```
>> % Input signal
x = [-0.01, -0.01, -0.00, ..., -0.00];

% Output signal
y = [0.00, -0.00, -0.00, ..., 0.00];

% Filter requirements.
order = 6;
fs = 30.0;       % sample rate, Hz
lowcut = 3.0;    % desired cutoff frequency of the filter, Hz
highcut = 6.0;   % desired cutoff frequency of the filter, Hz

% Create the filters
[b_low,a_low] = butter(order, lowcut/(fs/2), 'low');
[b_high,a_high] = butter(order, highcut/(fs/2), 'high');
[b_band,a_band] = butter(order, [lowcut highcut]/(fs/2), 'band');

% Apply the filters and convolve with the input signal
ylp = filter(b_low, a_low, x);
yhp = filter(b_high, a_high, x);
ybp = filter(b_band, a_band, x);

% Compare the filtered outputs with the given output signal using correlation
corr_ylp = xcorr(ylp, y, 'coeff');
corr_yhp = xcorr(yhp, y, 'coeff');
corr_ybp = xcorr(ybp, y, 'coeff');

% Identify which filtered output best matches the given output signal based on
the correlation result
```

```
[~, best_match] = max([max(corr_ylp), max(corr_yhp), max(corr_ybp)]);

if best_match == 1
    disp('The Low Pass filter output best matches the given output signal.');
elseif best_match == 2
    disp('The High Pass filter output best matches the given output signal.');
else
    disp('The Band Pass filter output best matches the given output signal.');
end
```