Pour traiter ce sujet le candidat est vivement invité à utiliser l'ordinateur à sa disposition, équipé de Python/Pyzo et de Scilab.

1.

a. Importer la fonction `fsolve` du sous-module `optimize` du module `scipy`, puis entrer le code suivant et l'expliquer.

```
def f(x):
    return [2*x[0]**2 + 3*x[1] - 11, 3*x[0] - 2*x[0]*x[1] - 2]
sol1 = fsolve(f, [0,0])
sol2 = fsolve(f, [1,1])
sol3 = fsolve(f, [2,1])
print(sol1, sol2, sol3)
```

b. Dans cette question, on considère la matrice $A_1 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$.

Déterminer une matrice S_1 symétrique réelle à valeurs propres positives et une matrice orthogonale U_1 telle que $A_1 = U_1\, S_1$.

Si les résultats obtenus sont des flottants on pourra les multiplier par la racine carrée d'un nombre premier inférieur à 10 pour obtenir les valeurs exactes.

2. Soit n un entier au moins égal à 2 et A de $\mathcal{M}_n(\mathbb{R})$ inversible.

a. Montrer qu'il existe un couple (U, S) où U est orthogonale et S est symétrique réelle à valeurs propres strictement positives telles que $A = U\,S$.

> On pourra commencer par établir l'existence d'une matrice P orthogonale et d'une matrice D diagonale à coefficients strictement positifs telles que ${}^tP({}^tA\,A)\,P = D^2$.

b. En déduire que pour toute A de $GL_n(\mathbb{R})$ il existe deux matrices orthogonales V et W et une matrice diagonale D telle que $V A W = D$.

c. Donner de telles matrices (V_1, W_1 et D_1) pour la matrice A_1 précisée ci-dessus.

Pour chacune de ces matrices on donnera si possible les valeurs exactes et des valeurs décimales approchées raisonnables des coefficients.