

Tarea 3

Inteligencia Artificial

1. Entregar un documento PDF con todas tus respuestas teoricas. No se aceptan otro tipo de formato.
2. Para la parte de codificación todas las respuestas deben estar en orden y etiquetadas de manera clara y correcta para recibir puntaje.
3. Todo acto de COPIA implica la nota de 0A. Evita copiar!.

El objetivo de esta tarea es poder repasar tus habilidades matemáticas y de programación necesarias para el curso. Si cumple con los requisitos previos, estos problemas deberían ser relativamente simples. Algunos de estos problemas volverán a ocurrir como subproblemas de tareas posteriores, así que asegúrese de saber cómo resolverlos **sin copiar**.

1 Ejercicios teóricos

1. Supongamos que te dan una bolsa que contiene n monedas no sesgadas. También se te dice que $n - 1$ de estas monedas son normales, es decir, tienen una cara en un lado y un sello en el otro. La moneda restante es falsa y tiene dos caras en ambos lados.
 - (a) Supongamos que se escoge una moneda de la bolsa al azar y luego la lanzas y obtienes una cara. Dado este resultado, ¿cuál es la probabilidad de que la moneda que elija sea la moneda falsa? (Tenga en cuenta que estamos pidiendo una probabilidad condicional).
 - (b) Supongamos que continuas lanzando la misma moneda un total de k veces y ves k caras. Ahora bien, ¿cuál es la probabilidad condicional de que hayas elegido la moneda falsa?.
 - (c) Ahora supongamos que diseñas el siguiente método para determinar si la moneda es falsa o no. Tu lanzas k veces y tu método devuelve que es una moneda falsa si todos los k lanzamientos dan como resultado caras. De lo contrario, el método devuelve que es una moneda normal. ¿Cuál es la probabilidad de que este método devuelva un error?.
2. Calcula los valores singulares y los vectores singulares derechos e izquierdos de la siguiente matriz.

$$A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

y además el SVD de la matriz:

$$A = \begin{pmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{pmatrix}$$

3. En un concurso de televisión, el concursante debe elegir entre tres puertas cerradas. Detrás de una puerta aguarda un premio: un coche. Detrás de las otras dos puertas hay cabras. El concursante elige una puerta, por ejemplo la número uno. El anfitrión, que sabe dónde está el coche, abre otra puerta, por ejemplo la número tres y aparece una cabra. El concursante tiene ahora la oportunidad de elegir entre las dos puertas restantes (una y dos). ¿Cuál es la mejor opción desde su punto de vista? ¿Quedarse con la puerta que eligió originalmente o cambiar a la otra puerta cerrada?.

4. Sea X una variable aleatoria normal bivariada (tomando valores en R^2) con media $\mu = (1, 1)$ y una matriz de covarianza $\Sigma = \begin{pmatrix} 3 & 1 \\ 1 & 2 \end{pmatrix}$. La densidad de X viene dada por:

$$f_X(x) = \frac{1}{\sqrt{(2\pi)^2 \det(\Sigma)}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right)$$

Encuentra la distribución condicional de $Y = X_1 + X_2$ dado $Z = X_1 - X_2 = 0$.

5. Sean x_1, \dots, x_n números reales que representan posiciones en una recta numérica. Sean w_1, \dots, w_n números reales positivos que representan la importancia de cada una de estas posiciones. Considera la función cuadrática: $f(\theta) = \frac{1}{2} \sum_{i=1}^n w_i (\theta - x_i)^2$. ¿Qué valor de θ minimiza $f(\theta)$? ¿Qué problemas podrían surgir si algunos de los w_i son negativos?
6. Para $w \in R^d$ (representado como un vector de columna) y constantes $a_i, b_j \in R^d$ (también representado como vectores de columna) y $\lambda \in R$, se define la función con valores escalares:

$$f(w) = \sum_{i=1}^n \sum_{j=1}^n (a_i^T w - b_j^T w)^2 + \lambda \|w\|_2^2.$$

donde $w = (w_1, \dots, w_d)^T$ y $\|w\|_2 = \sqrt{\sum_{k=1}^d w_k^2}$ es la norma L_2 . Calcula el gradiente $\nabla f(w)$.

7. Considera la función con valores escalares $f(w)$ del problema anterior. Diseña una estrategia que primero haga el preprocesamiento en $O(nd^2)$ y luego para cualquier vector dado w tome $O(d^2)$ para calcular $f(w)$.
8. Supongamos que tenemos la imagen de un rostro humano que consta de $n \times n$ píxeles. En nuestra configuración simplificada, una cara consta de dos ojos, dos oídos, una nariz y una boca, cada uno representado como un rectángulo arbitrario alineado con el eje (es decir, los ejes del rectángulo están alineados con los ejes de la imagen). Como también nos gustaría manejar los retratos de Escher, no existen restricciones en la ubicación o el tamaño de los rectángulos. ¿Cuántas caras posibles (elección de los rectángulos que lo componen) hay? En general, para la pregunta nos preocupamos por la complejidad asintótica, así que entrega tu respuesta en forma de $O(n^c)$ u $O(c^n)$ para algún número entero c .
9. Supongamos que tenemos una cuadrícula de $n \times n$. Comenzamos en la esquina superior izquierda (coordenada $(1, 1)$) y nos gustaría llegar a la esquina inferior derecha (coordenada (n, n)) dando pasos individuales hacia abajo y hacia la derecha. Define una función $c(i, j)$ como el costo de tocar el cuadrado (i, j) y supongamos que se necesita un tiempo constante para calcular. Proporciona un algoritmo para calcular el costo mínimo de la manera más eficiente. ¿Cuál es el tiempo de ejecución?
10. Calcule el gradiente de la función de pérdida de regresión lineal:

$$Loss(x, y, w) = \begin{cases} \frac{1}{2}(w \cdot \phi(x) - y)^2 & \text{para } |w \cdot \phi(x) - y| \leq 1, \\ |w \cdot \phi(x) - y| - \frac{1}{2} & \text{para } |w \cdot \phi(x) - y| > 1. \end{cases}$$

¿Cuál es el gradiente, $\nabla_w Loss(x, y, w)$ evaluado con el vector de peso $w = [1, 1]$ en el ejemplo (x, y) con $\phi(x) = [4, 10]$ e $y = 16$?

2 Ejercicios de programación

Debes modificar el código en `respuestas.py` entre

```
# Inicio de Código
# Completar
# Fin de Código
```

pero puede agregar otras funciones auxiliares fuera de este bloque si lo deseas. No realices cambios en archivos que no sean `respuestas.py`.

Completa el cuaderno de jupyter notebook, donde se pueda visualizar todos tus resultados.

2.1 Opcional -solo para esta practica

Tu código se evaluará en dos tipos de casos de prueba, básicos y ocultos, que puedes ver en `prueba.py`. Las pruebas básicas, no sobrecargan tu código con entradas grandes o casos complicados. Las pruebas ocultas son más complejas y sobrecargan tu código. Las entradas de las pruebas ocultas se proporcionan en `prueba.py`, pero las salidas correctas no. Para ejecutar las pruebas, necesitará tener `pruebasUtil.py` en el mismo directorio que tu código y `pruebas.py`. Luego, puede ejecutar todas las pruebas escribiendo

```
python prueba.py
```

Esto te dirá solo si pasó las pruebas básicas. En las pruebas ocultas, el script le alertará si su código tarda demasiado o se bloquea, pero no dice si obtuvo el resultado correcto. También puedes ejecutar una única prueba (por ejemplo, 1-0-basico) escribiendo:

```
python prueba.py 1-0-basico
```

Se recomienda que leas y comprendas los casos de prueba, crea tus propios casos de prueba y no simplemente ejecute las cosas así, para las futuras evaluaciones.