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 $K$ of $N$ points $Q_1, \ldots, Q_j, \ldots, Q_N$ in the plane.\textsuperscript{1} Consider now the line $L$ with equation $y = \beta x + \alpha$. It has slope $\beta$ and y-intercept $\alpha$. At a given point $Q = (x,y)$, the vertical (signed) distance to $L$ is

\begin{equation}
\label{eq:distance}
v := [\alpha + \beta x] - y.
\end{equation}

Letting $v_j$ denote the vertical distance at $Q_j$, define the least-square distance from $K$ to $L$ by

\begin{equation}
\label{eq:least-square}
g(\alpha, \beta) := \sum_{j=1}^{N} [v_j]^2.
\end{equation}

Our goal is to find all pairs $(\alpha, \beta)$ 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 $x_1 = \cdots = x_N$.

The quantities that we will need are

\begin{align*}
X &:= \sum_{j=1}^{N} x_j, & Y &:= \sum_{j=1}^{N} y_j, \\
S &:= \sum_{j=1}^{N} x_j^2, & P &:= \sum_{j=1}^{N} y_j x_j.
\end{align*}

("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

\begin{align*}
\frac{dv}{d\alpha} &= 1, & \frac{dv}{d\beta} &= 2v \cdot 1 \quad \text{and} \\
\frac{dv}{d\alpha} &= \beta, & \frac{dv}{d\beta} &= 2v \cdot x.
\end{align*}

by the Chain Rule. Consequently

\begin{equation}
\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.
\end{equation}

Thus, the pair $(\alpha, \beta)$ is a critical point of $g$ IFF at $(\alpha, \beta)$ we have that

\begin{align*}
2: & \quad 0 = \sum_{j=1}^{N} v_j \quad \text{and} \quad 0 = \sum_{j=1}^{N} v_j x_j.
\end{align*}

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

\begin{align*}
2': & \quad 0 = N \alpha + X \beta - Y, \quad 0 = X \alpha + S \beta - P.
\end{align*}

We can rewrite this to say that $(\alpha, \beta)$ is a critical point of $g$ IFF

\begin{align*}
3: & \quad Y = N \alpha + X \beta, \\
P = X \alpha + S \beta.
\end{align*}

Matrices. Let $M$ denote the matrix $[X \ Y \ S]$ and let $D := \text{Det}(M)$ note $NS - X^2$.

It follows from a standard\textsuperscript{2,3} inequality that: All the points $x_1, \ldots, x_N$ 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

\begin{align*}
\alpha &= \frac{1}{D}[SY - XP], \\
\beta &= \frac{1}{D}[-XY + NP].
\end{align*}

\textsuperscript{2}Jensen’s Inequality implies that $D$ is positive. For that assertion is equivalent to “$D/[N^2] > 0$”, i.e, to

\begin{equation}
\frac{1}{N} \sum_{j=1}^{N} [x_j]^2 > \left[ \frac{1}{N} \sum_{j=1}^{N} x_j \right]^2.
\end{equation}

This has form $\frac{1}{N} \sum_{i=1}^{N} f(x_i) > f \left( \frac{1}{N} \sum_{i=1}^{N} x_j \right)$, where $f$ is the squaring-map. Since $f$ is strictly convex-up, Jensen’s yields “$\geq$”, with equality IFF $x_1 = \cdots = x_N$.

\textsuperscript{3}We use the Cauchy-Schwarz inequality [CS] with inner-product $\langle (p_1, \ldots, p_N), (q_1, \ldots, q_N) \rangle := \sum_{j=1}^{N} p_j \cdot q_j$. For let $1 := (1, \ldots, 1)$ and $w := (x_1, \ldots, x_N)$. CS gives

\begin{equation}
\langle (1, w), (1, w) \rangle \leq \langle (1, 1), (w, w) \rangle.
\end{equation}

i.e, $X^2 \leq NS$. There is equality IFF $w$ is a multiple of $1$, 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

\[
\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \mathbf{M}^{-1} \cdot \begin{bmatrix} y \\ p \end{bmatrix},
\]

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

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

\[
\mathbf{W}^\perp := \{ \mathbf{g} \in \mathbf{H} \mid \forall \mathbf{w} \in \mathbf{W}; \mathbf{g} \perp \mathbf{w} \}.
\]

The *orthogonal projection* operator is the map \( \text{Proj}: \mathbf{H} \rightarrow \mathbf{W} \) satisfying, for each \( Q \in \mathbf{H} \), that

\[
Q - \text{Proj}(Q) \in \mathbf{W}^\perp.
\]

Point \( P := \text{Proj}(Q) \) 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 \( \mathbf{A}: \mathbf{F}^K \rightarrow \mathbf{H} \); so let

\[
\mathbf{W} := \text{Range}(\mathbf{A}) \overset{\text{note}}{=} \{ \mathbf{U} \mid \mathbf{U} \in \mathbf{F}^K \}.
\]

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

\[
\text{There is a unique } \mathbf{U}_0 \in \mathbf{F}^K \text{ with } \mathbf{A}\mathbf{U}_0 = P \quad \text{iff} \quad \mathbf{U} \mapsto \mathbf{A}\mathbf{U} \text{ is 1-to-1, i.e, } \text{Rank}(\mathbf{A}) = K.
\]

We want to state this rank-condition in terms of an adjoint operator, so equip \( \mathbf{F}^K \) with the *conjugate* dot-product.\(^\text{V4}\) Thus we have a well-defined *adjoint map* \( \mathbf{A}^* : \mathbf{H} \rightarrow \mathbf{F}^K \), defined by

\[
\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = [\mathbf{A}^*]^\dagger [\mathbf{A}^*]^{-1} [\mathbf{A}^*] \mathbf{Q},
\]

\(^\text{V4}\) Actually, any inner-product on \( \mathbf{F}^K \) works in (8), but note that changing the IP will change what “\( \mathbf{A}^* \)” means.

(Exer. E2: Show that \( \mathbf{A}^* \mathbf{A} = \mathbf{A} \).) Hence we have linear maps \( \mathbf{A} : \mathbf{F}^K \rightarrow \mathbf{F}^K \) and \( \mathbf{A} : \mathbf{H} \rightarrow \mathbf{H} \). A standard result (Exer. E3) is that \( \text{Ker}(\mathbf{A}^* \mathbf{A}) = \text{Ker}(\mathbf{A}) \).

A corollary of this is that \( \text{Rank}(\mathbf{A}^* \mathbf{A}) = \text{Rank}(\mathbf{A}) \).

So we can restate the above as

\[
\text{There is a unique } \mathbf{U}_0 \in \mathbf{F}^K \text{ with } \mathbf{A}\mathbf{U}_0 = P \quad \text{iff} \quad \mathbf{U} \mapsto \mathbf{A}\mathbf{U} \text{ is 1-to-1, i.e, } \text{Rank}(\mathbf{A}) = K.
\]

(Exer. E3: \( \mathbf{A} \) is invertible.)

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

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

\[
\langle \mathbf{A}^* \mathbf{A}\mathbf{U}_0 - \mathbf{A}^*Q, \mathbf{U} \rangle = 0.
\]

But the only vector orthogonal to all \( \mathbf{U} \in \mathbf{F}^K \) is \( \mathbf{0} \in \mathbf{F}^K \). Thus \( \mathbf{U}_0 \) satisfies \( \mathbf{A}^* \mathbf{A}\mathbf{U}_0 = \mathbf{A}^*Q \). Hence

\[
\mathbf{U}_0 = [\mathbf{A}^*]\dagger [\mathbf{A}^*]^{-1} \mathbf{A}^*Q,
\]

courtesy (??’).

**Least squares.** We can apply this to our line-fitting of (1). After all, \( \text{Rhs}(??’) \) is the square of the dot-product norm on \( \mathbf{H} := \mathbf{F}^N \). We are minimizing the square-norm of column vector \( [v_1, \ldots, v_N]^T \). Our unknown vector is \( \mathbf{U} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \); so \( K = 2 \). With

\[
\mathbf{A} := \begin{bmatrix}
1 & x_1 \\
1 & x_2 \\
\vdots & \vdots \\
1 & x_N
\end{bmatrix} \quad \text{and} \quad \mathbf{Q} := \begin{bmatrix} y_1 \\ y_2 \\
\vdots \\
y_N\end{bmatrix},
\]

we are minimizing the norm of \( \mathbf{A}\mathbf{U} - \mathbf{Q} \) over all \( \mathbf{U} \).

Applying (8), the minimum occurs at

\[
\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = [\mathbf{A}^*]\dagger [\mathbf{A}^*]^{-1} \mathbf{A}^* \mathbf{Q}.
\]

Of course, \( \text{Rhs}(??’) \) must equal \( \text{Rhs}(??’) \). Indeed we find that \( \text{Rank}(\mathbf{A}) = K \), 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 $\mathbf{F} \times \mathbf{F}$ plane, we wish to least-squares fit the closest $K$-topped (i.e., $\text{Deg } < K$) polynomial

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

Copying what we did in (9), define our “unknown” col-vector $\mathbf{U} := [\alpha_0 \; \alpha_1 \; \ldots \; \alpha_{K-1}]^\top$, as well as

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

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

Vandermonde matrices. The $N \times K$ matrix $\mathbf{A}_K$ of (??) is called a Vandermonde matrix.\(^{\text{C5}}\) When $N$ and $K$ equal a common value, $L$, then –it turns out–

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

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 $\lfloor K \geq L \rfloor$. Remove the duplicate rows, then only keep the first $L$ many columns. We have thus produced an $L \times L$ Vandermonde matrix \textit{inside} our original $N \times K$ matrix, and (11) implies that this $L \times L$ has non-zero determinant. We have thus proven:

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

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.

\(^{\text{C5}}\)The Vandermonde-matrix Wikipedia article is nice.

\textbf{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 $\mathbf{S}$, functions $f_0, f_1, \ldots, f_{K-1}: \mathbf{S} \rightarrow \mathbf{F}$, and let $\mathcal{G}$ be the set of linear combinations $\sum_{j=0}^{K-1} c_j f_j()$.

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

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

Points $\{s_j\}_{j=1}^N \subset \mathbf{S}$ 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

$$12: \quad \mathbf{A} := \begin{bmatrix} f_0(s_1) & f_1(s_1) & \ldots & f_{K-1}(s_1) \\ f_0(s_2) & f_1(s_2) & \ldots & f_{K-1}(s_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_0(s_N) & f_1(s_N) & \ldots & f_{K-1}(s_N) \end{bmatrix}, \quad \mathbf{Q} := \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_N \end{bmatrix}.$$ 

When $\mathbf{A}$ has rank $K$, then

$$8'': \quad \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{K-1} \end{bmatrix} := [\mathbf{A}^\top \mathbf{A}]^{-1} \mathbf{A}^\top \mathbf{Q}$$ 

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

13: \textbf{Apply: 1-variable polynomials.} The setup in (??) is a special case of (12), by setting

$$\mathbf{S} := \mathbf{F} \text{ and } f_j := [x \mapsto x^j]. \quad \square$$
14: **Appl: Closest plane.** Suppose now you want to find the plane

\[(x, y) \mapsto a + bx + cy\]

least-square closest to \(\{Q_j\}_1^N\), where \(s_j = (x_j, y_j)\), a point in \(F \times F\). So apply (12) and (??′′), where

\[S := F \times F \text{ and functions } f_0, f_1, f_2 \text{ send } s := (x, y) \text{ to, respectively: } 1, x, y.\]

Then

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

\[
\square
\]