

## APPENDIX TO PART II

This Appendix, designated as **A2**, contains additional analytical results for Part II of the NOTEBOOK, and follows the notational conventions in Appendix A1.

### A2.1. Covariograms for Sums of Independent Spatial Processes

First recall that the covariance of any random variables,  $Z_1$  and  $Z_2$ , with respective means,  $\mu_1$  and  $\mu_2$ , is given by

$$\begin{aligned}
 \text{(A2.1.1)} \quad \text{cov}(Z_1, Z_2) &= E[(Z_1 - \mu_1)(Z_2 - \mu_2)] = E(Z_1 Z_2 - Z_1 \mu_2 - \mu_1 Z_2 + \mu_1 \mu_2) \\
 &= E(Z_1 Z_2) - E(Z_1) \mu_2 - \mu_1 E(Z_2) + \mu_1 \mu_2 \\
 &= E(Z_1 Z_2) - \mu_1 \mu_2 - \mu_1 \mu_2 + \mu_1 \mu_2 \\
 &= E(Z_1 Z_2) - \mu_1 \mu_2
 \end{aligned}$$

so that if  $Z_1$  and  $Z_2$  are *independent* then

$$\text{(A2.1.2)} \quad E(Z_1 Z_2) = E(Z_1)E(Z_2) = \mu_1 \mu_2 \Rightarrow \text{cov}(Z_1, Z_2) = 0$$

Hence if a given covariance stationary stochastic process,  $\{Y(s) : s \in R\}$ , with mean,  $\mu$ , is the sum of two *independent* covariance stationary components

$$\text{(A2.1.3)} \quad Y(s) = Y_1(s) + Y_2(s), \quad s \in R,$$

with respective means,  $\mu_1$  and  $\mu_2$ , then it follows by definition that  $\mu = \mu_1 + \mu_2$ , and that  $Y_1(s)$  and  $Y_2(v)$  are independent for all  $s, v \in R$ . Hence for any  $h \geq 0$  and  $s, v \in R$  with  $\|s - v\| = h$ , we see that the *covariogram*,  $C$ , of the  $Y$ -process must satisfy,

$$\begin{aligned}
 \text{(A2.1.1)} \quad C(h) &= \text{cov}[Y(s), Y(v)] \\
 &= E[Y(s) \cdot Y(v)] - E[Y(s)] \cdot E[Y(v)] = E[Y(s) \cdot Y(v)] - \mu^2 \\
 &= E\left[(Y_1(s) + Y_2(s))(Y_1(v) + Y_2(v))\right] - (\mu_1 + \mu_2)^2 \\
 &= E[Y_1(s)Y_1(v) + Y_1(s)Y_2(v) + Y_2(s)Y_1(v) + Y_2(s)Y_2(v)] \\
 &\quad - (\mu_1^2 + 2\mu_1\mu_2 + \mu_2^2)
 \end{aligned}$$

$$\begin{aligned}
&= E[Y_1(s)Y_1(v)] + E[Y_1(s)]E[Y_2(v)] + E[Y_2(s)]E[Y_1(v)] + E[Y_2(s)Y_2(v)] \\
&\quad - (\mu_1^2 - \mu_1\mu_2 - \mu_2\mu_1 + \mu_2^2) \\
&= E[Y_1(s)Y_1(v)] + \mu_1\mu_2 + \mu_2\mu_1 + E[Y_2(s)Y_2(v)] - (\mu_1^2 - \mu_1\mu_2 - \mu_2\mu_1 + \mu_2^2) \\
&= (E[Y_1(s)Y_1(v)] - \mu_1^2) + (E[Y_2(s)Y_2(v)] - \mu_2^2) \\
&= \text{cov}[Y_1(s), Y_1(v)] + \text{cov}[Y_2(s), Y_2(v)] \\
&= C_1(h) + C_2(h)
\end{aligned}$$

where  $C_1$  and  $C_2$  are the respective covariograms for the  $Y_1$  and  $Y_2$  components of  $Y$ .

## A2.2. Expectation of the Sample Covariance Estimator under Spatial Dependence

Given any collection of  $2n$  jointly distributed random variables,  $\{(Y_{1i}, Y_{2i}), i = 1, \dots, n\}$  where the pairs  $(Y_{1i}, Y_{2i})$  have common means  $E(Y_{1i}) = \mu_1$ ,  $E(Y_{2i}) = \mu_2$  and covariance  $\text{cov}(Y_{1i}, Y_{2i}) = \sigma_{12}$  for all  $i = 1, \dots, n$ , consider the following estimator of  $\sigma_{12}$ ,

$$(A2.2.1) \quad \hat{\sigma}_{12} = \frac{1}{n-1} \sum_{i=1}^n (Y_{1i} - \bar{Y}_1)(Y_{2i} - \bar{Y}_2)$$

where  $\bar{Y}_j = \frac{1}{n} \sum_{i=1}^n Y_{ji}$ ,  $j = 1, 2$ . Here  $\hat{\sigma}_{12}$  and  $\sigma_{12}$  are taken to correspond to the estimator  $\hat{C}(h)$  of the covariance  $C(h)$  in expressions (4.10.2) and (4.10.1), respectively. To analyze this estimator, it is convenient to begin with the rescaled version

$$(A2.2.2) \quad \tilde{\sigma}_{12} = \frac{n}{n-1} \hat{\sigma}_{12} = \frac{1}{n} \sum_{i=1}^n (Y_{1i} - \bar{Y}_1)(Y_{2i} - \bar{Y}_2)$$

and recall the following standard decomposition of sums of squares:

$$\begin{aligned}
(A2.2.3) \quad \tilde{\sigma}_{12} &= \frac{1}{n} \sum_{i=1}^n (Y_{1i}Y_{2i} - Y_{1i}\bar{Y}_2 - \bar{Y}_1Y_{2i} + \bar{Y}_1\bar{Y}_2) \\
&= \frac{1}{n} \sum_{i=1}^n Y_{1i}Y_{2i} - \left(\frac{1}{n} \sum_{i=1}^n Y_{1i}\right)\bar{Y}_2 - \bar{Y}_1\left(\frac{1}{n} \sum_{i=1}^n Y_{2i}\right) + n\left(\frac{1}{n}\bar{Y}_1\bar{Y}_2\right) \\
&= \frac{1}{n} \sum_{i=1}^n Y_{1i}Y_{2i} - \bar{Y}_1\bar{Y}_2 - \bar{Y}_1\bar{Y}_2 + \bar{Y}_1\bar{Y}_2 \\
&= \frac{1}{n} \sum_{i=1}^n Y_{1i}Y_{2i} - \bar{Y}_1\bar{Y}_2
\end{aligned}$$

But since

$$(A2.2.4) \quad \bar{Y}_1 \bar{Y}_2 = \left( \frac{1}{n} \sum_{i=1}^n Y_{1i} \right) \left( \frac{1}{n} \sum_{i=1}^n Y_{2i} \right) = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n Y_{1i} Y_{2j}$$

it follows from (A2.2.2) through (A2.2.4) that

$$(A2.2.5) \quad \begin{aligned} E(\hat{\sigma}_{12}) &= \frac{n}{n-1} E(\tilde{\sigma}_{12}) = \frac{n}{n-1} \left[ \frac{1}{n} \sum_{i=1}^n E(Y_{1i} Y_{2i}) - E(\bar{Y}_1 \bar{Y}_2) \right] \\ &= \frac{n}{n-1} \left[ \frac{1}{n} \sum_{i=1}^n E(Y_{1i} Y_{2i}) - \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n E(Y_{1i} Y_{2j}) \right] \\ &= \frac{n}{n-1} \left[ \frac{1}{n} \sum_{i=1}^n E(Y_{1i} Y_{2i}) - \frac{1}{n^2} \sum_{i=1}^n E(Y_{1i} Y_{2i}) - \frac{1}{n^2} \sum_{i=1}^n \sum_{j \neq i} E(Y_{1i} Y_{2j}) \right] \\ &= \frac{n}{n-1} \left[ \frac{n-1}{n^2} \sum_{i=1}^n E(Y_{1i} Y_{2i}) - \frac{1}{n^2} \sum_{i=1}^n \sum_{j \neq i} E(Y_{1i} Y_{2j}) \right] \\ &= \frac{1}{n} \sum_{i=1}^n E(Y_{1i} Y_{2i}) - \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j \neq i} E(Y_{1i} Y_{2j}) \end{aligned}$$

Finally, if we let  $\mu_j \equiv E(Y_{ji})$ ,  $j=1,2$ , then since by definition,  $E(Y_{1i} Y_{2i}) = \sigma_{12} + \mu_1 \mu_2$  and  $E(Y_{1i} Y_{2j}) = \text{cov}(Y_{1i}, Y_{2j}) + \mu_1 \mu_2$  both hold for all  $i=1, \dots, n$  and  $j \neq i$ , it follows from (A2.2.5) that

$$(A2.2.6) \quad \begin{aligned} E(\hat{\sigma}_{12}) &= \frac{n}{n-1} (\sigma_{12} + \mu_1 \mu_2) - \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n (\text{cov}(Y_{1i}, Y_{2j}) + \mu_1 \mu_2) \\ &= \sigma_{12} + \mu_1 \mu_2 - \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j \neq i} \text{cov}(Y_{1i}, Y_{2j}) - \frac{n(n-1)}{n(n-1)} \mu_1 \mu_2 \\ &= \sigma_{12} - \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j \neq i} \text{cov}(Y_{1i}, Y_{2j}) \end{aligned}$$

### A2.3. A Bound on the Binning Bias of Empirical Variogram Estimates

Here it suffices to consider the variogram,  $\gamma(h)$ , on the interval of distance values,  $d_{k-1} \leq h < d_k$ , for a typical bin  $k$ . Recall from (4.7.1) that for a given sample of values  $(Y(s_i) : i=1, \dots, n)$ , if  $N_k$  denotes the set of *distance pairs*,  $(s_i, s_j)$ , in bin  $k$ , and if the distance between each such pair is denoted by  $h_{ij} = \|s_i - s_j\|$ , then the *lag distance*,  $h_k$ , for bin  $k$  is defined to be

$$(A2.3.1) \quad h_k = \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} h_{ij}$$

Recall also that if the  $\varepsilon_k$ -linear approximation to  $\gamma(h)$  on this interval is denoted by

$$(A2.3.2) \quad l_k(h) = a_k \cdot h + b_k$$

then by definition,

$$(A2.3.3) \quad h \in [d_{k-1}, d_k) \Rightarrow |\gamma_k(h) - l_k(h)| \leq \varepsilon_k$$

In this context we have the following bound on the bias of the empirical variogram estimates,

$$(A2.3.4) \quad \hat{\gamma}(h_k) = \frac{1}{2|N_k|} \sum_{(s_i, s_j) \in N_k} (Y(s_i) - Y(s_j))^2$$

at lag distance,  $h_k$  :

**Proposition A2.1.** *If for any bin,  $k = 1, \dots, \bar{k}$ , the true variogram,  $\gamma(h)$ , has an  $\varepsilon_k$ -linear approximation, then at lag distance,  $h_k$ , it must be true that*

$$(A2.3.5) \quad |E[\hat{\gamma}(h_k)] - \gamma(h_k)| \leq 2\varepsilon_k$$

**Proof:** If for each  $(s_i, s_j) \in N_k$  we let

$$(A2.3.6) \quad \gamma_{ij} = \gamma(h_{ij}) = \frac{1}{2} E \left[ (Y(s_i) - Y(s_j))^2 \right]$$

with  $h_{ij} = \|s_i - s_j\|$ , then by (A2.3.4),

$$(A2.3.7) \quad \begin{aligned} E[\hat{\gamma}(h_k)] &= E \left[ \frac{1}{2|N_k|} \sum_{(s_i, s_j) \in N_k} (Y(s_i) - Y(s_j))^2 \right] \\ &= \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} \left\{ \frac{1}{2} E \left[ (Y(s_i) - Y(s_j))^2 \right] \right\} \\ &= \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} \gamma_{ij} \end{aligned}$$

But since  $h_{ij} \in [d_{k-1}, d_k)$  for all  $(s_i, s_j) \in N_k$ , we see from (A2.3.2) that  $|\gamma_{ij} - l_k(h_{ij})| \leq \varepsilon_k$ , and thus that

$$(A2.3.8) \quad -\varepsilon_k \leq \gamma_{ij} - l_k(h_{ij}) \leq \varepsilon_k \text{ for all } (s_i, s_j) \in N_k$$

Hence by summing this set of inequalities and taking averages [with the observation that  $(1/|N_k|) \sum_{(s_i, s_j) \in N_k} \varepsilon_k = (|N_k|/|N_k|) \varepsilon_k = \varepsilon_k$ ], we have

$$(A2.3.9) \quad -\varepsilon_k \leq \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} [\gamma_{ij} - l_k(h_{ij})] \leq \varepsilon_k$$

Next, by using (A2.3.1), (A2.3.2) and (A2.3.7), the middle expression of (A2.3.9) can be rewritten as,

$$\begin{aligned} (A2.3.10) \quad \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} [\gamma_{ij} - l_k(h_{ij})] &= \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} \gamma_{ij} - \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} l_k(h_{ij}) \\ &= E[\hat{\gamma}(h_k)] - \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} [a_k h_{ij} + b_k] \\ &= E[\hat{\gamma}(h_k)] - a_k \left( \frac{1}{|N_k|} \sum_{(s_i, s_j) \in N_k} h_{ij} \right) - b_k \\ &= E[\hat{\gamma}(h_k)] - a_k h_k - b_k \\ &= E[\hat{\gamma}(h_k)] - l_k(h_k) \end{aligned}$$

so that (A2.3.9) is seen to imply that

$$(A2.3.11) \quad -\varepsilon_k \leq E[\hat{\gamma}(h_k)] - l_k(h_k) \leq \varepsilon_k$$

But since  $h_k \in [d_{k-1}, d_k)$  it also follows from (A2.3.3) that  $|l_k(h_k) - \gamma(h_k)| \leq \varepsilon_k$  and hence that

$$(A2.3.12) \quad -\varepsilon_k \leq l_k(h_k) - \gamma(h_k) \leq \varepsilon_k$$

Finally, by adding (A2.3.11) and (A2.3.12) we may conclude that

$$\begin{aligned} (A2.3.13) \quad -2\varepsilon_k &\leq (E[\hat{\gamma}(h_k)] - l_k(h_k)) + (l_k(h_k) - \gamma(h_k)) \leq 2\varepsilon_k \\ &\Rightarrow -2\varepsilon_k \leq E[\hat{\gamma}(h_k)] - \gamma(h_k) \leq 2\varepsilon_k \\ &\Rightarrow |E[\hat{\gamma}(h_k)] - \gamma(h_k)| \leq 2\varepsilon_k \end{aligned}$$

and thus that (A2.3.5) must hold. ■

### A2.4 Some Basic Vector Geometry

In order to understand multidimensional analysis, one must begin with vector geometry. In particular, all matrix manipulations are interpretable geometrically. If for any vector,  $x = (x_1, \dots, x_n) \in \mathbb{R}^n$  we denote the (Euclidean) *length* of  $x$  by

$$(A2.4.1) \quad \|x\| = \sqrt{x'x} = \sqrt{\sum_{i=1}^n x_i^2}$$

then for any two vectors,  $x = (x_1, \dots, x_n), y = (y_1, \dots, y_n) \in \mathbb{R}^n$ , the *distance* between  $x$  and  $y$  is just the length of the vector  $x - y = (x_1 - y_1, \dots, x_n - y_n) \in \mathbb{R}^n$ , i.e.,

$$(A2.4.2) \quad \|x - y\| = \sqrt{(x - y)'(x - y)} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

This is illustrated for two dimensions ( $\mathbb{R}^2$ ) in Figures A2.1 and A2.2 below.

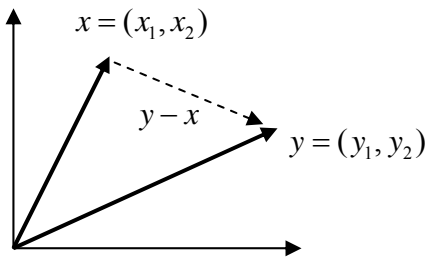


Figure A2.1. Vectors

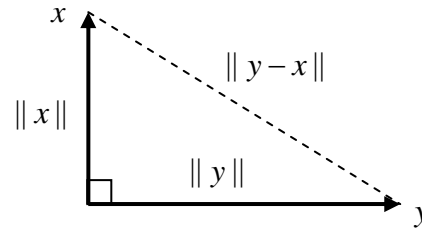


Figure A2.2. Orthogonal Vectors

These distances in turn define angles, that complete the geometry of Euclidean spaces,  $\mathbb{R}^n$ . All that is really required here is the notion of *orthogonal vectors* which constitute the sides of a right triangle, as shown for  $\mathbb{R}^2$  in Figure A2.2. Recall from the Pythagorean Theorem, that such triangles are characterized by the familiar identity that the square of the hypotenuse equals the sum of squares of the sides, i.e.,

$$(A2.4.3) \quad \|x\|^2 + \|y\|^2 = \|x - y\|^2$$

Hence if we now write this orthogonality relation as,  $x \perp y$ , then terms of the notation above, this implies that

$$(A2.4.4) \quad \begin{aligned} x \perp y &\Leftrightarrow \|x\|^2 + \|y\|^2 = \|x - y\|^2 \\ &\Leftrightarrow x'x + y'y = (x - y)'(x - y) = x'x - 2x'y + y'y \\ &\Leftrightarrow x'y = 0 \end{aligned}$$

Hence we are led to the fundamental geometric relation that *orthogonality between vectors is equivalent to zero inner products*. This essentially defines *vector geometry* in Euclidean spaces. (A somewhat sharper derivation of this result is given in terms of cosines in Section ?? below.)

### A2.5 Differentiation of Functions

Our main objective here is to develop *multidimensional optimization problems*, both with and without constraints. The key analytical tools are differential measurements of change in functional values. First recall that the *derivative* of a scalar (i.e., one-dimensional) function  $f(x)$  at a point  $x_0$  is just the slope of the function at  $x_0$ , as defined by the limiting slope of a series of triangles shown in Figure A2.3 below.

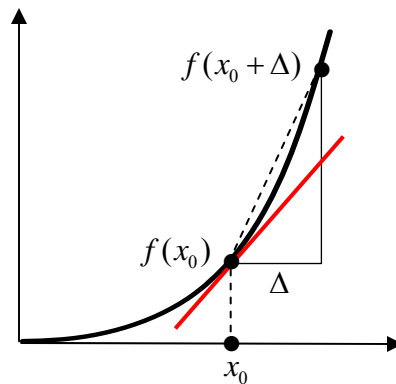


Figure A2.3. Derivatives of Scalar Functions

In formal terms, this is written as<sup>1</sup>

$$(A2.5.1) \quad \frac{d}{dx} f(x_0) = \lim_{\Delta \rightarrow 0} \frac{f(x_0 + \Delta) - f(x_0)}{\Delta}$$

The example in Figure A2.3 is a simple parabolic function,  $f(x) = x^2$ , for which the derivative is given explicitly by

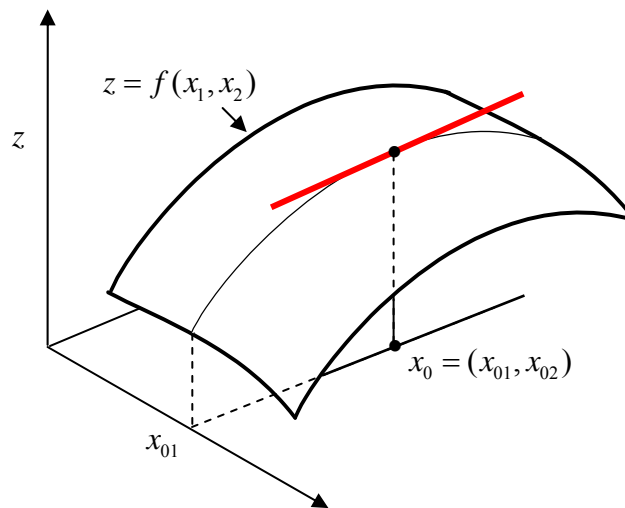
$$(A2.5.2) \quad \begin{aligned} \frac{d}{dx} f(x_0) &= \lim_{\Delta \rightarrow 0} \frac{(x_0 + \Delta)^2 - (x_0)^2}{\Delta} = \lim_{\Delta \rightarrow 0} \frac{x_0^2 + 2x_0\Delta + \Delta^2 - x_0^2}{\Delta} \\ &= \lim_{\Delta \rightarrow 0} (2x_0 + \Delta) = 2x_0 \end{aligned}$$

Such limiting slopes values cannot usually be obtained so easily. But this case serves to illustrate the basic idea.

<sup>1</sup> In Figure A2.3 we have implicitly assumed that increments are positive ( $\Delta > 0$ ). But for smooth functions, the same limiting slope results for negative increments as well.

From a geometric viewpoint, this limiting slope defines the unique *tangent line* to  $f$  at  $x_0$  (shown in red in Figure A2.3). More importantly, the linear function defined by this line yields the *best linear approximation* to function  $f$  in small intervals around  $x_0$  (since by construction it has the same value and slope as  $f$  at  $x_0$ ).

For multidimensional functions,  $f(x) = f(x_1, \dots, x_n)$ , there is no direct parallel to (A2.5.2), since small movements (increments) can occur in many different directions. However, the most fundamental directions are those defined by changes of individual variables holding all others fixed. More formally, the *partial derivative* of  $f(x)$  with respect to variable,  $x_i$ , at a point  $x_0 = (x_{01}, \dots, x_{0i}, \dots, x_{0n})$ , is just the slope of the function when moving in the  $x_i$  direction. This is shown for the  $n=2$  case in Figure A2.4 below, where the partial derivative of  $f(x) = f(x_1, x_2)$  with respect to  $x_1$  at  $x_0 = (x_{01}, x_{02})$  corresponds to the slope of the red line shown.



**Figure A2.4 Partial Derivative**

Again, this can be represented mathematically by the limit

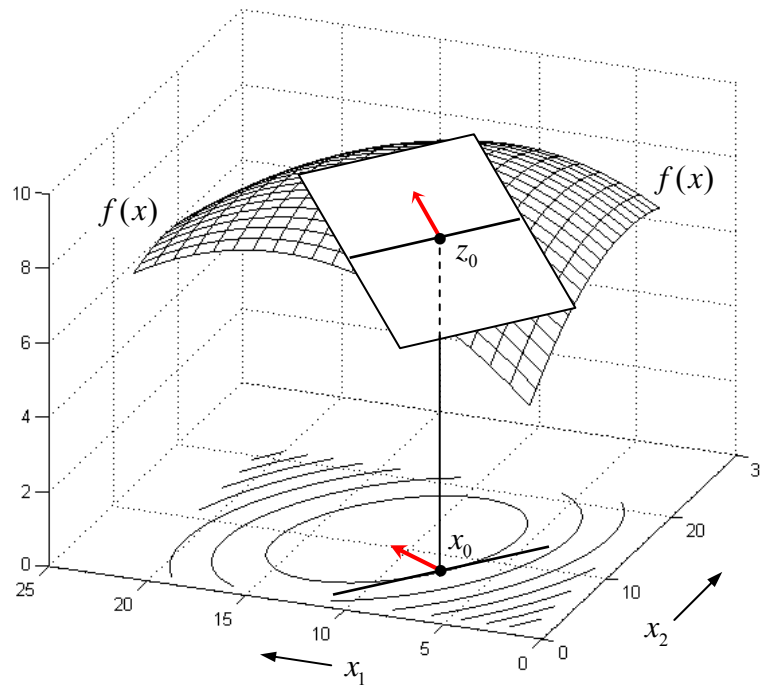
$$(A2.5.3) \quad \frac{\partial}{\partial x_i} f(x_0) = \lim_{\Delta_i \rightarrow 0} \frac{f(x_{01}, \dots, x_{0i} + \Delta_i, \dots, x_{0n}) - f(x_0)}{\Delta_i}$$

For example, if  $f(x_1, x_2) = 2x_1^2 + x_2^2$ , then

$$(A2.5.3) \quad \frac{\partial}{\partial x_1} f(x_0) = \lim_{\Delta_1 \rightarrow 0} \frac{[2(x_{01} + \Delta_1)^2 + x_{02}^2] - [2x_{01}^2 + x_{02}^2]}{\Delta_1}$$

$$\begin{aligned}
 &= \lim_{\Delta_1 \rightarrow 0} \frac{[2(x_{01}^2 + 2x_{01}\Delta_1 + \Delta_1^2) + x_{02}^2] - [2x_{01}^2 + x_{02}^2]}{\Delta_1} \\
 &= \lim_{\Delta_1 \rightarrow 0} (4x_{01} + 2\Delta_1) = 4x_{01}
 \end{aligned}$$

These partial derivatives can in turn be used to define differential changes in *any* direction. The key point to note is that for smooth functions,  $f(x) = f(x_1, \dots, x_n)$ , in higher dimensions, the unique *tangent line* defining the scalar derivative in Figure A2.3 is replaced by a unique *tangent plane*. This is again illustrated by the two-dimensional function,<sup>2</sup>  $f(x) = f(x_1, x_2)$ , shown in Figure A2.5 below:

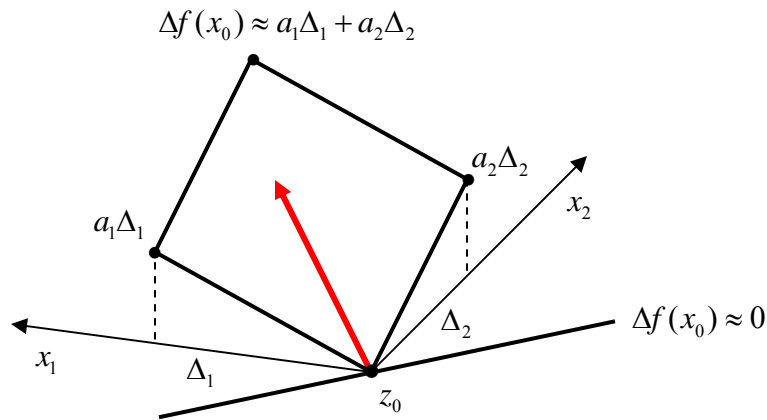


**Figure A2.5. Tangent Planes**

As in the scalar case, the plane tangent to  $f$  at a given point,  $x_0 = (x_{01}, \dots, x_{0n})$ , is essentially the “best linear approximation” to  $f$  in small neighborhoods of  $x_0$ . In geometric terms, this tangent plane is more accurately described as the  $n$ -dimensional (hyper) plane tangent to the *surface* (or *graph*) of  $f$  at the point  $[x_0, f(x_0)] \in \mathbb{R}^{n+1}$ , as illustrated by the 2-dimensional plane tangent to  $f$  at  $z_0 = [x_0, f(x_0)] \in \mathbb{R}^3$  in the figure (where the “red arrows” can be ignored for the moment).

<sup>2</sup> The actual function plotted is the quadratic function,  $f(x) = f(x_1, x_2) = 10 - [2y_1 + y_1y_2 + y_2^2]$  with  $y_i = x_i - 10$ ,  $i = 1, 2$ .

If we continue to focus on this two-dimensional case for the present, and consider any small change in  $x_0$ , say  $x_0 \rightarrow x_0 + \Delta = (x_{01} + \Delta_1, x_{02} + \Delta_2)$ , then the corresponding change in  $f$ , denoted by  $\Delta f(x_0)$ , is well approximated by a corresponding movement on this tangent plane. As we have already seen, movement in the  $x_1$  direction (with  $\Delta_2 = 0$ ) yields changes governed entirely by the partial derivative of  $f$  with respect to  $x_1$  at  $x_0$ . This can now be depicted graphically as in Figure A2.6 below, where for notational simplicity we have represented the partial derivative of  $f$  with respect to  $x_i$  at  $x_0$  by  $a_i = \partial f(x_0) / \partial x_i$ ,  $i = 1, 2$ . Here we have also shifted the origin up to the point,  $z_0 = [x_0, f(x_0)]$ , so that local movements away from  $x_0$  can be represented simply by pairs  $(\Delta_1, \Delta_2)$ . [Note that the size of these shifts (relative to the “red arrow” from Figure A2.5) have been exaggerated for visual clarity.]



**Figure A2.6. Local Linear Approximations**

In this graphical depiction, a movement of  $(\Delta_1, 0)$  yields an increase in  $f(x_0)$  given approximately by,  $a_1 \Delta_1$ , as shown in the figure. Similarly, a movement of  $(0, \Delta_2)$  yields an approximate increase of  $a_2 \Delta_2$ . So by linearity, it follows that for the combined movement,  $(\Delta_1, \Delta_2)$ , the *total increment* in  $f(x_0)$  is approximated by,<sup>3</sup>

$$(A2.5.4) \quad \Delta f(x_0) \approx a_1 \Delta_1 + a_2 \Delta_2 = \left( \frac{\partial f(x_0)}{\partial x_1} \right) \Delta_1 + \left( \frac{\partial f(x_0)}{\partial x_2} \right) \Delta_2$$

Finally, if these  $\Delta$ -shifts are allowed to become “arbitrarily small”, then we obtain the limiting differential relation

<sup>3</sup> Here the symbol,  $\approx$ , can be loosely read as “is approximately equal to”.

$$(A2.5.5) \quad df(x_0) = \left( \frac{\partial f(x_0)}{\partial x_1} \right) dx_1 + \left( \frac{\partial f(x_0)}{\partial x_2} \right) dx_2$$

designated as the *total derivative* of  $f$ . Hence in higher dimensions, scalar derivatives in (A2.5.1) are replaced by the total derivatives in (A2.5.5).

## A2.6 Gradient Vectors

But for our present purposes, the key property of total derivatives is what they imply about partial derivatives in particular. Here we use some vector geometry by first writing the vector of differential elements in (A2.5.5) as  $dx = (dx_1, dx_2)'$ . In geometric terms, this can be viewed as a *directional vector* of small movements from any given point. Similarly, if we designate the vector of partial derivatives of  $f$  at  $x = (x_1, x_2)'$  as,

$$(A2.6.1) \quad \nabla f(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \end{pmatrix}$$

then (A2.5.5) can be rewritten in vector form as:

$$(A2.6.2) \quad df(x_0) = \nabla f(x_0)' dx$$

To interpret this geometrically, observe that if we now consider the contour representation of  $f$ , shown as ellipses on the  $(x_1, x_2)$ -plane in Figure A2.5, then the curve passing through  $x_0$  is by definition the contour with constant value,  $f(x_0)$ . Similarly, the line tangent to this contour is simply the “linear contour” of the corresponding tangent plane, shown by the horizontal (constant height) line passing through  $z_0$ . This tangent line thus defines the directions of movement from  $x_0$  yielding no change in  $f$ . But by (A2.5.4) these directions,  $dx$ , are given precisely by the *no change condition*:

$$(A2.6.3) \quad 0 = df(x_0) = \nabla f(x_0)' dx$$

Hence, by recalling (A2.4.4), we see that the key geometric consequence of this zero-inner-product condition is that the vector of partial derivatives,  $\nabla f(x_0)$ , must necessarily be *orthogonal* to the directions of no change in  $f$ . In Figure A2.5,  $\nabla f(x_0)$  thus corresponds to the *red arrow* on the  $(x_1, x_2)$ -plane starting at  $x_0$ . Moreover, since its three-dimensional counterpart starting at  $z_0$  on the tangent plane (in both Figures A2.5 and A2.6) is necessarily the steepest direction of movement on this plane, it

follows that  $\nabla f(x_0)$  defines the direction of movement in the  $(x_1, x_2)$ -plane yielding a *maximum increase* in  $f$  at  $x_0$ . For this reason, the vector of partial derivatives,  $\nabla f(x_0)$ , is usually called the *gradient vector* of  $f$  at  $x_0$ .

Finally, while the  $n=2$  case is extremely useful for gaining geometric intuition, it should be emphasized that all relationships above are immediately extendable to general functions,  $f(x) = f(x_1, \dots, x_n)$ . In particular, if we let  $dx = (dx_1, \dots, dx_n)'$  and define the general gradient vector at  $x = (x_1, \dots, x_n) \in \mathbb{R}^n$  by

$$(A2.6.4) \quad \nabla f(x) = \begin{pmatrix} \nabla_1 f(x) \\ \vdots \\ \nabla_n f(x) \end{pmatrix} = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{pmatrix}$$

then (A2.6.2) and (A2.6.3) continue to hold in  $\mathbb{R}^n$ .

## A2.7 Unconstrained Optimization of Smooth Functions

Given these key geometric results, we can now consider *optimization problems* involving smooth multidimensional functions,  $f(x) = f(x_1, \dots, x_n)$ . These amount to finding points,  $x$ , in some specified region,  $R \subseteq \mathbb{R}^n$ , with either maximum or minimum values,  $f(x)$ , in  $R$ , depending on the given problem. Here it is important to emphasize that *maximizing* the function,  $f(x)$ , over  $R$  is equivalent to minimizing the function,  $-f(x)$ , over  $R$ . For this reason, it suffices to consider only *maximization problems* (which are usually easier to depict graphically for the  $n=2$  case).<sup>4</sup>

In this context, an *unconstrained maximization problem* for our purposes is taken to be one in which the maximum of  $f(x)$  is known to be achieved at some *interior point* of  $R$ , and hence is a smooth maximum that can be characterized by the derivatives of  $f$ . In the scalar case, this is the usual “zero-slope” condition that the derivative be zero at the maximum, as shown for the scalar function,  $f(x)$ , in Figure A2.7 below. Here the maximum at  $x_0$  is seen to be uniquely characterized by this zero-slope condition. But even with a unique maximum, this condition is by no means sufficient. Even when there are no other local maxima (or minima), it is still possible to have other *singular points*, i.e., with zero slope. Figure A2.8 illustrates a *singular inflection*<sup>5</sup> point which is neither a local minimum or maximum. In the scalar case, such possibilities can be

<sup>4</sup> Here it is also worth noting that optimization software (such the MATLAB optimization toolbox) is typically designed to do only *minimization* problems. So all maximization problems must be reformulated as minimization problems.

<sup>5</sup> An *inflection point*,  $x$ , for  $f$  is a point at which the *second derivative* of  $f$  changes sign.

eliminated by requiring that the *second derivative* be *negative* at all singular points, so that the unique maximum is always characterized by the zero-slope condition. This is precisely analogous to the one-dimensional kriging problem in Section 6.2.1 of the text. Here a global minimum was insured for the simple quadratic function in (6.2.19) with *positive* second derivative in (6.2.20).

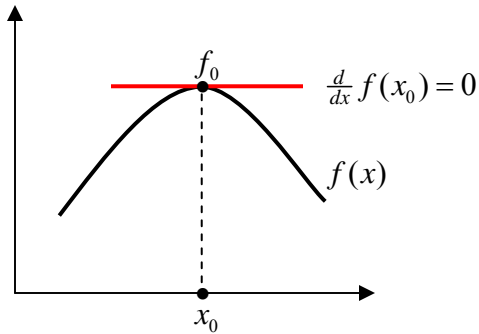


Figure A2.7. Scalar Maximum

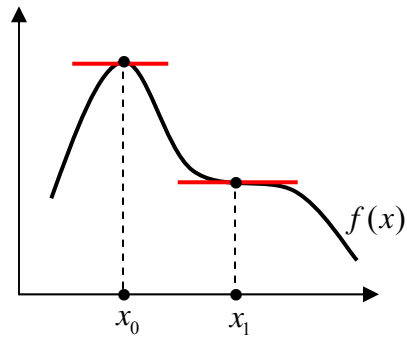


Figure A2.8. Singular Inflection

The situation is more complex for multidimensional functions. Here the first-order “zero-slope” condition,  $\frac{d}{dx} f(x_0) = 0$ , is replaced by a more general “zero-gradient” condition,  $\nabla f(x_0) = 0$ , which ensures that the total derivative in (A2.6.2) is zero in *all* directions,  $dx$ .<sup>6</sup> Geometrically, this first-order condition requires that the tangent plane at  $x_0$  be flat, as is illustrated in Figure A2.9 below.

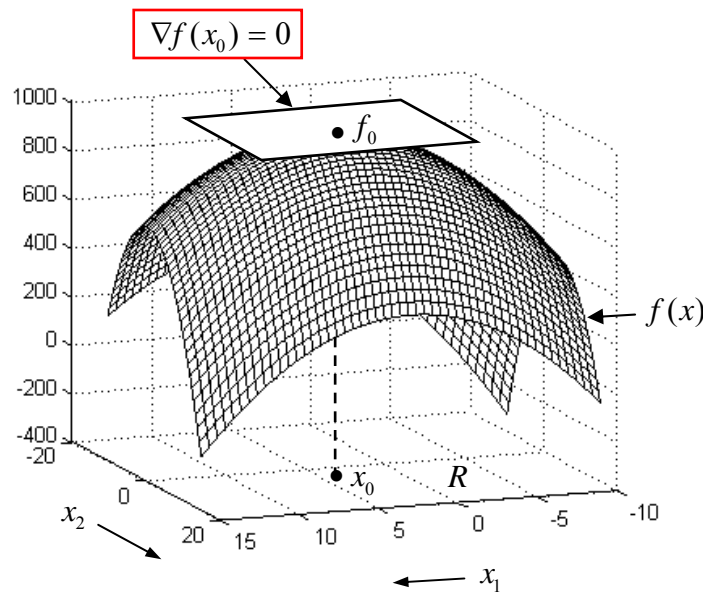


Figure A2.9. First Order Condition for a Maximum

<sup>6</sup> Note that since  $\nabla f(x_0)$  is an  $n$ -vector, the “0” here is also an  $n$ -vector,  $0 = (0, \dots, 0)'$ . While we could write this as  $0_n$ , standard practice is to take the dimension of zero vectors as understood by context.

The function,  $f(x) = f(x_1, x_2)$ , actually shown Figure A2.9 is bivariate quadratic function, which takes the explicit form

$$(A2.7.1) \quad f(x_1, x_2) = 928 + 26x_1 + 20x_2 - 3x_1^2 - x_1x_2 - 4x_2^2$$

So by taking the partial derivatives of this function and setting them equal to zero, we obtain the relations,

$$(A2.7.2) \quad 0 = \frac{\partial}{\partial x_1} f(x_0) = 26 - 6x_1 - x_2$$

$$(A2.7.3) \quad 0 = \frac{\partial}{\partial x_2} f(x_0) = 20 - 8x_1 - x_2$$

These linear equations can easily be solved to yield the unique solution point,  $x_0 = (x_{01}, x_{02})' = (4, 2)'$ , shown in the figure. However when the dimension,  $n$ , is much larger than two, it is practically possible to write down the full expression for  $f(x) = f(x_1, \dots, x_n)$ , let alone the simultaneous equation system corresponding to the first-order condition. Here is where the power of matrix algebra takes full force. If we let

$$(A2.7.4) \quad A = \begin{pmatrix} 3 & 2 \\ -1 & 4 \end{pmatrix}, \quad b = \begin{pmatrix} 26 \\ 20 \end{pmatrix}, \quad c = 928$$

then it can easily be verify (by matrix multiplication) that the function in (A2.7.1) can be equivalently written in matrix form for all  $x = (x_1, x_2)'$  as,

$$(A2.7.5) \quad f(x) = c + b'x - x'Ax$$

Notice the similarity of this *quadratic form* to the general expression for mean squared error,  $MSE(\lambda_0)$ , in expression (6.2.27), where  $x$  now plays the role of the weight vector,  $\lambda_0$ .<sup>7</sup> The power of this notation is that the quadratic form in (A2.7.5) can be analyzed in the same way *regardless* of the dimension,  $n$ . All that is required here is that we formalize the vector version of the partial derivatives in (A2.7.2) and (A2.7.3). To do so, notice first that for any coefficient vector,  $b = (b_1, b_2)$ , such as in (A2.7.4), if we now employ the gradient notation in (A2.6.4) then it follows that,

$$(A2.7.6) \quad \nabla(b'x) = \begin{pmatrix} \nabla_1(b'x) \\ \nabla_2(b'x) \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial x_1}(b_1x_1 + b_2x_2) \\ \frac{\partial}{\partial x_2}(b_1x_1 + b_2x_2) \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = b$$

<sup>7</sup> It is also worth noticing the difference in *signs* of the quadratic term, where  $MSE$  was to be minimized, and  $f$  is to be maximized. We shall return to this distinction below.

More generally, for any linear compound,  $b'x = \sum_{i=1}^n b_i x_i$ , exactly the same argument shows that

$$(A2.7.7) \quad \nabla(b'x) = b$$

Turning next to the quadratic term in (A2.7.5) observe that for any  $2 \times 2$  matrix,  $A$ ,

$$(A2.7.8) \quad x'Ax = (x_1 \ x_2) \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = (x_1 \ x_2) \begin{pmatrix} a_{11}x_1 + a_{12}x_2 \\ a_{21}x_1 + a_{22}x_2 \end{pmatrix} \\ = a_{11}x_1^2 + a_{12}x_1x_2 + a_{21}x_2x_1 + a_{22}x_2^2$$

so that the corresponding partial derivative expression can be written as

$$(A2.7.9) \quad \nabla(x'Ax) = \begin{pmatrix} \frac{\partial}{\partial x_1} (a_{11}x_1^2 + a_{12}x_1x_2 + a_{21}x_2x_1 + a_{22}x_2^2) \\ \frac{\partial}{\partial x_2} (a_{11}x_1^2 + a_{12}x_1x_2 + a_{21}x_2x_1 + a_{22}x_2^2) \end{pmatrix} \\ = \begin{pmatrix} 2a_{11}x_1 + a_{12}x_2 + a_{21}x_2 \\ 2a_{22}x_2 + a_{12}x_1 + a_{21}x_1 \end{pmatrix} = \begin{pmatrix} a_{11}x_1 + a_{12}x_2 \\ a_{21}x_1 + a_{22}x_2 \end{pmatrix} + \begin{pmatrix} a_{11}x_1 + a_{21}x_2 \\ a_{12}x_1 + a_{22}x_2 \end{pmatrix} \\ = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = Ax + A'x$$

More generally, for any quadratic expression,  $x'Ax = \sum_{i=1}^n \sum_{j=1}^n a_{ij}x_i x_j$ , essentially the same argument shows that

$$(A2.7.10) \quad \nabla(x'Ax) = (A + A')x$$

Here there is one important special case, namely when the matrix  $A$  is symmetric, i.e., when  $A' = A$ . For this case it follows at once from (A2.7.10) that

$$(A2.7.11) \quad \nabla(x'Ax) = 2Ax$$

To see the special relevance of this case, notice that every square matrix,  $A$ , has an associated *symmetrization*,

$$(A2.7.12) \quad A_s = \frac{1}{2}(A + A') \Rightarrow A'_s = \frac{1}{2}(A' + A) = A_s$$

But since  $x'y = y'x$  for all vectors, it then follows that

$$\begin{aligned}
 \text{(A2.7.13)} \quad x'A_s x &= x' \left[ \frac{1}{2}(A + A') \right] x = \frac{1}{2} [x'Ax + x'A'x] \\
 &= \frac{1}{2} [x'(Ax) + (Ax)'x] = \frac{1}{2} [x'(Ax) + x'(Ax)] = x'Ax
 \end{aligned}$$

So in fact, *every* quadratic expression,  $x'Ax$ , can be represented by a *symmetric* matrix as  $x'A_s x$ . As one illustration, observe that the matrix  $A$  in (A2.7.4) is not symmetric. So in this case, one could replace  $A$  with the symmetric matrix,

$$\text{(A2.7.14)} \quad A_s = \frac{1}{2} \left[ \begin{pmatrix} 3 & 2 \\ -1 & 4 \end{pmatrix} + \begin{pmatrix} 3 & -1 \\ 2 & 4 \end{pmatrix} \right] = \begin{pmatrix} 3 & 1/2 \\ 1/2 & 4 \end{pmatrix}$$

### A2.7.1 First-Order Conditions

Using these identities, we can now establish first-order conditions for any quadratic maximization problem as follows. If  $f(x)$  is assumed to have the general quadratic form

$$\text{(A2.7.15)} \quad f(x) = c + b'x + x'Ax$$

with  $A$  symmetric, then by linearity of differentiation [i.e.,  $\nabla(f + g) = \nabla f + \nabla g$ ] we have:

$$\text{(A2.7.16)} \quad \nabla f(x) = \nabla(c + b'x + x'Ax) = 0 + \nabla(b'x) + \nabla(x'Ax) = b + 2Ax$$

So the *first-order condition* for a maximum of  $f(x)$  can be solved as follows:

$$\text{(A2.7.17)} \quad 0 = \nabla f(x_0) = b + 2Ax_0 \Rightarrow 2Ax_0 = -b \Rightarrow x_0 = -\frac{1}{2}A^{-1}b$$

In the present case, where (symmetric)  $A$  is given by the negative of (A2.7.14) [to be consistent with (A2.7.15)] it follows that

$$\text{(A2.7.18)} \quad x_0 = -\frac{1}{2}(-A_s)^{-1}b = \frac{1}{2}A_s^{-1}b = \frac{1}{2} \begin{pmatrix} 3 & 1/2 \\ 1/2 & 4 \end{pmatrix}^{-1} \begin{pmatrix} 26 \\ 20 \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \end{pmatrix}$$

which is precisely the solution shown in Figure A2.9.

If this same line of reasoning is applied to the *mean-squared-error* function

$$\text{(A2.7.19)} \quad MSE(\lambda_0) = \sigma^2 - 2c'_0 \lambda_0 + \lambda'_0 V_0 \lambda_0$$

in expression (6.2.25), we can now solve the corresponding first-order condition for the *optimal weight vector*,  $\hat{\lambda}_0$ , as follows

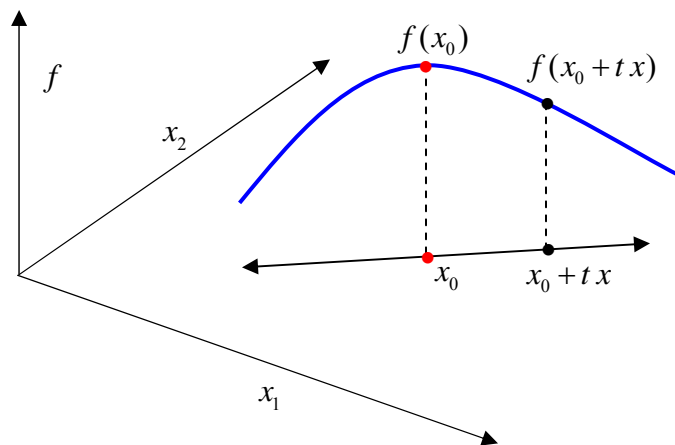
$$(A2.7.20) \quad 0 = \nabla \text{MSE}(\hat{\lambda}_0) = -2c_0 + 2V_0 \hat{\lambda}_0 \Rightarrow V_0 \hat{\lambda}_0 = c_0 \Rightarrow \hat{\lambda}_0 = V_0^{-1}c_0$$

which is seen to be precisely the simple kriging solution in expression (6.2.26).

But while these first-order conditions are necessary for optimal solutions, they are *not* sufficient. In particular, (A2.7.18) is claimed to be the solution of a *maximization* problem, and (A2.7.20) is claimed to be the solution of a *minimization* problem. Hence to check whether either of these are actually solutions of their respective problems, we must develop appropriated second-order conditions.

### A2.7.2 Second-Order Conditions

Recall that in the scalar case, the second-order condition for a *maximum* (or *minimum*) of  $f(x)$  at  $x_0$  is that the second derivative,  $\frac{d^2}{dx^2} f(x_0)$ , be *negative* (or *positive*), as seen for the case of a maximum in Figure A2.7 above. In the multidimensional case the conditions are similar in nature, but are necessarily somewhat more complex. The simplest way to motivate the basic idea here is to reduce the problem to “one dimension” in the following way. For a two dimensional function,  $f(x) = f(x_1, x_2)$ , with a maximum at point,  $x_0$ , such as in Figure A2.9 above, consider a one-dimensional “slice” through this function such as the one shown in Figure A2.10 below.



**Figure A2.10 One-Dimensional Slices**

Such a slice can be defined formally by choosing any fixed nonzero vector,  $x$ , and considering all linear combinations,  $x_0 + tx$ . As the scalar,  $t$ , increases from zero, one moves away from  $x_0$  in “direction”  $x$ . Similarly, as  $t$  decreases from zero, one moves

in the opposite direction. The one-dimensional slice through  $f$  shown in the figure thus corresponds precisely to the *scalar* function of  $t$  defined by  $g_x(t) = f(x_0 + tx)$ . So if  $f$  achieves its maximum at  $x_0$ , then in particular, it must exhibit a maximum along this slice at  $t = 0$ . This of course implies that  $\frac{d}{dt}g_x(0) = 0$ , and more importantly for our present purposes, that  $\frac{d^2}{dt^2}g_x(0) < 0$ . To analyze this latter condition more explicitly, we introduce the following simplifying notation. For any function,  $f(x) = f(x_1, \dots, x_n)$  of  $n$  arguments, let

$$(A2.7.21) \quad f_i(x) = \frac{\partial}{\partial x_i} f(x_1, \dots, x_i, \dots, x_n)$$

denote the *partial derivative* of  $f$  with respect to its  $i$ -th argument, and for each  $i, j = 1, \dots, n$  let

$$(A2.7.22) \quad f_{ij}(x) = \frac{\partial}{\partial x_i} f_j(x_1, \dots, x_i, \dots, x_n) = \frac{\partial^2}{\partial x_i \partial x_j} f(x_1, \dots, x_i, \dots, x_j, \dots, x_n)$$

denote the *cross partial derivative* of  $f$  with respect to its  $i$ -th and  $j$ -th arguments (so that in particular,  $f_{ii}(x)$  is the *second partial derivative* of  $f$  with respect to its  $i$ -th argument). In terms of this notation, if we consider a compound function,  $g(t) = f[h_1(t), h_2(t)]$ , and recall from the chain rule for derivatives that

$$(A2.7.23) \quad \frac{d}{dt}g(t) = f_1[h_1(t), h_2(t)] \frac{d}{dt}h_1(t) + f_2[h_1(t), h_2(t)] \frac{d}{dt}h_2(t)$$

then by applying this rule to the function  $g_x(t)$  above, we see that

$$\begin{aligned} (A2.7.24) \quad \frac{d}{dt}g_x(t) &= \frac{d}{dt}f(x_0 + tx) = \frac{d}{dt}f(x_{01} + tx_1, x_{02} + tx_2) \\ &= f_1(x_0 + tx) \cdot \frac{d}{dt}(x_{01} + tx_1) + f_2(x_0 + tx) \cdot \frac{d}{dt}(x_{02} + tx_2) \\ &= f_1(x_0 + tx) \cdot x_1 + f_2(x_0 + tx) \cdot x_2 \end{aligned}$$

Differentiating once again we have

$$(A2.7.25) \quad \frac{d^2}{dt^2}g_x(t) = \frac{d}{dt}[\frac{d}{dt}g_x(t)] = \frac{d}{dt}[f_1(x_0 + tx)] \cdot x_1 + \frac{d}{dt}[f_2(x_0 + tx)] \cdot x_2$$

So by applying the chain rule to the first term on the right, we obtain

$$\begin{aligned} (A2.7.26) \quad \frac{d}{dt}[f_1(x_0 + tx)] \cdot x_1 &= \frac{d}{dt}[f_{11}(x_{01} + tx_1, x_{02} + tx_2)] \cdot x_1 \\ &= [f_{11}(x_0 + tx)x_1 + f_{12}(x_0 + tx) \cdot x_2] \cdot x_1 \end{aligned}$$

$$= f_{11}(x_0 + tx) \cdot x_1^2 + f_{12}(x_0 + tx) \cdot x_1 x_2$$

Similarly, the second term in (A2.7.25) can be written out as

$$(A2.7.27) \quad \frac{d}{dt}[f_2(x_0 + tx)] \cdot x_2 = f_{21}(x_0 + tx) \cdot x_2 x_1 + f_{22}(x_0 + tx) \cdot x_2^2$$

By combining these, we can now write the second derivative in (A2.7.25) more explicitly as

$$(A2.7.28) \quad \frac{d^2}{dt^2} g_x(t) = f_{11}(x_0 + tx) \cdot x_1^2 + f_{12}(x_0 + tx) \cdot x_1 x_2 \\ + f_{21}(x_0 + tx) \cdot x_2 x_1 + f_{22}(x_0 + tx) \cdot x_2^2$$

Finally, by evaluating this at  $t = 0$ , we obtain the explicit second-order condition

$$(A2.7.29) \quad \frac{d^2}{dt^2} g_x(0) = f_{11}(x_0) \cdot x_1^2 + f_{12}(x_0) \cdot x_1 x_2 + f_{21}(x_0) \cdot x_2 x_1 + f_{22}(x_0) \cdot x_2^2 < 0$$

### The Hessian Matrix

This second-order condition can be written more compactly in matrix form as follows. If we now designate the matrix of cross partial derivatives of  $f$  at point  $x_0$  as the *Hessian matrix*,

$$(A2.7.30) \quad H_f(x_0) = \begin{bmatrix} f_{11}(x_0) & f_{21}(x_0) \\ f_{12}(x_0) & f_{22}(x_0) \end{bmatrix}$$

then the right hand side of (A2.7.29) can be written in matrix terms as

$$(A2.7.31) \quad \frac{d^2}{dt^2} g_x(0) = (x_1, x_2) \begin{bmatrix} f_{11}(x_0) & f_{21}(x_0) \\ f_{12}(x_0) & f_{22}(x_0) \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x' H_f(x_0) x$$

Hence the desired second order condition for a maximum of  $f$  at  $x_0$  with respect to direction  $x$  takes the simple form:

$$(A2.7.32) \quad x' H_f(x_0) x < 0$$

Before proceeding, it is appropriate to extend condition (A2.7.32) to the general case of  $n$  dimensions. Here it is enough to observe that while the  $n = 2$  case permits one-dimensional slices in each direction to be seen graphically (as in Figure A2.10 above), none of the analysis is in any way restricted to this case. Hence, if for any smooth

function,  $f(x) = f(x_1, \dots, x_n)$ , and point  $x_0 \in \mathbb{R}^n$  in the domain of  $f$  we now define the associated Hessian matrix at  $x_0$  by

$$(A2.7.33) \quad H_f(x_0) = \begin{pmatrix} f_{11}(x_0) & \cdots & f_{n1}(x_0) \\ \vdots & \ddots & \vdots \\ f_{1n}(x_0) & \cdots & f_{nn}(x_0) \end{pmatrix}$$

then the argument leading to (A2.7.32) continues to hold for any direction vector,  $x \in \mathbb{R}^n$  and Hessian matrix given by (A2.7.33).

Given this “one dimensional” condition, it remains only to observe that for a true maximum at  $x_0$ , this same condition must hold in *all directions* with respect to  $x_0$ . So if we now designate an  $n$ -square matrix,  $A$ , to be *negative definite* if and only if

$$(A2.7.34) \quad x' A x < 0 \quad \text{for all } x \neq 0$$

then it follows at once from (A2.7.32) and (A2.7.34) that the desired full-dimensional condition for a maximum of  $f$  at  $x_0$  is precisely that the Hessian matrix,  $H_f(x_0)$ , be *negative definite*.

This condition for a maximum also yields a corresponding condition for a *minimum* of  $f$  at  $x_0$ . For the  $n = 2$  case, simply observe that if the “mountain” shape of  $f(x)$  in Figure A2.9 is inverted to “bowl” shape, then it is clear that the function,  $g_x(t) = f(x_0 + tx)$ , corresponding to each slice in Figure A2.10 must now have a *positive* second derivative at  $t = 0$ , i.e.,  $\frac{d^2}{dt^2} g_x(0) > 0$ . Hence same the argument leading to (A2.7.32) now shows that

$$(A2.7.35) \quad x' H_f(x_0) x > 0$$

must hold in each nonzero direction  $x$ . This argument is again directly extendable to  $n$  dimensions (but without pictures). So if we now designate an  $n$ -square matrix,  $A$ , as *positive definite* if and only if

$$(A2.7.36) \quad x' A x > 0 \quad \text{for all } x \neq 0$$

then the parallel full-dimensional condition for a *minimum* of  $f$  at  $x_0 \in \mathbb{R}^n$  is simply that the Hessian matrix,  $H_f(x_0)$ , be *positive definite*.

### Conditions for Symmetric Positive Definiteness

The task remaining is to establish readily testable conditions for determining when a matrix is positive or negative definite. Here we begin by observing from (A2.7.34) and (A2.7.36) that a matrix,  $A$ , is *positive definite* if and only if  $-A$  is *negative definite*. Hence, it suffices to consider only one of these two conditions. Follows standard practice, we here focus on positive definiteness. Next recall from the identity in (A2.7.13) that to establish positive definiteness, we may assume that the matrix  $A$  is *symmetric* (for if not then use its symmetrization,  $A_s$ ). For Hessian matrices in particular, it turns out that such matrices are guaranteed to be *symmetric*, i.e.,  $f_{ij}(x) \equiv f_{ji}(x)$ , whenever these cross partial derivatives are continuous.<sup>8</sup> So we shall focus on conditions for establishing that a symmetric matrix is positive definite.

To motivate the conditions characterizing *symmetric positive definite* (SPD) matrices, we begin with the following fundamental observation which forms the basis for essentially all characterizations of such matrices. An  $n$ -square matrix,  $A$ , is SPD if and only if it can be “decomposed” into a product of the form,

$$(A2.7.37) \quad A = BB'$$

for some nonsingular  $n$ -square matrix,  $B$ . To see this, observe first that since

$$(A2.7.38) \quad A' = (BB')' = (B')'B' = BB' = A$$

it follows that  $A$  must be symmetric. More importantly, observe that since the inner product of a nonzero vector,  $x$ , with itself is always positive, i.e.,

$$(A2.7.39) \quad x \neq 0 \Rightarrow x'x = \sum_{i=1}^n x_i^2 > 0$$

and since the nonsingularity of  $B$  insures that  $Bx \neq 0$  whenever  $x \neq 0$ , it then follows from (A2.7.39) that for all  $x \neq 0$ ,

$$(A2.7.40) \quad x'Ax = x'(BB')x = (B'x)'(B'x) > 0$$

and hence that  $A$  is SPD. This characterization helps to clarify the real meaning of positive definiteness. In particular, if we consider the simplest case,  $n=1$ , and let  $a$  denote the scalar matrix,  $A$ , then the *positive definiteness* condition simply says that for all nonzero scalars,  $x$ , we must have  $x(ax)x = ax^2 > 0$ , which of course simply characterizes positivity of the scalar,  $a$ . So again letting  $b$  denote the scalar matrix,  $B$ , condition (A2.7.39) simply says that  $a$  is positive if and only if it can be written as

<sup>8</sup> This result is usually known as Young's Theorem, and can be found in most calculus textbooks.

$a = b^2$  for some scalar  $b$ , i.e., if and only if it has a real square root,  $b$ .<sup>9</sup> So in this sense, positive definite matrices are the natural generalization of positive numbers. But while this decomposition characterizations is very informative, it is no more “testable” than positive definiteness itself. However, there do exist *testable* conditions for ensuring the existence of such decompositions as we now show.

The simplest and most commonly used test for positive definiteness is based on the properties of certain determinants. If the *determinant* of a  $n$ -square matrix,  $A$ , is denoted by  $\det(A)$ , then this condition involves positivity of the determinants of certain sub-matrices of  $A$ . In particular, for each  $k = 1, \dots, n$  we now designate the  $k$ -square matrix,  $A_k = (a_{ij} : i, j = 1, \dots, k)$ , in the “upper left-hand corner” of  $A = (a_{ij} : i, j = 1, \dots, n)$ , i.e.,

$$(A2.7.41) \quad A = \begin{pmatrix} a_{11} & \cdots & a_{1k} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} & & \vdots \\ \vdots & & & \ddots & \vdots \\ a_{n1} & \cdots & \cdots & \cdots & a_{nn} \end{pmatrix}$$

as the  $k^{\text{th}}$  *leading principle sub-matrix* of  $A$ , and designate its determinant,  $\det(A_k)$ , as the  $k^{\text{th}}$  *leading principle minor* of  $A$ , then the following condition, known as Sylvester’s Condition is both necessary and sufficient for positive definiteness:

**Sylvester’s Condition.** *A symmetric matrix,  $A$ , is positive definite if and only if all principle minors of  $A$  are positive.*

This result will be shown later to a simple consequence of the Spectral Decomposition Theorem for symmetric matrices. To illustrate its application, consider the symmetrized matrix in (A2.7.14) above, i.e.,

$$(A2.7.42) \quad A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} 3 & 1/2 \\ 1/2 & 4 \end{pmatrix}$$

Observe that since the principle minors are  $\det(a_{11}) = a_{11} = 3 > 0$  and

$$(A2.7.43) \quad \det(A) = a_{11}a_{22} - a_{21}a_{12} = (3)(4) - (1/2)^2 > 0$$

it follows at once from Sylvester’s Condition that  $A$  is positive definite.

---

<sup>9</sup> Later we shall see that SPD matrices,  $A$ , actually have square roots as well, i.e., can be written as  $A = B^2$  for a nonsingular symmetric matrix,  $B$ . But this requires the Spectral Decomposition Theorem for symmetric matrices.

But our main interest in Sylvester's condition is that it provides the basis for establish a more useful testable condition that has many applications of its own. In particular, it yields a simple decomposition of SPD matrices known as the *Cholesky decomposition*. In particular, if a matrix,  $T$ , with zeros everywhere above the diagonal, i.e., of the form

$$(A2.7.44) \quad T = \begin{pmatrix} t_{11} & 0 & \cdots & 0 \\ t_{21} & t_{22} & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ t_{n1} & t_{n2} & \cdots & t_{nn} \end{pmatrix}$$

is designated as a *lower triangular* matrix, then matrix  $A$  is said to have a *Cholesky decomposition* if and only if there is a nonsingular lower triangular matrix,  $T$ , such that

$$(A2.7.45) \quad A = TT'$$

By the argument above, every matrix of this form is SPD. Moreover, this again turns out to completely characterize SPD matrices as we now show:<sup>10</sup>

**Cholesky Theorem.** *A symmetric matrix  $A$  is positive definite if and only if there exists a Cholesky decomposition for  $A$ .*

**Proof:** If  $A$  has a Cholesky decomposition then the argument in (A2.7.40) shows that  $A$  is positive definite. Conversely, if  $A$  is positive definite, then by Sylvester's condition, all leading principle minors of  $A$  are positive. Using this property, we can now construct a Cholesky decomposition by induction on the dimension of the  $n$ -square matrix,  $A$ . For  $n = 1$ ,  $A$  is by hypothesis a positive *scalar*, so that we may set  $T = T' = \sqrt{A}$ . Now suppose that it is true for  $n - 1 > 0$  and consider a symmetric  $n$ -square matrix,  $A$ , with all positive principle minors. We may write  $A$  in partitioned form as

$$(A2.7.46) \quad A = \begin{pmatrix} A_{n-1} & a_{n-1} \\ a'_{n-1} & a_{nn} \end{pmatrix}$$

where  $A_{n-1}$  is the  $(n - 1)^{st}$  leading principle sub-matrix of  $A$ . By construction,  $A_{n-1}$  has all positive leading principle minors (namely the first  $n - 1$  leading principle minors of  $A$ ). Thus by hypothesis,  $A_{n-1}$  must have a Cholesky decomposition, say

$$(A2.7.47) \quad A_{n-1} = T_{n-1}T'_{n-1}$$

Our objective is to extend  $T_{n-1}$  to a Cholesky decomposition,  $A = TT'$ , for  $A$  as follows. By lower triangularity,  $T$  must have the form

<sup>10</sup> The following proof is based on an argument given by Prof. David Hill that is available online at: [http://astro.temple.edu/~dhill001/course/math254/CHOLESKYDECOMPOSITION\\_stu.pdf](http://astro.temple.edu/~dhill001/course/math254/CHOLESKYDECOMPOSITION_stu.pdf)

$$(A2.7.48) \quad T = \begin{pmatrix} T_{n-1} & 0 \\ h' & c \end{pmatrix}$$

for some unknown  $(n-1)$ -vector,  $h$ , and scalar,  $c$ . Hence by (A2.7.46) and (A2.7.48) we seek values for  $h$  and  $a$  such that,

$$(A2.7.49) \quad \begin{pmatrix} A_{n-1} & a_{n-1} \\ a'_{n-1} & a_{nn} \end{pmatrix} = \begin{pmatrix} T_{n-1} & 0 \\ h' & c \end{pmatrix} \begin{pmatrix} T'_{n-1} & h \\ 0' & c \end{pmatrix} = \begin{pmatrix} T_{n-1}T'_{n-1} & T_{n-1}h \\ h'T'_{n-1} & h'h + c^2 \end{pmatrix}$$

In particular, this implies both that

$$(A2.7.50) \quad a_{n-1} = T_{n-1}h \quad , \quad \text{and}$$

$$(A2.7.51) \quad a_{nn} = h'h + c^2$$

But by the nonsingularity of  $T_{n-1}$ , we can solve for  $h$  in (A2.7.50) as

$$(A2.7.52) \quad h = T_{n-1}^{-1}a_{n-1}$$

Similarly by (A2.7.51), the value of  $c$  must be given by

$$(A2.7.53) \quad c = \sqrt{a_{nn} - h'h}$$

Hence to complete this construction, it remains only to show that the last operation is legitimate, i.e., that

$$(A2.7.54) \quad a_{nn} - h'h > 0$$

But by the determinant rule for partitioned matrices, it follows from (A2.7.46) that

$$(A2.7.55) \quad \det(A) = \det \begin{pmatrix} A_{n-1} & a_n \\ a'_n & a_{nn} \end{pmatrix} = \det(A_{n-1})(a_{nn} - a'_n A_{n-1}^{-1} a_n)$$

(since  $a_{nn} - a'_n A_{n-1}^{-1} a_n$  is a scalar).<sup>11</sup> Moreover, since the hypothesis of positive leading principle minors for  $A$  implies in particular that  $\det(A) > 0$  and  $\det(A_{n-1}) > 0$ , we see from (A2.7.55) that

<sup>11</sup> To gain some intuition for this determinant rule, observe simply that for the case of  $n = 2$ , we must have

$$\det \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = a_{11}a_{22} - a_{12}a_{21} = (a_{11})(a_{22} - a_{12}a_{11}^{-1}a_{21}).$$

$$(A2.7.56) \quad a_{nn} - a'_{n-1} A_{n-1}^{-1} a_{n-1} > 0$$

Finally by substituting (A2.7.50) into (A2.7.56), we may conclude that

$$\begin{aligned} (A2.7.57) \quad 0 &< a_{nn} - (T_{n-1}h)'(T_{n-1}T_{n-1}')^{-1}(T_{n-1}h) \\ &= a_{nn} - h'T_{n-1}'[(T_{n-1}')^{-1}T_{n-1}^{-1}]T_{n-1}h \\ &= a_{nn} - h'[T_{n-1}'(T_{n-1}')^{-1}][T_{n-1}^{-1}T_{n-1}]h \\ &= a_{nn} - h'h \end{aligned}$$

Thus (A2.7.54) must hold, and the result is established.  $\square$

**Remark:** It should also be noted that this proof yields a *recursive construction* for  $T$ , and in particular shows that it is *unique*. This is obvious for  $n = 1$ , where  $T = \sqrt{A}$  is the only possible choice. Moreover, by recursive use of the constructions in (A2.7.53) and (A2.7.54), one must obtain a unique extension  $T$  for each  $n > 1$ .  $\square$

As noted above, the most attractive feature of Cholesky decompositions is their ease of calculation. As mentioned in the text, this is easily accomplished with the command,

```
>> T = chol(A)';
```

If this algorithm fails then one obtains the error message “Matrix must be positive definite”. So by the Cholesky Theorem above, this procedure yields a *practical test* of positive definiteness, which can be designated as the *Cholesky Test*. In summary, while Sylvester’s Condition provides a useful test for relatively small matrices, such as (A2.7.42), the calculation of principle minors is very time consuming for larger matrices. Here the Cholesky Test is much faster and more practical.<sup>12</sup> If the algorithm succeeds, then the matrix is SPD, and otherwise, it is not.<sup>13</sup>

### Calculation of Hessians

To see how these conditions can be applied in practice, it is instructive to analyze the maximization example in (A2.7.4) and (A2.7.5). While in this simple case, the desired Hessian can of course be calculated term by term (i.e., each cross partial derivative), for larger problem it is much more efficient to do the calculations in matrix terms. So it is appropriate to see how this can be accomplished. To do so, we begin by rewriting the gradient vector of first partial derivatives in expression (A2.6.4) in terms of our present notation as follows

<sup>12</sup> Even for  $n$ -square SPD matrices,  $A$ , as large as  $n = 1000$ , the MATLAB command, `chol(A)'`, produces the unique Cholesky decomposition in about 0.03 seconds.

<sup>13</sup> Care must be taken for “almost singular” SPD matrices, where rounding errors can sometimes lead to failure. Methods of numerical analysis must then be used to check whether this is the case.

$$(A2.7.58) \quad \nabla f(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{pmatrix} = \begin{pmatrix} f_1(x) \\ \vdots \\ f_n(x) \end{pmatrix}$$

This can be viewed as a *vector of functions*,  $f_i(x)$ ,  $i = 1, \dots, n$ . Notice that in the Hessian of (A2.7.33) the  $i$ -th column is just the gradient of the  $i$ -th function,  $f_i(x)$ , in (A2.7.58). So if we now define the gradient of a vector of smooth functions,  $[g(x), \dots, h(x)]'$  with common arguments,  $x = (x_1, \dots, x_n)$ , by

$$(A2.7.59) \quad \nabla \begin{pmatrix} g(x) \\ \vdots \\ h(x) \end{pmatrix} = [\nabla g(x), \dots, \nabla h(x)] = \begin{pmatrix} g_1(x) & \cdots & h_1(x) \\ \vdots & \ddots & \vdots \\ g_n(x) & \cdots & h_n(x) \end{pmatrix}$$

then the Hessian in (A2.7.33) is seen to be of the form,

$$(A2.7.60) \quad H_f(x_0) = \nabla \begin{pmatrix} f_1(x_0) \\ \vdots \\ f_n(x_0) \end{pmatrix} = \nabla[\nabla f(x_0)] = \nabla^2 f(x_0)$$

As a second application of (A2.7.59), note that if the  $i$ -th row of a matrix,  $A$ , is denoted by  $a'_i$ , then the linear expression,  $Ax$ , can be written as a vector of linear functions as follows,

$$(A2.7.61) \quad Ax = \begin{pmatrix} a'_1 \\ \vdots \\ a'_n \end{pmatrix} x = \begin{pmatrix} a'_1 x \\ \vdots \\ a'_n x \end{pmatrix}$$

so that by (A2.7.59) and (A2.7.7),

$$(A2.7.62) \quad \nabla(Ax) = \nabla \begin{pmatrix} a'_1 x \\ \vdots \\ a'_n x \end{pmatrix} = [\nabla(a'_1 x), \dots, \nabla(a'_n x)] = (a_1, \dots, a_n) = A'$$

With these preliminaries, we can now reconsider the maximization of the general quadratic expression in (A2.7.5),

$$(A2.7.63) \quad f(x) = c + b'x - x'Ax$$

with  $A$  assumed to be symmetric. Using (A2.7.58) through (A2.7.62), the Hessian matrix for this problem is now given by

$$\begin{aligned}
 \text{(A2.7.64)} \quad H_f(x) &= \nabla^2 f(x) = \nabla[\nabla(c + b'x - x'Ax)] \\
 &= \nabla(b - 2Ax) = 0 - 2\nabla(Ax) \\
 &= -2A
 \end{aligned}$$

Hence any point,  $x_0$ , satisfying the first-order condition for  $f(x)$  will be a maximum if and only if the matrix  $A$  is *positive definite* (so that the associated matrix,  $-2A$ , is *negative definite*). But for the specific maximum problem with parameters in (A2.7.4), we have already seen that the symmetrized matrix,  $A$ , in (A2.7.56) above is positive definite. Thus the unique point,  $x_0 = (4, 2)'$ , satisfying the first-order conditions is indeed a maximum (which was already evident in Figure A2.9).

Finally, it is important to reconsider the mean squared error function in (A2.7.19) above, where it was shown in (A2.7.20) that the unique weight vector satisfying the first-order conditions for minimization of

$$\text{(A2.7.65)} \quad MSE(\lambda_0) = \sigma^2 - 2c_0'\lambda_0 + \lambda_0'V_0\lambda_0$$

was given by  $\hat{\lambda}_0 = V_0^{-1}c_0$ . We are now in a position to complete that analysis. If the Hessian for this function is denoted by  $H_{MSE}$ , then by recalling that every covariance matrix is symmetric, it follows the same analysis in (A2.7.64) now yields

$$\begin{aligned}
 \text{(A2.7.66)} \quad H_{MSE}(\lambda_0) &= \nabla^2 MSE(\lambda_0) = \nabla[\nabla(\sigma^2 - 2c_0'\lambda_0 + \lambda_0'V_0\lambda_0)] \\
 &= \nabla(-2c_0 + 2V_0\lambda_0) = 0 + 2\nabla(V_0\lambda_0) = 2V_0
 \end{aligned}$$

Thus to ensure that  $\hat{\lambda}_0 = V_0^{-1}c_0$  is the unique minimum of (A2.6.65), it remains only to show that  $V_0$  is *positive definite*. In fact, it turns out that:

**Positive Definiteness Property.** *Every (nonsingular) covariance matrix is positive definite.*

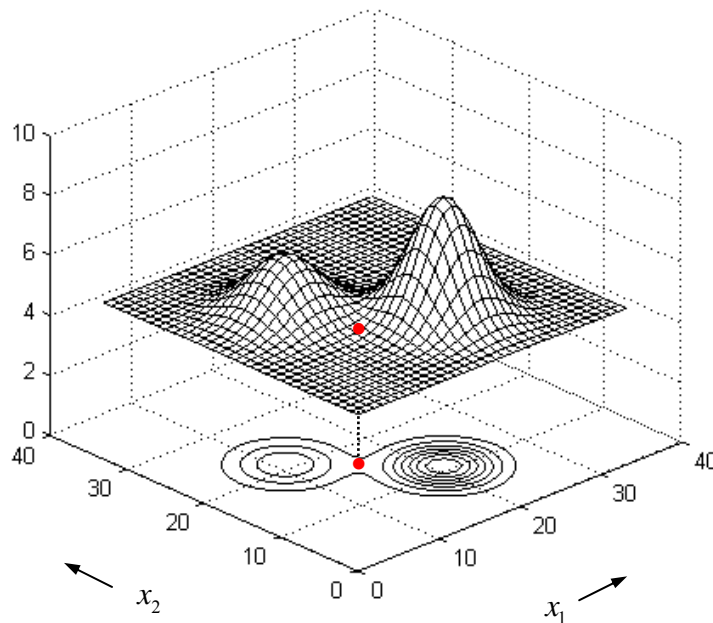
While we don't yet have all the tools to show this fully, we can establish the most essential part of this condition as follows. Recall from the covariance result in (3.2.21) that for any random vector,  $X$ , with covariance matrix,  $\Sigma = \text{cov}(X)$ , the variance of each linear compound,  $a'X$  is given by  $\text{var}(a'X) = a'\Sigma a$ . So it must certainly be true that

$$\text{(A2.7.67)} \quad a'\Sigma a \geq 0 \quad \text{for all } a \neq 0$$

This condition is called *positive semidefiniteness*, and must be exhibited by *every* covariance matrix. What remains to be shown is that for *nonsingular* covariance matrices the inequality in (A2.6.67) is *strict*. Since this is a simple consequence of the Spectral Decomposition Theorem (to be developed later), we postpone it for now.<sup>14</sup>

### Non-Definite Hessians

Before proceeding to the case of constrained optimization, it is of interest to ask whether one can have stationary points that are neither maxima or minima. An example for scalar functions was shown in Figure A2.8 above. But unlike this highly special case in one dimension, it turns out that such examples are quite common in higher dimensions. This is illustrated by the ( $n = 2$ ) example in Figure A2.10 below, where there exist two *local* maxima (the one on the right being the *global* maximum).



**Figure A2.11. Saddle Point Example**

However, there is seen to be a third point (shown in red) between these two local maxima which also satisfies the first-order condition that the gradient be zero. Notice also that movement from this point toward either maximum point must go “uphill”, so that second derivative is *positive* in these directions. But movement orthogonal to these directions leads “downhill” and hence yields *negative* second derivatives. At such *saddle point* locations, the Hessian is neither positive nor negative definite. Note finally

<sup>14</sup> This can actually be shown without the Spectral Decomposition Theorem. For a simple proof that positive semidefiniteness plus nonsingularity implies positive definiteness, see Horn and Johnson (1985, p.400)

that such saddle points are not rare. Indeed, whenever there are multiple maxima one can expect to find intermediate saddle points.

### A2.7.3 Application to Ordinary Least Squares Estimation

Before considering constrained optimization problems, we consider one final application of the above concepts, namely to the least squares estimation of  $\beta = (\beta_0, \beta_1, \dots, \beta_k)'$  in the classical linear regression model. Recall from (7.16) that the objective function is given by

$$(A2.7.68) \quad SSD(\beta) = y'y - 2y'X\beta + \beta'X'X\beta$$

and hence is seen to be a quadratic form very similar in nature to the mean squared error function,  $MSE(\lambda_0)$  in (A2.7.19) above. Thus, as in (A2.7.2), we see from the symmetry of the matrix  $X'X$  that the first-order condition for this minimization problem takes the form:

$$(A2.7.69) \quad 0 = \nabla SSD(\beta) = -2X'y + 2X'X\beta \Rightarrow X'X\beta = X'y$$

But if it is assumed that there are no collinearities between the columns of  $X$  (so that  $X$  is of full column rank) then the  $(k+1)$ -square matrix,  $X'X$ , is *nonsingular*. Hence the unique solution to (7.17), designated as the *ordinary least squares* (OLS) estimator of  $\beta$  is given by

$$(A2.7.70) \quad \hat{\beta} = (X'X)^{-1}X'y$$

The only question remaining is whether this yields a proper minimum. Here we can answer this question definitively. In particular, recall first from (A2.7.64) that in this case,

$$(A2.7.71) \quad H_{SSD}(\beta) = \nabla(-2X'y + 2X'X\beta) = 2X'X$$

so that it remains only to show that  $X'X$  is positive definite. But in the argument of (A2.7.37) through (A2.7.40) above it was shown that for any nonsingular matrix,  $B$ , the matrix  $B'B$  is necessarily positive definite. Hence it is enough to observe that this continues to hold as long as  $B$  is of full column rank. For if it were true that  $0 = x'(B'B)x = (Bx)'Bx$  for some  $x \neq 0$ , then the same argument shows that

$$(A2.7.72) \quad 0 = Bx = (b_1, \dots, b_m) \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = \sum_{j=1}^m x_j b_j$$

which together with  $x \neq 0$  implies the existence of a linear dependency (collinearity) among the columns  $(b_1, \dots, b_m)$  of  $B$ . Hence for any matrix of *full column rank*, such as  $X$ , it follows that  $X'X$  must be *positive definite*.

## A2.8 Constrained Optimization of Smooth Functions

As with the development of unconstrained optimization above, we shall be concerned here with those cases of constrained optimization that are relevant for the applications in the text. Hence we consider only linear equality constraints, where the optimum will again be seen to be characterized by appropriate “tangency” conditions.

To motivate the main ideas, we again begin with a two-dimensional example in which the relevant tangency conditions can be depicted graphically. For ease of visualization, it is convenient to switch to a minimization problem. So consider minimizing the quadratic objective function defined for each  $x = (x_1, x_2)'$  by,

$$(A2.8.1) \quad f(x) = c + b'x + x'Ax$$

with  $c = 20$ ,  $b' = (1, 2)$  and

$$(A2.8.2) \quad A = \begin{pmatrix} 25 & 1 \\ 1 & 15 \end{pmatrix}$$

As in (A2.7.18), this function has a unique stationary point,

$$(A2.8.3) \quad x^* = -\frac{1}{2}A^{-1}b = (-0.017, -0.066),$$

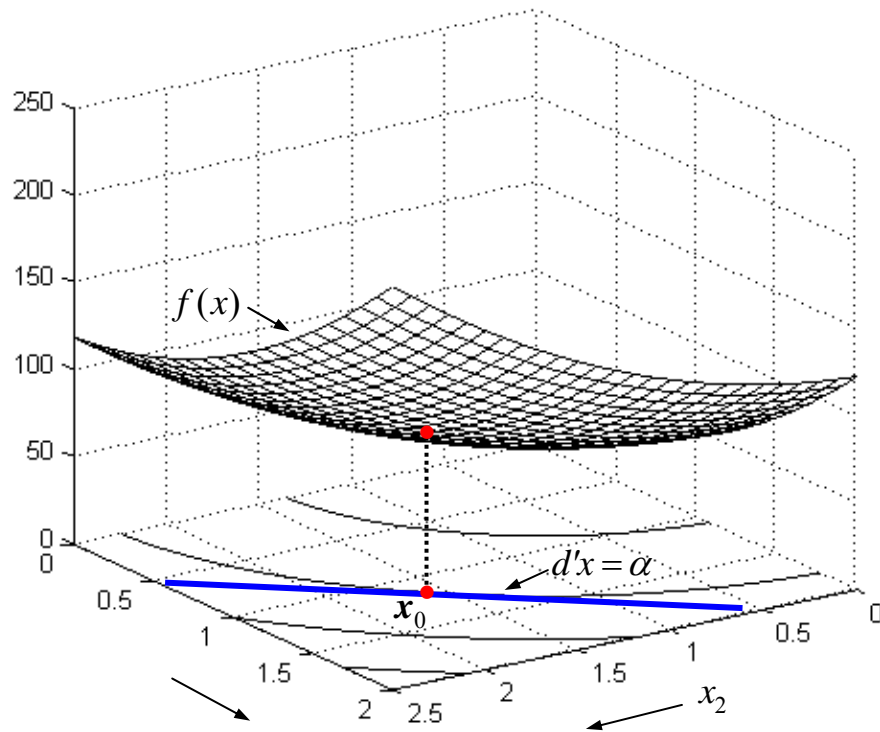
in the negative quadrant. Moreover, since  $A$  is seen by inspection to be *symmetric positive definite* [with its two leading principle minors,  $\det(25) = 25$  and  $\det(A) = 25 \cdot 15 - 1$ , both positive], it follows as in (A2.7.64) that  $H_f(x^*) = 2A$  is positive definite, and hence that  $x^*$  is a *global minimum*. This function is depicted in Figure A2.12 below [where again for visual convenience the origin (0,0) has been placed at the back corner of the figure]. The global minimum point,  $x^*$ , is out of view, since it is not the relevant minimum for our present purposes.

### A2.8.1 Minimization with a Single Constraint

In particular, we now suppose that feasible values of  $x$  for this minimization problem are also required to satisfy a *linear constraint* of the following form,

$$(A2.8.4) \quad d'x = \alpha$$

with  $d' = (5, 4)$  and  $\alpha = 13$ . In other words, the only relevant values of  $x$  for this problem are those lying on the blue line shown in Figure A2.12.



**Figure A2.12. Constrained Minimization Example**

To put this problem in more standard form, let the function  $g(x)$  be defined by

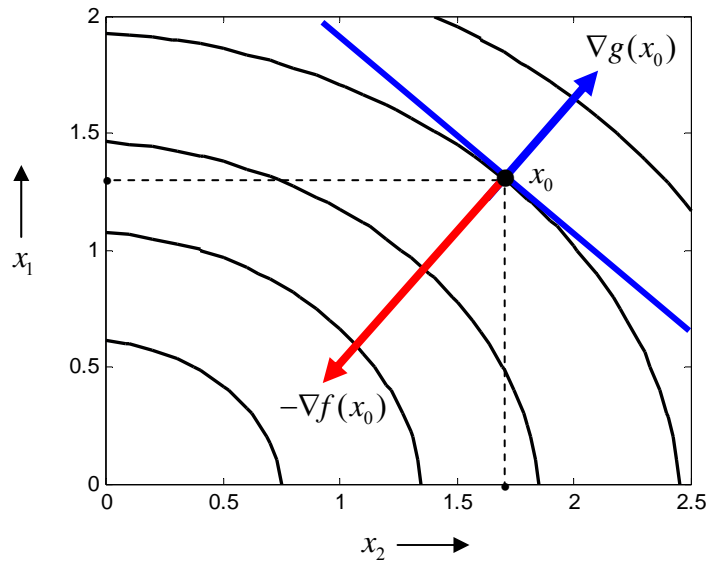
$$(A2.8.5) \quad g(x) = d'x$$

so that (A2.8.4) is equivalent to the condition that  $g(x) = \alpha$ . In these terms, the present problem is formally a *constrained minimization problem* of form,

$$(A2.8.6) \quad \text{minimize: } f(x) \quad \text{subject to: } g(x) = \alpha$$

To solve this problem, observe next that [in a manner similar to Figure A2.5 (and Figure A2.11) above] the contours of the function  $f(x)$  are shown on the  $(x_1, x_2)$  plane in Figure A2.12. Moreover, we know from (A2.8.3) above that this function decreases toward its global minimum,  $x^*$ , in the negative quadrant. So the lowest contour touching the blue line in Figure A2.12 clearly defines the desired *constrained minimum* point,  $x_0$ , solving problem (A2.8.6).

With these observations, the key question is how to identify this point analytically. Here it is convenient to give a planar representation of these contours as in Figure A2.13 below [where the  $(x_1, x_2)$  plane has now been rotated to place the origin in its more natural position at the lower left corner of the figure].<sup>15</sup>



**Figure A2.13 Tangency Condition for Constrained Minimum**

Here the solution point,  $x_0$ , is again identified by a tangency between the linear constraint,  $g(x) = \alpha$  (blue line), and the lowest contour of  $f(x)$ . But recall from (A2.6.3) that the gradient,  $\nabla f(x_0)$ , of  $f$  at  $x_0$  must be *orthogonal* to this tangent line, which by definition defines the directions of “no change” in  $f$  at  $x_0$ . Recall also that gradients point in the direction of maximum increase in  $f$ . But since we are here interested in *minimizing*  $f$  it is more appropriate to consider the (opposite) direction of maximum *decrease* in  $f$  at  $x_0$ , as given by the negative gradient,  $-\nabla f(x_0)$ . This is negative gradient is shown by the red arrow in Figure A2.13.

Similarly, since the blue tangent line is also a constant-value contour for the constraint function,  $g$  [i.e., the set of  $x$  values where  $g(x) = \alpha$ ], it then follows that the gradient,  $\nabla g(x_0)$ , of  $g$  at  $x_0$  must be *orthogonal* to this same tangent line, as shown by the blue arrow in Figure A2.13. [Since the positivity of the coefficient vector,  $d$ , in this case

<sup>15</sup> Note also that for compatibility with Figure A2.12, the horizontal axis is  $x_2$  rather than  $x_1$ .

implies that the function,  $g(x) = d'x$ , is increasing in  $x$ , this gradient points toward higher values  $x$ ].

Finally, since there is only a *single* line in the plane that is orthogonal to this blue line, it follows that the two gradients  $-\nabla f(x_0)$  and  $\nabla g(x_0)$  must both lie on this same line, i.e., must be *collinear*. Since this implies that  $-\nabla f(x_0)$  and  $\nabla g(x_0)$  must be scalar multiples of one another, the fundamental *tangency condition* in Figure A2.13 implies that for some scalar,  $\theta_0$ , it must be true that  $-\nabla f(x_0) = \theta_0 \nabla g(x_0)$ , or equivalently that

$$(A2.8.7) \quad \nabla f(x_0) + \theta_0 \nabla g(x_0) = 0$$

Algebraically, this two-dimensional tangency condition yields two equations in three unknown, namely  $x_0 = (x_{01}, x_{02})$  together with  $\theta_0$ . However, since  $x_0$  must lie on the blue line, it is also required that

$$(A2.8.8) \quad g(x_0) = \alpha$$

These equation system allows all unknowns to be solved for. But before doing so, it is important to note that while the above derivation is geometrical in nature, and hence can be illustrated graphically, there is a mathematically more powerful way of deriving the same conditions. In particular, if we now combine the functions,  $f$  and  $g$ , into a single function of the form

$$(A2.8.9) \quad L(x, \theta) = f(x) + \theta[g(x) - \alpha]$$

then this augmented function, called the *Lagrangian function*, actually yields conditions (A2.8.8) and (A2.8.9) as first-order conditions. In particular, if for any function,  $h(y, z)$  of vectors,  $y = (y_1, \dots, y_k)'$  and  $z = (z_1, \dots, z_m)'$ , we write the gradients of  $h$  with respect to  $y$  and  $z$  as,

$$(A2.8.10) \quad \nabla_y h(y, z) = \begin{pmatrix} \frac{\partial}{\partial y_1} h(y, z) \\ \vdots \\ \frac{\partial}{\partial y_k} h(y, z) \end{pmatrix} \quad \text{and} \quad \nabla_z h(y, z) = \begin{pmatrix} \frac{\partial}{\partial z_1} h(y, z) \\ \vdots \\ \frac{\partial}{\partial z_m} h(y, z) \end{pmatrix}$$

respectively, then it follows from (A2.8.9) that

$$(A2.8.11) \quad \nabla_x L(x, \theta) = \nabla f(x) + \theta \nabla g(x)$$

$$(A2.8.12) \quad \nabla_\theta L(x, \theta) = g(x) - \alpha$$

So (A2.8.7) and (A2.8.8) are seen to be precisely the first order conditions of  $L$  with respect to  $(x, \theta)$  evaluated at  $(x_0, \theta_0)$ , i.e.,

$$(A2.8.13) \quad 0 = \nabla_x L(x_0, \theta_0) = \nabla f(x_0) + \theta_0 \nabla g(x_0)$$

$$(A2.8.14) \quad 0 = \nabla_\theta L(x_0, \theta_0) = g(x_0) - \alpha$$

This is no coincidence, and in fact provides a general way of “converting” constrained optimization problems into larger-dimensional *unconstrained* problems. Here the original arguments,  $x$ , are augmented to  $(x, \theta)$ , where the dimension of  $\theta = (\theta_1, \dots, \theta_k)$  corresponds precisely to the number of constraints imposed on the optimization problem. These unknown scalars, known as *Lagrange multipliers*, play the same geometric role as in our one-constraint example above.

We shall consider a general Lagrangian problem of this type below. But for the present, it is instructive to complete the solution of our particular example. First, recall from expressions (A2.8.1) and (A2.8.5) that (A2.8.9) can be written more explicitly as follows:

$$(A2.8.15) \quad L(x, \theta) = (c + b'x + x'Ax) + \theta(d'x - \alpha)$$

Hence by employing the gradient identities in (A2.7.7) and (A2.7.11) together with (A2.8.11) and (A2.8.11), we see that (A2.8.13) and (A2.8.14) take the explicit form:

$$(A2.8.16) \quad 0 = \nabla_x L(x_0, \theta_0) = b + 2Ax_0 + \theta_0 d$$

$$(A2.8.17) \quad 0 = \nabla_\theta L(x_0, \theta_0) = d'x_0 - \alpha$$

But by the nonsingularity of  $A$  we can solve (A2.8.16) for  $x_0$  as follows:

$$(A2.8.18) \quad 2Ax_0 = -(\theta_0 d + b) \Rightarrow x_0 = -\frac{1}{2}A^{-1}(\theta_0 d + b)$$

Condition (A2.8.17) then yields the following explicit solution for  $\theta_0$ ,

$$(A2.8.19) \quad \alpha = d'x_0 = -\frac{1}{2}d'A^{-1}(\theta_0 d + b) \Rightarrow -2\alpha = \theta_0(d'A^{-1}d) + d'A^{-1}b$$

$$\Rightarrow \theta_0 = -\left(\frac{2\alpha + d'A^{-1}b}{d'A^{-1}d}\right)$$

Finally, substitution of (A2.8.19) into (A2.8.18) yields the following explicit solution for  $x_0$ :

$$(A2.8.20) \quad x_0 = \frac{1}{2} A^{-1} \left[ \left( \frac{2\alpha + d'A^{-1}b}{d'A^{-1}d} \right) d - b \right]$$

Substitution of the values,  $c = 20$ ,  $b' = (1, 2)$ ,  $\alpha = 13$ ,  $d' = (5, 4)$  together with  $A$  in (A2.8.2) yields the final solution

$$(A2.8.21) \quad x_0 = (1.2721, 1.6599)'$$

which is seen to correspond to the graphical solution shown in Figure A2.13.

### Solution for Ordinary Kriging

Finally, we apply these results to the case of ordinary kriging. Here we proceed in two steps. First we derive a *BLU estimator* for the unknown mean parameter,  $\mu$ , and then use this to interpret the solution to the optimal weight vector problem. Turning first to the BLU estimator for  $\mu$ , recall from expression (6.3.7) of the text that the optimal coefficient vector,  $\hat{a}$ , is given by the solution of the constrained minimization problem:

$$(A2.8.22) \quad \text{minimize: } a'Va \quad \text{subject to: } a'1_n = 1$$

This is seen to be a special case of the constrained minimization problem in (A2.8.15) with  $(c = 0, b = 0, A = V, \alpha = 1, d = 1_n)$ . Hence by setting  $x_0 = \hat{a}$  in (A2.8.20) and making these appropriate substitutions, it follows that the unique optimal coefficient vector is given by

$$(A2.8.23) \quad \hat{a} = \frac{1}{2} V^{-1} \left[ \left( \frac{2 + (0)}{1_n' V^{-1} 1_n} \right) 1_n - (0) \right] = \left( \frac{1}{1_n' V^{-1} 1_n} \right) V^{-1} 1_n$$

This in turn implies that the unique *BLU estimator*,  $\hat{\mu}_n$ , of  $\mu$  given sample vector  $Y$  is given by

$$(A2.8.24) \quad \hat{\mu}_n = \hat{a}'Y = \left( \frac{1}{1_n' V^{-1} 1_n} \right) 1_n' V^{-1} Y = \frac{1_n' V^{-1} Y}{1_n' V^{-1} 1_n}$$

Turning next to the problem of determining a *BLU predictor* of  $Y_0 = Y(s_0)$ , recall from expression (6.3.18) in the text that the desired weight vector,  $\hat{\lambda}_0$ , solves the constrained minimization problem:

$$(A2.8.25) \quad \text{minimize: } \sigma^2 - 2c_0' \lambda_0 + \lambda_0' V_0 \lambda_0 \quad \text{subject to: } 1_n' \lambda_0 = 1$$

But this is again a special case of the constrained minimization problem in (A2.8.15) with  $(c = \sigma^2, b = -2c_0, A = V_0, \alpha = 1, d = 1_n)$ . Hence by now setting  $x_0 = \hat{\lambda}_0$  in (A2.8.20), it follows that

$$(A2.8.26) \quad \hat{\lambda}_0 = \frac{1}{2}V_0^{-1} \left[ \left( \frac{2 - 2 \mathbf{1}'_{n_0} V_0^{-1} c_0}{\mathbf{1}'_{n_0} V_0^{-1} \mathbf{1}_{n_0}} \right) \mathbf{1}_{n_0} + 2c_0 \right]$$

$$= \left( \frac{1 - \mathbf{1}'_{n_0} V_0^{-1} c_0}{\mathbf{1}'_{n_0} V_0^{-1} \mathbf{1}_{n_0}} \right) V_0^{-1} \mathbf{1}_{n_0} + V_0^{-1} c_0$$

Hence the desired *BLU predictor* of  $Y_0$  is given by

$$(A2.8.27) \quad \hat{Y}_0 = \hat{\lambda}'_0 Y = \left( \frac{1 - \mathbf{1}'_{n_0} V_0^{-1} c_0}{\mathbf{1}'_{n_0} V_0^{-1} \mathbf{1}_{n_0}} \right) \mathbf{1}'_{n_0} V_0^{-1} Y + c'_0 V_0^{-1} Y$$

For purposes of interpreting this expression, observe that since  $\mathbf{1}'_{n_0} V_0^{-1} c_0 = c'_0 V_0^{-1} \mathbf{1}_{n_0}$ , we may rewrite (A2.8.27) as

$$(A2.8.28) \quad \hat{Y}_0 = \left( \frac{\mathbf{1}'_{n_0} V_0^{-1} Y}{\mathbf{1}'_{n_0} V_0^{-1} \mathbf{1}_{n_0}} \right) + c'_0 V_0^{-1} Y - c'_0 V_0^{-1} \mathbf{1}_{n_0} \left( \frac{\mathbf{1}'_{n_0} V_0^{-1} Y}{\mathbf{1}'_{n_0} V_0^{-1} \mathbf{1}_{n_0}} \right)$$

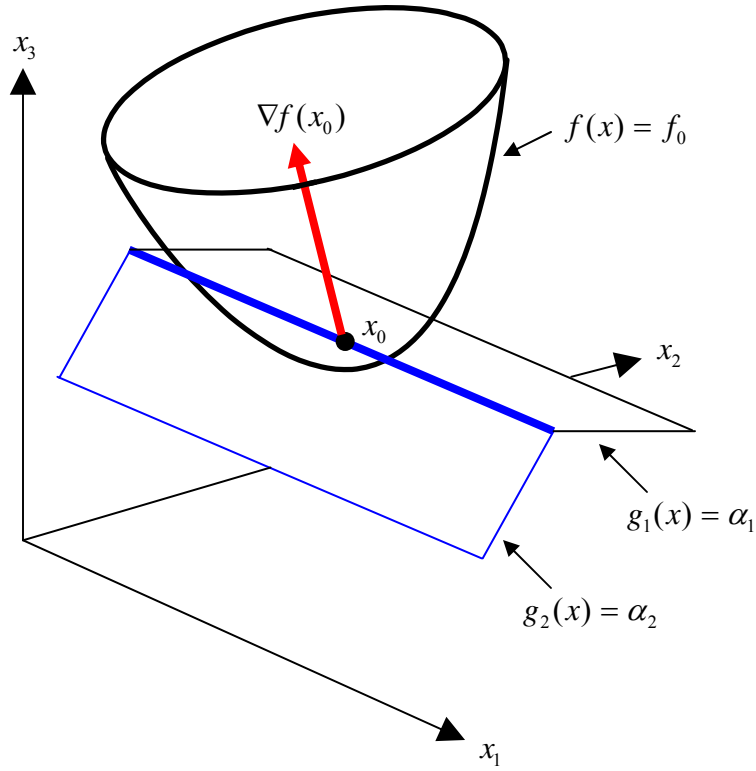
By using (A2.8.24), this expression may then be simplified, as is done in expression (6.3.21) of the text.

### A2.8.2 Minimization with Multiple Constraints

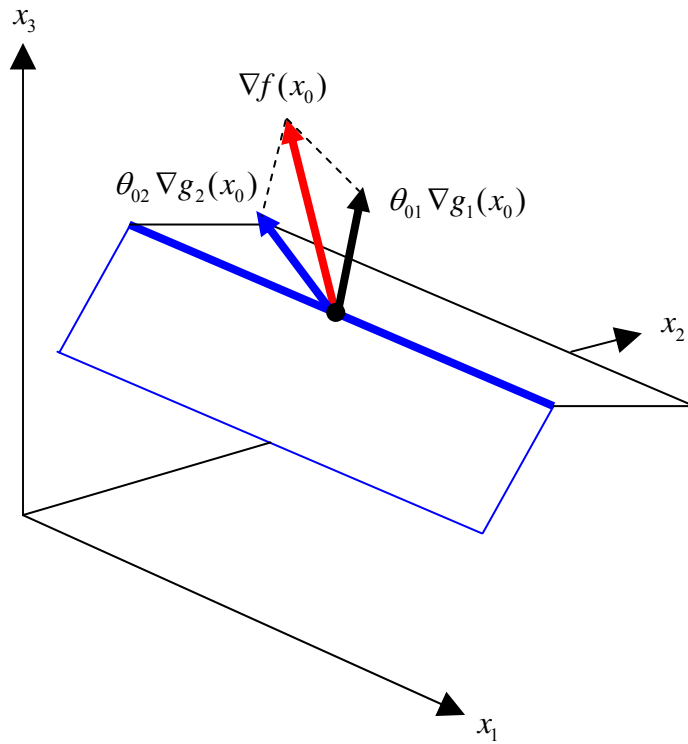
Given the results above for a single constraint, we now proceed to the case of multiple constraints. For purposes of illustration we begin with the case of two (linear) constraints on functions of three variables,  $f(x) = f(x_1, x_2, x_3)$ , where is still possible to obtain some geometric intuition. As an extension of (A2.8.6) we thus consider the following constrained minimization problem:

$$(A2.8.29) \quad \text{minimize: } f(x) \quad \text{subject to: } \begin{pmatrix} g_1(x) \\ g_2(x) \end{pmatrix} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}$$

where  $x = (x_1, x_2, x_3) \in \mathbb{R}^3$ . Paralleling Figures A2.12 and A2.13 above, the solution conditions for this problem are shown schematically in Figures A2.14 and A2.15 below.



**Figure A2.14. Constrained Tangency Condition**



**Figure A2.15. Constrained Gradient Condition**

To compare these figures with the single constraint case above, we start by restricting attention to the  $x$ -space in Figure A2.12, i.e., the  $(x_1, x_2)$ -plane. Recall that the single linear constraint corresponds to the blue line in this plane, and the critical tangency condition for a minimum is shown in terms of the contour representation of  $f(x)$  on this plane. The situation in Figure A2.14 is conceptually the same, except that the  $x$ -space is now *three* dimensional. Here the two linear constraints,  $g_1(x) = \alpha_1$  and  $g_2(x) = \alpha_2$ , are shown, respectively, by the blue and black *planes* in this space. Note that these planes constitute constant-value *contour surfaces* for the functions  $g_1$  and  $g_2$ . Hence, like Figure A2.12, the *constraint space* defined by the intersection of these two planes is again one dimensional, as shown by the heavy blue line. With respect to the objective function,  $f(x) = f(x_1, x_2, x_3)$ , constant-value *contour surfaces* in this space are curvilinear. Hence for visual clarity, only the single contour surface,  $f(x) = f(x_0) = f_0$ , *tangent* to the constraint space at point  $x_0$  is shown. As in Figure A2.13, the negative gradient vector,  $-\nabla f(x_0)$ , at  $x_0$  must be orthogonal to the constraint space, as shown by the red arrows in both Figures A2.13 and A2.14. So the tangency conditions in these two cases are seen to be conceptually the same.

Turning next to the relation between this gradient vector and those for the constraints recall that in Figure A2.13 the single gradient vector,  $\nabla g(x_0)$ , was also orthogonal to the constraint space as defined by a constant-value contour of  $g$ . Moreover, since all vectors orthogonal to this constraint line at  $x_0$  must necessarily be *collinear*, this in turn implied that  $-\nabla f(x_0)$  must be a scalar multiple of  $\nabla g(x_0)$ . But in higher dimensions this is no longer true. In the present case, the set of vectors orthogonal to the blue line at  $x_0$  must define a *plane* (not shown) which is called the *orthogonal complement* of this line at  $x_0$ . So all that can be said is that these three gradient vectors,  $-\nabla f(x_0)$ ,  $\nabla g_1(x_0)$  and  $\nabla g_2(x_0)$ , must all lie in this plane. But assuming that the two constraint planes [ $g_1(x) = \alpha_1$  and  $g_2(x) = \alpha_2$ ] have a well-defined linear intersection (and hence are not parallel), it follows that  $\nabla g_1(x_0)$  and  $\nabla g_2(x_0)$  cannot themselves be collinear. Hence they must *span* this plane, which means that every vector in the plane can be written as a *unique linear combination* of  $\nabla g_1(x_0)$  and  $\nabla g_2(x_0)$ . In particular this implies that for the negative gradient vector,  $-\nabla f(x_0)$ , there must exist unique scalars,  $\theta_{01}$  and  $\theta_{02}$ , such that  $-\nabla f(x_0) = \theta_{01} \nabla g_1(x_0) + \theta_{02} \nabla g_2(x_0)$ , or equivalently,

$$(A2.8.30) \quad \nabla f(x_0) + \theta_{01} \nabla g_1(x_0) + \theta_{02} \nabla g_2(x_0) = 0$$

as shown in Figure A2.15. This is the fundamental *constrained gradient condition* that generalizes (A2.8.7) for the single-constraint case. Hence, as an extension of (A2.8.9), if we now consider the Lagrangian function:

$$(A2.8.31) \quad L(x, \theta_1, \theta_2) = f(x) + \theta_1 [g_1(x) - \alpha_1] + \theta_2 [g_2(x) - \alpha_2]$$

with first-order conditions

$$(A2.8.32) \quad 0 = \nabla_x L(x_0, \theta_{01}, \theta_{02}) = \nabla f(x_0) + \theta_{01} \nabla g_1(x_0) + \theta_{02} \nabla g_2(x_0)$$

$$(A2.8.33) \quad 0 = \nabla_{\theta_1} L(x_0, \theta_{01}, \theta_{02}) = g_1(x_0) - \alpha_1$$

$$(A2.8.34) \quad 0 = \nabla_{\theta_2} L(x_0, \theta_{01}, \theta_{02}) = g_2(x_0) - \alpha_2$$

then it is clear that the minimum for this function satisfies both the constrained gradient condition in (A2.8.30) together with the two constraints in (A2.8.29).

The extension of this programming problem to objective functions,  $f(x) = f(x_1, \dots, x_n)$ , in  $n$  dimensions with  $k$  equality constraints is a straightforward generalization of the geometric representations in Figures A2.14 and A2.15. In particular, if for any  $k$ -vector of constraint functions,

$$(A2.8.35) \quad G(x) = \begin{bmatrix} g_1(x) \\ \vdots \\ g_k(x) \end{bmatrix}, \quad x = (x_1, \dots, x_n)' \in \mathbb{R}^n$$

(with  $k < n$ ) and corresponding constants,  $\alpha = (\alpha_1, \dots, \alpha_k)'$  we consider the *constrained minimization problem*:

$$(A2.8.36) \quad \text{minimize: } f(x) \quad \text{subject to: } G(x) = \alpha$$

then letting  $\theta = (\theta_1, \dots, \theta_k)'$  denote a vector of *Lagrange multipliers*, we may again form the corresponding *Lagrangian function*,

$$(A2.8.37) \quad \begin{aligned} L(x, \theta) &= f(x) + \sum_{j=1}^k \theta_j [g_j(x) - \alpha_j] \\ &= f(x) + \theta' [G(x) - \alpha] \end{aligned}$$

Hence by employing (A2.7.58), (A2.7.59) and (A2.8.10), it follows that a minimizing pair,  $(x_0, \theta_0)$ , is now characterized by the first-order conditions:

$$(A2.8.38) \quad \begin{aligned} 0 &= \nabla_x L(x_0, \theta_0) = \nabla f(x_0) + \sum_{j=1}^k \theta_{0j} \nabla g_j(x_0) \\ &= \nabla f(x_0) + [\nabla g_1(x_0), \dots, \nabla g_k(x_0)] \begin{pmatrix} \theta_{01} \\ \vdots \\ \theta_{0k} \end{pmatrix} \\ &= \nabla f(x_0) + \nabla G(x_0) \theta_0 \end{aligned}$$

and,

$$(A2.8.39) \quad 0 = \nabla_{\theta} L(x_0, \theta_0) = G(x_0) - \alpha$$

In terms of Figures A2.14 and A2.15, condition (A2.8.38) again reflects the constrained gradient condition that the negative gradient,  $-\nabla f(x_0)$ , be a linear combination of the constraint gradients. As a generalization of the constraint space in these figures (with dimension  $3 - 2 = 1$ ), it is implicitly assumed here that the relevant constraint set (i.e., the intersection of  $k$  constraint surfaces) is a well defined surface of dimension  $n - k$ , so that the orthogonal complement to this surface at  $x_0$  has dimension  $k$ . If so, then the constraint gradients must span this complement, so that (A2.8.38) must hold for some unique vector of multipliers,  $\theta_0 = (\theta_{01}, \dots, \theta_{0k})'$ .

Our objective is to apply this general formulation to the case of *quadratic* objective functions

$$(A2.8.40) \quad f(x) = c + b'x + x'Ax$$

on  $\mathbb{R}^n$  with *linear* constraints,

$$(A2.8.41) \quad Dx = \begin{bmatrix} d_1'x \\ \vdots \\ d_k'x \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix} = \alpha$$

where the above constrained gradient condition is guaranteed to hold as long as these  $k$  constraints are linearly independent (i.e.,  $D$  is of full row rank,  $k$ ). Here the minimization problem in (A2.8.36) takes the form:

$$(A2.8.42) \quad \text{minimize: } c + b'x + x'Ax \quad \text{subject to: } Dx = \alpha$$

with associated Lagrangian in (A2.8.37) of the form

$$(A2.8.43) \quad L(x, \theta) = [c + b'x + x'Ax] + \theta'(Dx - \alpha)$$

Assuming that  $A$  is symmetric positive definite, this problem always has a unique solution,  $(x_0, \theta_0)$ , which is characterized by the first-order conditions,

$$(A2.8.44) \quad 0 = \nabla_x L(x_0, \theta_0) = [b + 2Ax_0] + D'\theta_0$$

$$(A2.8.45) \quad 0 = \nabla_{\theta} L(x_0, \theta_0) = Dx - \alpha$$

which are seen to reduce precisely to (A2.8.16) and (A2.8.17) for the case of a single constraint. Hence the solution is quite similar. Again we start by using the nonsingularity of  $A$  to solve for  $x_0$  in (A2.8.44) as

$$(A2.8.46) \quad 2Ax_0 = -(D'\theta_0 + b) \Rightarrow x_0 = -\frac{1}{2}A^{-1}(D'\theta_0 + b) ,$$

and then use (A2.8.46) to solve for  $\theta_0$ :

$$(A2.8.47) \quad \alpha = Dx_0 = -\frac{1}{2}DA^{-1}(D'\theta_0 + b) \Rightarrow -2\alpha = (DA^{-1}D')\theta_0 + DA^{-1}b \\ \Rightarrow \theta_0 = -(DA^{-1}D')^{-1}(DA^{-1}b + 2\alpha)$$

Substitution of (A2.8.47) into (A2.8.46) then yields the following solution for  $x_0$ :

$$(A2.8.48) \quad x_0 = \frac{1}{2}A^{-1} \left[ D'(DA^{-1}D')^{-1}(DA^{-1}b + 2\alpha) - b \right]$$

### A2.8.3 Solution for Universal Kriging

We now apply these results to the case of Universal Kriging. As with Ordinary Kriging above, we proceed in two steps. Given the linear model

$$(A2.8.49) \quad Y = X\beta + \varepsilon , \quad \varepsilon \sim N(0, V)$$

we first determine the unique *BLU estimator* of  $\beta$ , and then use this to interpret the solution of the optimal weight vector problem. But in this case, the first step is of major interest in itself, and in fact yields an important characterization of Generalized Least Squares estimation.

#### Best Linear Unbiased Estimation of $\beta$

Here we proceed to show that the GLS estimator for  $\beta$  as developed in Section 7.1.2 of the text is a BLU estimator as defined there. Moreover, since this argument is required to hold for all possible linear compounds,  $a \in \mathbb{R}^{k+1}$ , it suffices to pick a representative compound,  $a$ , and consider the problem of finding that estimator of  $\beta$  in the set of linear unbiased estimators,

$$(A2.8.50) \quad LU_a(\beta) = \{ \tilde{\beta} = \tilde{\beta}(X, V, Y) : [a'\tilde{\beta} = \theta'Y] \& [E(a'\tilde{\beta}) = a'\beta] \}$$

with smallest variance. The solution to this problem will show that this estimator is always given by the *GLS estimator*,

$$(A2.8.51) \quad \hat{\beta} = (X'V^{-1}X)^{-1}X'V^{-1}Y$$

To do so we can construct the appropriate constrained minimization problem as follows.<sup>16</sup> If we choose any estimator,  $\tilde{\beta} \in LU_a(\beta)$ , then linearity require that for some weight vector,  $\tilde{\theta}$  [which may depend on  $(a, X, V)$ ] we must have

$$(A2.8.52) \quad a' \tilde{\beta} = \tilde{\theta}' Y$$

Moreover, the *unbiased condition* requires that

$$(A2.8.53) \quad a' \beta = E(a' \tilde{\beta}) = E(\tilde{\theta}' Y) = \tilde{\theta}' E(Y) = \tilde{\theta}' X \beta$$

But this can only hold for all possible values of  $\beta$  if  $a' = \tilde{\theta}' X$ , or equivalently,

$$(A2.8.54) \quad X' \tilde{\theta} = a$$

Moreover, since the variance of  $\tilde{\theta}' Y$  is given by

$$(A2.8.55) \quad \text{var}(\tilde{\theta}' Y) = \tilde{\theta}' \text{cov}(Y) \tilde{\theta} = \tilde{\theta}' \text{cov}(\varepsilon) \tilde{\theta} = \tilde{\theta}' V \tilde{\theta}$$

it follows that weight vector,  $\tilde{\theta}$ , of the desired BLU estimator must solve the constrained minimization problem:

$$(A2.8.56) \quad \text{minimize: } \tilde{\theta}' V \tilde{\theta} \quad \text{subject to: } X' \tilde{\theta} = a$$

But since this is the special case of (A2.8.42) with  $(c = 0, b = 0, A = V, D = X', \alpha = a)$ , it follows from (A2.8.48) that optimum values of  $\tilde{\theta}$  for compound  $a$  is given by

$$(A2.8.57) \quad \tilde{\theta}_a = \frac{1}{2} V^{-1} [X(X' V^{-1} X)^{-1} (0 + 2a) - 0] = V^{-1} X (X' V^{-1} X)^{-1} a$$

and hence that the corresponding linear estimator in (A2.8.50), say  $\tilde{\beta}_a$  satisfies

$$(A2.8.58) \quad a' \tilde{\beta}_a = \tilde{\theta}' Y = a' (X' V^{-1} X)^{-1} X' V^{-1} Y = a' \hat{\beta}$$

Finally, since this holds identically for *all* linear compounds,  $a$ , we see that the unique estimator satisfying all these conditions is given precisely by the GLS estimator. To make this precise, observe that by setting  $a$  equal to the  $i^{\text{th}}$  column,  $e_i$ , of  $I_{k+1}$  for each  $i = 1, \dots, k+1$  [as in (3,2,16) of the text], it must follow from (A2.8.58) that

$$(A2.8.59) \quad (\tilde{\beta}_{e_i})_i = e_i' \tilde{\beta}_{e_i} = e_i' \hat{\beta} = \hat{\beta}_i, \quad i = 1, \dots, k+1$$

<sup>16</sup> Our present approach is based on the development in Searle (1971, Section 3.3.d).

and hence that all components of  $\hat{\beta}$  are uniquely identified by these particular choices of  $a$ .

Finally, it should be noted that this result is usually referred to as the *Gauss-Markov Theorem* in the literature.<sup>17</sup> The above constrained minimization approach thus yields a constructive proof of this theorem.

### Best Linear Unbiased Prediction of $Y(s_0)$

Next we derive the solution of the constrained minimization problem for Universal Kriging in expression (7.2.12) of the text:

$$(A2.8.60) \quad \text{minimize: } \sigma^2 - 2c'_0 \lambda_0 + \lambda'_0 V_0 \lambda_0 \quad \text{subject to: } X'_0 \lambda_0 = x_0$$

Since this is now seen to be an instance of the general constrained minimization problem (A2.8.42) with  $(c = \sigma^2, b = -2c_0, A = V_0, D = X'_0, \alpha = x_0)$ , it follows from (A2.8.48) that

$$(A2.8.61) \quad \hat{\lambda}_0 = \frac{1}{2} V_0^{-1} \left[ X_0 (X'_0 V_0^{-1} X_0)^{-1} (-2X'_0 V_0^{-1} c_0 + 2x_0) + 2c_0 \right] \\ = V_0^{-1} X_0 (X'_0 V_0^{-1} X_0)^{-1} (x_0 - X'_0 V_0^{-1} c_0) + V_0^{-1} c_0$$

Hence the *BLU predictor* of  $Y_0$  is given by

$$(A2.8.62) \quad \hat{Y}_0 = \hat{\lambda}'_0 Y = (x_0 - X'_0 V_0^{-1} c_0)' (X'_0 V_0^{-1} X_0)^{-1} X'_0 V_0^{-1} Y + c'_0 V_0^{-1} Y \\ = (x_0 - X'_0 V_0^{-1} c_0)' (X'_0 V_0^{-1} X_0)^{-1} X'_0 V_0^{-1} Y + c'_0 V_0^{-1} Y \\ = (x'_0 - c'_0 V_0^{-1} X_0) (X'_0 V_0^{-1} X_0)^{-1} X'_0 V_0^{-1} Y + c'_0 V_0^{-1} Y \\ = x'_0 (X'_0 V_0^{-1} X_0)^{-1} X'_0 V_0^{-1} Y + c'_0 V_0^{-1} [Y - X_0 (X'_0 V_0^{-1} X_0)^{-1} X'_0 V_0^{-1} Y]$$

### Standard Error of Prediction

Finally, to determine the prediction error variance for Universal Kriging, one must substitute  $\hat{\lambda}_0$  into the general expression for the prediction error variance [as given by the objective function in (8.2.60)], to obtain:

<sup>17</sup> See for example Section 4.4 in Green (2003).

$$(A2.8.63) \quad \hat{\sigma}_0^2 = \text{var}(e_0) = \sigma^2 - 2c_0' \hat{\lambda}_0 + \hat{\lambda}_0' V_0 \hat{\lambda}_0$$

To evaluate  $\hat{\sigma}_0^2$ , it is convenient to simplify the expression for  $\hat{\lambda}_0$  in (A2.8.61) as follows. If we now let

$$(A2.8.63) \quad \Sigma_0 = (X_0' V_0^{-1} X_0)^{-1}, \text{ and}$$

$$(A2.8.64) \quad \alpha_0 = x_0 - X_0' V_0^{-1} c_0,$$

then  $\hat{\lambda}_0$  can be written as

$$(A2.8.65) \quad \hat{\lambda}_0 = V_0^{-1} X_0 \Sigma_0 \alpha_0 + V_0^{-1} c_0$$

Then second term in (A2.8.63) becomes

$$(A2.8.66) \quad -2c_0' \hat{\lambda}_0 = -2c_0' [V_0^{-1} X_0 \Sigma_0 \alpha_0 + V_0^{-1} c_0] = -2c_0' V_0^{-1} X_0 \Sigma_0 \alpha_0 - 2c_0' V_0^{-1} c_0$$

and the third term becomes

$$\begin{aligned} (A2.8.67) \quad \hat{\lambda}_0' V_0 \hat{\lambda}_0 &= (V_0^{-1} X_0 \Sigma_0 \alpha_0 + V_0^{-1} c_0)' V_0 (V_0^{-1} X_0 \Sigma_0 \alpha_0 + V_0^{-1} c_0) \\ &= (V_0^{-1} X_0 \Sigma_0 \alpha_0 + V_0^{-1} c_0)' (X_0 \Sigma_0 \alpha_0 + c_0) \\ &= (\alpha_0' \Sigma_0 X_0' V_0^{-1} + c_0' V_0^{-1}) (X_0 \Sigma_0 \alpha_0 + c_0) \\ &= \alpha_0' \Sigma_0 (X_0' V_0^{-1} X_0) \Sigma_0 \alpha_0 + \alpha_0' \Sigma_0 X_0' V_0^{-1} c_0 + c_0' V_0^{-1} X_0 \Sigma_0 \alpha_0 + c_0' V_0^{-1} c_0 \end{aligned}$$

But since the two center terms are the same, and since  $(X_0' V_0^{-1} X_0) \Sigma_0 = I_{n_0}$  by (A2.8.63), we see that,

$$(A2.8.68) \quad \hat{\lambda}_0' V_0 \hat{\lambda}_0 = \alpha_0' \Sigma_0 \alpha_0 + 2\alpha_0' \Sigma_0 X_0' V_0^{-1} c_0 + c_0' V_0^{-1} c_0$$

Finally, by substituting (A2.8.66) and (A2.8.67) into (A2.8.63) and cancelling terms, we obtain an explicit expression for *prediction error variance*:

$$\begin{aligned} (A2.8.69) \quad \hat{\sigma}_0^2 &= \sigma^2 - c_0' V_0^{-1} c_0 + \alpha_0' \Sigma_0 \alpha_0 \\ &= \sigma^2 - c_0' V_0^{-1} c_0 + (x_0 - X_0' V_0^{-1} c_0)' (X_0' V_0^{-1} X_0)^{-1} (x_0 - X_0' V_0^{-1} c_0) \end{aligned}$$

In addition, since it is clear from a comparison of (A2.8.25) and (A2.8.60) that Ordinary Kriging is simply the special case of Universal Kriging in which  $x_0 = 1$  and  $X_0 = 1_{n_0}$ , it follows (A2.8.68) that *prediction error variance* for Ordinary Kriging is given by

$$\begin{aligned}
 \text{(A2.8.70)} \quad \hat{\sigma}_0^2 &= \sigma^2 - c_0' V_0^{-1} c_0 + (1 - 1_{n_0}' V_0^{-1} 1_{n_0})' (1_{n_0}' V_0^{-1} 1_{n_0})^{-1} (1 - 1_{n_0}' V_0^{-1} 1_{n_0}) \\
 &= \sigma^2 - c_0' V_0^{-1} c_0 + \frac{(1 - 1_{n_0}' V_0^{-1} 1_{n_0})^2}{1_{n_0}' V_0^{-1} 1_{n_0}}
 \end{aligned}$$