1) Resolver la ecuación no lineal: $e^{-x} - x = 0$ con una tolerancia de 10^{-8} con el método de Newton con un programa en el lenguaje de su elección. El programa debe realizar monitoreo de las iteraciones realizadas, utilizar 3 criterios de paro: $|x_{k+1} - x_k| > (1 + |x_{k+1}|)tol$, $|f(x_k)| > tol$ y máximo número de iteraciones, para esto el programa imprime una tabla de la forma: Iteración | norma de f | x_k | Error Relativo de x_k | norma de la derivada de f |.

Usa $x^{(0)} = 1$. $x^* = 0.567143290409784\textcolor{white}{}0$.

2) Considera la función de masa de probabilidad $p(x|\theta) = \frac{\theta^x}{x(-\log(1-\theta))}$ para $x = 1, 2, 3, \ldots$ $0 < \theta < 1$ (discrete logarithmic series distribution). Utiliza el método de Newton con $x^{(0)} = .9$ con una tolerancia de 10^{-8} con un programa en el lenguaje de su elección para calcular el estimador por máxima verosimilitud a la función log-verosimilitud $\ell(\theta|\underline{x})$ con $\underline{x}$ una muestra i.i.d. observada tal que $n = 10$, $\sum_{i=1}^{n} x_i = 15$, $\underline{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_{10} \end{pmatrix}$. El programa debe realizar monitoreo de las iteraciones realizadas, utilizar 2 criterios de paro: $|f'(\theta_k)| > tol$ y máximo número de iteraciones, para esto el programa imprime una tabla de la forma:

Iteración | norma de f' | θ_k | Error Relativo de θ_k | norma de la 2ª derivada de f |. $\theta^* = 0.533589233919995$.

3) Sea $A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & -1 & 4 \end{pmatrix}$. Define $f: \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ como $f(x, \lambda) = \begin{pmatrix} Ax - \lambda x \\ x^T x - 1 \end{pmatrix}$ y resuelve $f(x, \lambda) = 0$ con el método de Newton en más de una dimensión y una tolerancia de 10^{-8} con un programa en el lenguaje de su elección para aproximar $\left(1, \begin{pmatrix} -0.816496580927726 \\ 0.408248290463863 \\ 0.408248290463863 \end{pmatrix}\right)$ pareja eigenvalor y eigenvector de A.

Utiliza $x_N^{(0)} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$ (unitario) y $\lambda_0 = x_N^{(0)T} A x_N^{(0)}$ como puntos iniciales y monitorea con una tabla de la forma: Iteración | norma de f | Error Relativo de x_k | número de condición de la derivada de f | y criterios de paro: $\|x_{k+1}-x_k\| > (1+\|x_{k+1}\|)tol_1$, $\|f(x_k)\| > tol_2$ con $tol_1 = tol_2 =$ tolerancia y máximo número de iteraciones.

4) Ajustar 2 modelos por mínimos cuadrados y penalización lasso sin intercepto al conjunto de datos "Motor trend car road tests" cuyo nombre es mtcars en el paquete glmnet de R de la forma:

a) $y_{mpg} = \beta_{disp} x_{disp} + \beta_{drat} x_{drat}$ b) $y_{mpg} = \beta_{disp} x_{disp} + \beta_{hp} x_{hp} + \beta_{drat} x_{drat}$ donde: mpg

es variable que mide miles/galon, disp, hp, drat son variables que miden displacement, gross horse power y rear axle ratio respectivamente, utilizando método de Newton, tolerancia 10^{-8}, $\delta = 0.5$ en el lenguaje de su elección y realizando tabla de la forma: |Iteración |norma de f' | Error relativo de $\beta^{(k)}$|

y puntos iniciales: a) $\beta^{(0)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, b) $\beta^{(0)} = \begin{pmatrix} -1 \\ -1 \\ -1 \end{pmatrix}$. Solución de glmnet: a) $\beta^* = \begin{pmatrix} -0.01682177 \\ 6.59053287 \end{pmatrix}$

b) $\beta^* = \begin{pmatrix} -0.0009079199 \\ -0.0343832997 \\ 6.9774336908 \end{pmatrix}$. Criterios de paro $\|f'(\beta_k)\| > tol$ y máximo número de iteraciones.

↳ Para resolver este problema usamos el desarrollo siguiente:

Problema en glmnet: $\displaystyle\min_{(\beta_0,\beta)\in\mathbb{R}^{n+1}} \frac{1}{2N}\sum_{i=1}^{N}(y_i - \beta_0 - x_i^T\beta)^2 + \delta\|\beta\|_1$, $\delta > 0$

($\delta = \lambda\alpha$ en glmnet, $\alpha = 1$ para lasso)

Suponemos modelo sin intercepto:

$$\min_{\beta\in\mathbb{R}^n} \frac{1}{2N}\|A\beta - y\|_2^2 + \delta\|\beta\|_1, \quad A = \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} \in \mathbb{R}^{m\times n}, \quad \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_n \end{pmatrix}, \quad y = \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix}$$

función objetivo: $f(\beta) = \frac{1}{2N}(A\beta - y)^T(A\beta - y) + \delta\sum_{i=1}^{n}|\beta_i| = \frac{1}{2N}\left[\beta^TA^TA\beta - 2\beta^TA^Ty + y^Ty\right] + \delta\sum_{i=1}^{n}|\beta_i|$

Si $\beta_i \neq 0 \ \forall i$: $\nabla f(\beta) = \frac{1}{N}(A^TA\beta - A^Ty) + \delta\tilde{\beta}$ con $\tilde{\beta} = \begin{pmatrix} \text{signo}(\beta_1) \\ \vdots \\ \text{signo}(\beta_n) \end{pmatrix}$, $\nabla^2 f(\beta) = \frac{1}{N}A^TA$

Si $A=QR$ entonces $\nabla f(\beta)=\frac{1}{N}\left(R^TR\beta-R^TQ^Ty\right)+\delta\tilde{\beta}$, $\nabla^2 f(\beta)=\frac{1}{N}R^TR$

Mét. Newton: β_0 punto inicial entonces: $\beta_{k+1}=\beta_k+s_k$

donde: s_k solución de: $\nabla^2 f(\beta_k)s_k=-\nabla f(\beta_k)$, esto es:

$$\frac{1}{N}R^TR\,s_k=-\left(\frac{1}{N}\left(R^TR\beta-R^TQ^Ty\right)+\delta\tilde{\beta}\right)$$

$$\text{entonces: } R^TRs_k=-\left(R^T(R\beta-Q^Ty)+N\delta\tilde{\beta}\right)$$

$$=-R^T\left(R\beta-Q^Ty+N\delta R^{-T}\tilde{\beta}\right)$$

$$=-R^T\left(R\beta-Q^Ty+N\delta\tilde{\tilde{\beta}}\right)$$

$\hookrightarrow \tilde{\tilde{\beta}}$ solución de $R^T\tilde{\tilde{\beta}}=\tilde{\beta}$

$\therefore$ resolver sistema: $Rs_k=-\left(R\beta-Q^Ty+\delta\tilde{\tilde{\beta}}\right)\to$ válido si $\beta_i\neq 0\ \forall i$

y hacer actualización: $\beta_{k+1}=\beta_k+s_k$

5) Realizar un programa en el lenguaje de su elección para utilizar 3 métodos de descenso: gradiente, por coordenadas y dirección de Newton. Todos deben utilizar backtracking para cortar el paso. Tolerancia: 10^{-8} y el programa imprime una tabla de la forma: | Iteración | Norma de f' | Error relativo x^* | Error relativo p^* | $t_{\text{backtracking}}$ |

Para resolver: a) $\min\limits_{x}\frac{1}{2}(x_1^2+\gamma x_2^2)$, $\gamma=10$, punto inicial $x^{(0)}=\begin{pmatrix}1\\1\end{pmatrix}$, $x^*=\begin{pmatrix}0\\0\end{pmatrix}$

b) $\min\limits_{x} e^{x_1+3x_2-0.1}+e^{x_1-3x_2-0.1}+e^{-x_1-0.1}$, punto inicial $x^{(0)}=\begin{pmatrix}-1\\-1\end{pmatrix}$, $x^*=\begin{pmatrix}-.34654\\-.0000076725\end{pmatrix}$

Criterios de paro $\|f'(x_k)\|>tol$ y máximo número de iteraciones.