Práctica

Parte 1

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 y su resultado.

Buscad y elegid un **dataset** para su análisis, no olvidéis **incluirlo en el zip**. Dicho dataset 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. Un documento en formato **PDF**, con una extensión no superior a **3 páginas**, que incluya **capturas de pantalla**, donde se identifique vuestro usuario, y se vea la ejecución de cada script (el script a.sh deberá verse la invocación con y sin parámetro), junto con el resultado íntegro o bien una parte de éste. *Si no hay imágenes en la memoria donde se vea vuestro usuario, se considerará la*

práctica como plagiada y no recibiréis puntuación alguna.

2. Cada uno de los scripts solicitados se debe adjuntar listos para ser ejecutados, **con el nombre y la extensión que se indican en los planteamiento de los ejercicios, en los que obviamente 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 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).

Una forma de buscar datasets es a partir de uno de los sitios con estadísticas que permiten restringir el conjunto del total de lo que disponen. Esta restricción puede ser geográfica, temporal, por edad, etc.

Ejemplos:

OECD stat: https://stats.oecd.org

INE: https://www.ine.es/

Datos abiertos de la generalidad: https://analisi.transparenciacatalunya.cat/ca/

Socrata: https://dev.socrata.com/

Open Data Inception: portal de portales de datos accesible desde el área de recursos de la asignatura.

Se pueden encontrar unos datos cuyo contenido incluya todos los tipos solicitados, y si el dataset total resulta demasiado grande, en la propia web, elegid alguna o algunas restricciones que lo hagan menor de 1.5MB. También es posible ajustar cualquier conjunto de datos, no sólo de las fuentes mencionadas anteriormente, es decir, podéis manipular un fichero para quedaros con una parte suficiente para realizar la práctica. ¡De eso precisamente trata esta asignatura!

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

A. Cread un script denominado `a.sh` que descargue el dataset (si no es público debe subirse a google drive o similar) y utilizando las herramientas vistas durante el curso muestre:

- La URL de descarga del dataset.
- Formato del o de los ficheros que lo componen.
- Número total de columnas y registros.
- Para todas las columnas cuyos datos vayan a ser usados, el tipo de datos (entero, decimal, fecha, texto, etc) a partir de comandos (no vale escribir el texto literalmente a base de `echo` o `print`, etc). Recordad que debe haber columnas de tipo entero, texto y fecha o decimal (float).

El script deberá al menos contemplar una opción / argumento de línea de comandos.

Proponemos la siguiente idea:

Si el script se ejecuta sin opciones, muestra la URL de descarga y el número de columnas y registros. En cambio, si se incluye en su invocación la opción `-v` deberá además mostrar el formato del o de los ficheros y el tipo de datos de sus columnas.

En este sentido, os proporcionamos una muestra de `getops` a modo de ejemplo:

```
#!/bin/bash
```

```
while getopts "o:" option; do
  case $option in
    o) L=${OPTARG}
      echo "Ejemplo de uso de argumentos para dataset a --> $L"
      exit;;
...
  esac
done
```

El script debe funcionar usando la invocación (el uso entre corchetes del parámetro hace alusión a que dicho parámetro es opcional y, por lo tanto, el script debería funcionar igualmente si no se especifica):

./a.sh [-v]

(Este apartado vale 3 puntos)

B. La práctica final 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 primera entrega, **debéis crear el primero de estos scripts** que se llamará **b1.sh** (aunque sea un script en bash puede usar comandos sed, pero no awk) o **b1.awk** (si es un script íntegramente en awk), el cual deberá cumplir obligatoriamente con los siguientes requisitos:

- Incluir como mínimo **una sentencia condicional tipo if.**
- Incluir como mínimo **tres expresiones regulares en comandos distintos** (sin encadenar).
- Que contenga **un mínimo de 5 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.

Con respecto al cómputo de expresiones regulares para llegar al mínimo indicado, si se quiere usar sed u otro comando que en su argumento necesite una expresión regular, aunque ésta sea casi un literal, no hay problema. Sin embargo, si por ejemplo se usa esa misma expresión regular que es realmente un literal (no usa nada propio de las expresiones regulares como los cuantificadores), en un if, en el que claramente es mucho mejor usar una comparación entre strings (supongamos que se quiere comparar el valor de un campo con la cadena "%"), entonces no computará, ya que no hay un utilidad clara desde el punto de vista técnico. En caso de duda, consultadlo con vuestro profesor en el aula.

Dicho script debe comprobar la consistencia de los datos a usar para el resto de la práctica, es decir, que todos los datos de las columnas a usar tienen la misma estructura. Por ejemplo, el script podría comprobar cosas como que:

- Todas las fechas están en formato MM/DD/AAAA

- Todos los precios se muestran con exactamente dos decimales.

- Todas las notas de las asignaturas son menores o iguales a 10 y mayores que 0.

- El número de jugadores en una partida es positivo.

- Todos los números están en notación científica.

El script deberá, a ser posible, corregir los datos o en caso contrario, descartar los registros con datos incorrectos o sustituir su valor por "NA", "null" u otro valor similar.

El script generará un fichero “limpio” que se llamará `data_cleaned.csv`, ya que no se debe modificar el fichero original, es decir, el dataset que hayáis elegido. Este fichero `data_cleaned.csv` servirá de inicio de la parte 2 de esta práctica.

El script debe 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.
```

Si el script está en bash debe funcionar usando este tipo de invocación:

```
./b1.sh <nombreDelFicheroDeDatos>
```

Si el script está en awk debe funcionar usando este tipo de invocación:

```
gawk -f b1.awk <nombreDelFicheroDeDatos>
```

(Este apartado vale 4 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 y b1 indicando de forma clara cuál es el objetivo y que se pretende con cada uno de ellos.

C. En el script b1 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.

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.

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 (28/05/2023 23:59:59). *Si por alguna razón pensáis que no vais a poder entregar a tiempo, consultadlo con vuestro profesor siempre con anterioridad*.