



Linear Algebra, part 3

Anna-Karin Tornberg

Mathematical Models, Analysis and Simulation
Fall semester, 2010

Going back to least squares

(Sections 1.7 and 2.3 from Strang).

We know from before:

The vector $\mathbf{x}$ that minimizes $\|\mathbf{Ax} - \mathbf{b}\|^2$ is the solution to the *normal equations*

$$\mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b}.$$

This vector $\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$ is the *least squares solution* to $\mathbf{Ax} = \mathbf{b}$.

The error $\mathbf{e} = \mathbf{b} - \mathbf{Ax} = \mathbf{b} - \mathbf{p}$.

The projection $\mathbf{p}$ is the closest point to $\mathbf{b}$ in the column space of $\mathbf{A}$.

The error is orthogonal to the column space of $\mathbf{A}$, i.e. orthogonal to each column of $\mathbf{A}$.

$$\text{Let } \mathbf{A} = \begin{bmatrix} | & & | \\ \mathbf{a}_1 & \cdots & \mathbf{a}_n \\ | & & | \end{bmatrix}, \quad \text{then } \mathbf{A}^T \mathbf{e} = \begin{bmatrix} \mathbf{a}_1^T \mathbf{e} \\ \vdots \\ \mathbf{a}_n^T \mathbf{e} \end{bmatrix} = \mathbf{0}.$$

(each $\mathbf{a}_j$ and $\mathbf{e}$ is $m \times 1$).

Least squares solutions - how to compute them?

Form the matrix $\mathbf{K} = \mathbf{A}^T \mathbf{A}$.

Solve $\mathbf{K}\mathbf{x} = \mathbf{b}$ using LU decomposition.

Another idea: Obtain an orthogonal basis for **A** by e.g. the Gram-Schmidt orthogonalization.

First: What if the columnvectors of A were orthogonal, what good would it do?

[WORKSHEET]

Gram-Schmidt orthogonalization

Assume all columns of $\mathbf{A}$ are linearly independent.

An orthogonal basis for **A** can be found by e.g. Gram-Schmidt orthogonalization.

Here is the algorithm:

$$r_{11} = \|\mathbf{a}_1\|, \mathbf{q}_1 = \mathbf{a}_1 / r_{11}$$

for $k=2,3,\dots,n$

for $j=2,3,\dots,k-1$

$$r_{jk} = \mathbf{a}_k^T \mathbf{q}_j = \langle \mathbf{a}_k^T, \mathbf{q}_j \rangle \text{ inner product}$$

end

$$\tilde{\mathbf{q}}_k = \mathbf{a}_k - \sum_{j=1}^{k-1} r_{jk} \mathbf{q}_j$$

$$r_{kk} = \|\mathbf{q}_k\|$$

$$\mathbf{q}_k = \frac{\mathbf{q}_k}{r_{kk}}$$

end

$\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n$ are now orthogonal.

How does this algorithm work? [NOTES]

QR factorization

Given an $m \times n$ matrix $\mathbf{A}$, $m \geq n$.

- Assume that all columns $\mathbf{a}_i$ $i = 1, \dots, n$ are linearly independent. (Matrix is of rank n).
- Then the orthogonalization procedure produces a factorization $\mathbf{A} = \mathbf{QR}$, with

$$\mathbf{Q} = \begin{bmatrix} | & | & \dots & | \\ \mathbf{q}_1 & \dots & \mathbf{q}_n & \\ | & | & \dots & | \end{bmatrix}, \quad \text{and } \mathbf{R} \text{ upper triangular}$$

and r_{jk} and r_{kk} as given by the Gram-Schmidt algorithm.

- $\mathbf{Q}$ is $m \times n$, $\mathbf{R}$ is $n \times n$.
- $\mathbf{R}$ is non-singular. (Upperdiagonal and diagonal entries non-zero).

QUESTION: What do the normal equations become?

FULL QR factorization

Given an $m \times n$ matrix $\mathbf{A}$, $m > n$.

- Matlab's QR command produces a FULL QR factorization.
- It appends an additional $m - n$ orthogonal columns to $\mathbf{Q}$ such that it becomes an $m \times m$ unitary matrix.
- Rows of zeros are appended to $\mathbf{R}$ so that it becomes a $m \times n$ matrix, still upper triangular.
- The command $\text{QR}(\mathbf{A}, 0)$ yields what they call the "economy size" decomposition, without this extension.

The Householder algorithm

- The Householder algorithm is an alternative to Gram-Schmidt for computing the QR decomposition.
- Process of "orthogonal triangularization", making a matrix triangular by a sequence of unitary matrix operations.
- Read in Strang!

Singular Value Decomposition

- Eigenvalue decomposition ($\mathbf{A}$ is $n \times n$): $\mathbf{A} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1}$.
Not all square matrices are diagonalizable.
Need a full set of linearly independent eigenvectors.
- Singular Value Decomposition: $\mathbf{A}_{m \times n} = \mathbf{U}_{m \times n}\mathbf{\Sigma}_{n \times n}\mathbf{V}_{n \times n}^T$
where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices and $\mathbf{\Sigma}$ is a diagonal matrix with the "singular values" (σ_i) of $\mathbf{A}$ on the diagonal.
- Singular values of $\mathbf{A}$: square root of eigenvalues of $\mathbf{A}^T\mathbf{A}$.
- Assume that the columns of $\mathbf{A}$ are linearly independent, then all singular values are positive.
($\mathbf{A}^T\mathbf{A}$ is SPD, all eigenvalues strictly positive).

QUESTION: How to solve the least squares problem, ones we know the SVD?

Singular Value Decomposition, contd.

Definition: A non-negative real number σ is a singular value for $\mathbf{A}$ ($m \times n$) if and only if there exist unit length vectors $\mathbf{u} \in \mathbb{R}^m$ and $\mathbf{v} \in \mathbb{R}^n$ such that

$$\mathbf{A}\mathbf{v} = \sigma\mathbf{u} \quad \text{and} \quad \mathbf{A}^T\mathbf{u} = \sigma\mathbf{v}.$$

The vectors $\mathbf{u}$ and $\mathbf{v}$ are called the left singular and singular vectors for $\mathbf{A}$, respectively.

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T.$$

- The column vectors of $\mathbf{V}$ are the orthonormal eigenvectors of $\mathbf{A}^T\mathbf{A}$.
- The column vectors of $\mathbf{U}$ are the orthonormal eigenvectors of $\mathbf{A}\mathbf{A}^T$.
- $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ forms a basis for the column space of $\mathbf{A}$.
- $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ forms a basis for the row space of $\mathbf{A}$.

SVD, columns of $\mathbf{A}$ linearly dependent

- $\mathbf{A}$ is $m \times n$, $m \geq n$.
Assume that the $\text{rank}(\mathbf{A}) = r < n$.
- $\mathbf{A}$ and $\mathbf{A}^T\mathbf{A}$ have the same null space, the same row space and the same rank.
- $\mathbf{A}^T\mathbf{A}$ no longer positive definite, but at least definite:
 $\mathbf{x}^T\mathbf{A}^T\mathbf{A}\mathbf{x} \geq 0 \quad \forall \mathbf{x}$.
- All eigenvalues of $\mathbf{A}^T\mathbf{A}$ are non negative, $\lambda_i \geq 0$.
- $\mathbf{A}^T\mathbf{A}$ symmetric, i.e. diagonalizable. Rank of $\mathbf{A}^T\mathbf{A}$ and hence of $\mathbf{A}$, number of strictly positive eigenvalues.
- $\sigma_i = \sqrt{\lambda_i}$. The rank r of $\mathbf{A}$ is the number of strictly positive singular values of $\mathbf{A}$.

Reduced and full SVD

- Number the $\mathbf{u}_i$, $\mathbf{v}_i$ and σ_i such that $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$.
- The orthonormal eigenvectors of $\mathbf{A}^T \mathbf{A}$ are in $\mathbf{V}$.
- $\mathbf{AV} = \mathbf{U}\Sigma$. Define $\mathbf{u}_i = \mathbf{A}\mathbf{v}_i / \sigma_i$, $i = 1, \dots, r$.
- **Reduced SVD:** $\mathbf{A} = \mathbf{U}_{m \times r} \Sigma_{r \times r} \mathbf{V}_{r \times n}^T$.
- To complete the $\mathbf{v}$'s add any orthonormal basis $\mathbf{v}_{r+1}, \dots, \mathbf{v}_n$ for the nullspace of $\mathbf{A}$.
- To complete the $\mathbf{u}$'s add any orthonormal basis $\mathbf{u}_{r+1}, \dots, \mathbf{u}_m$ for the nullspace of $\mathbf{A}^T$.
- **Full SVD:** $\mathbf{A} = \mathbf{U}_{m \times m} \Sigma_{m \times n} \mathbf{V}_{n \times n}^T$.

Pseudoinverse: $\mathbf{A}^+ = \mathbf{V}\Sigma^+ \mathbf{U}^T$, where Σ^+ has $1/\sigma_1, \dots, 1/\sigma_r$ on its diagonal, and zeros elsewhere.

Solution to least squares problem: $\mathbf{x} = \mathbf{A}^+ \mathbf{b} = \mathbf{V}\Sigma^+ \mathbf{U}^T \mathbf{b}$.



QR for rank deficient matrices

- The QR algorithm can also be adapted to the case when columns of $\mathbf{A}$ are linearly dependent.
- Gram-Schmidt algorithm combined with column reordering to choose the "largest" remaining column at each step.
- Permutation matrix $\mathbf{P}$.
 $\mathbf{AP} = \mathbf{QR}$.
- **R:** Last $n - r$ rows zero, where r is rank of $\mathbf{A}$.
(Or for full QR: $m - r$ last rows zero).



Condition number

- Condition number of matrix measures the sensitivity of the linear system $\mathbf{Ax} = \mathbf{b}$.
- Perturb rhs by $\Delta\mathbf{b}$. What change $\Delta\mathbf{x}$ does it give? What relative change $\|\Delta\mathbf{x}\|/\|\mathbf{x}\|$?
- Condition number of $\mathbf{A}$ defined as $c(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^{-1}\|$.
- Computer algorithms for solving the linear system loses about $\log c$ decimals to round off error. If large condition number, render inaccurate solution.
- $\mathbf{A}^T \mathbf{A}$ might have a very large condition number. Working with LU factorization (or Cholesky) on the normal equations is then not a good algorithm.
- The QR-algorithm is better conditioned. Standard method for least squares problems.
- If matrix close to rank-deficient, the algorithm based on the SVD is however more stable.



Operation count

- So far, have focused on how algorithms computes the desired solution.
 - Speed is important! (Think huge systems with thousands to millions of unknowns).
 - Operation count: Counting how many elementary operations must be done given parameters of matrix size.
 - Asymptotic cost for:
 - i) Matrix vector multiply: $2nm$.
 - ii) LU decomposition: $2/3n^3$.
 - iii) Backsolve: $2n^2$
 - iii) Total solution of linear $n \times n$ system: $2/3n^3 + 2n^2$.
 - Asymptotic cost for least squares solution for $m \times n$ matrix $\mathbf{A}$:
 - i) Normal equations (Cholesky): $mn^2 + n^3/3$
 - ii) QR algorithm with Gram-Schmidt: $2mn^2$.
 - iii) SVD algorithm: $2mn^2 + 11n^3$.
- Depends on details: Changing Gram-Schmidt to Householder, different SVD algorithm etc.



Useful property of the SVD

- We can write (assuming $\sigma_{r+1} = \dots = \sigma_n = 0$):
$$\mathbf{A} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_1 \mathbf{u}_2 \mathbf{v}_2^T + \dots + \sigma_r \mathbf{u}_r \mathbf{v}_r^T.$$
- Can remove the terms with small singular values and still have a good approximation of $\mathbf{A}$. "Low rank approximation".
- If a low rank approximation of a matrix is found: takes less space to store the matrix, less time to compute a matrix vector multiply etc.
- Randomized ways to find low rank approximations of matrices without actually computing the SVD, active research area (cheaper than doing the SVD).
- Used also for image compression. See HW2.