Lecture Notes 1a: Brief Guide to Matrices

what meaneth we by matrix?

A matrix is a 2-dimensional array of things, where these things can be numbers, variables, or math expressions. The matrix is navigated by rows and columns, and we always name the row first when locating any particular element of the matrix. When presenting matrices, we double-underline the summary name $\underline{\underline{A}}$ of the matrix, corral the elements in braces, and provide row-column subscripts to locate individual elements. (Regard a_{ij} as the name for some quantity of interest – number, variable, expression.)

$$\underline{\underline{A}} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix}$$

Matlab indicates an element by putting subscripts in parentheses after the matrix name

```
>> a = [ 1 2 ; 3 4 ]

a =
     1     2
     3     4

>> a(1,2)

ans =
     2
```

A row matrix is a single row, and a column matrix a single column; we collectively call these 1-dimensional matrices "vectors". We use a single underline for the name. We could still use double R,C subscripts, but I don't think most people bother with that.

$$\underline{y} = \begin{bmatrix} y_{11} & y_{12} & y_{13} \end{bmatrix} = \begin{bmatrix} y_1 & y_2 & y_3 \end{bmatrix} \qquad \underline{x} = \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

In matlab, use the semicolon to make a row shift.

```
>> y = [ 1 2 3 ]

y =
     1     2     3

>> x = [1; 2; 3 ]

x =
     1
     2
     3
```

combining a matrix with a scalar quantity

$$c\underline{\underline{A}} = \underline{\underline{A}}c = c\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} ca_{11} & ca_{12} \\ ca_{21} & ca_{22} \end{bmatrix}$$

$$c + \underline{\underline{A}} = \underline{\underline{A}} + c = c + \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} c + a_{11} & c + a_{12} \\ c + a_{21} & c + a_{22} \end{bmatrix}$$

adding two matrices (they must be of the same shape)

$$\underline{\underline{A}} + \underline{\underline{B}} = \underline{\underline{B}} + \underline{\underline{A}} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} \\ a_{21} + b_{21} & a_{22} + b_{22} \end{bmatrix}$$

the transpose of a matrix is another matrix; rows and columns are interchanged

$$\underline{x}^T = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}^T = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \qquad \underline{\underline{A}}^T = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix}^T = \begin{bmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{32} \end{bmatrix}$$

(Notice in this last example that the subscripts are not to be interpreted merely as placeholders, but as identifying a particular element. That is, a_{31} means "the actual quantity that was in position (3,1) in matrix $\underline{\underline{A}}$, but is in position (1,3) in matrix $\underline{\underline{A}}^T$.)

In matlab, compute the transpose by a single quote after the matrix name

```
a =
     1     2
     3     4

>> a'

ans =
     1     3
     2     4
```

multiplying matrices is where things get more complicated

The "inner dimensions" must be the same for multiplication to be possible. Multiplication is <u>not</u> commutative.

$$\underline{x}\underline{y} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\begin{bmatrix} y_1 & y_2 \end{bmatrix} = \begin{bmatrix} x_1 y_1 & x_1 y_2 \\ x_2 y_1 & x_2 y_2 \end{bmatrix} \qquad [2,1][1,2] \to [2,2]$$

$$\underline{y}\underline{x} = \begin{bmatrix} y_1 & y_2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = x_1 y_1 + x_2 y_2 \qquad [1,2][2,1] \to [1,1] = \text{scalar}$$

$$\underline{\underline{A}}\,\underline{\underline{B}} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$

$$\underline{\underline{B}}\,\underline{\underline{A}} = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} + a_{21}b_{12} & a_{12}b_{11} + a_{22}b_{12} \\ a_{11}b_{21} + a_{21}b_{22} & a_{12}b_{21} + a_{22}b_{22} \end{bmatrix}$$

Matlab uses * to indicate matrix multiplication

```
a =
     1     2
     3     4

b =
     2     4
     3     6

>> a*b

ans =
     8    16
    18    36

>> b*a

ans =
    14    20
    21    30
```

element-by-element multiplication is only for same-shape matrices

$$\underline{\underline{A}} \otimes \underline{\underline{B}} = \underline{\underline{B}} \otimes \underline{\underline{A}} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} \\ a_{21}b_{21} & a_{22}b_{22} \end{bmatrix}$$

Matlab uses .* to indicate element by element multiplication

```
>> a.*b

ans =
     2     8
     9    24
```

the determinant is a scalar quantity computed from the elements of a square matrix
For a 2×2 matrix, the determinant is the difference of two products. The products are taken along diagonals in the matrix: positive for the diagonal that slopes down to the right, and negative for that sloping down to the left.

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$$

\- +

$$\det\left(\underline{\underline{A}}\right) = \left|\underline{\underline{A}}\right| = a_{11}a_{22} - a_{12}a_{21}$$

For a 3×3 matrix, the determinant requires six products. The diagonal rule is extended: positive for the three diagonals that slope down to the right, and negative for those sloping down to the left.

$$\left|\underline{\underline{A}}\right| = \left|\begin{matrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{matrix}\right|\begin{matrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{matrix}$$

\- +

$$\det\left(\underline{\underline{A}}\right) = \left|\underline{\underline{A}}\right| = a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} - a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33}$$

For larger determinants, the diagonal rule does not work; we will not go into these, however.

Matlab

```
d =
     1     2     2
     3     4     3
     2     4     1

>> det(d)

ans =
     6
```

the cofactor is a sub-determinant pulled from a larger determinant

Take any element a_{ij}; delete row i and column j. The remaining determinant, when multiplied by $(-1)^{i+j}$, is the cofactor of a_{ij}. For example, the cofactor of a_{21} in a 3×3 determinant is

$$\left|\underline{\underline{A}}\right| = \left|\begin{matrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{matrix}\right|$$

$$(-1)^{2+1}\left|\begin{matrix} a_{12} & a_{13} \\ a_{32} & a_{33} \end{matrix}\right| = -\left(a_{12}a_{33} - a_{13}a_{32}\right)$$

(In this example, the subscripts are not mere placeholders, but refer to those particular elements of a 3×3 determinant that were extracted into the cofactor.)

the adjoint is a transposed square matrix of cofactors

Each matrix element is replaced by its cofactor, and then the new matrix is transposed.

$$\operatorname{adj}\left(\underline{\underline{A}}\right)=\operatorname{adj}\left(\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\right)=\begin{bmatrix} \begin{vmatrix} a_{22} & a_{22} \\ a_{32} & a_{33} \end{vmatrix} & -\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} & \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \\ -\begin{vmatrix} a_{12} & a_{13} \\ a_{32} & a_{33} \end{vmatrix} & \begin{vmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{vmatrix} & -\begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} \\ \begin{vmatrix} a_{12} & a_{13} \\ a_{22} & a_{23} \end{vmatrix} & -\begin{vmatrix} a_{11} & a_{13} \\ a_{21} & a_{23} \end{vmatrix} & \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \end{bmatrix}^T$$

$$=\begin{bmatrix} \begin{vmatrix} a_{22} & a_{22} \\ a_{32} & a_{33} \end{vmatrix} & -\begin{vmatrix} a_{12} & a_{13} \\ a_{32} & a_{33} \end{vmatrix} & \begin{vmatrix} a_{12} & a_{13} \\ a_{22} & a_{23} \end{vmatrix} \\ -\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} & \begin{vmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{vmatrix} & -\begin{vmatrix} a_{11} & a_{13} \\ a_{21} & a_{23} \end{vmatrix} \\ \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} & -\begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} & \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \end{bmatrix}$$

the inverse is the adjoint divided by the determinant

$$\operatorname{inv}\left(\underline{\underline{A}}\right)=\underline{\underline{A}}^{-1}=\frac{\operatorname{adj}\left(\underline{\underline{A}}\right)}{\det\left(\underline{\underline{A}}\right)}$$

We care about the inverse, because

$$\underline{\underline{A}}\,\underline{\underline{A}}^{-1}=\underline{\underline{A}}^{-1}\underline{\underline{A}}=\underline{\underline{I}}=\begin{bmatrix} 1 & 0 & & 0 \\ 0 & 1 & & 0 \\ & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

$\underline{\underline{I}}$ is the identity matrix; any matrix $\underline{\underline{A}}$ multiplied by (suitably sized) $\underline{\underline{I}}$ is just $\underline{\underline{A}}$.

Matlab

```
>> d, inv(d)

d =
     1     2     2
     3     4     3
     2     4     1

ans =
   -1.3333    1.0000   -0.3333
    0.5000   -0.5000    0.5000
    0.6667         0   -0.3333

>> d*inv(d)

ans =
    1.0000         0    0.0000
    0.0000    1.0000    0.0000
   -0.0000         0    1.0000
```

the inverse allows us to solve matrix equations

Solve for $\underline{x}$; the order of multiplication is important in the following steps.

$$\underline{\underline{A}}\,\underline{x} = \underline{b}$$

$$\underline{\underline{A}}^{-1}\underline{\underline{A}}\,\underline{x} = \underline{\underline{A}}^{-1}\underline{b} \qquad \left(\text{not } \underline{b}\,\underline{\underline{A}}^{-1}\right)$$

$$\underline{\underline{I}}\,\underline{x} = \underline{\underline{A}}^{-1}\underline{b}$$

$$\underline{x} = \underline{\underline{A}}^{-1}\underline{b}$$

The inverse is not always computationally efficient. In Matlab, an alternative is “left division”, \.

```
a =
     1     2
     3     4

b =
     2
     3

>> x=inv(a)*b

x =
   -1.0000
    1.5000

>> x=a\b          % alternative to inverse; order still matters!

x =
   -1.0000
    1.5000
```