Overview and Project Description

This laboratory assignment examines the problem of interpolation from the viewpoint of least square signal estimation. Deterministic estimation (or optimal recovery) theory, as given by Golomb and Weinberger [1], allows the estimation of any linear functional of a signal, given some set of known linear functionals of the signal and the signal class of which the signal is a member. The theory also allows for the determination of error bounds. Interpolation is a special case of linear functional estimation, with known samples as the known linear functionals. We will consider the problem of optimal interpolation of signals that are members of a given filter class and have finite energy.

In this assignment, we obtain an optimal estimate of a signal with missing samples. We assume that the signal, \( u \), belongs to an averaging filter class such that

\[
u = Ca
\]

where \( C \) is a linear filtering operation with a length \( L \) impulse response \( h \) and \( a \) is an arbitrary finite length signal with a suitable bound on the norm. Thus

\[
\begin{bmatrix}
u_1 \\
u_2 \\vdots \\\nu_N
\end{bmatrix} = \begin{bmatrix}
h_L & h_{L-1} & \cdots & h_1 & 0 & \cdots & 0 \\
0 & h_L & \cdots & h_2 & h_1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & h_L & h_{L-1} & \cdots & h_1
\end{bmatrix} \begin{bmatrix}
a_1 \\
a_2 \\vdots \\
a_N
\end{bmatrix}
\]

where

\[
h[n] = \begin{cases} 
\frac{1}{L}, & 1 \leq n \leq L \\
0, & \text{else}
\end{cases}
\]

Exercise 1: Creating the averaging matrix

In Matlab, the \( N \times M \) averaging matrix, \( C \), with \( N = 31 \) and \( L = 5 \) is created as

\[
N = 31; L = 5; h = 1/L*ones(1,L); C = convmtx(h,N);
\]

Exercise 2: Creating some test signals

A length \( M \) input signal \( a \) with normally distributed samples is created and transformed with the linear transformation \( C \) in order to obtain \( u \). This is sampled at some fixed indices in order to obtain the known linear functionals, \( y \). Note that in practice, one would not have access to either \( a \) or \( u \). This is accomplished by the Matlab code

\[
M = size(C,2); a = randn(M,1); u = C*a; xx = [3 10 14 18 21 29]; yy = u(xx);
\]

Exercise 3: Create the inner product matrix

In this exercise, we find the inner product matrix of the representors corresponding to the known functionals, using the inner product of the filter class. The inner product for the filter class is related to the correlation matrix of the filter, and is given as \( \langle x, y \rangle = x^T Q y \), where the matrix \( Q \) is calculated as

\[
R = C*C'; Q = pinv(R);
\]
The matrix $\Phi$, the matrix of inner products of known functionals, $\varphi$:

$$\{\Phi\}_{ij} = \langle \phi_i, \phi_j \rangle$$

and may be calculated by

$$\text{PHI} = R(:,xx)' \times Q \times R(:,xx);$$

or by noting that the $Q$ inner-product with columns of $R$ results in sampling

$$\text{PHI} = R(xx,xx);$$

Exercise 4: Calculating the optimal estimate

The best estimate of the signal, $\bar{u}$, is a linear combination of the representors of the known functionals for scalars that are determined from the fact that these known linear functionals are correct for $\bar{u}$. This best estimate is calculated by the Matlab commands

$$c = \text{PHI}\backslash\text{yy}; \quad \text{ubar} = R(:,xx)c;$$

Figure 1 shows the actual signal $u$, the known linear functionals (samples), and the optimal estimate $\bar{u}$.

Exercise 5: The Q-norm of $\text{ubar}$

For this exercise, a constant to the filter input $a$ on the order of twice the variance was added. For the resultant sub-sampled $u$, the corresponding $\bar{u}$ was calculated. The optimal estimate is always less than the original signal. Figure 2 shows an example, and Table 1 shows the Q-norms of $u$, $\bar{u}$, and $u - \bar{u}$ for various $a$. 

![Figure 1](image-url)
Exercise 6: Special Case: the averaging operation

For the case when \( N = 30, \ L = 6, \) and the known samples are evenly spaced 3 samples apart \( (P = 3), \) the optimal interpolation closely approximates linear interpolation, as seen in Figure 3. This can be seen by plotting the representors corresponding to known samples, which are triangles, as seen in Figure 4. Another example which leads to linear interpolation is when \( N = 22, \ L = 8, \) and \( P = 4. \) In general, if \( L \) is some integer multiple of \( P, \) then optimal interpolation is linear interpolation for this signal class.
Figure 3.

Figure 4.
Exercise 7: Error Bounds: Calculating the Maximum Error

Exercise 8: Worst case signals and error bounds

Now that the optimal estimate \( \bar{u} \) may be calculated, we would like to bound the error in the estimate. We find the maximum error possible in each sample estimate, as well as worst case signals in the filter class that achieve the maximum error at some samples. The formulae for calculating the worst case signals and error bounds are given in [1], and are implemented in the following Matlab code

\[
\text{PHIINV} = \text{inv}(\text{PHI}); \\
\text{for} \ ii = 1: \text{N}; \\
\quad \text{cbar}(; , ii) = \text{PHIINV} \ast \text{R}(; , \text{xx})' \ast \text{Q} \ast \text{R}(; , ii); \\
\text{end} \\
\text{for} \ ii = 1: \text{N} \\
\quad \text{ybar}(; , ii) = \text{R}(; , [ii \ xx]) \ast [1; -\text{cbar}(; , ii)]; \\
\quad \text{if} \ \text{sum}(\text{ybar}(; , ii)) > \text{eps} \\
\quad \quad \text{ybar}(; , ii) = \text{ybar}(; , ii) / \sqrt{\text{ybar}(; , ii)' \ast \text{Q} \ast \text{ybar}(; , ii)}; \\
\quad \text{end} \\
\text{end} \\
\text{scale} = \sqrt{\text{a}' \ast \text{a} - \text{ubar}' \ast \text{Q} \ast \text{ubar}}; \\
\text{for} \ k = 1: \text{N} \\
\quad \text{uworst}(; , k) = \text{ubar} + \text{scale} \ast \text{ybar}(; , k); \\
\quad \text{uworst}_-(; , k) = \text{ubar} - \text{scale} \ast \text{ybar}(; , k); \\
\quad \text{normvect}(k) = \sqrt{\text{uworst}(; , k)' \ast \text{Q} \ast \text{uworst}(; , k)}; \\
\text{end} \\
\text{maxerror} = \text{abs(}\text{scale} \ast \text{diag}(\text{ybar})\text{;} \\
\]

Exercise 9: Plotting the estimate

The original signal, the optimal estimate, and the error bounds are plotted together in Figure 5. As seen, the original signal is well within the error bounds. The worst case signals may also be plotted, however there is a different worst case signal for each of the linear functionals to be estimated, i.e. each of the points to be interpolated. Figure 6 shows the worst case signals for estimating the value of \( u[10] \) (other worst case signals are omitted for clarity). Figure 7 shows all of the worst case signals for all of the linear functionals. As seen, no worst case signal achieves the maximum error at more than one sample index.
Figure 5.

Figure 6.
Figure 7.

References