- Scrivere nella forma $F(v) = 0$ il sistema di equazioni non-lineari

$$
\begin{aligned}
2x^2 + y + z &= 1. \\
x + 3y^2 + z &= 1, \\
x + y + 4z^2 &= 1,
\end{aligned}
$$

  con $v = [x, y, z]^T \in \mathbb{R}^3$ and $F : \mathbb{R}^3 \to \mathbb{R}^3$, e calcolarne lo Jacobiano J_F.
- Scrivere "function handles" F = @(v) [2*v(1)^2 + ...] e JF = @(v) [4*v(1) ...] che calcolano F e JF su un vettore v in input.
- Per controllare che lo Jacobiano sia corretto, generare vettori casuali v = randn(3,1), h = 1e−5 * randn(3,1) e controllare che $F(v+h) - F(v) - JF(v)h = O(\|h\|^2)$.
- Scrivere una function x1 = multinewton_step(F, JF, x0) che esegue un passo del metodo di Newton multivariato. Testarla sul sistema precedente con $x_0 = [1, 1, 1]^T$, che dovrebbe produrre $x_1 = [\frac{1}{2}, \frac{1}{2}, \frac{1}{2}]^T$.
- Scrivere una function x = multinewton(F, JF, x0, m) che esegue m passi.
- Analizzare la convergenza come nella lezione scorsa, modificando la funzione in function [x, e] = multinewton_error(F, JF, x0, m, z), per restituire il vettore degli errori $e_i = \|x^{(k)} - z\|_\infty$, $i = 1, 2, \dots, m$.
- Scrivere una function x = multicorde(F, JFx0, x0, m) che esegue m passi del metodo delle corde, calcolando una volta sola una fattorizzazione della matrice JFx0. L'argomento JFx0 dev'essere una matrice, non una handle!