



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΟΜΕΑΣ ΜΑΘΗΜΑΤΙΚΩΝ

Μέθοδος Ελαχίστων Τετραγώνων:
Παραγοντοποίηση QR και Υπολογισμοί

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

της

ΑΘΗΝΑΣ ΠΑΠΑΤΡΙΑΝΤΑΦΥΛΛΟΥ

Επιβλέπων: Κωνσταντίνος Χρυσάφινος

Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2024



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών
Τομέας Μαθηματικών

Μέθοδος Ελαχίστων Τετραγώνων: Παραγοντοποίηση QR και Υπολογισμοί

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

της

ΑΘΗΝΑΣ ΠΑΠΑΤΡΙΑΝΤΑΦΥΛΛΟΥ

Επιβλέπων: Κωνσταντίνος Χρυσάφινος
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

(Υπογραφή)

(Υπογραφή)

(Υπογραφή)

.....
Κωνσταντίνος Χρυσάφινος
Καθηγητής Ε.Μ.Π.

.....
Εμμανουήλ Γεωργούλης
Καθηγητής Ε.Μ.Π.

.....
Βασίλειος Κοκκίνης
Αναπλ. Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2024



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών
Τομέας Μαθηματικών

Copyright © – All rights reserved. Με την επιφύλαξη παντός δικαιώματος.

Αθηνά Παπατριανταφύλλου, 2024.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα.

Το περιεχόμενο αυτής της εργασίας δεν απηχεί απαραίτητα τις απόψεις του Τμήματος, του Επιβλέποντα ή της επιτροπής που την ενέκρινε.

Υπεύθυνη Δήλωση

Βεβαιώνω ότι είμαι συγγραφέας αυτής της διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης, βεβαιώνω ότι αυτή η διπλωματική εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τις απαιτήσεις του προπτυχιακού προγράμματος της Σχολής Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών του Εθνικού Μετσόβιου Πολυτεχνείου.

(Υπογραφή)

.....
Αθηνά Παπατριανταφύλλου

Περίληψη

Η παρούσα εργασία παρουσιάζει μια μελέτη σχετικά με την εφαρμογή της παραγοντοποίησης QR στην επίλυση προβλημάτων ελαχίστων τετραγώνων, έναν βασικό τομέα των εφαρμοσμένων μαθηματικών με σημαντικές εφαρμογές στην ανάλυση δεδομένων, τις αριθμητικές μεθόδους και την υπολογιστική αποδοτικότητα.

Στην εργασία γίνεται διερεύνηση της μεθόδου των ελαχίστων τετραγώνων, η οποία είναι κομβικής σημασίας για την προσέγγιση λύσεων υπερκαθορισμένων συστημάτων γραμμικών εξισώσεων. Ελαχιστοποιώντας το άθροισμα των τετραγώνων των υπολοίπων, η προσέγγιση των ελαχίστων τετραγώνων παρέχει βέλτιστες προσαρμογές για διάφορους τύπους δεδομένων, καθιστώντας την ένα κρίσιμο εργαλείο σε τομείς που κυμαίνονται από τη μηχανική έως την οικονομία.

Στη συνέχεια, εμβαθύνουμε στην παραγοντοποίηση QR. Η τελευταία συμβάλλει στη βελτίωση της αριθμητικής σταθερότητας και της υπολογιστικής απόδοσης της επίλυσης προβλημάτων ελαχίστων τετραγώνων.

Επιπλέον, η εργασία εξετάζει διάφορους αλγορίθμους για την παραγοντοποίηση QR, συμπεριλαμβανομένης της διαδικασίας Gram-Schmidt, των ανακλάσεων Householder και της SVD, συγκρίνοντας τις υπολογιστικές τους πολυπλοκότητες και την καταλληλότητά τους για διαφορετικούς τύπους προβλημάτων.

Η εργασία αυτή όχι μόνο ενισχύει τη σημασία αυτών των μαθηματικών τεχνικών, αλλά υπογραμμίζει τη σημασία τους για την αντιμετώπιση των σύγχρονων προκλήσεων σε διάφορους επιστημονικούς και μηχανολογικούς τομείς.

Η μελέτη και η παρουσίαση του υλικού, βασίζεται κυρίως στο βιβλίο των David Trefethen Lloyd N. and Bau, με τίτλο «Numerical Linear Algebra».

Abstract

This thesis presents a comprehensive study on the application of QR decomposition in solving least squares problems, an essential area in applied mathematics with significant implications for data analysis, numerical methods, and computational efficiency.

The research begins with an exploration of the least squares method, which is pivotal for approximating solutions to overdetermined systems of linear equations. By minimizing the sum of the squares of residuals, the least squares approach provides optimal fits for various types of data, making it a critical tool in fields ranging from engineering to economics.

Subsequently, the study delves into QR decomposition. This decomposition is instrumental in improving the numerical stability and computational efficiency of solving least squares problems.

Additionally, the thesis discusses various algorithms for QR decomposition, including the Gram-Schmidt process, Householder reflections, and SVD, comparing their computational complexities and suitability for different types of problems.

This work not only reinforces the importance of these mathematical techniques but also underscores their relevance in addressing contemporary challenges in various scientific and engineering domains.

The study and writing of the material is based mainly on the book by David Trefethen Lloyd N. and Bau, entitled "Numerical Linear Algebra".

Ευχαριστίες

Με το πέρας της διπλωματικής μου εργασίας, θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες προς τον Καθηγητή Κωνσταντίνο Χρυσάφινο, για την καθοδήγηση και την υποστήριξή του, στην εξέλιξη και ολοκλήρωση της εργασίας μου. Κατά τη διάρκεια της συνεργασίας μας, υπήρξε αρωγός και σύμβουλος.

Επιπλέον, θα ήθελα να ευχαριστήσω, από τα βάθη της καρδιάς μου, την οικογένειά μου, που με βοηθούν και με στηρίζουν με κάθε τρόπο, στεκόμενοι δίπλα μου σε κάθε βήμα.

Τέλος, ευχαριστώ τους φίλους μου, για τη συμπαράσταση, την ενθάρρυνση και την υπομονή τους. Η στήριξή τους ήταν πολύτιμη και με βοήθησε να συνεχίσω σε δύσκολες στιγμές.

Περιεχόμενα

Περίληψη	vii
Abstract	ix
Ευχαριστίες	xi
1 Εισαγωγή	1
1.1 Εισαγωγικές Έννοιες	1
1.2 Νόρμες	3
1.3 Ορθογώνια Διανύσματα	5
1.4 Ανάλυση πίνακα σε ιδιάζουσες τιμές - SVD	6
1.5 Προβολές	16
2 Παραγοντοποίηση QR	21
2.1 Ορθοκανονικοποίηση Gram-Schmidt	21
2.2 Παραγοντοποίηση QR	23
2.3 Householder	26
3 Μέθοδος Ελαχίστων Τετραγώνων	33
3.1 Ανασκόπηση	33
3.2 Μαθηματική διατύπωση	34
3.3 Η γεωμετρία των ελαχίστων τετραγώνων	36
3.4 Προσαρμογή πολυωνυμικών δεδομένων	37
3.5 Κανονικές εξισώσεις, QR και SVD	41
3.6 Ευστάθεια προβλήματος ελαχίστων τετραγώνων	51
4 Μερικές Εφαρμογές της Μεθόδου Ελαχίστων Τετραγώνων	53

4.1	Εφαρμογές στην ανάλυση παλινδρόμησης	53
4.2	Μη γραμμικά ελάχιστα τετράγωνα	54
4.3	Πρακτικά παραδείγματα	55
4.4	Συμπεράσματα	65
	Βιβλιογραφία	68

Κεφάλαιο **1**

Εισαγωγή

1.1 Εισαγωγικές Έννοιες

Αρχικά θα ορίσουμε μερικές βασικές έννοιες για τους πίνακες, θεωρώντας ότι $b = Ax$, είναι ένας γραμμικός συνδυασμός των στηλών του A . Οι έννοιες αυτές είναι απαραίτητες, ώστε να κατανοήσουμε τις μεθόδους που θα αναλύσουμε στη συνέχεια.

Ορισμός

Έστω x ένα n -διάστατο διάνυσμα στήλη και A ένας $m \times n$ πίνακας (m γραμμές, n στήλες). Τότε το γινόμενο πίνακα-διανύσματος $b = Ax$ είναι ένα m -διάστατο διάνυσμα στήλη που ορίζεται ως:

$$b_i = \sum_{j=1}^n a_{ij}x_j, \quad i = 1, \dots, m. \quad (1.1)$$

Εδώ το b_i δηλώνει την i -οστή συντεταγμένη του b , το a_{ij} δηλώνει το i, j στοιχείο του A (i γραμμή, j στήλη) και το x_j δηλώνει την j -οστή συντεταγμένη του x .

Η απεικόνιση $x \mapsto Ax$ είναι γραμμική, που σημαίνει ότι για κάθε $x, y \in \mathbb{R}^n$ (ή \mathbb{C}^n) και για κάθε $a \in \mathbb{R}$ (ή \mathbb{C})

$$A(x + y) = Ax + Ay$$

$$A(ax) = aAx$$

Παρατήρηση

Αν το a_j συμβολίζει την j -ιοστή στήλη του A , τότε η (1.1) μπορεί να γραφτεί ως:

$$b = Ax = \sum_{j=1}^n x_j a_j. \quad (1.2)$$

Εύρος Πίνακα

Το εύρος του πίνακα A , γράφεται ως $\text{range}(A)$ και συμβολίζει ένα σύνολο διανυσμάτων που μπορεί να εκφραστεί ως Ax για κάποιο x .

Θεώρημα 1.1. [Theorem 1.1, [2]]

Το $\text{range}(A)$ είναι ο χώρος που παράγεται από τις στήλες του A .

Απόδειξη

Από την (1.2), κάθε Ax είναι ένας γραμμικός συνδυασμός των στηλών του A . Αντίστροφα, κάθε διάνυσμα y στο χώρο που παράγεται από τις στήλες του A μπορεί να γραφτεί ως γραμμικός συνδυασμός των στηλών $y = \sum_{j=1}^n x_j a_j$. Σχηματίζοντας ένα διάνυσμα x από τους συντελεστές x_j , έχουμε $y = Ax$ οπότε το y είναι στο range του A .

Μηδενικός χώρος

Ο μηδενικός χώρος του $A \in \mathbb{R}^{m \times n}$ (ή $\mathbb{C}^{m \times n}$), γράφεται ως $\text{null}(A)$, είναι το σύνολο διανυσμάτων x το οποίο ικανοποιεί τη σχέση $Ax = 0$, όπου 0 είναι το 0-διάνυσμα στο \mathbb{R}^m (ή \mathbb{C}^m). Οι συντεταγμένες κάθε διανύσματος $x \in \text{null}(A)$ δίνουν τους συντελεστές της επέκτασης του μηδενός ως ένα γραμμικό συνδυασμό των στηλών του A : $0 = x_1 a_1 + x_2 a_2 + \dots + x_n a_n$.

Βαθμός (Τάξη) Πίνακα

Ο βαθμός στηλών ενός πίνακα είναι η διάσταση του χώρου των στηλών του. Αντίστοιχα, ο βαθμός γραμμών είναι η διάσταση του χώρου που παράγεται από τις γραμμές του. Ο βαθμός των γραμμών ισούται πάντα με τον βαθμό των στηλών, οπότε αναφερόμαστε σε αυτόν τον αριθμό απλώς ως βαθμό πίνακα.

Ένας $m \times n$ πίνακας πλήρους βαθμού είναι ένας πίνακας ο οποίος έχει τον μέγιστο δυνατό βαθμό (το μικρότερο των m και n). Αυτό σημαίνει ότι ένας πίνακας πλήρους βαθμού, με $m \geq n$, πρέπει να έχει n γραμμικά ανεξάρτητες στήλες.

Θεώρημα 1.2. [Theorem 1.2, [2]]

Ένας πίνακας $A \in \mathbb{R}^{m \times n}$ (ή $\mathbb{C}^{m \times n}$) με $m \geq n$ είναι πλήρους βαθμού αν και μόνο αν δεν απεικονίζει δύο διαφορετικά διανύσματα στο ίδιο διάνυσμα.

Απόδειξη

(\Rightarrow) Αν ο A είναι ένας πίνακας πλήρους βαθμού, οι στήλες του είναι γραμμικά ανεξάρτητες, οπότε σχηματίζουν μία βάση του $\text{range}(A)$. Αυτό σημαίνει ότι κάθε $b \in \text{range}(A)$ έχει μία μοναδική γραμμική επέκταση σε σχέση με τις στήλες του A και για αυτό, από την (1.2), κάθε $b \in \text{range}(A)$ έχει ένα μοναδικό x , τέτοιο ώστε $b = Ax$.

(\Leftarrow) Αντίθετα, αν ο A δεν είναι πλήρους βαθμού, οι στήλες του a_j είναι εξαρτημένες και για αυτό υπάρχει ένας μη τετριμμένος γραμμικός συνδυασμός, τέτοιος ώστε $\sum_{j=1}^n c_j a_j = 0$. Το μη μηδενικό διάνυσμα c που σχηματίζεται από τους συντελεστές c_j ικανοποιεί τη σχέση $Ac = 0$. Όμως τότε ο A αντιστοιχίζει διαφορετικά διανύσματα στο ίδιο διάνυσμα, αφού για οποιοδήποτε x , $Ax = A(x+c)$.

Αντίστροφος Πίνακας

Αν A και X ανήκουν στο $\mathbb{R}^{n \times n}$ και ικανοποιούν την εξίσωση $AX = I$, τότε ο X είναι ο αντίστροφος του A και γράφεται ως A^{-1} . Αν υπάρχει ο A^{-1} , τότε ο A είναι αντιστρέψιμος.

1.2 Νόρμες

Η νόρμα είναι μία γενίκευση της απολύτου τιμής, ένας τρόπος για να μετρήσει κανείς το μέγεθος ενός διανύσματος, πίνακα ή τανυστή.

Ορισμός [Ορισμός 3.1, [1]]

Έστω X ένας γραμμικός χώρος στο \mathbb{R} ή στο \mathbb{C} και $\mathbb{K} = \mathbb{R}$ ή $\mathbb{K} = \mathbb{C}$, αντίστοιχα. Μία απεικόνιση

$$\|\cdot\| : X \rightarrow \mathbb{R}, \quad x \mapsto \|x\|, \quad (1.3)$$

λέγεται *νόρμα* (στάθμη, norm), αν ισχύουν:

1. $x \in X \quad \|x\| = 0 \iff x = 0$,
2. $\forall \lambda \in \mathbb{K} \quad \forall x \in X \quad \|\lambda x\| = |\lambda| \|x\|$,
3. $\forall x, y \in X \quad \|x + y\| \leq \|x\| + \|y\|$ (Τριγωνική Ανισότητα).

Παρατηρήσεις

- $\|x\| \geq 0, \quad \forall x \in X$,
- $\|x\|_2 = \sqrt{x^T x} = (\sum_{i=1}^n |x_i|^2)^{1/2}$ (Ευκλείδεια Νόρμα),
- $|x^T y| \leq \|x\|_2 \|y\|_2 \quad \forall x, y \in \mathbb{R}^n$ (Ανισότητα Cauchy-Schwartz).

Παραδείγματα

- $\|x\|_1 = \sum_{i=1}^n |x_i|$
- $\|x\|_2 = (\sum_{i=1}^n |x_i|^2)^{1/2}$
- $\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$
- $\|x\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p} \quad (1 \leq p \leq \infty)$

Ορισμός [Ορισμός 3.5, [1]]

Εστω $\|\cdot\|$ μία νόρμα στον \mathbb{R}^n . Η απεικόνιση

$$\|\cdot\| : \mathbb{R}^{n,n} \rightarrow \mathbb{R}, \quad \|A\| := \sup_{\substack{x \in \mathbb{R}^n \\ x \neq 0}} \frac{\|Ax\|}{\|x\|}, \quad (1.4)$$

λέγεται *φυσική νόρμα* πινάκων ή *νόρμα πινάκων παραγόμενη* από τη νόρμα $\|\cdot\|$ του \mathbb{R}^n .

Ιδιότητες

- $\|A\| \geq 0 \quad \forall A \in \mathbb{R}^{n \times n}, \mu\epsilon \quad \|A\| = 0 \iff A = 0$,
- $\|Ax\| \leq \|A\| \|x\|, \quad \forall x \in \mathbb{R}^n$,
- $\|\alpha A\| = |\alpha| \|A\|, \quad \forall \alpha \in \mathbb{R}$,
- $\|A + B\| \leq \|A\| + \|B\|, \quad \forall A, B \in \mathbb{R}^{n \times n}$,
- $\|A \cdot B\| \leq \|A\| \|B\|, \quad \forall A, B \in \mathbb{R}^{n \times n}$.

Νόρμα Frobenius

Η πιο σημαντική νόρμα πίνακα είναι η *νόρμα του Frobenius*, η οποία ορίζεται ως:

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

Δεν αποτελεί φυσική νόρμα πίνακα, καθώς δεν ικανοποιεί την ιδιότητα $\|I\| = \sup_{x \neq 0} \frac{\|Ix\|}{\|x\|} = 1$.

1.3 Ορθογώνια Διανύσματα

Δύο διανύσματα x, y είναι ορθογώνια αν $x^T y = 0$. Αν x, y είναι πραγματικοί αριθμοί, αυτό σημαίνει ότι βρίσκονται σε ορθή γωνία μεταξύ τους στο \mathbb{R} . Δύο σύνολα διανυσμάτων X και Y είναι ορθογώνια (ή αλλιώς “Το X είναι ορθογώνιο στο Y ”) αν κάθε $x \in X$ είναι ορθογώνιο σε κάθε $y \in Y$. Ένα σύνολο S από μη μηδενικά ορθογώνια διανύσματα είναι ορθογώνιο αν τα στοιχεία του είναι κατά ζεύγη ορθογώνια, π.χ. αν για $x, y \in S, x \neq y \Rightarrow x^T y = 0$. Ένα σύνολο διανυσμάτων είναι *ορθοκανονικό* αν είναι ορθογώνιο και επιπλέον, κάθε $x \in S$ έχει $\|x\| = 1$.

Θεώρημα 1.3. [Theorem 2.1, [2]]

Τα διανύσματα σε ένα ορθογώνιο σύνολο S είναι γραμμικά ανεξάρτητα.

Απόδειξη

Αν τα διανύσματα στο S δεν είναι ανεξάρτητα, τότε κάποιο $v_k \in S$ μπορεί να εκφραστεί ως γραμμικός συνδυασμός άλλων μελών $v_1, \dots, v_n \in S$,

$$v_k = \sum_{i=1, i \neq k}^n c_i v_i.$$

Αφού $v_k \neq 0, v_k^T v_k = \|v_k\|^2 > 0$. Χρησιμοποιώντας την συγγραμικότητα του εσωτερικού γινομένου και της ορθογωνιότητας του S , υπολογίζουμε

$$v_k^T v_k = \sum_{i=1, i \neq k}^n c_i v_k^T v_i = 0,$$

το οποίο αντικρούει στην υπόθεση ότι τα διανύσματα στο S είναι μη μηδενικά.

Θεώρημα 1.4. [Unitary Matrices, [2]]

Ένας τετραγωνικός πίνακας $Q \in \mathbb{R}^{m \times m}$ είναι ορθογώνιος αν $Q^T = Q^{-1}$, δηλαδή αν $Q^T Q = I$. Σε μορφή πίνακα:

$$\begin{bmatrix} q_1^T \\ q_2^T \\ \vdots \\ q_m^T \end{bmatrix} \left[q_1 \mid q_2 \mid \dots \mid q_m \right] = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}.$$

Οι στήλες ενός ορθογώνιου πίνακα Q , σχηματίζουν μία ορθομοναδιαία βάση στο \mathbb{R}^m .

1.4 Ανάλυση πίνακα σε ιδιάζουσες τιμές - SVD

Η μέθοδος SVD (Singular Value Decomposition) είναι μια θεμελιώδης τεχνική παραγοντοποίησης πινάκων στη γραμμική άλγεβρα. Είναι ένα μαθηματικό εργαλείο που χρησιμοποιείται για διάφορες εφαρμογές, όπως η συμπίεση δεδομένων, η μείωση της διαστατικότητας και η εξαγωγή σημαντικών μεταβλητών. Επίσης, μπορεί να χρησιμοποιηθεί για πραγματικούς και για μιγαδικούς πίνακες.

Θεώρημα 1.5. [Theorem 2.4.1, [3]]

Αν ο A είναι ένας πραγματικός $m \times n$ πίνακας, τότε υπάρχουν ορθογώνιοι πίνακες

$$U = [u_1 | \dots | u_m] \in \mathbb{R}^{m \times m} \quad \text{και} \quad V = [v_1 | \dots | v_n] \in \mathbb{R}^{n \times n}$$

τέτοιοι ώστε

$$U^T A V = \Sigma = \text{diag}(\sigma_1, \dots, \sigma_p) \in \mathbb{R}^{m \times n}, \quad p = \min\{m, n\},$$

όπου $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$.

Ελαττωμένη SVD (Reduced SVD)

Η παραγοντοποίηση του A καλείται *ελαττωμένη SVD* του A και γράφεται ως:

$$A = \hat{U} \hat{\Sigma} V^T. \tag{1.5}$$

Αυτό προκύπτει, αφού οι εξισώσεις που αφορούν δεξιά μοναδιαία διανύσματα $\{v_j\}$ και αριστερά

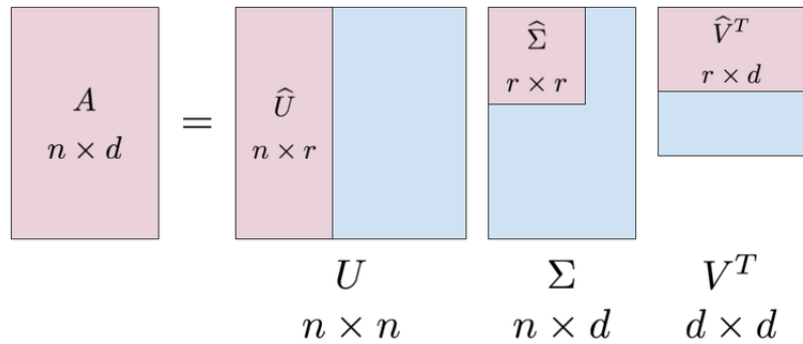
μοναδιαία διανύσματα $\{u_j\}$, μπορούν να γραφτούν ως:

$$Av_j = \sigma_j u_j, \quad 1 \leq j \leq n. \quad (1.6)$$

Αυτή τη σχέση μπορούμε να την εκφράσουμε σε μορφή εξίσωσης πίνακα ως:

$$AV = \hat{U}\hat{\Sigma}.$$

Εδώ, ο $\hat{\Sigma}$ είναι ένας $n \times n$ διαγώνιος πίνακας με θετικές τιμές, αφού ο A θεωρείται μέγιστου βαθμού n . Ο \hat{U} είναι ένας $m \times m$ πίνακας με ορθοκανονικές στήλες και ο V ένας $n \times n$ πίνακας με ορθοκανονικές στήλες. Επομένως, ο V είναι μοναδιαίος και μπορούμε να πολλαπλασιάσουμε με τον αντίστροφό του, V^T , ώστε να καταλήξουμε στη σχέση: $A = \hat{U}\hat{\Sigma}V^T$.



Σχήμα 1.1: SVD και Ελλατωμένη SVD

Θεώρημα 1.6. [Theorem 4.1, [2]]

Θεώρημα Ύπαρξης και Μοναδικότητας

Κάθε πίνακας $A \in \mathbb{C}^{m \times n}$ μπορεί να αναλυθεί σε ιδιάζουσες τιμές. Επιπλέον, οι ιδιάζουσες τιμές $\{\sigma_j\}$ είναι μοναδικά ορισμένες και αν ο A είναι τετραγωνικός και οι σ_j διαφορετικές, τα αριστερά και δεξιά ιδιάζοντα διανύσματα $\{u_j\}$ και $\{v_j\}$, καθορίζονται μοναδικά με έννοιες πολυπλοκότητας (δηλ. μιγαδικοί βαθμωτοί παράγοντες με απόλυτη τιμή 1).

Απόδειξη

Για να αποδείξουμε την ύπαρξη της SVD, απομονώνουμε την κατεύθυνση της μεγαλύτερης δράσης του A και μετά προχωράμε με επαγωγή στη διάσταση του A .

Ορίζουμε $\sigma_1 = \|A\|_2$. Από επιχείρημα συμπάγιας, πρέπει να υπάρχουν διανύσματα $v_1 \in \mathbb{C}^n$ και $u_1 \in \mathbb{C}^m$ με $\|v_1\|_2 = \|u_1\|_2 = 1$ και $Av_1 = \sigma_1 u_1$. Θεωρούμε μία επέκταση του v_1 σε μία ορθοκανονική βάση $\{v_j\}$ του \mathbb{C}^n και του u_1 σε μία ορθοκανονική βάση $\{u_j\}$ του \mathbb{C}^m και έστω U_1 και V_1 συμβολίζουν

τους ορθομοναδιαίους πίνακες με στήλες u_j και v_j αντίστοιχα. Τότε έχουμε:

$$U_1^T A V_1 = S = \begin{bmatrix} \sigma_1 & w^T \\ 0 & B \end{bmatrix}, \quad (1.7)$$

όπου 0 είναι ένα διάνυσμα στήλη διάστασης $m - 1$, w^T είναι ένα διάνυσμα γραμμή διάστασης $n - 1$ και ο B έχει διαστάσεις $(m - 1) \times (n - 1)$. Επίσης,

$$\left\| \begin{bmatrix} \sigma_1 & w^T \\ 0 & B \end{bmatrix} \begin{bmatrix} \sigma_1 \\ w \end{bmatrix} \right\|_2 \geq \sigma_1^2 + w^T w = (\sigma_1^2 + w^T w)^{1/2} \left\| \begin{bmatrix} \sigma_1 \\ w \end{bmatrix} \right\|_2$$

το οποίο οδηγεί στη σχέση: $\|S\|_2 \geq (\sigma_1^2 + w^T w)^{1/2}$. Αφού U_1 και V_1 είναι ορθομοναδιαίοι, γνωρίζουμε ότι $\|S\|_2 = \|A\|_2 = \sigma_1$, οπότε $w = 0$.

Αν $n = 1$ ή $m = 1$, η απόδειξη έχει τελειώσει. Σε διαφορετική περίπτωση, ο υποπίνακας B περιγράφει την ενέργεια του A στον ορθογώνιο υπόχωρο του v_1 . Από την υπόθεση επαγωγής, ο B έχει SVD: $B = U_2 \Sigma_2 V_2^T$. Τώρα επιβεβαιώνεται εύκολα ότι:

$$A = U_1 \begin{bmatrix} 1 & 0 \\ 0 & U_2 \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & V_2 \end{bmatrix}^T V_1^T$$

είναι SVD του A , το οποίο ολοκληρώνει την απόδειξη της ύπαρξης.

Για να αποδείξουμε τη μοναδικότητα, η γεωμετρική αιτιολόγηση είναι απλή: αν τα μήκη των η-μαξόνων μιας υπερέλλειψης είναι διακριτά, τότε οι ίδιοι οι ημιάξονες ορίζονται από τη γεωμετρία. Αλγεβρικά, εργαζόμαστε ως εξής: Αρχικά σημειώνουμε ότι το σ_1 είναι μοναδικά ορισμένο, λόγω της ισότητάς του με το $\|A\|_2$. Τώρα, υποθέτουμε ότι εκτός από το v_1 , υπάρχει ένα άλλο γραμμικά ανεξάρτητο διάνυσμα w , με $\|w\|_2 = 1$ και $\|Aw\|_2 = \sigma_1$. Ορίζουμε ένα μοναδιαίο διάνυσμα v_2 , ορθογώνιο στο v_1 , ως γραμμικό συνδυασμό των v_1 και w ,

$$v_2 = \frac{w - (v_1^T w)v_1}{\|w - (v_1^T w)v_1\|_2}.$$

Αφού $\|A\|_2 = \sigma_1$, $\|Av_2\|_2 \leq \sigma_1$, αλλά αυτό θα έπρεπε να είναι ισότητα, αφού διαφορετικά $w = v_1 c + v_2 s$ για κάποιες σταθερές c και s με $|c|^2 + |s|^2 = 1$, θα είχαμε $\|Aw\|_2 < \sigma_1$. Αυτό το διάνυσμα v_2 είναι ένα δεύτερο δεξιό ιδιάζον διάνυσμα του A που αντιστοιχεί στην ιδιάζουσα τιμή σ_1 , το οποίο οδηγεί στην εμφάνιση ενός διανύσματος y (που ισούται με τα τελευταία $n - 1$ στοιχεία του $V_1^T v_2$) με $\|y\|_2 = 1$ και $\|By\|_2 = \sigma_1$. Καταλήγουμε ότι αν το ιδιάζον διάνυσμα v_1 δεν είναι μοναδικό, τότε η αντίστοιχη ιδιάζουσα τιμή σ_1 δεν είναι απλή. Για να ολοκληρώσουμε την απόδειξη

της μοναδικότητας, σημειώνουμε ότι αφού τα σ_1 , v_1 και u_1 είναι ορισμένα, το υπόλοιπο της SVD είναι ορισμένο από την ενέργεια του A στο χώρο ορθογώνιο του v_1 . Αφού το v_1 είναι μοναδικό με έννοιες πολυπλοκότητας, ο ορθογώνιος χώρος είναι μοναδικά ορισμένος, που συνεπάγεται και τη μοναδικότητα των υπόλοιπων ιδιάζόντων τιμών και διανυσμάτων.

Παρατήρηση

Αν ο $A \in \mathbb{R}^{n \times m}$, τότε

$$\|A\|_2 = \sigma_1 \quad \text{και} \quad \|A\|_F = \sqrt{\sigma_1^2 + \dots + \sigma_p^2}.$$

Παράδειγμα 1

Θα υπολογίσουμε την SVD του πίνακα $A = \begin{bmatrix} i+1 & 1 \\ 1-i & -i \end{bmatrix}$.

Αρχικά υπολογίζουμε το γινόμενο

$$A^T A = \begin{bmatrix} 4 & 2(1-i) \\ 2(1+i) & 2 \end{bmatrix}.$$

Στη συνέχεια υπολογίζουμε τις ιδιοτιμές του $A^T A$ ως

$$\det \begin{vmatrix} 4 - \lambda & 2(1-i) \\ 2(i+1) & 2 - \lambda \end{vmatrix} = 0,$$

με χαρακτηριστικό πολυώνυμο

$$(4 - \lambda)(2 - \lambda) - 8 = 0.$$

Οπότε τα $\lambda_1 = 6$ και $\lambda_2 = 0$ είναι οι ιδιοτιμές. Οι ιδιάζουσες τιμές είναι αντίστοιχα $\sigma_1 = \sqrt{6}$ και $\sigma_2 = 0$. Άρα

$$\Sigma = \begin{pmatrix} \sqrt{6} & 0 \\ 0 & 0 \end{pmatrix}$$

Ένα κανονικοποιημένο ιδιοδύνασμα της ιδιοτιμής $\lambda_1 = 6$ είναι το:

$$w_1 = \begin{pmatrix} \frac{2}{\sqrt{6}} \\ \frac{1+i}{\sqrt{6}} \end{pmatrix}$$

και ένα κανονικοποιημένο ιδιοδύνασμα της ιδιοτιμής $\lambda_2 = 0$ είναι το:

$$w_2 = \begin{pmatrix} \frac{1-i}{\sqrt{6}} \\ \frac{-2}{\sqrt{6}} \end{pmatrix}.$$

Οπότε έχουμε

$$W = \begin{pmatrix} \frac{2}{\sqrt{6}} & \frac{1-i}{\sqrt{6}} \\ \frac{1+i}{\sqrt{6}} & \frac{-2}{\sqrt{6}} \end{pmatrix}.$$

Άρα μπορούμε να υπολογίσουμε τις τιμές του πίνακα U ως

$$u_1 = \frac{1}{\sqrt{6}}Aw_1 = \begin{pmatrix} \frac{1+i}{2} \\ \frac{1-i}{2} \end{pmatrix}$$

και

$$u_2 = \frac{1}{\sqrt{6}}Aw_2 = \begin{pmatrix} \frac{1+i}{2} \\ \frac{-1+i}{2} \end{pmatrix}.$$

Συνεπώς μία SVD είναι

$$A = \begin{pmatrix} \frac{1+i}{2} & \frac{1+i}{2} \\ \frac{1-i}{2} & \frac{-1+i}{2} \end{pmatrix} \begin{pmatrix} \sqrt{6} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{2}{\sqrt{6}} & \frac{1-i}{\sqrt{6}} \\ \frac{1+i}{\sqrt{6}} & \frac{-2}{\sqrt{6}} \end{pmatrix}^T.$$

Σε αυτό το παράδειγμα οι ιδιάζουσες τιμές είναι μοναδικά ορισμένες για τον πίνακα A , αλλά οι ορθομοναδιαίες βάσεις διανυσμάτων που σχηματίζουν τις στήλες των πινάκων V και W , δεν είναι μοναδικά ορισμένες. Αυτό συμβαίνει επειδή υπάρχει πάνω από μία ορθομοναδιαία βάση που περιλαμβάνει τα ιδιοδιανύσματα του $A^T A$.

Παράδειγμα 2

Θα βρούμε την SVD, με τη χρήση του προγράμματος matlab, για έναν 5-διαγώνιο πίνακα, μεγάλης διάστασης, ο οποίος είναι συμμετρικός και θετικά ορισμένος. Έστω πίνακας A, με τα στοιχεία της κύριας διαγωνίου να είναι ο αριθμός 3. Τα στοιχεία της διαγωνίου πάνω και κάτω από την κύρια διαγώνιο είναι -1 και τα στοιχεία πάνω και κάτω από αυτά είναι -2. Τα υπόλοιπα στοιχεία του πίνακα ισούνται με 0. Αρχικά, ορίζουμε τον πίνακα A με τις εντολές:

```

1 %% Sample Matlab code
2 n=1000;
3 A = diag(3 * ones(n, 1));
4 offDiagonalValues = -ones(1, n);
5 offDiagonalValues2 = -2*ones(1, n);
6 A = A + spdiags(offDiagonalValues1', 1, n, n) + ...
7       spdiags(offDiagonalValues1', -1, n, n) + ...
8       spdiags(offDiagonalValues2', 2, n, n) + ...
9       spdiags(offDiagonalValues2', -2, n, n);
10 B=full(A);

```

Οπότε ο B είναι:

$$B = \begin{bmatrix} 3 & -1 & -2 & 0 & \dots & & & & & & \\ -1 & 3 & -1 & -2 & \dots & & & & & & \\ -2 & -1 & 3 & -1 & \dots & & & & & & \\ 0 & -2 & -1 & 3 & \dots & & & & & & \\ & & & & \dots & \dots & & & & & \\ & & & & & & \dots & 3 & -1 & -2 & 0 \\ & & & & & & \dots & -1 & 3 & -1 & -2 \\ & & & & & & \dots & -2 & -1 & 3 & -1 \\ & & & & & & \dots & 0 & -2 & -1 & 3 \end{bmatrix}$$

και στη συνέχεια χρησιμοποιούμε την εντολή $[U, S, V] = \text{svd}(B)$ για να πάρουμε ως αποτέλεσμα τους πίνακες U, S, V .

$$U = \begin{bmatrix} 0.0003 & -0.0002 & 0.0006 & \dots \\ 0.0001 & 0.0004 & 0.0003 & \dots \\ -0.0007 & 0.0003 & -0.0014 & \dots \\ -0.0001 & -0.0008 & -0.0002 & \dots \\ & & \dots & \\ & & \dots & -0.0175 & -0.0162 & -0.0168 \\ & & \dots & -0.0427 & -0.0424 & -0.0426 \\ & & \dots & -0.0393 & -0.0391 & -0.0392 \\ & & \dots & -0.0412 & -0.0417 & -0.0414 \end{bmatrix}$$

$$S = \begin{bmatrix} 7.1249 & 0 & 0 & 0 & \dots \\ 0 & 7.1249 & 0 & 0 & \dots \\ 0 & 0 & 7.1247 & 0 & \dots \\ 0 & 0 & 0 & 7.1247 & \dots \\ & & \dots & & \\ & & \dots & 0.0536 & 0 & 0 & 0 \\ & & \dots & 0 & 0.0280 & 0 & 0 \\ & & \dots & 0 & 0 & 0.0263 & 0 \\ & & \dots & 0 & 0 & 0 & 0.0009 \end{bmatrix}$$

$$V = \begin{bmatrix} 0.0003 & -0.0002 & 0.0006 & \dots \\ 0.0001 & 0.0004 & 0.0003 & \dots \\ -0.0007 & 0.0003 & -0.0014 & \dots \\ -0.0001 & -0.0008 & -0.0002 & \dots \\ & & \dots & \\ & & \dots & 0.0175 & -0.0162 & 0.0168 \\ & & \dots & 0.0427 & -0.0424 & 0.0426 \\ & & \dots & 0.0393 & -0.0391 & 0.0392 \\ & & \dots & 0.0412 & -0.0417 & 0.0414 \end{bmatrix}$$

Επαληθεύουμε υπολογίζοντας το $U * S * V'$, όπου λαμβάνουμε τον πίνακα B.

Για να υπολογίσουμε την ελαττωμένη SVD, χρησιμοποιούμε την εντολή $[U, S, V] = \text{svd}(B, "econ")$, όπου παίρνουμε το ίδιο αποτέλεσμα.

Καταλαβαίνουμε, λοιπόν, ότι η SVD είναι μία διαδικασία που λειτουργεί στην πράξη και για προβλήματα μεγάλων διαστάσεων.

Ο χρόνος που χρειάστηκε το matlab για να τρέξει αυτό τον κώδικα ήταν $\text{time} = 0.5291$ δευτερόλεπτα.

Έστω τώρα, ότι έχουμε έναν τυχαίο πίνακα C διάστασης 50×50 και έναν τυχαίο πίνακα διάστασης 100×100 , και θέλουμε να υπολογίσουμε τον χρόνο που θα χρειαστεί για να εκτελέσουμε την svd, σε κάθε περίπτωση.

```

1 %% Matlab code for random matrix
2 tic
3 C=rand(50,50);
4 [U,S,V]=svd(C);
5 timeElapsed1 = toc;
6
7 tic
8 C=rand(1000,1000);
9 [U,S,V]=svd(C);
10 timeElapsed2 = toc;
```

Λαμβάνουμε τα αποτελέσματα $\text{timeElapsed1} = 0.0078$ και $\text{timeElapsed2} = 0.3169$.

Οπότε, παρατηρούμε ότι για πίνακες μεγαλύτερης διάστασης χρειάζεται περισσότερος χρόνος για να εκτελεστεί η SVD. Όμως, χρειάστηκε λιγότερο χρόνο για έναν τυχαίο πίνακα σε σχέση με τον 5-διαγώνιο που είχαμε στο Παράδειγμα 2.

Μερικές χρήσεις της SVD

- Υπολογισμός του ψευδοαντιστρόφου
- Προσέγγιση πινάκων
- Προσδιορισμός της τάξης, του εύρους και του μηδενικού χώρου ενός πίνακα

- Επίλυση προβλημάτων ελαχίστων τετραγώνων
- Συμπίεση εικόνας
- Ανάλυση σήματος

Υπολογισμός του ψευδοαντιστρόφου

Η SVD μπορεί να διευκολύνει τον έλεγχο ύπαρξης του αντιστρόφου ενός πίνακα. Συγκεκριμένα, αν οι ιδιάζουσες τιμές είναι $\sigma_i = 0$, τότε ο Σ^{-1} δεν υπάρχει, επειδή η αντίστοιχη διαγώνια τιμή θα ήταν $\frac{1}{\sigma_i} = \frac{1}{0}$.

Με άλλα λόγια, αν ένας πίνακας A έχει οποιαδήποτε μηδενική ιδιάζουσα τιμή (έστω $\sigma_j = 0$), τότε πολλαπλασιάζοντας επί A , καταστρέφεται η πληροφορία, επειδή η συνιστώσα του διανύσματος μαζί με το δεξιά ιδιάζων διάνυσμα u_j πολλαπλασιάζονται με το 0. Δε μπορούμε να ανακτήσουμε αυτή την πληροφορία, οπότε δεν υπάρχει τρόπος να αντιστρέψουμε το Ax ώστε να ανακτήσουμε το αρχικό x . Το καλύτερο που μπορούμε να κάνουμε, είναι να ανακτήσουμε τα στοιχεία του x , τα οποία δεν καταστράφηκαν από τον πολλαπλασιασμό με το 0.

Ο πίνακας που μετατρέπει όλες τις ανακτημένες πληροφορίες καλείται *ψευδοαντίστροφος* και συμβολίζεται με A^\dagger . Μπορούμε να υπολογίσουμε τον ψευδοαντίστροφο με την SVD, αντιστρέφοντας τις μη-μηδενικές ιδιάζουσες τιμές και αφήνοντας όλες τις μηδενικές ιδιάζουσες τιμές μηδέν.

Η εξίσωση για τον υπολογισμό του ψευδοαντιστρόφου είναι: $A^\dagger = VS^\dagger U^T$.

Με τη χρήση του matlab υπολογίζουμε:

```
1 %% Matlab code for pseudoinverse
2 A = [ 2 4 6; 8 5 7 ; 0 3 1];
3 [U,S,V] = svd(A);
4 A_cross=V*inv(S)*U'
```

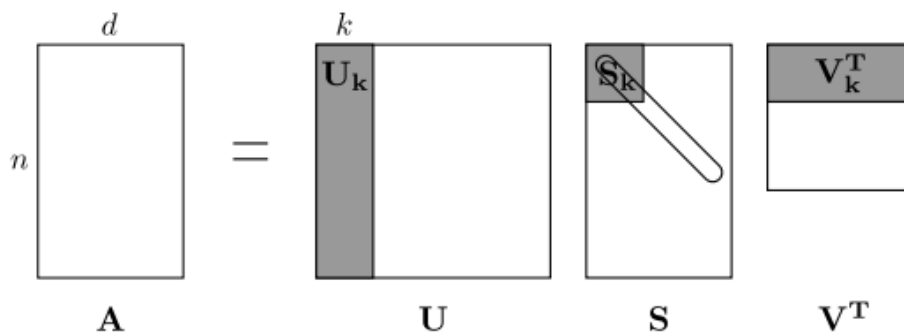
και παίρνουμε ως αποτέλεσμα: $A^\dagger = \begin{bmatrix} -0.200000 & 0.175000 & -0.025000 \\ -0.100000 & 0.025000 & 0.425000 \\ 0.300000 & -0.075000 & -0.275000 \end{bmatrix}$.

Προσέγγιση πινάκων

Πώς θα μπορούσαμε να προσεγγίσουμε με τον καλύτερο δυνατό τρόπο έναν πίνακα A με έναν πίνακα τάξης k ; Αν είχαμε μια αναπαράσταση του πίνακα δεδομένων A ως άθροισμα πολλών συστατικών, με

αυτά τα συστατικά ταξινομημένα κατά «σπουδαιότητα», τότε θα μπορούσαμε να κρατήσουμε μόνο τα k «πιο σημαντικά». Η SVD μας δίνει ακριβώς μια τέτοια αναπαράσταση. Αυτό συμβαίνει επειδή η SVD, εκφράζει έναν πίνακα A ως άθροισμα πινάκων τάξης 1 (σταθμισμένων με τις αντίστοιχες ιδιάζουσες τιμές). Μία διαδικασία που θα μπορούσαμε να ακολουθήσουμε είναι η εξής:

1. Υπολογίζουμε την SVD: $A = USV^T$, όπου ο U είναι ένας ορθογώνιος πίνακας $m \times m$, ο S είναι ένας μη αρνητικός $m \times n$ διαγώνιος πίνακας, με διαγώνιες καταχωρήσεις ταξινομημένες από την πιο υψηλή προς την πιο χαμηλή, και V^T είναι ένας ορθογώνιος πίνακας $n \times n$.
2. Κρατάμε μόνο τα κορυφαία k δεξιά ιδιάζοντα διανύσματα: ορίζουμε V_k^T ίσο με τις k πρώτες γραμμές του V^T (ένας $k \times n$ πίνακας).
3. Κρατάμε μόνο τα κορυφαία k αριστερά ιδιάζοντα διανύσματα: θέτουμε U_k ίσο με τις πρώτες k στήλες του U (ένας $m \times k$ πίνακας).
4. Κρατάμε μόνο τις κορυφαίες k ιδιάζουσες τιμές: θέτουμε S_k ίσο με τις πρώτες k γραμμές και στήλες του S (ένας $k \times k$ πίνακας), που αντιστοιχούν στις k μεγαλύτερες ιδιάζουσες τιμές του A .
5. Η προσέγγιση τάξης- k είναι: $A_k = U_k S_k V_k^T$.



Σχήμα 1.2: Low rank approximation via SVD

Με τη χρήση του matlab:

```

1 %% Matlab code for Low Rank Matrix Approximation
2 k=5;
3 A = randn(300,300);
4 [U,S,V] = svd(A);
5 Ak = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';

```

Στη συνέχεια θα δούμε και τη χρήση της SVD σε προβλήματα ελαχίστων τετραγώνων.

1.5 Προβολές

Εάν θέλουμε να υπολογίσουμε ένα πίνακα ή ένα διάνυσμα, τότε οι νόρμες είναι χρήσιμες για την αξιολόγηση της ακρίβειας της απάντησης ή για τη μέτρηση της προόδου κατά τη διάρκεια μιας επανάληψης. Εάν το αντικείμενο ενός υπολογισμού είναι ο υπολογισμός ενός υποχώρου, τότε για να κάνουμε παρόμοια σχόλια πρέπει να μπορούμε να ποσοτικοποιήσουμε την απόσταση μεταξύ δύο υποχώρων. Για αυτό χρησιμοποιούμε τις ορθογώνιες προβολές.

Ορισμός

Η *προβολή* είναι ένας γραμμικός μετασχηματισμός P , από ένα διανυσματικό χώρο στον εαυτό του (ενδομορφισμός), τέτοιος ώστε $P \circ P = P$. Επομένως, όποτε ο P εφαρμόζεται δύο φορές σε οποιοδήποτε διάνυσμα, δίνει το ίδιο αποτέλεσμα σαν να είχε εφαρμοστεί μία φορά (δηλαδή ο P είναι ταυτοδύναμος), αφήνοντας αναλλοίωτη την εικόνα του διανύσματος.

Ορισμός

Ένας τετραγωνικός πίνακας P ονομάζεται *πίνακας προβολής*, αν ικανοποιεί τη σχέση

$$P^2 = P. \tag{1.8}$$

Ένας τέτοιος πίνακας καλείται επίσης ταυτοδύναμος.

Σημείωση

Ο πίνακας P δεν είναι απαραίτητα ορθογώνιος πίνακας προβολής. Συγκεκριμένα, τις περισσότερες φορές δεν είναι ούτε ορθογώνιος πίνακας.

Παρατήρηση

Έχουμε $v = Px$ για κάποιο x και

$$Pv = P^2x = Px = v.$$

Δηλαδή, αν $v \in \text{range}(P)$, τότε βρίσκεται ακριβώς πάνω στη σκιά του και εφαρμόζοντας την προβολή έχει ως αποτέλεσμα το ίδιο το v .

Ορισμός [Complementary Projectors, [2]]

Αν ο P είναι προβολή, τότε και ο $I - P$ είναι προβολή και ταυτοδύναμος:

$$(I - P)^2 = I - 2P + P^2 = I - P.$$

Ο πίνακας $I - P$ καλείται συμπληρωματική προβολή του P .

Λήμμα [Lemma 3.1, [5]]

Αν ο P είναι προβολή τότε:

$$\text{range}(I - P) = \text{null}(P)$$

$$\text{null}(I - P) = \text{range}(P)$$

Απόδειξη

Για την πρώτη σχέση:

Αρχικά, θα δείξουμε ότι $\text{null}(P) \subseteq \text{range}(I - P)$.

Θεωρούμε διάνυσμα v , τέτοιο ώστε $Pv = 0$. Τότε, $(I - P)v = v - Pv = v$. Με άλλα λόγια, οποιοδήποτε v στον μηδενικό χώρο P , βρίσκεται επίσης στο $\text{range}(I - P)$.

Τώρα, θα δείξουμε ότι $\text{range}(I - P) \subseteq \text{null}(P)$. Ξέρουμε ότι κάθε $x \in \text{range}(I - P)$ χαρακτηρίζεται από $x = (I - P)v$ για κάποιο v . Οπότε, $x = v - Pv = -(Pv - v) \in \text{null}(P)$, αφού $P(Pv - v) = P^2v - Pv = Pv - Pv = 0$. Άρα, αν $x \in \text{range}(I - P)$, τότε $x \in \text{null}(P)$.

Για τη δεύτερη σχέση:

Ακολουθούμε την ίδια διαδικασία για $P = I - (I - P)$.

Θεώρημα 1.7. [Theorem 6.1, [2]]

Μία προβολή P είναι ορθογώνια αν και μόνο αν $P = P^T$.

Απόδειξη

Έστω δύο υπόχωροι του \mathbb{C}^m , S_1, S_2 , με $S_1 = \text{range}(P)$ και $S_2 = \text{null}(P)$, για τους οποίους ισχύει $S_1 + S_2 = \mathbb{C}^m$. Αν $P = P^T$, τότε το εσωτερικό γινόμενο μεταξύ ενός διανύσματος $Px \in S_1$ και ενός διανύσματος $(I - P)y \in S_2$ είναι μηδέν:

$$x^T P^T (I - P)y = x^T (P - P^2)y = 0.$$

Επομένως η προβολή είναι ορθογώνια.

Για την αντίθετη φορά της απόδειξης, πρέπει να κάνουμε χρήση της SVD. Υποθέτουμε ότι ο P προβάλλεται στον S_1 κατά μήκος του S_2 , όπου $S_1 \perp S_2$ και ο S_1 έχει διάσταση n . Τότε μία SVD του P , μπορεί να κατασκευαστεί ως εξής. Έστω $\{q_1, q_2, \dots, q_m\}$ μία ορθομοναδιαία βάση του \mathbb{C}^m , όπου $\{q_1, \dots, q_n\}$ είναι μία βάση του S_1 και $\{q_{n+1}, \dots, q_m\}$ είναι μία βάση του S_2 . Για $j \leq n$, έχουμε $Pq_j = q_j$ και για $j > n$, έχουμε $Pq_j = 0$. Τώρα θεωρούμε Q ένα μοναδιαίο πίνακα, του οποίου η j -ιοστή στήλη είναι η q_j . Τότε έχουμε

$$PQ = \left[\begin{array}{c|c|c|c|c} q_1 & \dots & q_n & 0 & \dots \\ \hline \end{array} \right],$$

οπότε

$$Q^T P Q = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & & 0 & \\ & & & & \ddots \end{bmatrix} = \Sigma,$$

ένας διαγώνιος πίνακας με μονάδες στις πρώτες n θέσεις και μηδενικά οπουδήποτε αλλού. Άρα έχουμε κατασκευάσει μία SVD του P :

$$P = Q \Sigma Q^T. \quad (1.9)$$

Συνεπώς, παρατηρούμε ότι ο P είναι ερμιτιανός, δηλαδή $P^T = (Q \Sigma Q^T)^T = Q \Sigma^T Q^T = Q \Sigma Q^T = P$.

Θεώρημα 1.8. [Theorem 1,[6]]

Έχοντας μία ορθογώνια προβολή P και ένα διάνυσμα x , το Px λύνει μοναδικά το πρόβλημα βελτιστοποίησης

$$\min_{z \in \text{range}(P)} \|z - x\|_2.$$

Με άλλα λόγια, το Px είναι το μοναδικό κοντινότερο σημείο στο $\text{range}(P)$.

Απόδειξη

Θα το αποδείξουμε δείχνοντας ότι για κάθε $z \in \text{range}(P)$ με $z \neq Px$

$$\|z - x\|_2 > \|Px - x\|_2.$$

Για να το πετύχουμε αυτό παρατηρούμε

$$z - x = z - Px + Px - x,$$

όπου παίρνοντας τις τετραγωνικές νόρμες κάθε μέρους προκύπτει

$$\|z - x\|_2^2 = \|z - Px + Px - x\|_2^2. \quad (1)$$

Παρατηρούμε επίσης, ότι $z - Px \perp Px - x$. Αυτό συμβαίνει επειδή $z = Pw$ για κάποιο w και $P = P^T$. Οπότε,

$$\begin{aligned} (z - Px)^T(Px - x) &= (Pw - Px)^T(Px - x) \\ &= (w - x)^T P(Px - x) \\ &= (w - x)^T(P^2x - Px) \\ &= (w - x)^T(Px - Px) \\ &= 0. \end{aligned}$$

Τέλος, γυρίζοντας στην (1) συμπεραίνουμε ότι

$$\begin{aligned} \|z - x\|_2^2 &= \|z - Px + Px - x\|_2^2 \\ &= (z - Px)^T(z - Px) + 2(z - Px)^T(Px - x) + (Px - x)^T(Px - x) \\ &= (z - Px)^T(z - Px) + (Px - x)^T(Px - x) \\ &= \|z - x\|_2^2 + \|Px - x\|_2^2 \\ &> \|Px - x\|_2^2. \end{aligned}$$

όπου η τελευταία ανισότητα προκύπτει από το γεγονός ότι $z \neq Px$, για αυτό $\|z - x\|_2^2 > 0$.

Κεφάλαιο **2**

Παραγοντοποίηση QR

Σε αυτό το κεφάλαιο, θα ορίσουμε αρχικά την ορθοκανονικοποίηση Gram-Schmidt, η οποία θα μας βοηθήσει να ορίσουμε την Παραγοντοποίηση QR. Θα αναφερθούμε επίσης και στη μέθοδο Householder.

2.1 Ορθοκανονικοποίηση Gram-Schmidt

Ορισμός [Ορισμός 5.3, [1]]

Έστω $(X, (\cdot, \cdot))$ ένας Ευκλείδειος χώρος. Ένα σύνολο $S = \{a_1, \dots, a_n\} \subset X$ λέγεται *ορθογώνιο* (ορθογώνιο σύστημα), αν τα στοιχεία του είναι ορθογώνια μεταξύ τους, δηλαδή αν για $i \neq j$ ισχύει $(a_i, a_j) = 0$. Αν, επιπλέον, κάθε διάνυσμα του S είναι *μοναδιαίο*, δηλαδή $\|a_i\| = 1$, για κάθε i , τότε συνολικά έχουμε $(a_i, a_j) = \delta_{ij}$, τότε το S λέγεται *ορθοκανονικό* (ορθοκανονικό σύστημα). Αν ένα ορθοκανονικό σύνολο S είναι βάση του X , τότε το S λέγεται *ορθοκανονική βάση* του X .

Λήμμα [Λήμμα 5.3, [1]]

Έστω $(X, (\cdot, \cdot))$ ένας χώρος με εσωτερικό γινόμενο και $S = \{q_1, \dots, q_n\}$ ένα ορθοκανονικό σύνολο στοιχείων του X . Τα q_1, \dots, q_n είναι τότε *γραμμικά ανεξάρτητα*.

Απόδειξη

Έστω $\{\lambda_1, \dots, \lambda_n\} \in \mathbb{R}$ και $\lambda_1 q_1 + \dots + \lambda_n q_n = 0$. Τότε, για $j \in \{1, \dots, n\}$, έχουμε

$$0 = (0, q_j) = (\lambda_1 q_1 + \dots + \lambda_n q_n, q_j) = \sum_{i=1}^n \lambda_i \delta_{ji} = \lambda_j.$$

Ασταθής Αλγόριθμος Gram-Schmidt

Έστω ένα σύνολο ανεξάρτητων διανυσμάτων $\{a_1, a_2, \dots, a_n\}$ και μία ορθοκανονική βάση $\{q_1, q_2, \dots, q_n\}$. Μπορούμε να ορίσουμε την ορθοκανονικοποίηση Gram-Schmidt ως:

$$\begin{aligned}
 q_1 &= a_1, \\
 q_2 &= a_2 - \frac{\langle a_2, q_1 \rangle}{\langle q_1, q_1 \rangle} q_1 \\
 q_3 &= a_3 - \frac{\langle a_3, q_1 \rangle}{\langle q_1, q_1 \rangle} q_1 - \frac{\langle a_3, q_2 \rangle}{\langle q_2, q_2 \rangle} q_2 \\
 &\vdots \\
 q_n &= a_n - \sum_{j=1}^{n-1} \frac{\langle a_n, q_j \rangle}{\langle q_j, q_j \rangle} q_j.
 \end{aligned} \tag{2.1}$$

Στη συνέχεια παίρνουμε

$$q_i = \frac{q_i}{\|q_i\|}.$$

Ο αλγόριθμος αυτός κατασκευάζει ορθοκανονικά $\{q_i\}_{i=1, \dots, n}$.

Όταν αυτή η διαδικασία εκτελείται σε υπολογιστή, συχνά τα διανύσματα q_i δεν είναι αρκετά ορθογώνια, λόγω σφαλμάτων στρογγυλοποίησης. Αυτή η απώλεια ορθογωνιότητας είναι ιδιαίτερα σοβαρή, γι' αυτό και η (κλασική) μέθοδος Gram-Schmidt χαρακτηρίζεται αριθμητικά ασταθής.

Ο αλγόριθμος για την ασταθής Gram-Schmidt είναι:

```

q1=a1 # Initialize the first vector
for j=2, ..., n:
    for k=1, ..., j-1:
        qj=a_j-(a_j, qk)/(qk, qk) * qk # Project and subtract using qk
    end
    qj=qj/||qj|| # Normalize the updated vector
end

```

Ευσταθής Αλγόριθμος Gram-Schmidt

Έστω ένα σύνολο ανεξάρτητων διανυσμάτων $\{a_1, a_2, \dots, a_n\}$ και μία ορθοκανονική βάση $\{q_1, q_2, \dots, q_n\}$. Μπορούμε να ορίσουμε την ορθοκανονικοποίηση Gram-Schmidt ως:

$$\begin{aligned}
\hat{q}_1 &= \frac{a_1}{\|a_1\|}, \\
\hat{q}_2 &= \frac{a_2 - \langle a_2, \hat{q}_1 \rangle \hat{q}_1}{\|a_2 - \langle a_2, \hat{q}_1 \rangle \hat{q}_1\|}, \\
\hat{q}_3 &= \frac{a_3 - \langle a_3, \hat{q}_1 \rangle \hat{q}_1 - \langle a_3, \hat{q}_2 \rangle \hat{q}_2}{\|a_3 - \langle a_3, \hat{q}_1 \rangle \hat{q}_1 - \langle a_3, \hat{q}_2 \rangle \hat{q}_2\|}, \\
&\vdots \\
\hat{q}_n &= \frac{a_n - \sum_{j=1}^{n-1} \langle a_n, \hat{q}_j \rangle \hat{q}_j}{\|a_n - \sum_{j=1}^{n-1} \langle a_n, \hat{q}_j \rangle \hat{q}_j\|}.
\end{aligned} \tag{2.2}$$

Η ευσταθής Gram-Schmidt χρησιμοποιεί τα κανονικοποιημένα ορθογώνια διανύσματα που έχουν υπολογιστεί προηγουμένως, για να υπολογίσει το άθροισμα. Αντίθετα, η κανονική μορφή της Gram-Schmidt χρησιμοποιεί τα αρχικά.

Ο αλγόριθμος για την ευσταθής Gram-Schmidt είναι:

```

q1 = a1 / ||a1|| # Normalize the first vector
for j=2,...,n:
    u = a_j
    for k = 1,...,j - 1:
        u -= (u, qk) / (qk, qk) * qk # Project and subtract using normalized qk
    qj = u / ||u|| # Normalize the updated vector
end
end

```

2.2 Παραγοντοποίηση QR

Θα ορίσουμε τώρα την Παραγοντοποίηση QR με τη βοήθεια της Gram-Schmidt.

Ελαττωμένη Παραγοντοποίηση QR (Reduced QR Factorization)

Υποθέτουμε ότι $A \in \mathbb{C}^{m \times n}$ ($m \geq n$) είναι μέγιστου βαθμού n . Θέλουμε η ακολουθία q_1, q_2, \dots να έχει την ιδιότητα

$$\langle q_1, q_2, \dots, q_j \rangle = \langle a_1, a_2, \dots, a_j \rangle, \quad j = 1, \dots, n,$$

όπου οι στήλες $a_1, a_2, \dots \in A$.

Παρατηρούμε ότι αυτό συνεπάγεται

$$\left[\begin{array}{c|c|c|c|c} a_1 & a_2 & \dots & a_n & \\ \hline \end{array} \right] = \left[\begin{array}{c|c|c|c|c} q_1 & q_2 & \dots & q_n & \\ \hline \end{array} \right] \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & r_{22} & & \\ & & \dots & \vdots \\ & & & r_{nn} \end{bmatrix},$$

όπου τα διαγώνια στοιχεία r_{kk} είναι μη μηδενικά, άρα τα a_1, \dots, a_k μπορούν να εκφραστούν ως γραμμικοί συνδιασμοί των q_1, \dots, q_k . Η αντιστρεψιμότητα του πάνω αριστερά $k \times k$ μπλοκ, του τριγωνικού πίνακα, υποδηλώνει αντίστοιχα, ότι τα q_1, \dots, q_k μπορούν να εκφραστούν ως γραμμικοί συνδιασμοί των a_1, \dots, a_k . Οι εξισώσεις αυτές γράφονται ως

$$\begin{aligned} a_1 &= r_{11}q_1, \\ a_2 &= r_{12}q_1 + r_{22}q_2, \\ a_3 &= r_{13}q_1 + r_{23}q_2 + r_{33}q_3, \\ &\vdots \\ a_n &= r_{1n}q_1 + r_{2n}q_2 + \dots + r_{nn}q_n. \end{aligned} \tag{2.3}$$

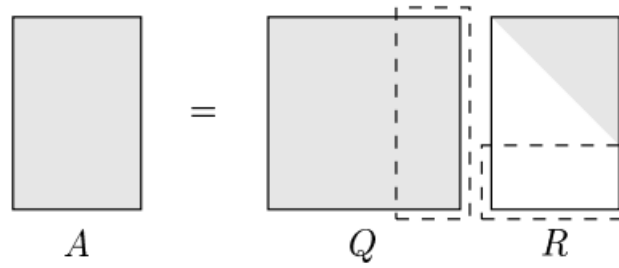
Οπότε, σε μορφή πίνακα έχουμε

$$A = \hat{Q}\hat{R},$$

όπου ο \hat{Q} είναι $m \times n$ με ορθομοναδιαίες στήλες και ο \hat{R} είναι $n \times n$ και άνω-τριγωνικός.

Παραγοντοποίηση QR (Full QR Factorization)

Σε αυτή την περίπτωση, προσθέτουμε στον πίνακα \hat{Q} , $m - n$ ορθομοναδιαίες στήλες, οπότε προκύπτει ένας μοναδιαίος $m \times m$ πίνακας Q . Επίσης, προσθέτουμε μηδενικές σειρές στον \hat{R} , οπότε προκύπτει ένας $m \times n$ άνω τριγωνικός πίνακας R .



Σχήμα 2.1: QR και Ελλατωμένη QR

Θεώρημα 2.9. [Theorem 7.1, [2]]

Κάθε πίνακας $A \in \mathbb{C}^{m \times n}$ ($m \geq n$) έχει παραγοντοποίηση QR, οπότε έχει επίσης και ελαττωμένη παραγοντοποίηση QR.

Απόδειξη

Υποθέτουμε ότι ο A είναι πλήρους βαθμού και ότι θέλουμε μόνο την ελαττωμένη παραγοντοποίηση QR. Σε αυτή την περίπτωση, η απόδειξη της ύπαρξης προκύπτει αυτόματα από τον αλγόριθμο Gram-Schmidt. Κατασκευαστικά, η διαδικασία αυτή παράγει τις ορθομοναδιαίες στήλες του \hat{Q} και τα στοιχεία του \hat{R} , ώστε να κατασκευαστεί η σχέση $A = \hat{Q}\hat{R}$. Η μόνη περίπτωση αποτυχίας είναι μόνο αν σε κάποιο βήμα, το $u_j = a_j - (q_1^T a_j)q_1 - (q_2^T a_j)q_2 - \dots - (q_{j-1}^T a_j)q_{j-1}$ είναι μηδέν, οπότε δε μπορεί να κανονικοποιηθεί ώστε να παράγει το q_j . Όμως αυτό σημαίνει ότι $a_j \in \langle q_1, \dots, q_{j-1} \rangle = \langle a_1, \dots, a_{j-1} \rangle$, το οποίο αντιτίθεται στην υπόθεση ότι ο A είναι πλήρους βαθμού.

Τώρα υποθέτουμε ότι ο A δεν είναι πλήρους βαθμού. Τότε σε ένα ή περισσότερα βήματα j , βρίσκουμε ότι $u_j = 0$. Άρα παίρνουμε τυχαίο q_j , ώστε να είναι οποιοδήποτε κανονικοποιημένο διάνυσμα διαγώνιο στο $\langle q_1, \dots, q_{j-1} \rangle$ και συνεχίζουμε με τη διαδικασία Gram-Schmidt.

Τέλος, η παραγοντοποίηση QR ενός $m \times n$ πίνακα, με $m \geq n$, μπορεί να κατασκευαστεί εισάγωντας τυχαία ορθομοναδιαία διανύσματα με τον ίδιο τρόπο. Ακολουθούμε τη διαδικασία Gram-Schmidt μέχρι το βήμα n , τότε συνεχίζουμε με τα επιπλέον $m - n$ βήματα, εισάγοντας διανύσματα q_j σε κάθε βήμα.

Θεώρημα 2.10. [Theorem 7.2, [2]]

Κάθε $A \in \mathbb{C}^{m \times n}$ ($m \geq n$) πλήρους βαθμού, έχει μοναδική ελαττωμένη παραγοντοποίηση QR, $A = \hat{Q}\hat{R}$ με $r_{jj} > 0$.

Απόδειξη

Θα χρησιμοποιήσουμε πάλι την επανάληψη Gram-Schmidt. Η σχέση (2.3) μπορεί να γραφτεί

ως

$$\begin{aligned}
 q_1 &= \frac{a_1}{r_{11}}, \\
 q_2 &= \frac{a_2 - r_{12}q_1}{r_{22}}, \\
 q_3 &= \frac{a_3 - r_{13}q_1 - r_{23}q_2}{r_{33}}, \\
 &\vdots \\
 q_n &= \frac{a_n - \sum_{i=1}^{n-1} r_{in}q_i}{r_{nn}}.
 \end{aligned} \tag{2.4}$$

Ορίζουμε τους συντελεστές r_{ij} στον αριθμητή των (2.4) ως

$$r_{ij} = q_i^T a_j \quad (i \neq j). \tag{2.5}$$

Οι συντελεστές r_{jj} στον παρονομαστή των (2.5) διαλέγονται για κανονικοποίηση

$$|r_{jj}| = \|a_j - \sum_{i=1}^{j-1} r_{ij}q_i\|_2. \tag{2.6}$$

Από τη σχέση $A = \hat{Q}\hat{R}$, την ορθοκανονικότητα των στηλών του \hat{Q} και την άνω-τριγωνικότητα του \hat{R} , προκύπτει ότι κάθε ελαττωμένη παραγοντοποίηση QR του A, πρέπει να ικανοποιεί τις σχέσεις (2.4-2.6). Από την υπόθεση ότι ο A είναι πλήρους βαθμού, οι παρονομαστές (2.6) των (2.4) είναι μη-μηδενικοί, οπότε σε κάθε επόμενο βήμα j , αυτές οι σχέσεις καθορίζουν πλήρως τα r_{ij} και q_j , εκτός από μία θέση, το πρόσημο του r_{jj} που δεν καθορίζεται από την (2.6). Αμέσως μόλις αυτό λυθεί, από την συνθήκη $r_{jj} > 0$, η παραγοντοποίηση είναι πλήρως καθορισμένη.

2.3 Householder

Οι μετασχηματισμοί Householder χρησιμοποιούνται ευρέως στην αριθμητική γραμμική άλγεβρα, για παράδειγμα, για την εκμηδένιση των στοιχείων κάτω από την κύρια διαγώνιο ενός πίνακα, για την εκτέλεση αναλύσεων QR και στο πρώτο βήμα του αλγορίθμου QR.

Ορισμός

Ένας πίνακας $n \times n$ της μορφής:

$$H = I - \frac{2uu^T}{u^T u}, \quad u \neq 0, \quad u \in \mathbb{R}^n, \tag{2.7}$$

λέγεται πίνακας Householder ή μετασχηματισμός Householder.

Παρατηρούμε ότι ο πίνακας uu^T είναι $\text{rank}=1$.

Εάν θέσουμε $u^T u = \|u\|_2^2$ και $v = \frac{u}{\|u\|_2}$, τότε ο πίνακας H μπορεί να πάρει τη μορφή:

$$H' = I - 2 \frac{uu^T}{\|u\|_2 \|u\|_2} = I - 2vv^T, \quad v \in \mathbb{R}^n, \quad \|v\|_2^2 = 1$$

Βασικές Ιδιότητες:

1. Ο πίνακας Householder είναι συμμετρικός. Πράγματι

$$(H')^T = I - (2vv^T)^T = I - 2vv^T = H'.$$

2. Ο πίνακας Householder είναι ορθογώνιος

$$(H')^T(H') = (H')^2 = (I - 2vv^T)(I - 2vv^T) = I - 4vv^T + 4vv^T vv^T = I - 4vv^T + 4vv^T = I,$$

αφού $vv^T = \|v\|_2^2 = 1$.

3. Οι πίνακες Householder λέγονται και στοιχειώδεις πίνακες ανάκλασης.

Μετασχηματισμός Householder και Παραγοντοποίηση QR

Υποθέτουμε έναν πίνακα A 5×3 και έναν πίνακα Q_k , ο οποίος επιλέγεται για να εισάγει μηδενικά κάτω από τη διαγώνιο στην k -ιοστή στήλη, ενώ διατηρεί όλα τα μηδενικά που έχουν εισαχθεί προηγουμένως. Συγκεκριμένα, για τον πίνακα A , εφαρμόζουμε τρεις φορές τον Q_k :

$$\begin{array}{c} \begin{bmatrix} x & x & x \\ x & x & x \\ x & x & x \\ x & x & x \\ x & x & x \end{bmatrix} \\ A \end{array} \xrightarrow{Q_1} \begin{array}{c} \begin{bmatrix} x & x & x \\ \mathbf{0} & x & x \\ \mathbf{0} & x & x \\ \mathbf{0} & x & x \\ \mathbf{0} & x & x \end{bmatrix} \\ Q_1 A \end{array} \xrightarrow{Q_2} \begin{array}{c} \begin{bmatrix} x & x & x \\ & x & x \\ \mathbf{0} & x & \\ \mathbf{0} & x & \\ \mathbf{0} & x & \end{bmatrix} \\ Q_2 Q_1 A \end{array} \xrightarrow{Q_3} \begin{array}{c} \begin{bmatrix} x & x & x \\ & x & x \\ & & x \\ & & \mathbf{0} \\ & & \mathbf{0} \end{bmatrix} \\ Q_3 Q_2 Q_1 A \end{array} \quad (2.8)$$

Αρχικά ο Q_1 εφαρμόζεται στις σειρές 1, ..., 5 εισάγοντας μηδενικά στις θέσεις (2,1), (3,1), (4,1), (5,1).

Στη συνέχεια, ο Q_2 εφαρμόζεται στις σειρές 2, ..., 5 εισάγοντας μηδενικά στις θέσεις (3,2), (4,2), (5,2),

χωρίς όμως να «καταστρέφει» τα μηδενικά που εισήγαγε ο Q_1 . Τέλος, ο Q_3 εφαρμόζεται στις σειρές 3, ..., 5 εισάγοντας μηδενικά στις θέσεις (4,3), (5,3), χωρίς να «καταστρέφει» τα μηδενικά που εισήχθησαν νωρίτερα.

Γενικότερα, ο Q_k εφαρμόζεται στις σειρές k, \dots, m . Στην αρχή του βήματος k , υπάρχει ένα block

από μηδενικά στις πρώτες $k-1$ στήλες αυτών των σειρών. Η εφαρμογή του Q_k δημιουργεί τους γραμμικούς συνδυασμούς αυτών των σειρών και οι γραμμικοί συνδυασμοί των μηδενικών στοιχείων παραμένουν μηδέν. Μετά από n βήματα, όλα τα στοιχεία κάτω από τη διαγώνιο έχουν διαγραφεί και ο $Q_n \dots Q_2 Q_1 A = R$ είναι άνω τριγωνικός.

Ο Αλγόριθμος [Lecture 10,[2]]

Θα περιγράψουμε τον αλγόριθμο Householder, όπως θα τον αναπαριστούσαμε στο matlab.

Αν έχουμε έναν πίνακα A , ορίζουμε $A_{i:i',j:j'}$ να είναι ο $(i' - i + 1) \times (j' - j + 1)$ υποπίνακας του A με στοιχεία της πάνω αριστεράς γωνίας a_{ij} και στοιχεία της κάτω δεξιάς γωνίας $a_{i',j'}$. Σε αυτή την ειδική περίπτωση, όπου ο υποπίνακας μειώνεται σε ένα υποδιάνυσμα μίας γραμμής ή μίας στήλης, γράφουμε $A_{i,j:j'}$ ή $A_{i:i',j}$ αντίστοιχα.

Ο παρακάτω αλγόριθμος υπολογίζει τον παράγοντα R της QR παραγοντοποίησης ενός πίνακα A $m \times n$ με $m \geq n$, αφήνοντας το αποτέλεσμα στη θέση του A . Στην πορεία, η διανύσματα ανάκλασης v_1, \dots, v_n αποθηκεύονται για μελλοντική χρήση.

Αλγόριθμος 1: Householder QR Παραγοντοποίηση

Για $k = 1, \dots, n$
 $x = A_{k:m,k}$
 $v_k = \text{sign}(x_1) \|x\|_2 e_1 + x$
 $v_k = v_k / \|v_k\|_2$
 $A_{k:m,k:n} = A_{k:m,k:n} - 2v_k(v_k^T A_{k:m,k:n})$

Μετά από αυτή τη διαδικασία, ο πίνακας A έχει μειωθεί στην άνω-τριγωνική του μορφή. Αυτός είναι ο πίνακας R στην παραγοντοποίηση QR, $A = QR$. Ο μοναδιαίος πίνακας Q , δεν έχει κατασκευαστεί, ούτε έχει τον υποπίνακα με n -στήλες \hat{Q} που αντιστοιχεί στη μειωμένη παραγοντοποίηση QR. Αυτό συμβαίνει γιατί η κατασκευή του Q ή του \hat{Q} χρειάζεται επιπλέον βήματα και πολλές εφαρμογές, το οποίο μπορούμε να αποφύγουμε αν δουλέψουμε απευθείας με τη σχέση:

$$Q^T = Q_n \dots Q_2 Q_1 \tag{2.9}$$

ή με τον συζυγή του

$$Q = Q_1 Q_2 \dots Q_n.$$

Χρησιμοποιώντας τη σχέση (2.9) μπορούμε να υπολογίσουμε το $Q^T b$ με μία σειρά από n πράξεις που εφαρμόζονται στο b . Είναι οι ίδιες πράξεις που εφαρμόστηκαν στον A για να τον κάνουν τριγωνικό.

Ο αλγόριθμος για αυτό είναι:

Αλγόριθμος 2: Έμμεσος υπολογισμός του γινομένου $Q^T b$

Για $k = 1, \dots, n$

$$b_{k:m} = b_{k:m} - 2v_k(v_k^T b_{k:m})$$

Αντίστοιχα, ο υπολογισμός του γινομένου Qx μπορεί να επιτευχθεί με την ίδια διαδικασία να εκτελείται με αντίστροφη σειρά.

Αλγόριθμος 3: Έμμεσος υπολογισμός του γινομένου $Q^T x$

Για $k = n, \dots, 1$

$$x_{k:m} = x_{k:m} - 2v_k(v_k^T x_{k:m})$$

Παράδειγμα

Θα χρησιμοποιήσουμε τον παρακάτω αλγόριθμο στο matlab για να βρούμε την παραγοντοποίηση QR μέσω των πινάκων Householder

```

1  %Function for Householder
2  function u = house_gen(x)
3      % u = house_gen(x)
4      % Generate Householder reflection.
5      % u = house_gen(x) returns u with norm(u) = sqrt(2), and
6      % H(u,x) = x - u*(u'*x) = +- norm(x)*e_1.
7
8      % Modify the sign function so that sign(0) = 1.
9      sig = @(u) sign(u) + (u==0);
10
11     nu = norm(x);
12     if nu ~= 0
13         u = x/nu;
14         u(1) = u(1) + sig(u(1));
15         u = u/sqrt(abs(u(1)));
16     else
17         u = x;

```

```
18         u(1) = sqrt(2);
19     end
20 end
21
22 %Function for Householder - QR
23 function [R,U] = house_qr(A)
24     % Householder reflections for QR decomposition.
25     % [R,U] = house_qr(A) returns
26     % R, the upper triangular factor, and
27     % U, the reflector generators for use by house_apply.
28     H = @(u,x) x - u*(u'*x);
29     [m,n] = size(A);
30     U = zeros(m,n);
31     R = A;
32     for j = 1:min(m,n)
33         u = house_gen(R(j:m,j));
34         U(j:m,j) = u;
35         R(j:m,j:n) = H(u,R(j:m,j:n));
36         R(j+1:m,j) = 0;
37     end
38 end
39
40 %Householder Application
41 function Z = house_apply(U,X)
42     % Apply Householder reflections.
43     % Z = house_apply(U,X), with U from house_qr
44     % computes Q*X without actually computing Q.
45     H = @(u,x) x - u*(u'*x);
46     Z = X;
47     [~,n] = size(U);
48     for j = n:-1:1
49         Z = H(U(:,j),Z);
50     end
51 end
```

Για έναν τυχαίο πίνακα A διάστασης 6×6 παίρνουμε

```
1 A=rand(6)
2 A =
3
4     0.8147     0.2785     0.9572     0.7922     0.6787     0.7060
5     0.9058     0.5469     0.4854     0.9595     0.7577     0.0318
6     0.1270     0.9575     0.8003     0.6557     0.7431     0.2769
7     0.9134     0.9649     0.1419     0.0357     0.3922     0.0462
8     0.6324     0.1576     0.4218     0.8491     0.6555     0.0971
9     0.0975     0.9706     0.9157     0.9340     0.1712     0.8235
10
11 [R,U] = house_qr(A)
12
13 R =
14
15    -1.6565    -1.1588    -1.0907    -1.3634    -1.2817    -0.4969
16         0     -1.3596    -0.8286    -0.6385    -0.3513    -0.5608
17         0         0    -0.9760    -0.9745    -0.4221    -0.7252
18         0         0         0     0.5972     0.1664    -0.1969
19         0         0         0         0         0    -0.4773     0.3675
20         0         0         0         0         0         0    -0.0688
21
22
23 U =
24
25     1.2214         0         0         0         0         0
26     0.4477     1.0074         0         0         0         0
27     0.0628     0.6452     1.1586         0         0         0
28     0.4514     0.3166    -0.7004    -1.2982         0         0
29     0.3125    -0.1534    -0.0145     0.5588     1.1627         0
30     0.0482     0.6672     0.4083    -0.0508    -0.8051     1.4142
31
32 I = eye(size(U));
```

```
33 Q = house_apply(U,I)
34
35 Q =
36
37     -0.4918     0.2143    -0.6130    -0.5675     0.0852     0.0839
38     -0.5468     0.0638     0.0596     0.5237    -0.0363     0.6463
39     -0.0767    -0.6389    -0.1919    -0.0732    -0.7366     0.0339
40     -0.5514    -0.2397     0.6743    -0.3550     0.1153    -0.2108
41     -0.3817     0.2094    -0.1833     0.4751    -0.1746    -0.7215
42     -0.0589    -0.6637    -0.3090     0.2157     0.6364    -0.0949
43
44 QR = Q*R
45
46 QR =
47
48     0.8147     0.2785     0.9572     0.7922     0.6787     0.7060
49     0.9058     0.5469     0.4854     0.9595     0.7577     0.0318
50     0.1270     0.9575     0.8003     0.6557     0.7431     0.2769
51     0.9134     0.9649     0.1419     0.0357     0.3922     0.0462
52     0.6324     0.1576     0.4218     0.8491     0.6555     0.0971
53     0.0975     0.9706     0.9157     0.9340     0.1712     0.8235
```

Άρα βλέπουμε ότι παίρνουμε ξανά τον πίνακα A ($QR=A$).

Κεφάλαιο **3**

Μέθοδος Ελαχίστων Τετραγώνων

3.1 Ανασκόπηση

Η μέθοδος των ελαχίστων τετραγώνων είναι μια μαθηματική τεχνική που χρησιμοποιείται για την εύρεση της βέλτιστης γραμμικής προσέγγισης σε ένα σύνολο σημείων δεδομένων. Ειδικότερα είναι ιδιαίτερα χρήσιμη όταν πρόκειται για υπερκαθορισμένα συστήματα, όπου οι εξισώσεις είναι περισσότερες από τους αγνώστους.

Στη μέθοδο αυτή, ο στόχος είναι η ελαχιστοποίηση του αθροίσματος των τετραγωνικών διαφορών μεταξύ των παρατηρούμενων τιμών και των τιμών που προβλέπει το γραμμικό μοντέλο. Η λύση της, προσδιορίζεται με την προσαρμογή των παραμέτρων του γραμμικού μοντέλου, ώστε να ελαχιστοποιηθεί το συνολικό σφάλμα.

Η μέθοδος των ελαχίστων τετραγώνων εφαρμόζεται ευρέως σε διάφορους τομείς, όπως η στατιστική, η ανάλυση δεδομένων και η μηχανική μάθηση, όπου χρησιμεύει ως βάση για την ανάλυση παλινδρόμησης, την προσαρμογή καμπυλών και τη βελτιστοποίηση μοντέλων.

Ιστορικά Στοιχεία

Η μέθοδος των ελαχίστων τετραγώνων εισήχθη από τον Γάλλο μαθηματικό και αστρονόμο Adrien-Marie Legendre στις αρχές του 19ου αιώνα. Ο Legendre την εφάρμοσε στο πρόβλημα του προσδιορισμού των τροχιών των ουράνιων σωμάτων, με βάση τα δεδομένα που είχε παρατηρήσει. Αυτή ήταν και η πρώτη χρήση της μεθόδου.

Παρόλο που ο Legendre αναγνωρίζεται για την επισημοποίηση της μεθόδου, ο Γερμανός μαθηματικός και φυσικός Carl Friedrich Gauss ανέπτυξε ανεξάρτητα τη μέθοδο περίπου την ίδια εποχή.

Το έργο του Gauss παρέμεινε αδημοσίευτο για πολλά χρόνια, αλλά σήμερα αναγνωρίζεται ως συνιδρυτής της μεθόδου. Ο Gauss εισήγαγε την έννοια των κανονικών εξισώσεων, η οποία είναι σημαντική για την εξαγωγή της λύσης των ελαχίστων τετραγώνων.

3.2 Μαθηματική διατύπωση

Θεωρούμε ένα γραμμικό σύστημα εξισώσεων με n αγνώστους και $m > n$ εξισώσεις. Θέλουμε να βρούμε ένα διάνυσμα $x \in \mathbb{R}^n$, το οποίο να ικανοποιεί τη σχέση $Ax = b$, όπου $A \in \mathbb{R}^{m \times n}$ και $b \in \mathbb{R}^m$. Γενικά αυτά τα προβλήματα δεν έχουν λύση. Ένα κατάλληλο διάνυσμα x , υπάρχει μόνο αν το b ανήκει στο $\text{range}(A)$ και αφού το b είναι ένα m -διάνυσμα, ενώ το $\text{range}(A)$ είναι διάστασης το πολύ n . Αυτό αληθεύει μόνο για εξαιρετικές επιλογές του b .

Λέμε ότι ένα ορθογώνιο σύστημα εξισώσεων με $m > n$ είναι υπερκαθορισμένο. Το διάνυσμα $r = b - Ax \in \mathbb{R}^m$, που είναι γνωστό ως το υπόλοιπο, μπορεί να γίνει πολύ μικρό με την κατάλληλη επιλογή του x , αλλά γενικά δε μπορεί να γίνει ίσο με μηδέν.

Αφού λοιπόν, δε μπορούμε να κάνουμε το υπόλοιπο ίσο με μηδέν, τότε θα προσπαθήσουμε να το κάνουμε όσο πιο μικρό γίνεται. Η μέτρηση για το μικρότερο r υπονοεί την επιλογή μίας νόρμας. Αν διαλέξουμε μία 2-νόρμα, το πρόβλημα λαμβάνει την παρακάτω μορφή:

$$\begin{aligned} &\text{Δεδομένου του } A \in \mathbb{R}^{m \times n}, m \geq n, b \in \mathbb{R}^m, \\ &\text{βρίσκουμε } x \in \mathbb{R}^n \text{ τέτοιο ώστε το } \|b - Ax\|_2 \text{ να ελαχιστοποιείται.} \end{aligned} \quad (3.1)$$

Περίπτωση γραμμικής παλινδρόμησης

Συγκεκριμένα για τη γραμμική παλινδρόμηση, η οποία είναι και η πιο συχνή χρήση της μεθόδου των ελαχίστων τετραγώνων, το πρόβλημα εκφράζεται ως:

Δεδομένου ενός συνόλου σημείων δεδομένων (x_i, y_i) για $i=1,2,\dots,n$, όπου τα x_i είναι οι ανεξάρτητες μεταβλητές και y_i είναι οι αντίστοιχες παρατηρούμενες εξαρτημένες μεταβλητές, ο στόχος είναι να βρούμε ένα γραμμικό μοντέλο $y = ax + b$ που ταιριάζει καλύτερα στα δεδομένα. Το πρόβλημα των ελαχίστων τετραγώνων, είναι να ελαχιστοποιήσει το άθροισμα των τετραγωνικών διαφορών μεταξύ των παρατηρούμενων τιμών y_i και των τιμών που προβλέπει το μοντέλο $ax_i + b$. Μαθηματικά αυτό εκφράζεται ως:

$$\text{Ελαχιστοποίηση της } J(a, b) := \sum_{i=1}^n (y_i - (ax_i + b))^2 \quad (3.2)$$

Η αντικειμενική συνάρτηση $J(a, b)$ αντιπροσωπεύει το άθροισμα των τετραγωνικών υπολοίπων, όπου τα υπόλοιπα είναι οι διαφορές μεταξύ των παρατηρούμενων τιμών y_i και των τιμών που προβλέπονται από το γραμμικό μοντέλο. Η λύση των ελαχίστων τετραγώνων επιδιώκει να βρει τις τιμές των a και b που ελαχιστοποιούν αυτό το άθροισμα.

Εξαγωγή της λύσης της μεθόδου

Ξεκινώντας από τη σχέση $J(a, b) = \sum_{i=1}^n (y_i - (ax_i + b))^2$, παίρνουμε τις μερικές παραγώγους του J ως προς a και b . Συγκεκριμένα,

$$\frac{\partial J}{\partial a} = -2 \sum_{i=1}^n x_i (y_i - (ax_i + b))$$

$$\frac{\partial J}{\partial b} = -2 \sum_{i=1}^n (y_i - (ax_i + b))$$

Θέτουμε τις παραγώγους αυτές ίσες με το μηδέν και παίρνουμε τις λύσεις για τις τιμές των a και b , για τις οποίες ελαχιστοποιείται το άθροισμα. Δηλαδή,

$$\begin{pmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & n \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{pmatrix}. \quad (3.3)$$

Εύκολα βλέπουμε ότι η ορίζουσα Δ του πίνακα δίνεται από τη σχέση:

$$\Delta = \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^n (x_i - x_j)^2,$$

η οποία είναι διάφορη του μηδενός, αν δύο τουλάχιστον από τα σημεία x_i είναι διαφορετικά. Η λύση του συστήματος (3.3) είναι τότε

$$\begin{cases} a = \frac{1}{\Delta} [n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i] \\ b = \frac{1}{\Delta} [\sum_{i=1}^n x_i^2 \sum_{i=1}^n y_i - \sum_{i=1}^n x_i y_i \sum_{i=1}^n x_i] \end{cases} \quad (3.4)$$

Επομένως η (3.4) μας δίνει το μοναδικό σημείο στο οποίο η J λαμβάνει το ελάχιστό της.

3.3 Η γεωμετρία των ελαχίστων τετραγώνων

Η γεωμετρική ερμηνεία της μεθόδου ελαχίστων τετραγώνων παρέχει πληροφορίες για τη διαδικασία βελτιστοποίησης της μεθόδου και τη φύση της λύσης. Παράλληλα, προσφέρει μια οπτική κατανόηση του τρόπου με τον οποίο η μέθοδος προσαρμόζει βέλτιστα ένα μοντέλο στα παρατηρούμενα σημεία δεδομένων, στηριζόμενη στην ορθογώνια προβολή.

Ο στόχος μας, είναι να βρούμε το πλησιέστερο σημείο Ax στο $\text{range}(A)$ έως b , έτσι ώστε η νόρμα του υπολοίπου $r = b - Ax$ να ελαχιστοποιηθεί. Γεωμετρικά, είναι προφανές ότι αυτό θα συμβεί αν έχουμε $Ax = Pb$, όπου $P \in \mathbb{R}^{m \times m}$ είναι η ορθογώνια προβολή που επεικονίζει το \mathbb{R}^m στο $\text{range}(A)$. Με άλλα λόγια, το υπόλοιπο $r = b - Ax$ πρέπει να είναι ορθογώνιο στο $\text{range}(A)$.

Θεώρημα 3.11. [Theorem 11.1, [2]]

Έστω $A \in \mathbb{C}^{m \times n}$ ($m \geq n$) και $b \in \mathbb{C}^m$ γνωστό. Ένα διάνυσμα $x \in \mathbb{C}^n$ ελαχιστοποιεί τη νόρμα του υπολοίπου $\|r\|_2 = \|b - Ax\|_2$, επομένως λύνει το πρόβλημα των ελαχίστων τετραγώνων (3.1), αν και μόνο αν $r \perp \text{range}(A)$, δηλαδή

$$A^T r = 0, \quad (3.5)$$

ή ισοδύναμα

$$A^T Ax = A^T b, \quad (3.6)$$

ή ισοδύναμα

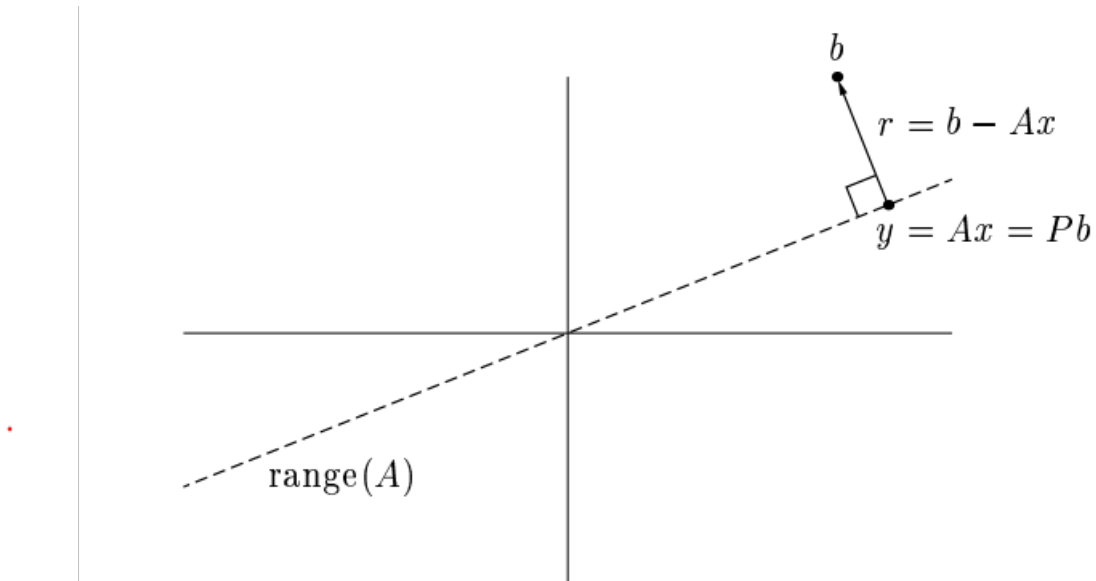
$$Pb = Ax, \quad (3.7)$$

όπου $P \in \mathbb{C}^{m \times m}$ είναι η ορθογώνια προβολή πάνω στο $\text{range}(A)$. Το $n \times n$ σύστημα εξισώσεων (3.6), γνωστό ως οι κανονικές εξισώσεις, είναι μη-ιδιάζον αν και μόνο αν ο A είναι πλήρους βαθμού. Συνεπώς, η λύση x είναι μοναδική αν και μόνο αν ο A είναι πλήρους βαθμού.

Απόδειξη

Η ισοδυναμία των σχέσεων (3.5) και (3.7) έπεται από τις ιδιότητες των ορθογώνιων προβολών, και η ισοδυναμία των σχέσεων (3.5) και (3.6) έπεται από τον ορισμό του r . Για να δείξουμε ότι το $y = Pb$ είναι το μοναδικό σημείο στο $\text{range}(A)$ που ελαχιστοποιεί το $\|b - y\|_2$, υποθέτουμε $z \neq y$ ένα άλλο σημείο στο $\text{range}(A)$. Αφού το $z - y$ είναι ορθογώνιο στο $b - y$, από Πυθαγόρειο Θεώρημα έχουμε $\|b - z\|_2^2 = \|b - y\|_2^2 + \|y - z\|_2^2 > \|b - y\|_2^2$, όπως απαιτείται. Τελικά, παρατηρούμε ότι αν ο $A^T A$ είναι ιδιάζον, τότε $A^T A = 0$ για κάποιο μη-μηδενικό x , το οποίο οδηγεί στο $x^T A^T Ax = 0$. Άρα $Ax = 0$, το οποίο συνεπάγεται ότι ο A δεν έχει βαθμό. Αντίθετα, αν ο A δεν έχει βαθμό, τότε $Ax = 0$ για κάποιο μη-μηδενικό x , που σημαίνει ότι $A^T Ax = 0$ επίσης, οπότε ο $A^T A$ είναι

ιδιάζον. Από τη σχέση (3.6), αυτός ο χαρακτηρισμός των μη-ιδιάζόντων πινάκων $A^T A$ συνεπάγεται τη δήλωση σχετικά με τη μοναδικότητα του x .



Σχήμα 3.1: Η μέθοδος ελαχίστων τετραγώνων σε σχέση με την ορθογώνια προβολή

3.4 Προσαρμογή πολυωνυμικών δεδομένων

Πολυωνυμική Παρεμβολή

Έστω ότι έχουμε m διαφορετικά σημεία $x_1, \dots, x_m \in \mathbb{R}$ και δεδομένα $y_1, \dots, y_m \in \mathbb{R}$ σε αυτά τα σημεία. Τότε, υπάρχει μοναδική πολυωνυμική παρεμβολή σε αυτά τα δεδομένα σε αυτά τα σημεία, η οποία είναι ένα πολώνυμο βαθμού το πολύ $m-1$,

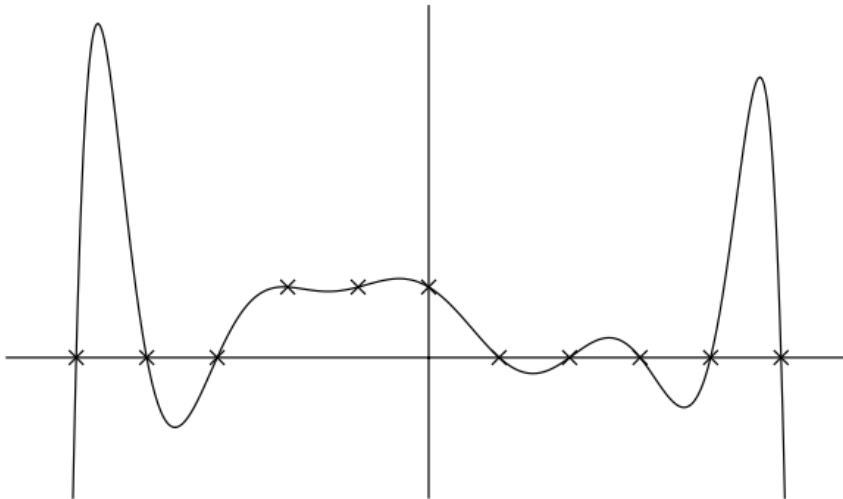
$$p(x) = c_0 + c_1x + \dots + c_{m-1}x^{m-1}, \quad (3.8)$$

με την ιδιότητα ότι σε κάθε x_i , $p(x_i) = y_i$. Η σχέση των δεδομένων $\{x_i\}$, $\{y_i\}$ με τους συντελεστές $\{c_i\}$ μπορεί να εκφραστεί από το τετραγωνικό σύστημα Vandermonde:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{m-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{m-1} \\ 1 & x_3 & x_3^2 & \dots & x_3^{m-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \dots & x_m^{m-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{m-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix}. \quad (3.9)$$

Για να ορίσουμε τους συντελεστές $\{c_i\}$ για ένα δοσμένο σύστημα δεδομένων, μπορούμε να λύσουμε αυτό το σύστημα εξισώσεων, το οποίο είναι εγγυημένο να είναι μη-ιδιάζον, όσο τα σημεία $\{x_i\}$ είναι διαφορετικά.

Το παρακάτω γράφημα παριστάνει ένα παράδειγμα αυτής της διαδικασίας της πολυωνυμικής προσέγγισης. Έχουμε 11 σημεία δεδομένων στη μορφή ενός διακριτού τετραγωνικού κύματος, που παριστάνονται με σταυρούς και την καμπύλη $p(x)$ η οποία περνάει από αυτά. Όμως, η προσαρμογή δεν είναι καθόλου ομαλή. Κοντά στα άκρα του διαστήματος, το $p(x)$ εμφανίζει μεγάλες ταλαντώσεις, οι οποίες είναι φανερά ένα τεχνούργημα της διαδικασίας παρεμβολής, όχι μία λογική αναπαράσταση των δεδομένων.



Σχήμα 3.2: Πολυώνυμο 10ου βαθμού που παρεμβάλλεται σε 11 σημεία δεδομένων

Είναι σύνηθες να παρατηρείται αυτή η μη-ικανοποιητική συμπεριφορά της πολυωνυμικής προσέγγισης. Οι προσεγγίσεις που παράγει είναι συνήθως κακές και τείνουν να χειροτερεύουν αντί να καλυτερεύουν, όταν χρησιμοποιούνται περισσότερα δεδομένα. Ακόμα και αν η προσέγγιση είναι καλή, η παρεμβολική διαδικασία μπορεί να επηρεαστεί, για παράδειγμα να είναι ευαίσθητη σε διαταραχές των δεδομένων. Για να αποφύγουμε αυτά τα προβλήματα, μπορούμε να χρησιμοποιήσουμε ένα ανομοιόμορφο σύνολο από σημεία παρεμβολής, όπως τα σημεία Chebyshev στο διάστημα $[-1,1]$. Παρόλο αυτά, σε εφαρμογές, δεν θα είναι πάντα εφικτό να διαλέξουμε ελεύθερα τα σημεία παρεμβολής.

Προσαρμογή Πολυωνυμικών Ελαχίστων Τετραγώνων

Χωρίς να αλλάζουμε τα σημεία δεδομένων, μπορούμε να μειώσουμε με καλύτερο τρόπο το βαθμό του πολυωνύμου. Έχοντας τα x_i, \dots, x_m και y_i, \dots, y_m ξανά, θεωρούμε τώρα ένα πολυώνυμο βαθμού $n-1$

$$p(x) = c_0 + c_1x + \dots + c_{n-1}x^{n-1} \quad (3.10)$$

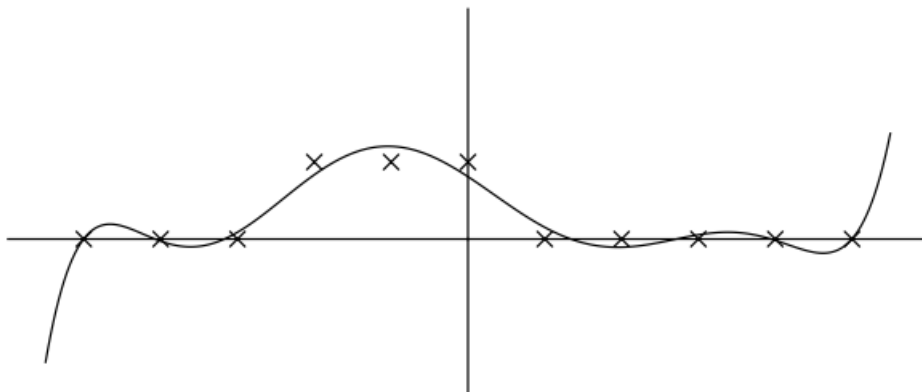
για κάποια $n < m$. Τέτοια πολυώνυμα, είναι μία προσαρμογή ελαχίστων τετραγώνων στα δεδομένα, αν ελαχιστοποιούν το άθροισμα των τετραγώνων των αποκλίσεων των δεδομένων,

$$\sum_{i=1}^m |p(x_i) - y_i|^2.$$

Αυτό το άθροισμα των τετραγώνων είναι ίσο με το τετράγωνο της νόρμας του υπολοίπου, $\|r\|_2^2$, του ορθογώνιου συστήματος Vandermonde

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} \\ 1 & x_3 & x_3^2 & \dots & x_3^{n-1} \\ \vdots & & & & \vdots \\ 1 & x_m & x_m^2 & \dots & x_m^{n-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{n-1} \end{bmatrix} \approx \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix}. \quad (3.11)$$

Το παρακάτω γράφημα απεικονίζει το αποτέλεσμα της προσέγγισης των ίδιων 11 δεδομένων της προηγούμενης περίπτωσης, με ένα πολυώνυμο βαθμού 7. Το καινούργιο πολυώνυμο δεν παρεμβάλεται στα δεδομένα, αλλά αποτυπώνει τη συνολική συμπεριφορά τους πολύ καλύτερα από το προηγούμενο πολυώνυμο. Επίσης, είναι λιγότερο ευαίσθητο σε διαταραχές.



Σχήμα 3.3: Πολυωνυμική προσέγγιση ελαχίστων τετραγώνων 7ου βαθμού στα ίδια δεδομένα

Παράδειγμα

Θα χρησιμοποιήσουμε τη μέθοδο της πολυωνυμικής παρεμβολής σε πίνακα μεγάλης διάστασης, ώστε να δείξουμε ότι δε λαμβάνουμε τα επιθυμητά αποτελέσματα.

Με χρήση του matlab έχουμε:

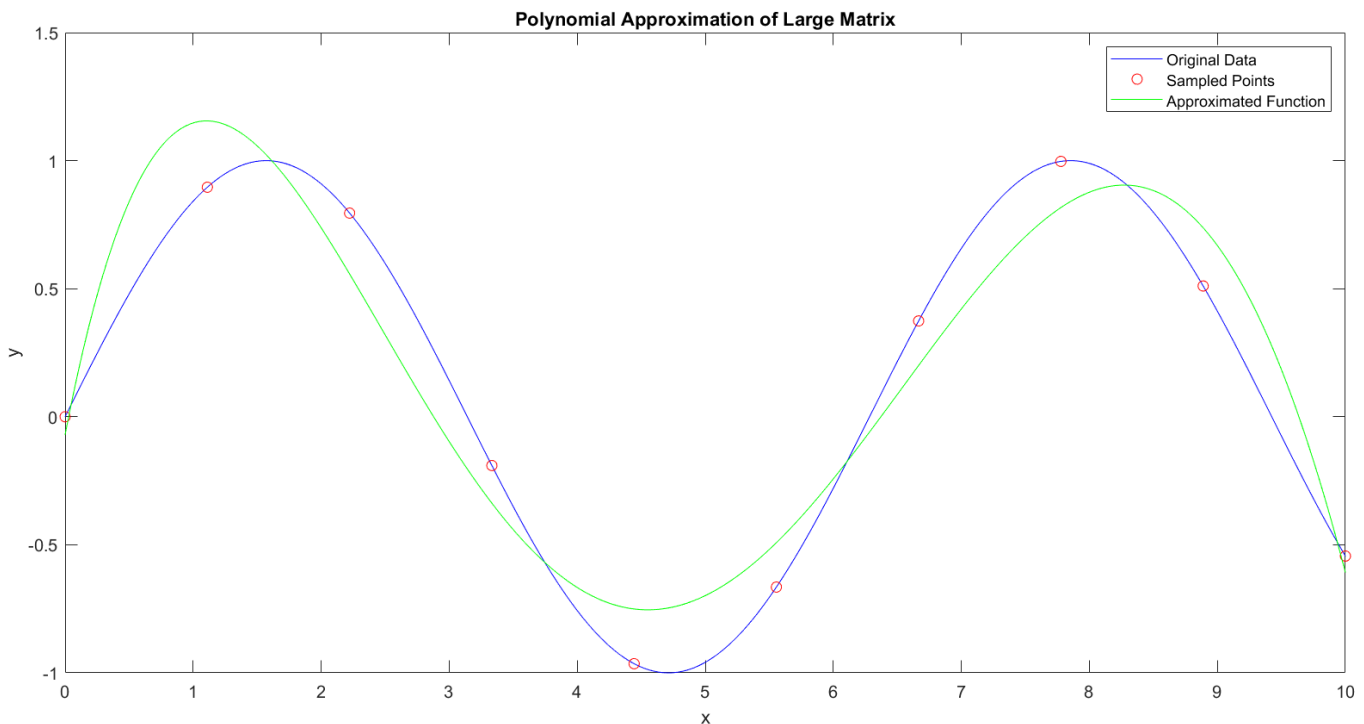
```
1 % Define a large matrix with a simple sine wave function
2 n = 1000; % Size of the matrix
3 x = linspace(0, 10, n);
4 y = sin(x);
5 data = [x; y]';
6
7 % Sample a smaller subset of data points for interpolation
8 sample_size = 10;
9 sample_indices = round(linspace(1, n, sample_size));
10 sampled_data = data(sample_indices, :);
11
12 % Perform polynomial approximation
13 interpolation_degree = 5;
14 p = polyfit(sampled_data(:,1), sampled_data(:,2),
15             interpolation_degree);
16 interpolated_y = polyval(p, x);
17
18 % Plot original data, sampled points, and approximated function
19 figure;
20 plot(x, y, 'b-', 'DisplayName', 'Original Data');
21 hold on;
22 plot(sampled_data(:,1), sampled_data(:,2), 'ro', 'DisplayName', '
23         Sampled Points');
24 plot(x, interpolated_y, 'g-', 'DisplayName', 'Interpolated
25         Function');
26 legend;
27 xlabel('x');
28 ylabel('y');
29 title('Polynomial Interpolation of Large Matrix');
```

```

28 % Observe the significant deviation of the interpolated function
    from the original data
29 hold off;

```

Οπότε, λαμβάνουμε το παρακάτω γράφημα:



Σχήμα 3.4: Πολυωνυμική προσέγγιση ελαχίστων τετραγώνων 5ου βαθμού σε πίνακα μεγάλων διαστάσεων

Παρατηρούμε λοιπόν, ότι η μέθοδος πολυωνυμικής παρεμβολής δε λειτουργεί με επιτυχία για πίνακες μεγάλων διαστάσεων και πολυώνυμα μεγάλου βαθμού. Αξίζει να σημειωθεί, ότι όσο αυξάνεται το μέγεθος του πίνακα και ο βαθμός του πολυωνύμου, λαμβάνουμε χειρότερα αποτελέσματα.

3.5 Κανονικές εξισώσεις, QR και SVD

Κανονικές εξισώσεις

Ο κλασικός τρόπος για την επίλυση προβλημάτων ελαχίστων τετραγώνων, είναι η επίλυση των κανονικών εξισώσεων. Αν ο A είναι πλήρους βαθμού, αυτό είναι ένα τετραγωνικό, ερμιτιανό, θετικά ορισμένο σύστημα εξισώσεων, διαστάσεων n . Η βασική μέθοδος επίλυσης τέτοιου συστήματος

είναι με την παραγοντοποίηση Cholesky. Αυτή η μέθοδος κατασκευάζει μία παραγοντοποίηση $A^T A = R^T R$, όπου ο R είναι άνω-τριγωνικός, που ανάγεται στις εξισώσεις:

$$R^T R x = A^T b. \quad (3.12)$$

Αλγόριθμος 1: Ελάχιστα Τετράγωνα μέσω των Κανονικών Εξισώσεων

1. Σχηματίζουμε τον πίνακα $A^T A$ και το διάνυσμα $A^T b$.
2. Υπολογίζουμε την παραγοντοποίηση Cholesky $A^T A = R^T R$.
3. Λύνουμε το κάτω-τριγωνικό σύστημα $R^T w = A^T b$ ως προς το w .
4. Λύνουμε το άνω-τριγωνικό σύστημα $R x = w$ ως προς το x .

Παραγοντοποίηση QR

Με την ορθοκανονικοποίηση Gram-Schmidt ή με την τριγωνοποίηση Householder, κατασκευάζουμε μία παραγοντοποίηση $A = \hat{Q}\hat{R}$. Ο ορθογώνιος τελεστής P μπορεί να γραφτεί ως $P = \hat{Q}\hat{Q}^T$, οπότε έχουμε

$$y = P b = \hat{Q}\hat{Q}^T b.$$

Αφού το $y \in \text{range}(A)$, το σύστημα $Ax = y$ έχει ακριβώς μία λύση. Συνδυάζοντας την παραγοντοποίηση QR και την παραπάνω σχέση παίρνουμε

$$\hat{Q}\hat{R}x = \hat{Q}\hat{Q}^T b, \quad (3.13)$$

και πολλαπλασιάζοντας με \hat{Q}^T λαμβάνουμε

$$\hat{R}x = \hat{Q}^T b. \quad (3.14)$$

Πολλαπλασιάζοντας με \hat{R}^{-1} παίρνουμε τη σχέση $A^\dagger = \hat{R}^{-1}\hat{Q}^T$ για τον ψευδοαντίστροφο. Η σχέση (3.14) είναι ένα άνω-τριγωνικό σύστημα, μη-ιδιάζον αν ο A είναι πλήρους βαθμού και επιλύεται εύκολα με αντίστροφη αντικατάσταση.

Αλγόριθμος 2: Ελάχιστα Τετράγωνα μέσω της Παραγοντοποίησης QR

1. Υπολογίζουμε την ελαττωμένη παραγοντοποίηση QR, $A = \hat{Q}\hat{R}$.
2. Υπολογίζουμε το διάνυσμα $\hat{Q}^T b$.
3. Λύνουμε το άνω-τριγωνικό σύστημα $\hat{R}^T x = \hat{Q}^T b$ ως προς το x .

Παρατηρούμε ότι η σχέση (3.14) μπορεί να παραχθεί επίσης από τις κανονικές εξισώσεις. Αν

$A^T A = A^T b$, τότε $\hat{R}^T \hat{Q}^T \hat{Q} \hat{R} = \hat{R}^T \hat{Q}^T b$, που συνεπάγεται $\hat{R}x = \hat{Q}^T b$.

SVD

Μία άλλη μέθοδος επίλυσης προβλημάτων ελαχίστων τετραγώνων είναι με τη χρήση της ελαττωμένης μεθόδου ανάλυσης πίνακα σε ιδιάζουσες τιμές $A = \hat{U} \hat{\Sigma} \hat{V}^T$. Τώρα ο P αναπαριστάται με τη μορφή $P = \hat{U} \hat{U}^T$, δίνοντας

$$y = Pb = \hat{U} \hat{U}^T b,$$

ενώ οι ανάλογες εξισώσεις των (3.13) και (3.14) είναι

$$\hat{U} \hat{\Sigma} \hat{V}^T x = \hat{U} \hat{U}^T b, \quad (3.15)$$

και

$$\hat{\Sigma} \hat{V}^T x = \hat{U}^T b. \quad (3.16)$$

Αλγόριθμος 3: Ελάχιστα Τετράγωνα μέσω της SVD

1. Υπολογίζουμε την ελαττωμένη SVD, $A = \hat{U} \hat{\Sigma} \hat{V}^T$.
2. Υπολογίζουμε το διάνυσμα $\hat{U}^T b$.
3. Λύνουμε το διαγώνιο σύστημα $\hat{\Sigma} w = \hat{U}^T b$ ως προς το w .
4. Θέτουμε $x = V w$.

Παρατηρούμε ότι ενώ η παραγοντοποίηση QR μειώνει το πρόβλημα ελαχίστων τετραγώνων σε ένα τριγωνικό σύστημα εξισώσεων, η SVD το μειώνει σε ένα διαγώνιο σύστημα εξισώσεων. Αν ο A είναι πλήρους βαθμού, το διαγώνιο σύστημα είναι μη-ιδιάζον.

Όπως και στην προηγούμενη περίπτωση, η σχέση (3.16) μπορεί να παραχθεί από τις κανονικές εξισώσεις. Αν $A^T A x = A^T b$, τότε $V \hat{\Sigma}^T \hat{U}^T \hat{U} \hat{\Sigma} V^T x = V \hat{\Sigma}^T \hat{U}^T b$, που συνεπάγεται $\hat{\Sigma} V^T x = \hat{U}^T b$.

Παράδειγμα

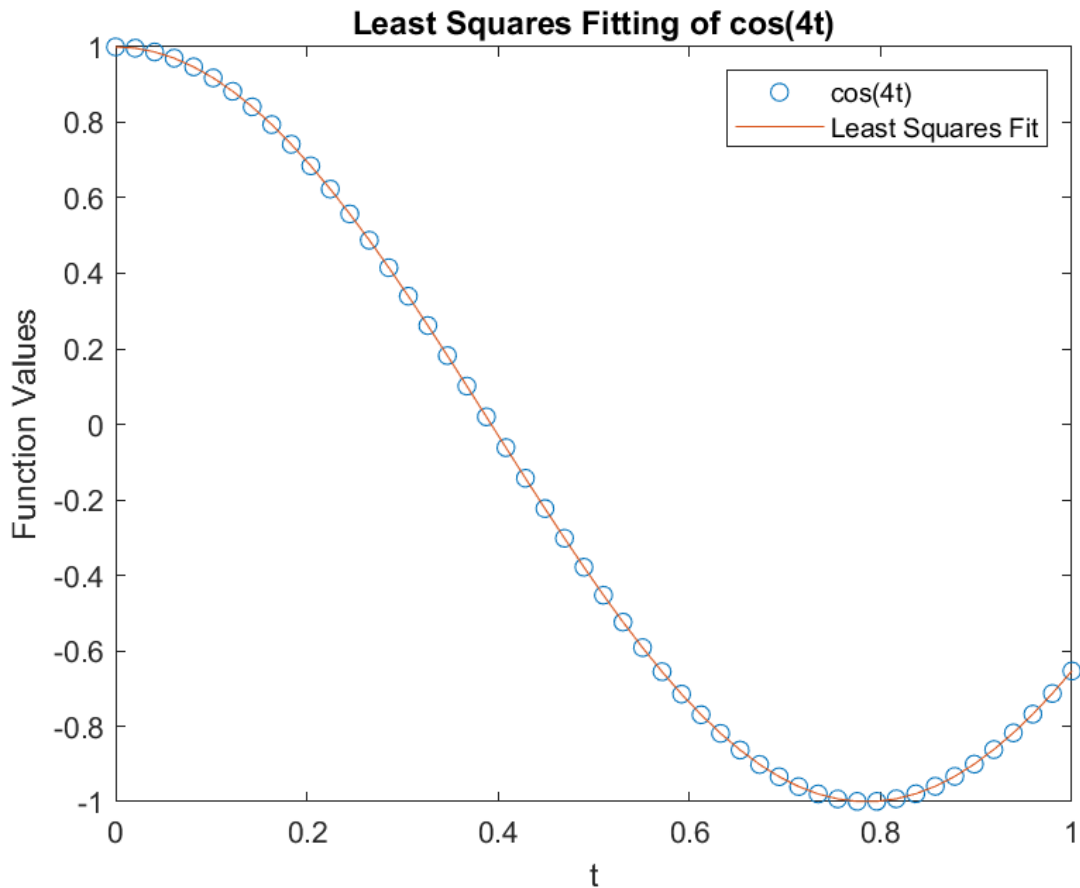
Θα κατασκευάσουμε, με τη χρήση του matlab, έναν πίνακα A $m \times n$ και θα πάρουμε t να είναι το m-διάνυσμα, που αντιστοιχεί σε γραμμικά καταναμημένα σημεία πλέγματος από 0 έως 1. Επίσης, ορίζουμε $b = \cos(4t)$ και παρουσιάζουμε το γράφημα.

```

1 % Define parameters
2 m = 50;
3 n = 12;

```

```
4
5 % Define linearly spaced grid points
6 t = linspace(0, 1, m)';
7
8 % Define the function cos(4t)
9 b = cos(4 * t);
10
11 % Construct the Vandermonde matrix
12 A = fliplr(vander(t));
13 A = A(:, 1:n); % Take the first n columns for a polynomial of
    degree n-1
14
15 % Perform least squares fitting
16 coefficients = A\b;
17
18 % Evaluate the polynomial using the coefficients
19 p = A * coefficients;
20
21 % Plot the results
22 figure;
23 plot(t, b, 'o', t, p, '-');
24 legend('cos(4t)', 'Least Squares Fit');
25 xlabel('t');
26 ylabel('Function Values');
27 title('Least Squares Fitting of cos(4t)');
```



Σχήμα 3.5: Προσαρμογή μοντέλου ελαχίστων τετραγώνων στη συνάρτηση $\cos(4t)$

Στη συνέχεια θα υπολογίσουμε τους συντελεστές της μεθόδου των ελαχίστων τετραγώνων με τους τρεις διαφορετικούς τρόπους που περιγράψαμε παραπάνω, καθώς και με τη μέθοδο Householder.

1. Μέθοδος κανονικών εξισώσεων
2. Παραγοντοποίηση QR με τη χρήση της μεθόδου `qr` του matlab
3. Μέθοδος SVD με τη χρήση της μεθόδου `svd` του matlab
4. Παραγοντοποίηση QR με χρήση των πινάκων Householder

```

1 % Method 1: Formation and solution of the normal equations
2 coefficients_normal_equations = (A' * A) \ (A' * b);
3
4 % Method 2: QR factorization computed by MATLAB's qr

```

```

5 [Q, R] = qr(A);
6 coefficients_qr = R \ (Q' * b);
7
8 % Method 3: SVD using MATLAB's svd
9 [U, S, V] = svd(A);
10 coefficients_svd = V * (S \ (U' * b));
11
12 % Method 4: QR factorization with Householder reflections
13 [U,R_house] = hqrd(A);
14 I = eye(size(U));
15 Q_house = house_apply(U,I);
16 coefficients_qr_house = R_house \ (Q_house' * b);
17
18 % Evaluate the least squares fits using the coefficients from each
    method
19 p_normal_equations = A * coefficients_normal_equations;
20 p_qr = A * coefficients_qr;
21 p_svd = A * coefficients_svd;
22 p_qr_house = A * coefficients_qr_house;
23
24 % Plot the results
25 figure;
26 plot(t, b, 'o', t, p_normal_equations, '-', t, p_qr, '--', t,
    p_svd, ':', t, p_qr_house, '-.');
27 legend('cos(4t)', 'Normal Equations Fit', 'QR Fit', 'SVD Fit', 'QR
    Householder Fit');
28 xlabel('t');
29 ylabel('Function Values');
30 title('Least Squares Fitting Comparison');
31
32 %Function hqrd
33 function [U,R] = hqrd(X)
34     % Householder triangularization. [U,R] = hqrd(X);
35     % Generators of Householder reflections stored in U.

```

```

36     % H_k = I - U(:,k)*U(:,k)'.
37     % prod(H_m ... H_1)X = [R; 0]
38     [n,p] = size(X);
39     U = zeros(size(X));
40     m = min(n,p);
41     R = zeros(m,m);
42     for k = 1:min(n,p)
43         [U(k:n,k),R(k,k)] = housegen(X(k:n,k));
44         v = U(k:n,k)'*X(k:n,k+1:p);
45         X(k:n,k+1:p) = X(k:n,k+1:p) - U(k:n,k)*v;
46         R(k,k+1:p) = X(k,k+1:p);
47     end
48 end
49
50 %Function housegen
51 function [u,nu] = housegen(x)
52     % Generate Householder reflection
53     % H = I - uu' with Hx = -+ nu e_1
54     % returns nu = norm(x)
55     u = x;
56     nu = norm(x);
57     if nu == 0
58         u(1) = sqrt(2);
59         return
60     end
61     u = x/nu;
62     if u(1) >= 0
63         u(1) = u(1) + 1;
64         nu = -nu;
65     else
66         u(1) = u(1) - 1;
67     end
68     u = u/sqrt(abs(u(1)));
69 end

```

Τα αποτελέσματα των συντελεστών που παίρνουμε από κάθε μέθοδο είναι:

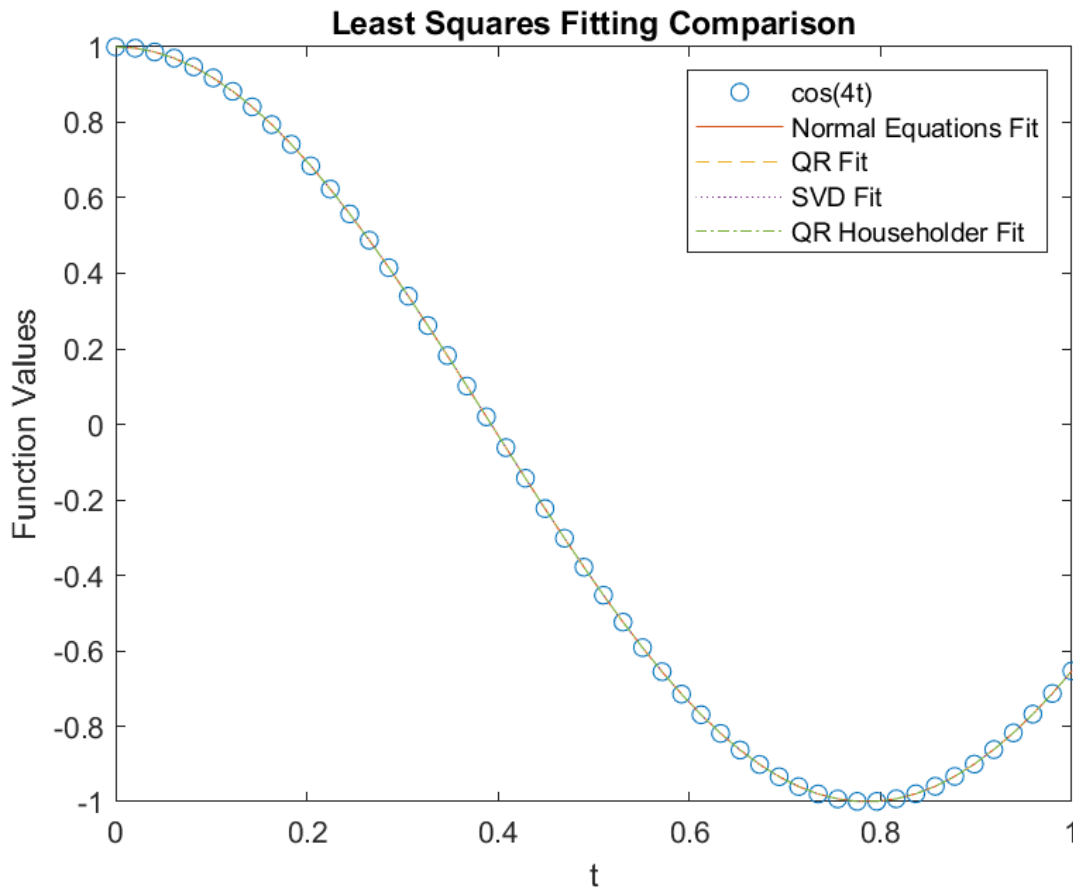
coefficients normal equations =	coefficients qr =
1.000000004464734	1.000000000996605
-0.000001735881690	-0.000000422742859
-7.999927332820611	-7.999981235690551
-0.001164086630733	-0.000318763171777
10.676220872321924	10.669430795484192
-0.045669164461211	-0.013820286242889
-5.553853641610956	-5.647075632082337
-0.250670185502446	-0.075316015937716
1.905583520406809	1.693606953829228
-0.153114645765417	0.006032115717738
-0.306716135166133	-0.374241706299403
0.075668905284410	0.088040576578004

coefficients svd =	coefficients qr house =
1.000000000996607	1.000000000996605
-0.000000422742861	-0.000000422742877
-7.999981235690470	-7.999981235689906
-0.000318763173354	-0.000318763180768
10.669430795497432	10.669430795549223
-0.013820286304435	-0.013820286520703
-5.647075631909320	-5.647075631335162
-0.075316016242881	-0.075316017238264
1.693606954167837	1.693606955292510
0.006032115489239	0.006032114690217
-0.374241706214205	-0.374241705889927
0.088040576564645	0.088040576507287

Βλέποντας τα αποτελέσματα για τους συντελεστές οι διαφορές είναι αμελητέες, καθώς αυτές βρίσκονται μετά το 3ο δεκαδικό ψηφίο.

Παρατηρούμε επίσης, στο παρακάτω γράφημα, ότι και οι τρεις περιπτώσεις προσαρμόζουν το μοντέλο ικανοποιητικά. Δε βλέπουμε καμία διαφορά ανάμεσα στις μεθόδους, οπότε δε μπορούμε να αποφανθούμε αν κάποια μέθοδος είναι πιο κατάλληλη.

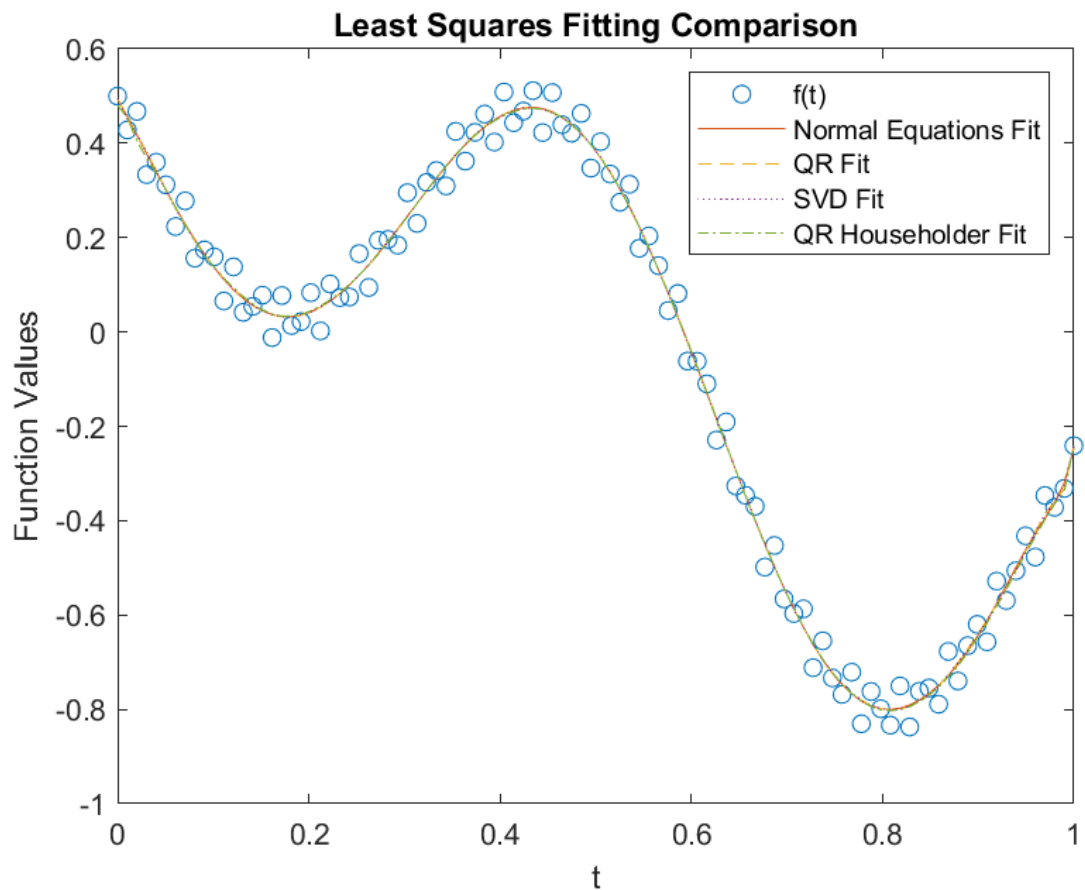
Στο παράδειγμα που εκτελέσαμε, η συνάρτηση που θέλαμε να προσαρμόσουμε ήταν πολύ απλή, για αυτό δε βλέπουμε κάποια διαφορά στις μεθόδους. Στη συνέχεια, θα εκτελέσουμε τον ίδιο αλγόριθμο για μία πιο πολύπλοκη συνάρτηση, ώστε να δούμε αν υπάρχουν διαφορές.



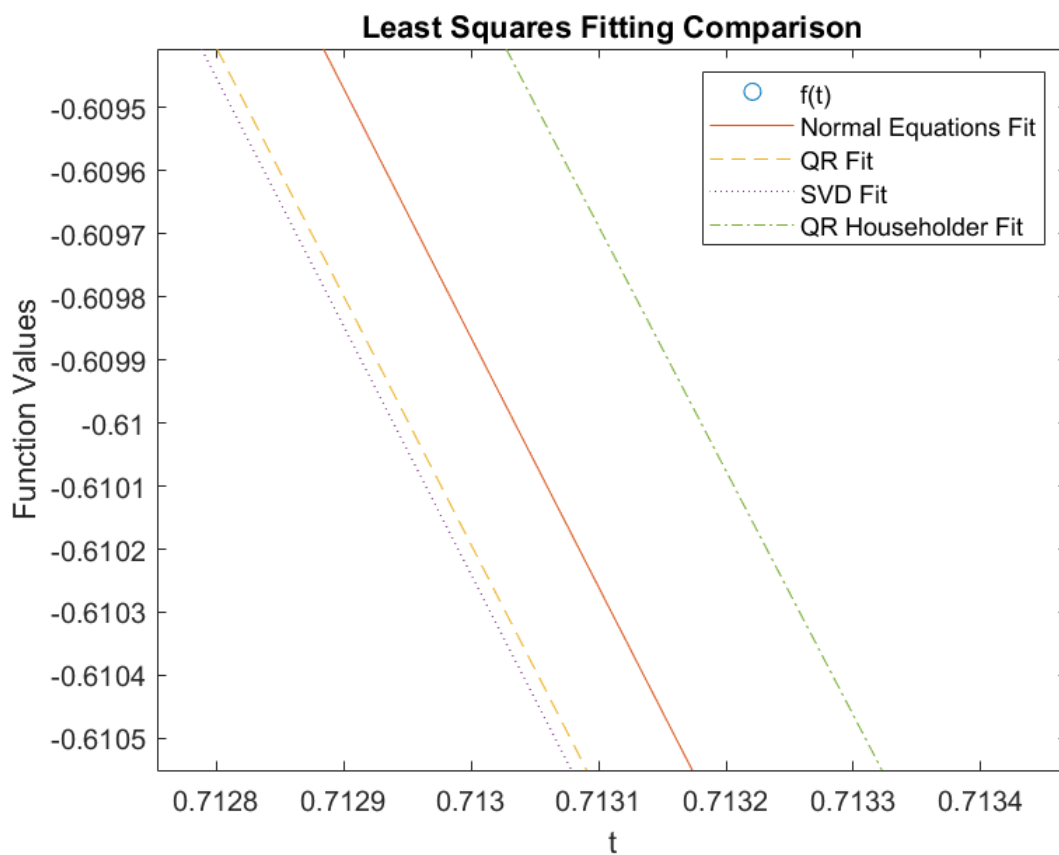
Σχήμα 3.6: Προσαρμογή μοντέλου ελαχίστων τετραγώνων με 4 τρόπους

Εκτελώντας τον ίδιο αλγόριθμο με $m = 100$, $n = 24$ και $b = f(t) = 0.05 * \sin(1000 * t) + 0.5 * \cos(\pi * t) - 0.4 * \sin(10 * t)$ λαμβάνουμε το ακόλουθο γράφημα, όπου βλέπουμε ότι όλες οι μέθοδοι προσαρμόζουν ικανοποιητικά τη συνάρτηση.

Επίσης, εστιάζοντας περισσότερο, ώστε να βρούμε τις διαφορές των μεθόδων, φαίνεται ότι η απόκλιση είναι σχεδόν μηδενική. Άρα και σε αυτή την περίπτωση, ενώ μπορούμε να διακρίνουμε μικρές διαφορές, δε φαίνεται να μπορούμε να απαντήσουμε σίγουρα στο ερώτημα, ποια μέθοδος είναι πιο κατάλληλη για τη συγκεκριμένη συνάρτηση.



Σχήμα 3.7: Προσαρμογή μοντέλου ελαχίστων τετραγώνων με 4 τρόπους



Σχήμα 3.8: Οι διαφορές των τεσσάρων μεθόδων

3.6 Ευστάθεια προβλήματος ελαχίστων τετραγώνων

Ορισμός

Όταν μιλάμε για την ευστάθεια των προβλημάτων ελαχίστων τετραγώνων, εννοούμε την ευαισθησία της λύσης σε διαταραχές των αρχικών δεδομένων ή των παραμέτρων του συστήματος. Μια σταθερή λύση είναι αυτή που παραμένει σχετικά αμετάβλητη, ακόμη και όταν συμβαίνουν μικρές μεταβολές, ενώ μια ασταθής λύση μπορεί να παρουσιάζει σημαντικές μεταβολές, ως αποτέλεσμα μικρών διαταραχών.

Τι ονομάζουμε αλγόριθμο [Lecture 14, [2]]

Αρχικά θα ορίσουμε το *μαθηματικό πρόβλημα* ως μία συνάρτηση $f : X \rightarrow Y$ από ένα διανυσματικό χώρο δεδομένων X , σε ένα διανυσματικό χώρο λύσεων Y .

Ένας *αλγόριθμος*, μπορεί να θεωρηθεί ως μία συνάρτηση $\tilde{f} : X \rightarrow Y$ ανάμεσα στους δύο ίδιους χώρους.

Ακρίβεια [Lecture 14, [2]]

Εκτός από προφανείς περιπτώσεις, η \tilde{f} δε μπορεί να είναι συνεχής. Παρόλο αυτά, ένας καλός αλγόριθμος θα πρέπει να προσεγγίζει το σχετικό πρόβλημα f . Για να το δούμε αυτό αριθμητικά, θεωρούμε το *απόλυτο σφάλμα* ενός υπολογισμού, $\|\tilde{f}(x) - f(x)\|$, ή το *σχετικό σφάλμα*,

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|}. \quad (3.17)$$

Αν η \tilde{f} είναι ένας καλός αλγόριθμος, θα περιμέναμε ότι το σχετικό σφάλμα θα είναι μικρό, τάξης $\epsilon_{machine}$. Θεωρούμε ότι ένας αλγόριθμος \tilde{f} ενός προβλήματος f είναι *ακριβής* αν για κάθε $x \in X$,

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|} = \mathcal{O}(\epsilon_{machine}). \quad (3.18)$$

Γενικά, το σύμβολο $\mathcal{O}(\epsilon_{machine})$ σημαίνει “στη τάξη του $\epsilon_{machine}$ ”.

Ευστάθεια [Lecture 14, [2]]

Αν το πρόβλημα f είναι κακώς ορισμένο, ο στόχος για την ακρίβεια όπως ορίζεται από την (3.18) είναι αδικαιολόγητα φιλόδοξος. Η στρογγυλοποίηση των αρχικών δεδομένων είναι αναπόφευκτη σε έναν ψηφιακό υπολογιστή και ακόμη και αν όλοι οι επόμενοι υπολογισμοί μπορούσαν να εκτελεστούν τέλεια, αυτή η στρογγυλοποίηση από μόνη της μπορεί να οδηγήσει σε σημαντική αλλαγή στο

αποτέλεσμα. Αντί να στοχεύουμε στην ακρίβεια σε όλες τις περιπτώσεις, το μέγιστο που είναι σκόπιμο να επιδιώκουμε γενικά, είναι η ευστάθεια. Λέμε ότι ένας αλγόριθμος \tilde{f} για ένα πρόβλημα f είναι ευσταθής, αν για κάθε $x \in X$,

$$\frac{\|\tilde{f}(x) - f(\tilde{x})\|}{\|f(\tilde{x})\|} = \mathcal{O}(\epsilon_{machine}) \quad (3.19)$$

για κάποια \tilde{x} με

$$\frac{\|\tilde{x} - x\|}{\|x\|} = \mathcal{O}(\epsilon_{machine}). \quad (3.20)$$

Δηλαδή, ένας σταθερός αλγόριθμος δίνει σχεδόν τη σωστή απάντηση σε σχεδόν τη σωστή ερώτηση.

Κεφάλαιο 4

Μερικές Εφαρμογές της Μεθόδου Ελαχίστων Τετραγώνων

4.1 Εφαρμογές στην ανάλυση παλινδρόμησης

Η ανάλυση παλινδρόμησης είναι ένα σημαντικό στατιστικό εργαλείο, το οποίο χρησιμοποιείται για να καταλάβουμε τη σχέση μεταξύ εξαρτημένων και ανεξάρτητων μεταβλητών. Η μέθοδος ελαχίστων τετραγώνων είναι μία από τις πρωταρχικές μεθόδους που χρησιμοποιούνται στην ανάλυση παλινδρόμησης, ώστε να εκτιμήσουμε τις παραμέτρους του μοντέλου.

Για παράδειγμα, έχουμε ένα διάγραμμα διασποράς σημείων δεδομένων, τα οποία αναπαριστούν τις τιμές από δύο μεταβλητές. Ο στόχος μας είναι να σχηματίσουμε μία γραμμή ή καμπύλη, η οποία ελαχιστοποιεί τη συνολική απόκλιση μεταξύ των πραγματικών σημείων δεδομένων και των τιμών που προβλέπει η γραμμή/καμπύλη. Αυτή η διαισθητική ιδέα αποτελεί τη βάση της μεθόδου των ελαχίστων τετραγώνων.

Μαθηματική Φόρμουλα

Στη μέθοδο ελαχίστων τετραγώνων, οι παράμετροι του μοντέλου παλινδρόμησης εκτιμώνται αν ελαχιστοποιήσουμε το υπόλοιπο του αθροίσματος τετραγώνων (RSS), το οποίο ορίζεται ως:

$$RSS = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (4.1)$$

όπου Y_i είναι η παρατηρούμενη τιμή της εξαρτημένης μεταβλητής και \hat{Y}_i είναι η προβλεπόμενη τιμή της εξαρτημένης μεταβλητής που προκύπτει από το μοντέλο παλινδρόμησης.

Η λύση των ελαχίστων τετραγώνων, περιλαμβάνει τον μηδενισμό των παραγώγων των RSS ως προς τις παραμέτρους και την επίλυση του συστήματος εξισώσεων. Η διαδικασία αυτή οδηγεί στην εκτίμηση των συντελεστών που ελαχιστοποιούν το άθροισμα των τετραγωνικών υπολοίπων.

4.2 Μη γραμμικά ελάχιστα τετράγωνα

Μέχρι στιγμής είδαμε τη διαδικασία εξαγωγής πληροφοριών από τα δεδομένα, σε περιπτώσεις όπου η σχέση μεταξύ του μοντέλου και των δεδομένων είναι γραμμική. Ωστόσο, τα φαινόμενα του πραγματικού κόσμου συχνά παρουσιάζουν πολύπλοκες σχέσεις που δεν μπορούν να αποτυπωθούν από γραμμικά μοντέλα. Αυτό το κεφάλαιο εμβαθύνει στην μέθοδο των μη γραμμικών ελαχίστων τετραγώνων (NLS) για την προσαρμογή τέτοιων μη γραμμικών μοντέλων σε δεδομένα.

Το πρόβλημα των μη γραμμικών ελαχίστων τετραγώνων επιδιώκει να βρει ένα σύνολο παραμέτρων, που συμβολίζεται με το διάνυσμα x , το οποίο ελαχιστοποιεί το άθροισμα των τετραγωνικών υπολοίπων μεταξύ μιας συνάρτησης, $f(x, t)$, και ενός συνόλου m σημείων δεδομένων, (t_i, y_i) , όπου $i = 1, 2, \dots, m$. Αυτά τα υπόλοιπα αντιπροσωπεύουν τη διαφορά μεταξύ των προβλεπόμενων τιμών του μοντέλου και των πραγματικών παρατηρήσεων. Μαθηματικά, το πρόβλημα NLS μπορεί να εκφραστεί ως εξής:

$$\min_x \sum_{i=1}^n (y_i - f(x, t_i))^2 \quad (4.2)$$

Εδώ, το άθροισμα επαναλαμβάνεται σε όλα τα m σημεία δεδομένων. Η αντικειμενική συνάρτηση, την οποία θέλουμε να ελαχιστοποιήσουμε, είναι το άθροισμα των τετραγωνικών υπολοίπων.

Χρήσιμοι Ορισμοί

Υπόλοιπα: Όπως αναφέρθηκε προηγουμένως, τα υπόλοιπα αντιπροσωπεύουν τη διαφορά μεταξύ των προβλεπόμενων τιμών του μοντέλου και των παρατηρούμενων σημείων δεδομένων. Η διαδικασία ελαχιστοποίησης του αθροίσματος των τετραγωνικών υπολοίπων, εξασφαλίζει ότι το μοντέλο ταιριάζει στενά στη συνολική τάση των δεδομένων.

Εκτίμηση παραμέτρων: Ο στόχος της NLS, είναι να βρεθούν οι βέλτιστες τιμές για τις παραμέτρους στη συνάρτηση του μοντέλου, x , που ελαχιστοποιούν την αντικειμενική συνάρτηση. Αυτές οι εκτιμώμενες παράμετροι θεωρούνται η «καλύτερη προσαρμογή» για το δεδομένο μοντέλο και το σύνολο δεδομένων.

Ιακωβιανός πίνακας: Ο Ιακωβιανός πίνακας, $J(x)$, παίζει καθοριστικό ρόλο στην επίλυση του

προβλήματος NLS. Περιέχει τις μερικές παραγώγους της συνάρτησης του μοντέλου ως προς κάθε παράμετρο στο x . Η Ιακωβιανή, μας βοηθά να κατανοήσουμε πώς οι αλλαγές στις παραμέτρους επηρεάζουν το αποτέλεσμα του μοντέλου.

Τεχνικές αλγορίθμων για προβλήματα NLS

Για την επίλυση του προβλήματος NLS χρησιμοποιούνται διάφοροι επαναληπτικοί αλγόριθμοι. Αυτοί οι αλγόριθμοι ξεκινούν με μια αρχική εικασία για τις παραμέτρους και τις ενημερώνουν επαναληπτικά, με βάση την αντικειμενική συνάρτηση και τις παραγώγους της. Οι συνήθεις μέθοδοι περιλαμβάνουν:

- **Μέθοδος Gauss-Newton:** Ένας επαναληπτικός αλγόριθμος βελτιστοποίησης, που γραμμικοποιεί το μη γραμμικό μοντέλο, χρησιμοποιώντας προσέγγιση σειρών Taylor πρώτης τάξης και ενημερώνει τις παραμέτρους επαναληπτικά.
- **Μέθοδος Levenberg-Marquardt:** Μια τροποποίηση της μεθόδου Gauss-Newton, που εισάγει μια παράμετρο απόσβεσης για τη βελτίωση των ιδιοτήτων σύγκλισης και της σταθερότητας.
- **Gradient Descent:** Ένας αλγόριθμος βελτιστοποίησης, που προσαρμόζει επαναληπτικά τις παραμέτρους προς την κατεύθυνση της αρνητικής κλίσης της αντικειμενικής συνάρτησης.
- **Μέθοδος Quasi-Newton:** Αλγόριθμοι επαναληπτικής βελτιστοποίησης, που προσεγγίζουν τον πίνακα Hessian της αντικειμενικής συνάρτησης, χρησιμοποιώντας πληροφορίες από προηγούμενες επαναλήψεις.

Κάθε μέθοδος έχει τα δικά της πλεονεκτήματα και περιορισμούς όσον αφορά την ταχύτητα σύγκλισης, την σταθερότητα και το υπολογιστικό κόστος. Η επιλογή της μεθόδου εξαρτάται από τα συγκεκριμένα χαρακτηριστικά της συνάρτησης του μοντέλου και του συνόλου δεδομένων.

4.3 Πρακτικά παραδείγματα

Με βάση το παράδειγμα της `censusgui` [Κεφάλαιο 5.3, [7]], θα υλοποιήσουμε ένα παράδειγμα με το `matlab`, για να προβλέψουμε τον πληθυσμό της Ελλάδας και της Ευρώπης το 2030. Τα δεδομένα μας τα πήραμε από το DataBank του WorldBank [8]. Μπορούμε να δούμε παρακάτω τα δεδομένα για την Ελλάδα και την Ευρώπη αντίστοιχα, για τα χρόνια από το 2000 έως το 2022.

Ελλάδα		Ευρώπη	
Year	Population	Year	Population
2000	10805808	2000	429342455
2001	10862132	2001	429910140
2002	10902022	2002	430898141
2003	10928070	2003	432434810
2004	10955141	2004	434059923
2005	10987314	2005	435600393
2006	11020362	2006	437014860
2007	11048473	2007	438484072
2008	11077841	2008	439892213
2009	11107017	2009	440934530
2010	11121341	2010	441552554
2011	11104899	2011	440769682
2012	11045011	2012	441419873
2013	10965211	2013	442496175
2014	10892413	2014	443601373
2015	10820883	2015	444570054
2016	10775971	2016	445515422
2017	10754679	2017	446215182
2018	10732882	2018	447001100
2019	10721582	2019	447367191
2020	10698599	2020	447692315
2021	10569207	2021	447178112
2022	10426919	2022	447370510

Θα κάνουμε την πρόβλεψη για τον πληθυσμό το 2030, χρησιμοποιώντας την QR και την SVD, βλέποντας επίσης τα αντίστοιχα γραφήματα. Επιπλέον, θα υπολογίσουμε την ακρίβεια των μεθόδων με δύο τρόπους, υπολογίζοντας το RMSE (Τετραγωνική Ρίζα του Μέσου Τετραγωνικού Σφάλματος) και το MAE (Μέσο Απόλυτο Σφάλμα).

Τα έτη, θα τα κανονικοποιήσουμε αφαιρώντας το 2000, για καλύτερη αριθμητική σταθερότητα. Επίσης, θα χρησιμοποιήσουμε ένα τετραγωνικό πολυωνυμικό μοντέλο, για να προσαρμόσουμε στα δεδομένα. Οι προσαρμοσμένες τιμές για τα έτη, υπολογίζονται χρησιμοποιώντας τους συντελεστές που προκύπτουν από την QR και την SVD. Στο διάγραμμα φαίνονται τα πραγματικά δεδομένα του

πληθυσμού, τα προσαρμοσμένα δεδομένα με τη χρήση τόσο της QR όσο και της SVD, καθώς και ο προβλεπόμενος πληθυσμός για το έτος 2030.

```

1  % Given data for greek population
2  years = (2000:2022)';
3  population = [10805808; 10862132; 10902022; 10928070; 10955141;
4               10987314; 11020362; 11048473; 11077841; 11107017; 11121341;
5               11104899; 11045011; 10965211; 10892413; 10820883; 10775971;
6               10754679; 10732882; 10721582; 10698599; 10569207; 10426919];
7
8  % Normalize years for numerical stability
9  years_normalized = years - 2000;
10
11 % Design matrix for a second degree polynomial (quadratic fit)
12 X = [ones(size(years_normalized)), years_normalized,
13      years_normalized.^2];
14
15 % QR decomposition
16 [Q, R] = qr(X, 0);
17 coefficients_qr = R \ (Q' * population);
18
19 % Predict the population for the year 2030 using QR
20 year_2030 = 2030 - 2000; % Normalized year
21 population_2030_qr = [1, year_2030, year_2030^2] * coefficients_qr
22 ;
23 fprintf('Predicted population for the year 2030 using QR: %f\n',
24         population_2030_qr);
25
26 % Singular Value Decomposition
27 [U, S, V] = svd(X, 'econ');
28 coefficients_svd = V * (S \ (U' * population));
29
30 % Predict the population for the year 2030 using SVD
31 population_2030_svd = [1, year_2030, year_2030^2] *

```



```
    coefficients_svd;
26 fprintf('Predicted population for the year 2030 using SVD: %f\n',
    population_2030_svd);
27
28 % Calculate predicted values for the existing data points
29 predicted_population_qr = X * coefficients_qr;
30 predicted_population_svd = X * coefficients_svd;
31
32 % Calculate errors
33 errors_qr = population - predicted_population_qr;
34 errors_svd = population - predicted_population_svd;
35
36 % Calculate RMSE
37 rmse_qr = sqrt(mean(errors_qr.^2));
38 rmse_svd = sqrt(mean(errors_svd.^2));
39
40 % Calculate MAE
41 mae_qr = mean(abs(errors_qr));
42 mae_svd = mean(abs(errors_svd));
43
44 % Display accuracy metrics
45 fprintf('RMSE for QR: %f\n', rmse_qr);
46 fprintf('MAE for QR: %f\n', mae_qr);
47 fprintf('RMSE for SVD: %f\n', rmse_svd);
48 fprintf('MAE for SVD: %f\n', mae_svd);
49
50 % Plotting
51 figure;
52 hold on;
53 plot(years, population, 'ko-', 'LineWidth', 1.5, 'DisplayName', '
    Actual Population');
54 plot(years, predicted_population_qr, 'b--', 'LineWidth', 1.5, '
    DisplayName', 'Fitted Population (QR)');
55 %plot(years, predicted_population_svd, 'r-.', 'LineWidth', 1.5, '
```

```

    DisplayName, 'Fitted Population (SVD)');
56 plot(2030, population_2030_qr, 'bo', 'MarkerSize', 10, '
    DisplayName, 'Predicted 2030 (QR)');
57 %plot(2030, population_2030_svd, 'ro', 'MarkerSize', 10, '
    DisplayName, 'Predicted 2030 (SVD)');
58 xlabel('Year');
59 ylabel('Population');
60 title('Population of Greece (2000-2022) and Prediction for 2030');
61 legend('show');
62 grid on;
63 hold off;

```

```

1 % Given data for europe population
2 years = (2000:2022)';
3 population = [429342455; 429910140; 430898141; 432434810;
    434059923; 435600393; 437014860; 438484072; 439892213;
    440934530; 441552554; 440769682; 441419873; 442496175;
    443601373; 444570054; 445515422; 446215182; 447001100;
    447367191; 447692315; 447178112; 447370510];
4
5 % Normalize years for numerical stability
6 years_normalized = years - 2000;
7
8 % Design matrix for a second degree polynomial (quadratic fit)
9 X = [ones(size(years_normalized)), years_normalized,
    years_normalized.^3];
10
11 % QR decomposition
12 [Q, R] = qr(X, 0);
13 coefficients_qr = R \ (Q' * population);
14
15 % Predict the population for the year 2030 using QR
16 year_2030 = 2030 - 2000; % Normalized year
17 population_2030_qr = [1, year_2030, year_2030^2] * coefficients_qr

```

```
    ;
18 fprintf('Predicted population for the year 2030 using QR: %f\n',
    population_2030_qr);
19
20 % Singular Value Decomposition
21 [U, S, V] = svd(X, 'econ');
22 coefficients_svd = V * (S \ (U' * population));
23
24 % Predict the population for the year 2030 using SVD
25 population_2030_svd = [1, year_2030, year_2030^2] *
    coefficients_svd;
26 fprintf('Predicted population for the year 2030 using SVD: %f\n',
    population_2030_svd);
27
28 % Calculate predicted values for the existing data points
29 predicted_population_qr = X * coefficients_qr;
30 predicted_population_svd = X * coefficients_svd;
31
32 % Calculate errors
33 errors_qr = population - predicted_population_qr;
34 errors_svd = population - predicted_population_svd;
35
36 % Calculate RMSE
37 rmse_qr = sqrt(mean(errors_qr.^2));
38 rmse_svd = sqrt(mean(errors_svd.^2));
39
40 % Calculate MAE
41 mae_qr = mean(abs(errors_qr));
42 mae_svd = mean(abs(errors_svd));
43
44 % Display accuracy metrics
45 fprintf('RMSE for QR: %f\n', rmse_qr);
46 fprintf('MAE for QR: %f\n', mae_qr);
47 fprintf('RMSE for SVD: %f\n', rmse_svd);
```

```

48 fprintf('MAE for SVD: %f\n', mae_svd);
49
50 % Plotting
51 figure;
52 hold on;
53 plot(years, population, 'ko-', 'LineWidth', 1.5, 'DisplayName', '
    Actual Population');
54 plot(years, predicted_population_qr, 'b--', 'LineWidth', 1.5, '
    DisplayName', 'Fitted Population (QR)');
55 plot(years, predicted_population_svd, 'r-.', 'LineWidth', 1.5, '
    DisplayName', 'Fitted Population (SVD)');
56 plot(2030, population_2030_qr, 'bo', 'MarkerSize', 10, '
    DisplayName', 'Predicted 2030 (QR)');
57 plot(2030, population_2030_svd, 'ro', 'MarkerSize', 10, '
    DisplayName', 'Predicted 2030 (SVD)');
58 xlabel('Year');
59 ylabel('Population');
60 title('Population of Europe (2000-2022) and Prediction for 2030');
61 legend('show');
62 grid on;
63 hold off;

```

Παίρνουμε ως αποτέλεσμα τα παρακάτω:

Για την Ελλάδα:

Predicted population for the year 2030 using QR: 9513659.831677

Predicted population for the year 2030 using SVD: 9513659.831677

RMSE for QR: 47797.503974

MAE for QR: 37999.985309

RMSE for SVD: 47797.503974

MAE for SVD: 37999.985309

Για την Ευρώπη:

Predicted population for the year 2030 using QR: 447680243.109486

Predicted population for the year 2030 using SVD: 447680243.109477

RMSE for QR: 627254.174092

MAE for QR: 549710.293727

RMSE for SVD: 627254.174092

MAE for SVD: 549710.293727

Παρατηρούμε ότι και στα δύο παραδείγματα, οι τιμές της πρόβλεψης του πληθυσμού για το 2030, με τις δύο μεθόδους, ταυτίζονται. Επίσης, ταυτίζονται και οι τιμές για την ακρίβεια των μεθόδων. Για να καταλάβουμε αν οι τιμές των RMSE και MAE είναι αποδεκτές, μπορούμε να υπολογίσουμε τα ποσοστιαία σφάλματα και να εκτιμήσουμε αν εμπίπτουν σε ένα λογικό εύρος. Συγκεκριμένα, υπολογίζουμε τον μέσο πληθυσμό στην Ελλάδα και στην Ευρώπη αντίστοιχα και στη συνέχεια το ποσοστό, κάνοντας την πράξη $\frac{RMSE}{AveragePopulation} * 100$ και $\frac{MAE}{AveragePopulation} * 100$.

Υλοποιώντας το παραπάνω στο matlab:

```

1  % Calculate the average population
2  average_population_greece = mean(population_greece);
3  average_population_europe = mean(population_europe);
4
5  % RMSE and MAE values
6  rmse_qr_greece = 47797.503974;
7  mae_qr_greece = 37999.985309;
8  rmse_qr_europe = 627254.174092;
9  mae_qr_europe = 549710.293727;
10
11 % Percentage errors for Greece
12 percentage_rmse_greece = (rmse_qr_greece /
    average_population_greece) * 100;
13 percentage_mae_greece = (mae_qr_greece / average_population_greece
    ) * 100;
14
15 fprintf('Greece - Percentage RMSE: %f%%\n', percentage_rmse_greece
    );
16 fprintf('Greece - Percentage MAE: %f%%\n', percentage_mae_greece);
17

```

```
18 % Percentage errors for Europe
19 percentage_rmse_europe = (rmse_qr_europe /
    average_population_europe) * 100;
20 percentage_mae_europe = (mae_qr_europe / average_population_europe
    ) * 100;
21
22 fprintf('Europe - Percentage RMSE: %f%%\n', percentage_rmse_europe
    );
23 fprintf('Europe - Percentage MAE: %f%%\n', percentage_mae_europe);
```

Average population of Greece: 10883642.47826087

Greece - Percentage RMSE: 0.439168%

Greece - Percentage MAE: 0.349148%

Average population of Europe: 440492220.8695652

Europe - Percentage RMSE: 0.142398%

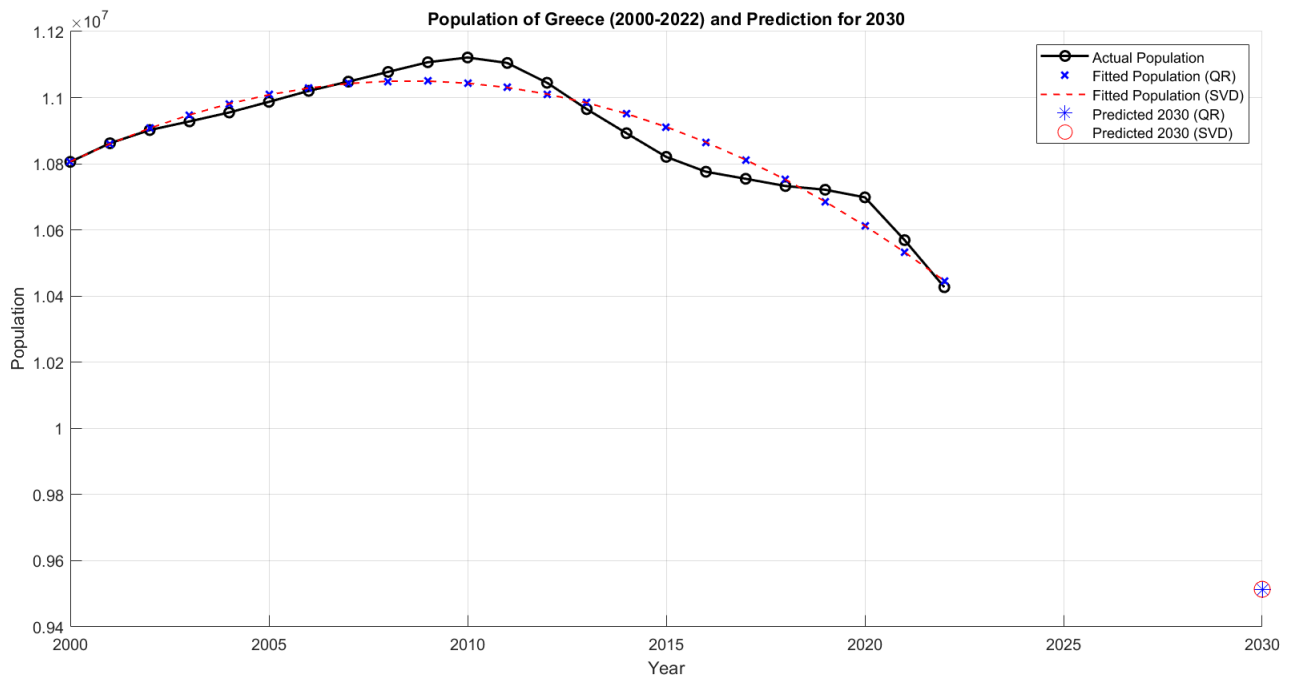
Europe - Percentage MAE: 0.124795%

Οπότε, τα σφάλματα αυτά είναι πολύ μικρά σε σχέση με τον συνολικό πληθυσμό, υποδεικνύοντας καλή προσαρμογή του μοντέλου.

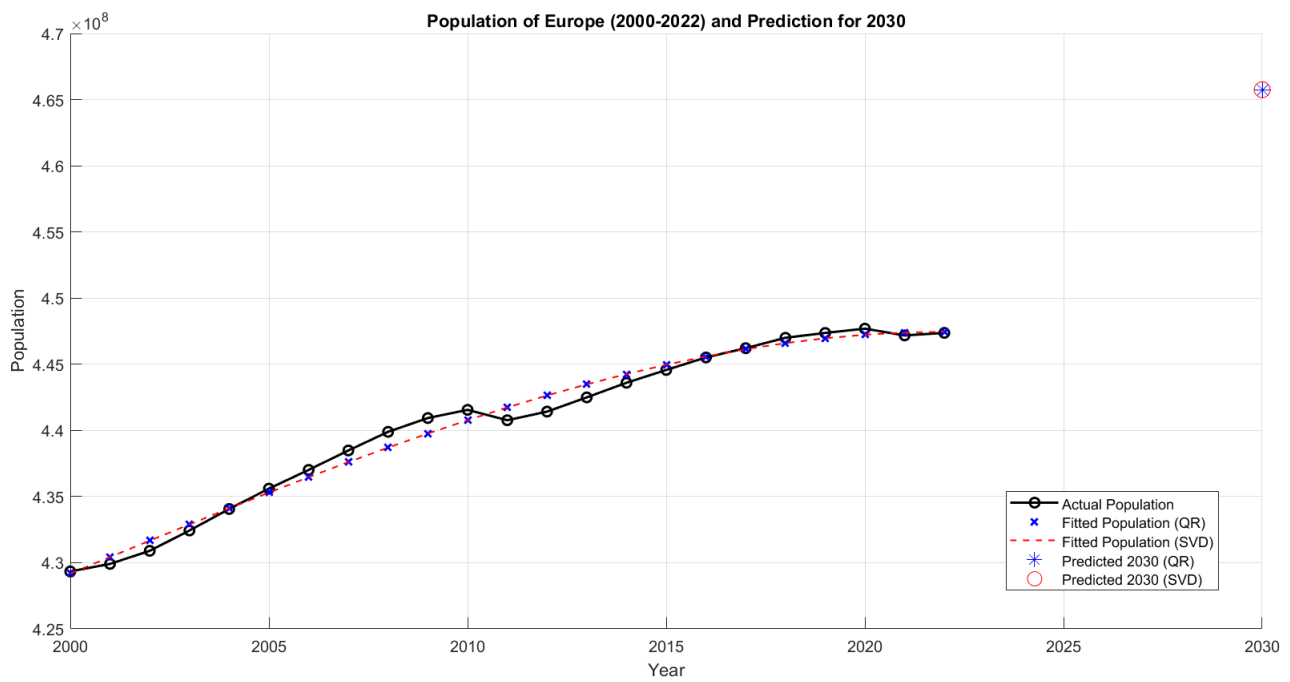
Βλέποντας τώρα τα γραφήματα, τα πραγματικά πληθυσμιακά δεδομένα απεικονίζονται με μαύρους κύκλους και γραμμές. Τα προσαρμοσμένα πληθυσμιακά δεδομένα με τη χρήση της QR, απεικονίζονται με μπλε διακεκομμένες γραμμές και αυτά με χρήση της SVD, απεικονίζονται με κόκκινες διακεκομμένες γραμμές. Ο προβλεπόμενος πληθυσμός για το έτος 2030 με τη χρήση της QR απεικονίζεται με μπλε κύκλο, ενώ με τη χρήση της SVD απεικονίζεται με κόκκινο κύκλο.

Καταλαβαίνουμε ότι η QR ταυτίζεται με την SVD και στους δύο πληθυσμούς, τόσο στην προσαρμογή των δεδομένων όσο και στην προβλεπόμενη τιμή του πληθυσμού.

4.3 Πρακτικά παραδείγματα



Σχήμα 4.1: Διάγραμμα πληθυσμού Ελλάδας



Σχήμα 4.2: Διάγραμμα πληθυσμού Ευρώπης

4.4 Συμπεράσματα

Παρατηρώντας, λοιπόν, την κατασκευή όλων των αλγορίθμων, μπορούμε να καταλήξουμε στα εξής συμπεράσματα:

1. Ο αλγόριθμος των κανονικών εξισώσεων, φαίνεται να είναι ο πιο γρήγορος και απλός στον υπολογισμό.
2. Ο αλγόριθμος των κανονικών εξισώσεων δεν είναι σταθερός, οπότε προτιμάμε τον αλγόριθμο της παραγοντοποίησης QR.
3. Στην περίπτωση που ο πίνακας A δεν είναι πλήρους βαθμού, τότε δεν υπάρχει σταθερότητα και προτιμάμε τον αλγόριθμο της SVD.

Με βάση όμως, την υλοποίηση στο matlab, με τις έτοιμες συναρτήσεις QR και SVD που μας παρέχει, βλέπουμε τους παρακάτω χρόνους για κάθε μέθοδο. (Οι χρόνοι αφορούν τον υπολογισμό των συντελεστών, για το παράδειγμα της συνάρτησης $b = \cos(4 * t)$).

Κανονικές Εξισώσεις - `timeElapsed1 = 0.0365`

Παραγοντοποίηση QR - `timeElapsed2 = 0.0153`

Μέθοδος SVD - `timeElapsed3 = 0.0307`

Householder - `timeElapsed4 = 0.0071`

Και για το παράδειγμα εύρεσης του πληθυσμού της Ελλάδας και της Ευρώπης, μετρώντας το χρόνο που χρειάζεται κάθε μέθοδος για τον υπολογισμό των συντελεστών και τον υπολογισμό της προβλεπόμενης τιμής:

Για την Ελλάδα:

QR - `timeElapsed1 = 0.0526`

SVD - `timeElapsed2 = 0.0226`

Για την Ευρώπη:

QR - `timeElapsed1 = 0.0533`

SVD - `timeElapsed2 = 0.0267`

Γραφικά, παρατηρούμε ότι δεν υπάρχει μεγάλη διαφορά μεταξύ των μεθόδων, οπότε μπορούμε να επιλέξουμε τη μέθοδο που θα έχει τη μικρότερη πολυπλοκότητα, για το εκάστοτε πρόβλημα που μελετάμε.

Συμπερασματικά, η μέθοδος των ελαχίστων τετραγώνων έχει αναδειχθεί σε ένα ισχυρό εργαλείο για την αντιμετώπιση προβλημάτων, στα οποία είναι δύσκολη η εύρεση ακριβών λύσεων. Ελαχιστοποιώντας τα τετραγωνικά σφάλματα μεταξύ των παρατηρούμενων δεδομένων και των προβλεπόμενων τιμών, μπορούμε να βρούμε την «καλύτερη δυνατή» λύση. Η μέθοδος των ελαχίστων τετραγώνων, σε συνδυασμό με αποτελεσματικές τεχνικές όπως η παραγοντοποίηση QR, μας δίνει τη δυνατότητα να εξάγουμε σημαντικές πληροφορίες από δεδομένα του πραγματικού κόσμου, ακόμη και με την παρουσία διαταραχών και αβεβαιοτήτων.

Βιβλιογραφία

- [1] Γ.Δ.Ακριβής - Β.Α.Δουγαλής. *Εισαγωγή στην Αριθμητική Ανάλυση*. Πανεπιστημιακές Εκδόσεις Κρήτης, 2010.
- [2] David Trefethen Lloyd N. and Bau. *Numerical Linear Algebra*. SIAM, 1997. ISBN: 0898713617.
- [3] Gene H. Golub και Charles F. Van Loan. *Matrix computations*. Fourth edition. Johns Hopkins studies in the mathematical sciences. Baltimore: The Johns Hopkins University Press, 2013. ISBN: 9781421407944.
- [4] Peter Bhatia Rajendra and Šemrl. “Orthogonality of matrices and some distance problems”. *Linear Algebra and its Applications* **287**.1–3 (1999), 77–85. DOI: 10.1016/s0024-3795(98)10134-9.
- [5] Greg Fasshauer. *Lecture notes in Numerical Linear Algebra/Computational Mathematics*. Illinois Institute of Technology. FallFall 2006.
- [6] Anil Damle. *Lecture notes in Orthogonal projectors*. Cornell University. 2019.
- [7] Cleve B. Moler. *Numerical Computing with MATLAB*. eng. Revised edition. Society for Industrial και Applied Mathematics, 2008. ISBN: 9780898717952.
- [8] *Data Bank, World Bank*. URL: <https://databank.worldbank.org/reports.aspx?source=2&series=SP.POP.TOTL&country=EUU#>.
- [9] Åke Björck. *Numerical methods for least squares problems*. Philadelphia: SIAM, 1996. ISBN: 9780898713602.
- [10] Richard A. Brualdi και Stephen Mellendorf. “Regions in the Complex Plane Containing the Eigenvalues of a Matrix”. en. *The American Mathematical Monthly* **101**.10 (Δεκ. 1994), 975–985. ISSN: 0002-9890, 1930-0972. DOI: 10.1080/00029890.1994.12004577. URL: <https://www.tandfonline.com/doi/full/10.1080/00029890.1994.12004577>.
- [11] Richard A. Brualdi και Herbert John Ryser. *Combinatorial matrix theory*. Encyclopedia of mathematics and its applications. Cambridge [England]; New York: Cambridge University Press, 1991. ISBN: 9780521322652.

- [12] Philippe G. Ciarlet, Bernadette Miara και Jean-Marie Thomas. *Introduction to numerical linear algebra and optimisation*. engfre. Cambridge texts in applied mathematics. Cambridge [England]; New York: Cambridge University Press, 1989. ISBN: 9780521327886.
- [13] Don Coppersmith και Shmuel Winograd. “Matrix multiplication via arithmetic progressions”. en. *Journal of Symbolic Computation* **9.3** (Μαρ. 1990), 251–280. ISSN: 07477171. DOI: 10.1016/S0747-7171(08)80013-2. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0747717108800132>.
- [14] Biswa Nath Datta. *Numerical linear algebra and applications*. 2nd ed. Philadelphia: Society for Industrial και Applied Mathematics, 2010. ISBN: 9780898716856.
- [15] George E. Forsythe και Cleve B. Moler. *Computer solution of linear algebraic systems*. eng. Prentice-Hall series in automatic computation. Englewood Cliffs, N.J.: Prentice-Hall, 1967. ISBN: 9780131657793.
- [16] J. G. F. Francis. “The QR Transformation A Unitary Analogue to the LR Transformation—Part 1”. en. *The Computer Journal* **4.3** (Μαρ. 1961), 265–271. ISSN: 0010-4620, 1460-2067. DOI: 10.1093/comjnl/4.3.265. URL: <https://academic.oup.com/comjnl/article-lookup/doi/10.1093/comjnl/4.3.265>.
- [17] David S. Watkins. “Understanding the QR Algorithm”. en. *SIAM Review* **24.4** (Οκτ. 1982), 427–440. ISSN: 0036-1445, 1095-7200. DOI: 10.1137/1024100. URL: <http://epubs.siam.org/doi/10.1137/1024100>.
- [18] A. Van Der Sluis. “Stability of solutions of linear algebraic systems”. en. *Numerische Mathematik* **14.3** (Φεβ. 1970), 246–251. ISSN: 0029-599X, 0945-3245. DOI: 10.1007/BF02163333. URL: <http://link.springer.com/10.1007/BF02163333>.
- [19] Charles L. Lawson και Richard J. Hanson. *Solving least squares problems*. Classics in applied mathematics. Philadelphia: SIAM, 1995. ISBN: 9780898713565.
- [20] C. Lanczos. “Solution of systems of linear equations by minimized iterations”. en. *Journal of Research of the National Bureau of Standards* **49.1** (Ιούλ. 1952), 33. ISSN: 0091-0635. DOI: 10.6028/jres.049.006. URL: https://nvlpubs.nist.gov/nistpubs/jres/049/jresv49n1p33_A1b.pdf.