

March 17, 2021

1 Fundamentos de Programación

1.1 PEC 4 - Enunciado

En este Notebook se encontraréis el conjunto de actividades evaluables como PEC de la asignatura. Veréis que cada una de ellas tiene asociada una puntuación, que indica el peso que tiene la actividad sobre la nota final de la PEC. Adicionalmente, hay un ejercicio opcional, que no tiene puntuación dentro de la PEC, pero que se valora al final del semestre de cara a conceder las matrículas de honor y redondear las notas finales. Podréis sacar la máxima nota de la PAC sin necesidad de hacer este ejercicio. El objetivo de este ejercicio es que sirva como pequeño reto para los estudiantes que quieran profundizar en el contenido de la asignatura.

Veréis que todas las actividades de la PEC tienen una etiqueta, que indica los recursos necesarios para llevarla a cabo. Hay tres posibles etiquetas:

- **NM Sólo materiales:** las herramientas necesarias para realizar la actividad se pueden encontrar en los materiales de la asignatura.
- **EG Consulta externa guiada:** la actividad puede requerir hacer uso de herramientas que no se encuentran en los materiales de la asignatura, pero el enunciado contiene indicaciones de dónde o cómo encontrar la información adicional necesaria para resolver la actividad.
- **EI Consulta externa independiente:** la actividad puede requerir hacer uso de herramientas que no se encuentran en los materiales de la asignatura, y el enunciado puede no incluir la descripción de dónde o cómo encontrar esta información adicional. Será necesario que el estudiante busque esta información utilizando los recursos que se han explicado en la asignatura.

Es importante notar que estas etiquetas no indican el nivel de dificultad del ejercicio, sino únicamente la necesidad de consulta de documentación externa para su resolución. Además, recordad que las **etiquetas son informativas**, pero podréis consultar referencias externas en cualquier momento (aunque no se indique explícitamente) o puede ser que podáis hacer una actividad sin consultar ningún tipo de documentación. Por ejemplo, para resolver una actividad que sólo requiera los materiales de la asignatura, podéis consultar referencias externas si queréis, ya sea tanto para ayudaros en la resolución como para ampliar el conocimiento!

En cuanto a la consulta de documentación externa en la resolución de los ejercicios, recordad **citar siempre la bibliografía utilizada** para resolver cada actividad.

1.2 Ejercicios para la PEC

NumPy - Ejercicios

1.2.1 Ejercicio 1

Crea y muestra por pantalla las matrices siguientes:

- Una matriz A 4x4 que contenga todos los números pares entre 0 y 31.
- Una matriz B 3x4 que contenga valores enteros aleatorios entre 0 y 100 ordenados de mayor a menor por columna.
- Una matriz C 4x4 donde las dos primeras filas de la matriz C corresponden a las dos primeras columnas de la matriz A transpuestas y las dos últimas filas de la matriz C corresponden a las dos últimas filas de la matriz B.

(0.5 puntos) NM

```
[ ]: # Respuesta
```

1.2.2 Ejercicio 2

Considerando las matrices del apartado anterior, muestra por pantalla el resultado de las siguientes operaciones:

- Permutación de las filas por columnas de la matriz B. ¿Cuáles son las dimensiones de la matriz permutada?
- Producto matricial de A por B transpuesta. Explique brevemente si se puede realizar el producto matricial de A por B.
- El determinante de C. Muestra por pantalla el resultado sin decimales.

(0.5 puntos) NM

```
[ ]: # Respuesta
```

1.2.3 Ejercicio 3

Crea una matriz de 5x5 y rellena con muestras aleatorias a partir de una distribución gama considerando que el factor de escala es 3 y el factor de forma es 2. (0.5 puntos) EG

Nota: Para resolver este ejercicio puede ser de utilidad la función [random](#).

```
[ ]: # Respuesta
```

1.2.4 Ejercicio 4

Crea una matriz de 5x5 donde los valores de cada posición (i,j) corresponden a $i^{(j+1)}$ para todo i primo, y $i+j^2$ para el resto de posiciones. (1 punto) EG

Nota: Para resolver este ejercicio puede ser de utilidad la instrucción `break`.

```
[ ]: # Respuesta
```

1.2.5 Ejercicio 5

Crea una matriz 10x10 con 0 la diagonal principal y la opuesta y el resto de valores 1, se tiene que obtener una X. Comprueba el resultado visualmente con la función `imshow` de Matplotlib que nos permite visualizar una imagen 2D. **(1 punto)** EG

Nota: Para resolver este ejercicio puede ser de utilidad la función `fill_diagonal`.

```
[ ]: # Respuesta
```

Matplotlib - Ejercicios

1.2.6 Ejercicio 1

Representa en un único gráfico las funciones [hiperbólicas](#) $\sinh(x)$, $\cosh(x)$ y $\tanh(x)$, evaluadas en el intervalo $[-5, 5]$ y con paso (resolución) 0.01. Haz que los rangos de valores mostrados sean $[-5, 5]$ para el eje X y $[-3, 3]$ por el eje Y. Adicionalmente, muestra la gráfica correspondiente a $\sinh(x)$ con color verde, la gráfica correspondiente a $\cosh(x)$ con color rojo y la gráfica correspondiente a $\tanh(x)$ con color azul. Se tiene que mostrar la leyenda en la parte superior derecha.

Nota: Podéis consultar la documentación de funciones matemáticas de [numpy](#)

(1 punto) NM

```
[8]: # Definimos las funciones
def f1(x):
    return np.sinh(x)

def f2(x):
    return np.cosh(x)

def f3(x):
    return np.tanh(x)
```

```
[ ]: # Respuesta
```

1.2.7 Ejercicio 2

Carga los datos del archivo `VideoGamesSales`, que puedes encontrar en la carpeta `data`, en forma de dataframe. Este conjunto de datos de la web [Kaggle](#) recoge información de más de 6000 videojuegos.

Parte A **(1 punto)** :

¿Qué tipo de juego tiene más unidades a la venta y para qué plataforma? Responda la pregunta gráficamente. La visualización del gráfico debe ser óptima (leyenda, etiquetas en los ejes, ...).

Parte B **(0.5 puntos)** :

Muestra un gráfico de líneas de puntos de las ventas por año. Se debe generar una línea para cada mercado reportado en el dataset: América del Norte (NA), Europa (EU), Japón (JP), otros

(Other) y Global. ¿Qué mercado contribuye más a las ventas Global de videojuegos?

Consideraciones

- La línea debe ser con puntos (*dotted line*) de diferentes colores según mercado y de tamaño 2.
- Se debe mostrar la leyenda en la parte superior izquierda.

(1.5 puntos) EG

```
[ ]: # Respuesta
```

pandas - Ejercicios

1.2.8 Ejercicio 1

Carga los datos del archivo `clinvar_conflicting.csv`, que puedes encontrar en la carpeta `data`, en un *dataframe*. Este conjunto de datos es un subconjunto del que encontraréis en la web [Kaggle](#). Este conjunto de datos recoge información relacionada con variantes genéticas según la base de datos [Clinvar](#).

Muestra por pantalla las variables del dataset, el número de filas y columnas.

(0.5 puntos) NM

```
[ ]: # Respuesta
```

1.2.9 Ejercicio 2

La variable *IMPACT* es una categorización del posible impacto de la variante en la síntesis de la proteína. Muestra las diferentes categorías de la variable *IMPACT*. Para cada cromosoma, agrupa el número de variantes según impacto. Muestra el resultado de mayor a menor según la categoría *HIGH* de la variable *IMPACT*. **(0.5 puntos)** NM

```
[ ]: # Respuesta
```

1.2.10 Ejercicio 3

A continuación nos centraremos con las variantes del chr11 y de impacto *High*.

- Crea un nuevo dataset con sólo estos datos. Muestra por pantalla las 5 primeras líneas.
- Muestra los genes (*SYMBOL*) y el número de variantes con un alto impacto sobre la proteína en el chr11. ¿Cuál es el gen con más variantes?
- Toda variante se define por su base de referencia (*REF*) y su correspondiente cambio (*ALT*). Para más información puede consultar el siguiente [enlace](#). Utilizando el dataset generado para las variantes con alto impacto en el chr11. Determine el cambio (*ALT*) que tiene una longitud mayor para el gen *ATM* y su correspondiente longitud.

(1 punto) NM

```
[ ]: # Respuesta
```

1.2.11 Ejercicio 4

Las enfermedades [minoritarias](#) son un conjunto muy amplio y heterogéneo de enfermedades, generalmente, monogénicas. Son enfermedades graves, crónicas y degenerativas. La frecuencia de una enfermedad minoritaria es menor a un caso por cada 1000 habitantes. Considerando esta premisa y el dataset original, realiza las siguientes tareas:

- Crea una nueva columna con un valor booleano para indicar si la variante tiene una frecuencia (*AF_EXAC*) inferior al 0.001. (True si es inferior o igual a 0.001 y False si es superior). Muestra por pantalla las primeras 10 líneas.
- Si asumimos que sólo aquellas variantes con frecuencias (*AF_EXAC*) inferiores a 0.001 pueden ser putativas de ser la causa de una enfermedad minoritaria y, por tanto, tener un impacto alto sobre la proteína. ¿Cuántas variantes con una frecuencia inferior a 0.001 tienen un impacto alto a la síntesis de la proteína? Muestra en un gráfico circular (*pie chart*) el número de variantes por categoría *IMPACT* en el conjunto de variantes con frecuencias inferiores o igual a 0.001.

Consideraciones:

- Se debe incluir el porcentaje (%) de cada categoría de Impacto dentro de su respectivo espacio del gráfico.
- Se debe mostrar la leyenda en la parte derecha.

(1 punto) NM

```
[ ]: # Respuesta
```

SciPy - Ejercicio

1.2.12 Ejercicio 1

Resuelve numéricamente la siguiente integral triple definida.

$$\int_{z=0}^{z=5} \int_{y=0}^{y=2} \int_{x=0}^{x=3} x^2 y (\cos(\pi * z) + 2) dx dy dz$$

Nota: Podéis utilizar la librería [integrate](#) (1 punto) EG

```
[ ]: # Respuesta
```

1.2.13 Ejercicio Opcional

Realiza el ejercicio 1.A de *Matplotlib* utilizando la función *heatmap* de la librería *Seaborn*. Los ejes deben mostrar los nombres de las etiquetas, debéis incluir la leyenda, el valor absoluto de cada tipo de juego dentro de su respectivo espacio del gráfico y la paleta de colores debe ser “YlGnBu”. La visualización del heatmap debe permitir leer el valor de cada tipo de juego.

```
[1]: # Respuesta
```
