

Exercício 2. Uma aplicação de transformações lineares é em processamento de imagens, rodando ou deformando imagens.

Lembre que o comando `plot(x,y)` do Matlab desenha segmentos de reta ligando os pontos (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , etc. Use o comando `plot` para desenhar um triângulo equilátero com um dos vértices no ponto $(1,0)$ e o baricentro na origem. Dica: use `axis equal` para que as escalas horizontal e vertical fiquem iguais, e o triângulo pareça mesmo equilátero).

Exercício 3. Vamos começar a deformar o triângulo. Por exemplo, para refletir o triângulo ao longo do eixo y , você pode transformar as coordenadas de cada vértice pela transformação

$$T_1 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Para esticar o triângulo ao longo do eixo x e encolher ao longo do eixo y , você pode multiplicar as coordenadas dos vértices por

$$T_2 = \begin{bmatrix} 2 & 0 \\ 0 & 0,5 \end{bmatrix}.$$

Por fim, se você quiser rodar o triângulo por um ângulo θ , você pode usar

$$T_3 = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}.$$

Mostre que isto é verdade. Observe que $T_3^{-1} = T_3^T$ — matrizes que obedecem a essa propriedade são especiais, chamadas de *matrizes ortogonais*.

Exercício 4. Experimente criar um triângulo isósceles com base de comprimento 1, girado de 45° com relação ao eixo x .

Exercício 5. Vamos tentar algo mais complicado agora. Você consegue imaginar uma sequência de transformações para refletir o triângulo em torno da reta $x = 2y$?

Exercício 6. Agora ache uma sequência de transformações que deforme o triângulo, expandindo pontos na direção $x = y$ por 2 e contraindo pontos na direção $x = -y$ pela metade.

Exercício 7. Rode agora o triângulo por 45° , mas em torno do ponto $(1,0)$.

Exercício 8. Carregue agora o arquivo `exp03.mat`, e dê os comandos `plot(x,y, 'r')`, `axis equal`. Rode a figura de -30° em torno do ponto $(0,0)$.

Exercício 9. Expanda a figura por 50% na direção $x = 2y$ e contraia por 50% na direção $y = -2x$.

Exercício 10. Como você poderia expandir a figura na direção $x = 2y$, e contrair na direção $x = -y$? Alguma matriz de transformação que você deve usar neste exercício é ortogonal?

Exercício 11. Outra aplicação interessante de transformações lineares é filtragem de sinais. Imagine que você tenha um vetor com amostras de um sinal, por exemplo, o sinal $\text{sen}(2\pi ft)$ pode ser amostrado a uma taxa de uma amostra a cada Δt segundos:

$$s(n) = \text{sen}(2\pi fn\Delta t), \quad n = 1 \dots 1000.$$

Suponha no entanto que o sinal esteja misturado com ruído:

$$y = s + r,$$

em que $r = 0,1 * \text{randn}(1, 200)$. Desenhe um gráfico de y em função de t , para uma frequência $f = 100$ Hz e $\Delta t = 0,1$ ms.

Exercício 12. Para reduzir o ruído sem modificar o sinal, podemos aplicar um *filtro*, por exemplo, trocando o sinal em cada instante pela média do sinal nos instantes próximos, assim:

$$s_f(n) = \frac{s(n) + s(n-1) + s(n-2)}{3}.$$

Você pode pensar nisso como a aplicação da transformação linear

$$T_f = \frac{1}{3} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \end{bmatrix}$$

ao vetor s . Obviamente, a matriz fica muito grande. Montar a matriz T_f e fazer a multiplicação $T_f s$ não é a maneira mais inteligente de aplicar o filtro.

Repare que, excetuando as duas primeiras e as duas últimas linhas da matriz, todas as linhas são iguais, apenas deslocadas. Escreva um programa que tenha como entrada um vetor h com uma linha da matriz, a outra entrada o vetor s , e que forneça como saída o vetor $s_f = T_f s$. Procure fazer o cálculo sem criar a matriz explicitamente.

Teste o seu programa usando `h=0.1*ones(1,3); sf=filter(h,1,s);`. Compare o sinal de saída com o de entrada. O que aconteceu com o ruído? O que acontece se você aumenta ou diminui o número de pontos no vetor h ?