Least Squares

• More specifically: **Discrete Linear Least Squares**.
• Recall the linear system $Ax = b$.
• If $A$ has more rows than columns we have more equations than variables.
• We have an **overdetermined system**:

\[
Ax = b \quad A \quad m \times n, \quad m > n.
\]  \hspace{1cm} (1)

• We have already seen the basic idea of dealing with (1). Rewrite it as

\[
\begin{align*}
r &= Ax - b = 0
\end{align*}
\]

and instead of making the residual $r$ zero make it as small as possible. For least squares we use the 2-norm:

\[
\|Ax - b\|_2 = \min!
\]  \hspace{1cm} (2)

• This is a big subject, quite a few books have been written about it, including


Scattered Data Fitting

- A major application of Least Squares is scattered data fitting. Suppose we are given scattered data

\[(t_i, b_i), \quad i = 1, \ldots, m\]
• We want to approximate, fit, or represent, the data by a linear combination of \( n < m \) basis functions \( \phi_i \):

\[
p(t) = \sum_{i=1}^{n} x_i \phi_i(t)
\]

where the \( x_i \) are coefficients to be determined.

The choice of the basis functions depends on the problem. They don’t have to be polynomials!

• Examples of possible basis functions include:
  – polynomials (for good reasons, or if we don’t know what else to do)
  – periodic functions (for periodic problems)
  – exponentials (for problems involving exponential growth or decay)
  – rational functions (for example, to model an explosion)

But we do assume that we form a linear combination.

• For example, approximating with

\[
p(t) = x_1 e^{x_2 t} + x_3 \sin x_4 t
\]

would be a nonlinear discrete least squares problem.

“linear least squares” does not mean that we approximate with a linear function. It means that we approximate with a linear combination of basis functions. A consequence of doing so is that we end up with having to solve a linear system.

**Computing the Coefficients**

• We want to minimize

\[
F(x_1, x_2, \ldots, x_n) = \sum_{k=1}^{m} \left( b_k - \sum_{j=1}^{n} x_j \phi_j(t_k) \right)^2 = \min
\]

\[ (3) \]
• This problem can be rewritten in the form (2). Let

\[ A = [\phi_j(t_i)]_{i=1,...,m, j=1,...,n}, \quad b = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}. \quad m > n \]

Then

\[ F(x) = \|Ax - b\|_2^2 = \min \]

(4)

• Note that minimizing the norm is equivalent to minimizing the square of the norm.

• The matrix \( A \) in (4) is sometimes called the design matrix.

• We will solve this discrete linear least squares problem in three different ways:
  – Calculus
  – Geometry
  – Numerical Linear Algebra

**Calculus Solution**

• Compute the gradient of \( F \) and set it to zero. We get

\[
\frac{\partial F(x)}{\partial x_i} = \frac{\partial}{\partial x_i} \sum_{k=1}^{m} \left( b_k - \sum_{j=1}^{n} x_j \phi_j(t_i) \right)^2 \\
= -2 \sum_{k=1}^{m} \left( b_k - \sum_{j=1}^{n} x_j \phi_j(t_k) \right) \phi_i(t_k) \\
= 0
\]

• After dividing by -2 and reorganizing this turns into

\[
\sum_{k=1}^{m} \sum_{j=1}^{n} x_j \phi_j(t_k) \phi_i(t_k) = \sum_{k=1}^{m} b_k \phi_i(t_k)
\]
which in matrix form can be written as the **normal equations**

\[ A^T Ax = A^T b \]

(exercise).

- A shortcut to remembering these equations is: “Just multiply with \( A^T \) on both sides of an overdetermined system \( Ax = b \)”. Of course, this does not explain anything.

- Why are these equations called “normal equations”?
Geometric Solution

\[ A = [a_1, a_2, \ldots, a_m] \]

\[ \|Ax - b\|_2 = \min \]

\[ b - Ax \text{ orthogonal to } \text{col}(A) \]

\[ (\ast) \quad a_i^T (b - Ax) = 0 \quad i = 1, \ldots, m \]

\[ (\ast\ast) \quad A^T (b - Ax) = 0 \]

\[ A^T Ax = A^T b \]
**Linear Regression—Example**

- Linear Regression means fitting scattered data in the least squares sense with a linear function. It’s probably available on your calculator.

- In terms of our previous notation we pick
  
  \[ \phi_1(t) = 1, \quad \phi_2(t) = t, \quad p(t) = x_1 + x_2t. \]

- We get

  \[
  A = \begin{bmatrix}
  1 & t_1 \\
  1 & t_2 \\
  \vdots & \vdots \\
  1 & t_m
  \end{bmatrix},
  \]

  \[
  A^T A = \begin{bmatrix}
  \sum_{k=1}^m 1 \cdot 1 & \sum_{k=1}^m 1 \cdot t_k \\
  \sum_{k=1}^m 1 \cdot t_k & \sum_{k=1}^m t_k^2
  \end{bmatrix} = \begin{bmatrix}
  \sum_{k=1}^m 1 & \sum_{k=1}^m t_k \\
  \sum_{k=1}^m t_k & \sum_{k=1}^m t_k^2
  \end{bmatrix}
  \]

  and

  \[
  A^T b = \begin{bmatrix}
  \sum_{k=1}^m b_k \\
  \sum_{k=1}^m b_k t_k
  \end{bmatrix}
  \]

- Your calculator sets up and solves the \( 2 \times 2 \) system

  \[
  A^T A x = A^T b.
  \]
• Example: Consider the data in the Table:

<table>
<thead>
<tr>
<th>i</th>
<th>( t_i )</th>
<th>( b_i )</th>
<th>( t_i^2 )</th>
<th>( t_i b_i )</th>
<th>( p(t_i) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>27/28</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>61/28</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>4</td>
<td>9</td>
<td>12</td>
<td>95/28</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>3</td>
<td>9</td>
<td>9</td>
<td></td>
</tr>
</tbody>
</table>

• We get the system

\[
A^T A x = \begin{bmatrix} 5 & 11 \\ 11 & 27 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 13 \\ 32 \end{bmatrix}
\]

which has the solution

\[
x_1 = -\frac{1}{14} \quad \text{and} \quad x_2 = \frac{17}{14}.
\]

![Figure 1. Example Least Squares.](image)

• see Figure 1 for a graphical illustration.
A Better Way

- However, solving the normal equations is a poor technique, because the condition number of $A^TA$ is the square of the condition number of $A$. (We will define the condition number of a rectangular matrix later).

- A better way is based on the $QR$ factorization of $A$:

\[
A = QR
\]

where $Q$ is $m \times m$ orthogonal, i.e.,

\[
Q \ TQ = I,
\]

and

\[
R = \begin{bmatrix}
R_1 \\
0
\end{bmatrix}
\]

where $R_1$ is $n \times n$ upper triangular.

- Thus $R = Q^TA$.

- The significance of orthogonal matrices is that they do not alter the 2-norm of a vector (including the 2-norm of an error):

\[
\|Qx\|_2^2 = (Qx)^TQx = x^TQ^TQx = x^Tx = \|x\|_2^2.
\]

- Writing

\[
Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}
\]

where $Q_1$ is $m \times n$ and $Q_2$ is $m \times (m-n)$ our problem turns into

\[
\|Ax - b\|_2^2 = \|Q^T(Ax - b)\|_2^2
\]

\[
= \left\| \begin{bmatrix}
R_1 \\
0
\end{bmatrix}x - \begin{bmatrix} Q_1^Tb \\
Q_2^Tb \end{bmatrix} \right\|_2^2
\]

\[
= \|R_1x - Q_1^Tb\|_2^2 + \|Q_2^Tb\|_2^2
\]

\[
= \min!
\]

\[
R_1x = Q_1^Tb
\]
The QR approach

Suppose we write

$$A = QR$$

(5)

where

$$Q = \begin{pmatrix} n & m - n \\ m & Q_1 & Q_2 \end{pmatrix}$$

(6)

is orthogonal and

$$R = \begin{pmatrix} n \\ m - n \end{pmatrix} \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$$

(7)

with $R_1$ being upper triangular. An orthogonal matrix $Q$ is one that satisfies

$$Q^{-1} = Q^T.$$  

(8)

Obviously, if $Q$ is orthogonal, so is $Q^T$. An orthogonal matrix is never singular (why), its condition number (w.r.t. $\| \cdot \|_2$) is 1, and it is not always symmetric. A significant property of an orthogonal matrix is that multiplying with it does not alter the (Euclidean) norm of a vector:

$$\| Qx \|_2^2 = (Qx)^T(Qx) = x^TQ^TQx = x^Tx = \| x \|_2^2.$$  

(9)

Thus the first $n$ columns of $Q$ form an orthonormal basis of the column space of $A$. We obtain

$$\| Ax - b \|_2^2 = \| Q^T(Ax - b) \|_2^2$$

$$= \| Q^T Ax - Q^Tb \|_2^2$$

$$= \left\| \begin{pmatrix} R_1 x \\ 0 \end{pmatrix} - \begin{pmatrix} Q_1^Tb \\ Q_2^Tb \end{pmatrix} \right\|_2^2$$

$$= \| R_1 x - Q_1^Tb \|_2^2 + \| Q_2^Tb \|_2^2.$$  

(10)
Of the two terms on the right we have no control over the second, and we can render the first one zero by solving (the square triangular \( n \times n \) linear system)

\[
R_1x = Q_1^T b.
\]  

(11)

Note that we do not use \( A^T A \) with it’s squared condition number, and of course we don’t have to calculate \( Q_2 \).

**Computation of the QR factorization**

Actual Computation of the QR factorization is based on

**Householder Reflections**

Given a vector

\[
u \in \mathbb{R}^m : \|u\|_2 = 1
\]

(12)

the corresponding Householder reflection (or Householder matrix) \( H \) is defined by

\[
H = I - 2uu^T \quad \text{with} \quad u^Tu = 1.
\]

(13)

The following properties of \( H \) are easily verified:

1. It is symmetric, i.e., \( H^T = H \).
2. It is orthogonal, i.e., \( H^{-1} = H^T \).
3. It projects vectors through the (hyper-)plane orthogonal to \( u \). Specifically:

\[
Hu = u - 2uu^Tu = u - 2u = -u
\]

(14)

and

\[
u^Tv = 0 \implies Hv = v - 2uu^Tv = v.
\]

(15)

Thus \( u \) gets transformed into its negative and a vector \( v \) in the plane is preserved.
Zeroing a column

The key ingredients in computing the QR factorization are Householder reflections $H$ that take a vector $a$ (which basically will be a column of $A$) and take it to a vector $Ha$ that is zero below the first entry. Let $e$ denote the vector that is 1 in the first entry, and zero everywhere else. We write $u$ as

$$u = \frac{v}{\|v\|_2}. \quad (16)$$

Thus we first find a vector $v$ that has the right direction, and then we normalize it. Since multiplication with an orthogonal matrix does not alter the norm of a vector we have to have

$$Ha = (I - 2uu^T)a = a - (2u^Ta) \frac{v}{\|v\|_2} = \pm \|a\|_2 e. \quad (17)$$

So clearly, $v$ must be a linear combination of $a$ and $e$. So we let

$$v = a + \alpha e \quad (18)$$

where $\alpha$ is as yet unknown. (Thus we have reduced the problem from finding an unknown vector to the problem of finding an unknown scalar.) Letting $a_1$ denote the first entry of $A$, observe that

$$v^Tv = a^Ta + 2\alpha a_1 + \alpha^2 \quad (19)$$

and

$$Ha = a - \frac{2(a + \alpha e)^T a}{a^Ta + 2\alpha a_1 + \alpha^2}(a + \alpha e)$$

$$= \left(1 - \frac{2(a + \alpha e)^T a}{a^Ta + 2\alpha a_1 + \alpha^2}\right)a - \frac{2\alpha v^T a}{v^Tv} e. \quad (20)$$

Since $Ha$ is a multiple of $e$ the coefficient of $a$ must vanish, i.e.,

$$2a^Ta + 2\alpha a_1 = a^Ta + 2\alpha a_1 + \alpha^2. \quad (21)$$
This gives
\[ \|a\|_2^2 = \alpha^2, \quad (22) \]
i.e.,
\[ \alpha = \pm \|a\|_2. \quad (23) \]
Thus
\[ u = \frac{a \pm \|a\|_2 e}{\|a \pm \|a\|_2 e\|_2}. \quad (24) \]

Since the purpose of doing these calculations in such detail is to illustrate common techniques it is worthwhile to check directly that this choice works.

We obtain
\[
\|a \pm \|a\|_2 e\|_2^2 = (a \pm \|a\|_2 e)^T (a \pm \|a\|_2 e) \\
= a^T a \pm 2\|a\|_2 a_1 + \|a\|_2^2 \\
= 2 (a^T a \pm \|a\|_2 a_1). 
\]

Thus
\[
Ha = \left( I - 2 \frac{(a \pm \|a\|_2 e) (a \pm \|a\|_2 e)^T}{\|a \pm \|a\|_2 e\|_2^2} \right) a \\
= a - \frac{2(a \pm \|a\|_2 e)^T a}{\|a \pm \|a\|_2 e\|_2^2} (a \pm \|a\|_2 e) \\
= a - \frac{2(a^T a \pm \|a\|_2 e^T a)}{\|a \pm \|a\|_2 e\|_2^2} (a \pm \|a\|_2 e) \\
= a - \frac{2(a^T a \pm \|a\|_2 a_1)}{\|a \pm \|a\|_2 e\|_2^2} (a \pm \|a\|_2 e) \\
= a - (a \pm \|a\|_2 e) \\
= \pm \|a\|_2 e. 
\]

So how do we pick the sign of \( \alpha \)? The usual choice is
\[ \alpha = \text{sign} \ (a_1) \|a\|_2. \]
This makes the first entry of $u$ as large as possible (in absolute value) and tends to counteract the risk of dividing by a small number when normalizing $v$. So our final choice is

$$u = \frac{a + \text{sign}(a_1)\|a\|_2 e}{\|a + \text{sign}(a_1)\|a\|_2 \|2 \|e\|}. \quad (27)$$

It is also worth checking that the coefficient of $e$ in (20) has absolute value equal to $\|a\|_2$, as it must since multiplication with an orthogonal matrix does not alter the 2-norm of a vector. Using (19) and (23) we have

$$\frac{2\alpha v^T a}{v^T v} = \frac{2\alpha (a^T a + \alpha a_1)}{a^T a + 2\alpha a_1 + \alpha^2} = \frac{2\alpha (a^T a + \alpha a_1)}{2a^T a + 2\alpha a_1} = \alpha = \pm \|a\|. \quad (28)$$

**Putting it together**

Let $H_1$ be the Householder reflection that zeros the first column of $A$ below the diagonal. Thus

$$H_1 A = \begin{bmatrix} x & x \\ 0 & \hat{A} \end{bmatrix} \quad (29)$$

where $\hat{A}$ is an $m-1 \times n-1$ matrix (and $x$ is a generic notation for non-zero entries). Let $\hat{H}_2$ be the $m-1 \times m-1$ Householder reflection that similarly reduces the first of column of $\hat{A}$ to zero below the diagonal. Then let

$$H_2 = \begin{bmatrix} 1 & 0^T \\ 0 & \hat{H}_2 \end{bmatrix} \quad (30)$$

Clearly

$$H_2 H_1 A = \begin{bmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & A_3 \end{bmatrix}. \quad (31)$$
Continuing in this fashion we construct Householder reflections $H_1, H_2, \ldots, H_n$ such that

$$H_nH_{n-1} \cdots H_2H_1A = R. \quad (32)$$

The matrix $Q_1^T$ contains the first $n$ rows of the product $H_nH_{n-1} \cdots H_2H_1$. As a practical matter, of course we do not store the (full $m \times m$) matrices $H_i$, but rather only the vectors that define them.

### Multiplying with Householder Reflections

Note that

$$HA = (I - 2uu^T)A = A - 2u(u^TA). \quad (33)$$

Multiplication of $A$ with a Householder reflection changes $A$ by a matrix of rank 1. This can be implemented in $O(n^2)$ operations. If the multiplication is implemented as an ordinary matrix multiplication $O(n^3)$ operations are required, which would be grossly wasteful.

### The Gram-Schmidt Process

A better known method for computing the QR factorization is the Gram-Schmidt process. It is not as stable numerically as the Householder approach, as discussed in Golub and van Loan. However, it applies in more general inner product spaces and is worth knowing.

So consider again the factorization $A = QR$, where $Q$ is orthogonal and $R$ is triangular. It’s clear that the first column of $A$ is a multiple of the first column of $Q$. In general, the first $k$ columns of $A$ are linear combinations of the first $k$ columns of $Q$, and vice versa. Thus the first $k$ columns of $Q$ span the same space as the first $k$ columns of $A$. So we can think of computing $Q$ as the first $n$ steps in the following problem:

Given a sequence of linearly independent vectors

$$a_1, a_2, a_3, \ldots \quad (34)$$
construct an orthonormal sequence of vectors

q_1, q_2, q_3, \ldots \quad (35)

such that

q_i^T q_j = \begin{cases} 
1 & \text{if } i = j \\
0 & \text{else}
\end{cases} \quad (36)

and

\text{span} \{q_1, q_2, \ldots q_k\} = \text{span} \{a_1, a_2, \ldots a_k\}, \quad k = 1, 2, 3, \ldots

This problem can be solved as follows:

1. Let

q_1 = \frac{a_1}{\|a_1\|} \quad (37)

2. For k = 1, 2, \ldots
   a. Define v_k = a_k - \sum_{j=1}^{k-1} q_j^T a_k q_j
   b. Let q_k = \frac{v_k}{\|v_k\|}.

It’s clear by induction that this sequence has the desired properties since, for i < k,

v_k^T q_i = a_k^T q_i - \sum_{j=1}^{k-1} a_j^T a_k q_i^T q_j = a_k^T q_i - a_k^T q_i = 0. \quad (38)