ch1_phase2_MDEXEC

```python
[1]: from sympy import Matrix, Symbol, derive_by_array, Lambda, Function, MatrixSymbol, Derivative, diff, symbols
     from sympy import var
     from sympy.abc import x, i, j, a, b


     from sympy.interactive import init_printing

     init_printing(pretty_print=True, wrap_line=True, num_columns=60)
```

```python
[2]: def myvar(letter: str, i: int, j: int) -> Symbol:
         letter_ij = Symbol('{}_{}{}'.format(letter, i+1, j+1), is_commutative=True)
         return letter_ij


     n,m,p = 3,3,2

     X = Matrix(n, m, lambda i,j : myvar('x', i, j)); X
```

$$[2]: \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}$$

```python
[3]: W = Matrix(m, p, lambda i,j : myvar('w', i, j)); W
```

$$[3]: \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}$$

```python
[4]: A = MatrixSymbol('X',3,3); Matrix(A)
     B = MatrixSymbol('W',3,2)
```

Defining $N = \nu(X, W) = X \times W$

- $\nu : \mathbb{R}^{(n \times m) \times (m \times p)} \to \mathbb{R}^{n \times p}$
- $N \in \mathbb{R}^{n \times p}$

```python
[5]: v = lambda a,b: a*b

     vL = Lambda((a,b), a*b)
```

```python
n = Function('v') #, Lambda((a,b), a*b))

vN = lambda mat1, mat2: Matrix(mat1.shape[0], mat2.shape[1], lambda i, j: Symbol("n_{}{}".format(i+1, j+1))); vN

Nelem = vN(X, W); Nelem
```

[5]:
$$\begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \\ n_{31} & n_{32} \end{bmatrix}$$

[6]: `n(X,W)`

[6]:
$$v\left(\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}, \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}\right)$$

[7]: `n(A,B)`

[7]: $v(X,W)$

[8]: `n(X,W).replace(n, v) # replace works when v = python lambda`

[8]:
$$\begin{bmatrix} w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13} & w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13} \\ w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23} & w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23} \\ w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33} & w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33} \end{bmatrix}$$

[9]: `n(X,W).subs({n: vL}) # subs works when v = sympy lambda`

[9]:
$$\begin{bmatrix} w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13} & w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13} \\ w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23} & w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23} \\ w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33} & w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33} \end{bmatrix}$$

[10]: `n(X,W).replace(n, vL)`

[10]:
$$\begin{bmatrix} w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13} & w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13} \\ w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23} & w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23} \\ w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33} & w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33} \end{bmatrix}$$

[11]: `n(X,W).subs({n: v})# subs() doesn't work when v is python lambda`

[11]:
$$v\left(\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}, \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}\right)$$

[12]: `Matrix(n(A,B).subs({n: vL}))`

[12]:
$$\begin{bmatrix} W_{0,0}X_{0,0} + W_{1,0}X_{0,1} + W_{2,0}X_{0,2} & W_{0,1}X_{0,0} + W_{1,1}X_{0,1} + W_{2,1}X_{0,2} \\ W_{0,0}X_{1,0} + W_{1,0}X_{1,1} + W_{2,0}X_{1,2} & W_{0,1}X_{1,0} + W_{1,1}X_{1,1} + W_{2,1}X_{1,2} \\ W_{0,0}X_{2,0} + W_{1,0}X_{2,1} + W_{2,0}X_{2,2} & W_{0,1}X_{2,0} + W_{1,1}X_{2,1} + W_{2,1}X_{2,2} \end{bmatrix}$$

```python
[13]: #N = v(X, W); N
      N = n(A,B); N
```

$[13]$: $v(X, W)$

```python
[14]: N.replace(n, v)
```

$[14]$: XW

```python
[15]: N.replace(n, v).subs({A: X, B:W}) # replacing ariable values after doing function doesn't make the function apply directly on the values (matrices), need to␣
      ↪replace values before the function is replaced, so that the function can act on them while they are given/alive.
```

$[15]$:
$$\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}$$

```python
[16]: N.subs({n: vL, A:X, B:W})
```

$[16]$:
$$\begin{bmatrix} w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13} & w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13} \\ w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23} & w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23} \\ w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33} & w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33} \end{bmatrix}$$

```python
[17]: Nspec = N.subs({A:X, B:W}).replace(n, v); Nspec
```

$[17]$:
$$\begin{bmatrix} w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13} & w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13} \\ w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23} & w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23} \\ w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33} & w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33} \end{bmatrix}$$

```python
[18]:
```

```python
[18]: N.diff(N)
```

$[18]$: 1

```python
[19]: N.diff(X)
```

$[19]$:
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

```python
[20]:
```

```python
[20]:
```

```python
[20]: # way 2 of declaring S (better way)
      sigma = Function('sigma')

      sigmaApply = Function("sigma_apply") #lambda matrix:  matrix.applyfunc(sigma)

      sigmaApply_ = lambda matrix: matrix.applyfunc(sigma)
```

```
sigmaApply(A)
```

[20]: $\sigma_{apply}(X)$

[21]:
```
sigmaApply(A).subs({A: X})
```

[21]:
$$\sigma_{apply}\left(\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}\right)$$

[22]:
```
sigmaApply_(A)
```

[22]: $(d \mapsto \sigma(d))_\circ (X)$

[23]:
```
sigmaApply(A).subs({A: X}).replace(sigmaApply, sigmaApply_) # NOTE: subs of functions doesn't work, replace actually evaluates the replaced function!
```

[23]:
$$\begin{bmatrix} \sigma(x_{11}) & \sigma(x_{12}) & \sigma(x_{13}) \\ \sigma(x_{21}) & \sigma(x_{22}) & \sigma(x_{23}) \\ \sigma(x_{31}) & \sigma(x_{32}) & \sigma(x_{33}) \end{bmatrix}$$

[24]:
```
S = sigmaApply(N); S
```

[24]: $\sigma_{apply}(v(X, W))$

[25]:
```
Derivative(S, S)
```

[25]: $\dfrac{\partial}{\partial \sigma_{apply}(v(X,W))} \sigma_{apply}(v(X,W))$

[26]:
```
Derivative(S, S).doit()
```

[26]: 1

[27]:
```
Derivative(S, n(A,B)).doit()
```

[27]: $\dfrac{\partial}{\partial v(X,W)} \sigma_{apply}(v(X,W))$

[28]:
```
#lambd = Function("lambda")
#Lagain = lambd(sigmaApply(n(A))); Lagain


# diff(Lagain, A) # never execute
#
```

[29]:
```
S.replace(A,X).replace(B,W)
```

[29]:

4

$$\sigma_{apply}\left(v\left(\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}, \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}\right)\right)$$

```
[30]: S.replace(n, v)
```

[30]:

$$\sigma_{apply}(XW)$$

```
[31]: S.subs({A:X, B:W}).replace(n, v)
```

[31]:

$$\sigma_{apply}\left(\begin{bmatrix} w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13} & w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13} \\ w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23} & w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23} \\ w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33} & w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33} \end{bmatrix}\right)$$

```
[32]: Sspec = S.subs({A:X, B:W}).replace(n, v).replace(sigmaApply, sigmaApply_)
      Sspec
```

[32]:

$$\begin{bmatrix} \sigma(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}) & \sigma(w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13}) \\ \sigma(w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23}) & \sigma(w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23}) \\ \sigma(w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33}) & \sigma(w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33}) \end{bmatrix}$$

```
[33]: S.replace(n, vN) #.replace(sigmaApply, sigmaApply_)
```

[33]:

$$\sigma_{apply}\left(\begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \\ n_{31} & n_{32} \end{bmatrix}\right)$$

```
[34]: Selem = S.replace(n, vN).replace(sigmaApply, sigmaApply_); Selem
```

[34]:

$$\begin{bmatrix} \sigma(n_{11}) & \sigma(n_{12}) \\ \sigma(n_{21}) & \sigma(n_{22}) \\ \sigma(n_{31}) & \sigma(n_{32}) \end{bmatrix}$$

```
[35]: import itertools

      elemToSpecD = dict(itertools.chain(*[[(Nelem[i, j], Nspec[i, j]) for j in range(2)] for i in range(3)]))

      elemToSpec = list(elemToSpecD.items())

      Matrix(elemToSpec)
```

[35]:

$$\begin{bmatrix} n_{11} & w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13} \\ n_{12} & w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13} \\ n_{21} & w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23} \\ n_{22} & w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23} \\ n_{31} & w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33} \\ n_{32} & w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33} \end{bmatrix}$$

```
[36]: elemToSpecFuncD = dict(itertools.chain(*[[(Nelem[i, j], Function("n_{}{}".format(i + 1, j + 1))(Nspec[i, j])) for j in range(2)] for i in range(3)]))
```

```python
elemToSpecFunc = list(elemToSpecFuncD.items())

Matrix(elemToSpecFunc)
```

[36]:
$$\begin{bmatrix} n_{11} & \mathbf{n}_{11} \left(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13} \right) \\ n_{12} & \mathbf{n}_{12} \left(w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13} \right) \\ n_{21} & \mathbf{n}_{21} \left(w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23} \right) \\ n_{22} & \mathbf{n}_{22} \left(w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23} \right) \\ n_{31} & \mathbf{n}_{31} \left(w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33} \right) \\ n_{32} & \mathbf{n}_{32} \left(w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33} \right) \end{bmatrix}$$

[37]:
```python
elemToSpecFuncArgsD = dict(itertools.chain(*[[(Nelem[i, j], Function("n_{}{}".format(i + 1, j + 1))(*X,*W)) for j in range(2)] for i in range(3)]))

elemToSpecFuncArgs = list(elemToSpecFuncArgsD.items())

Matrix(elemToSpecFuncArgs)
```

[37]:
$$\begin{bmatrix} n_{11} & \mathbf{n}_{11} \left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32} \right) \\ n_{12} & \mathbf{n}_{12} \left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32} \right) \\ n_{21} & \mathbf{n}_{21} \left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32} \right) \\ n_{22} & \mathbf{n}_{22} \left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32} \right) \\ n_{31} & \mathbf{n}_{31} \left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32} \right) \\ n_{32} & \mathbf{n}_{32} \left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32} \right) \end{bmatrix}$$

[38]:
```python
Selem
```

[38]:
$$\begin{bmatrix} \sigma(n_{11}) & \sigma(n_{12}) \\ \sigma(n_{21}) & \sigma(n_{22}) \\ \sigma(n_{31}) & \sigma(n_{32}) \end{bmatrix}$$

[39]:
```python
Selem.subs(elemToSpecD)
```

[39]:
$$\begin{bmatrix} \sigma(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}) & \sigma(w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13}) \\ \sigma(w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23}) & \sigma(w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23}) \\ \sigma(w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33}) & \sigma(w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33}) \end{bmatrix}$$

[40]:
```python
Selem[0,1].diff(Nelem[0,1])
```

[40]:
$$\frac{d}{dn_{12}}\sigma(n_{12})$$

[41]:
```python
Selem[0,1].diff(Nelem[0,1]).subs({Nelem[0,1] : Nspec[0,1]})
#Selem[0,1].diff(Nelem[0,1]).subs(dict([{Nelem[0,1] : Nspec[0,1]}]))
```

[41]:
$$\left. \frac{d}{dn_{12}}\sigma(n_{12}) \right|_{n_{12}=w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13}}$$

[42]:
```python
Selem[0,1].diff(Nelem[0,1]).subs({Nelem[0,1] : Nspec[0,1]}).subs({Nspec[0,1] : 23})
```

[42]:

$$\left.\frac{d}{dn_{12}}\sigma(n_{12})\right|_{n_{12}=23}$$

```
[43]: Selem[0,1].diff(Nelem[0,1]).subs({Nelem[0,1] : Nspec[0,1]}).replace(sigma, lambda x: 8*x**3)
```

[43]:
$$\left.\frac{d}{dn_{12}}8n_{12}^3\right|_{n_{12}=w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13}}$$

```
[44]: Selem[0,1].diff(Nelem[0,1]).replace(sigma, lambda x: 8*x**3)
```

[44]:
$$\frac{d}{dn_{12}}8n_{12}^3$$

```
[45]: Selem[0,1].diff(Nelem[0,1]).replace(sigma, lambda x: 8*x**3).doit()
```

[45]:
$$24n_{12}^2$$

```
[46]: # ### GOT IT: can replace now with expression and do derivative with respect to that expression.
      Selem[0,1].diff(Nelem[0,1]).subs({Nelem[0,1] : Nspec[0,1]}).replace(sigma, lambda x: 8*x**3).doit()
```

[46]:
$$24\left(w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13}\right)^2$$

```
[47]: Selem[0,1].subs({Nelem[0,1] : Nspec[0,1]}).diff(X[0,1])#.subs({Nelem[0,1] : Nspec[0,1]})
```

[47]:
$$\left.w_{22}\frac{d}{d\xi_1}\sigma(\xi_1)\right|_{\xi_1=w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13}}$$

```
[48]: Selem
```

[48]:
$$\begin{bmatrix}\sigma(n_{11}) & \sigma(n_{12}) \\ \sigma(n_{21}) & \sigma(n_{22}) \\ \sigma(n_{31}) & \sigma(n_{32})\end{bmatrix}$$

```
[49]: nt = Nelem.subs(elemToSpecFunc); nt
```

[49]:
$$\begin{bmatrix}\mathbf{n}_{11}\left(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13}\right) & \mathbf{n}_{12}\left(w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13}\right) \\ \mathbf{n}_{21}\left(w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23}\right) & \mathbf{n}_{22}\left(w_{12}x_{21}+w_{22}x_{22}+w_{32}x_{23}\right) \\ \mathbf{n}_{31}\left(w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33}\right) & \mathbf{n}_{32}\left(w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33}\right)\end{bmatrix}$$

```
[50]: st = Selem.subs(elemToSpecFunc); st
```

[50]:
$$\begin{bmatrix}\sigma(\mathbf{n}_{11}\left(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13}\right)) & \sigma(\mathbf{n}_{12}\left(w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13}\right)) \\ \sigma(\mathbf{n}_{21}\left(w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23}\right)) & \sigma(\mathbf{n}_{22}\left(w_{12}x_{21}+w_{22}x_{22}+w_{32}x_{23}\right)) \\ \sigma(\mathbf{n}_{31}\left(w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33}\right)) & \sigma(\mathbf{n}_{32}\left(w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33}\right))\end{bmatrix}$$

```
[51]: init_printing(num_columns=10, wrap_line=True, pretty_print=True)#



      st.diff(nt)
```

[51]:

$$\begin{bmatrix} \begin{bmatrix} \frac{\partial}{\partial \mathbf{n}_{11}(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13})}\sigma(\mathbf{n}_{11}(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13})) & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & \frac{\partial}{\partial \mathbf{n}_{12}(w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13})}\sigma(\mathbf{n}_{12}(w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13})) \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ \frac{\partial}{\partial \mathbf{n}_{21}(w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23})}\sigma(\mathbf{n}_{21}(w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23})) & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & \frac{\partial}{\partial \mathbf{n}_{22}(w_{12}x_{21}+w_{22}x_{22}+w_{32}x_{23})}\sigma(\mathbf{n}_{22}(w_{12}x_{21}+w_{22}x_{22}+w_{32}x_{23})) \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{\partial}{\partial \mathbf{n}_{31}(w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33})}\sigma(\mathbf{n}_{31}(w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33})) & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & \frac{\partial}{\partial \mathbf{n}_{32}(w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33})}\sigma(\mathbf{n}_{32}(w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33})) \end{bmatrix} \end{bmatrix}$$

[52]: `st[0,0].diff(st[0,0].args[0])`

[52]:

$$\frac{\partial}{\partial \mathbf{n}_{11}(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13})}\sigma(\mathbf{n}_{11}(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13}))$$

[53]: `st[0,0].diff(X[0,0])`

[53]:

$$w_{11}\frac{\partial}{\partial \mathbf{n}_{11}(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13})}\sigma(\mathbf{n}_{11}(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13}))\frac{d}{d\xi_1}\mathbf{n}_{11}(\xi_1)\Big|_{\xi_1=w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13}}$$

[54]: `st[0,0].diff(st[1,0].args[0])`

[54]:

$$0$$

[55]: `Selem.diff(Nelem)`

[55]:

$$\begin{bmatrix} \begin{bmatrix} \frac{d}{dn_{11}}\sigma(n_{11}) & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & \frac{d}{dn_{12}}\sigma(n_{12}) \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ \frac{d}{dn_{21}}\sigma(n_{21}) & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & \frac{d}{dn_{22}}\sigma(n_{22}) \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{d}{dn_{31}}\sigma(n_{31}) & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & \frac{d}{dn_{32}}\sigma(n_{32}) \end{bmatrix} \end{bmatrix}$$

[56]: `Selem.diff(Nelem).subs(elemToSpecFunc)`

[56]:

$$\begin{bmatrix} \begin{bmatrix} \frac{\partial}{\partial \mathbf{n}_{11}(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13})}\sigma(\mathbf{n}_{11}(w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13})) & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & \frac{\partial}{\partial \mathbf{n}_{12}(w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13})}\sigma(\mathbf{n}_{12}(w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13})) \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ \frac{\partial}{\partial \mathbf{n}_{21}(w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23})}\sigma(\mathbf{n}_{21}(w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23})) & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & \frac{\partial}{\partial \mathbf{n}_{22}(w_{12}x_{21}+w_{22}x_{22}+w_{32}x_{23})}\sigma(\mathbf{n}_{22}(w_{11}x_{21}+w_{22}x_{22}+w_{32}x_{23})) \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{\partial}{\partial \mathbf{n}_{31}(w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33})}\sigma(\mathbf{n}_{31}(w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33})) & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & \frac{\partial}{\partial \mathbf{n}_{32}(w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33})}\sigma(\mathbf{n}_{32}(w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33})) \end{bmatrix} \end{bmatrix}$$

```
[57]: # CAN even replace elements after have done an operation on them!!! replacing n_21 * 2 with the number 4.
      Sspec.subs({Nspec[0, 0]: 3}).replace(sigma, lambda x: 2 * x).replace(Nspec[2, 1] * 2, 4)
```

[57]:
$$\begin{bmatrix} 6 & 2w_{12}x_{11} + 2w_{22}x_{12} + 2w_{32}x_{13} \\ 2w_{11}x_{21} + 2w_{21}x_{22} + 2w_{31}x_{23} & 2w_{12}x_{21} + 2w_{22}x_{22} + 2w_{32}x_{23} \\ 2w_{11}x_{31} + 2w_{21}x_{32} + 2w_{31}x_{33} & 4 \end{bmatrix}$$

```
[58]: lambd = Function("lambda")
      lambd_ = lambda matrix : sum(matrix)


      vN(X, W)
```

[58]:
$$\begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \\ n_{31} & n_{32} \end{bmatrix}$$

```
[59]: vN(A, B)
```

[59]:
$$\begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \\ n_{31} & n_{32} \end{bmatrix}$$

```
[60]: L = lambd(S); L
```

[60]: $\lambda(\sigma_{apply}(v(X, W)))$

```
[61]: Nelem
```

[61]:
$$\begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \\ n_{31} & n_{32} \end{bmatrix}$$

```
[62]: L.replace(n, vN)
```

[62]: $\lambda\left(\sigma_{apply}\left(\begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \\ n_{31} & n_{32} \end{bmatrix}\right)\right)$

```
[63]: L.replace(n, vN).replace(sigmaApply, sigmaApply_)
```

[63]: $\lambda\left(\begin{bmatrix} \sigma(n_{11}) & \sigma(n_{12}) \\ \sigma(n_{21}) & \sigma(n_{22}) \\ \sigma(n_{31}) & \sigma(n_{32}) \end{bmatrix}\right)$

```
[64]: L.replace(n, v)
```

[64]: $\lambda(\sigma_{apply}(XW))$

```
[65]: L.replace(n, v).replace(sigmaApply, sigmaApply_)
```

$$\lambda\left((d \mapsto \sigma(d))_\circ (XW)\right)$$

```
[66]: L.subs({A:X, B:W}).replace(n, vL).replace(sigmaApply, sigmaApply_)
```

[66]:
$$\lambda\left(\begin{bmatrix} \sigma(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}) & \sigma(w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13}) \\ \sigma(w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23}) & \sigma(w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23}) \\ \sigma(w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33}) & \sigma(w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33}) \end{bmatrix}\right)$$

```
[67]: L.replace(n, vN)
```

[67]:
$$\lambda\left(\sigma_{apply}\left(\begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \\ n_{31} & n_{32} \end{bmatrix}\right)\right)$$

```
[68]: L.replace(n, vN).subs({A:X, B:W}).replace(sigmaApply, sigmaApply_).replace(lambd, lambd_)
```

[68]:
$$\sigma(n_{11}) + \sigma(n_{12}) + \sigma(n_{21}) + \sigma(n_{22}) + \sigma(n_{31}) + \sigma(n_{32})$$

```
[69]: from sympy import symbols, Derivative

      x, y, r, t = symbols('x y r t') # r (radius), t (angle theta)
      f, g, h = symbols('f g h', cls=Function)
      h = g(f(x))

      Derivative(h, f(x)).doit()
```

[69]:
$$\frac{d}{df(x)} g(f(x))$$

```
[70]: # Never do this gives recursion ERROR (max depth exceeded)
      # h = g(f(A))
      # Derivative(h, A).doit()
```

```
[71]:
```

```
[71]: from sympy.abc import a, b

      Llower = lambd(sigmaApply(n(a, b)))
      Llower
```

[71]:
$$\lambda(\sigma_{apply}(v(a,b)))$$

```
[72]: Derivative(Llower, a).doit()
```

[72]:
$$\frac{\partial}{\partial \sigma_{apply}(v(a,b))} \lambda(\sigma_{apply}(v(a,b))) \frac{\partial}{\partial v(a,b)} \sigma_{apply}(v(a,b)) \frac{\partial}{\partial a} v(a,b)$$

```
[73]:
```

```
[73]: # ### WAY 1: of substituting to differentiate with respect to expression:
      n_ij = Function('n_ij')
      n_ij(A,B) # (N[0,0]); n_ij
```

[73]: $\mathbf{n_{ij}}\left(X, W\right)$

```
[74]: n_ij(A,B).args
```

[74]: $\left(X, \ W\right)$

```
[75]: # sigma(n_ij).diff(n_ij).replace(n_ij, N[0,0]) # ERROR cannot deriv wi.r.t to the expression w11*x11 + ...

      sigma(n_ij(A,B)).diff(n_ij(A,B))
```

[75]: $\dfrac{\partial}{\partial\, \mathbf{n_{ij}}\left(X, W\right)}\sigma\big(\mathbf{n_{ij}}\left(X, W\right)\big)$

```
[76]: sigma(n_ij(*X,*W)).diff(X[0,0])
```

[76]: $\dfrac{\partial}{\partial x_{11}}\mathbf{n_{ij}}\left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32}\right)\dfrac{\partial}{\partial\, \mathbf{n_{ij}}\left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32}\right)}\sigma\big(\mathbf{n_{ij}}\left(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}, w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32}\right)\big)$

```
[77]: nab_ij = n_ij(A,B)
      sigma(nab_ij).diff(nab_ij)#.subs({nab_ij : Nspec[0, 0]})
```

[77]: $\dfrac{\partial}{\partial\, \mathbf{n_{ij}}\left(X, W\right)}\sigma\big(\mathbf{n_{ij}}\left(X, W\right)\big)$

```
[78]: sigma(nab_ij).diff(nab_ij).subs({nab_ij : Nspec[2, 1]})
```

[78]: $\dfrac{d}{d\xi}\sigma(\xi)\Big|_{\xi=w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33}}$

```
[79]: sigma(nab_ij).diff(nab_ij).subs({nab_ij : Nspec[2,1]}).subs({X[2,1]:77777})
```

[79]: $\dfrac{d}{d\xi}\sigma(\xi)\Big|_{\xi=w_{12}x_{31}+77777w_{22}+w_{32}x_{33}}$

```
[80]: sigma(nab_ij).diff(nab_ij).subs({nab_ij : 23}) # ERROR if using replace() since it says can't calc derivs w.r.t to the x_11*w_11 + ...
```

[80]: $\dfrac{d}{d\xi}\sigma(\xi)\Big|_{\xi=23}$

```
[81]: sigma(nab_ij).diff(nab_ij).subs({nab_ij : Nspec[2,1]}).doit()
```

[81]: $\dfrac{d}{d\xi}\sigma(\xi)\Big|_{\xi=w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33}}$

```
[82]: sigma(nab_ij).subs({nab_ij : Nspec[2,1]})#.diff(X[2,1])
```

[82]:

$$\sigma(w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33})$$

```
[83]: # Substituting the value of the function n_ij first, and THEN differentiating with respect to something in that substitution. (X_21)
      sigma(nab_ij).subs({nab_ij : Nspec[2,1]}).diff(X[2,1])
```

[83]: $$w_{22} \left. \frac{d}{d\xi_1} \sigma(\xi_1) \right|_{\xi_1 = w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33}}$$

```
[84]: Selem[2,1].subs({Nelem[2,1] : Nspec[2,1]}).diff(X[2,1])
```

[84]: $$w_{22} \left. \frac{d}{d\xi_1} \sigma(\xi_1) \right|_{\xi_1 = w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33}}$$

```
[85]: # ### WAY 2:
      n_11 = Function('n_11')(Nspec[0, 0]); n_11
```

[85]: $$\mathfrak{n}_{11}\left(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}\right)$$

```
[86]: sigma(n_11)
```

[86]: $$\sigma(\mathfrak{n}_{11}\left(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}\right))$$

```
[87]: assert Nspec[0,0] == n_11.args[0]

      sigma(n_11).subs({n_11 : n_11.args[0]})
```

[87]: $$\sigma(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13})$$

```
[88]: sigma(n_11).diff(n_11) #.replace(n_ij, n_ij.args[0])
```

[88]: $$\frac{\partial}{\partial \mathfrak{n}_{11}\left(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}\right)} \sigma(\mathfrak{n}_{11}\left(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}\right))$$

```
[89]: sigma(n_11).diff(n_11).subs({n_11 : n_11.args[0]}).subs({X[0,0]:77777})
```

[89]: $$\left. \frac{d}{d\xi} \sigma(\xi) \right|_{\xi = 77777 w_{11} + w_{21}x_{12} + w_{31}x_{13}}$$

```
[90]: sigma(n_11).diff(n_11).subs({n_11 : n_11.args[0]}).replace(n_11.args[0], 23) # same as subs in this case
```

[90]: $$\left. \frac{d}{d\xi} \sigma(\xi) \right|_{\xi = 23}$$

```
[91]: sigma(n_11).diff(X[0,0])
```

[91]: $$w_{11} \left. \frac{\partial}{\partial \mathfrak{n}_{11}\left(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}\right)} \sigma(\mathfrak{n}_{11}\left(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}\right)) \frac{d}{d\xi_1} \mathfrak{n}_{11}\left(\xi_1\right) \right|_{\xi_1 = w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}}$$

```
[92]: id = Lambda(x, x)
```

```python
sigma(n_11).diff(X[0,0]).subs({n_11 : id})
```

[92]:
$$w_{11} \left. \frac{d}{d\xi_1} \mathbf{n_{11}}\left(\xi_1\right) \right|_{\xi_1 = w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}} \left. \frac{d}{d\xi}\sigma(\xi) \right|_{\xi = (x \mapsto x)}$$

[93]:
```python
# NOTE: so I don't think WAY 2 is correct because here it doesn't simplify the derivative d n11 / d eps11, since this should equal 1 because now n11 = eps11.⬚
 ↪Correct one is below (repeated from above)
sigma(n_11).diff(X[0,0]).subs({n_11 : Nspec[0,0]})
```

[93]:
$$w_{11} \left. \frac{d}{d\xi_1} \mathbf{n_{11}}\left(\xi_1\right) \right|_{\xi_1 = w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}} \left. \frac{d}{d\xi}\sigma(\xi) \right|_{\xi = w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}}$$

[94]:
```python
# CORRECT WAY 1
sigma(n_11).subs({n_11 : Nspec[0,0]}).diff(X[0,0])
```

[94]:
$$w_{11} \left. \frac{d}{d\xi_1}\sigma(\xi_1) \right|_{\xi_1 = w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}}$$

[95]:
```python
# CORRECT WAY 2

sigma(nab_ij).subs({nab_ij : Nspec[0,0]}).diff(X[0,0])
```

[95]:
$$w_{11} \left. \frac{d}{d\xi_1}\sigma(\xi_1) \right|_{\xi_1 = w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}}$$

[96]:
```python
# CORRECT WAY 3
Selem[2,1].subs({Nelem[2,1] : Nspec[2,1]}).diff(X[2,1])
```

[96]:
$$w_{22} \left. \frac{d}{d\xi_1}\sigma(\xi_1) \right|_{\xi_1 = w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33}}$$

[97]:
```python
sigma(n_11) # WAY 1: sigma argument is already hardcoded
```

[97]:
$$\sigma(\mathbf{n_{11}}\left(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}\right))$$

[98]:
```python
sigma(nab_ij) # Way 2: sigma argument is function of matrixsymbol (better than 1)
```

[98]:
$$\sigma(\mathbf{n_{ij}}\left(X, W\right))$$

[99]:
```python
Selem[2,1] # WAY 3: sigma argument is just symbol and we replace it as function with argument hardcoded only later. (better than 2)
```

[99]:
$$\sigma(n_{32})$$

[100]:
```python
L
```

[100]:
$$\lambda(\sigma_{apply}(v(X, W)))$$

[101]:
```python
assert Selem == S.replace(n, vN).replace(sigmaApply, sigmaApply_)

Selem
```

[101]:
$$\begin{bmatrix} \sigma(n_{11}) & \sigma(n_{12}) \\ \sigma(n_{21}) & \sigma(n_{22}) \\ \sigma(n_{31}) & \sigma(n_{32}) \end{bmatrix}$$

[102]:
```
L.replace(n, vN).replace(sigmaApply, sigmaApply_)
```

[102]:
$$\lambda\left(\begin{bmatrix} \sigma(n_{11}) & \sigma(n_{12}) \\ \sigma(n_{21}) & \sigma(n_{22}) \\ \sigma(n_{31}) & \sigma(n_{32}) \end{bmatrix}\right)$$

[103]:
```
#L.replace(n, vN).replace(sigmaApply, sigmaApply_).diff(Nelem[0,0])
```

[104]:
```
Lsum = L.replace(n, vN).replace(sigmaApply, sigmaApply_).replace(lambd, lambd_)
Lsum
```

[104]:
$$\sigma(n_{11}) + \sigma(n_{12}) + \sigma(n_{21}) + \sigma(n_{22}) + \sigma(n_{31}) + \sigma(n_{32})$$

[105]:
```
Lsum.diff(Nelem)
```

[105]:
$$\begin{bmatrix} \frac{d}{dn_{11}}\sigma(n_{11}) & \frac{d}{dn_{12}}\sigma(n_{12}) \\ \frac{d}{dn_{21}}\sigma(n_{21}) & \frac{d}{dn_{22}}\sigma(n_{22}) \\ \frac{d}{dn_{31}}\sigma(n_{31}) & \frac{d}{dn_{32}}\sigma(n_{32}) \end{bmatrix}$$

[106]:
```
Lsum.subs(elemToSpec)#.diff(X[2,1])
```

[106]:
$$\sigma(w_{11}x_{11} + w_{21}x_{12} + w_{31}x_{13}) + \sigma(w_{11}x_{21} + w_{21}x_{22} + w_{31}x_{23}) + \sigma(w_{11}x_{31} + w_{21}x_{32} + w_{31}x_{33}) + \sigma(w_{12}x_{11} + w_{22}x_{12} + w_{32}x_{13}) + \sigma(w_{12}x_{21} + w_{22}x_{22} + w_{32}x_{23}) + \sigma(w_{12}x_{31} + w_{22}x_{32} + w_{32}x_{33})$$

[107]:
```
Lsum.subs(elemToSpec).diff(X)
```

[107]:
$$\begin{bmatrix} w_{11}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13}} + w_{12}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13}} & w_{21}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13}} + w_{22}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{11}+w_{22}x_{12}+w_{32}x_{13}} & w_{31}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{11}+w_{21}x_{12}+w_{31}x_{13}} + w_{32}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{21}+w_{22}x_{12}+w_{32}x_{13}} \\ w_{11}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23}} + w_{12}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{21}+w_{22}x_{22}+w_{32}x_{23}} & w_{21}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23}} + w_{22}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{21}+w_{22}x_{22}+w_{32}x_{23}} & w_{31}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{21}+w_{21}x_{22}+w_{31}x_{23}} + w_{32}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{21}+w_{22}x_{22}+w_{32}x_{23}} \\ w_{11}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33}} + w_{12}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33}} & w_{21}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33}} + w_{22}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33}} & w_{31}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{11}x_{31}+w_{21}x_{32}+w_{31}x_{33}} + w_{32}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=w_{12}x_{31}+w_{22}x_{32}+w_{32}x_{33}} \end{bmatrix}$$

[108]:
```
specToElemD = {v : k for k, v in elemToSpecD.items()}

Lsum.subs(elemToSpecD).diff(X).subs(specToElemD)
```

[108]:
$$\begin{bmatrix} w_{11}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{11}} + w_{12}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{12}} & w_{21}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{11}} + w_{22}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{12}} & w_{31}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{11}} + w_{32}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{12}} \\ w_{11}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{21}} + w_{12}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{22}} & w_{21}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{21}} + w_{22}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{22}} & w_{31}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{21}} + w_{32}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{22}} \\ w_{11}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{31}} + w_{12}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{32}} & w_{21}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{31}} + w_{22}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{32}} & w_{31}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{31}} + w_{32}\frac{d}{d\xi_1}\sigma(\xi_1)\Big|_{\xi_1=n_{32}} \end{bmatrix}$$