STAT 309: MATHEMATICAL COMPUTATIONS I
FALL 2015
LECTURE 15

1. MULTIPLE RIGHT-HAND SIDES AND INVERSE

- let $A \in \mathbb{R}^{m\times n}$ and $\mathbf{b}_1, \dots, \mathbf{b}_p \in \mathbb{R}^m$
- suppose we need to solve p linear systems with the same coefficient matrix but different right-hand sides
$$A\mathbf{x}_1 = \mathbf{b}_1, \quad A\mathbf{x}_2 = \mathbf{b}_2, \quad \dots, \quad A\mathbf{x}_p = \mathbf{b}_p \tag{1.1}$$
- this is equivalent to solving the matrix equation
$$AX = B$$
where $X = [\mathbf{x}_1, \dots, \mathbf{x}_p] \in \mathbb{R}^{n\times p}$ and $B = [\mathbf{b}_1, \dots, \mathbf{b}_p] \in \mathbb{R}^{m\times p}$
- for example, this is what we do when we need to compute the inverse of an $n\times n$ nonsingular matrix A:
$$AX = I,$$
which is equivalent to the systems of equations
$$A\mathbf{x}_j = \mathbf{e}_j, \quad j = 1, \dots, n$$
- since only the right-hand side is different in each of these systems, we need only compute the LU factorization of A once
- more generally, this is how we should compute $A^{-1}B$ for matrices A and B, we should solve (1.1) instead of finding the explicit inverse A^{-1} and then multiplying it to B (exercise: what if you need AB^{-1}?)
- we didn't say too much about why it's a bad idea to compute the explicit inverse of a matrix, for more information about this topic, see Chapter 14 in: N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2nd Ed, SIAM, 2002

2. BLOCK FACTORIZATIONS AND SCHUR COMPLEMENT

- a surprisingly simple and powerful idea that appeared implicitly several times in our earlier discussions is that of *block elimination* and *block factorization*
- all it involves is to consider a matrix $A \in \mathbb{R}^{n\times n}$ as a 2×2 block matrix
$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$
where $A_{11} \in \mathbb{R}^{p\times p}$, $A_{22} \in \mathbb{R}^{q\times q}$, $A_{12} \in \mathbb{R}^{q\times p}$, $A_{21} \in \mathbb{R}^{p\times q}$ for some p and q where $p+q=n$
- this works for rectangular matrices too but we keep our discussion to square matrices for simplicity
- many of the stuff that we discussed can be carried over to block matrices
- for example, if A_{11} is nonsingular, we could define an $n\times n$ block elimination matrix
$$M_1 = I - U_1 V_1^{\mathsf{T}}$$

Date: November 17, 2015, version 1.0. Comments, bug reports: `lekheng@galton.uchicago.edu`.

where $U_1, V_1 \in \mathbb{R}^{n \times p}$ are

$$U_1 = \begin{bmatrix} 0 \\ A_{21}A_{11}^{-1} \end{bmatrix}, \quad V_1 = \begin{bmatrix} I_p \\ 0 \end{bmatrix}$$

- in other words

$$M_1 = \begin{bmatrix} I_p & 0 \\ 0 & I_q \end{bmatrix} - \begin{bmatrix} 0 \\ A_{21}A_{11}^{-1} \end{bmatrix} \begin{bmatrix} I_p & 0 \end{bmatrix} = \begin{bmatrix} I_p & 0 \\ -A_{21}A_{11}^{-1} & I_q \end{bmatrix}$$

- applying this to A gives

$$M_1 A = \begin{bmatrix} I_p & 0 \\ -A_{21}A_{11}^{-1} & I_q \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ 0 & S \end{bmatrix}$$

  where

$$S = A_{22} - A_{21}A_{11}^{-1}A_{12}$$

  is called the *Schur complement* of A_{11} in A
- we can easy verify that

$$L_1 := M_1^{-1} = \begin{bmatrix} I_p & 0 \\ A_{21}A_{11}^{-1} & I_q \end{bmatrix}$$

- the analogue of LU factorization of A as a 2×2 block matrix

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix}$$

  is called a *block LU factorization*
- note that L_{11} and L_{22} can be any matrices, not necessarily lower triangular, ditto for U_{11} and U_{22}
- multiplying out the RHS, we see that

$$A_{11} = L_{11}U_{11}$$

- it is also easy to see that

$$L_{22}U_{22} = S = A_{22} - A_{21}A_{11}^{-1}A_{12}$$

- we omitted permutation matrices but they can be easily incorporated: for example, if

$$A_{11} = \Pi_1^{\mathsf{T}} L_1 U_1 \Pi_2^{\mathsf{T}}, \quad S = \Pi_3^{\mathsf{T}} L_2 U_2$$

  then we have

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \Pi_1^{\mathsf{T}} & 0 \\ 0 & \Pi_3^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} L_1 & 0 \\ \Pi_3 A_{21} \Pi_2 U_1^{-1} & L_2 \end{bmatrix} \begin{bmatrix} U_1 & L_1^{-1}\Pi_1 A_{12} \\ 0 & U_2 \end{bmatrix} \begin{bmatrix} \Pi_2^{\mathsf{T}} & 0 \\ 0 & I \end{bmatrix}$$

- what we discuss here also apply to LDU, LDL^{T}, and Cholesky factorizations
- for example if A is symmetric positive definite, then its Cholesky factorization written in 2×2 block form

$$\begin{bmatrix} A_{11} & A_{21}^{\mathsf{T}} \\ A_{21} & A_{22} \end{bmatrix} = A = R^{\mathsf{T}} R = \begin{bmatrix} R_{11}^{\mathsf{T}} & 0 \\ R_{12}^{\mathsf{T}} & R_{22}^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix} = \begin{bmatrix} R_{11}^{\mathsf{T}} R_{11} & R_{11}^{\mathsf{T}} R_{12} \\ R_{12}^{\mathsf{T}} R_{11} & R_{12}^{\mathsf{T}} R_{12} + R_{22}^{\mathsf{T}} R_{22} \end{bmatrix}$$

  is called *block Cholesky factorization*
- again R_{11} and R_{22} need not be upper triangular
- note that since A is symmetric positive definite, so is A_{11} (why?)
- multiplying out the RHS, we see that

$$A_{11} = R_{11}^{\mathsf{T}} R_{11}$$

- it is also easy to see that

$$R_{22}^{\mathsf{T}} R_{22} = A_{22} - A_{21}A_{11}^{-1}A_{21}^{\mathsf{T}}$$

3. MORE ON THE SCHUR COMPLEMENT

- Schur complement is a very useful notion
- in the following we will assume that A is partitioned as in the previous section with A_{11} nonsingular
- the first useful property is that A is nonsingular if and only if S is nonsinguar
- a second very useful application is in *solving linear equations by block elimination*, i.e., solving $A\mathbf{x} = \mathbf{b}$ by partitioning it into

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \tag{3.1}$$

  where $\mathbf{b}_1 \in \mathbb{R}^p, \mathbf{b}_2 \in \mathbb{R}^q$
- plugging the first equation

$$\mathbf{x}_1 = A_{11}^{-1}(\mathbf{b}_1 - A_{12}\mathbf{x}_2) \tag{3.2}$$

  into the second equation yields

$$(A_{22} - A_{21}A_{11}^{-1}A_{12})\mathbf{x}_2 = \mathbf{b}_2 - A_{21}A_{11}^{-1}\mathbf{b}_1 \tag{3.3}$$

- this allows us to solve $A\mathbf{x} = \mathbf{b}$ as follows
  - form $A_{11}^{-1}A_{12}$ and $A_{11}^{-1}\mathbf{b}$ by solving a system with multiple right hand sides
  - form $S = A_{22} - A_{21}A_{11}^{-1}A_{12}$ and $\widetilde{\mathbf{b}} = \mathbf{b}_2 - A_{21}A_{11}^{-1}\mathbf{b}_1$
  - solve $S\mathbf{x}_2 = \widetilde{\mathbf{b}}$ for $\mathbf{x}_2$
  - solve $A_{11}\mathbf{x}_1 = \mathbf{b}_1 - A_{12}\mathbf{x}_2$ for $\mathbf{x}_1$
- this would be very useful if A_{11} is an ‘easy to invert’ matrix, e.g., A_{11} is diagonal, banded, orthogonal, Toeplitz, sparse, etc
- such situations where the ‘top left corner’ of a matrix A has special structure arise more often than you think, especially in
  - numerical optimization (KKT matrix — A_{11} corresponds to the Hessian, the other blocks correpond to the constraints)
  - numerical PDE (discretized version of differential operator with boundary conditions — A_{11} corresponds to the operator, the other blocks to the boundary conditions)
- another way to view the above method is via the factorization

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & 0 \\ A_{21} & S \end{bmatrix} \begin{bmatrix} I & A_{11}^{-1}A_{12} \\ 0 & I \end{bmatrix} \tag{3.4}$$

- so solving $A\mathbf{x} = \mathbf{b}$ can be broken up into two steps

$$\begin{cases} \begin{bmatrix} A_{11} & 0 \\ A_{21} & S \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \\[2ex] \begin{bmatrix} I & A_{11}^{-1}A_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} \end{cases}$$

- or equivalently

$$\begin{cases} A_{11}\mathbf{y}_1 = \mathbf{b}_1 \\ S\mathbf{y}_2 = \mathbf{b}_2 - A_{21}\mathbf{y}_1 \\ \mathbf{x}_2 = \mathbf{y}_2 \\ \mathbf{x}_1 = \mathbf{y}_1 - A_{11}^{-1}A_{12}\mathbf{y}_2 \end{cases}$$

- a third application is to use (3.4) to evaluate determinant

$$\det(A) = \det(A_{11})\det(S)$$

- while a fourth is in inverting block matrices

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}^{-1} = \begin{bmatrix} A_{11}^{-1} + A_{11}^{-1} A_{12} S^{-1} A_{21} A_{11}^{-1} & -A_{11}^{-1} A_{12} S^{-1} \\ -S^{-1} A_{21} A_{11}^{-1} & S^{-1} \end{bmatrix}$$

- the trick to derive this expression is to consider (3.1) and try to express

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}$$

  and in which case $B = A^{-1}$
- we already have (3.3) which expresses $\mathbf{x}_2$ in terms of $\mathbf{b}_1$ and $\mathbf{b}_2$
- we need something similar for $\mathbf{x}_1$ and so we plug (3.3) back into (3.2) which gives us

$$\mathbf{x}_1 = (A_{11}^{-1} + A_{11}^{-1} A_{12} S^{-1} A_{21} A_{11}^{-1})\mathbf{b}_1 - A_{11}^{-1} A_{12} S^{-1} \mathbf{b}_2 \tag{3.5}$$

- now we just write (3.3) and (3.5) in block form

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} A_{11}^{-1} + A_{11}^{-1} A_{12} S^{-1} A_{21} A_{11}^{-1} & -A_{11}^{-1} A_{12} S^{-1} \\ -S^{-1} A_{21} A_{11}^{-1} & S^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}$$

  which yields the required formula

4. Rank-1 updating

- suppose that we have solved the problem $A\mathbf{x} = \mathbf{b}$ and we wish to solve the perturbed problem

$$(A + \mathbf{u}\mathbf{v}^{\mathsf{T}})\mathbf{y} = \mathbf{b}$$

- such a perturbation is called a *rank-one update* of A, since the matrix $\mathbf{u}\mathbf{v}^{\mathsf{T}}$ has rank 1 (unless $\mathbf{u}$ or $\mathbf{v}$ is zero)
- as an example, we might find that there was an error in the element a_{11} and we update it with the value $\bar{a}_{11}$
- we can accomplish this update by setting

$$\bar{A} = A + (\bar{a}_{11} - a_{11})\mathbf{e}_1\mathbf{e}_1^{\mathsf{T}}, \quad \mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

- for a general rank-one update, we can use the *Sherman–Morrison formula*, which we will derive here
- multiplying through the equation $(A + \mathbf{u}\mathbf{v}^{\mathsf{T}})\mathbf{y} = \mathbf{b}$ by A^{-1} yields

$$(I + A^{-1}\mathbf{u}\mathbf{v}^{\mathsf{T}})\mathbf{y} = A^{-1}\mathbf{b} = \mathbf{x}$$

- we therefore need to find $(I + \mathbf{w}\mathbf{v}^{\mathsf{T}})^{-1}$ where $\mathbf{w} = A^{-1}\mathbf{u}$
- we assume that $(I + \mathbf{w}\mathbf{v}^{\mathsf{T}})^{-1}$ is a matrix of the form $(I + \sigma\mathbf{w}\mathbf{v}^{\mathsf{T}})$ where σ is some constant
- from the relationship

$$(I + \mathbf{w}\mathbf{v}^{\mathsf{T}})(I + \sigma\mathbf{w}\mathbf{v}^{\mathsf{T}}) = I$$

  we obtain

$$\sigma\mathbf{w}\mathbf{v}^{\mathsf{T}} + \mathbf{w}\mathbf{v}^{\mathsf{T}} + \sigma\mathbf{w}\mathbf{v}^{\mathsf{T}}\mathbf{w}\mathbf{v}^{\mathsf{T}} = 0$$

- however, the quantity $\mathbf{v}^{\mathsf{T}}\mathbf{w}$ is a scalar, so this simplifies to

$$(\sigma + 1 + \sigma\mathbf{v}^{\mathsf{T}}\mathbf{w})\mathbf{w}\mathbf{v}^{\mathsf{T}} = 0$$

  which yields

$$\sigma = -\frac{1}{1 + \mathbf{v}^{\mathsf{T}}\mathbf{w}}$$

- it follows that the solution $\mathbf{y}$ to the perturbed problem is given by

$$\mathbf{y} = (I + \sigma \mathbf{w}\mathbf{v}^\mathsf{T})\mathbf{x} = \mathbf{x} + \sigma(\mathbf{v}^\mathsf{T}\mathbf{x})\mathbf{w}$$

  and the perturbed inverse is given by

$$\begin{aligned}(A + \mathbf{u}\mathbf{v}^\mathsf{T})^{-1} &= (I + A^{-1}\mathbf{u}\mathbf{v}^\mathsf{T})^{-1}A^{-1} \\ &= \left(I - \frac{1}{1+\mathbf{v}^\mathsf{T}\mathbf{w}}\mathbf{w}\mathbf{v}^\mathsf{T}\right)A^{-1} \\ &= A^{-1} - \frac{1}{1+\mathbf{v}^\mathsf{T}A^{-1}\mathbf{u}}A^{-1}\mathbf{u}\mathbf{v}^\mathsf{T}A^{-1} \qquad (4.1)\end{aligned}$$

  which is the Sherman–Morrison formula
- an efficient algorithm for solving the perturbed problem $(A + \mathbf{u}\mathbf{v}^\mathsf{T})\mathbf{y} = \mathbf{b}$ can therefore proceed as follows:
  - solve $A\mathbf{x} = \mathbf{b}$
  - solve $A\mathbf{w} = \mathbf{u}$
  - compute $\sigma = -1/(1 + \mathbf{v}^\mathsf{T}\mathbf{w})$
  - compute $\mathbf{y} = \mathbf{x} + \sigma(\mathbf{v}^\mathsf{T}\mathbf{x})\mathbf{w}$
- note that we already have the solution to $A\mathbf{x} = \mathbf{b}$ but we have to solve another system $A\mathbf{w} = \mathbf{u}$
- so how is this better than simply solving $(A + \mathbf{u}\mathbf{v}^\mathsf{T})\mathbf{y} = \mathbf{b}$?
- the answer is that if we have LU factorization of A, then solving $A\mathbf{w} = \mathbf{u}$ requires two back solves, which takes $O(n^2)$ operations whereas solving $(A + \mathbf{u}\mathbf{v}^\mathsf{T})\mathbf{y} = \mathbf{b}$ from scratch would require $O(n^3)$ operations
- note that this also works if we have the QR or any other factorizations of A that facilitate solving linear equations involving A
- an alternative approach is to note that

$$\begin{aligned}(A + \mathbf{u}\mathbf{v}^\mathsf{T})^{-1} &= [A(I + A^{-1}\mathbf{u}\mathbf{v}^\mathsf{T})]^{-1} \\ &= (I + \sigma A^{-1}\mathbf{u}\mathbf{v}^\mathsf{T})A^{-1} \\ &= A^{-1} + \sigma A^{-1}\mathbf{u}\mathbf{v}^\mathsf{T}A^{-1}\end{aligned}$$

  which yields

$$\begin{aligned}(A + \mathbf{u}\mathbf{v}^\mathsf{T})^{-1}\mathbf{b} &= A^{-1}(I + \sigma\mathbf{u}\mathbf{v}^\mathsf{T}A^{-1})\mathbf{b} \\ &= A^{-1}(\mathbf{b} + \sigma(\mathbf{v}^\mathsf{T}A^{-1}\mathbf{b})\mathbf{u})\end{aligned}$$

  and therefore we can solve $(A + \mathbf{u}\mathbf{v}^\mathsf{T})\mathbf{y} = \mathbf{b}$ by solving a problem of the form $A\mathbf{x} = \mathbf{b}$ where the right-hand side $\mathbf{b}$ is perturbed

5. RANK-r UPDATE

- what we have in the previous section can be generalized by repeated application of the same technique

$$A + \mathbf{u}_1\mathbf{v}_1^\mathsf{T} + \cdots + \mathbf{u}_r\mathbf{v}_r^\mathsf{T} = A + UV^\mathsf{T} \qquad (5.1)$$

  where $U = [\mathbf{u}_1, \dots, \mathbf{u}_r], V = [\mathbf{v}_1, \dots, \mathbf{v}_r] \in \mathbb{R}^{n\times r}$
- (5.1) is called a *rank-r update* of A
- this is useful if, for example, r entries of A are modified, requiring us to obtain the solution of $(A + UV^\mathsf{T})\mathbf{x} = \mathbf{b}$ from the original solution $A\mathbf{x} = \mathbf{b}$
- the notion of rank-r update is very much related to that of Schur complement

- if we introduce new variables $\mathbf{y} = C\mathbf{x}$, then
$$(A + BC)\mathbf{x} = \mathbf{b}$$
can be written as
$$\begin{cases} A\mathbf{x} + B\mathbf{y} = \mathbf{b} \\ \mathbf{y} = C\mathbf{x} \end{cases} \tag{5.2}$$
or equivalently
$$\begin{bmatrix} A & B \\ C & -I \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix}$$
- in other words $A + BC$ is the Schur complement of $-I$ in $\begin{bmatrix} A & B \\ C & -I \end{bmatrix}$
- we now derive a generalization of the Sherman–Morrison formula (4.1) by solving (5.2)
- plug $\mathbf{x} = A^{-1}(\mathbf{b} - B\mathbf{y})$ into $\mathbf{y} = C\mathbf{x}$ to get
$$(I + CA^{-1}B)\mathbf{y} = CA^{-1}\mathbf{b}$$
and plug the expression $\mathbf{y} = (I + CA^{-1}B)^{-1}CA^{-1}\mathbf{b}$ back into $\mathbf{x} = A^{-1}(\mathbf{b} - B\mathbf{y})$ to get
$$\mathbf{x} = [A^{-1} - A^{-1}B(I + CA^{-1}B)^{-1}CA^{-1}]\mathbf{b}$$
- since $\mathbf{b}$ is arbitrary, this must mean that
$$(A + BC)^{-1} = A^{-1} - A^{-1}B(I + CA^{-1}B)^{-1}CA^{-1} \tag{5.3}$$
- this is called the *Sherman–Woodbury–Morrison formula* and is useful for find rank-r updates of solutions to $A\mathbf{x} = \mathbf{b}$
- a word of caution: both (4.1) and (5.3) should not be used for computing explicit inverse (which is a bad idea in the first place) because they are numerically unreliable