Remember to adequately label all plots and include any requested code listings with your solutions. *Only include those scripts and functions which are requested.* Show your work for problems that you do by hand. A clear and complete presentation of your solutions is required for full credit.

1. *QR Factorization and Least Squares.* Given an $n \times m$ matrix A with $n \geq m$, you can decompose A into an $n \times m$ matrix Q_1 with orthonormal columns and an $m \times m$ upper triangular matrix R as $A = Q_1 R$. We called this the thin QR factorization.

   (a) By hand, compute the factors Q_1 and R for the following matrix:

   $$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$$

   You can use your result to test the function `gram_schmidt` that you implement below.

   (b) Write a function with the declaration `function [Q,R] = gram_schmidt(A)` which computes the thin QR factorization of the given matrix using Gram-Schmidt orthogonalization. Its input `A` is an $n \times m$ matrix A with $n \geq m$. Its outputs `Q` and `R` are the $n \times m$ matrix Q_1 and the $m \times m$ matrix R with $A = Q_1 R$ respectively.

   (c) The daily high temperature in Albuquerque is roughly modeled by a function of the form

   $$y(x) = a_1 + a_2 \cos(2\pi x/12) + a_3 \sin(2\pi x/12),$$

   where x is measured in months since January and y is measured in degrees Fahrenheit. Data from the National Weather Service, averaged over the years 1914 to 2005, give the following average high temperatures:

x	0	1	2	3	4	5	6	7	8	9	10	11
y	47.2	53.2	60.6	70.0	79.4	89.3	91.7	88.9	82.4	71.0	56.9	47.7

   What is the overdetermined system of equations that relates the model to the data?

   (d) Use the thin QR factorization given by your function `gram_schmidt` to solve the least squares problem associated with this overdetermined system for the coefficients a_1, a_2, and a_3. Plot the data and the fitted model on the same axes. What do you predict will be the high temperature on April 15 ($x = 3.5$) using this fit?

2. *Numerical Differentiation.* We have so far only discussed numerical approximations of the first derivative, but it is just as reasonable to approximate higher derivatives. For instance, a second order approximation to $f''(x)$ at $x = x_0$ is given by

   $$f''(x_0) \approx \frac{f(x_0 + h) - 2f(x_0) + f(x_0 - h)}{h^2}.$$

   (a) Suppose that $p(x)$ is the polynomial interpolating $f(x)$ at $x = x_0 - h$, x_0, and $x_0 + h$. Show that the approximation above can be found by evaluating $p''(x_0)$.

(b) Substitute the expressions given by Taylor's remainder theorem for $f(x_0 - h)$ and $f(x_0 + h)$ to prove that this approximation is second order. Find an upper bound for the absolute error term $|f''(x_0) - p''(x_0)|$ given that $|f^{(4)}(x)| < M$ for $x_0 - h < x < x_0 + h$.

3. *Newton-Cotes Quadrature.*

(a) Write three functions with the following declarations:

```
function i = trapezoid(f,a,b,m)
function i = midpoint(f,a,b,m)
function i = simpsons(f,a,b,m)
```

These implement the trapezoid rule, midpoint rule, and Simpson's rule respectively for approximating the integral

$$I = \int_a^b f(x)\,dx$$

on m subintervals. The input `f` is a function handle for the integrand which can accept a vector as input, `a` and `b` are the bounds of the integral, and `m` is the number of subintervals to use for the approximation. The output `i` is an approximation to the integral.

Important: Do not use a `for` loop in any of your functions. You may, however, use vector operations.

(b) Approximate the integral I for $[a, b] = [-1, 1]$ and $f(x) = \frac{1}{1+x^2}$ using each of these three methods, taking $h = 1/2, 1/4, 1/8, 1/16, 1/32, 1/64$. Show the error for the methods as a function of h on a single `loglog` plot. You can find the true solution analytically. Hand in your plot and comment on the order of accuracy and cost of the methods.

(c) Repeat part (b) for $f(x) = \exp(\sin(6\pi x))$. Comment on your results.