GEOPH 531

Computing Vertical and Horizontal derivatives operators via Kronecker products

Mauricio D Sacchi

Winter 2023

Abstract

This essay is to help students to solve Assignment 2 of GEOPH 531. I will show a simple trick that uses Kronecker products to compute the derivative operators $\widetilde{\mathbf{D}}_x$ and $\widetilde{\mathbf{D}}_z$.

1 Definitions

- **Kronecker Product:** If $\mathbf{A}$ is an $m \times n$ matrix and $\mathbf{B}$ is a $p \times q$ matrix, then the Kronecker product is the matrix $\mathbf{C}$ of size $pm \times qn$ given by

$$\mathbf{C} = \mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \dots & a_{1n}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \dots & a_{2n}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & a_{n2}\mathbf{B} & \dots & a_{mn}\mathbf{B} \end{pmatrix} . \tag{1}$$

- **Vectorization of the product of 3 matrices:** If $\mathbf{A}$, $\mathbf{X}$, $\mathbf{B}$ and $\mathbf{C}$ are matrices such that

$$\mathbf{AXB} = \mathbf{C}\,, \tag{2}$$

then

$$(\mathbf{B}^T \otimes \mathbf{A})\,\mathtt{vec}(\mathbf{X}) = \mathtt{vec}(\mathbf{C}) \longrightarrow (\mathbf{B}^T \otimes \mathbf{A})\mathbf{x} = \mathbf{c}\,, \tag{3}$$

where $\mathbf{x} = \mathtt{vec}(\mathbf{X})$ denotes the vectorization of the matrix $\mathbf{X}$, formed by stacking the columns of $\mathbf{X}$ into a single column vector. Similarly, $\mathbf{c} = \mathtt{vec}(\mathbf{C})$ denotes the vectorization of the matrix $\mathbf{C}$, formed by stacking the columns of $\mathbf{C}$ into a single column vector.

2 Vertical derivative operator

Consider a model, for instance, a slowness model in x-z, which was discretized and expressed as a matrix where each element of the matrix is a cell that represents a value of slowness. For instance, if the model is discretized via $nx = 4$ and $nz = 5$ cells

$$\mathbf{M} = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} \end{pmatrix} . \tag{4}$$

Applying the vertical derivative operator to $\mathbf{M}$ is equivalent to the following

$$\mathbf{V} = \mathbf{D}_z\,\mathbf{M} = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} \end{pmatrix} , \tag{5}$$

where

$\mathbf{V}$ is $nz - 1 \times nx$

$\mathbf{D}_z$ is $nz - 1 \times nz$

$\mathbf{M}$ is $nz \times nx$.

We can write equation 5 as

$$\mathbf{V} = \mathbf{D}_z\,\mathbf{M}\,\mathbf{I}_{nx} , \tag{6}$$

where $\mathbf{I}_{nx}$ is the identity matrix of size $nx \times nx$. We can now apply the property given by equations 2-3

$$\mathtt{vec}(\mathbf{V}) = (\mathbf{I}_{nx} \otimes \mathbf{D}_z)\mathtt{vec}(\mathbf{M}) . \tag{7}$$

Call $\mathbf{v} = \mathtt{vec}(\mathbf{V})$ and $\mathbf{m} = \mathtt{vec}(\mathbf{M})$ and $\widetilde{\mathbf{D}}_z = \mathbf{I}_{nx} \otimes \mathbf{D}_z$, then

$$\mathbf{v} = \widetilde{\mathbf{D}}_z\mathbf{m} . \tag{8}$$

3 Horizontal derivative operator

Consider the model again in x-z as in the example given by equation 4. Applying the horizontal derivative operator to $\mathbf{M}$ is equivalent to the following

$$\mathbf{H} = \mathbf{M}\mathbf{D}_x \; = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} \end{pmatrix} \begin{pmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} , \tag{9}$$

where

$\mathbf{H}$ is $nz \times nx - 1$

$\mathbf{M}$ is $nz \times nx$

$\mathbf{D}_x$ is $nx \times nx - 1$.

Notice that I needed to operate on rows rather than columns, so I applied the difference operator to the right of $\mathbf{M}$. We can write equation 9 as

$$\mathbf{H} = \mathbf{I}_{nz}\, \mathbf{M}\, \mathbf{D}_x \, . \tag{10}$$

From the above and using expressions 2-3 we have

$$\texttt{vec}(\mathbf{H}) = (\mathbf{D}_x^T \otimes \mathbf{I}_{nz})\texttt{vec}(\mathbf{M}) \, . \tag{11}$$

Call $\mathbf{h} = \texttt{vec}(\mathbf{H})$, $\mathbf{m} = \texttt{vec}(\mathbf{M})$ and $\widetilde{\mathbf{D}}_x = \mathbf{D}_x^T \otimes \mathbf{I}_{nz}$ then,

$$\mathbf{h} = \widetilde{\mathbf{D}}_x \mathbf{m} \, .$$

4 Putting it all together

$\widetilde{\mathbf{D}}_x$ and $\widetilde{\mathbf{D}}_z$ are the matrices of derivatives acting on the vectorized model $\mathbf{m} = \texttt{vec}(\mathbf{M})$. It would help if you had these matrices in Assignment 2. The latter is not the only way to compute these matrices. It is ok if you are using a different method.

5 Julia code - not optimized for sparse matrix computations

```
# Show how one can make vertical and horizontal derivative operators
# acting on a vectorized model m = vec(M)

nx = 40
nz = 15

M = 0.001*ones(Float64,nz,nx)
m = reshape(M,nz*nx)

Inx = diagm(ones(Float64,nx))
Inz = diagm(ones(Float64,nz))

 Dz = diagm(0 => ones(nz), 1 => -ones(nz-1))
 Dz = Dz[1:nz-1,:]
 Dtz = kron(Inx,Dz)        # Tilde Dz in my notes
 v = Dtz*m

 Dx = diagm(0 => ones(nx), 1 => -ones(nx-1))
 Dx = Dx[:,2:nx]
 Dtx = kron(Dx',Inz)       # Tilde Dx in my notes
 h = Dtx*m

# Output v and h should be zero because M is constant.

if sum(h.^2)<10.e-16; println("Dtx is ok"); end
if sum(v.^2)<10.e-16; println("Dtz is ok"); end
```