M442, Fall 2013 Practice Problems for the Midterm

The midterm for M442 will be Wednesday, Oct. 23, 7-9 p.m., in Blocker 122. The exam will consist of two parts: Part 1 will not require MATLAB, while Part 2 will require MATLAB. Students will turn in Part 1 before beginning Part 2, but for Part 2 students will have access to all M-files we’ve used this semester, from both lecture and homework. Students will be expected to access data from the course web site.

The midterm will cover the following topics: Regression, including both theoretical aspects and MATLAB implementation; Dimensional analysis, including the simple and general methods, and also including implementation with both regression and structured experiments; Solving differential equations in MATLAB, including initial value problems, event location, boundary value problems, and parameter estimation; Modeling with Newtonian mechanics, compartment analysis, and population dynamics.

This set of practice problems is in the general format of the exam, but is almost twice the length of the actual exam.

Part 1 Problems

1. Suppose a set of \( N \) data points \( \{(x_k, y_k)\}_{k=1}^{N} \) appears to arise from the relation

\[
y = a + \frac{b}{x}.
\]

Use the method of least squares regression to find approximate values for the parameters \( a \) and \( b \).

2. Use the method of dimensional analysis to determine a general form for the period \( P \) of a pendulum of length \( l \) released from rest at angle \( \theta \). Ignore air resistance, and note that the angle \( \theta \) between the pendulum and the vertical is dimensionless.

3. Consider a simple pendulum that has been submerged into a viscous fluid, as depicted in Figure 1. Ignoring the effects we described in class as fluid resistance (as opposed to resistance due to viscosity), use Newtonian mechanics to find an ordinary differential equation for the angle \( \theta(t) \).

Note. Assume the mass is spherical with radius \( r \), and recall that according to Stokes’ Law the viscous force on a sphere with radius \( r \) moving through a fluid with viscosity \( \mu \) is

\[
F = 6\pi \mu rv.
\]

4. Suppose \( M \) grams of a certain heart medication are injected into a patient at time 0, and that whenever the drug is present in the body (excluding the heart) its rate of absorption out of the bloodstream is proportional to the concentration in the body (excluding the heart) with proportionality constant \( r_B \ S^{-1}L \), while whenever the drug is in the heart its rate of absorption out of the bloodstream is proportional to the concentration in the heart with proportionality constant \( r_H \ S^{-1}L \). If blood flows into the patient’s heart with variable rate \( r_I(t) \ L/s \) and out with variable rate \( r_O(t) \ L/s \), and if the initial volume of blood in the heart
is $V_H \ L$ while the initial volume of blood in the body (including the heart) is $V_B \ L$, develop a model for the amount of drug absorbed into heart tissue by $t$.

5. When a raccoon is infected by the rabies virus, one of two (equally likely) things can happen:

1. The raccoon develops *furious rabies*, in which case he becomes hyperactive and is quick to attack other raccoons.

2. The raccoon develops *dumb rabies*, in which case he becomes paralyzed and does not spread the disease.

In either case, the raccoon will not recover, and will die within a week of becoming infected. Develop a model for the spread of the rabies virus through an isolated population of raccoons. Which parameter in your model is determined by the fact that infected raccoons will die within a week of becoming infected, and what is the value of this parameter?
Part 2 Problems

1. In class we used data stored in lvdata.m (available on the course web site) to estimate parameters for the Lotka-Volterra predator-prey model

\[
\frac{dy_1}{dt} = ay_1 - by_1y_2 \\
\frac{dy_2}{dt} = -ry_2 + cy_1y_2.
\]

Re-work this example (in particular, the nonlinear regression in lvnonlinearfit2.m), taking the intial values \(y_1(0)\) and \(y_2(0)\) to be parameters.

2. The M-file lkq.m, available on the course web site, contains economic growth data for the Massachusetts economy, 1890-1926. In particular, three ratios are given for each year—a labor index \(L\), a capital index \(K\), and an output index \(Q\). In this problem, we’ll model this data with the model

\[ Q = aL^\gamma K^{1-\gamma}, \]

where \(a>0\) and it is expected that \(0 < \gamma < 1\).

a. Transform this equation into a linear form, and use this form to obtain estimates for the parameters \(a\) and \(\gamma\). Compute the approximate standard deviation \(s\) for your fit.

b. Find nonlinear regression values for \(a\) and \(\gamma\), and compute the approximate standard deviation \(s\) for your fit.

3. The general single species population model is

\[
\frac{dy}{dt} = r\frac{ay}{1 - (\frac{y}{K})^a}; \quad y(0) = y_0,
\]

which can be solved exactly as

\[
y(t) = \frac{K\cdot y_0}{\left(y_0^a + (K^a - y_0^a) e^{-rt}\right)^{1/a}}.
\]

It’s easy to see that if \(a = 1\) this is simply the logistic model, and it’s straightforward to show, using L’Hospital’s rule, that the Gompertz model is obtained in the limit as \(a \to 0\). Fit this model to the U. S. population data in uspop.m (available on the course web site), and use your results to argue that the Gompertz model is the best model from this family for fitting U.S. population growth. Certainly one of the things you’ll want to compute for this problem is the standard deviation \(s\) associated with your fit.

**Note.** Physically \(a\) should be a positive parameter. Use MATLAB’s documentation on lsqcurvefit to find out how to incorporate a lower bound on your parameter values.

4. Data for a simple pendulum is given in the M-file pdata.m (available on the course web site). Use this data, and your results from Part 1, Problem 2 to obtain a model for pendulum motion as a function of \(l\), \(g\), and \(\theta\).

5. The March 4, 1978 issue of a journal called the British Medical Journal reported on an influenza epidemic at a boys boarding school. The school had 763 students, and the following data was (more or less\(^1\)) observed:

\(^1\)I had to take this data from a graph, so it’s approximate.
In this problem we will model this data with the SIR epidemic model.

\[
\begin{align*}
\frac{dy_1}{dt} &= -ay_1y_2 \\
\frac{dy_2}{dt} &= ay_1y_2 - by_2 \\
\frac{dy_3}{dt} &= by_2.
\end{align*}
\]

a. Use the linear form of these equations to determine approximate values for the parameters \( a \) and \( b \). Compute the approximate standard deviation \( s \) for this fit, and create a stacked plot of your solutions \( y_1(t) \) (susceptible population) and \( y_2(t) \) (infected population) to the SIR model, each plotted along with the corresponding data.

b. Use nonlinear least squares regression to refine your values of \( a \) and \( b \) from Part (a). Compute the approximate standard deviation \( s \) for this fit, and create a stacked plot for your model and data.
Solutions to Part 1

1. The error function in this case is

\[ E(a, b) = \sum_{k=1}^{N} (y_k - a - bx_k^{-1})^2. \]

In order to minimize this, we compute

\[ \frac{\partial E}{\partial a} = 2 \sum_{k=1}^{N} (y_k - a - bx_k^{-1})(-1) = 0 \]
\[ \frac{\partial E}{\partial b} = 2 \sum_{k=1}^{N} (y_k - a - bx_k^{-1})(-x_k^{-1}) = 0, \]

which can be rearranged as

\[ a \sum_{k=1}^{N} x_k^{-1} + b \sum_{k=1}^{N} x_k^{-2} = \sum_{k=1}^{N} y_k x_k^{-1}. \]

We can eliminate \( a \) from this system by multiplying the top equation by \( \frac{1}{N} \sum_{k=1}^{N} x_k^{-1} \), and subtracting the bottom equation. We find

\[ b \left( \frac{1}{N} \left( \sum_{k=1}^{N} x_k^{-1} \right)^2 - \sum_{k=1}^{N} x_k^{-2} \right) = \frac{1}{N} \left( \sum_{k=1}^{N} x_k^{-1} \right) \left( \sum_{k=1}^{N} y_k \right) - \sum_{k=1}^{N} y_k x_k^{-1}. \]

Solving for \( b \), we conclude

\[ b = \frac{\frac{1}{N} \left( \sum_{k=1}^{N} x_k^{-1} \right) \left( \sum_{k=1}^{N} y_k \right) - \sum_{k=1}^{N} y_k x_k^{-1}}{\frac{1}{N} \left( \sum_{k=1}^{N} x_k^{-1} \right)^2 - \sum_{k=1}^{N} x_k^{-2}}. \]

Given \( b \), we can compute \( a \) directly from the first equation

\[ a = \frac{1}{N} \left( \sum_{k=1}^{N} y_k - b \sum_{k=1}^{N} x_k^{-1} \right). \]

(Note. It’s also possible to solve this by setting \( X = \frac{1}{x} \), and using \( y = a + bX \), and the steps in that case are essentially identical to the ones carried out here.)

2. In the absence of air resistance, the period of a pendulum should depend on the length of the pendulum, \( l \), the force of gravity, \( g \), and the angle the pendulum is pulled from the vertical, \( \theta \). We look for dimensionless products

\[ \pi = l^a \theta^b g^c P^d, \]
with dimensions

\[ 1 = L^a L^c T^{-2c} T^d. \]

This gives the dimension equations

\[
\begin{align*}
L : 0 &= a + c \\
T : 0 &= -2c + d.
\end{align*}
\]

For \( \pi_1 \) we choose

\[ \pi_1 = \theta, \]

while for \( \pi_2 \) we choose \( d = 1 \) to get \( c = \frac{1}{2} \) and \( a = -\frac{1}{2} \), giving

\[ \pi_2 = \sqrt{\frac{g}{l}}. \]

According to Buckingham’s Theorem we can express the equation for the period of a pendulum in the form

\[ f(\pi_1, \pi_2) = 0, \]

and the Implicit Function Theorem suggests there exists some function \( \phi(\pi_1) \) so that

\[ \pi_2 = \phi(\pi_1). \]

I.e.,

\[ \sqrt{\frac{g}{l}} = \phi(\theta), \]

and so

\[ P = \phi(\theta) \sqrt{\frac{l}{g}}. \]

3. For this problem, we have three acting forces: gravity, viscosity, and buoyancy (see Figure 2).

The total force in the vertical direction is

\[ F_v = F_b - mg = \frac{4}{3} \pi r^3 g(\rho_f - \rho_m), \]

where we’ve used Archimedes’ Principle for buoyancy, and expressed our mass in terms of its density. The part of this force pulling along the direction of motion is

\[ F = \frac{4}{3} \pi r^3 g(\rho_f - \rho_m) \sin \theta. \]

The force due to viscosity can be expressed as

\[ F_\mu = -6\pi \mu r l \frac{d\theta}{dt}. \]
where we’ve observed that the velocity of the mass is
\[ \frac{ds}{dt} = l \frac{d\theta}{dt}, \]
and the negative sign indicates that the viscous force opposes motion. According to Newton’s second law, we have
\[ ml \frac{d^2 \theta}{dt^2} = \frac{4}{3} \pi r^3 g (\rho_f - \rho_m) \sin \theta - 6 \pi \mu r l \frac{d\theta}{dt}. \]
We conclude
\[ \frac{d^2 \theta}{dt^2} = \frac{4}{3} \pi r^3 g (\rho_f - \rho_m) \frac{ml}{m} \sin \theta - 6 \pi \mu r l \frac{d\theta}{dt}. \]

4. The relevant variables are
\[ y(t) = \text{amount of drug in the heart at time } t, \text{ not yet absorbed} \]
\[ H(t) = \text{amount of drug absorbed into the heart at time } t \]
\[ B(t) = \text{amount of drug absorbed into the body tissue at time } t. \]

Observing that the amount of drug in the body, excluding the heart and not yet absorbed is \( M - y(t) - H(t) - B(t) \), and setting
\[ V(t) = V_H + \int_0^t r_I(s) - r_O(s) ds \]
to be the amount of blood in the heart at time \( t \), we have
\[ \frac{dy}{dt} = \frac{M - y(t) - H(t) - B(t)}{V_B - V(t)} r_I(t) - \frac{(r_O(t) + r_H)y(t)}{V(t)} \]
\[ \frac{dH}{dt} = \frac{r_H y}{V(t)} \]
\[ \frac{dB}{dt} = \frac{r_B (M - y(t) - H(t) - B(t))}{V_B - V(t)}. \]
In this case, the initial conditions are all 0.

5. We define variables as follows:

\[ y_1(t) = \#\text{of susceptible raccoons at time } t \]
\[ y_2(t) = \#\text{who have furious rabies at time } t \]
\[ y_3(t) = \#\text{who have dumb rabies at time } t. \]

In terms of these variables, the simplest reasonable model is

\[
\frac{dy_1}{dt} = -ay_1y_2 \\
\frac{dy_2}{dt} = a\frac{1}{2}y_1y_2 - by_2 \\
\frac{dy_3}{dt} = a\frac{1}{2}y_1y_2 - by_3.
\]

Clearly, we can add a removed population, but it’s not required since this is a closed system. If we measure time in days,

\[ b = \frac{1}{7}. \]
Solutions to Part 2

1. In this case, we require only minor modifications to the file `lvnonlinearfit2.m`. The new file, given below, is `lvnonlinearfit3.m`.

```matlab
function lvnonlinearfit3

%LVNONLINEARFIT3: MATLAB function M-file that takes an initial approximation of parameter values and carries out nonlinear regression to obtain best-fit parameter values for the Lotka-Volterra system and the Hudson Bay data.
%Data is scaled.
%In this case initial conditions are treated as parameters

global years L H w1 w2;
lvdata
w1 = std(H);
w2 = std(L);
guess = [.4732; 0.0240; 0.0234; 0.7646; H(1); L(1)];
[p, error] = fminsearch(@lverr, guess);
a = p(1)
b = p(2)
c = p(3)
r = p(4)
H0 = p(5)
L0 = p(6)
s = sqrt(error/(2*length(H)-length(p)))
s1 = s*w1
s2 = s*w2
%We use length(H)-1 because the initial value is not a data point for purposes of this fit
pause
% [t, y] = ode45(@(lvrhs, [0, 20], [H(1); L(1)], [], p);
subplot(2,1,1)
plot(t, y(:, 1), years, H, 'o')
title('Prey population, model and data', 'FontSize', 14)
subplot(2,1,2)
plot(t, y(:, 2), years, L, 'o')
title('Predator population, model and data', 'FontSize', 14)
%
function error = lverr(p)
%LVERR: Function defining error function for example with Lotka-Volterra equations.
```

[72x708]Solutions to Part 2

1. In this case, we require only minor modifications to the file `lvnonlinearfit2.m`. The new file, given below, is `lvnonlinearfit3.m`.

```matlab
function lvnonlinearfit3

%LVNONLINEARFIT3: MATLAB function M-file that takes an initial approximation of parameter values and carries out nonlinear regression to obtain best-fit parameter values for the Lotka-Volterra system and the Hudson Bay data.
%Data is scaled.
%In this case initial conditions are treated as parameters

global years L H w1 w2;
lvdata
w1 = std(H);
w2 = std(L);
guess = [.4732; 0.0240; 0.0234; 0.7646; H(1); L(1)];
[p, error] = fminsearch(@lverr, guess);
a = p(1)
b = p(2)
c = p(3)
r = p(4)
H0 = p(5)
L0 = p(6)
s = sqrt(error/(2*length(H)-length(p)))
s1 = s*w1
s2 = s*w2
%We use length(H)-1 because the initial value is not a data point for purposes of this fit
pause
% [t, y] = ode45(@(lvrhs, [0, 20], [H(1); L(1)], [], p);
subplot(2,1,1)
plot(t, y(:, 1), years, H, 'o')
title('Prey population, model and data', 'FontSize', 14)
subplot(2,1,2)
plot(t, y(:, 2), years, L, 'o')
title('Predator population, model and data', 'FontSize', 14)
%
function error = lverr(p)
%LVERR: Function defining error function for example with Lotka-Volterra equations.
```
params = [p(1) p(2) p(3) p(4)];
[t,y] = ode45(@lvrhs,years,[p(5);p(6)],[],params); %Notice that we pass
%a parameter vector
error = norm(y(:,1)-H')ˆ2/w1ˆ2+norm(y(:,2)-L')ˆ2/w2ˆ2;

function value = lvrhs(t,y,p)
%LVRHS: ODE for example Lotka-Volterra parameter
%estimation example. p(1)=a, p(2) = b, p(3) = c, p(4) = r.
value=[p(1)*y(1)-p(2)*y(1)*y(2);-p(4)*y(2)+p(3)*y(1)*y(2)];

Notice, in particular, how the parameters are incorporated into the subfunction lverr. We find

\[
a = .4823 \\
b = .0248 \\
c = .0273 \\
r = .9178
\]

\[
y_1(0) = 35.3006 \\
y_2(0) = 3.9684 \\
s_1 = 4.5125 \\
s_2 = 3.5099.
\]

2. We begin by transforming this relation to a linear form

\[
\ln\left(\frac{Q}{K}\right) = \ln a + \gamma \ln\left(\frac{L}{K}\right).
\]

Plotting \(\ln(Q/K)\) versus \(\ln(L/K)\), and fitting a regression line, we obtain the initial guess

\[
a = 1.0077, \quad \gamma = .7435,
\]

with standard deviation for this fit \(s = .1662\).
The remainder of the problem is solved with the MATLAB M-file \textit{lkqfit.m}.

\[
\text{%LKQFIT: MATLAB script M-file for analyzing the labor-capital-output}
\text{%data in lkq. Computes values } \mu_\text{X} \text{ along with error estimates.}
\text{%Define the data}
\text{lkq;}
\text{%Carry out a linear fit}
p = polyfit(log(L./K),log(Q./K),1);
plot(log(L./K),log(Q./K),’o’,log(L./K),p(1)*log(L./K)+p(2))
title(’Plot of ln(Q/K) vs. ln(L/K),’FontSize’,15)
\text{%initial guess at parameter values}
We find

\[ a = 1.0036 \]
\[ \gamma = .7357 \]
\[ s = .1634. \]

3. For this model, there doesn’t seem to be a reasonable way to get approximate parameter values directly, so we’ll begin with the parameter values obtained in class for the logistic model. These are \( r = .0208, K = 486.8046, \) and \( y_0 = 8.2241. \) We also take \( a = 1. \) We use the M-file `ussinglespecies.m`.

---

%USSINGLESPECIES: MATLAB script M-file that carries out a
%nonlinear regression for U.S. population data modeled
%by the general single-species population model
%Define data
uspop
%Define GSSM solution (p(1)=r, p(2)=K, p(3)=y0, p(4)=a)
y = @(p,t) p(2)*p(3)/(p(3)^p(4)+(p(2)^p(4)-p(3)^p(4))*exp(-p(1)*t)).^(1/p(4));
%Start with logistic fit values
p0 = [.0208 486.8046 8.2241 1];
[p error]=lsqcurvefit(y,p0,decades,pops,[0 0 0 0],[])
sd=sqrt(error/(length(decades)-4))
%
%Plot model along with data
modelpops = y(p,decades);
figure
plot(decades,pops,'o',decades,modelpops)
title('General Single Species Population Fit for U.S. Data','FontSize',15)

Running this, we find

\[
\begin{align*}
    r &= 0.0061 \\
    K &= 1.3928 \times 10^3 \\
    y_0 &= 4.3762 \\
    a &= 3.5591 \times 10^{-6},
\end{align*}
\]

and the sample standard deviation is

\[
s = 3.1753.
\]

Also, MATLAB’s termination text in this case read \textit{lsqcurvefit stopped because the size of the current step is less than the default value of the step size tolerance.} This suggests that MATLAB is simply trying to make a smaller; i.e. MATLAB is doing the best it can to get the Gompertz model.

4. We solve this problem with the M-file \textit{pdatafit.m}.

%PDATAFIT: MATLAB script M-file that uses the pendulum data in pdata.m
%to find a function for the period of a pendulum in terms of
%length l, angle theta, and gravity g
%
%Define data
pdata;
p=polyfit(theta,P.*sqrt(g./l),1)
plot(theta,P.*sqrt(g./l),'o',theta,p(1)*theta+p(2))
title('\text{Plot of } P \text{ as a function of } \theta', 'FontSize',15)
axis equal
%Comparison with exact solution
Figure 4: Fit of $\pi_2 = P\sqrt{\frac{t}{l}}$ as a function of $\pi_1 = \theta$.

The fit for $\pi_2 = \phi(\pi_1)$ is shown in Figure 4.

We find $p_1 = .5976$ and $p_2 = 6.0462$, so that our model is

$$P(l, g, \theta) = \sqrt{\frac{l}{g}}(.5976\theta + 6.0462).$$

5. The central difference approximation for evaluating derivatives of, say, $y_2(t)$ is (as in class)

$$\frac{dy_2}{dt} \approx \frac{y_2(t + h) - y_2(t - h)}{2h}.$$

Here, the most straightforward approach is to fit both $a$ and $b$ from the second equation. Dividing by $y_2(t)$, we have

$$\frac{1}{y_2(t)} \frac{dy_2}{dt} = ay_1 - b,$$
so that a plot of \( \frac{1}{y_2(t)} \frac{dy_2}{dt} \) versus \( y_1(t) \) should give \( a \) and \( b \) as slope and intercept respectively. In order to stick with a stepsize of \( h = 1 \) day, we will only use the data points corresponding with days 4 through 13. Alternatively, we could use the forward difference method for day 3. An estimate of \( a \) and \( b \) using every day possible with \( h = 1 \) was developed according to the following MATLAB M-file \textit{sirfit1.m}.

```matlab
%SIRFIT1:
S=[762 740 650 400 250 120 80 50 20 18 15 13 10];
I=[1 20 820 300 260 190 120 80 20 5 2];
days=[0 3 4 5 6 7 8 9 10 11 12 13 14];
for k=1:10
   Y(k)=(1/I(k+2))*(I(k+3)-I(k+1))/2;
   X(k)=S(k+2);
end

c = polyfit(X,Y,1);
a = c(1)
b = -c(2)
plot(X,Y,'o',X,c(1)*X+c(2))
title({{'Plot of \( \frac{y_2(t+h)-y_2(t-h)}{2y_2(t)} \) vs. \( y_1(t) \)}, 'Data and linear fit'}); %

sirrhs = @(t,y) [-a*y(1)*y(2);a*y(1)*y(2)-b*y(2)];
[t,y]=ode45(sirrhs,[0 14],[762;1]);
subplot(2,1,1)
plot(days,S,'o',t,y(:,1))
title({{'Susceptible Population, model and data', 'Obtained by linear fit'}});
subplot(2,1,2)
plot(days,I,'o',t,y(:,2))
title({{'Infected Population, model and data', 'Obtained by linear fit'}});
```

The linear plot created by this code is given as Figure 5. we find

\[
a = .0036 \quad b = .9395 \quad s = 155.9986.
\]

The fit obtained using these values is given in Figure 6. For Part (b), we use the MATLAB script M-file \textit{sirnonlinearfit.m}.

14
Figure 5: Plot of linear fit for the SIR model.

Figure 6: Fit obtained using the values $a = .0036$ and $b = .9395$. 
function sirnonlinearfit
%SIRNONLINEARFIT: MATLAB function M-file that takes an initial
%approximation of parameter values and carries out nonlinear
%regression to obtain best-fit parameter values for the SIR
%system and the influenza data.
global days S I;
S=[762 740 650 400 250 120 80 50 20 18 15 13 10];
I=[1 20 80 220 300 260 240 190 120 80 20 5 2];
days=[0 3 4 5 6 7 8 9 10 11 12 13 14];
guess = [.0036 .9395];
[p,error]=fminsearch(@sirerr, guess);
a = p(1)
b = p(2)
s = sqrt(error/(2*(length(S)-1)-length(p)))
%
[t,y]=ode45(@sirpe,[0,14],[S(1); I(1)],[],p);
subplot(2,1,1)
plot(t,y(:,1),days,S,'o')
title('Susceptible population, model and data','FontSize',14)
subplot(2,1,2)
plot(t,y(:,2),days,I,'o')
title('Infected population, model and data','FontSize',14)
%
function error = sirerr(p)
%LVERR: Function defining error function for
%example with SIR equations.
global days S I;
[t,y] = ode45(@sirpe,days,[S(1);I(1)],[],p); %Notice that we pass
%a parameter vector
error = norm(y(:,1)-S')^2+norm(y(:,2)-I')^2;
%
function value = sirpe(t,y,p)
%LVPE: ODE for example SIR parameter
%estimation example. p(1)=a, p(2) = b.
value=[-p(1)*y(1)*y(2);p(1)*y(1)*y(2)-p(2)*y(2)];

We find

\[
a = .0022
\]
\[
b = .4361,
\]

and \( s = 20.2424 \). The graph created by this file is given in Figure 7.
Figure 7: Fit obtained using the values $a = .0022$ and $b = .4361$. 