Programación: mínimos cuadrados usando QR

Objetivos. Programar una función que resuelva el problema de mínimos cuadrados usando la descomposición QR. Se supone que el rango de la matriz coincide con el número de sus columnas.

1. Funciones que programamos anteriormente. Suponemos que ya hemos programado las siguientes funciones:

- `myqr(A)` construye una factorización QR reducida de la matriz dada A, cuyas columnas son linealmente independientes.
- `solveut(U, b)` resuelve el sistema de ecuaciones lineales $Ux = b$, si la matriz dada U es triangular superior e invertible.

2. Problema de mínimos cuadrados. Sean $A \in \mathcal{M}_{n\times m}(\mathbb{R})$ y $b \in \mathbb{R}^n$. Suponemos que $\mathrm{r}(A) = m \leq n$. Hay que encontrar un vector $z \in \mathbb{R}^m$ que minimice la función $f(x) \coloneqq \|Ax - b\|^2$.

3. Gradiente de la función que queremos minimizar. Recuerde la fórmula para el gradiente de la función f:

$$\nabla f(x) =$$

4. Sistema de ecuaciones normales para el problema de mínimos cuadrados. Si z es la solución del problema de mínimos cuadrados, entonces

$$A^\top A z = \underbrace{\qquad\qquad}_{?}.$$

5. La solución es la proyección ortogonal sobre el subespacio generado por las columnas de la matriz. Supongamos que z es la solución del problema. Entonces para cada $x \in \mathbb{R}^m$, el vector Ax es ortogonal al vector $Az - b$:

$$\big(\qquad\big)^\top \big(\qquad\qquad\big) = x^\top \qquad\qquad\qquad = x^\top \big(A^\top A z - A^\top b\big) = 0.$$

6. Factorización QR reducida. Sea $A \in \mathcal{M}_{n\times m}(\mathbb{R})$ con $\mathrm{r}(A) = m \leq n$. Una factorización QR reducida de A es un par de matrices (Q, R) tales que:

- $QR = \underbrace{\quad}_{?}$, Q es de tamaño $\quad\times\quad$, R es de tamaño $\quad\times\quad$.
- Las columnas de Q forman una lista ortonormal, esto es, $Q^\top Q =$.
- R es triangular $\underbrace{\qquad\qquad}_{?}$ e $\underbrace{\qquad\qquad}_{?}$,
  por eso sus entradas diagonales son $\underbrace{\qquad\qquad}_{?}$.

7. Mínimos cuadrados por medio de QR reducida. Muestre cómo hallar la solución z del problema de mínimos cuadrados usando la factorización QR de la matriz A. Empiece con el sistema de ecuaciones normales:

$$A^\top A z = \underbrace{\qquad}_{?} \iff (QR)^\top QRz = \underbrace{\qquad}_{?}$$

$$\iff \ldots$$

8. Programación: solución del problema de mínimos cuadrados por medio de QR reducida.

```
function [z] = leastsquares(A, b),
   [Q, R] = myqr(A);
   z =
end
```

9. Pruebas con datos pequeños.

```
function [] = testleastsquares(),
   n = 8; m = 4;
   A = 2 * rand(n, m) - ones(n, m);
   b = 2 * rand(n, 1) - ones(n, 1);
   z = leastsquares(A, b);
   fz = norm(A * z - b) ^ 2;
   normaleq = A' * A * z - ???;
   display(fz);
   display(normaleq);
   # construct some neighbors of z:
   nneighb = 5;
   X = repmat(z, 1, nneighb) + 0.1 * randn(m, nneighb);
   fX = norm(A * X - repmat(b, 1, nneighb), 'cols') .^ 2;
   display(fX);
   testorth = (A * X)' * (A * z - b);
   display(testorth);
end
```

10. Pruebas con datos de tamaños grandes. Modifique la función anterior de tal manera que n y m sean sus argumentos. Hay que medir el tiempo de ejecución de `leastsquares`. Además de `fz` y `normaleq`, mostrar el elemento mínimo de `fX` y la norma del vector `testorth`.