MECH 420 SENSORS AND ACTUATORS
Solutions to Assignment 4

Problem 1 (Problem 4.2 from Textbook)

$y = ax^p$

In log scale: $\ln(y) = \ln(a) + p\ln(x)$ with Slope = p and y-intercept = $\ln(a)$

We use the following MATLAB script:

```
% Prob 4.2
xx=exp(1); % initial value of x
dx=0.5;% x increment
a= 1.5; p=2; % parameter values
x=[]; y=[];
for i=1:25
yy=(a+normrnd(0.1,0.2))*xx^p; % y data point
y(i)=log(yy); % log y
x(i)=log(xx); % log x
xx=xx+dx; % increment x
end
x=x'; %convert x data to a column vector
y=y'; %convert y data to a column vector
xlabel('ln x'), ylabel('ln y') %Label the axes
fit(x,y,'poly1')
```

MATLAB Results:

```
Linear model Poly1:
     f(x) = p1*x + p2
Coefficients (with 95% confidence bounds):
       p1 =       1.988  (1.867, 2.109)
       p2 =      0.4739  (0.2179, 0.7298)
Goodness of fit:
  SSE: 0.4535
  R-square: 0.9805
  Adjusted R-square: 0.9797
  RMSE: 0.1404
```

The fitted curve is shown in Figure S4.2.

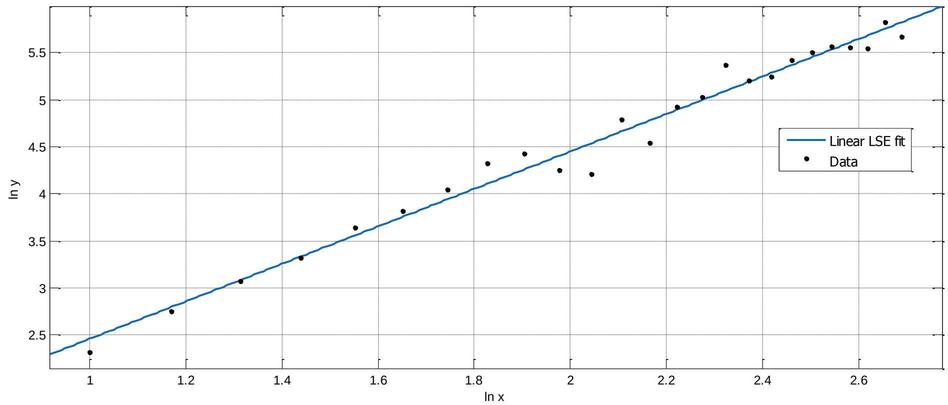


Figure S4.2: Linear LSE fit of data in log scale.

The estimated *p* = p1 = 1.988. This estimate is quite accurate.
The estimated *a* = exp(p2) = exp(0.4739) = 1.606
This estimate of *a* has some error because the random error has a non-zero mean of 0.1, and this bias is added to the estimate.

Problem 2 (Problem 4.3 from Textbook)

(a) According to the problem, Y_i are iid. Take the expected value of $\bar{Y}$:

$$E(\bar{Y}) = E(\frac{1}{N}\sum_{i=1}^{N} Y_i) = \frac{1}{N} E(\sum_{i=1}^{N} Y_i) = \frac{1}{N}\sum_{i=1}^{N} E(Y_i) = \frac{1}{N}\sum_{i=1}^{N} \mu = \frac{1}{N} N\mu = \mu$$

➔ $\bar{Y}$ is an unbiased estimate of μ.

Next, take the expected value of S^2:

$$E(S^2) = \frac{1}{(N-1)}\sum_{i=1}^{N} E\left(Y_i - \bar{Y}\right)^2 = \frac{1}{(N-1)}\sum_{i=1}^{N} E\left(Y_i^2 - 2Y_i\bar{Y} + \bar{Y}^2\right)$$

Now consider the 2nd and the 3rd terms in the summation separately.
Here use the fact that Y_i are iid, and hence $E(Y_iY_j) = E(Y_i)E(Y_j) = \mu^2$ for $i \neq j$

$$\sum_{i=1}^{N} E\left(Y_i\bar{Y}\right) = \sum_{i=1}^{N} E\left(Y_i \frac{1}{N}(Y_1 + Y_2 + \ldots + Y_N)\right) = \frac{1}{N}\sum_{i=1}^{N}(E\left(Y_i^2\right) + (N-1)\mu^2)$$

$$= \frac{1}{N}\sum_{i=1}^{N} E\left(Y_i^2\right) + (N-1)\mu^2$$

$$\sum_{i=1}^{N} E\left(\bar{Y}^2\right) = NE\left(\bar{Y}^2\right) = \frac{1}{N} E(Y_1 + Y_2 + \ldots + Y_N)(Y_1 + Y_2 + \ldots + Y_N) = \frac{1}{N}(\sum_{i=1}^{N} E\left(Y_i^2\right) + N(N-1)\mu^2)$$

$$= \frac{1}{N}\sum_{i=1}^{N} E\left(Y_i^2\right) + (N-1)\mu^2$$

Substitute in the original expression:

$$E(S^2) = \frac{1}{(N-1)}\left[\sum_{i=1}^{N} E(Y_i^2) - 2\left\{\frac{1}{N}\sum_{i=1}^{N} E(Y_i^2) + (N-1)\mu^2\right\} + \left\{\frac{1}{N}\sum_{i=1}^{N} E(Y_i^2) + (N-1)\mu^2\right\}\right]$$

$$= \frac{1}{(N-1)}\left[\sum_{i=1}^{N} E(Y_i^2) - \left\{\frac{1}{N}\sum_{i=1}^{N} E(Y_i^2) - (N-1)\mu^2\right\}\right] = \frac{1}{N}\sum_{i=1}^{N} E(Y_i^2) - \mu^2 = \frac{1}{N}\sum_{i=1}^{N}\left[E(Y_i^2) - \mu^2\right]$$

$$= \frac{N\sigma^2}{N} \quad \Rightarrow \quad S^2 \text{ is an unbiased estimate of } \sigma^2$$

(b)

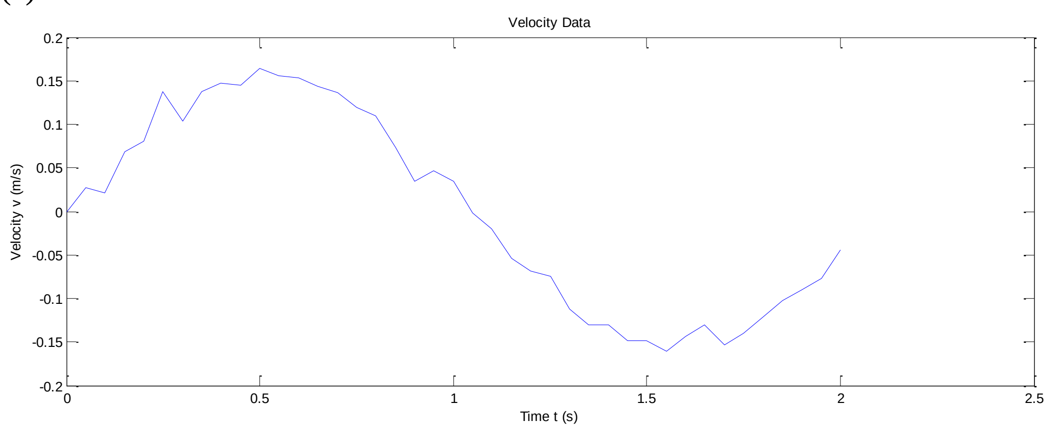
1. For $N = 1$ (i.e., with just one data point, we cannot logically talk about estimating the "variance." This is consistent with the formula for sample variance when $N = 1$ (i.e., 0/0, which is indeterminate); 2. This formula corresponds to an unbiased estimate.

These two reasons justify the use of this formula for estimating the variance.

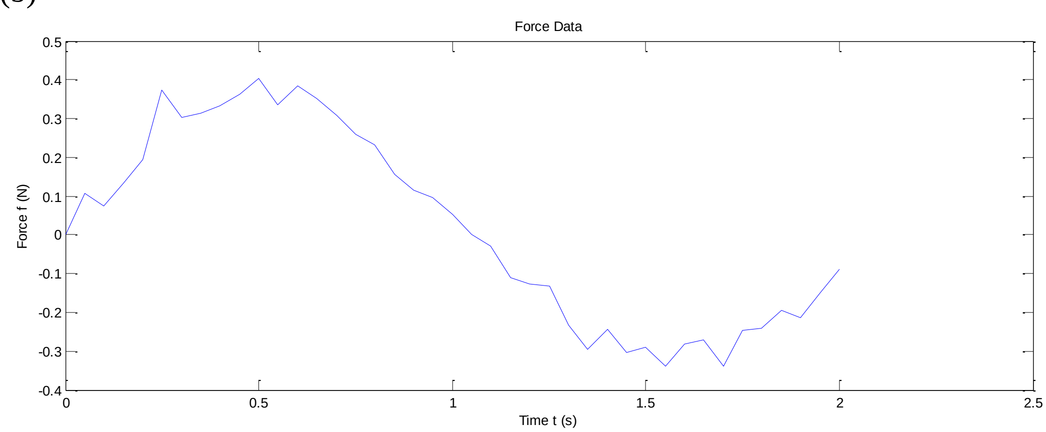
Problem 3 (Problem 4.6 from Textbook)

The data plots are given in Figure S4.6 (a)-(c).

(a)



(b)



(c)

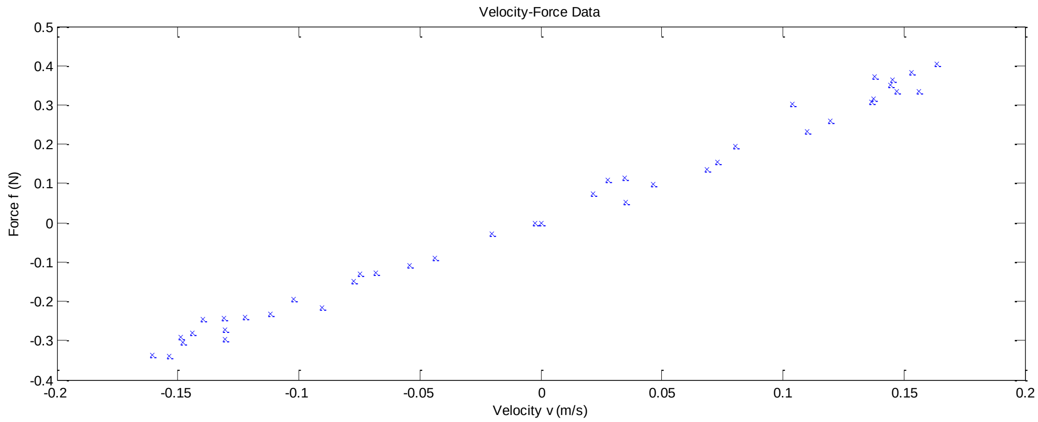


Figure S4.6: (a) Velocity data, (b) Force data, (c) Velocity-force data.

(a)
Possible error sources: Sensor error (noise, calibration error, drift, etc.), actuator error, mechanical loading, electrical loading, system nonlinearity (model error), experimental setup error (mounting, connections, etc.), environmental effects (temperature, etc.), external disturbances, dynamic coupling, synchronization of velocity data and force data, sampling (aliasing) and quantization errors.

(b)

The linear fit is shown in Figure S4.6(d).

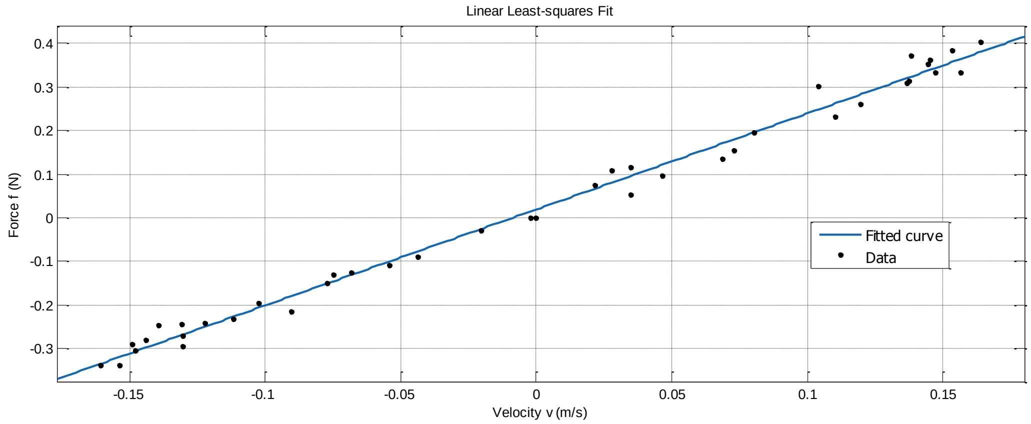


Figure 4.6(d): The linear curve fit.

The parameters and statics of the linear fit are given below. The estimated viscous damping parameter (2.204) closely agrees with the true value (2.2) which was used in simulating the data.

```
Linear model Poly1:
     f(x) = p1*x + p2
Coefficients (with 95% confidence bounds):
       p1 =       2.204  (2.136, 2.271)
```

```
    p2 =      0.01939  (0.01185, 0.02692)

Goodness of fit:
  SSE: 0.02217
  R-square: 0.9911
  Adjusted R-square: 0.9908
  RMSE: 0.02384
```

(c)

The quadratic polynomial fit is shown in Figure S4.6(e). The parameters and statics of the linear fit are given below.

```
Model Poly2:
     f(x) = p1*x^2 + p2*x + p3
Coefficients (with 95% confidence bounds):
       p1 =      0.7154  (-0.1291, 1.56)
       p2 =       2.202  (2.135, 2.268)
       p3 =     0.01055  (-0.00222, 0.02331)

Goodness of fit:
  SSE: 0.02058
  R-square: 0.9917
  Adjusted R-square: 0.9913
  RMSE: 0.02327
```

It is seen that the estimated viscous damping parameter (2.202) closely agrees with the true value (2.2) which was used in simulating the data. However the quadratic coefficient did not match properly and had a wide error range.

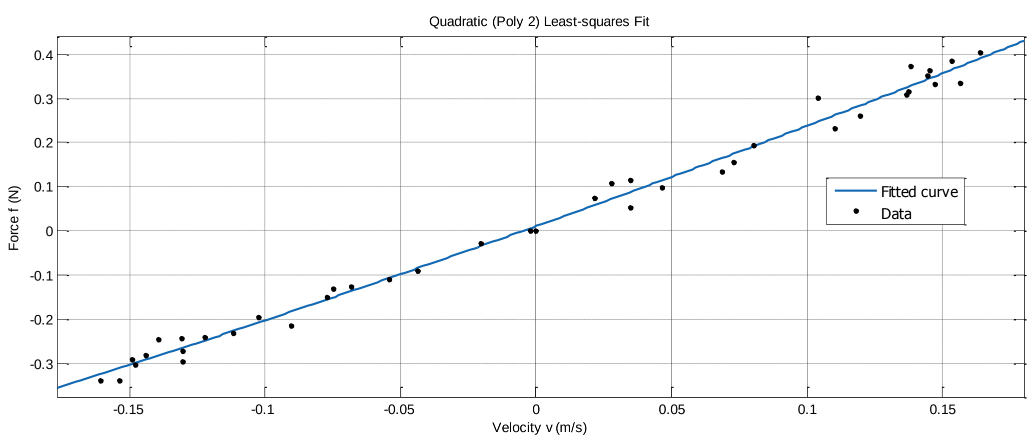


Figure 4.6(e): The quadratic curve fit.

(d)

The estimates of the linear viscous parameter closely agree with the true value. However, the estimates of the quadratic parameter and the offset appear to have considerable error. This is primarily due to measurement error. Since these parameter values are small, it appears that the linear model (with linear fit) is adequate.

Problem 4 (Problem 4.14 from Textbook)

n pulses/rev = $n \times 0.5 \times 10^{-3}$ s/rev = $\dfrac{2000}{n}$ rev/s .

We will convert the data into rev/s and use them in the recursive estimation.

We use the following MATLAB program to obtain the speed estimation with LSE:

```
% Prob 4.14(a)
iter=[];y=[]; ym=[]; v=[]; s=[]; % declare storage vector
iter(1)=1; % first iteration
y(1)=round(normrnd(800,5)); % first data point (pulses/rev)
ym(1)=2000.0/y(1); % first sample mean (rev/s)
s(1)=0.0; % first std
v(1)=0.0; % first sample varianceiter=0; smean=ym(1); sstd=s(1); % initialize
plotting variables
for i=1:24
y(i+1)=round(normrnd(800,5)); % next data point
ym(i+1)=(i*ym(i)+2000.0/y(i+1))/(i+1); % next sample mean
v(i+1)=(v(i)*(i-1)+(2000.0/y(i+1)-ym(i+1))^2)/i; % next sample variance
s(i+1)=sqrt(v(i+1))*100.0; % sample stdx100
iter(i+1)=i+1; % store recursion number
end
% plot the results
plot(iter,ym,'-',iter,s,'-',iter,s,'o')
xlabel('Recursion Number')
```

The final value of the speed estimation is as follows, which is quite close to the true value (2.5 rev/s):

```
Final estimate of speed:
>> ym(25)
ans = 2.4905 rev/s
```

The results from the present recursive estimation are plotted in Figure S4.14(a). Initially (with only a few points of data) we do not observe a particular trend in the sample standard deviation. However, as the samples increase to a reasonably large number, the sample std appears to generally decrease (but not monotonically).

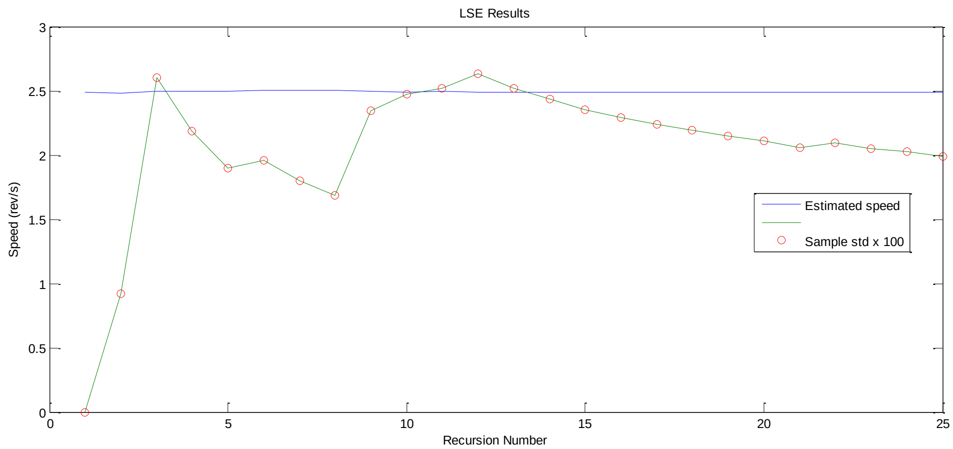


Figure S4.14(a): Results from recursive least-squares estimation.