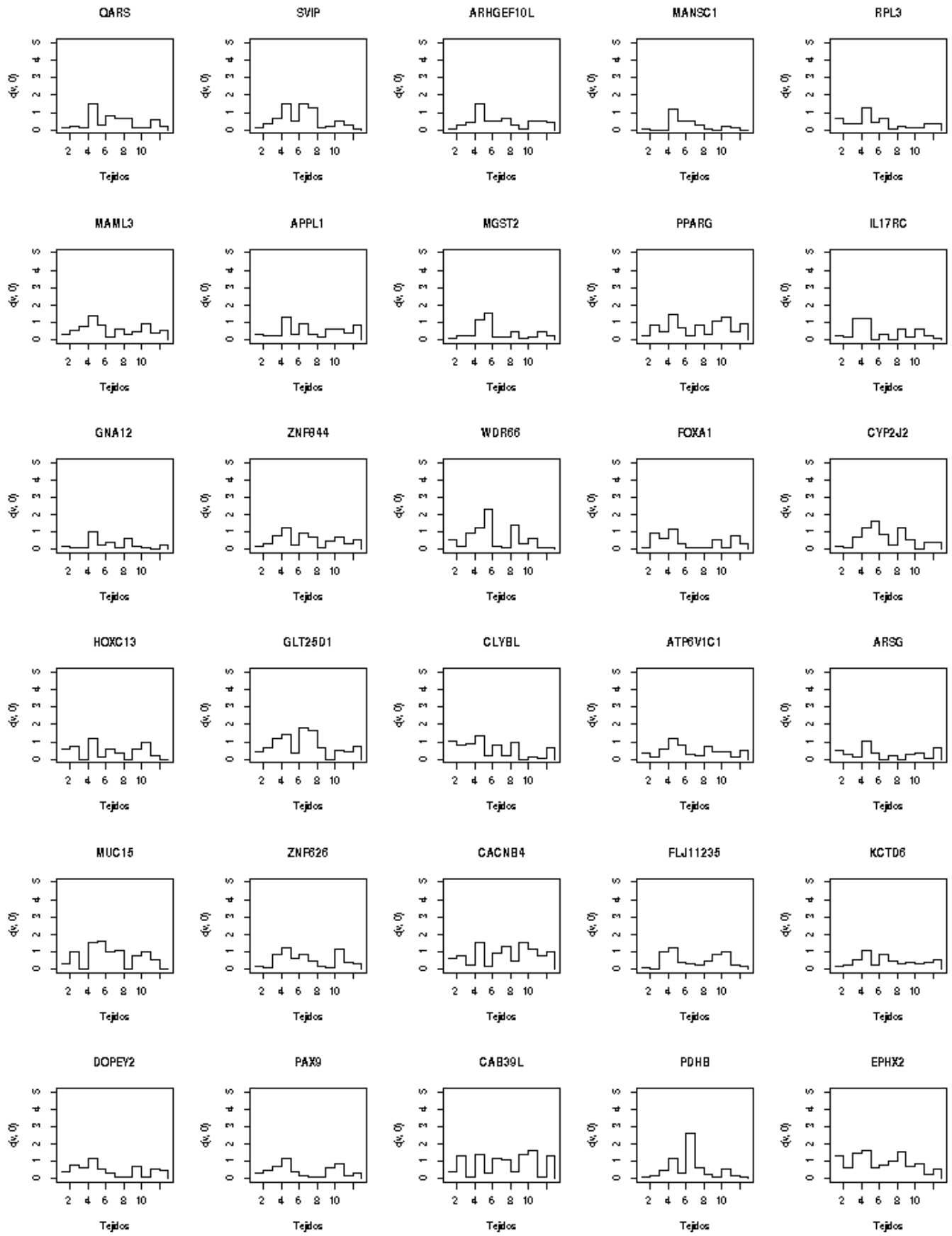

Figura A.4: Ratio de los mejores 30 genes en el tejido **HNSC**

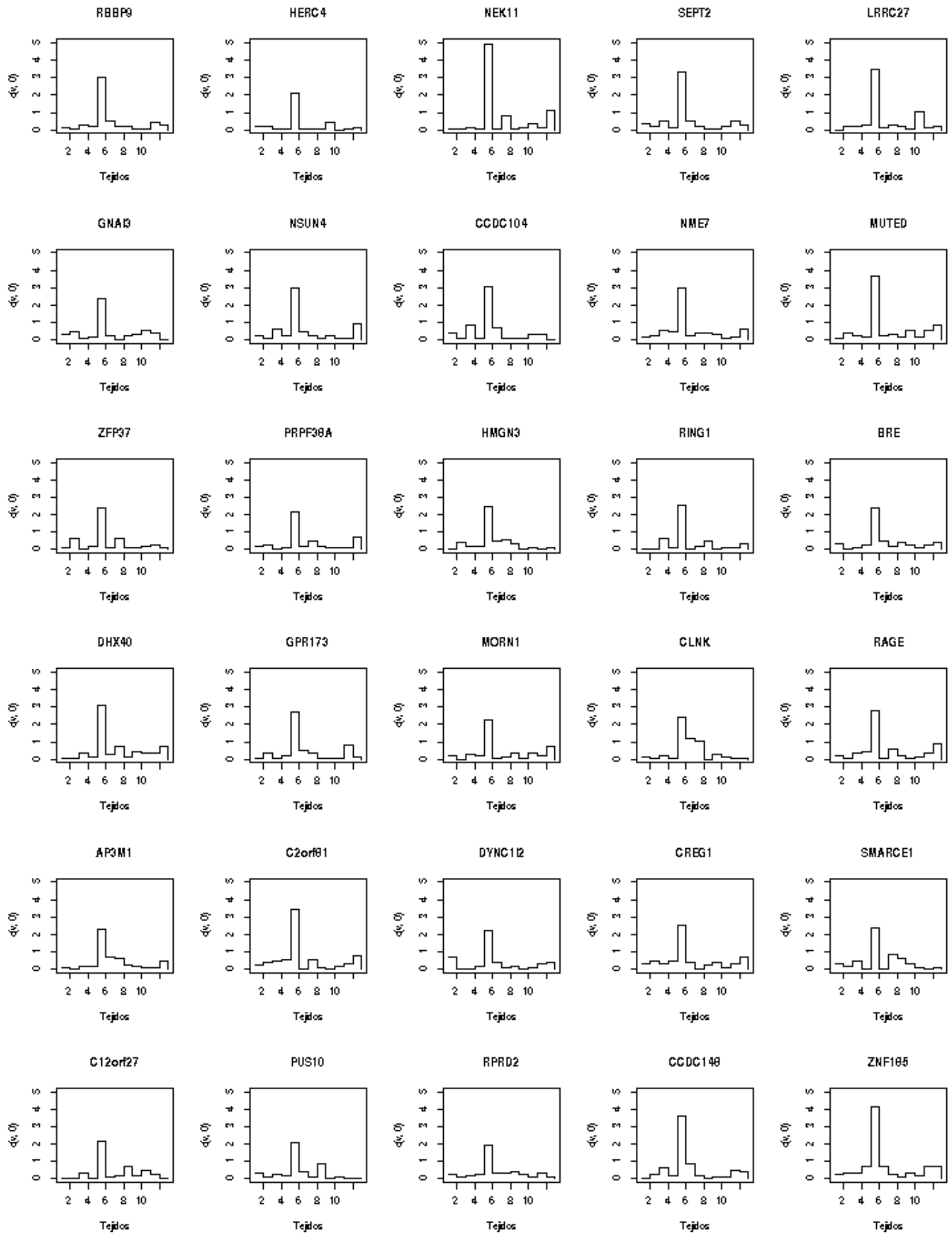

Figura A.5: Ratio de los mejores 30 genes en el tejido **KICH**

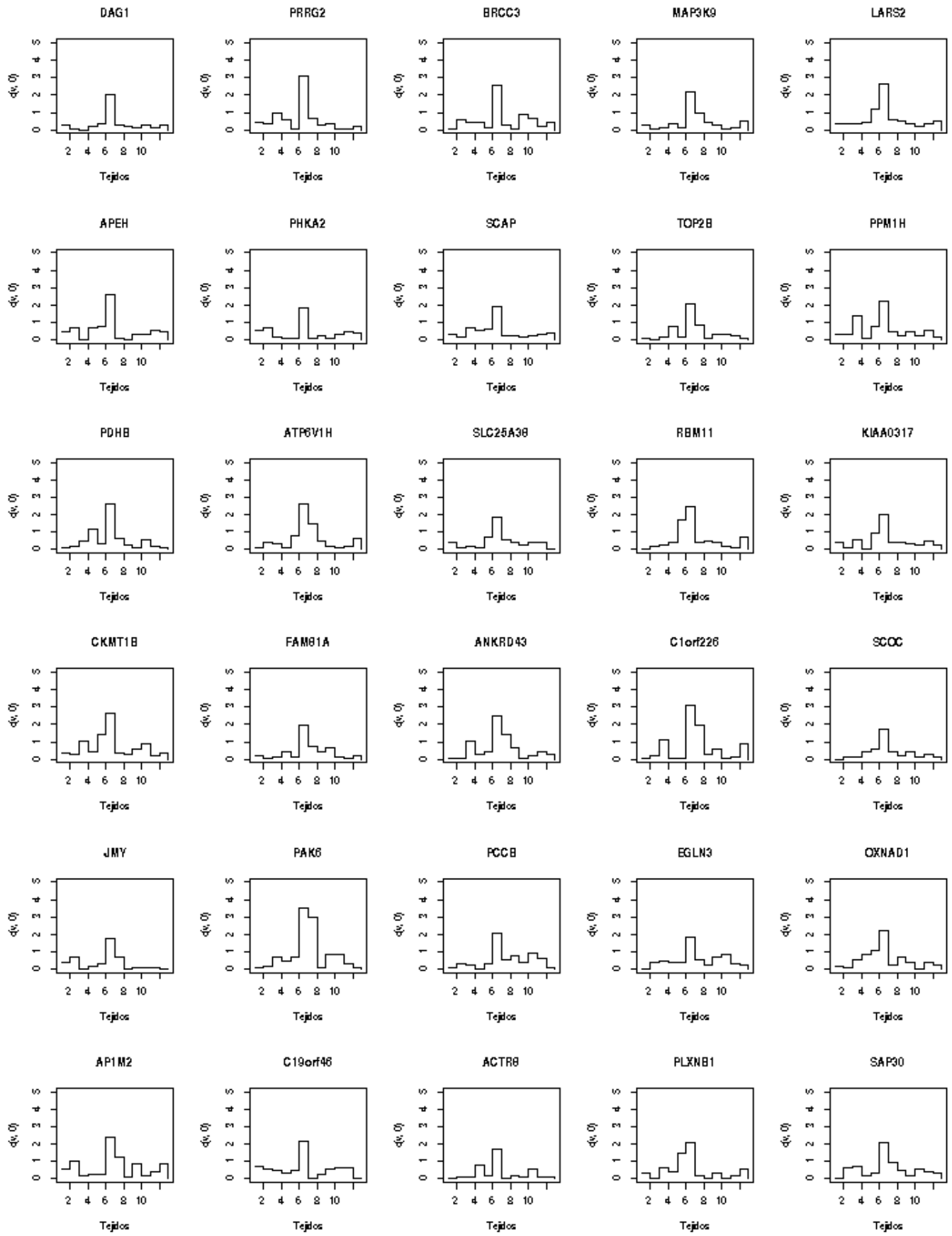

Figura A.6: Ratio de los mejores 30 genes en el tejido **KIRC**

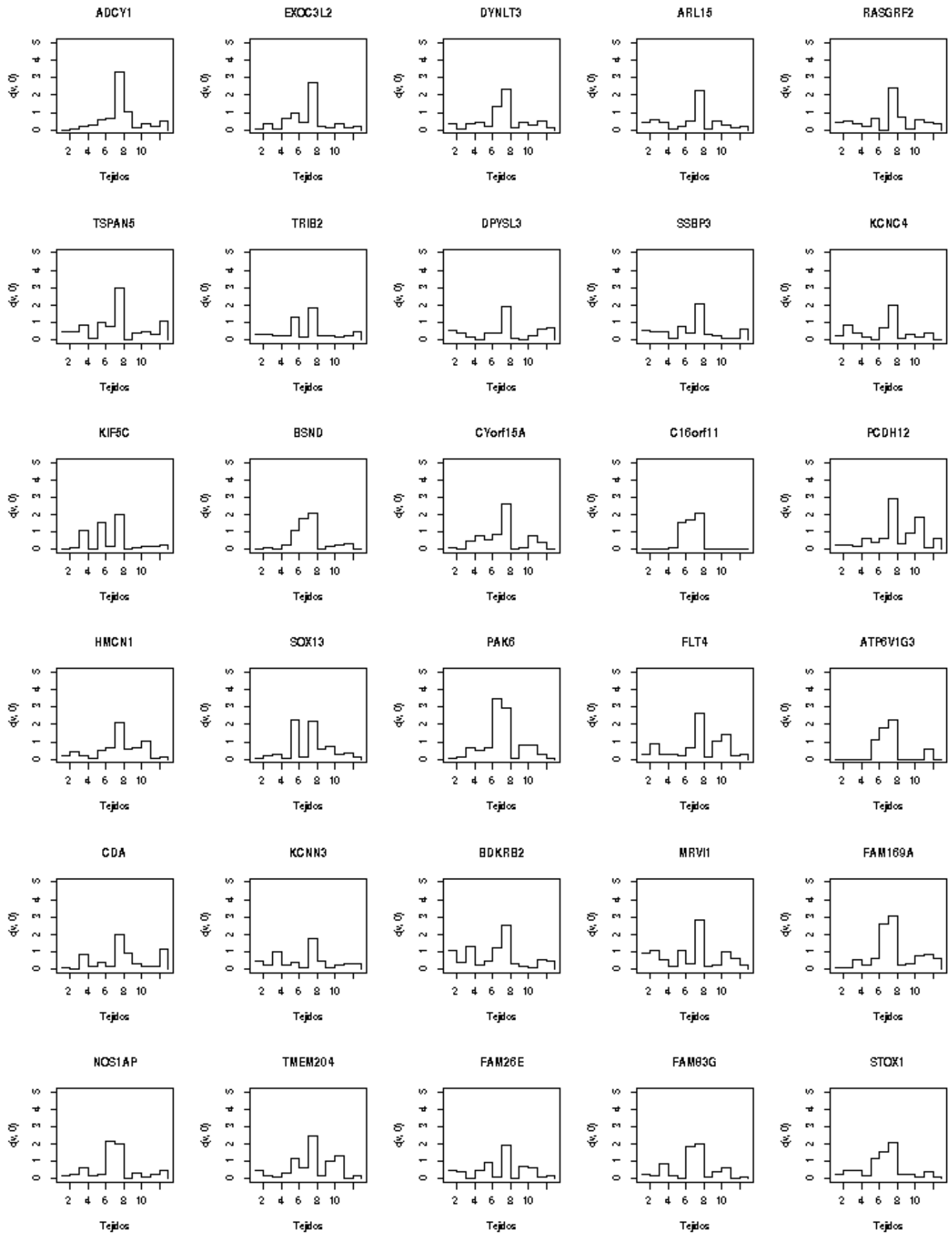

Figura A.7: Ratio de los mejores 30 genes en el tejido **KIRP**

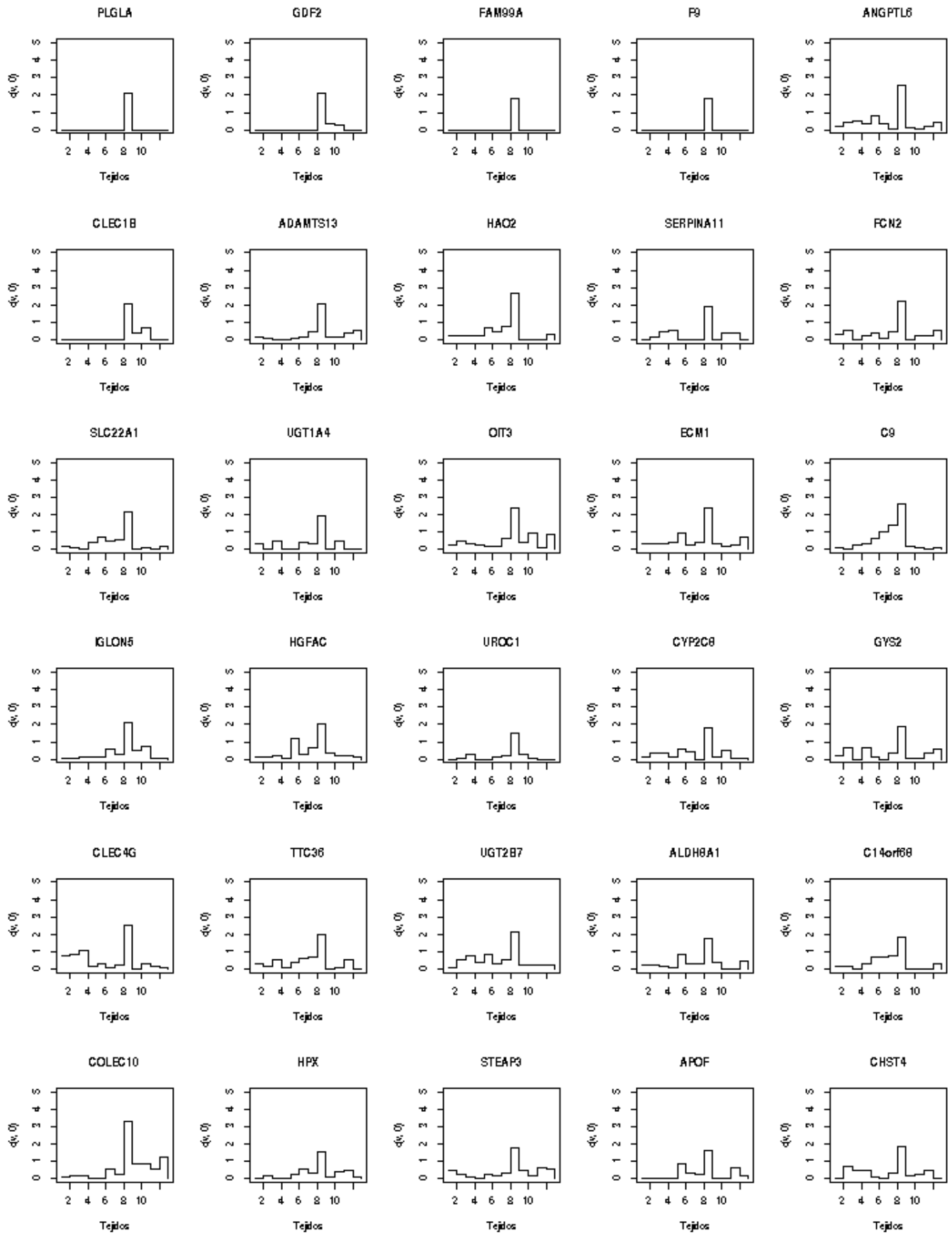

Figura A.8: Ratio de los mejores 30 genes en el tejido **LIHC**

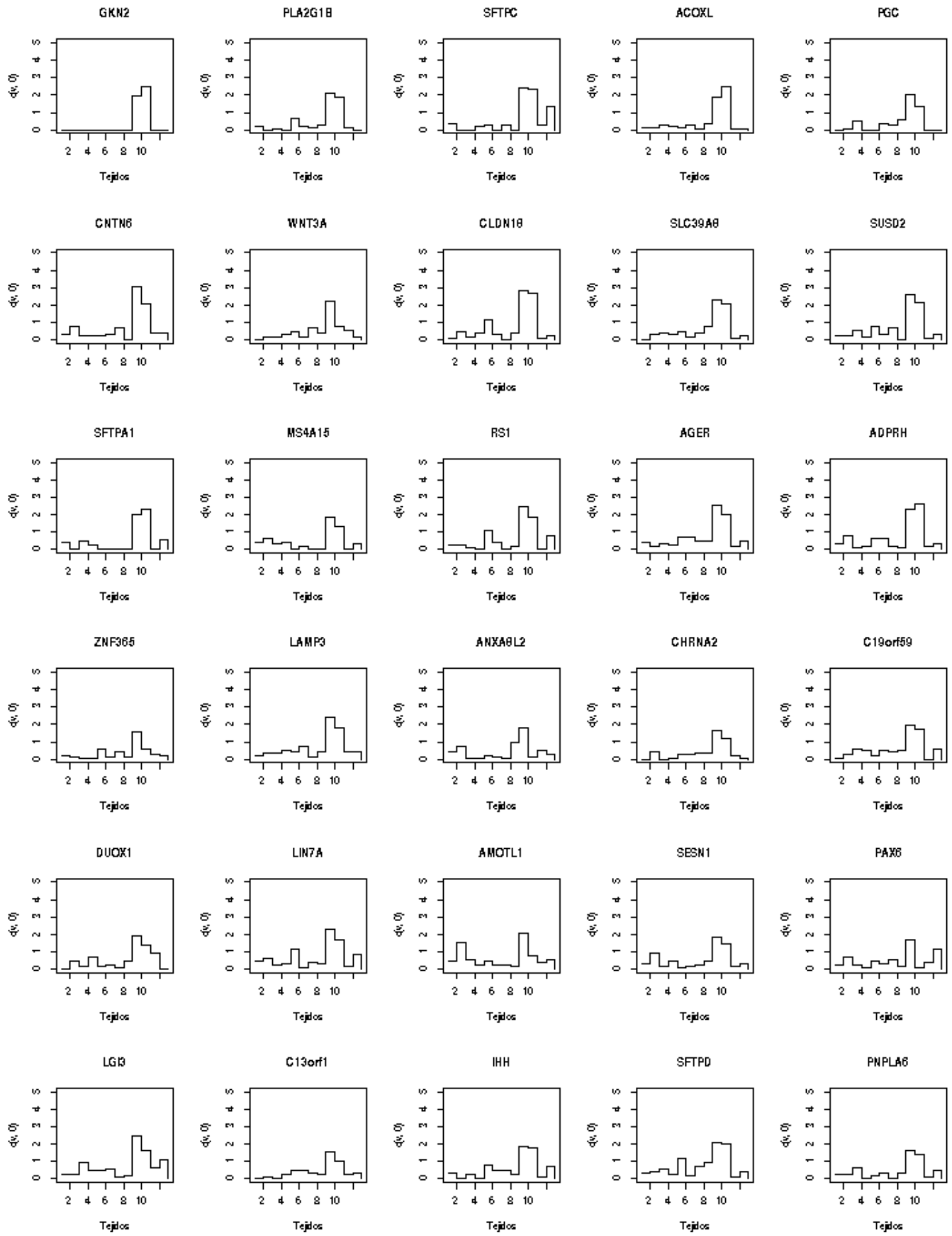

Figura A.9: Ratio de los mejores 30 genes en el tejido **LUAD**

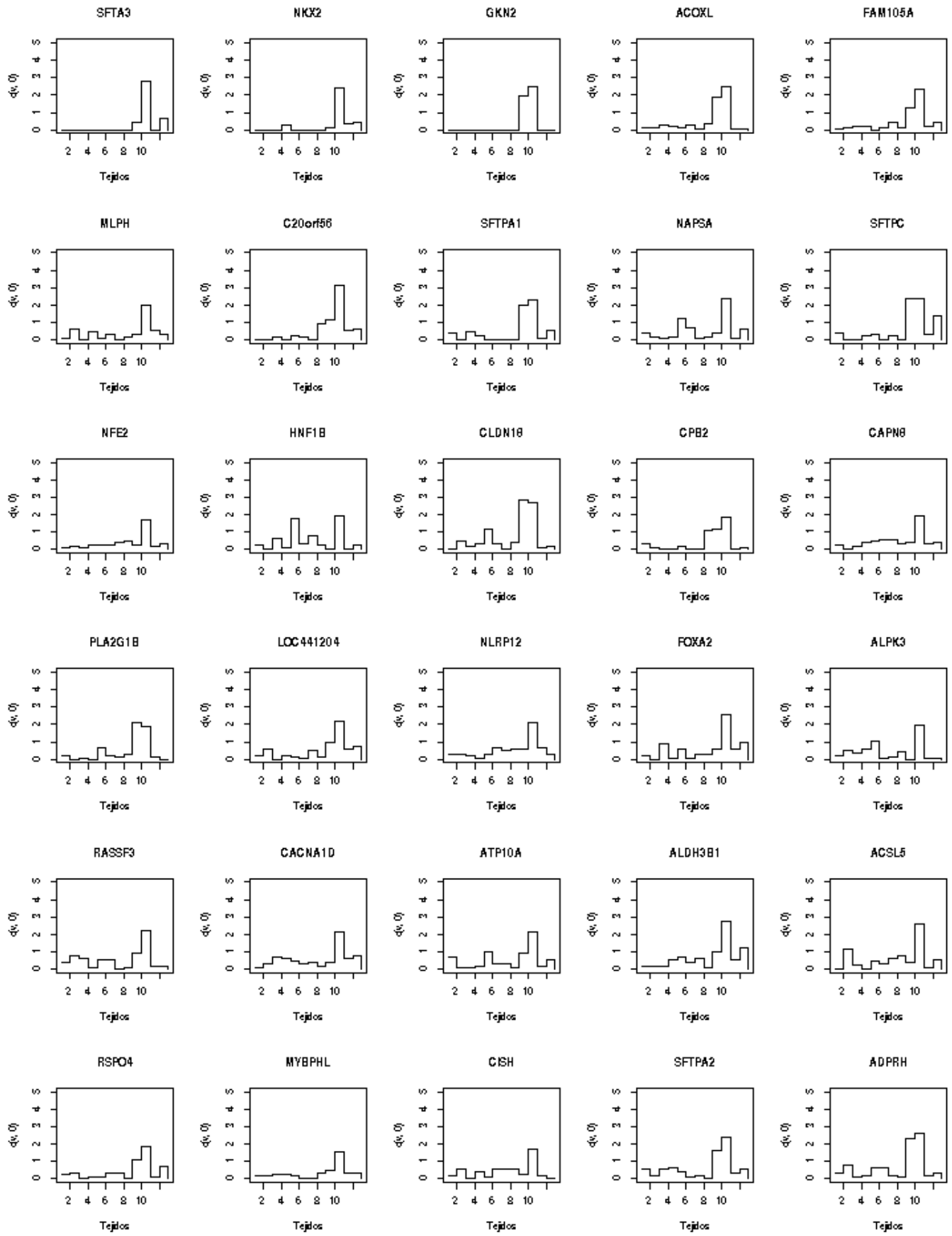

Figura A.10: Ratio de los mejores 30 genes en el tejido **LUSC**

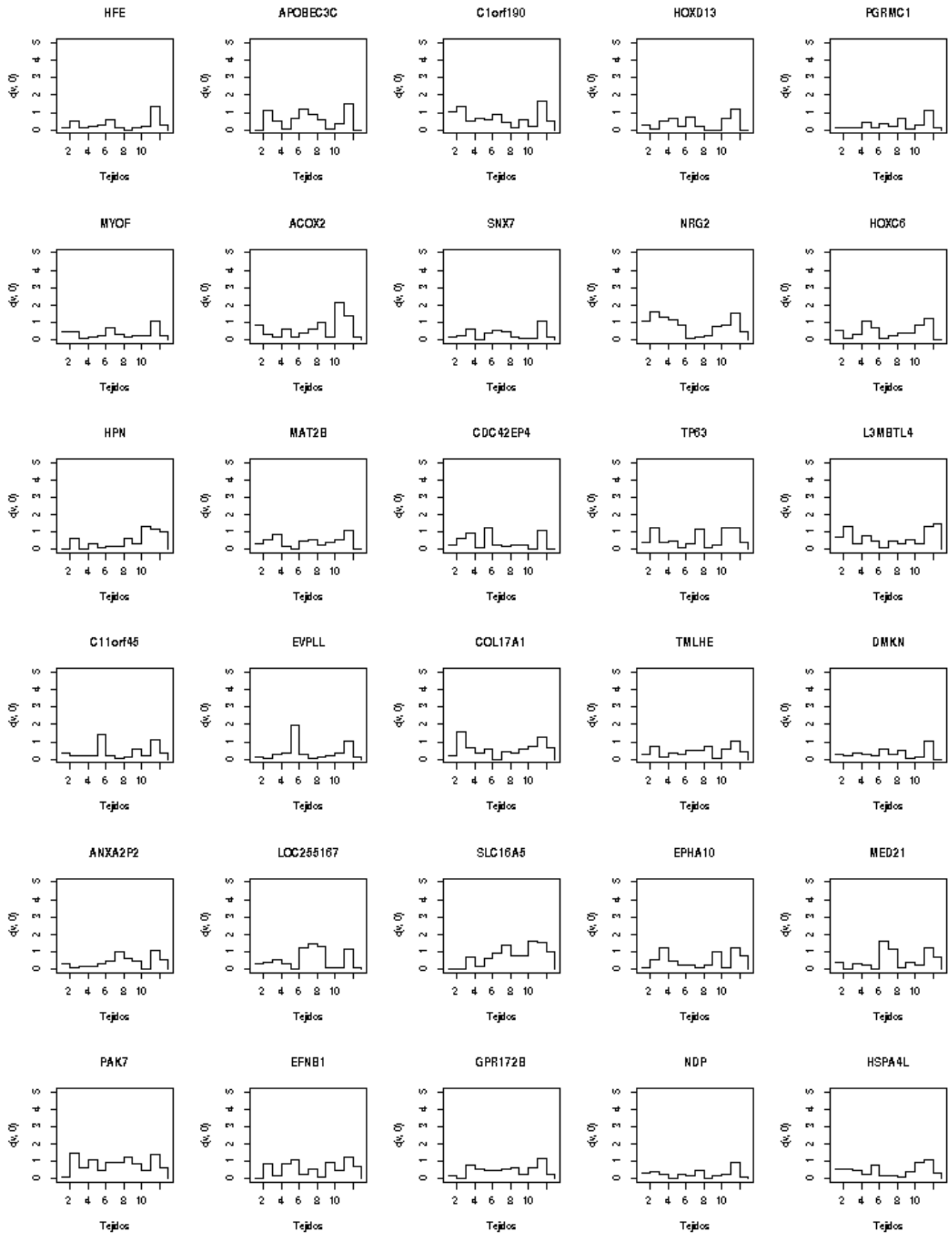

Figura A.11: Ratio de los mejores 30 genes en el tejido **PRAD**

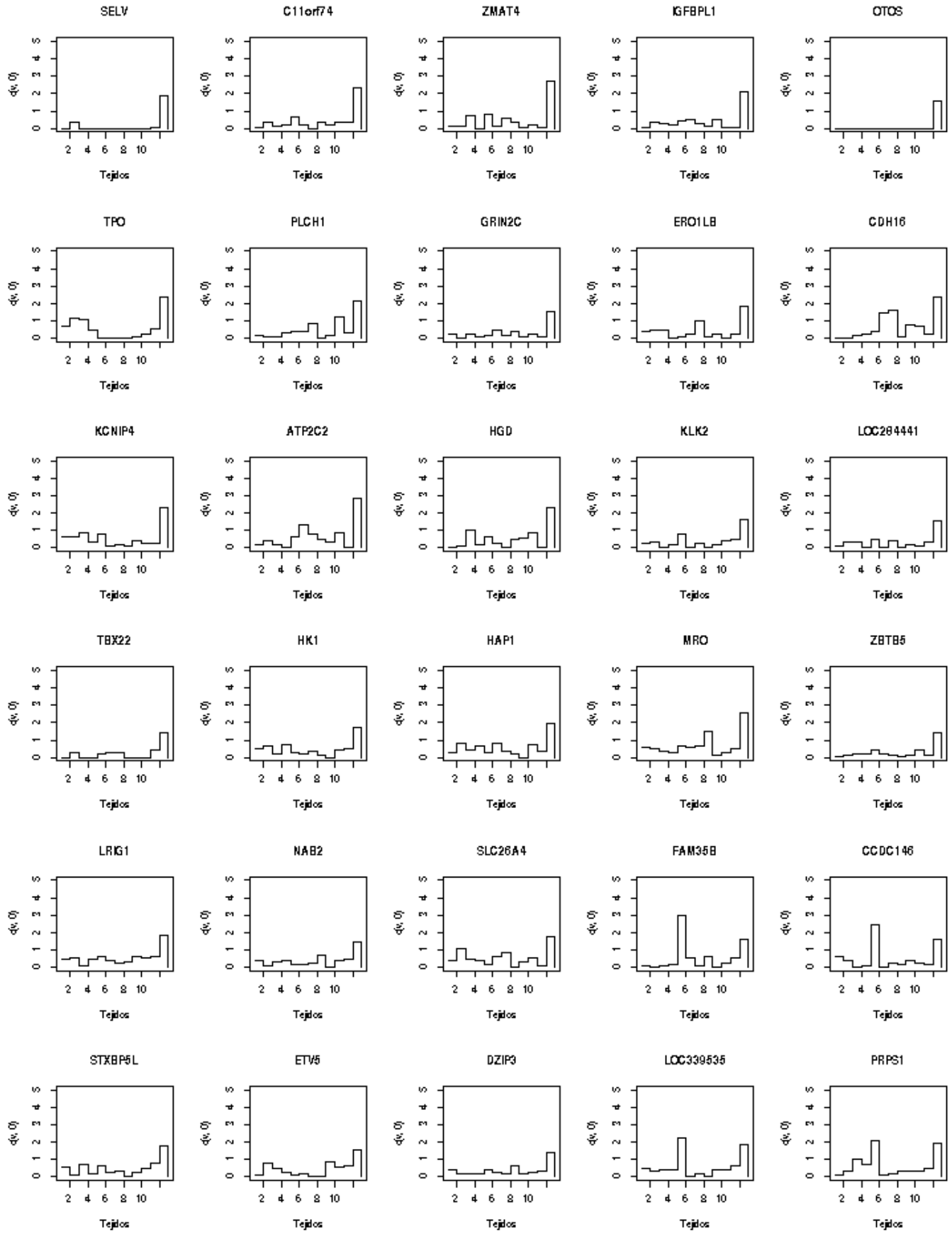

Figura A.12: Ratio de los mejores 30 genes en el tejido **THCA**