# Trabajo final

Teoría y Algoritmia de Optimización

2022

## Instrucciones

**Objetivo** El objetivo del examen es mostrar que el estudiante es capaz de resolver una serie de problemas teóricos e implementar y analizar una serie de problemas prácticos. En el primer caso, es fundamental justificar cualquier paso no trivial de la resolución. En el caso de problemas prácticos, es fundamental analizar y comentar todo resultado que se obtenga.

**Contenido** El entregable debe contener: resolución detallada de problemas teóricos, resultados de los problemas prácticos, análisis y discusión de ellos. No es necesario (ni aconsejable) incluir: letra de ejercicios, código de los ejercicios prácticos. El código de los ejercicios prácticos debe incluirse en un archivo aparte para posible referencia por parte de los docentes.

**Autoría** Esta es un examen *individual*. Sus ejercicios deben ser resueltos por el estudiante cuyo nombre, cédula y firma se deben incluir en la carátula del informe. No es admisible la realización colectiva de ninguno de los ejercicios ni sus partes. Tampoco es admisible la búsqueda y/o reutilización, total o parcial, de material en Internet u otros medios, así como entregas disponibles de años anteriores.

Sí es admisible y aconsejable consultar, cotejar, e intercambiar ideas y sugerencias con otros estudiantes.

También es admisible utilizar material de referencia tales como: documentación sobre lenguajes de programación, resultados, definiciones y propiedades matemáticas, incluyendo todo el material expuesto en el teórico de este curso, tanto teórico como práctico.

También es admisible la reproducción e inclusión de recetas y código relacionado con aspectos auxiliares, tales como el graficado de funciones, etc., que no hacen al objetivo de los ejercicios. Se sugiere citar la fuente para evitar posibles confusiones.

**Sanciones** Cualquier violación a las anteriores reglas constituye una *falta disciplinaria* y tendrá como consecuencia la pérdida del curso.

**Conformidad** El examen debe incluir una carátula identificando claramente a) fecha y b) nombre, cédula, y firma (preferentemente digital) del autor y c) el siguiente texto, copiado textualmente:

i) He leído y estoy de acuerdo con las Instrucciones especificadas en el obligatorio. ii) He resuelto por mi propia cuenta los ejercicios, sin recurrir a informes de otros compañeros, o soluciones existentes. iii) el presente informe y todo programa implementado como parte de la resolución del trabajo son de mi autoría y no incluyen partes ni fragmentos tomados de otros informes u otras fuentes, salvo las excepciones mencionadas.

**Puntaje** El puntaje total de todos los ejercicios del examen es de **140 puntos**. Con **100 puntos** se alcanza la nota máxima nominal de la prueba (que cuenta como un **40 %** del total del curso).

Puntos obtenidos por encima de 100 se tendrán en cuenta y permitirán sumar hasta un **10 % adicional** del total del curso.

## Ejercicio 1 - Norma dual (40 puntos)

Dada una norma vectorial cualquiera $\|\mathbf{x}\|$, su norma dual $\|\mathbf{x}\|_*$ se define como:

$$\|\mathbf{x}\|_* = \max \ \mathbf{z}^T\mathbf{x} \quad \text{sujeto a} \quad \|\mathbf{z}\| \leq 1. \tag{1}$$

En el curso vimos las normas duales para algunas normas conocidas. En particular se mencionó que la norma dual de la $\ell_\infty$, $\|\mathbf{x}\|_\infty = \max\{|x_i| : i = 1, \ldots, n\}$, es la norma $\ell_1$, $\|\mathbf{x}\|_1 = \sum_i |x_i|$. El objetivo de este ejercicio es mostrar que la solución de:

$$\max \ \mathbf{z}^T\mathbf{x} \quad \text{sujeto a} \quad \|\mathbf{z}\|_\infty \leq 1, \tag{2}$$

es efectivamente la norma $\ell_1$ de $\mathbf{x}$.

**a)** Indique y justifique si se cumple la dualidad fuerte en este caso.

**b)** Escribir (2) como un problema equivalente de *minimización* con restricciones de desigualdad *lineales*. Sugerencias: i) la desigualdad $\|\mathbf{z}\|_\infty \leq k$ equivale a $n$ desigualdades $|z_i| \leq k$ y ii) la condición $|z_i| \leq k$ puede escribirse como $-k \leq z_i \leq k$.

**c)** Plantear el Lagrangeano $L(\mathbf{z}, \lambda)$ y resolver $d(\lambda) = \min_{\mathbf{z}} L(\mathbf{z}, \lambda)$ *s.a.* $\lambda \geq 0$.

**d)** Plantear y resolver el problema dual, $\rho = \max_{\lambda \geq 0} d(\lambda)$. Sugerencia: notar que si el par $(a, b)$ cumple la restricción $a - b = c$ entonces el par $(a + \delta, b + \delta)$ también la cumple (es obvio pero es útil tenerlo presente).

## Ejercicio 2 - Anti-LASSO (30 puntos)

Sea un problema de regresión lineal en donde queremos encontrar un vector $\mathbf{x} \in \mathbb{R}^n$ tal que $\mathbf{A}\mathbf{x} \approx \mathbf{b}$ en donde $\mathbf{A} \in \mathbb{R}^{m \times n}$ es una matriz arbitraria y $\mathbf{b} \in \mathbb{R}^m$ es un vector también arbitrario. El LASSO (Least Absolute Subset Selection Operator) es una variante regularizada del método de mínimos cuadrados en donde se penaliza la norma $\ell_1$ de la solución:

$$\mathbf{x}^* = \arg\min \frac{1}{2\beta}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \|\mathbf{x}\|_1, \tag{3}$$

donde $\beta > 0$ es un parámetro que controla el grado de regularización: a mayor valor de $\beta$, mayor el efecto de regularización.

La gracia del LASSO es que, además de regularizar, permite obtener soluciones en donde algunos (o muchos) de los elementos de $\mathbf{x}^*$ son exactamente cero. Esto tiene varias aplicaciones, entre ellas la de obtener una selección automática de variables relevantes.

La idea central de este ejercicio es resolver el siguiente problema de regresión regularizada, al que llamaremos Anti-LASSO:

$$\mathbf{x}^* = \arg\min \frac{1}{2\beta}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \|\mathbf{x}\|_\infty, \tag{4}$$

donde la norma infinito se define como $\|\mathbf{x}\|_\infty = \max |x_i|$. Notar que la única diferencia entre el LASSO y esta formulación es la sustitución de la norma $\ell_1$ por la $\ell_\infty$. Le llamamos *anti-LASSO* porque la solución de este problema tiene propiedades diametralmente opuestas a la anterior: los elementos de $\mathbf{x}^*$ son todos no nulos y son lo más parecidos entre sí en términos de magnitud. Si bien no es popular, esta formulación tiene también su utilidad.

Al igual que la norma $\ell_1$, la norma $\ell_\infty$ es no diferenciable, por lo que buscaremos resolver este problema utilizando métodos proximales.

**a)** Utilizando la descomposición de Moreau, expresar el operador proximal de la norma $\|\mathbf{x}\|_\infty$, en función de la proyección sobre una bola con norma $l_1$. Recordar que la norma dual de $\ell_\infty$ es $\ell_1$.

**b)** Resolver (4) utilizando el método de Proximal Gradient Descent para $\beta = 1$ con los datos $\mathbf{A}$ y $\mathbf{b}$ contenidos en los archivos `A.txt` y `b.txt`. Para esta parte será útil la función `proj_l1` del archivo `util.py`. Experimente con distintos parámetros del método.

**c)** Verificación y monitoreo: el archivo `x_antilasso.txt` contiene la solución, $\mathbf{x}^*$. Grafique $\|\mathbf{x}^* - \mathbf{x}^k\|$ en función de la iteración $k$, verifique que dicho valor tiende a 0 y reporte la diferencia mínima alcanzada.

Nota: no puede usarse el valor de $x^*$ como criterio de parada del algoritmo PGD, porque en teoría no se conoce! Debe usarse algún otro criterio visto en el curso.

## Ejercicio 3 - Regresión lineal robusta (35 puntos)

El problema de regresión lineal consiste en hallar un vector $\mathbf{x} \in \mathbb{R}^n$ tal que $\mathbf{Ax} \approx \mathbf{b}$ para una cierta matriz $\mathbf{A} \in \mathbb{R}^{m \times n}$ y un vector $\mathbf{b} \in \mathbb{R}^m$. El criterio más común utilizado es el error cuadrático mínimo:

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{Ax} - \mathbf{b}\|_2^2. \tag{5}$$

El problema con este critero es que la solución es demasiado sensible a valores que se apartan mucho del modelo (*outliers*). La regresión robusta busca sustituir el error cuadrático $\|\cdot\|_2^2$ por otro $h(\cdot)$ que sea menos sensible a outliers:

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \ h(\mathbf{Ax} - \mathbf{b}). \tag{6}$$

En este ejercicio proponemos $h(\mathbf{e}) = \sum_i h(e_i)$ donde $h(e_i) = e_i^2/(|e_i| + 1)$.

**a)** Determine si la función escalar $h(e)$ es: i) continua, ii) diferenciable.

**b)** Comparar gráficamente $h(e)$ contra el error cuadrático $e^2$ a dos escalas, $|e| \ll 1$ y $|e| \gg 1$ y verificar gráficamente si $h(e)$ efectivamente se comporta como queremos.

**c)** Halle el gradiente de $h$, y exprese el gradiente de la función objetivo $g(\mathbf{x}) = h(\mathbf{Ax} - \mathbf{b})$, en función del gradiente de $h$. Para esto se sugiere definir $\mathbf{e}(\mathbf{x}) = \mathbf{Ax} - \mathbf{b}$ y luego utilizar la regla de la cadena:

$$\frac{\partial}{\partial \mathbf{x}} h(\mathbf{e}) = \frac{\partial h(\mathbf{e})}{\partial \mathbf{e}} \frac{\partial \mathbf{e}}{\partial \mathbf{x}}.$$

**d)** Resuelva el problema de regresión robusta (6) utilizando descenso por gradiente y paso de Armijo para los mismos datos `A.txt` y `b.txt` del ejercicio anterior.

**e)** Verificación y monitoreo: hacer lo mismo que en el ejercicio anterior pero usando como referencia el archivo `x_robusto.txt`.

## Ejercicio 4 - Predicción adaptiva (o en línea) (35 puntos)

El problema general de predicción implica, en cada paso de tiempo $n$, estimar el próximo valor $x_{n+1}$ de una variable a partir de la secuencia de los valores observados hasta el momento, $(x_1, x_2, \ldots, x_n)$. Un modelo sencillo y muy utilizado es el de predicción lineal:

$$\hat{x}_{n+1} = \sum_{j=0}^{k-1} a_j x_{n-j},$$

donde $1 \leq k \leq n$ se denomina el *orden* del predictor y es un parámetro a elegir. El objetivo es encontrar $\mathbf{a} \in \mathbb{R}^k$ que minimiza el error de predicción cuadrático esperado:

$$\mathbb{E}\left[(x_{n+1} - \hat{x}_{n+1})^2\right] = \mathbb{E}\left[(x_{n+1} - \sum_{j=0}^{k-1} a_j x_{n-j})^2\right] = \mathbb{E}\left[(x_{n+1} - \mathbf{a}^T \mathbf{x}_n)^2\right].$$

donde $\mathbf{x}_n = [x_n, x_{n-1}, \ldots, x_{n-k+1}]$ es el vector de los $k$ valores anteriores a $x_{n+1}$.

Cuando se habla de predicción adaptiva o en línea, el problema anterior se aproxima utilizando el método de gradiente estocástico a medida que van llegando nuevos datos para actualizar $\mathbf{a}$.

En nuestro caso tendremos una secuencia de muestras $(x_1, x_2, \ldots, x_N)$. Las primeras $k$ muestras no pueden ser predichas porque no hay suficientes muestras pasadas. A partir de $n \geq k$, cada iteración $n$ toma $x_{n+1}$ como la "nueva muestra", y las $k$ muestras anteriores como $\mathbf{x}_n$. En base a $\mathbf{x}_n$ se construye la predicción $\hat{x}_{n+1}$ (que vamos a almacenar en un vector para luego comparar con la secuencia original), se calcula el error $e_{n+1} = (x_{n+1} - \hat{x}_{n+1})^2$, y luego se actualiza $\mathbf{a}$ utilizando un paso del algoritmo SGD.

El algoritmo anterior es conocido como Least Mean Squares (LMS) y es muy utilizado, por ejemplo, en cancelación adaptiva de ruido en auriculares y cancelación de eco en videoconferencias y celulares.

**a)** Escriba la iteración del algoritmo LMS para el objetivo:

$$\min_{\mathbf{a}} \mathbb{E}\left[(x_{n+1} - \mathbf{a}^T \mathbf{x}_n)^2\right]$$

para $n$ y $k \leq n$ genéricos.

**b)** Implemente el algoritmo LMS y aplíquelo a la serie de datos que se encuentra en el archivo `serie.txt` usando $k = 4$. **Nota:** En este algoritmo no hay *condición de parada*: se sigue actualizando hasta el final de la serie.

**c)** Grafique la evolución del error de predicción absoluto $|\hat{x}_{n+1} - x_{n+1}|$.

**d)** Verificación y monitoreo: hacer lo mismo que en los otros ejercicios pero con el archivo `a.txt` como referencia (la solución no tiene por qué ser idéntica a la referencia).