HW9 (9.1 – 9.4) Due Friday March 22 – On paper *and* on-line.

9.1 Fourier Analysis: Nyquist Frequency and Aliasing CONCEPTS

4 pts a) Consider the function from problem 8.3: $f(t) = 3 + 1.2\sin(2\pi t/3) - 3.5\sin(2\pi t) + 2.5\cos(4\pi t)$.
– What's the *Nyquist* frequency of the function?
– What's the minimum number of DFT "points" (value of 2N) you would have to use over one period to reproduce the signal *exactly* (*i.e.* no aliasing)?

b) Consider the exact "railway track pulse" motion you analyzed in 8.5 – 8.7.
– What's the *Nyquist* frequency of the function?
– What's the minimum number of "points" (value of 2N) you would have to use over one period, $0 \le t < 2$ s, for the DFT in 8.7 to reproduce the analytic signal *exactly* (*i.e*. no aliasing)?

(**Show your work** above and **justify** your answers. Unjustified answers get no grade.)

9.2 Fourier Analysis: Nyquist Frequency and Aliasing in MATLAB

6 pts By now I'm expecting you've got everything sorted out in 8.7 for automatically using the `fft` command for analyzing discrete data (for *any* function $f(t)$), so you should be able to use the same script, with minimal changes, to quickly do this analysis and make these plots!

Consider the function from problem 8.3: $f(t) = 3 + 1.2\sin(2\pi t/3) - 3.5\sin(2\pi t) + 2.5\cos(4\pi t)$, which is periodic with a period of $\tau = 3$ seconds.

a) In MATLAB, sample the function at **6 Hz** (*i.e.* $\Delta t = 1/6$ sec) for $2N = 18$ points over $0 \le t_j < \tau$. Remember to not include the point at $t_{2N+1} = \tau = 3.0$ seconds !!!!!

- Use `fft` in MATLAB to determine the coefficients of the Discrete Fourier Transform of $f(t_j)$.
- Use these coefficients to reconstruct the inverse transform of $f(t)$ over a finely-spaced time vector of 1000 points over $0 \le t \le \tau$.
- Make a plot (called **HW9_2a.pdf**) with three things plotted on top of one another: the original (exact) function in solid black, the 18 sampled data points as circles, and the reconstructed approximation in dashed red. A good command to use would be:

```
plot(t,f_exact,'-k', tj,fj,'or', t,f_approx,'--r')
```

where `t` is the finely-spaced time vector (with 1000 points), and `tj` and `fj` the sampled data (with 18 points).

b) Redo all the steps in part (a), but this time use a sampling rate of **2 Hz** (*i.e.* $\Delta t = 0.5$ seconds), for $2N = 6$ points over the period. Call this plot **HW9_2b.pdf**.

Now use the results from (a) and (b) to answer the following questions:

c) Compare the *analytic* Fourier transform coefficients for $f(t)$ (*hint*: see solution to HW8.3) with the *discrete* Fourier transform coefficients from parts (a) and (b). For each case, which coefficients are the same, and which ones are different?

d) Look at the plots you made in (a) and (b). Discuss how the "approximate" functions (*inverse* Fourier transform) compare to the exact function.

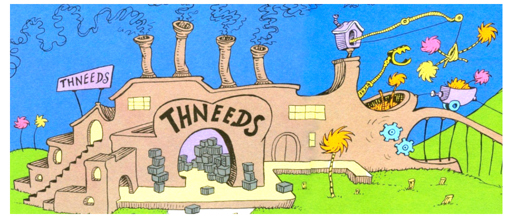
e) Explain *why* you get the differences you see. Did we do anything "bad" in one of our discrete Fourier analyses? If so … *what*?

Due on-paper: Answers and discussions for parts (c), (d) and (e).

Due on-line: plots **HW9_2a.pdf** and **HW9_2b.pdf** (no script, nothing in the comment box).

HW9 (9.1 – 9.4) Due Friday March 22 – On paper *and* on-line.

Concept: Machine Failure Probability



Given the "*failure density*" probability distribution $f(x)$ of a product, the likelihood of a product to fail by time t, is given by the integral

$$F(t) = \int_0^t f(x)\,dx$$

The integral for $F(t)$ above returns a value between 0 (no chance of failure) and 1 (100% chance of failure) by time t.

The Once-ler family's engineer has calculated a "constant failure rate" density $f(x) = \lambda e^{-\lambda x}$ for their *thneed*-making plant, with the constant value $\lambda = 0.01$ and time x in units of months.

The Once-ler insists their plant run trouble-free for at least 70 months (before his patent runs out and Truffula tree harvesting becomes open to other -lers). Your goal is simply to evaluate $F(70) = \int_0^{70} f(x)\,dx$ thereby calculating the likelihood their plant fails before operating 70 months.

9.3 Computational Integration: Newton-Cotes Rules (By-Hand)

6 pts a) Evaluate the integral for $F(70)$ **by hand**, accurate to 5 decimals, using the following two methods. For each method, divide the whole range into seven intervals with ***n* = 8** equally-spaced nodes.

i. Midpoint method
ii. Trapezoid method

b) Given that the integral has the exact (analytic) answer $F(t) = 1 - e^{\lambda t}$, calculate the **percent (relative) error** in calculating $F(70)$ for each method, and **rank** the two methods from best to worst.

Show all your work on paper, especially the values you're using for **all the nodes** and **midpoints** (as appropriate, depending on the method).

9.4 Computational Integration: MATLAB, and how Error Improves with More Nodes, *n*

8 pts Calculate the likelihood F that the plant fails before operating 70 months using two methods (Midpoint, Trapezoid), by writing three functions to run off one MATLAB script:

a) Function 1: `f = ` **`Fun`**`(x)`, the function that outputs the integrand $f(x)$ (*i.e.* the "thing" that is to be integrated over the range $x = 0$ to 70).

b) Function 2: `Imid = ` **`Midpoint`**`(a,b,n)` that outputs the integral of `Fun(x)` over $[a,b]$ using n nodes with the Midpoint rule.

c) Function 3: `Itrap = ` **`Trapezoid`**`(a,b,n)` that outputs the integral of `Fun(x)` over $[a,b]$ using n nodes with the Trapezoid rule.

d) Script: Write an m-file called **HW9_4.m** that …

i. Calls `Midpoint` multiple times to evaluate F using the midpoint method once for each of the following number of nodes: n = [3 8 51 501 5001 50001].

ii. Similarly calls `Trapezoid` to evaluate F using the trapezoid method for the same values of nodes: n = [3 8 51 501 5001 50001].

iii. Uses the exact value $F(70) = 1 - e^{(-0.01)(70)}$ to calculate the **percent** (relative) **error** in the approximations of F from (i) and (ii).

iv. Plots the **absolute value** of the % error from (i) & (ii) as a function of **interval size**, h.

9.4 (continued …)

When done I expect to see **two** lines on the plot: one line for each method (midpoint & trapezoid), each with 6 values of h, similar to the plot I showed (and posted) in Class 25.

Please make each line different, so the **midpoint** method shows **circles** connected by **black** lines, and the **Trapezoid** method shows **triangles** connected by **green** lines. Be sure to use the *loglog* plotting command to allow details to be seen even when the error gets very small. So, use something like this plot command:

```
loglog(h,PctErrMid,'ok-', h,PctErrTrap,'^b--')
```

Save this plot in pdf form as **HW9_4.pdf**.

Look at the plot: you should be seeing your error get very small as h gets smaller. Read values off the axis scales (by hand) to evaluate the *slopes* of the two lines in log-log space (*i.e.* the slope of log(*Error*) vs. log(h), like we discussed in class).

f) Finally, enter into the comment box on Carmen answers to all the following questions:

   i. From your approximation with the smallest error, what is the likelihood that the Once-ler's plant will fail by 60 months of operation? (Express your answer here as a percent – *e.g.* F = 0.25 = 25% chance of failure).

   ii. What is more **accurate** for this problem: midpoint or trapezoid method? Is that consistent with expectations from class?

   iii. What are the (approximate) **slopes** of each of the two lines in your log-log plot? Are they consistent with expectations from class?

Hints for developing your 9.4 script and functions:

1) Notice that one of the number nodes I'm asking you to do is n = 8, which is the same as what you did by-hand in 9.3. So you can use that example to double-check your MATLAB codes give the same answer as your by-hand work!
2) You should know from class qualitatively what the plot in 9.4 should look like (it's shape, direction, slope, etc.) so that should help you see if you've got major errors in your code.
3) I gave you a lot of hints as to how your script, functions and plot should look in (posted) class 25 notes. Make sure your TA gives you even more hints!

So to summarize the *whole* homework:

Due **on-paper:**

- All of problem 9.1
- Answers and discussions from problem 9.2 parts (c), (d) and (e)
- All of problem 9.3
- Nothing from 9.4

Due **on-line:**

- Plots **HW9_2a.pdf** and **HW9_2b.pdf** (from Fourier problem 9.2)
- Documented script **HW9_4.m**, and three functions: **Fun.m**, **Midpoint.m**, **Trapezoid.m**.
- One plot **HW9_4.pdf**.
- Answers to all three questions in 9.4 part (f) in the comment box.

Double-check **all** your information is there and looks right before logging off!