Linear Least Squares Optimization

MA 348 Optimization

Kurt Bryan

Linear Least-Squares

Consider an objective function of the special form

$$f(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{m} f_i^2(\mathbf{x}) \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ and each function f_i is LINEAR, that is, of the form $f_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$ for some $\mathbf{a}_i \in \mathbb{R}^n$ and some scalar b_i. Such functions occur frequently when fitting linear models to data; usually m is larger than n, sometimes much larger. For example, one might try to fit a model $\phi(x) = a + bx + cx^2$ to (x, y) data points $(1.0, 2.2), (1.6, 2.8), (2.3, 3.9), (3.4, 4.4)$, and $(4.1, 5.2)$, by adjusting a, b, and c to minimize the squared error

$$f(a,b,c) = \frac{1}{2}((\phi(1.0)-2.2)^2+(\phi(1.6)-2.8)^2+(\phi(2.3)-3.9)^2+(\phi(3.4-4.4)^2+(\phi(4.1)-5.2)^2).$$

In this case $\mathbf{x} = (a, b, c)$, $f_1(a, b, c) = a + b + c - 2.2$, $f_2(a, b, c) = a + 1.6b + (1.6)^2c - 2.8$, etc. The factor of $1/2$ above and in equation (1) is there only to make some formulas that appear later prettier; it doesn't really change the problem.

Another typical situation in which such objective functions arise is that of solving $\mathbf{Ax} = \mathbf{b}$ when there are more equations than unknowns (or more generally when the system is inconsistent). In the case that $\mathbf{Ax} = \mathbf{b}$ has no solution, we might instead seek that $\mathbf{x}$ which does the "best" job of solving the equations, in the sense that $\mathbf{x}$ minimizes

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{Ax} - \mathbf{b}\|^2 \tag{2}$$

where $\|\mathbf{p}\|^2 = \sum_i p_i^2$ is the usual Euclidean 2-norm. This fits into the mold of equation (1) where $\mathbf{a}_i$ in (1) is the ith row of $\mathbf{A}$. Thus the problem of minimizing a function of the form in (2) can be cast in the form of minimizing f in equation (1).

In fact, the converse is also true: minimization of the function f in (1) can be cast into the form (2) by taking $\mathbf{A}$ to have the vectors $\mathbf{a}_i^T$ as rows and $\mathbf{b} = (b_1, \ldots, b_n)^T$. The problems of minimizing the functions in equations (1) and (2) are thus totally equivalent.

Of course you could attack this optimization problem with any general nonlinear algorithm, e.g., conjugate gradients or quasi-Newton methods. But the objective function (1) has a very special structure—setting the gradient to zero leads to a linear system of equations which can be solved quite efficiently.

The Normal Equations

The objective function in (2) can be written out in gory detail as

$$f(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{m}(\sum_{j=1}^{n} A_{ij}x_j - b_i)^2 \tag{3}$$

Differentiate with respect to x_k to find

$$\frac{\partial f}{\partial x_k} = \sum_{i=1}^{m}\sum_{j=1}^{n} A_{ij}A_{ik}x_j - \sum_{i=1}^{m} A_{ik}b_i. \tag{4}$$

We can do the double sum in any order we like. You can easily check that $\sum_{i=1}^{m} A_{ik}b_i$ is just the kth component of $\mathbf{A}^T\mathbf{b}$, which I'll write as $(\mathbf{A}^T\mathbf{b})_k$. Also, $\sum_{i=1}^{m} A_{ij}A_{ik}$ is just $(\mathbf{A}^T\mathbf{A})_{jk}$ or $(\mathbf{A}^T\mathbf{A})_{kj}$ (since $\mathbf{A}^T\mathbf{A}$ is symmetric), so that $\sum_{j=1}^{n}(\sum_{i=1}^{m} A_{ij}A_{ik})x_j$ is just $(\mathbf{A}^T\mathbf{A}\mathbf{x})_k$.

If we arrange ∇f as a column vector (stack components $k = 1$ to $k = n$) we find that $\nabla f(x) = \mathbf{A}^T\mathbf{A}\mathbf{x} - \mathbf{A}^T\mathbf{b}$ and the condition for a critical point is that $\nabla f = 0$, or

$$\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{b}. \tag{5}$$

These are the so-called *normal equations*. They ought to look familiar from DE I.

Solving the Normal Equations

The normal equations are just an n by n system of linear equations for $\mathbf{x}$. The matrix $\mathbf{A}^T\mathbf{A}$ is symmetric and positive semi-definite. In fact this matrix is very likely positive definite (exactly when $\mathbf{A}$ has full column rank), so we can try solving by using Cholesky Factorization, which buys a factor of 2 efficiency over LU. This is a reasonable way to solve the problem, although you can sometimes run into trouble in certain cases $\mathbf{A}$.

As an example, consider the case in which

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ \epsilon & -2\epsilon \\ 0 & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

with $\epsilon = 10^{-5}$. The matrix $\mathbf{A}$ is "close" to being rank 1, but it seems reasonable that if we work in 10 digit arithmetic we shouldn't have any trouble. The EXACT solution to the normal equations (5) in this case is $x_1 = 2/3, x_2 = 1/3$, which you can compute symbolically, with no rounding error. But if you solve the normal equations numerically using either Cholesky or LU factorization with 10 digit arithmetic you get $x_1 = 0.5$ and $x_2 = 0.5$, which is horrible. If you work in anything LESS than 10 digits you find that $\mathbf{A}^T\mathbf{A}$ is outright singular.

The problem is that forming the product $\mathbf{A}^T\mathbf{A}$ encourages round off error, essentially halving the number of significant figures we have at our disposal. If you want to know more, take numerical analysis. This isn't a problem if $\mathbf{A}$ is far enough from singular (though you still lose significant figures). What we'd like is a way to minimize $f(\mathbf{x})$ in equation (2) but without forming $\mathbf{A}^T\mathbf{A}$.

QR Factorization

Instead of using a Cholesky Factorization $\mathbf{A} = \mathbf{L}^T\mathbf{D}\mathbf{L}$ and backsubstitution to solve, we'll use a *QR factorization*. It's a fact that any m by n matrix $\mathbf{A}$ can factored as $\mathbf{A} = \mathbf{Q}\mathbf{R}$ where $\mathbf{Q}$ is an m by m *orthogonal* matrix (meaning $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$) and $\mathbf{R}$ is an m by n upper triangular matrix. This can be used to minimize the function in equation (2).

Example

Let

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & -1 \\ 2 & 2 \end{bmatrix}.$$

The you can check that $\mathbf{A} = \mathbf{Q}\mathbf{R}$ where

$$\mathbf{Q} = \begin{bmatrix} \frac{\sqrt{5}}{5} & 0 & \frac{2\sqrt{5}}{5} \\ 0 & -1 & 0 \\ \frac{2\sqrt{5}}{5} & 0 & -\frac{\sqrt{5}}{5} \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} \sqrt{5} & \sqrt{5} \\ 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

Computing the QR Decomposition

The QR decomposition is computed by selectively transforming each column of $\mathbf{A}$ into the required form from left to right. We'll multiply $\mathbf{A}$ by an orthogonal matrix $\mathbf{Q}_1$ which will zero out all entries in the first column of $\mathbf{A}$ except the top entry. We'll then multiply $\mathbf{Q}_1\mathbf{A}$ by an orthogonal matrix $\mathbf{Q}_2$ which will zero out all but the top two elements in column 2, but without messing up column 1. We continue like this, so at the kth stage we multiply $\mathbf{Q}_{k-1}\cdots\mathbf{Q}_1\mathbf{A}$ by an orthogonal matrix $\mathbf{Q}_k$ which eliminates all but the top k elements in column k, without messing up the previous columns. If w do this for $k = 1$ to $k = n$ we obtain something like

$$\mathbf{Q}_n\mathbf{Q}_{n-1}\cdots\mathbf{Q}_1\mathbf{A} = \mathbf{R} \tag{6}$$

where $\mathbf{R}$ is upper triangular. We then have $\mathbf{Q}\mathbf{A} = \mathbf{R}$ for some orthogonal matrix $\mathbf{Q} = \mathbf{Q}_n\mathbf{Q}_{n-1}\cdots\mathbf{Q}_1$ (since the product of orthogonal matrices is orthogonal). From this we can obtain $\mathbf{A} = \mathbf{Q}\mathbf{R}$ (exercise).

Exercise

- Show that the product of orthogonal matrices is orthogonal.
- Prove $\mathbf{A} = \mathbf{Q}\mathbf{R}$ using equation (6).

The trick is to find an orthogonal matrix $\mathbf{Q}$ that selectively zeroes out all but the top k elements of a given vector (since each column of $\mathbf{A}$ is processed one at a time, we can consider them as vectors).

A useful fact to note is this: $\|\mathbf{Qx}\| = \|x\|$ for any orthogonal $\mathbf{Q}$ and vector $\mathbf{x}$, so multiplication by $\mathbf{Q}$ preserves Euclidean length. To see this just note that

$$\begin{aligned} \|\mathbf{Qx}\|^2 &= (\mathbf{Qx}) \cdot (\mathbf{Qx}) \\ &= (\mathbf{x}^T\mathbf{Q}^T)(\mathbf{Qx}) \\ &= \mathbf{x}^T\mathbf{x} \\ &= \|\mathbf{x}\|^2 \end{aligned}$$

Exercise

- True or False: The converse of the above holds, i.e. if $\|\mathbf{Px}\| = \|\mathbf{x}\|$ for all $\mathbf{x}$ then $\mathbf{P}$ is orthogonal.

Here's how to construct the magic $\mathbf{Q}_i$ matrices. Let $\mathbf{v}$ be a vector in $\mathbb{R}^n$. Form the matrix

$$\mathbf{H} = \mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} \tag{7}$$

an n by n matrix. The matrix $\mathbf{H}$ is orthogonal, for

$$\begin{aligned} \mathbf{H}^T\mathbf{H} &= (\mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}})^T(\mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}) \\ &= (\mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}})(\mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}) \\ &= \mathbf{I} - 4\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} + 4\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} \\ &= \mathbf{I} - 4\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} + 4\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} \\ &= \mathbf{I} \end{aligned}$$

since the middle $\mathbf{v}^T\mathbf{v}$ in the last term on the second-to-last line is just a scalar and cancels with one of the denominator copies. Matrices of the form of equation (7) are called *Householder* matrices.

Let $\mathbf{a}$ be a vector in $\mathbb{R}^n$. Consider the chore of choosing $\mathbf{v}$ so that $\mathbf{H}$ as defined by equation (7) has the effect that forming $\mathbf{Ha}$ zeros out all but the first element of $\mathbf{a}$, i.e.,

$$\mathbf{Ha} = \begin{bmatrix} \alpha \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \alpha\mathbf{e}_1.$$

Since $\mathbf{H}$ is length-preserving we must have $\alpha = \pm\|\mathbf{a}\|$. How should we choose $\mathbf{v}$? If you write out $\mathbf{Ha}$ explicitly you obtain

$$\mathbf{Ha} = \left(\mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}\right)\mathbf{a} = \mathbf{a} - 2\mathbf{v}\frac{\mathbf{v}^T\mathbf{a}}{\mathbf{v}^T\mathbf{v}} = \alpha\mathbf{e}_1$$

or, if we solve for $\mathbf{v}$,

$$\mathbf{v} = (\mathbf{a} - \alpha \mathbf{e}_1) \frac{\mathbf{v}^T \mathbf{v}}{2\mathbf{v}^T \mathbf{a}}. \tag{8}$$

Now the quantity $\frac{\mathbf{v}^T \mathbf{v}}{2\mathbf{v}^T \mathbf{a}}$ is just a scalar, and if you examine the definition of $\mathbf{H}$ you see that multiplying $\mathbf{v}$ by a scalar doesn't change anything. We then might just as well scale $\mathbf{v}$ so $\frac{\mathbf{v}^T \mathbf{v}}{2\mathbf{v}^T \mathbf{a}} = 1$ and so take

$$\mathbf{v} = \mathbf{a} \pm \|\mathbf{a}\| \mathbf{e}_1 \tag{9}$$

where I've used $\alpha = \pm \|\mathbf{a}\|$. It's conventional to take the plus sign in equation (9) if $a_1 \geq 0$ and the minus sign if $a_1 < 0$, to avoid cancellation errors.

Example:

Let $\mathbf{a} = [1, 2, 4, -2]^T$. Then $\|\mathbf{a}\| = 5$ and we obtain $\mathbf{v} = \mathbf{a} + [5, 0, 0, 0]^T = [6, 2, 4, -2]^T$. The matrix $\mathbf{H}$ is

$$\mathbf{H} = \begin{bmatrix} -\frac{1}{5} & -\frac{2}{5} & -\frac{4}{5} & \frac{2}{5} \\ -\frac{2}{5} & \frac{13}{15} & -\frac{4}{15} & \frac{2}{15} \\ -\frac{4}{5} & -\frac{4}{15} & \frac{7}{15} & \frac{4}{15} \\ \frac{2}{5} & \frac{2}{15} & \frac{4}{15} & \frac{13}{15} \end{bmatrix}.$$

You can check that $\mathbf{Ha} = [-5, 0, 0, 0]^T$.

To zero out all but the top k elements of a vector $\mathbf{a}$, partition $\mathbf{a}$ into two pieces

$$\mathbf{a} = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix}$$

where $\mathbf{a}_1$ is in $\mathbb{R}^{k-1}$ and $\mathbf{a}_2$ is in $\mathbb{R}^{n-k+1}$. We can find a vector $\mathbf{v}_2 \in \mathbb{R}^{n-k+1}$ and corresponding $n - k + 1$ by $n - k + 1$ Householder matrix $\mathbf{H}_2$ with the property that

$$\mathbf{H}_2 \mathbf{a}_2 = \begin{bmatrix} \alpha \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Now let $\mathbf{v}$ be defined by

$$\mathbf{v} = \begin{bmatrix} \mathbf{0} \\ \mathbf{v}_2 \end{bmatrix}.$$

The corresponding Householder matrix looks like

$$\mathbf{H} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_2 \end{bmatrix}$$

and has the desired effect of zeroing out the bottom $n - k$ elements of $\mathbf{a}$.

Now that we have a recipe for constructing orthogonal matrices which zero all but the first k elements of a vector $\mathbf{a}$, we can apply the recipe leading to equation (6). The $\mathbf{Q}_i$ will be appropriate Householder matrices. There are certain efficiency and numerical issues we haven't addressed, but the sequence of Householder transformations as presented above are the basic technique used by most software for computing the QR factorization. And the QR algorithm, properly implemented, is very stable numerically.

Solving Least-Squares Problems with QR Decomposition

Let's return to $\mathbf{Ax} = \mathbf{b}$. We factor $\mathbf{A} = \mathbf{QR}$ to obtain

$$\mathbf{QRx} = \mathbf{b}.$$

Multiply both sides by $\mathbf{Q}^T$ to find

$$\mathbf{Rx} = \mathbf{Q}^T\mathbf{b}.$$

Now for simplicity let's consider the case where $\mathbf{R}$ is of full column rank. This means that $\mathbf{R}$ has non-zero elements in positions $\mathbf{R}_{kk}$ for $k = 1$ to n. This is certainly the generic case. In this case we can write

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix}$$

where $\mathbf{R}_1$ is an n by n invertible upper triangular matrix and $\mathbf{0}$ means an $m-n$ by n matrix of zeros. We can write $\mathbf{Rx} = \mathbf{Q}^T\mathbf{b}$ as

$$\begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix} \tag{10}$$

where $\mathbf{c}_1$ is the first n components of $\mathbf{Q}^T\mathbf{b}$ and $\mathbf{c}_2$ the last $m-n$ components. This matrix equation is really two equations, namely $\mathbf{R}_1\mathbf{x} = \mathbf{c}_1$ and $\mathbf{0x} = \mathbf{c}_2$. We can't do anything with the second equation—no matter what the value of $\mathbf{x}$, the equation $\mathbf{0x} = \mathbf{c}_2$ will never be satisfied (unless we're so lucky that $\mathbf{c}_2 = \mathbf{0}$). But we can solve the first equation $\mathbf{R}_1\mathbf{x} = \mathbf{c}_1$ exactly (since $\mathbf{R}_1$ is invertible), by doing a simple backsubstitution. Moreover, since $\mathbf{Q}$ is orthogonal we have (for any choice of $\mathbf{x}$) that

$$\begin{aligned} \|\mathbf{Ax} - \mathbf{b}\|^2 &= \|\mathbf{QRx} - \mathbf{b}\|^2 \\ &= \|\mathbf{Rx} - \mathbf{Q}^T\mathbf{b}\|^2 \\ &= \|\mathbf{R}_1\mathbf{x} - \mathbf{c}_1\|^2 + \|\mathbf{c}_2\|^2. \end{aligned} \tag{11}$$

We clearly minimize the right side of equation (11) (and hence the value of $\|\mathbf{Ax} - \mathbf{b}\|$) by taking $\mathbf{x}$ so that $\mathbf{R}_1\mathbf{x} = \mathbf{c}_1$, which yields the least-squares solution.

Example:

Let

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}.$$

Then $\mathbf{A} = \mathbf{QR}$ with

$$\mathbf{Q} = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{6}}{6} & \frac{\sqrt{3}}{3} \\ 0 & \frac{\sqrt{6}}{3} & -\frac{\sqrt{3}}{3} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{6}}{6} & -\frac{\sqrt{3}}{3} \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} \sqrt{2} & \frac{3}{2}\sqrt{2} \\ 0 & \frac{\sqrt{6}}{2} \\ 0 & 0 \end{bmatrix}.$$

Then $\mathbf{Rx} = \mathbf{Q}^T\mathbf{b}$ becomes

$$\begin{bmatrix} \sqrt{2} & \frac{3}{2}\sqrt{2} \\ 0 & \frac{\sqrt{6}}{2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \frac{3}{2}\sqrt{2} \\ \frac{\sqrt{6}}{6} \\ \frac{\sqrt{3}}{3} \end{bmatrix}.$$

This yields equation $\frac{\sqrt{6}}{2}x_2 = \frac{\sqrt{6}}{6}$ from which we obtain $x_2 = 1/3$. The first equation then yields $x_1 = 1$. This is the least squares solution to $\mathbf{Ax} = \mathbf{b}$.

Stability

The QR decomposition is a more stably way to solve the linear least squares problem. We won't do a detailed analysis, but let's reconsider the problem from above in which

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ \epsilon & -2\epsilon \\ 0 & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

with $\epsilon = 0.00001$. If you QR factor $\mathbf{A}$ you obtain (using 10 significant figure arithmetic)

$$\mathbf{Q} = \begin{bmatrix} 1.0 & 0.00001 & 0.0 \\ 0.00001 & -1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} -1.0 & -1.0 \\ 0.0 & 0.00003 \\ 0.0 & 0.0 \end{bmatrix}.$$

and then form $\mathbf{Rx} = \mathbf{Q}^T\mathbf{b}$, which leads to equations $1.0x_1 + 0.9999999997x_2 = 1.0$ and $0.00003x_2 = 0.00001$, with solutions $x_1 = 0.6666666668$ and $x_2 = 0.3333333333$, which are correct to 9 places.