Ejercicios evaluables

1. Tal y como ya hemos visto en clase, la variedad de herramientas proporcionadas por el álgebra lineal son cruciales para desarrollar y fundamentar las bases de una variedad de técnicas relacionadas con el aprendizaje automático. Con ella, podemos describir el proceso de propagación hacia adelante en una red neuronal, identificar mínimos locales en funciones multivariables (crucial para el proceso de retropropagación) o la descripción y empleo de métodos de reducción de la dimensionalidad, como el análisis de componentes principales (PCA), entre muchas otras aplicaciones.

   Cuando trabajamos en la práctica dentro de este ámbito, la cantidad de datos que manejamos puede ser muy grande, por lo que es especialmente importante emplear algoritmos eficientes y optimizados para reducir el coste computacional en la medida de lo posible. Por todo ello, el objetivo de este ejercicio es el de ilustrar las diferentes alternativas que pueden existir para realizar un proceso relacionado con el álgebra lineal y el impacto que puede tener cada variante en términos del coste computacional del mismo. En este caso en particular, y a modo de ilustración, nos centraremos en el cálculo del determinante de una matriz.

   a) [1 punto] Implementa una función, `determinante_recursivo`, que obtenga el determinante de una matriz cuadrada utilizando la definición recursiva de Laplace.

   b) [0.5 puntos] Si A es una matriz cuadrada $n \times n$ y triangular (superior o inferior, es decir, con entradas nulas por debajo o por encima de la diagonal, respectivamente), ¿existe alguna forma de calcular de forma directa y sencilla su determinante? Justifíquese la respuesta.

   c) [0.5 puntos] Determínese de forma justificada cómo alteran el determinante de una matriz $n \times n$ las dos operaciones elementales siguientes:

   - Intercambiar una fila (o columna) por otra fila (o columna).
   - Sumar a una fila (o columna) otra fila (o columna) multiplicada por un escalar α.

   d) [1 punto] Investiga sobre el método de eliminación de Gauss con pivoteo parcial e impleméntalo para escalonar una matriz (es decir, convertirla en una matriz triangular inferior) a partir de las operaciones elementales descritas en el apartado anterior.

   e) [0.5 puntos] ¿Cómo se podría calcular el determinante de una matriz haciendo beneficio de la estrategia anterior y del efecto de aplicar las operaciones elementales pertinentes? Implementa una nueva función, `determinante_gauss`, que calcule el determinante de una matriz utilizando eliminación gaussiana.

   f) [0.5 puntos] Obtén la complejidad computacional asociada al cálculo del determinante con la definición recursiva y con el método de eliminación de Gauss con pivoteo parcial.

   g) [1 punto] Utilizando `numpy.random.rand`, genera matrices cuadradas aleatorias de la forma $A_n \in \mathbb{R}^{n \times n}$, para $2 \leq n \leq 10$, y confecciona una tabla comparativa del tiempo de ejecución asociado a cada una de las variantes siguientes, interpretando los resultados:

   - Utilizando `determinante_recursivo`.
   - Empleando `determinante_gauss`.

- Haciendo uso de la función preprogramada `numpy.linalg.det`.

2. En este ejercicio trabajaremos con el método de descenso de gradiente, el cual constituye otra herramienta crucial, en esta ocasión de la rama del cálculo, para el proceso de retropropagación asociado al entrenamiento de una red neuronal.

   a) [1 punto] Prográmese en Python el método de descenso de gradiente para funciones de n variables. La función deberá tener como parámetros de entradas:

   - El gradiente de la función que se desea minimizar ∇f (puede venir dada como otra función previamente implementada, `grad_f`, con entrada un vector, representando el punto donde se quiere calcular el gradiente, y salida otro vector, representando el gradiente de f en dicho punto).
   - Un valor inicial $\boldsymbol{x}_0 \in \mathbb{R}^n$ (almacenado en un vector de n componentes).
   - El ratio de aprendizaje γ (que se asume constante para cada iteración).
   - Un parámetro de tolerancia `tol` (con el que finalizar el proceso cuando $\|\nabla f(x)\|_2 <$ tol).
   - Un número máximo de iteraciones `maxit` (con el fin de evitar ejecuciones indefinidas en caso de divergencia o convergencia muy lenta).

   La salida de la función deberá ser la aproximación del x que cumple $f'(x) \approx 0$, correspondiente a la última iteración realizada en el método.

   b) Sea la función $f : \mathbb{R} \to \mathbb{R}$ dada por

   $$f(x) = 3x^4 + 4x^3 - 12x^2 + 7.$$

   I [0.5 puntos] Aplica el método sobre $f(x)$ con $x_0 = 3$ $\gamma = 0.001$, `tol=1e-12`, `maxit=1e5`.

   II [0.5 puntos] Aplica de nuevo el método sobre $f(x)$ con $x_0 = 3$, $\gamma = 0.01$, `tol=1e-12`, `maxit=1e5`.

   III [0.5 puntos] Contrasta e interpreta los dos resultados obtenidos en los apartados anteriores y compáralos con los mínimos locales obtenidos analíticamente. ¿Qué influencia puede llegar a tener la elección del ratio de aprendizaje γ?

   IV [0.5 puntos] Aplica nuevamente el método sobre $f(x)$ con $x_0 = 3$, $\gamma = 0.1$, `tol=1e-12`, `maxit=1e5`. Interpreta el resultado.

   V [0.5 puntos] Finalmente, aplica el método sobre $f(x)$ con $x_0 = 0$, $\gamma = 0.001$, `tol=1e-12`, `maxit=1e5`. Interpreta el resultado y compáralo con el estudio analítico de f. ¿Se trata de un resultado deseable? ¿Por qué? ¿A qué se debe este fenómeno?

   c) Sea la función $g : \mathbb{R}^2 \to \mathbb{R}$ dada por

   $$g(x, y) = x^2 + y^3 + 3xy + 1.$$

   I [0.5 puntos] Aplíquese el método sobre $g(x, y)$ con $\boldsymbol{x}_0 = (-1, 1)$, $\gamma = 0.01$, `tol=1e-12`, `maxit=1e5`.

II [0.5 puntos] ¿Qué ocurre si ahora partimos de $\boldsymbol{x}_0 = (0, 0)$? ¿Se obtiene un resultado deseable?

III [0.5 puntos] Realícese el estudio analítico de la función y utilícese para explicar y contrastar los resultados obtenidos en los dos apartados anteriores.