Real-time Generalized Least Squares

Bill Menke, June 7, 2022

We imagine a generalized least squares (GLS) problem with space and time variability. The spatial part of the solution has size, \( M \), which is repeated for \( K + 1 \) time steps. The size of the problem grows with time, but we are most interested in the current ("last") time, \( K + 1 \). The Gram matrix of this problem has the form:

\[
\begin{bmatrix}
A & C^T \\
C & B
\end{bmatrix}
\begin{bmatrix}
m_1 \\
m_2
\end{bmatrix} =
\begin{bmatrix}
a_1 \\
a_2
\end{bmatrix}
\]

Here, the \( M \times M \) submatrix, \( B \), the \( M \times 1 \) right hand side (r.h.s) vector, \( a_2 \) and the \( M \times 1 \) solution vector, \( m_2 \), is associated with the final, \( K + 1 \), time step. The \( MK \times MK \) submatrix, \( A \), the \( MK \times 1 \) r.h.s. vector, \( a_1 \), and the \( MK \times 1 \) solution, \( m_1 \), is associated with the previous \( K \) time steps. The submatrix, \( C \) is \( M \times MK \).

The bordering method is used to construct the inverse of the Gram matrix. We start with the definition of the matrix inverse:

\[
\begin{bmatrix}
D & F^T \\
F & E
\end{bmatrix}
\begin{bmatrix}
A & C^T \\
C & B
\end{bmatrix} =
\begin{bmatrix}
I & 0 \\
0 & I
\end{bmatrix}
\]

We then multiply out this equation and solve for the initially unknown submatrices, \( D \), \( E \), and \( F \):

\[
DA + F^TC = I \quad \text{so} \quad D = [I - F^TC]A^{-1}
\]

\[
DC^T + F^TB = 0 \quad \text{so} \quad F^T = -DC^TB^{-1}
\]

\[
FA + EC = 0 \quad \text{so} \quad F = -ECA^{-1}
\]

\[
FC^T + EB = I \quad \text{so} \quad -ECA^{-1}C^T + EB = I \quad \text{so} \quad E = [B - CA^{-1}C^T]^{-1}
\]

\[
F = -ECA^{-1} = -[B - CA^{-1}C^T]^{-1}CA^{-1}
\]

\[
D = [I - F^TC]A^{-1} = [I + A^{-1}C^T]ECA^{-1} = [I + A^{-1}C^T][B - CA^{-1}C^T]^{-1}CA^{-1} = ZA^{-1}
\]

\[
D = ZA^{-1} \quad \text{with} \quad Z \equiv [I + A^{-1}C^T][B - CA^{-1}C^T]^{-1}C
\]

I have tested these formulas numerically; they give the matrix inverse to within machine precision. The solution of the GLS problem is:

\[
\begin{bmatrix}
m_1 \\
m_2
\end{bmatrix} =
\begin{bmatrix}
D & F^T \\
F & E
\end{bmatrix}^{-1}
\begin{bmatrix}
a_1 \\
a_2
\end{bmatrix}
\]

We now develop an implementation of this formula, presuming that the solution, \( \hat{m}_1 = A^{-1}a_1 \) for the previous time steps is known. However, we do not want to explicitly construct any matrix inverses (and especially not \( A^{-1} \)) because they may not be sparse. Instead, we want to solve linear equations, say via the conjugate gradient method. We start by defining variables \( U \), \( Y \), and \( \hat{m}_2 \), all of which can be calculated by solving linear equations:

the \( MK \times M \) matrix \( U \): \( A^{-1}C^T \equiv U \quad \text{or} \quad AU = C^T \)

the \( M \times MK \) matrix \( Y \): \( [B - CU]^{-1}C \equiv Y \quad \text{or} \quad [B - CU]Y = C \)

The size of the problem grows with time, we are most interested in the current ("last") time, \( K + 1 \). We start with the definition of the matrix inverse:
the $M \times 1$ vector $\hat{m}_2$: $B^{-1}a_2 \equiv \hat{m}_2$ or $B\hat{m}_2 = a_2$

These definitions imply:

$$E = [B - CU]^{-1}$$
$$Z = [I + UY]$$
$$D = [I + UY]A^{-1}$$

$$F^T a_2 = -DC^T B^{-1}a_2 = -DC^T \hat{m}_2$$

Then, we write

$$m_1 = Da_1 + F^T a_2 = [I + UY][\hat{m}_1 - U\hat{m}_2]$$

$$m_2 = Fa_1 + Ea_2 = E[a_2 - C\hat{m}_1]$$

or $[B - CU]m_2 = a_2 - C\hat{m}_1$

I have tested these formulas numerically. The recursion for the solution is then:

Step 1:

Set $K = 1$ and solve $A\hat{m}_1 = a_1$ for $\hat{m}_1$.

Step 2:

Increment $K$

Assign the submatrices $B$ and $C$

Solve for $U$, $Y$, and $\hat{m}_2$

Solve for $m_1$ and $m_2$, the latter is the real-time solution

Step 3:

$$\begin{bmatrix} m_1 \\ m_2 \end{bmatrix}$$ becomes $\hat{m}_1$

$$\begin{bmatrix} A & C^T \\ C & B \end{bmatrix}$$ becomes $A$

Go to Step 2

The down side is that the effort needed to compute $U$ and $Y$ grows with time.

This procedure can be considered a generalization of Kalman filtering.