Least Squares and matrices

Jonathan L.F. King
University of Florida, Gainesville FL 32611-2082, USA
squash@ufl.edu
Webpage http://squash.1gainesville.com/

The Problem

Suppose we have a collection $\mathcal{K}$ of N points $Q_1, \dots, Q_j, \dots, Q_N$ in the plane $\mathbb{R}\times\mathbb{R}$. Consider now the line $\mathbb{L}$ with equation $y = \beta x + \alpha$. It has slope β and y-intercept α. At a given point $Q = (x, y)$, the vertical (signed) distance to $\mathbb{L}$ is

1: $$v \;:=\; [\alpha + \beta x] - y\,.$$

Letting v_j denote the vertical distance at Q_j, define the ***least-square distance*** from $\mathcal{K}$ to $\mathbb{L}$ by

1′: $$g(\alpha, \beta) \;:=\; \sum_{j=1}^{N} [v_j]^2\,.$$

Our goal is to find all pairs (α, β) which minimize g. It will turn out there is a *unique* minimum, *except* in the silly case that all the given points lie on one vertical line. That is, writing Q_j as (x_j, y_j), except when $x_1 = \dots = x_N$.

The quantities that we will need are

$$X := \sum_{j=1}^{N} x_j\,,\quad Y := \sum_{j=1}^{N} y_j\,,$$
$$S := \sum_{j=1}^{N} {x_j}^2\,,\quad P := \sum_{j=1}^{N} y_j x_j\,.$$

("S" is for Squares and "P" is for Product.)

Using *Calculus*

Evidently in computing the first-partials of g we will want to compute them for each v_j. From (1) we compute that

$$\frac{dv}{d\alpha} = 1\,,\quad\text{so}\quad \frac{d}{d\alpha}(v^2) = 2v\cdot 1 \quad\text{and}$$
$$\frac{dv}{d\beta} = x\,,\quad\text{so}\quad \frac{d}{d\beta}(v^2) = 2v\cdot x\,,$$

by the Chain Rule. Consequently

$$\frac{dg}{d\alpha} = \sum_{j=1}^{N} 2v_j \quad\text{and}\quad \frac{dg}{d\beta} = \sum_{j=1}^{N} 2v_j x_j\,.$$

Thus, the pair (α, β) is a critical point of g IFF at (α, β) we have that

2: $$0 = \sum_{j=1}^{N} v_j \quad\text{and}\quad 0 = \sum_{j=1}^{N} v_j x_j\,.$$

Recall that v_j is $\alpha + x_j\beta - y_j$. So multiplying out and distributing the summations in (2) yields that

2′: $$0 = N\alpha + X\beta - Y\,,\quad 0 = X\alpha + S\beta - P\,.$$

We can rewrite this to say that (α, β) is a critical point of g IFF

3: $$Y = N\alpha + X\beta\,,$$
$$P = X\alpha + S\beta\,.$$

Matrices. Let M denote the matrix $\left[\begin{smallmatrix} N & X \\ X & S\end{smallmatrix}\right]$ and let

$$D := \mathrm{Det}(\mathsf{M}) \stackrel{\text{note}}{=\!=} NS - X^2\,.$$

It follows from a standard[♡1♡2] inequality that: *All the points $x_1, \dots, x_N$ are equal IFF $D = 0$.* We henceforth assume that our scatterplot has at least two distinct x-values.

Bare-hands computation [or matrix algebra] shows that (3) has a unique solution, which is

4: $$\alpha = \tfrac{1}{D}\Big[\;SY - XP\Big]\,,$$
$$\beta = \tfrac{1}{D}\Big[-XY + NP\Big]\,.$$

I.e, $$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \frac{1}{D}\begin{bmatrix} S & -X \\ -X & N\end{bmatrix}\cdot\begin{bmatrix}Y\\ P\end{bmatrix}.$$

[♡1] Jensen's Inequality implies that D is positive. For *that* assertion is equivalent to "$D/[N^2] > 0$", i.e, to

$$\frac{1}{N}\sum_{j=1}^{N}[x_j]^2 \;>\; \Big[\frac{1}{N}\sum_{j=1}^{N} x_j\Big]^2.$$

This has form $\frac{1}{N}\sum_1^N f(x_j) > f\big(\frac{1}{N}\sum_1^N x_j\big)$, where f is the squaring-map. Since f is strictly convex-up, Jensen's yields "$\geq$", with equality IFF $x_1 = \ldots = x_N$.

[♡2] We can use the Cauchy-Schwarz inequality [CS] with inner-product $\big\langle (p_1, \dots, p_N), (q_1, \dots, q_N)\big\rangle := \sum_1^N \overline{p_j}\cdot q_j$. For let $\mathbf{1} := (1, \overset{N}{\dots}, 1)$ and $\mathbf{w} := (x_1, \dots, x_N)$. CS gives

$$\big|\langle \mathbf{1}, \mathbf{w}\rangle\big|^2 \;\leq\; \langle \mathbf{1}, \mathbf{1}\rangle\cdot\langle \mathbf{w}, \mathbf{w}\rangle\,,$$

i.e, $X^2 \leq N{\cdot}S$. There is equality IFF $\mathbf{w}$ is a multiple of $\mathbf{1}$, i.e IFF all the x_j equal a common value.

Neat!

(Exer. E1: Let $Q_j := (j, j^2)$. For $N = 2,3,4,5$, find the best approximating line to scatterplot $Q_1, \ldots, Q_N$. How do the slopes of the lines change as you increase N? Taking two of the geometric points in the list, what happens to the fitting-line if you *repeat* each of them several times to make a new list?)

Using *Linear Algebra*

In matrix notation we can write (4) as

$$4': \qquad \begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \mathsf{M}^{-1} \cdot \begin{bmatrix}Y\\ P\end{bmatrix},$$

suggesting that *least-squares* secretly contains linear algebra. We set the stage for a more general problem, then apply it to *least-squares*.

With F either $\mathbb{R}$ or $\mathbb{C}$, consider an N-dim'al F–inner-product space $(\mathbf{H}, \langle\cdot,\cdot\rangle)$, a K-dim'al subspace $\mathbf{W} \subset \mathbf{H}$, and its ***ortho-complement***

$$\mathbf{W}^\perp := \left\{\mathbf{g} \in \mathbf{H} \,\middle|\, \forall \mathbf{w} \in \mathbf{W}\colon \mathbf{g} \perp \mathbf{w}\right\}.$$

The ***orthogonal projection*** operator is the map $\operatorname{Proj}: \mathbf{H} \to \mathbf{W}$ satisfying, for each $Q \in \mathbf{H}$, that

$$Q - \operatorname{Proj}(Q) \in \mathbf{W}^\perp.$$

Point $P := \operatorname{Proj}(Q)$ is the (unique) *closest-point* on $\mathbf{W}$ to Q; it minimizes $\langle \mathbf{w} - Q, \mathbf{w} - Q\rangle$ as $\mathbf{w}$ ranges over $\mathbf{W}$.

Subspaces. One way to get a subspace is as the range of a linear map $\mathsf{A}: \mathsf{F}^K \to \mathbf{H}$; so let

$$5: \qquad \mathbf{W} := \operatorname{Range}(\mathsf{A}) \overset{\text{note}}{=\!=} \{\mathsf{AU} \mid \mathsf{U} \in \mathsf{F}^K\}.$$

Consider a point $P \in \mathbf{W}$. Then

6: *There is a unique* $\mathsf{U}_0 \in \mathsf{F}^K$ *with* $\mathsf{AU}_0 = P$
IFF $\mathsf{U} \mapsto \mathsf{AU}$ *is* 1*-to-*1*, i.e,* $\operatorname{Rank}(\mathsf{A}) = K$.

We want to state this rank-condition in terms of an adjoint operator, so equip F^K with the [conjugate] dot-product.♡3 Thus we have a well-defined ***adjoint map*** $\mathsf{A}^*: \mathbf{H} \to \mathsf{F}^K$, defined by

♡3 Actually, any inner-product on F^K works in (8), but note that changing the IP will change what "A^*" means.

$$7: \qquad \forall \mathbf{g} \in \mathbf{H} \text{ and } \forall \mathsf{U} \in \mathsf{F}^K\colon \langle \mathsf{A}^*\mathbf{g}, \mathsf{U}\rangle = \langle \mathbf{g}, \mathsf{AU}\rangle.$$

(Exer. E2: Show that $[\mathsf{A}^*]^* = \mathsf{A}$.) Hence we have linear maps $\mathsf{A}^*\mathsf{A}: \mathsf{F}^K \to \mathsf{F}^K$ and $\mathsf{AA}^*: \mathbf{H} \to \mathbf{H}$. A standard result (Exer. E3) is that $\operatorname{Ker}(\mathsf{A}^*\mathsf{A}) = \operatorname{Ker}(\mathsf{A})$. A corollary of this is that $\operatorname{Rank}(\mathsf{A}^*\mathsf{A}) = \operatorname{Rank}(\mathsf{A})$, since our VSes are finite dimensional. [We used the Rank+Nullity thm.]

Thus, we can restate the above as

6′: *There is a unique* $\mathsf{U}_0 \in \mathsf{F}^K$ *with* $\mathsf{AU}_0 = P$
IFF $\mathsf{U} \mapsto \mathsf{AU}$ *is* 1*-to-*1*, i.e,* $\operatorname{Rank}(\mathsf{A}) = K$.
IFF $\mathsf{A}^*\mathsf{A}$ *is invertible.*

The Problem. Fix a rank-K linear-map $\mathsf{A}: \mathsf{F}^K \to \mathbf{H}$ and a point $Q \in \mathbf{H}$. We seek a formula for the unique point $\mathsf{U}_0 \in \mathsf{F}^K$ so that $\|\mathsf{AU}_0 - Q\|$ is the minimum of $\|\mathsf{AU} - Q\|$ taken over all $\mathsf{U} \in \mathsf{F}^K$.

The difference-vector $\mathsf{AU}_0 - Q$ is orthogonal to every vector in (5). I.e, for each $\mathsf{U} \in \mathsf{F}^K$, inner product $\left\langle \mathsf{AU}_0 - Q, \mathsf{AU}\right\rangle$ is zero. By (7), then,

$$\left\langle \mathsf{A}^*\mathsf{AU}_0 - \mathsf{A}^*Q, \mathsf{U}\right\rangle = 0.$$

But the only vector orthogonal to *all* $\mathsf{U} \in \mathsf{F}^K$ is $\vec{0} \in \mathsf{F}^K$. Thus U_0 satisfies $\mathsf{A}^*\mathsf{AU}_0 = \mathsf{A}^*Q$. Hence

$$8: \qquad \mathsf{U}_0 = [\mathsf{A}^*\mathsf{A}]^{-1}\mathsf{A}^*Q,$$

courtesy (6′).

Least squares. We can apply this to our line-fitting of (1). After all, RhS(1′) is the square of the dot-product norm on $\mathbf{H} := \mathsf{F}^N$. We are minimizing the square-norm of column vector $[v_1, \ldots, v_N]^{\mathsf{t}}$. Our unknown vector is $\mathsf{U} = \begin{bmatrix}\alpha\\ \beta\end{bmatrix}$; so $K = 2$. With

$$9: \qquad \mathsf{A} := \begin{bmatrix}1 & x_1\\ 1 & x_2\\ \vdots & \vdots\\ 1 & x_N\end{bmatrix} \quad \textit{and} \quad \mathsf{Q} := \begin{bmatrix}y_1\\ y_2\\ \vdots\\ y_N\end{bmatrix},$$

we are minimizing the norm of $\mathsf{AU} - \mathsf{Q}$ over all U.

Applying (8), the minimum occurs at

$$8': \qquad \begin{bmatrix}\alpha\\ \beta\end{bmatrix} = [\mathsf{A}^*\mathsf{A}]^{-1}\mathsf{A}^*\mathsf{Q}$$

Of course, RhS(8′) must equal RhS(4′). Indeed we find that $\mathsf{A}^*\mathsf{A} = \mathsf{M}$ and $\mathsf{A}^*\mathsf{Q} = \begin{bmatrix}Y\\ P\end{bmatrix}$.

Finally, note that Rank(A) equals K, i.e, equals 2, exactly when not all $x_1, \ldots, x_N$ are equal. This was precisely the "non-silly" condition we needed for the *Calculus* approach.

Fitting to a polynomial. To our N many data-points $Q_j = (x_j, y_j)$ in the F×F plane, we wish to least-squares fit the closest K-topped (i.e, Deg $< K$) polynomial

10: $$\alpha_0 + \alpha_1 x + \alpha_2 x^2 + \ldots + \alpha_{K-1} x^{K-1}.$$

Copying what we did in (9), define our "unknown" col-vector $\mathsf{U} := [\alpha_0 \ \ \alpha_1 \ \ \cdots \ \ \alpha_{K-1}]^{\mathsf{t}}$, as well as

9′: $$\mathsf{A}_K := \begin{bmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^{K-1} \\ 1 & x_2 & x_2^2 & \ldots & x_2^{K-1} \\ & & \vdots & & \\ 1 & x_N & x_N^2 & \ldots & x_N^{K-1} \end{bmatrix}, \ \mathsf{Q} := \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}.$$

When Rank(A_K) equals K, then (8) applies, telling us that the closest-fit polynomial (10) has coefficients $\mathsf{U} = [\mathsf{A}_K^* \mathsf{A}_K]^{-1} \mathsf{A}_K^* \mathsf{Q}$.

Vandermonde matrices. The $N{\times}K$ matrix A_K of (9′) is called a ***Vandermonde matrix***.[♡4] When N and K equal a common value, L, then –it turns out–

11: $$\text{Det}(\mathsf{A}_L) = \prod_{\substack{j,i \in [1..L] \\ \text{with } j>i}} [x_j - x_i].$$

Returning to the general $N{\times}K$ case, let L denote the number of *distinct* values in $\{x_1, \ldots, x_N\}$, and suppose that $\boxed{K \geq L}$. Remove the duplicate rows, then only keep the first L many columns. We have thus produced an $L{\times}L$ Vandermonde matrix *inside* our original $N{\times}K$ matrix, and (11) implies that this $L{\times}L$ has non-zero determinant. We have thus proven:

11′: *Fix an arbitrary field* F, *points* $x_1, \ldots, x_N \in$ F, *and let L be the number of* distinct *points in this list. Then, for each $K \geq L$, the Vandermonde matrix* $\mathsf{A}_K(x_1, x_2, \ldots, x_N)$ *has rank equaling L, the cardinality of set* $\{x_1, x_2, \ldots, x_N\}$.

[♡4] The Vandermonde-matrix Wikipedia article is nice.

So we get a *unique* K-topped polynomial least-squares–closest to our N many data-points *exactly when* there are at least K distinct x-values among the points.

Lagrange polynomials. Suppose points $x_1, \ldots, x_N$ are distinct. If K equals N, then *Lagrange Interpolation* tells us there is a *unique* K-topped polynomial whose graph passes through each of $Q_1, \ldots, Q_j, \ldots Q_N$; the least-squares distance is zero.

When $K{>}N$, then there is a *family* of K-topped polynomials (a $[K{-}N]$-dim'al family) which pass through the data-points; so no uniqueness in the least-squares fit.

Fitting to a family of functions. Fix an arbitrary set **S**, functions $f_0, f_1, \ldots, f_{K-1}$:**S**→F, and let $\mathcal{G}$ be the set of linear combinations $\sum_{j=0}^{K-1} c_j \cdot f_j()$.

A ***scatterplot*** is a multiset $\{Q_j\}_{j=1}^N$ of points

$$Q_j = (\mathbf{s}_j, \tau_j) \in \mathbf{S} \times \mathsf{F}.$$

Points $\{\mathbf{s}_j\}_{j=1}^N \subset \mathbf{S}$ are the ***sample points***, and $\{\tau_j\}_{j=1}^N$ are the ***target values***. [Previously we used "x_j" for a sample point, and "y_j" for a target value.]

We can use the preceding technique to find a function $g \in \mathcal{G}$ which minimizes the least-square distance to scatterplot $\{Q_j\}_{j=1}^N$ Namely, define this $N{\times}K$ matrix and column-vector

12: $$\mathsf{A} := \begin{bmatrix} f_0(\mathbf{s}_1) & f_1(\mathbf{s}_1) & \ldots & f_{K-1}(\mathbf{s}_1) \\ f_0(\mathbf{s}_2) & f_1(\mathbf{s}_2) & \ldots & f_{K-1}(\mathbf{s}_2) \\ \vdots & \vdots & & \\ f_0(\mathbf{s}_N) & f_1(\mathbf{s}_N) & \ldots & f_{K-1}(\mathbf{s}_N) \end{bmatrix}, \mathsf{Q} := \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_N \end{bmatrix}.$$

When A has rank K, then

8″: $$\begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{K-1} \end{bmatrix} := [\mathsf{A}^* \mathsf{A}]^{-1} \mathsf{A}^* \mathsf{Q}$$

is the coeff-vector giving this closest fnc $g()$.

13: *Appl: 1-variable polynomials.* The setup in (9′) is a special case of (12), by setting

$$\mathbf{S} := \mathsf{F} \ \text{ and } \ f_j := [x \mapsto x^j]. \qquad \square$$

14: *Appl: Closest plane.* Suppose now you want to find the plane

$$\big(x, y\big) \;\mapsto\; a + bx + cy$$

least-square closest to $\{Q_j\}_1^N$, where $\mathbf{s}_j = \big(x_j, y_j\big)$, a point in $\mathsf{F}{\times}\mathsf{F}$. So apply (12) and (8″), where

> $\mathbf{S} := \mathsf{F}{\times}\mathsf{F}$ *and functions* f_0, f_1, f_2 *send* $\mathbf{s} := \big(x, y\big)$ *to, respectively:* 1, x, y.

Then $\begin{bmatrix} a \\ b \\ c \end{bmatrix} = [\mathsf{A}^{*}\mathsf{A}]^{-1}\mathsf{A}^{*}\mathsf{Q}$. □

Filename: `Problems/Analysis/Calculus/least_squares.latex`
As of: *Monday 06Mar2006.* Typeset: *27Nov2023 at 21:15.*