M368K
Homework #4

Burden and Faires.

Section 8.5 (#2[1], 6[2], 7d[3], 17[4]). Section 8.6 (#1c, 2b[5]).

[1]Find trig poly for $n = 2$.

[2]Find trig poly for arbitrary n.

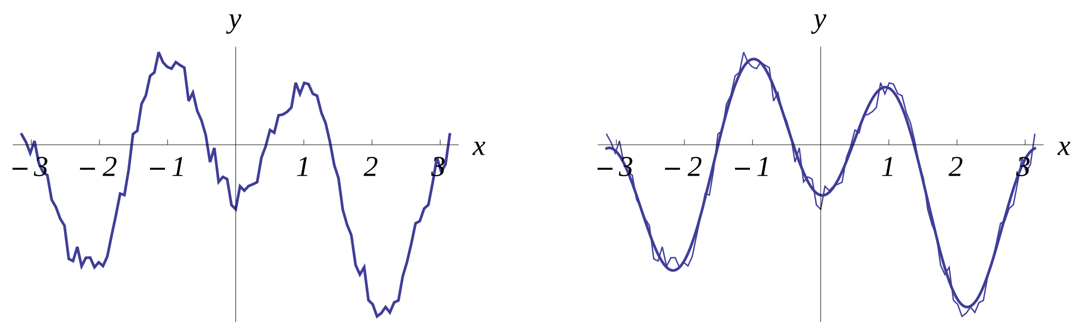
[3]Use $m = 3$, $n = 2$.

[4]Use the condition $\partial E/\partial a_k = 0$ and Eq.(8.24) to derive the expression for a_k when $k \neq 0$.

[5]For full credit use the matrix version of the FFT algo described in class.

Programming mini-project.

The voltage output y from a microphone is sampled at equally-spaced times $t \in [0, T]$ to produce data (t_j, y_j), $j = 0 \ldots 2m-1$, where $m = 50$. After a change of variable $x = -\pi + 2\pi t/T \in [-\pi, \pi]$, we obtain data (x_j, y_j) as shown on the left, where $x_j = -\pi + j\pi/m$. (The data points are given in a file.)



Here we remove or filter the high-frequency noise from the data by fitting it with a low-degree trigonometric polynomial as shown on the right.

(a) Download the C++ program file `program4.cpp` and the data file `program4.dat` from the course webpage. Complete the program so that it computes the coefficients of the least-squares trigonometric polynomial $S_n(x)$ for given data (x_j, y_j), $j = 0 \ldots 2m-1$, and degree $2 \leq n < m$. Your program should also compute the associated fitting error $E_n = \sum_j [y_j - S_n(x_j)]^2$.

(b) Find the least-squares trigonometric polynomials of degree $n = 2, 3, 4$ for the data in `program4.dat`. For each n, report the coefficients of $S_n(x)$ and the fitting error E_n. As before, since E_n generally decreases with n, an appropriate fitting degree can be identified by increasing n from its initial value until $|E_n - E_{n-1}|/E_{n-1} < 5\%$. Use this method to find an appropriate degree, and make a plot of $S_n(x)$, $x \in [-\pi, \pi]$ for this degree. Does $S_n(x)$ look like a filtered version of the noisy signal shown above? The number $\gamma = 1 - \frac{n}{m} \in [0, 1]$ represents a compression ratio: a given set of $2m$ data points can be converted into a smaller set of $2n$ coefficients which can be used to approximately re-create the data when desired; $\gamma = 0$ means no compression, $\gamma = 1$ means complete compression. What is the compression ratio γ for the fitting degree you found?

Turn in: completed copy of `program4.cpp` and response to (b).