Least-Squares Approximation and Bilinear Forms

The Normal Equations :

Suppose a column vector g is given in an Euclidean space into which a given matrix F maps another real space of column vectors u . In the Euclidean space, length $\|g\| := \sqrt{(g^Tg)}$. Usually matrix F is rectangular with more rows than columns. Our task is to choose u to minimize $\|Fu - g\|$, which will then be the distance from g to Range(F) . This task is called "Least-Squares" because the u we seek would minimize the sum of squares of the elements of $Fu - g$. Differentiating the sum of squares produces $d((Fu - g)^T(Fu - g)) = 2(Fu - g)^TF\,du$, which vanishes for all (infinitesimal) perturbations du if and only if $(Fu - g)^TF = o^T$. (Do you see why?) Transposed, this becomes the " Normal Equations " of the Least-Squares Problem,

$$(F^TF)u = F^Tg \;;$$

u must satisfy them to minimize $\|Fu - g\|$ and so best approximate g by a vector in Range(F) .

Do the Normal Equations have at least one solution $u = û$?

If so, is $\|Fû - g\| \le \|Fu - g\|$ for all u ? *i.e.,* does û minimize rather than maximize?

These questions among others are addressed in this note.

At first sight one might think the Normal Equations' solution should be $û = (F^TF)^{-1}F^Tg$. But this formula fails if the columns of F are linearly dependent. To see why, observe that

$$\text{“ } Fz = o \text{ ”} \iff \text{“ } (Fz)^T(Fz) = z^T(F^TF)z = 0 \text{ ”} \iff \text{“ } (F^TF)z = o \text{ ”},$$

so F^TF is invertible (nonsingular) if and only if the columns of F are linearly independent. This hasn't been assumed; in fact matrix F could be rectangular with more columns than rows.

Exercise: Show that, if the rows of F are linearly independent, a solution $û = F^T(FF^T)^{-1}g$ and, if not the only solution of the Normal Equations, it is the only solution that minimizes $û^Tû$ too.

In short, if neither the rows nor the columns of F are linearly independent, neither FF^T nor F^TF need be invertible, and then the existence of a minimizing solution $u = û$ is in question.

Warning: Even when an indicated inverse exists, neither formula $û = (F^TF)^{-1}F^Tg$ nor $û = F^T(FF^T)^{-1}g$ should be used with numerical data unless the computer's arithmetic carries at least twice as many sig. digits as are trusted in the data [F, g] or desired in the result û . Otherwise roundoff will degrade the result û too badly whenever F is too near a matrix of lower rank. The reason behind this warning will become clear after *Singular Values* have been discussed. If the arithmetic carries barely more sig. digits than are trusted in the data or desired in the result, it should be computed by means of a *QR factorization*, which will also be discussed later. Matlab uses such a factorization to compute û , which Matlab calls " `F\g` ", whenever F is not square. Least-Squares is built into Matlab.

Existence and Uniqueness of a Minimizing Solution û :

We shall use Fredholm's Alternatives (*q.v.*) to deduce that the Normal Equations always have at least one solution û , and to determine when it is unique. At least one solution exists if and only if $w^T(F^Tg) = 0$ whenever $w^T(F^TF) = o^T$, so consider any row w^T that satisfies the last equation. It must satisfy also $0 = w^T(F^TF)w = (Fw)^T(Fw)$, which implies $Fw = o$, which implies $w^T(F^Tg) = (Fw)^Tg = 0$, whereupon Fredholm's Alternative (1) implies that the Normal Equations have at least one solution û . It is unique if and only if the columns of F are linearly independent; otherwise add any nonzero solution z of $Fz = o$ to one û to get another.

How do we know that setting $u = \hat{u}$ minimizes $\|Fu - g\|$? For every u we find

$$\|Fu - g\|^2 - \|F\hat{u} - g\|^2 = \|F(u - \hat{u}) + (F\hat{u} - g)\|^2 - \|F\hat{u} - g\|^2$$
$$= \|F(u - \hat{u})\|^2 + 2(F(u - \hat{u}))^T(F\hat{u} - g) \qquad (\text{ since } \|z\|^2 = z^Tz)$$
$$= \|F(u - \hat{u})\|^2 + 2(u - \hat{u})^TF^T(F\hat{u} - g) = \|F(u - \hat{u})\|^2 \ge 0 ,$$

with equality instead of inequality just when u is a(nother) solution of the Normal Equations.

When the Normal Equations have many solutions $\hat{u}$, which does Matlab choose for `F\g` ? It has a near minimal number of nonzero elements. A different solution minimizes $\|\hat{u}\|^2 := \hat{u}^T\hat{u}$, as if also the space of vectors u were Euclidean. This doubly minimizing solution $\hat{u}$ satisfies both the Normal Equations $(F^TF)\hat{u} = F^Tg$ and an auxiliary equation $\hat{u} = F^TFv$ for some vector v of “Lagrange Multipliers.” In consequence v satisfies $(F^TF)^2v = F^Tg$, an equation with least one solution v whose existence is assured by an application of Fredholm’s first alternative very much like before except that the hypothesis $w^T(F^TF) = o^T$ is replaced by $w^T(F^TF)^2 = o^T$. (Can you carry out this inference?) Every other solution u of the Normal Equations satisfies $\|u\|^2 - \|\hat{u}\|^2 = \|(u-\hat{u}) + \hat{u}\|^2 - \|\hat{u}\|^2 = \ldots$
$\ldots = \|u-\hat{u}\|^2 + 2\hat{u}^T(u-\hat{u}) = \|u-\hat{u}\|^2 + 2v^TF^TF(u-\hat{u}) = \|u-\hat{u}\|^2 \ge 0$, so this $\hat{u} = F^TFv$ really is doubly minimizing; moreover it is determined uniquely by the data $[F, g]$. (Can you see why?) As we shall see later after Singular Values have been discussed, there is a matrix $F^\dagger$ called the “Moore-Penrose Pseudo-Inverse” of F such that the doubly minimizing $\hat{u} = F^\dagger g$ is a linear function of g . (Matlab’s name for $F^\dagger$ is pinv(F).) However, whenever neither F^TF nor FF^T is invertible, so $F^\dagger$ is interesting, it turns out to be a violently discontinuous function of F . This renders the doubly minimizing $\hat{u}$ doubly dubious because the space of vectors u need not be Euclidean. Matlab’s `F\g` can be discontinuous too, even when FF^T is invertible and the doubly minimizing $\hat{u}$ is continuous.

Linear Regression:

Least-Squares approximation has been applied to statistical estimation for over two centuries. An m-by-n matrix F is assumed given with linearly independent columns (so $m \ge n$); and a given m-vector $g = y + q$ of “data” is thought to include a systematic contribution y and a “random error” q . The question is how near is y to Range(F) ? The answer is obscured by the random error. The elements of this error q are assumed *independently distributed* with *mean* 0 and known *variance* $ß^2$. These terms are given meaning by an *Averaging* or *Expectation* operator Æ which acts upon every random variable r linearly to produce Ær , the average or *mean* of the population of values of r . Thus $Æq = o$ because every element of q has mean 0 ; and q has *covariance* matrix $Æ((q–Æq)(q–Æq)^T) = ß^2I$ since the square of every element of q has mean $ß^2$ but every product of different elements of q has mean 0 because they are independent. The smaller is ß , the less uncertainty does random error q introduce into the data g .

Define $x := (F^TF)^{-1}F^Ty$ to minimize $\|Fx - y\|$ although neither y nor x can be known. As the known g approximates y , so is x approximated by whatever $\hat{u}$ minimizes $\|F\hat{u} - g\|$. Get $\hat{u} = (F^TF)^{-1}F^Tg$; how well can it approximate x ? Since $Æg = y$, we find that $Æ\hat{u} = x$, so $\hat{u}$ is an *unbiased* estimate of x . The covariance matrix of $\hat{u}$ is computable too; it is

$$Æ((\hat{u} - x)(\hat{u} - x)^T) = Æ((F^TF)^{-1}F^Tqq^TF(F^TF)^{-1}) = (F^TF)^{-1}F^TÆ(qq^T)F(F^TF)^{-1} = ß^2(F^TF)^{-1} .$$

The smaller this is, the better does $\hat{u}$ approximate x on average. The smaller is $\|Fx - y\|$, the smaller do we expect $\|F\hat{u} - g\|$ to be. How small should we expect it to be? A calculation below shows that $Æ(\|F\hat{u} - g\|^2) = \|Fx - y\|^2 + (m–n)ß^2$. It means that $\|F\hat{u} - g\|$ is unlikely to exceed $ß\sqrt{(m–n)}$ much if y lies in or very near Range(F) ; conversely, $\|Fx - y\|$ is unlikely to be much smaller than $\|F\hat{u} - g\|$ if this is many times bigger than $ß\sqrt{(m–n)}$. Explanation follows.

Proof that $Æ(||F\hat{u} - g||^2) = ||Fx - y||^2 + (m–n)ß^2$: The Trace of a square matrix is defined to be the sum of its diagonal elements; evaluate this sum to confirm that $Trace(B^TC) = Trace(C\,B^T)$ for any matrices B^T and C whose products B^TC and $C\,B^T$ are both square, though perhaps of different dimensions. Next define $H := F(F^TF)^{-1}F^T$ and confirm that $H^T = H = H^2$. (H is the orthogonal projector onto Range(F) because " p = Fz for some z " $\Longleftrightarrow$ " p = Hp " , so Range(F) = Range(H) , and " Hz = o " $\Longleftrightarrow$ " $z^TH = o^T$ " , so $Nullspace(H) = Range(H)^\perp$.) Shortly we shall have use for $Trace(H) = Trace((F^TF)^{-1}F^TF) = Trace(I_n) = n$. Now we observe that û and x are so defined that $F\hat{u} – g = (H – I)g$ and $Fx – y = (H – I)y$ wherein I is the m-by-m identity matrix. Consequently

$$\begin{aligned} Æ(||F\hat{u} – g||^2) &= Æ(((H–I)g)^T(H–I)g) = Æ(Trace((H–I)g((H–I)g)^T)) && \dots \text{ because } Trace(b^Tc) = Trace(cb^T) \\ &= Æ(Trace((H–I)gg^T(H–I))) = Trace((H–I)Æ(gg^T)(H–I)) && \dots \text{ because } H = H^T \text{ isn't random} \\ &= Trace((H–I)Æ(yy^T + yq^T + qy^T + qq^T)(H–I)) && \dots \text{ because } g = y + q \\ &= Trace((H–I)(yy^T + O + O + ß^2I)(H–I)) = Trace((H–I)yy^T(H–I)) + ß^2Trace((H–I)^2) \\ &= Trace((Fx – y)(Fx – y)^T) + ß^2Trace(I – H) = ||Fx – y||^2 + ß^2(m–n) && \text{as was claimed.} \end{aligned}$$

Proof that $||F\hat{u} - g||^2$ is unlikely to be many times bigger than its mean $Æ(||F\hat{u} - g||^2)$: More precisely, we shall deduce that $||F\hat{u} - g||^2$ exceeds $\lambda Æ(||F\hat{u} - g||^2)$ with probability less than $1/\lambda$ for every $\lambda > 1$. This deduction is an instance of *Tchebyshev's Inequality* : If a positive random variable ρ has mean $\mu := Æ\rho$, then the probability that $\rho \ge \lambda\mu$ cannot exceed $1/\lambda$ for any $\lambda > 1$. Here is a proof of Tchebyshev's Inequality. Let $p(\xi)$ be the probability that $\rho \le \xi$. This $p(\xi)$ is a nondecreasing function increasing from $p(0) = 0$ to $p(\infty) = 1$, and $\mu = \int_0^\infty \xi\, dp(\xi)$ by virtue of the definition of Æ . We seek an overestimate for $\int_{\lambda\mu}^\infty dp(\xi)$, which is the probability that $\rho \ge \lambda\mu$. We find that $\int_{\lambda\mu}^\infty dp(\xi) \le \int_{\lambda\mu}^\infty \xi\, dp(\xi)/(\lambda\mu) \le \int_0^\infty \xi\, dp(\xi)/(\lambda\mu) = \mu/(\lambda\mu)$, which yields the result claimed. (This can be a gross overestimate because it uses almost no information about p . For almost all values of $\lambda > 1$, and for all values of $\lambda > 1$ for almost all probability functions p , the probability that $\rho \ge \lambda\mu$ is actually far tinier than $1/\lambda$.)

Thus the computed $||F\hat{u} – g||^2$ is unlikely to be many times bigger than $||Fx – y||^2 + ß^2(m–n)$ in which $ß^2(m–n)$ is given and $||Fx – y||^2$ is unknown, whence something probabilistic can be inferred about the unknown. Another similar application of Least-Squares is to the assumption that y = Fx and g = y + q for a random error q about which $ß^2$ is unknown but estimated from $||F\hat{u} – g||^2/(m–n)$. These applications are treated in Statistics courses.

Abstract Least-Squares:

Suppose a column vector g is given in an Euclidean space into which a given linear operator **F** maps a real space of abstract vectors **u** . In the Euclidean space, length $||g|| := \sqrt{(g^Tg)}$, but no such length is defined (yet) for Domain(**F**) . Again our task is to choose **u** to minimize ||**Fu** – g|| , which will then be the distance from g to Range(**F**) . Differentiating the sum of squares $||\mathbf{Fu} – g||^2 = (\mathbf{Fu} – g)^T(\mathbf{Fu} – g)$ produces $d((\mathbf{Fu} – g)^T(\mathbf{Fu} – g)) = 2(\mathbf{Fu} - g)^T\mathbf{F}\, d\mathbf{u}$, which vanishes for all (infinitesimal) perturbations d**u** if and only if $(\mathbf{Fu} – g)^T\mathbf{F} = \mathbf{o}^T$. This $\mathbf{o}^T$ is the linear functional that annihilates Domain(**F**) . The last equation says that when ||**Fu** – g|| is minimized the residual **Fu** – g must be *normal* (perpendicular, orthogonal) to Range(**F**) . (This explains the word "Normal" in "Normal Equations" and removes any suggestion that other equations are abnormal.) Drawing a picture helps; imagine Range(**F**) to be a plane in Euclidean 3-space containing a vector **Fu** which, when it comes closest to a given vector g not in the plane, comes to that point in the plane reached by dropping a perpendicular from g .

We could transpose " $(\mathbf{Fu} – g)^T\mathbf{F} = \mathbf{o}^T$ " to " $(\mathbf{F}^T\mathbf{F})\mathbf{u} = \mathbf{F}^Tg$ " if we knew what " $\mathbf{F}^T\mathbf{F}$ " meant.

The trouble with the expression " $\mathbf{F}^T\mathbf{F}$ " is that it is not what it first seems; if $\mathbf{F}$ were a matrix then $\mathbf{F}^T\mathbf{F}$ would map Domain($\mathbf{F}$) to itself, but a change of basis in Domain($\mathbf{F}$) does not change $\mathbf{F}^T\mathbf{F}$ to the expected *similar* matrix. Here is what happens instead:

Let $\mathbf{B}$ be a basis for Domain($\mathbf{F}$) . Then abstract vector $\mathbf{u} = \mathbf{B}u$ for some column vector u , and $\mathbf{Fu} = \mathbf{FB}u = Fu$ for a matrix $F = \mathbf{FB}$. The Normal Equations " $(\mathbf{Fu} - g)^T\mathbf{F} = \mathbf{o}^T$ " turn into " $(Fu - g)^TF = o^T$ " which becomes " $(F^TF)u = F^Tg$ " after matrix transposition. $\mathbf{B}C$ is a new basis for Domain($\mathbf{F}$) , and $\mathbf{u} = \mathbf{B}C\mathbb{u}$ for $\mathbb{u} = C^{-1}u$, and $\mathbf{Fu} = \mathbb{F}\mathbb{u}$ for matrix $\mathbb{F} = FC$, where C is any invertible matrix of the same dimension as Domain($\mathbf{F}$) . What was " $(F^TF)u = F^Tg$ " in the old basis becomes " $(\mathbb{F}^T\mathbb{F})\mathbb{u} = \mathbb{F}^Tg$ " in the new, replacing matrix F^TF by $\mathbb{F}^T\mathbb{F} = C^TF^TFC$. This differs from $C^{-1}F^TFC$, which is how the change in basis would have changed F^TF if it were the matrix of a map from Domain($\mathbf{F}$) to itself. Instead, F^TF is the matrix of a map from Domain($\mathbf{F}$) to its own dual space.

If you doubt that these choices of basis matter, try the following example: Let $g := 10101$, a scalar, and suppose $F = [1, 10, 100]$ in some coordinate system. Then get Matlab to compute `u = F\g` to solve the least-squares problem. Next change to a new basis using a diagonal matrix $C = \text{diag}([10, 1, 1/16])$. It changes F to $\mathbb{F} = FC$ and thus changes the solution of the least-squares problem to $\mathbb{u} = \mathbb{F}\backslash g$. This maps back to $C\mathbb{u}$ = `C*((F*C)\g)` in the old basis. Compare with the old solution `u` . Try again with 6-vectors g and 6-by-3 matrices F at random.

Bilinear Forms:

There is no uniquely defined operator $\mathbf{F}^T\mathbf{F}$ just as there is no functional $\mathbf{u}^T$ determined uniquely by vector $\mathbf{u}$ in a non-Euclidean space. The matrices that appear in the Normal Equations are not all matrices that represent linear maps from one space of column vectors to another or itself; matrix F^TF belongs to a *Symmetric Bilinear Form* that maps column vectors to row vectors.

Consider $(\mathbf{Fu})^T\mathbf{Fv}$. It maps pairs $\{\mathbf{u}, \mathbf{v}\}$ of vectors from Domain($\mathbf{F}$) to real scalars, and does so as a linear function of each vector separately; this is the definition of a *Bilinear Form*. And since $(\mathbf{Fu})^T\mathbf{Fv}$ is unaltered when $\mathbf{u}$ and $\mathbf{v}$ are swapped, it is a *Symmetric Bilinear Form*.

There are many notations for bilinear forms: $\mathbf{Huv}$, $H(\mathbf{u}, \mathbf{v})$, $(\mathbf{v}, \mathrm{H}\mathbf{u})$, … . They all mean this:

- **Hu_** is a linear functional in the space dual to vectors $\mathbf{v}$, and $\mathbf{Huv}$ is its scalar value;
- **H_v** is a linear functional in the space dual to vectors $\mathbf{u}$, and $\mathbf{Huv}$ is its scalar value;
- Given a basis $\mathbf{B}$ for vectors $\mathbf{u} = \mathbf{B}u$, and a basis $\mathbf{E}$ for vectors $\mathbf{v} = \mathbf{E}v$, there is a matrix H for which $\mathbf{Huv} = (\mathbf{HB}u)\mathbf{E}v = (Hu)^Tv = v^THu$;
- Changing bases from $\mathbf{B}$ to $\mathbf{B}C$ and $\mathbf{E}$ to $\mathbf{E}D$ changes u to $\mathbb{u} = C^{-1}u$, v to $\mathbb{v} = D^{-1}v$, and H to $\mathbb{H} = D^THC$ so that $\mathbf{Huv} = v^THu = \mathbb{v}^T\mathbb{H}\mathbb{u}$.

Exercise: Express the elements of matrix H in terms of the effect $\mathbf{H}$ has upon the elements of bases $\mathbf{B}$ and $\mathbf{E}$.

A *Symmetric* bilinear form maps vectors $\mathbf{u}$ and $\mathbf{v}$ from the same space to scalars, and does so in a way independent of the order of $\mathbf{u}$ and $\mathbf{v}$ thus: $\mathbf{Huv} = \mathbf{Hvu}$. A symmetric bilinear form has a symmetric matrix $H = H^T$ in any basis. (Why?) Changing the basis changes H to matrix $\mathbb{H} = C^THC$ for some invertible C ; the two matrices $\mathbb{H}$ and H are called " Congruent." This congruence is an *Equivalence*, so it preserves rank; *i.e.*, $\text{rank}(\mathbb{H}) = \text{rank}(H)$. Congruence also preserves a thing called "Signature" as we'll see when we come to Sylvester's *Inertia* Theorem.