Tarea 3. Condicionamiento y estabilidad

Ricardo Chávez Cáliz

September 20, 2017

Problema 1. Sea A una matriz de tamaño 50×20, aleatoria y fija, calcular su descomposición QR. Sean $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_{20} \geq 0$ y

$$B = Q^* diag(\lambda_1, \lambda_2, ..., \lambda_{20})Q \quad \text{y} \quad B_\varepsilon = Q^* diag(\lambda_1 + \varepsilon_1, \lambda_2 + \varepsilon_2, ..., \lambda_{20} + \varepsilon_{20})Q,$$

con $\varepsilon_i \sim N(0, \sigma)$, con $\sigma = 0.01\lambda_1$.

1. Comparar la descomposición de Cholesky de B y de B_ε usando el algoritmo de la tarea 1. Considerar los casos cuando B tiene un *buen* número de condición y un *mal* número de condición.
2. Con el caso mal condicionado, comparar el resultado de su algoritmo con el del algoritmo de Cholesky de scipy.
3. Medir el tiempo de ejecución de su algoritmo de Cholesky con el de scipy.

1.1 El número de condición de una matriz A es $\kappa(A) = \|A\| \cdot \|A^{-1}\|$. Cuando $\|\cdot\| = \|\cdot\|_2$ entonces

$$\kappa(A) = \frac{\lambda_{max}}{\lambda_{min}}$$

En este caso los eigenvalores de B están dados por $\lambda_1 \dots \lambda_{20}$. Para controlar el condicionamiento de B se tomaron $\lambda_i = 1, \forall i \in \{2, \dots, 20\}$ y $\lambda_1 = f(n)$ donde $f(n)$ es una función creciente, con $f(n) > 1$. De esta manera para cada n tendremos B_n con

$$\kappa(B_n) = \lambda_{max} = \lambda_1 = f(n)$$

Cuando $\lambda_1 = 1$ entonces B estará bien condicionada.

Para comparar la descomposición de Choleski de B_n y $B_{n,\epsilon}$ se calculó $\|U_n - U_{n,\epsilon}\|$ para cada n, donde U_n proviene de la descomposición de Choleski de B_n y $f(n) = n$ (es decir, el valor de λ_1 crece de manera lineal). Se graficó λ_1 contra $\|U_n - U_{n,\epsilon}\|$ (triángulos) hasta n=5000, dejando para cada n $\|U_1 - U_{1,\epsilon}\|$, para así tener una comparativa del caso bien condicionado contra el mal condicionado. Además se graficó $log(\|U_n - U_{n,\epsilon}\|)$ respecto a λ_1. Ver Figuras 1 y 2, en las cuales se puede apreciar la diferencia en el caso mal condicionado y el bien condicionado.

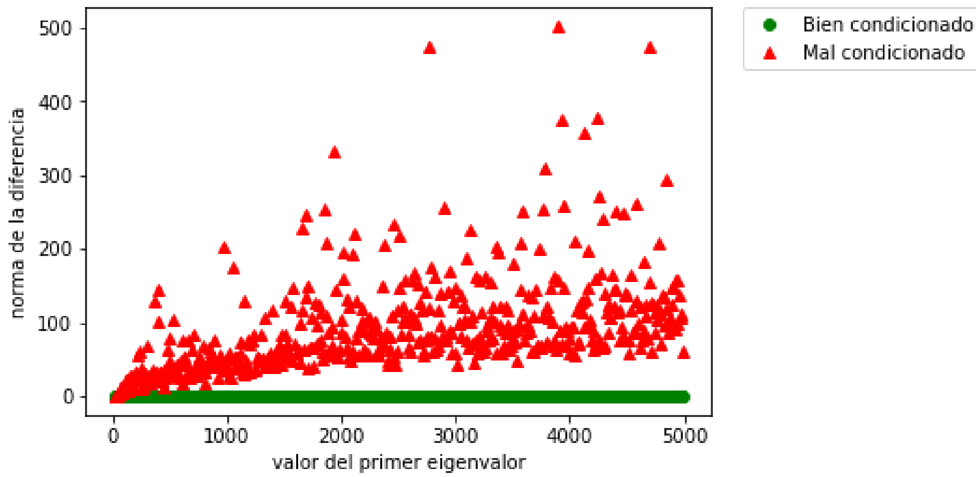

Figure 1: Gráfica de $\|U_n - U_{n,\epsilon}\|$ comparado con $\|U_1 - U_{1,\epsilon}\|$

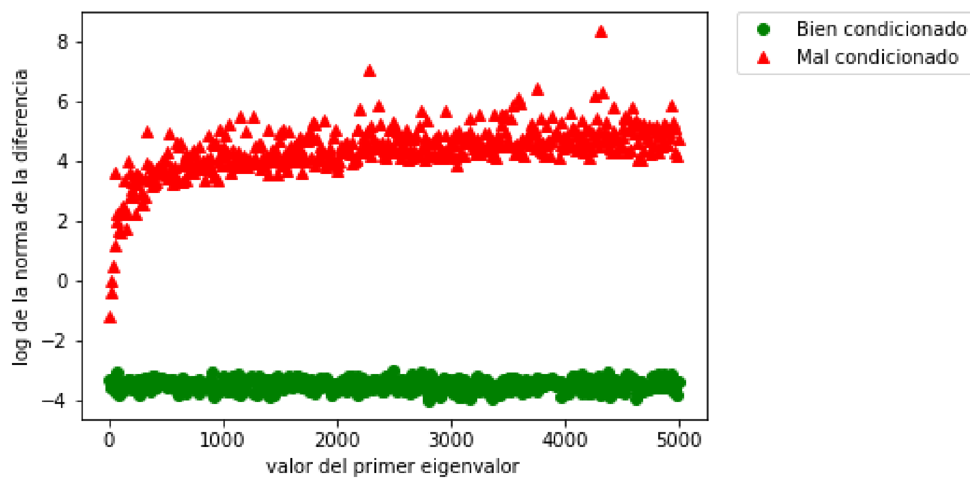

Figure 2: Gráfica de $log(\|U_n - U_{n,\epsilon}\|)$ comparado con $log(\|U_1 - U_{1,\epsilon}\|)$

Como $U_{n,\epsilon}$ está construido de manera aleatoria, las gráficas recién presentadas son diferentes en cada ejecución. Para conseguir gráficas *típicas* se ejecutó lo anterior 30 veces y se consideró el valor medio en cada $\|U_n - U_{n,\epsilon}\|$. Se tomó el valor medio para encontrar la tendencia central, por el sesgo presente en $\|U_n - U_{n,\epsilon}\|$. Se tomó hasta $n = 1000$ para reducir el tiempo de ejecución. Se obtuvieron las siguientes gráficas, que muestran la evidencia de antes, pero con menos ruido.

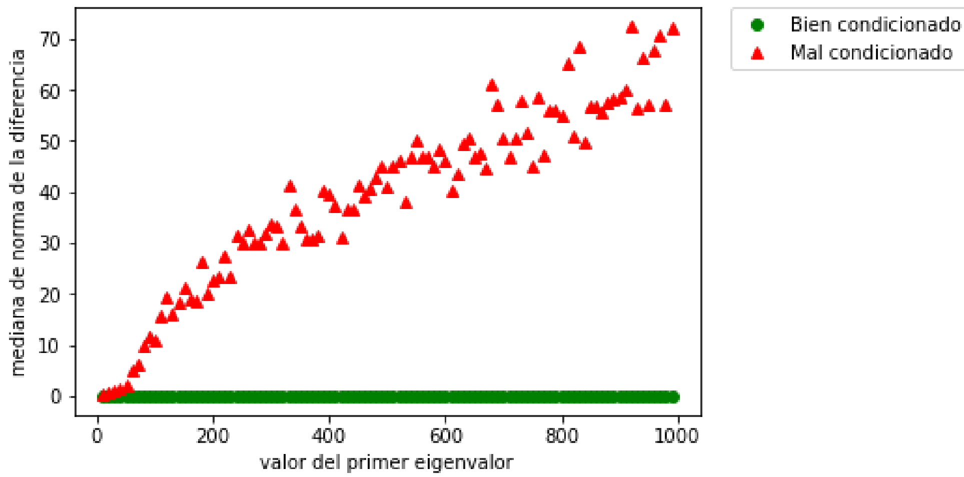

Figure 3: Gráfica del valor medio de $\|U_n - U_{n,\epsilon}\|$ comparado con $\|U_1 - U_{1,\epsilon}\|$

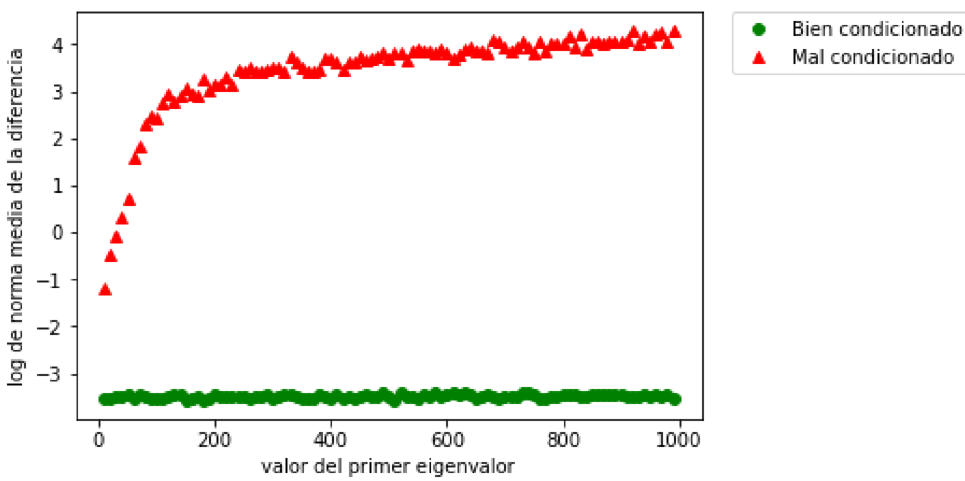

Figure 4: Gráfica del logaritmo del valor medio de $(\|U_n - U_{n,\epsilon}\|)$ comparado con $log(\|U_1 - U_{1,\epsilon}\|)$

Dado que la perturbación está dada en términos de λ_1, se pudiera pensar que el fenómeno reflejado en las gráficas anteriores es propio de dicho aumento en la perturbación de B. Para tomar en cuenta esto, consideremos los errores relativos. Además recuerde que el interés en comparar la descomposición de Choleski entre B y B_ϵ radica en estudiar el número de condición asociado al problema de obtener la descomposición de Cholesky de una matriz dada. Es decir, buscamos c en

$$c\frac{\|B - B_\epsilon\|}{\|B\|} = \frac{\|U - U_\epsilon\|}{\|U\|}$$

Por lo tanto para cada n se calculó el valor medio de

$$\frac{\|U - U_\epsilon\|}{\|U\|} \cdot \frac{\|B\|}{\|B - B_\epsilon\|}$$

en un experimento con 30 repeticiones y se obtuvo el siguiente resultado graficando como antes.

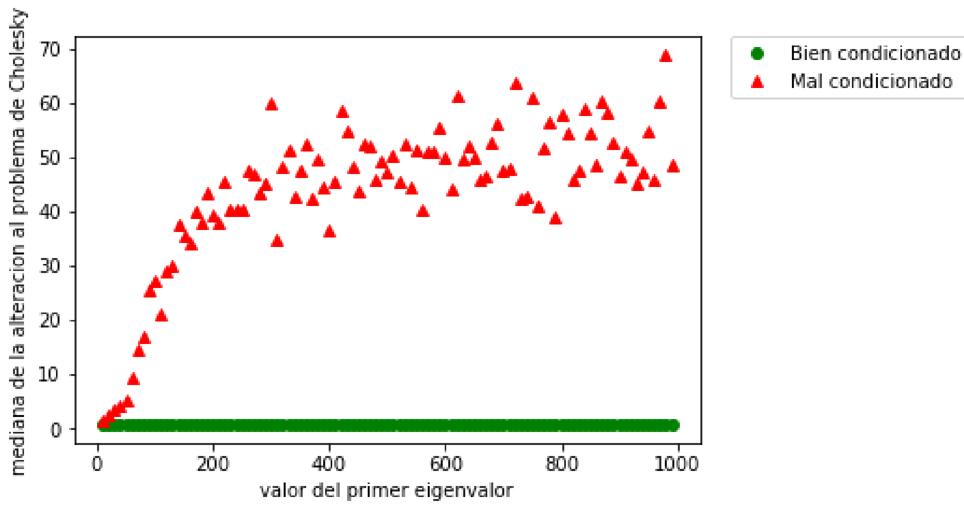

Figure 5: Gráfica del valor medio de $\frac{\|U_n - U_{n,\epsilon}\|}{\|U_n\|} \cdot \frac{\|B\|}{\|B - B_\epsilon\|}$ comparado con $\frac{\|U_1 - U_{1,\epsilon}\|}{\|U_1\|} \cdot \frac{\|B\|}{\|B - B_\epsilon\|}$

La gráfica anterior da evidencia para decir que el condicionamiento al problema de Cholesky está relacionado con el número de condición de la matriz a descomponer.

1.2 Al intentar usar el algoritmo de scipy, se obtiene un error: *(LinAlgError: n-th leading minor not positive definite)*. Lo cual quiere decir que para matrices mal condicionadas el algoritmo de scipy falla. Para verificar en qué punto falla se intentó decomponer B_n para $n \in \{0, 1, \ldots, 100\}$ y se verifica el primer n que genera un error. Para cada ejecución este número es diferente. Al realizar 100 pruebas el promedio fue: 41.54.

1.3 Para cada n entre 1 y 200 se generó una gráfica aleatoria de tamaño n, la cual se modificó para ser simétrica y definida positiva. Se midió el tiempo de ejecución de ambos algoritmos y graficando en escala logarítmica se obtuvo el siguiente resultado.

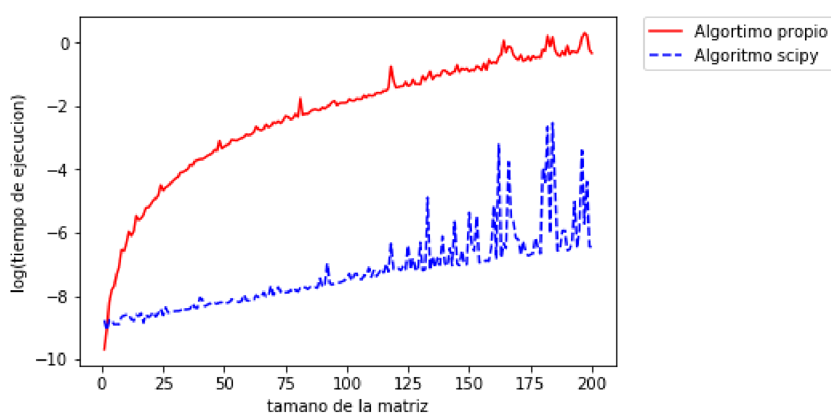

Figure 6: Diferencia en tiempos del algoritmo implementado en tarea 1 y el importado de scipy

En la figura se puede apreciar la eficiencia del algoritmo de scipy comparado con el implementado en la tarea 1. La "desventaja" de este es que no es capaz de realizar la descomposición para matrices mal condicionadas.

Problema 2. Resolver el problema de mínimos cuadrados,

$$y = X\beta + \varepsilon,\ \varepsilon_i \sim N(0,\sigma)$$

usando su implementación de la descomposición QR; β es de tamaño $d \times 1$ y X de tamaño $n \times d$.

Sean $d = 5, n = 20, \beta = (5,4,3,2,1)^t$ y $\sigma = 0.1$.

1. Hacer X con entradas aleatorias $U(0,1)$ y simular y. Encontrar $\hat{\beta}$ y compararlo con el obtenido $\hat{\beta}_p$ haciendo $X+\Delta X$, donde las entradas de ΔX son $N(0,\sigma = 0.01)$. Comparar a su vez con $\hat{\beta}_c = ((X+\Delta X)^t(X+\Delta X))^{-1}(X+\Delta X)y$ usando el algoritmo genérico para invertir matrices scipy.linalg.inv .

2. Lo mismo que el anterior pero con X mal condicionada (ie. con casi colinealidad).

2.1 Una manera de comparar $\widehat{\beta}$, $\widehat{\beta_p}$ y $\widehat{\beta_c}$ es calculando $\|\widehat{\beta} - \widehat{\beta_p}\|$ y $\|\widehat{\beta_p} - \widehat{\beta_c}\|$. Se realizó este cálculo con la norma usual. Para esto se tomaron 1000 repeticiones y se obtuvieron los siguientes promedios: de $\|\widehat{\beta} - \widehat{\beta_p}\|_2$ fue 0.234556096125, el cual representa un error de 0.0475338501589 respecto a la entrada más grande de β; de $\|\widehat{\beta_p} - \widehat{\beta_c}\|_\infty$ fue $2.15743201733e-14$ el cual representa un error de $4.31486403467e-15$ respecto a la entrada más grande de β. La comparación de $\widehat{\beta}$ y $\widehat{\beta_c}$ se deduce de una cota dada por desigualdad tringular. Dado que $\|\widehat{\beta_p} - \widehat{\beta_c}\|_\infty$ es en promedio $2.15743201733e-14$ (muy pequeña). Podemos reducir a analizar solamente $\widehat{\beta}$ y $\widehat{\beta_p}$

Pensando el problema de los mínimos cuadrados como una función $f : V \to S$ donde V es el espacio normado de los datos y S es el espacio normado de las soluciones, con $f(X) = \widehat{\beta}$ y $f(X+\delta X) = \widehat{\beta_p}$, la manera apropiada de comparar $\widehat{\beta}$ y $\widehat{\beta_p}$ será usando el número de condición asociado a f, tomando $\|\cdot\| = \|\cdot\|_2$ tenemos:

$$c_f = \frac{\|f(X) - f(X+\Delta X)\|}{\|f(X)\|} \cdot \frac{\|X\|}{\|X - (X+\Delta X)\|} = \frac{\|\widehat{\beta} - \widehat{\beta_p}\|}{\|\widehat{\beta}\|} \cdot \frac{\|X\|}{\|\Delta X\|}$$

Para esto, de igual manera se tomaron 1000 repeticiones y se obtuvo un c_f promedio de 1.83679946719. Es decir, que en promedio para una muestra de 1000 matrices con entradas como se establece, se tiene que el error relativo en el *output* no es más grande que el doble del error relativo en el *input*

2.2 Para el caso mal condicionado se tomaron 1000 repeticiones y se obtuvieron los siguientes promedios: de $\|\widehat{\beta} - \widehat{\beta_p}\|_2$ fue 9.91928629595, el cual representa un error de 1.98385725919 respecto a la entrada más grande de β; de $\|\widehat{\beta_p} - \widehat{\beta_c}\|_\infty$ fue $7.87736910659e-12$ el cual representa un error de $1.57547382132e-12$ respecto a la entrada más grande de β. Se obtuvo que el promedio de c_f fue 38.8253247212 (representa un cambio grande en el error relativo de entrada y el de salida).

En base a lo observado se puede decir que el condicionamiento de X influye en gran manera en el error relativo de output. y que en estos casos es mejor aproximar $\widehat{\beta_p}$ con $\widehat{\beta_c}$.