Práctica

Parte 2

El formato de este ejercicio es más abierto que el resto de las entregas anteriores, ya que consiste en el reto de desarrollar un proyecto propio de interés personal.

La finalidad es que cada estudiante escoja un objetivo de análisis de datos sobre un conjunto abierto y que elabore un pequeño proyecto. La pregunta sobre el porqué, y el qué hacer, se deja a vuestra elección e intereses.

Es imprescindible que en el **documento pdf** se responda de forma ordenada y numerada a las preguntas del ejercicio 2 y aparezcan todas las capturas de pantalla en las que se vea la invocación de cada uno de los scripts, el usuario de la UOC usado en anteriores trabajos, y su resultado.

Se deberá continuar con el dataset elegido en la parte 1 de esta práctica, siendo imprescindible enviarlo nuevamente en la entrega, por lo que no olvidéis **incluirlo en el zip**. Os recordamos que el dataset que hubierais elegido debe tener un **mínimo de 300 registros** y un **peso inferior a 1.5 MB** (sin comprimir).

El dataset original nunca debe ser sobreescrito, ya que sobreescribir el input es una mala praxis que puede ocasionar problemas, debido a que si se genera una excepción, se va la luz, etc, y el script no completa su ejecución, ya no se podría hacer una nueva ejecución a no ser que se vuelvan a recuperar los datos originales, algo que en un entorno profesional/real puede no ser posible.

De acuerdo con los objetivos de la asignatura, en la práctica se evaluarán solamente los aspectos de carácter técnico, no el contenido o temática del dataset, el cual debe ser interesante para vosotros. Debe utilizarse **el usuario creado en la PEC 1** y demostrar la ejecución de los scripts, mediante capturas de pantalla en la memoria.

La entrega debe realizarse con un único fichero **en formato zip** (se adjunta un ejemplo junto al enunciado) **sin carpetas** con todo lo siguiente al mismo nivel:

1. Una **memoria** en formato **PDF**, con una extensión no superior a **15 páginas** (incluyendo portada e índice), que incluya **capturas de pantalla**, donde se aprecie vuestro usuario, y se vea la ejecución de cada script, junto con el

resultado íntegro o bien parte de éste. *Si no hay imágenes en la memoria donde se vea vuestro usuario, se considerará plagio y no se otorgará puntuación.*

2. Cada uno de los scripts solicitados se debe adjuntar en ficheros listos para ser ejecutados, **con el nombre y la extensión indicados en los ejercicios, en los que se incluirá el código fuente**. Cercioraros bien que al ejecutar cada script la salida es la esperada y que funciona correctamente y sin errores, ya que será ejecutado en la corrección.

3. El **dataset** que hayáis elegido en la práctica 1 para el desarrollo de la práctica deberá incluir una columna (al menos) para cada uno de los siguientes tipos:

    - Cadena de caracteres.
    - Enteros.
    - Fechas o números decimales.
    - Se valorará que el dataset incluya la perspectiva de género de alguna manera (p.e. incluya datos sobre hombres y mujeres por separado).

Es responsabilidad del estudiante que todo el código funcione correctamente con una simple ejecución sin tener que editar nada. Para ello es imprescindible usar rutas relativas (./directorio/…) en vez de absolutas (/home/usuario/directorio…), puesto que el profesorado tendrá su propia estructura de directorios. Por lo tanto, no es posible usar rutas dentro del script que impliquen que tengamos que reproducir vuestro ambiente de trabajo para corregir la práctica. Este hecho supondrá una grave penalización.

1. Scripts

Puntuación: 7 puntos

B. Tal y como se ha comentado en la práctica 1, la práctica en su totalidad debe incorporar al menos la elaboración de **tres** scripts que hagan **transformaciones con los datos de entrada** (no se considerarán válidas simples visualizaciones, menús, barras de progreso, etc.).

En esta segunda entrega, debéis **crear los dos scripts que faltan** y que se llamarán **b.sh** (aunque sea un script en bash puede usar comandos sed, pero no awk) y **b.awk** (script íntegramente en awk), los cuales deberán cumplir obligatoriamente con los siguientes requisitos:

- Que contenga **una sentencia iterativa** que puede ser implementada a través de estructuras tipo `while` o `for`, teniendo en cuenta que la iteración implícita que se hace en el script AWK no es suficiente, por lo que para cumplir con lo que respecta a la estructura iterativa es necesario usar como mínimo un `while` o `for`.
- Que contenga **un mínimo de 7 líneas** que hagan transformaciones con los datos, por lo que **no se computarán sentencias que no manipulen los datos**, como las sentencias básicas para visualizar información (`echo`, `print, cat, etc)`, finalizar estructuras (`fi`, `do`, `done`, etc.), comentarios, asignaciones, etc. Es decir, los scripts deben demostrar el dominio adquirido en las herramientas del curso para el tratamiento de datos.
- Que **manipulen todos los tipos de datos obligatorios**: *enteros, texto y fechas o decimales* (no considerándose como manipulación la eliminación de una columna). Es muy importante que para cada manipulación realizada, en el último campo de la cabecera añadida en los comentarios (mostrada más abajo), se incluya el nombre del campo y entre paréntesis el tipo de dato manipulado, por ejemplo: ID (texto), nombre (carácter), compra (fecha), así como un comentario explicativo en el comando del código donde se lleva a cabo dicha manipulación.

Los **dos** scripts que se deben implementar han de contener la siguiente información (debidamente cumplimentada) en la cabecera a modo de comentario. **Los scripts que no contengan esta información totalmente cumplimentada**, mereciendo especial atención los dos últimos ítems, **no recibirán puntuación**:

```
#Nombre y apellidos del alumno:

#Usuario de la UOC del alumno:

#Fecha:

#Objetivos del script:

#Nombre, tipo y número de línea o líneas donde se realiza la manipulación:

Ejemplo: created (booleano) (19,20-23); description (texto) (24-25); etc.
```

El script en bash debe funcionar y ser ejecutado usando este tipo de invocación:

`./b.sh <nombreDelFicheroDeDatos>`

El script en awk debe funcionar y ser ejecutado usando este tipo de invocación:

`gawk -f b.awk <nombreDelFicheroDeDatos>`

(Este apartado vale 4 puntos)

C. Elaborad un script llamado `c` (la extensión será `.sh` o `.awk` dependiendo de la elección que hagáis) que **realice por lo menos una operación de agregación o categorización** *(han de hacerse obligatoriamente en este script no pudiendo hacerse en apartados anteriores)*, para posteriormente generar un documento en formato de texto plano con formato atractivo, o un HTML5 **con los resultados agregados por categorías o por intervalos**.

El script que no contenga la siguiente cabecera totalmente cumplimentada, mereciendo especial atención los tres últimos ítems, **no recibirá puntuación**:

```
#Nombre y apellidos del alumno:

#Usuario de la UOC del alumno:

#Fecha:

#Objetivo:

#Nombre y tipo de los campos de entrada: Ejemplo: birthday (fecha)

#Operaciones y nº línea o líneas donde se realizan: Ejemplo: agregación
(20-22); categorización (27-28); etc.

#Nombre y tipo de los nuevo campos generados: Ejemplo: age (entero)
```

Por ejemplo, si los datos fueran de acceso a una plataforma en línea, se podría realizar un informe con el número de usuarios por cada provincia, por edad o por otra agrupación de los resultados. Como se ha comentado, se valorará que se incluya la perspectiva de género si los datos la contemplan.

No se considerará válido una simple visualización (total o parcial) del dataset (por ejemplo, quitar columnas o filas). Debe haber cálculos en forma de agrupaciones o categorías con **una cabecera, y una leyenda** en caso de usar gráficas. Se revisará la veracidad de los cálculos y **se puntuará atendiendo a la sofisticación de la maquetación**, a saber, los recursos de formato (color, negrita, tablas y CSS, etc.). Además, **no se debe abrir programa alguno para visualizar la salida de forma automática**, es decir, no se debe lanzar el navegador.

El script debe funcionar usando una de las dos invocaciones siguientes dependiendo de la elección que hayais tomado:

```
./c.sh <nombreDelFicheroDeDatos>
```

```
gawk -f c.awk <nombreDelFicheroDeDatos>
```

(Este apartado vale 2.5 puntos)

D. Es necesario que haya un **script principal** (además de los anteriores), llamado `run.sh` que ejecute paso a paso todo el proyecto, llamando a los scripts anteriores. Debe encargarse de la ejecución de todos los scripts indicados en los apartados A, B, y C que forman parte del enunciado de la práctica 1 y la práctica 2.

El juego de pruebas usado durante la corrección descomprime automáticamente vuestro zip, otorga permisos de ejecución únicamente al script run.sh y luego lo ejecuta, de modo que todo lo que suceda después ya es responsabilidad del programador, es decir, vuestra.

Debéis tener en consideración que **todo debe funcionar sin tener que editar nada, ni llevar a cabo ninguna acción adicional** de ningún tipo. El **tiempo máximo de ejecución** no puede superar en ningún caso los **15 segundos en un entorno virtualizado**. Si vuestro script es muy lento puede ser que estéis haciendo alguna operación (bucles, etc.) de forma incorrecta.

El script debe funcionar usando la invocación:

./run.sh

(0.5 puntos)

2. Documento

Puntuación: 2 puntos

Responded en un documento en formato PDF a cada uno de las siguientes aspectos de forma ordenada y numerada:

A. Objetivos del proyecto (qué problema se pretende resolver, y por qué se puede resolver mediante el uso de *scripting*).

B. Explicar concisamente el funcionamiento de los scripts a, b1, b.sh y b.awk indicando de forma clara cuál es el objetivo y que se pretende con cada uno de ellos.

C. En los scripts b.sh y b.awk indica el nombre, el tipo, y el número de línea o líneas donde se realiza la manipulación para cada tipo de dato obligatorio que habéis de usar.

D. Genera una tabla en la que se indique: el campo o campos usados para la agrupación/categorización de los resultados (script C), el tipo de operación realizada, y los nuevos campos generados.

E. Adjunta una captura de pantalla con el informe que contiene los resultados del apartado C, y haz un análisis de los mismos. Establece también una relación con los objetivos del proyecto definidos en el primer apartado.

F. Crea un diagrama de flujo que modele la secuencia de transformación de los datos, es decir, el flujo de la información y sus transformaciones a través de los diferentes scripts de los apartados A, B, y C.

G. Explicar las tareas más importantes aprendidas con la elaboración del proyecto y las dificultades que hayáis encontrado. Además, proponed técnicas para mejorar o extender el proyecto.

3. Valoración global de la propuesta

Puntuación: 1 punto

Además de los puntos anteriores, se realizará una valoración global de esta primera parte: la claridad/presentación, el orden, la adecuación lingüística, los comentarios del código, la nomenclatura de las variables, buenas praxis de programación como no sobreescribir el dataset original, no llamar a programas de visualización, etc., la sofisticación del trabajo, cumplir los requisitos del enunciado, utilidad de los scripts de cara a los resultados y conclusiones finales, eficiencia computacional, tiempo de ejecución, complejidad del código, grado de elaboración en el formato y presentación de los resultados, grado de adecuación y presentación del documento, etc. *(1 punto)*

Resumen de la entrega

- Documento en pdf, con capturas de la ejecución de cada script con vuestro usuario y las respuestas a las preguntas indicadas en el apartado 2.
- Dataset con un mínimo de 300 registros y no superior a 1.5MB con campos de texto, enteros y fechas o decimales.
- Cada uno de los scripts listos para ser ejecutados en la versión de Ubuntu usada durante el curso atendiendo a los paquetes instalados en la PEC2.
- Debeis entregar por lo tanto todo lo que hayais entregado en la Práctica 1 (que deberá permanecer inalterado respecto a la entrega realizada) así como todo lo nuevo que se requiere en esta práctica 2.

Todo ello en un **fichero en formato zip** sin carpetas.

Comentarios

No es posible usar librerías de terceros, debiendo utilizarse sólo las herramientas oficiales del curso. En todo caso, consultadlo con vuestro profesor. Todo el código proporcionado **debe ejecutarse sin errores en la versión LTS de Ubuntu con los paquetes instalados en la PEC 2**.

Es **imprescindible** también que todo lo que se haga en este proyecto **se pueda reproducir mediante la ejecución de los scripts**, y que el tiempo máximo de

ejecución de todos los scripts del proyecto no supere los 15 segundos en una máquina virtual con Ubuntu usando el software que se ha indicado en las dos primeras PECs, por lo que no se podrá usar librería de terceros ni ningún software o lenguaje adicional.

IMPORTANTE: No se aceptará en ningún caso la entrega de la práctica después de la fecha máxima de entrega (02/07/2023 23:59:59). *Si por alguna razón pensáis que no vais a poder entregar a tiempo, consultadlo con vuestro profesor siempre con anterioridad*.