Predicción de interacción entre péptido y el complejo mayor de histocompatibilidad tipo I

Introducción

En esta PEC vamos a resolver un análisis relacionado con la modelización de la interacción entre el complejo mayor de histocompatibilidad tipo I (MHCI, en inglés) y péptidos.

En la inmunoterapia contra el cancer las células T deben activarse al exponerse a péptidos tumorales unidos a MHCI (pMHCI). Al analizar la genética del tumor, se pueden identificar péptidos relevantes y, dependiendo del tipo particular de MHCI que tiene el paciente, podemos predecir qué interacción péptido MHCI (pMHCI) es probable que esté presente en el tumor del paciente y, por lo tanto, qué pMHCI se deben usar para activar las células T.

Los ficheros necesarios para realizar la PEC estan en formato csv con separador punto y coma.

En cada registro del fichero *peptidos.csv* se tienen dos variables: 1) el péptido, 2) la clase de interacción donde *NB* significa no interacción y *SB* significa si interacción.

La manera elegida para representar los datos es un paso crucial en los algoritmos de clasificación. En el caso que nos ocupa, análisis basados en secuencias, se usarán dos técnicas por separado:

- One-hot encoding
- Transformación basada en una matriz de substitución

El *one-hot encoding* representa cada aminoácido por un vector de 20 componentes, con 19 de ellas a 0 y una a 1 indicando el aminoacido. Pongamos por ejemplo, el aminoácido A se representa por (1,0,. . . ,0) y el aminoácido R por (0,1,0, . . . , 0). Por tanto, para una secuencia de 9 aminoácidos, como en nuestro caso, se obtendrá un vector de 20*9 componentes, resultado de concatenar los vectores para cada uno de los 9 aminoácidos.

En cambio, la *Transformación basada en una matriz de substitución* asigna a cada aminoácido de la secuencia la probabilidad que mute en cualquiera de los 20 aminoácidos posibles según lo define la matriz de substitución escogida. Como matriz de substitución se ha elegido la matriz BLOSUM62 y se ha preparado para que sus valores sean probabilidades y por tanto, si se suma los valores de la matriz BLOSUM por fila se obtendra el valor 1 en cada fila. Esta matriz esta contenida en el archivo *BLOSUM62_probabilities.csv*

Como ejemplo de transformación, el aminoácido A se representa como:

```
##       A       R       N       D       C       Q       E       G       H
## 0.29015 0.03104 0.02564 0.02969 0.02159 0.02564 0.04049 0.07827 0.01484
##       I       L       K       M       F       P       S       T       W
## 0.04318 0.05938 0.04453 0.01754 0.02159 0.02969 0.08502 0.04993 0.00540
##       Y       V
## 0.01754 0.06883
```

que corresponde a la fila del aminoacido A de la matriz BLOSUM62. Observar que su suma es uno pues la transformación se basa en probabilidades.

Así que cada péptido de tamaño 9 se transformará en un vector de 20*9 valores, tal como pasa con la transformación one-hot encoding.

Para cada forma de representar la secuencia, *one-hot encoding* o *transformación basada en una matriz de substitución*, el objetivo es implementar un algoritmo `knn` para predecir si la secuencia peptídica interacciona o no con MHCI.

Nota 1: Recordar que desde Bioconductor podeis descargar el paquete `Biostrings`

https://bioconductor.org/packages/release/bioc/html/Biostrings.html

que dispone de funciones propias para la computación en strings que pueden ser de utilidad en esta PEC.

Nota 2: A modo de ejemplo se muestra el resultado de codificar el primer registro de la base de datos con *one-hot encoding* y con *transformación basada en una matriz de substitución*

- **one-hot encoding**: vector de 0s y 1s del primer registro.

```
##   [1] 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
##  [36] 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [71] 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
## [106] 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
## [141] 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [176] 0 0 0 0 1
```

- **transformación basada en una matriz de substitución**: vector de probabilidades de mutación del primer registro.

```
##   [1] 0.04453 0.02429 0.01417 0.01518 0.01619 0.01619 0.02024 0.02126
##   [9] 0.01012 0.11538 0.37551 0.02530 0.04960 0.05466 0.01417 0.02429
##  [17] 0.03340 0.00709 0.02227 0.09615 0.05221 0.03213 0.02008 0.02008
##  [25] 0.01606 0.02811 0.02811 0.02811 0.01606 0.10040 0.19679 0.03614
##  [33] 0.16064 0.04819 0.01606 0.03614 0.04016 0.00803 0.02410 0.09237
##  [41] 0.29015 0.03104 0.02564 0.02969 0.02159 0.02564 0.04049 0.07827
##  [49] 0.01484 0.04318 0.05938 0.04453 0.01754 0.02159 0.02969 0.08502
##  [57] 0.04993 0.00540 0.01754 0.06883 0.03383 0.01903 0.01691 0.01691
##  [65] 0.01057 0.01057 0.01903 0.02537 0.01691 0.06342 0.11416 0.01903
##  [73] 0.02537 0.38689 0.01057 0.02537 0.02537 0.01691 0.08879 0.05497
##  [81] 0.04050 0.02804 0.02181 0.01869 0.00935 0.02181 0.02804 0.02492
##  [89] 0.04673 0.04361 0.06854 0.03115 0.01869 0.13084 0.01558 0.03115
##  [97] 0.02804 0.02804 0.31776 0.04673 0.04453 0.02429 0.01417 0.01518
## [105] 0.01619 0.01619 0.02024 0.02126 0.01012 0.11538 0.37551 0.02530
## [113] 0.04960 0.05466 0.01417 0.02429 0.03340 0.00709 0.02227 0.09615
## [121] 0.04050 0.02804 0.02181 0.01869 0.00935 0.02181 0.02804 0.02492
## [129] 0.04673 0.04361 0.06854 0.03115 0.01869 0.13084 0.01558 0.03115
## [137] 0.02804 0.02804 0.31776 0.04673 0.05525 0.04972 0.04052 0.09024
## [145] 0.00737 0.06446 0.29650 0.03499 0.02578 0.02210 0.03683 0.07551
## [153] 0.01289 0.01657 0.02578 0.05525 0.03683 0.00552 0.01657 0.03131
## [161] 0.06996 0.02195 0.01646 0.01783 0.01920 0.01646 0.02332 0.02469
## [169] 0.00823 0.16461 0.13032 0.02606 0.03155 0.03567 0.01646 0.03292
## [177] 0.04938 0.00549 0.02058 0.26886
```

Enunciado

1. Escribir en el informe una sección con el título "Algoritmo k-NN" en el que se haga una breve explicación de su funcionamiento y sus características. Además, se presente una tabla de sus fortaleza y debilidades.
2. Desarrollar dos funciones en R:
   - Una que implemente la codificación "one-hot" (*one-hot encoding*) de las secuencias.
   - Y otra que implemente la *transformación basada en una matriz de substitución BLOSUM62* de las secuencias.
3. Desarrollar un script en R que implemente un clasificador `knn`. El script debe realizar los siguientes apartados:
   (a) Leer los datos `peptidos.csv` y hacer una breve descripción de ellos. Incluir en esta descripción el patrón de cada clase de péptido mediante la representación de su secuencia logo (https://en.

wikipedia.org/wiki/Sequence_logo). Para realizar esta representación se puede usar el paquete `ggseqlogo` descargable desde github.

(b) Para cada forma de representar los datos: *one-hot encoding* o *transformación basada en una matriz de substitución BLOSUM62*, realizar la implementación del algoritmo `knn`, con los siguientes pasos:

i. Transformar las secuencias de aminoácidos en vectores numéricos usando la función de transformación desarrollada anteriormente.

ii. Utilizando la semilla aleatoria `123`, separar los datos en dos partes, una parte para training (67%) y una parte para test (33%).

iii. Utilizar un `knn` (k =3, 5, 7, 11) basado en el training para predecir que péptidos del test interaccionan o no con el MHCI. Además, realizar una curva ROC para cada k.

iv. Comentar los resultados de la clasificación en función de la curva ROC y del número de falsos positivos, falsos negativos y error de clasificación obtenidos para los diferentes valores de k. La clase que será asignada como positiva es la `SB`.

(c) Comparar los resultados de clasificación obtenidos con las dos técnicas de representación de péptidos, *one-hot encoding* y *transformación basada en una matriz de substitución BLOSUM62*.

Informe de la PEC

El informe se presentará mediante un informe dinámico R markdown con la estructura habitual de los ejercicios no evaluables realizados hasta ahora. En primer lugar, el informe tendra un título (igual que el de la PEC), el autor, la fecha de creación y el índice de apartados de la PEC. En segundo lugar, se crea una sección con el título "Algoritmo k-NN" donde se haga una breve explicación de su funcionamiento y sus características. Además, se presenta la tabla de sus fortaleza y debilidades. En tercer lugar se realizan los diferentes apartados de la PEC.

Una característica que se valorará es hasta que punto es el informe "dinámico". En el sentido de adaptarse el informe a cambios en los datos.

Se entregaran los ficheros:

1. Fichero ejecutable (.Rmd) que incluya un texto explicativo que detalle los pasos implementados en el script y el código de los análisis. Si este fichero llama a otros ficheros como *.bib*, *imagenes*, ... se deben de incluir también para poder reproducir el informe.
2. Informe (pdf o/y html) resultado de la ejecución del fichero Rmd anterior.
3. Fichero/s de datos.

Puntuacions de los apartados

Apartado 1 (5%), Apartado 2 (25%), Apartado 3 (60%), Calidad del informe dinámico (10%).