ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΠΙΣΤΗΜΟΝΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ Ι

ΤΡΙΤΗ ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ

Καφφέζας Γιώργος · ΑΜ 4465 · kaffezas@ceid.upatras.gr

×

Η αναφορά αυτή αφορά την τρίτη εργαστηριακή άσκηση του μαθήματος «Επιστημονικού Υπολογισμού» για το χειμερινό εξάμηνο του ακαδημαϊκού έτους 2013-2014. Συντάχθηκε με τη βοήθεια του LaTeX και του editor TeXMaker, ενώ το εξώφυλλο βασίστηκε σε κώδικα από την ιστοσελίδα www.latextemplates.com. Επιλέχθηκε πρότυπο βιβλίου και η σελιδοποίηση έγινε ώστε η αναφορά να εκτυπώνεται καλά σε μορφή φυλλαδίου μεγέθους A4.

×

Πληροφορίες Συστήματος

• Τα στοιχεία του υπολογιστικού συστήματος στο οποίο πραγματοποιήθηκε η άσκηση είναι τα ακόλουθα, όπως προκύπτουν και μετά από την χρήση των προγραμμάτων CPU-Z και PC Wizard:

Λειτουργικό σύστημα Windows 7 Professional SP1 (×64)

Τύπος επεξεργαστή Intel Core 2 Quad Q6600 @2.40GHz

Επίπεδα κρυφής μνήμης
L1 data cache: 4 × 32 KB, 8-way set assoc, 64-byte line size
L1 instruction cache: 4 × 32 KB, 8-way set assoc, 64-byte line size
L2 cache: 2 × 4096 KB, 16-way set assoc, 64-byte line size

Πολιτική εγγραφής στην cache write-back

• Η έκδοση MATLAB που χρησιμοποιήθηκε για την άσκηση είναι η R2012b (8.0.0.783) για λειτουργικό σύστημα 64-bit.

ΕΡΩΤΗΜΑ 1 - Κατασκευή Μητρώων με Ορισμένο Δείκτη Κατάστασης

1) Ακολουθώντας τις οδηγίες της εκφώνησης, δημιουργήσαμε την παρακάτω συνάρτηση που παράγει μητρώα με δεδομένο δείκτη κατάστασης (ως προς τη νόρμα-2) και διάσταση:

```
function [A] = condrand_4465(n,kappa)

% Creating 2 random matrices.
X = rand(n);
Y = rand(n);

% Calculating QR for the matrices.
[Q_X,R_X] = qr(X);
[Q_Y,R_Y] = qr(Y);

% Creating the diagonal matrix D.
K = zeros(n,1);
for i=1:n, K(i) = kappa^(-(i-1)/(n-1)); end
D = diag(K);

% Final matrix with same condition number kappa.
A = Q_X*D*Q_Y;

% Display given and calculated condition number.
fprintf('CONDITION NUMBERS\n');
fprintf('--- As input: %d\n',kappa);
fprintf('--- As result: %d\n',cond(A));

end
```

files/1/1/condrand_4465.m

Για να βεβαιωθούμε ότι λειτουργεί σωστά, την εκτελέσαμε μία φορά για $n = 5$ και δείκτη κατάστασης ίσο με 2. Το μητρώο A που προέκυψε είναι το ακόλουθο:

$$A = \begin{pmatrix} 0.1611 & -0.6939 & 0.2524 & 0.1728 & 0.2822 \\ 0.1886 & -0.0661 & 0.0169 & -0.2541 & 0.6651 \\ -0.2018 & -0.1577 & 0.4740 & -0.5909 & -0.1656 \\ 0.6676 & -0.0058 & -0.0844 & -0.2160 & 0.1006 \\ 0.1772 & 0.1579 & 0.5412 & 0.1919 & 0.1426 \end{pmatrix}$$

Ο δείκτης κατάστασής του, σύμφωνα και με το μήνυμα που εκτυπώθηκε, ήταν όντως ίσος με 2, δηλαδή με τον δείκτη που προσδιορίσαμε εμείς κατά την εκτέλεση.

2) Σύμφωνα με τις υποδείξεις της εκφώνησης, χρησιμοποιήσαμε την συνάρτηση που δημιουργήσαμε στο προηγούμενο ερώτημα ώστε να πάρουμε ένα τυχαίο τετραγωνικό μητρώο με δεδομένο δείκτη κατάστασης. Έπειτα, υπολογίσαμε τη μορφή Hessenberg του μητρώου αυτού, όπως προτείνεται. Διαβάζοντας τις βοηθητικές σημειώσεις της ενδογενούς εντολής `hess()` διαπιστώσαμε ότι μπορούμε να τη χρησιμοποιήσουμε για να έχουμε ως αποτέλεσμα ένα τριδιαγώνιο συμμετρικό μητρώο, αλλά αφού εξασφαλίσουμε ότι της δίνουμε ως είσοδο ένα συμμετρικό μητρώο.

Έτσι, χρησιμοποιώντας τη λογική της προηγούμενης συνάρτησης, ακολουθήσαμε τα ίδια βήματα, μόνο που δημιουργήσαμε ένα τυχαίο μητρώο, και χρησιμοποιήσαμε το μητρώο Q που βγαίνει από την παραγοντοποίηση QR και το ανάστροφό του ώστε το μητρώο που θα προκύψει να είναι συμμετρικό, έπειτα από τις πράξεις $Q * D * Q'$.[1] Οπότε, έχοντας πλέον ένα συμμετρικό μητρώο, το δίνουμε ως είσοδο στην `hess()` και έχουμε ως αποτέλεσμα ένα τριδιαγώνιο μητρώο.

Κάποιες τιμές στο άνω μέρος του μητρώου ενδεχομένως να μην είναι μηδενικές, και για το λόγο αυτό κρατήσαμε μόνο τις τρεις κεντρικές διαγώνιους, χωρίς να έχουμε κάποιο ιδιαίτερο σφάλμα στον τελικό μας δείκτη κατάστασης. Ο εν λόγω κώδικας για όλα τα παραπάνω είναι ο εξής:

```
function [E] = st_condrand_4465(n,kappa)

% Creating a random matrix.
X = rand(n);

% Calculating QR for the matrix.
[Q_X,R_X] = qr(X);

% Creating the diagonal matrix D.
K = zeros(n,1);
for i=1:n, K(i) = kappa^(-(i-1)/(n-1)); end
D = diag(K);

% Creating a symmetric matrix using Q_X and D.
A = Q_X*D*Q_X';

% Using the symmetric matrix with hess() in order to
% create a tridiagonal symmetric matrix with
% condition number equal to kappa.
C = hess(A);

% Erasing the very small numbers above the three main diagonals.
E = diag(diag(C,-1),-1) + diag(diag(C)) + diag(diag(C,1),1);

% Printing out the calculated condition number.
fprintf('--- Final condition number: %d\n',cond(E));

end
```

files/1/2/st_condrand_4465.m

Για να βεβαιωθούμε ότι λειτουργεί σωστά, την εκτελέσαμε μία φορά και αυτή για $n = 5$ και δείκτη κατάστασης ίσο με 2. Το τυχαίο μητρώο A και το τελικό τριδιαγώνιο συμμετρικό μητρώο E είναι τα παρακάτω:

$$A = \begin{pmatrix} 0.6647 & -0.0570 & 0.1003 & 0.0578 & 0.0236 \\ -0.0570 & 0.7142 & 0.0151 & 0.1742 & 0.0895 \\ 0.1003 & 0.0151 & 0.7172 & -0.0111 & 0.1153 \\ 0.0578 & 0.1742 & -0.0111 & 0.7802 & 0.0708 \\ 0.0236 & 0.0895 & 0.1153 & 0.0708 & 0.7663 \end{pmatrix}$$

[1] Η σκέψη πίσω από αυτό βασίζεται στο θεώρημα που αναφέρεται και ως 9.16 στη σελίδα 572 του βιβλίου "Numerical Analysis" των Burden και Faires (Cengage Learning, 2010), το οποίο λέει ότι «ένα μητρώο A είναι συμμετρικό αν και μόνο αν υπάρχει ένα διαγώνιο μητρώο D και ένα ορθογώνιο μητρώο Q, τέτοια ώστε $A = QDQ'$».

$$E = \begin{pmatrix} 0.6647 & 0.1312 & 0 & 0 & 0 \\ 0.1312 & 0.6751 & -0.1392 & 0 & 0 \\ 0 & -0.1392 & 0.8173 & 0.1157 & 0 \\ 0 & 0 & 0.1157 & 0.8637 & 0.0791 \\ 0 & 0 & 0 & 0.0791 & 0.6218 \end{pmatrix}$$

Ο δείκτης κατάστασής του, σύμφωνα και με το μήνυμα που εκτυπώθηκε, είναι όντως ίσος με 2, με μια μικρή απόκλιση που βρισκόταν αρκετά κοντά στο eps.