9.5 Least-Squares Digital Filters

- objective is to fit sections of the data to a polynomial using least-squares
- choice of polynomial
- centering the data and convolution
- derivation of a 5-point quadratic filter
- Mathcad program to generate filters
- noise reduction vs. filter type and length
- smoothing Johnson noise
- non-smoothing of interference noise
- smoothing and simultaneous differentiation
Objective

**Objective**
- the noise is known to have a normal pdf, but the signal frequencies are unknown
- a least-squares-based technique can be used that fits a section of the measured data to a polynomial
- the entire data set is smoothed by fitting successive sections of the data to the same polynomial

**Choice of Polynomial**
- any polynomial can be chosen, however, it should be capable of matching the slope, curvature, inflection points, and positions of maxima and minima
- a straight line can match only the slope of the data
- a quadratic equation can match the slope and curvature
- a cubic equation can match the slope, curvature and inflection points of the data - however, the cubic equation being odd will shift the location of peaks
- the quartic equation is the first that possesses all the desired traits
centering the filter
- least squares filters work by convolution
- the filter has an odd number of elements and is centered about the data point being smoothed
- for smoothing only the intercept is computed
- centering simplifies the calculation by eliminating odd powers of the independent variable
- because of the elimination of odd powers, the quadratic and cubic filters are identical, and the quartic and quintic filters are identical

fitting procedure
- math is exactly like the Fourier transform filters
- the result of each step of multiplication, summing and normalizing is the least squares intercept for the polynomial
- after each step the filter is moved one element further along the data vector and the process repeated
- for a filter length of N+1 elements, the first N/2 and last N/2 data points are not smoothed
The figure shows the least squares smoothing operation for a quadratic filter of length 5.

The magenta-colored arrows show the first and last points that can be smoothed. The left comb shows the set of points involved with smoothing the sixth point. The blue line is the least squares quadratic fit to the five data points (red). The second comb is smoothing the 16th point. Again, the blue line is the least squares quadratic fit.
Write out the initial, un-weighted least squares equations for a quadratic.

\[
\begin{align*}
\Sigma y &= a_0 N + a_1 \Sigma x + a_2 \Sigma x^2 \\
\Sigma xy &= a_0 \Sigma x + a_1 \Sigma x^2 + a_2 \Sigma x^3 \\
\Sigma x^2 y &= a_0 \Sigma x^2 + a_1 \Sigma x^3 + a_2 \Sigma x^4
\end{align*}
\]

Solve by using determinants and simplify since odd powers of x go to zero with centered data.

\[
a_0 = \frac{\begin{vmatrix}
\Sigma y & \Sigma x & \Sigma x^2 \\
\Sigma xy & \Sigma x^2 & \Sigma x^3 \\
\Sigma x^2 y & \Sigma x^3 & \Sigma x^4 \\
\end{vmatrix}}{\begin{vmatrix}
N & \Sigma x & \Sigma x^2 \\
\Sigma x & \Sigma x^2 & \Sigma x^3 \\
\Sigma x^2 & \Sigma x^3 & \Sigma x^4 \\
\end{vmatrix}} = \frac{\begin{vmatrix}
\Sigma y & 0 & \Sigma x^2 \\
\Sigma xy & \Sigma x^2 & 0 \\
\Sigma x^2 y & 0 & \Sigma x^4 \\
\end{vmatrix}}{\begin{vmatrix}
N & 0 & \Sigma x^2 \\
0 & \Sigma x^2 & 0 \\
\Sigma x^2 & 0 & \Sigma x^4 \\
\end{vmatrix}}
\]
Five-Point Quadratic Filter (2)

Decompose the determinant using standard rules for a $3 \times 3$.

$$a_0 = \frac{\Sigma y \Sigma x^2 \Sigma x^4 - \Sigma x^2 y \left( \Sigma x^2 \right)^2}{N \Sigma x^2 \Sigma x^4 - \left( \Sigma x^2 \right)^3}$$

With a 5-point convolute the x-axis values are -2, -1, 0, +1 and +2. This makes $\Sigma x^2 = 10$ and $\Sigma x^4 = 34$. Plugging these sums and the remaining x-values into the above equation yields the following equation.

$$a_0 = \frac{-3y_{-2} + 12y_{-1} + 17y_0 + 12y_{+1} - 3y_{+2}}{35}$$

Thus the convolute would be the vector $[-3, 12, 17, 12, -3]$ with a normalization factor of 35.
Mathcad Generation of Filters

Generation of Quartic Least Squares Filters

size of filter
generate the centered x-axis values

\[
N := 7 \\
i := 0 .. N - 1 \\
x_i := i - \frac{N - 1}{2}
\]

compute the sums needed by the matrix algebra

\[
sx2 := \sum_i (x_i)^2 \\
sx4 := \sum_i (x_i)^4 \\
sx6 := \sum_i (x_i)^6 \\
sx8 := \sum_i (x_i)^8
\]

calculate the filter values (the zero subscript produces smoothing coefficients, a subscript of 1 would produce the first derivative coefficients, etc.)

\[
f_{0i} := \begin{bmatrix}
N & 0 & sx2 & 0 & sx4 \\
0 & sx2 & 0 & sx4 & 0 \\
sx2 & 0 & sx4 & 0 & sx6 \\
0 & sx4 & 0 & sx6 & 0 \\
sx4 & 0 & sx6 & 0 & sx8
\end{bmatrix}^{-1} \begin{bmatrix}
1 \\
x_i \\
(x_i)^2 \\
(x_i)^3 \\
(x_i)^4 \\
\end{bmatrix} = \begin{bmatrix}
0.022 \\
-0.13 \\
0.325 \\
0.567 \\
0.325 \\
0.022
\end{bmatrix}
\]
Signal-to-Noise Enhancement

The following standard deviations assume that $\sigma = 1$ for the unsmoothed data. For this case a computed first derivative would have $\sigma = 2^{1/2}$ and a computed second derivative would have $\sigma = 6^{1/2}$. Values <1, $2^{1/2}$, or $6^{1/2}$ enhance the SNR.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Output</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>smooth</td>
<td></td>
</tr>
<tr>
<td>Quadratic</td>
<td>dy/dx</td>
<td>0.316</td>
</tr>
<tr>
<td></td>
<td>$d^2y/dx^2$</td>
<td>0.535</td>
</tr>
<tr>
<td>Quartic</td>
<td>smooth</td>
<td>---</td>
</tr>
<tr>
<td></td>
<td>dy/dx</td>
<td>---</td>
</tr>
<tr>
<td></td>
<td>$d^2y/dx^2$</td>
<td>---</td>
</tr>
</tbody>
</table>

For a fixed-length filter the quadratic always provides a better SNR enhancement than the quartic. However, the quadratic distorts the waveform more than the quartic. Also note that small increases in filter length, e.g. 15 to 21, do not appreciably improve the SNR. To improve the SNR by 10, a quadratic filter would need a length of 225.

The SNR has improved by a factor of 2.6.
The least squares polynomial does not remove interference noise because it can adjust the coefficients to follow curved data.
The least squares filter shown in the upper graph takes the derivative of the signal in addition to smoothing. Note that the filter is consistent with our Fourier transform basis set.

The result, shown in the bottom graph is the smoothed derivative of the noisy spectrum in slide 6. Although the result seems too have too much noise, a careful examination of the y-axis voltage shows that it has been reduced by ~5. The predicted value is \[ \frac{1}{(6^{1/2} \times 0.07)} = 5.8. \]

*Usually the derivative of noise is more noise, making this filter special!*
For a graphic presentation of transfer functions for different order and types of filters see, K. R. Betty and G. Horlick, *Analytical Chemistry*, **49**, 351 (1977). The spectral transfer function is dominated by convolution with the sinc function produced by filter length truncation. As expected, first derivative filters produce odd spectral transfer functions.