Math 128A Spring 2002 Handout # 28
Sergey Fomel May 7, 2002

Homework 12: Systems of Linear Equations (due on May 14)

---

1. (a) A vector norm $||\mathbf{x}||$ has the following properties
   i. $||\mathbf{x}|| \geq 0$ for all $\mathbf{x} \in \mathbb{R}^n$; $||\mathbf{x}|| = 0$ only if $\mathbf{x} = \mathbf{0}$
   ii. $||\alpha\,\mathbf{x}|| = |\alpha|\,||\mathbf{x}||$ for all $\alpha \in \mathbb{R}$ and $\mathbf{x} \in \mathbb{R}^n$
   iii. $||\mathbf{x}+\mathbf{y}|| \leq ||\mathbf{x}|| + ||\mathbf{y}||$ for all $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^n$

   Prove that these properties are satisfied if a vector norm is defined as

$$||\mathbf{x}||_S = \left(\mathbf{x}^T\,\mathbf{S}\,\mathbf{x}\right)^{1/2}\ , \tag{1}$$

   where $\mathbf{S}$ is a symmetric positive definite matrix.

   (b) A matrix norm $||\mathbf{A}||$ has the following properties
   i. $||\mathbf{A}|| \geq 0$ for all $n \times n$ matrices $\mathbf{A}$; $||\mathbf{A}|| = 0$ only if $\mathbf{A} = \mathbf{0}$
   ii. $||\alpha\,\mathbf{A}|| = |\alpha|\,||\mathbf{A}||$
   iii. $||\mathbf{A}+\mathbf{B}|| \leq ||\mathbf{A}|| + ||\mathbf{B}||$
   iv. $||\mathbf{A}\,\mathbf{B}|| \leq ||\mathbf{A}||\,||\mathbf{B}||$

   Prove that these properties are satisfied for the *Frobenius* norm

$$||\mathbf{A}||_F = \left(\sum_{i=1}^{n}\sum_{j=1}^{n}|a_{ij}|^2\right)^{1/2} \tag{2}$$

2. Triangular matrix ($\mathbf{LU}$) decomposition requires $\frac{n(n-1)(2n-1)}{6}$ multiplications. Triangular matrix inversion ($\mathbf{L}$ or $\mathbf{U}$) requires $\frac{n(n-1)}{2}$ multiplications.

   (a) Find the number of multiplications necessary for solving two linear systems

$$\mathbf{A}\,\mathbf{x} = \mathbf{b}_1\ ,\quad \mathbf{A}\,\mathbf{x} = \mathbf{b}_2 \tag{3}$$

   with the non-singular square matrix $\mathbf{A}$.

   (b) Find the number of multiplications necessary for solving two linear systems

$$\mathbf{A}_1\,\mathbf{x} = \mathbf{b}\ ,\quad \mathbf{A}_2\,\mathbf{x} = \mathbf{b}\ , \tag{4}$$

   where $\mathbf{A}_1$ an $\mathbf{A}_2$ are non-singular square matrices that differ only by one element:

$$\mathbf{A}_2 - \mathbf{A}_1 = \alpha\,\mathbf{e}_i\,\mathbf{e}_j^T\ , \tag{5}$$

   where $\mathbf{e}_i$ is the i-th column of the identity matrix.
   Hint: Recall the Sherman-Morrison formula

$$\left(\mathbf{A}+\mathbf{u}\,\mathbf{v}^T\right)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\,\mathbf{u}\,\mathbf{v}^T\,\mathbf{A}^{-1}}{1+\mathbf{v}^T\,\mathbf{A}^{-1}\,\mathbf{u}}\ . \tag{6}$$

3. The following algorithm can be used for solving the equation

$$\mathbf{U}\,\mathbf{x} = \mathbf{b}\,, \tag{7}$$

where **U** is an upper triangular matrix:

```
UPPER TRIANGULAR(U, x)
for k ← n, n−1, ..., 1
do
    for i ← k+1, k+2, ..., n
    do
        x_k ← x_k − u_{k,i} x_i
    x_k ← x_k / u_{k,k}
```

The algorithm is initialized with the right-hand-side vector **b** and overwrites its elements with the elements of the solution vector **x**. The analogous algorithm for a lower triangular matrix is

```
LOWER TRIANGULAR(L, x)
for k ← 1, 2, ..., n
do
    for i ← 1, 2, ..., k−1
    do
        x_k ← x_k − l_{k,i} x_i
    x_k ← x_k / l_{k,k}
```

Both algorithms access the elements of the corresponding matrices *by row* and use the inner loop to compute the dot product of two vectors. On many computers, the dot product algorithm

```
DOT PRODUCT(a, x, y)
for i ← 1, 2, ..., n
do
    a ← a + x_i y_i
```

is less efficient than the scaled vector addition algorithm

```
ADD SCALED VECTOR(a, x, y)
for i ← 1, 2, ..., n
do
    y_i ← y_i + a x_i
```

because the latter can be easily performed in parallel.

Modify the upper and lower triangular inversion algorithms so that they access the corresponding matrices *by column*. Show that this transforms the dot product algorithm in the inner loop to the scaled vector addition algorithm.

4. (Programming) In this assignment, we revisit the method of least-squares polynomial approximation. Consider a function $f(x)$, measured at a number of points $x_1, x_2, \ldots, x_n$ and approximated with the sum of Chebyshev polynomials

$$f(x_i) \approx \sum_{k=0}^{m} c_k\, T_k(x_i)\,, \quad i = 1, 2, \ldots, n \tag{8}$$

In the matrix form, the system of approximate equations is

$$\mathbf{f} \approx \mathbf{A}\,\mathbf{c}\,, \tag{9}$$

where $\mathbf{f}$ is the vector with elements $f(x_i)$, $\mathbf{c}$ is the vector with the elements c_k, and the $n \times (m+1)$ matrix $\mathbf{A}$ has the elements $a_{ik} = T_k(x_i)$. The method of least squares leads to the square system

$$\mathbf{A}^T\,\mathbf{A}\,\mathbf{c} = \mathbf{A}^T\,\mathbf{f}\,, \tag{10}$$

which can be solved for the unknown coefficient vector $\mathbf{c}$.

The approximation algorithm consists of the following steps:

(a) Form the matrix $\mathbf{A}$.

The algorithm that utilizes the recursive relationship for Chebyshev polynomials is

```
CHEBYSHEV MATRIX(x, A)
for i ← 1, 2, ..., n
do
    a_{i,0} ← 1
    a_{i,1} ← x_i
    for k ← 1, 2, ..., m − 1
    do
        a_{i,k+1} ← 2 x_i a_{i,k} − a_{i,k−1}
```

(b) Form the normal matrix $\mathbf{C} = \mathbf{A}^T\,\mathbf{A}$.

```
NORMAL MATRIX(A, C)
for i ← 0, 1, 2, ..., m
do
    for j ← 0, 1, 2, ..., i
    do
        c_{i,j} ← 0
        for k ← 1, 2, ..., n
        do
            c_{i,j} ← c_{i,j} + a_{k,i} a_{k,j}
```

The algorithm fills only the lower triangular part of $\mathbf{C}$.

(c) Form the right-hand side $\mathbf{b} = \mathbf{A}^T\,\mathbf{f}$

```
RIGHT-HAND SIDE(A, f, b)
for k ← 0, 1, 2, ..., m
do
    b_k ← 0
    for i ← 1, 2, ..., n
    do
        b_k ← b_k + a_{i,k} f_i
```

(d) Cholesky factorization of $\mathbf{C} = \mathbf{L}\,\mathbf{L}^T$

```
CHOLESKY(C)
for k ← 0, 1, 2, ..., m
do
    c_{k,k} = sqrt(c_{k,k})
    for i ← k+1, k+2, ..., m
    do
        c_{i,k} = c_{i,k} / c_{k,k}
    for j ← k+1, k+2, ..., m
    do
        for i ← j, j+1, ..., m
        do
            c_{i,j} = c_{i,j} − c_{i,k} c_{j,k}
```

The algorithm overwrites the lower triangular part of **C** with **L**.

(e) Upper and lower triangular inversion using the Cholesky factor **L** and the right-hand side **b**. The output is the coefficient vector **c**.

(f) At each point x, evaluate the approximation using the fast algorithm from Homework 7:

```
CHEBYSHEV SUM(x, c)
ĉ_1 ← 0
ĉ_0 ← c_n
for k ← n−1, n−2, ..., 0
do
    t ← ĉ_1
    ĉ_1 ← ĉ_0
    ĉ_0 ← c_k + 2 x ĉ_0 − t
return (ĉ_0 − x ĉ_1)
```

Using the data from Homework 5

y	24	28	32	36	48	52	56	60	64	68	72	76	80	84	88	92	94	98
p	8	15	28	6	21	25	14	20	11	12	16	19	28	20	11	25	30	28

where y stands for the year, and p stands for the number of Olympic points, perform the steps above to approximate the dependence $p(y)$. Output the coefficient vector **c** and the Cholesky factor **L** for $m = 1, 2, 4, 8$ and plot the corresponding approximations in the interval $[22, 100]$ with the step size of one year.