Επιστημονικός Υπολογισμός Ι
Εργαστηριακή 'Ασκηση

Ημερομηνία κατάθεσης γιά πλήρη βαθμό ως τις 23:59 στις 4/2/2017

Εισαγωγή

Προσοχή, υπάρχουν επιπλέον ζητούμενα για τους εισαχθέντες πριν το 2014. Για διευκρινίσεις, απορίες, κ.λπ. στα Φροντιστήρια και ώρες γραφείου ή σε άλλες ώρες μετά από συνεννόηση.

Προκαταρκτικά Να περιγράψετε[1] σε μια παράγραφο τα χαρακτηριστικά του συστήματος που χρησιμοποιείτε (υλικό και λογισμικό):

1. Σύστημα, επεξεργαστή, συχνότητα, επίπεδα μνήμης και μέγεθός τους και ύπαρξη ή όχι FMA.
2. Την έκδοση του συστήματος MATLAB που χρησιμοποιείτε.

1 Μέρος Α

Έχουμε πει στις διαλέξεις ότι η χρήση ακόμα και της μερικής οδήγησης μπορεί να είναι ιδιαίτερα κοστοβόρα καθώς απαιτεί εύρεση του μέγιστου στοιχείου σε μία στήλη κάθε φορά. Στη συνέχεια θα προσπαθήσουμε να την αποφύγουμε χρησιμοποιώντας μια τεχνική που θα ονομάσουμε (ίσως όχι και πολύ πετυχημένα) "διαγώνια τόνωση" ή απλά "τόνωση". Η μέθοδος περιγράφτηκε για πρώτη φορά από τον G.W. Stewart [1].

Η ιδέα είναι απλή: Όποτε εμφανίζεται στοιχείο με μικρή τιμή ως οδηγός, εφαρμόζουμε "τόνωση" που σημαίνει αλλάζουμε την τιμή του με μία άλλη που είναι αρκούντως μεγάλη. Εδώ θα κάνουμε την τόνωση προσθέτοντας (ομόσημη) τιμή ώστε η τελική (απόλυτη) τιμή του διαγώνιου στοιχείου να είναι μεγαλύτερη από κάποιο κατώφλι.

Ένας βασικός στόχος της άσκησης είναι να δημιουργήστε συνάρτηση MATLAB η οποία να δέχεται ως είσοδο το μητρώο A και ένα ή περισσότερα δεξιά μέλη B και επιστρέφει X όπου το X σε αριθμητική άπειρης ακρίβειας ικανοποιεί την εξίσωση μητρώων $AX = B$. Επιπλέον θα εξετάσετε την επίδοση της μεθόδου από άποψη ταχύτητας, αποτελεσματικότητας και ακρίβειας και θα εξετάσετε την εφαρμογή της.

1. Να κατεβάσετε από το διαδίκτυο το Matrix Computation Toolbox του N. Higham (στο διαδίκτυο, π.χ. από το File Exchange της Mathworks).
2. Να ελέγξετε χρησιμοποιώντας μητρώα από τις παραπάνω συλλογές καθώς και τυχαία μητρώα ότι γενικά ισχύει η "σχέση τάξης" που προκύπτει από τον τύπο Sherman-Morrison-Wodbury (SMW). Ειδικότερα, να επαληθεύσετε ότι αν $A \in \mathbb{R}^{n\times n}$, και $\text{rank}(WH^\top) = t$, όπου $W, H \in \mathbb{R}^{n\times t}$ τότε ισχύει

$$\text{rank}((A + WH^\top)^{-1} - A^{-1}) = t. \tag{1}$$

   Ειδικότερα:

[1] Γιατί εσείς και εμείς πρέπει να γνωρίζουμε πού τρέχουν τα πειράματα - προσοχή, αυτό σημαίνει ότι πρέπει να αναφέρετε μετρήσεις από ένα μόνο σύστημα!

(α′) Να επαληθεύσετε τη σχέση (1) όταν $A = \mathrm{randn(n)}$, $W = \mathrm{randn(n,t)}, H = \mathrm{randn(n,t)}$ και n=128, 512, t=1, 2, 10. Να παρουσιάσετε τα αποτελέσματά σας σε πίνακα σε κάθε γραμμή του οποίου οι πρώτες στήλες αναφέρουν τις διαστάσεις n,t και την τάξη (rank) της διαφοράς.

(β′) Όπως και προηγουμένως αλλά αυτή τη φορά για το μητρώο $A = \mathrm{tril}\ (\mathrm{randn(n)})$.

(γ′) Να βρείτε περιπτώσεις αντιστρέψιμων μητρώων για τα οποία η παραπάνω σχέση (1) δεν ισχύει σε αριθμητική κινητής υποδιαστολής και να εξηγήσετε τι συμβαίνει.

(δ′) Να διερευνήσετε και να εξηγήσετε κατά πόσον η μέθοδος διορθώνει ή όχι την ενδεχόμενη αστάθεια της LU για το μητρώο gfpp(10).

3. Να υλοποιήσετε τη μέθοδο LU με τόνωση (αντί για οδήγηση) όπως την περιγράψαμε στην τάξη και να την γράψετε ως συνάρτηση με όνομα luboost χρησιμοποιώντας για μοντέλο τη συνάρτηση gep από το MC Toolbox. Η συνάρτησή σας πρέπει να παίρνει για είσοδο το A και να επιστρέφει τα L, U που προκύπτουν από την εφαρμογή διαδοχικών τονώσεων και μετασχηματισμών Gauss για την αποφυγή οδήγησης (όποτε κρίνεται απαραίτητο) και την άνω τριγωνοποίηση του A. Προσέξτε ότι αντί για την κλασική LU στην οποία υπολογίζονται οι παράγοντες L_j, U, P_j από τη διαδικασία $L_{n-1}P_{n-1}\cdots L_1P_1A = U$ όπου P_j είναι μητρώα εναλλαγής, L_j μετασχηματισμοί Gauss και U άνω τριγωνικό, η μέθοδος τόνωσης υπολογίζει

$$L_{n-1}T_{n-1}L_{n-2}T_{n-2}\cdots L_1T_1A = U$$

όπου $T_j, L_j, j = 1, ..., n-1$ είναι τα μητρώα τόνωσης και οι μετασχηματισμοί Gauss. Προσέξτε επίσης ότι όπως και στην LU με μερική οδήγηση, ορισμένα ή όλα τα T_j μπορεί να είναι το ταυτοτικό μητρωο. Σε αντίθεση όμως με την LU με μερική οδήγηση, τα υπόλοιπα T_j δεν είναι μητρώα εναλλαγής αλλά έχουν άλλη μορφή που πρέπει να προσδιορίσετε στα πλαίσια της άσκησης.

4. Να δείξετε ότι τα μητρώα L_j, U που προκύπτουν παραπάνω γενικά ικανοποιούν την εξής αριθμητική σχέση

$$\text{rank}(LU - A) = t$$

όπου t είναι το πόσες φορές έγινε τόνωση (μη τετριμμένη) και $L = L_1^{-1}\cdots L_{n-1}^{-1}$.

5. Να αποδείξετε αλγεβρικά ότι ισχύει η παραπάνω σχέση και να υπολογίσετε ποιό θα είναι το μητρώο $LU - A$ συναρτήσει των L_j και του A.

6. Να γράψετε κώδικα για την πλήρη επίλυση του $AX = B$ χρησιμοποιώντας την επιλογή της τόνωσης. Ονομάστε τον κώδικά σας geboost. Επιλέξτε την τόνωση έτσι ώστε για τα τυχαία μητρώα παραπάνω να μην προκύψει οδηγός μικρότερος από 10^{-2}.

7. Να υπολογίσετε το κόστος του υπολογισμού των παραγόντων L, U με τόνωση ως συνάρτηση του μεγέθους του μητρώου και ου πλήθους t των εφαρμογών της τόνωσης. Πώς συγκρίνεται αυτό με το κόστος της LU?

8. Να εξηγήσετε κατά πόσον θα μπορούσατε να οργανώστε όλα τα βήματα της τόνωσης από την αρχή, ανεξαρτήτως του μητρώου και κατά πόσον αυτό είναι πρακτικό ή όχι?

2 Μέρος Β

Στη συνέχεια θα προγραμματίσετε επίλυση συστημάτων με χρήση επαναληπτικής εκλεπτυνσης που όπως είδαμε στην τάξη έχει ενδιαφέρον λόγω των ιδιοτήτων των GPUs που παρέχουν πολύ μεγαλύτερη ταχύτησα για μονή ακρίβεια. Το ερώτημα για το μέρος αυτό της άσκησης όμως είναι λίγο διαφορετικό και αφορά στην εφαρμογή της ιδέας της Newton για την επίλυση του συστήματος $(A + WH^\top)x = b$ όπου το A είναι ένα αραιό μητρώο για το οποίο είναι εύκολη η επίλυση. Επίσης τα W, H είναι μητρώα με t στήλες το καθένα, οπου το t είναι μικρό σε σχέση με το μέγεθος του .

1. Να τροποποιήσετε τον κώδικα της επαναληπτικής εκλέπτυνσης ώστε να λύνει το παραπάνω σύστημα $(A+WH^\top)x = b$ όπου W, H τάξης 1 χρησιμοποιώντας αποκλειστικά αριθμητική διπλής ακρίβειας. Στόχος είναι να προγραμματίσετε τη μέθοδο ώστε να μειώσετε όσο μπορείτε το κόστος ανά βήμα και βέβαια το συνολικό Ω. Ειδικότερα, αν $E = WH^\top$ η μέθοδός σας θα βασίζεται σε μία αναδρομή
$$Ax^{(k+1)} = -Ex^{(k)} + b$$
Ο κώδικας πρέπει να τρέχει αρκετές επαναλήψεις μέχρις ότου το κατάλοιπο να γίνει μικρότερο από κάποια προκαθορισμένη τιμή ώστε το απόλυτο σφάλμα να είναι εξασφαλισμένα μικρότερο από 10^{-6}. Αν αυτό δεν επιτυγχάνεται, πρέπει να επιστρέφεται μήνυμα (ότι υπήρξε αστοχία επαρκούς σύγκλισης).

2. Να διατυπώσετε μία συνθήκη που εξασφαλίζει τη σύγκλιση της μεθόδου.

3. Έστω τώρα ότι το μητρώο A είναι `toeplitz ([h,-1, zeros(1,n-2)])` και W, H τα `W = randn(n,1); ... W = W/norm(W); H = W(n:-1:1);` Να χρησιμοποιήσετε τον παραπάνω κώδικα και να μελετήσετε την επίλυση (αν συγκλίνει και πόσο γρήγορα σε χρόνο και επαναλήψεις εκλέπτυνσης) για τις τιμές $h = 2, h = 3, h = 4$ και $n = 10, 100, 1000$ (όλους τους συνδυασμούς). Να εξηγήσετε κατά πόσον θα συγκλίνει με βάση τη θεωρία και κατά πόσον οποιαδήποτε αστοχία προβλέπεται από τη θεωρία για τα παραπάνω δεδομένα.

4. **Το τμήμα αυτό είναι υποχρεωτικό μόνο για τους εισαθέντες προ 2014. Θα πρέπει πρώτα να έχετε ολοκληρώσει τα προηγούμενα υποερωτήματα.**

   (α΄) Να γράψετε τον κώδικα επίλυσης του συστήματος χρησιμοποιώντας τον (άμεσο) τύπο SMW. Να μελετήσετε το χρόνο επίλυσης και την ακρίβεια που πετυχαίνει και να συγκρίνετε με την προηγούμενη διαδικασία για όλους τους παραπάνω συνδυασμούς διαστάσεων και τιμών h.

   (β΄) Να γράψετε κώδικα που να συνδυάζει τον τύπο SMW με την κλασική επαναληπτική εκλέπτυνση για να βελτιώσετε τα αποτελέσματα της SMW. Δηλαδή, κατ΄αρχήν ο κώδικάς σας θα εφαρμόζει επαναληπτική εκλέπτυνση για να βελτιώσει τη λύση που προκύπτει κάθε φορά με το μητρώο $A + WH^\top$ το οποίο θα επιλύετε με SMW χρησιμοποιώντας μονή ακρίβεια μόνον για την παραγοντοποίηση και κάθε επίλυση με το A. Δείτε μία σκιαγράφηση του κώδικα στον πίνακα 1. Επομένως, όποτε στην εκλέπτυνση απαιτείται να λύσετε με το $A + WH^\top$, εσείς θα πρέπει να χρησιμοποιείτε SMW η οποία θα παραγοντοποιεί και θα επιλύει (όποτε απαιτείται) με το A αποκλειστικά με *μονή ακρίβεια*. Η διαδικασία πρέπει να επαναλαμβάνετε ως τη σύγκλιση σε κάποιο προκαθορισμένο κατώφλι σφάλματος. Να αξιολογήσετε την επίδοση (χρόνους επίλυσης και ακρίβεια) για τους παραπάνω συνδυασμούς διαστάσεων και παραμέτρων με κατώφλι σύγκλισης το 10^{-6} για το σφάλμα.

Πίνακας 1: Αλγόριθμος επαναληπτικής εκλέπτυνσης. Το ((s)) δηλώνει μονή (απλή) ακρίβεια. Προσοχή, η διατύπωση του κώδικα δεν υπονοεί τη βέλτιστη υλοποίηση σε πλήθος πράξεων .

1. (SMW με (s) για επιλύσεις με το A)	$x^{(0)} = (A + WH^{\top})^{-1}b$
2.	$k = 1$
3.	**repeat**
4.	$r^{(k)} = b - (A + WH^{\top})x^{(k-1)}$
5. (SMW με (s) για επιλύσεις με το A)	$z^{(k)} = (A + WH^{\top})^{-1}r^{(k)}$
6.	$x^{(k)} = x^{(k-1)} + z^{(k)}$
7.	$k = k + 1$
8.	**until** convergence

3 Μέρος Γ

Στο τμήμα αυτό της άσκησης θα ασχοληθείτε με την επίλυση συστημάτων με ειδική μπλοκ μορφή. Ειδικότερα, είδαμε στις διαλέξεις ότι η επίλυση συστήματος με μητρώο που είναι σε μορφή βέλους με ”βορειοδυτική” (ΒΔ) φορά έχει το μειονέκτημα ότι οδηγεί σε πυκνούς παράγοντες L, U που μπορεί να είναι μεγάλο μειονέκτημα. Το πρόβλημα λύνεται με επαναρίθμηση ώστε το μητρώο να έχει δομή βέλους με ”νοτιανατολική” (ΝΑ) φορά. Εδώ θα εξετάσετε την περίπτωση που το αρχικό μητρώο έχει μορφή βέλους αλλά το κάθε στοιχείο είναι και αυτό τετραγωνικό μητρώο $m \times m$ (δηλ. έχουμε ”μπλοκ βέλη” με ΒΔ ή ΝΑ φορά). Συνολικά το μητρώο είναι $mn \times mn$. Θα πρέπει να επινοήσετε αλγόριθμο και να γράψετε κώδικα που αντί να λύνει το γραμμικό σύστημα με το μητρώο σε μορφή ΒΔ βέλους, να λύνει το ισοδύναμο πρόβλημα με μητρώο με μορφή ΝΑ βέλους. Στη συνέχεια να αξιολογήσετε την επίδοση των μεθόδων.

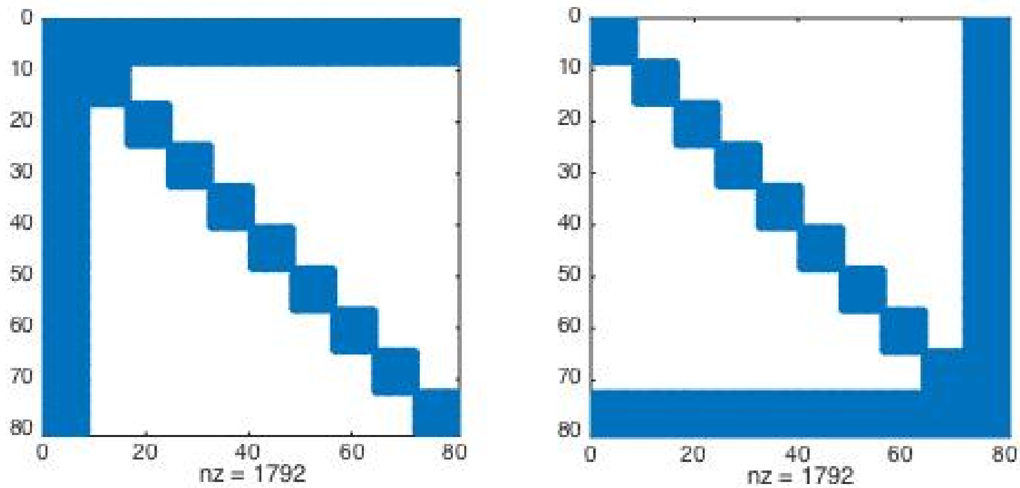
1. Στο πρώτο βήμα θα γράψετε κώδικα για να κατασκευάσετε τις τιμές του αρχικού ΒΔ μητρώου. Θα πρέπει πρώτα να ”προσωποποιήσετε” το μητρώο ως εξής:

   Θα πάτε στην ιστοσελίδα `https://goo.gl/wycKd8` και θα εισάγετε τα 4 τελευταία ψηφία του ΑΜ σας. Μετά θα ζητήσετε να γίνει ο υπολογισμός, θα δείτε την απάντηση που επιστρέφεται ως SHA-1 και από αυτήν θα επιλέξετε τα 4 *πρώτα αριθμητικά ψηφία με τη σειρά που εμφανίζονται*. Θεωρήστε αυτά ως την τιμή μίας μεταβλητής myHash. Στη συνέχεια να κατασκευάσετε το μπλοκ $T \in \mathbb{R}^{m \times m}$ με τις εντολές rng(myHash); T=full(sprand(m,m,0.6)); Ενδιαφέρουν οι ακόλουθες τιμές των (m, n): $(4, 250)$, $(8, 125)$, $(8, 250)$, $(16, 125)$ (προφανώς ως εδώ χρειάζεστε μόνον τις τιμές του $m = 4, 8, 16$).

   Προσοχή, για να βαθμολογηθείτε είναι απαραίτητο να ακολουθήσετε τα βήματα ακριβώς όπως τα περιγράψαμε και να χρησιμοποιήσετε την σωστή τιμή του myHash.

   Στη συνέχεια, θα γράψετε μία συνάρτηση arrowNW(T,n) που θα παίρνει για είσοδο το παραπάνω T και την τιμή του n και θα κατασκευάζει μητρώα $A \in \mathbb{R}^{mn \times mn}$ με μορφή ΒΔ βέλους βάσει του T και ενός μητρώου $D = T + mI$ όπου I το ταυτοτικό μεγέθους m. Για παράδειγμα, αν $n = 5$, τότε:

$$A = \begin{pmatrix} D & T & T & T & T \\ T & D & 0 & 0 & 0 \\ T & 0 & D & 0 & 0 \\ T & 0 & 0 & D & 0 \\ T & 0 & 0 & 0 & D \end{pmatrix}, \text{όπου } D = T + mI \text{ και } 0 \text{ το μηδενικό μητρώο.}$$



Σχήμα 1: Σύνθετο μητρώο με μορφή μπλοκ βέλους με ΒΔ φορά (αριστερό πάνελ) και με ΝΑ φορά (δεξιό πάνελ). Και στις δύο περιπτώσεις, $m = 8, n = 10$.

Προσπαθήστε να γράψετε τη συνάρτηση με τρόπο που να μειώνει όσο γίνεται τους βρόχους. Για το σκοπό αυτό θα βρείτε χρήσιμη την εντολή kron της MATLAB. Το μητρώο θα είναι αποθηκευμένο σε πυκνή μορφή.

2. Χρησιμοποιώντας την εντολή spy να απεικονίσετε σε σχήμα το μητρώο A=arrowNW(T,n) όπου για $m = 4, n = 20$ και να το συμπεριλάβετε στην αναφορά σας.

3. Να υπολογίσετε τους κάτω και άνω τριγωνικούς παράγοντες L, U που προκύπτουν από την LU της MATLAB για το παραπάνω μητρώο και να τους απεικονίσετε με τον ίδιο τρόπο στην αναφορά σας.

4. Να ελέγξετε το γέμισμα (fill-in) που δημιουργείται συγκρίνοντας το πλήθος των μη μηδενικών του A (συμβ. ως nnz(A)) με τα μη μηδενικά των L και U μείον mn (για να μην ληφθεί η διαγώνιος

υπόψη 2 φορές). Να το ποσοτικοποιήσετε ως

$$(\text{nnz}(L) + \text{nnz}(U) - mn)/\text{nnz}(A).$$

5. Στόχος τώρα είναι να μειώσετε αυτό το γέμισμα. Βάσει της συζήτησής μας σχετικά με τα μητρώα σε μορφή βέλους, να προτείνετε ένα μητρώο μετάθεσης W τέτοιο ώστε το μητρώο $B = WAW^\top$ να έχει πολύ μικρότερο γέμισμα στους αντίστοιχους παράγοντες LU. Ειδικότερα, να απεικονίσετε τα μητρώα B, L, U που αντιστοιχούν στο νέο σύστημα, να μετρήσετε όπως πριν το γέμισμα και να συγκρίνετε με την προηγούμενη περίπτωση.

6. Χάριν της άσκησης να θέσετε e = ones(m*n,1); b = A*e. Στη συνέχεια να υπολογίσετε με σωστό τρόπο (όπως έχουμε συζητήσει στα πλαίσια της timeit) το μέσο χρόνο για την επίλυση του $Ax = b$ και αντίστοιχα του ισοδύναμου $WAW^\top Wx = Wb$. Σχολιάστε τις επιδόσεις της μεθόδου στις δύο περιπτώσεις για τους παραπάνω συνδυασμούς των τιμών των (m, n), δηλ. $(4, 250)$, $(8, 125)$, $(8, 250)$, $(16, 125)$. Σε όλες τις περιπτώσεις να αναφέρετε και την τιμή του σχετικού σφάλματος για το x που υπολογίζει ο κώδικάς σας, δηλ. το $\|e - x\|_\infty / \|e\|_\infty$ (πρέπει να είναι πολύ μικρό).

7. Για όλους τους συνδυασμούς διαστάσεων να μετατρέψετε τα μητρώα σε αραιά, δηλ. sparse(A) και sparse(B), να τρέξετε τις χρονομετρήσεις και να αξιολογήσετε τις επιδόσεις σε σύγκριση με πριν αλλά και μεταξύ τους.

4 Μέρος Δ (μόνον για εισαχθέντες πριν το 2014)

Θέλουμε να εξετάσετε κατά πόσον οι χρόνοι εκτέλεσης των εντολών `X = qr(A)` και `[Q,R]=qr(A)` ακολουθούν την κυβική πολυπλοκότητα που προβλέπεται για το Ω της κάθε μίας (π.χ. δείτε στις σημειώσεις και διαφάνειες). Για να γίνει αυτό, πρέπει να κατασκευάσετε για κάθε εντολή, αντίστοιχη (μαθηματική) συνάρτηση, έστω $T_X(n)$ για την πρώτη και $T_{QR}(n)$ για τη δεύτερη, που μοντελοποιεί το χρόνο εκτέλεσης από μετρήσεις που θα διεξάγετε με τυχαία μητρώα μεγέθους $n \times n$ όπου το n λαμβάνει τιμές `n = [200:200:1400]`. Θα κατασκευάσετε κάθε συνάρτηση με την εντολή `polyfit` της MATLAB. Όπου ζητούνται χρονομετρήσεις, πρέπει να γίνονται προσεκτικά (όπως υποδείξαμε στην 1η άσκηση του εξαμήνου). Οπότε:

1. Χρησιμοποιώντας την `polyfit` να βρείτε κυβική συνάρτηση (δηλ. $\alpha_3 n^3 + \alpha_2 n^2 + \alpha_1 n + \alpha_0$) για κάθε εντολή που να ταιριάζει (να ελαχιστοποιεί το τετράγωνο του σφάλματος, όπως κάνει η `polyfit`) στις χρονομετρήσεις σας (δηλ. ένα σετ συντελεστών α για κάθε συνάρτηση).

2. Θεωρώντας ότι δεν γνωρίζετε τους χρόνους εκτέλεσης, να υπολογίσετε τις τιμές που προβλέπονται για τους χρόνους απο τις παραπάνω (κυβικές) συναρτήσεις. Αυτό να το κάνετε για n ίσο με τις παραπάνω τιμές (δηλ. `[200:200:1400]`) καθώς και για τις τιμές `[250:200:1750]`.

3. Στη συνέχεια να εξετάσετε την ακρίβεια της κάθε πρόβλεψης χρησιμοποιώντας α) τους χρόνους που υπολογίσατε στο πρώτο μέρος για `[200:200:1400]`) και β) χρονομετρώντας τις εντολές για `n = [250:200:1750]`. Να οπτικοποιήσετε τα αποτελέσματα σε μία γραφική παράσταση (για κάθε εντολή) η οποία θα περιέχει: *i*) τις χρονομετρήσεις για όλα τα παραπάνω n (με σύμβολο `+' χωρίς σύνδεση με γραμμή). *ii*) τις τιμές που προβλέπονται για το χρόνο εκτέλεσης κάθε συνάρτησης από τις κυβικές συναρτήσεις (με διακεκομμένη γραμμή).

4. Να επαναλάβετε το 1ο υποερώτημα παραπάνω ζητώντας από την polyfit να κατασκευάσει πολυώνυμα 2ου βαθμού και 4ου βαθμού. Να σχολιάσετε τις προβλέψεις τους με τα κυβικά πολυώνυμα.

ΣΥΣΤΑΣΗ 1): Μπορεί να έχετε πιο αξιόπιστα αποτελέσματα στην polyfit αν την καλέσετε ως [p,s,mu]=polyfit(x,y,deg) όπου το deg είναι ο βαθμός του επιθυμητού πολυωνύμου (π.χ. 3, 2,4). Δείτε help polyfit για λεπτομέρειες εφαρμογής.

5 Οδηγίες

Τα παραδοτέα (αποκλειστικά ηλεκτρονικά):

Αναφορά Σε μορφή pdf με σύνθετο όνομα τύπου ΕΕΙΣΓ_ΑΜ_ΕΠΙΘΕΤΟ.pdf δηλ. αποτελούμενο από το έτος εισαγωγής σας, τα τελευταία 4 ψηφία του ΑΜ σας, και το επίθετό σας με λατινικούς χαρακτήρες πρώτο γράμμα κεφαλαίο και τα υπόλοιπα πεζά. Για παράδειγμα 2001_9999_Gallopoulos.pdf.

Να είστε ιδιαίτερα προσεκτικοί ώστε η αναφορά να είναι αναγνώσιμη χωρίς πρόβλημα συμβατότητας των γραμματοσειρών κ.λπ. Ιδιαίτερη σημασία θα δοθεί στον τρόπο και στην οργάνωση της παρουσίασης.

Κώδικας: Τα προγράμματα που απορρέουν από την εργασία σας.

Κατάθεση στο eClass: Η αναφορά καθώς και τα όποια συνοδευτικά αρχεία θα πρέπει να αναρτηθούν στο eClass σε συμπιεσμένη μορφή (μόνο zip) με σύνθετο όνομα ακριβώς όπως και το pdf μόνον που η κατάληξη θα είναι zip.

Αναφορές

[1] G.W. Stewart. Modifying pivot elements in Gaussian elimination. *Math.Comp.*, 28(126):537–542, 1974.