

**APLICACIONES EN INTERNET**  
**EXAMEN DE PROBLEMAS- FEBRERO 2017**

1. Una aplicación web de “economía colaborativa” permite que usuarios que quieren realizar un viaje se pongan en contacto con otros que van a realizar un trayecto similar para compartir el vehículo. La aplicación se ha implementado mediante PHP y una BBDD MySQL. En la BBDD se dispone de las tablas *conductor* que incluye el *dni*, *nombre* y *apellido* del conductor. Además se dispone de una tabla de  *rutas* que incluye *id*, *origen*, *destino* y *hora\_salida*. Todos los campos de esta tabla son enteros (por ejemplo, el origen y destino se identifica con un código postal). Finalmente, se dispone de una tabla *conductor\_ruta\_precio* en el que se relacionan las rutas que ofrecen los conductores a un precio determinado. Se le pide que:
  - a) Implemente un script PHP que genere un formulario para que un conductor pueda generar una nueva ruta a un precio determinado. Es decir, el formulario permite rellenar los datos de la ruta y el precio. Además el *dni* del conductor está disponible mediante el método GET en la variable *dni*. **(0.5 puntos)**
  - b) Implemente un script PHP que procese el formulario anterior. Es decir, inserte una nueva ruta y la relacione con conductor y precio. El ID de un nuevo registro insertado está disponible en la función del PDO *lastInsertId()* después de una consulta de inserción o se puede consultar con *SELECT LAST\_INSERT\_ID()*. **(0.5 puntos)**
  - c) Hay usuarios que realizan un trayecto habitualmente. Se desea que se les muestre automáticamente los conductores y precios para la ruta habitual, nada más conectarse al servidor. El usuario marcará su ruta habitual y la ruta se conservará durante un año. Para ello se utilizarán cookies: cada vez que el usuario marque una ruta como habitual se invocará una función *habitual(\$idr)* a la que se le pasa como parámetro el *id* de la ruta. Cuando el usuario accede al servidor (*index.php*) se le mostrará la ruta habitual (todos los campos) y una tabla HTML con el nombre de los conductores que las ofertan y el precio al que se oferta. Implemente la función *habitual* y el script *index.php* **(1.5 puntos)**
2. Se pretende mejorar la forma en la que se seleccionan las rutas habituales, de manera que se puedan seleccionar varias rutas, utilizando Javascript. Para ello se mostrará una tabla HTML con todas las rutas, como en el siguiente fragmento de código HTML (numeración incluida para facilitar la respuesta si es necesario):
  1. <p> Mostrar rutas seleccionadas</p>
  2. <table>
  3. <tr><td>1</td><td>Murcia</td><td>Cartagena</td><td>10:00</td></tr>
  4. <tr><td>1</td><td>Cartagen</td><td>Almeria</td><td>09:00</td></tr>
  5. ... NO SE MUESTRA EL RESTO DE ENTRADAS
  6. </table>

Para seleccionar las rutas habituales el usuario pinchará en las filas de la tabla de las rutas que marcará como habituales. Es decir, cada vez que el usuario pinche en una fila se añadirá el id de la ruta a una lista. Para ello

  - a) Mediante Javascript **exclusivamente** haga que cada vez que un usuario pincha en cualquiera de las filas se añada el id a la lista de rutas. La lista se mostrará insertando el id de la ruta en un nuevo elemento P dentro del DIV que ya existe. Proporcione el código Javascript y asócielo adecuadamente al fragmento HTML **(1.75 puntos)**.
  - b) Proporcione una hoja de estilo o Javascript junto con las modificaciones/añadidos necesarias al fragmento para que cuando se seleccione una ruta se muestre la fila seleccionada con fondo verde **(0.75 puntos)**.

3. Consideremos una aplicación de descarga de películas que incorpora un sistema de recomendación por filtrado colaborativo para el que se definen  $n = 3$  características. Contamos con la siguiente base de datos de puntuaciones (definidas entre 0 y 5):

	Andrés	María	Carlos	Eusebio	Laura
Película 1	4	1	0	0	?
Película 2	3	?	2	?	4
Película 3	?	5	?	4	?
Película 4	2	5	4	?	1

Para optimizar la función de coste se ha de implementar en MATLAB la función

`function [J, X_grad, Theta_grad] = calcularGradiente(X, Theta, Y)`

donde  $Y$  es la matriz de puntuaciones, **en la que cada entrada no definida contiene “-1”**, y la matriz  $X$  contiene las características de las películas, y  $\Theta$  los parámetros de los usuarios:

$$X = \begin{bmatrix} \text{---} (x^{(1)})^T \text{---} \\ \text{---} (x^{(2)})^T \text{---} \\ \vdots \\ \text{---} (x^{(n_m)})^T \text{---} \end{bmatrix}, \quad \Theta = \begin{bmatrix} \text{---} (\theta^{(1)})^T \text{---} \\ \text{---} (\theta^{(2)})^T \text{---} \\ \vdots \\ \text{---} (\theta^{(n_u)})^T \text{---} \end{bmatrix}$$

siendo  $n_m$  es el número de películas y  $n_u$  el número de usuarios. La función devuelve:

- $J$ : un escalar con el valor de la función de coste para las matrices  $X$  y  $\Theta$  proporcionadas.
  - $X\_grad$ : una matriz  $n_m \times n$  con las derivadas parciales de la función de coste  $J$  respecto a cada elemento de  $X$
  - $\Theta\_grad$ : una matriz  $n_u \times n$  con las derivadas parciales de la función de coste  $J$  respecto a cada elemento de  $\Theta$
- a) Escriba el código MATLAB necesario para obtener:  $n_m$  (num\_peliculas),  $n_u$  (num\_usuarios),  $n$  (num\_caracteristicas) y la matriz  $R$  tal que  $R(i,j)=1$  si la película  $i$  ha sido puntuada por el usuario  $j$  y  $R(i,j) = 0$  en caso contrario. **(0.5 puntos)**.
- b) Escriba el código MATLAB necesario para calcular la función de coste  $J$  sin regularización. **(0.5 puntos)**.
- c) Considere que  $X$  y  $\Theta$  se inicializan a los siguientes valores:  
 $X = [0.2 \ 0.5 \ 1; \ 0 \ 1 \ 1; \ 1 \ 0 \ 0.5; \ 0 \ 0.5 \ 1];$   
 $\Theta = [1 \ 0 \ 3; \ 2 \ 1 \ 0; \ 3 \ 2 \ 0; \ 4 \ 1 \ 1; \ 0 \ 2 \ 1];$   
 Indique qué error cuadrático se estaría cometiendo en la predicción de la puntuación de Andrés para la película 4 y en la predicción de la puntuación de Carlos para la película 2. **(0.5 puntos)**.
- d) Escriba un código en MATLAB que, dado un vector de características  $x$  de una película, le permita encontrar la película más parecida de entre todas la que hay en la matriz  $X$ . **(0.5 puntos)**.
- e) Suponga que disponemos de un conjunto de datos de entrenamiento con más películas y más usuarios, y disponemos de unas 10000 puntuaciones. Decidimos introducir un factor de regularización  $\lambda$  en el algoritmo. Explique brevemente cómo podríamos ajustar el valor de  $\lambda$  para evitar sobreajuste. **(0.5 puntos)**.

4. Considere un conjunto de datos de entrenamiento  $X, y$ :

$$X = \begin{bmatrix} \text{---} (x^{(1)})^T \text{---} \\ \text{---} (x^{(2)})^T \text{---} \\ \vdots \\ \text{---} (x^{(m)})^T \text{---} \end{bmatrix} \quad \vec{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}$$

tal que  $y^{(i)}$  solo toma dos posibles valores  $\{0,1\}$ . Estos datos se emplearán para entrenar un **clasificador por regresión logística**. El vector de parámetros  $\theta$  del clasificador viene dado por  $\theta$  (vector columna en Matlab).

(a) Escriba el código de una función en Matlab,  $g = \text{sigmoid}(z)$ , para obtener el valor de la función logística  $g(z) = (1 + e^{-z})^{-1}$ . En la función de Matlab, el argumento  $z$  puede ser un escalar, o un vector **(0.5 puntos)**.

(b) Escriba un código Matlab para obtener la función de coste  $J(\theta)$  **(0.5 puntos)**.

(c) Suponga ahora que los datos están etiquetados con 5 posibles valores  $y^{(i)} \in \{1,2,3,4,5\}$ . Disponemos de 1850 ejemplos de entrenamiento,  $x^{(i)} \in \mathbb{R}^{15}$ . Indique **cómo realizar la clasificación multiclase**, cuántos vectores de parámetros ( $\theta^{(j)}$ ) obtendrá y de qué dimensión será cada vector ( $\theta^{(j)}$ ). **(0.5 puntos)**.

(d) Si deseamos añadir los factores cuadráticos  $x_1^2, x_2^2, \dots, x_{15}^2$  y cúbicos  $x_1^3, x_2^3, \dots, x_{15}^3$  a los ejemplos de entrenamiento para realizar la clasificación, indique el tamaño que tendrían los vectores de parámetros ( $\theta^{(j)}$ ) obtenidos **(0.5 puntos)**.

(e) Suponga que ya se ha entrenado el clasificador multiclase. Por tanto, dispone de  $N$  vectores  $\theta^{(j)}$  distribuidos en las filas de la matriz `Theta` (de forma que  $\theta^{(1)}$  viene dado por `Theta(1, :)`,  $\theta^{(2)}$  por `Theta(2, :)`, etc). Escriba el código necesario para determinar a qué clase asignar un nuevo dato de entrada  $x$ . **(0.5 puntos)**.