Linear Predictions of Stock Market Averages

Problem Part (a)

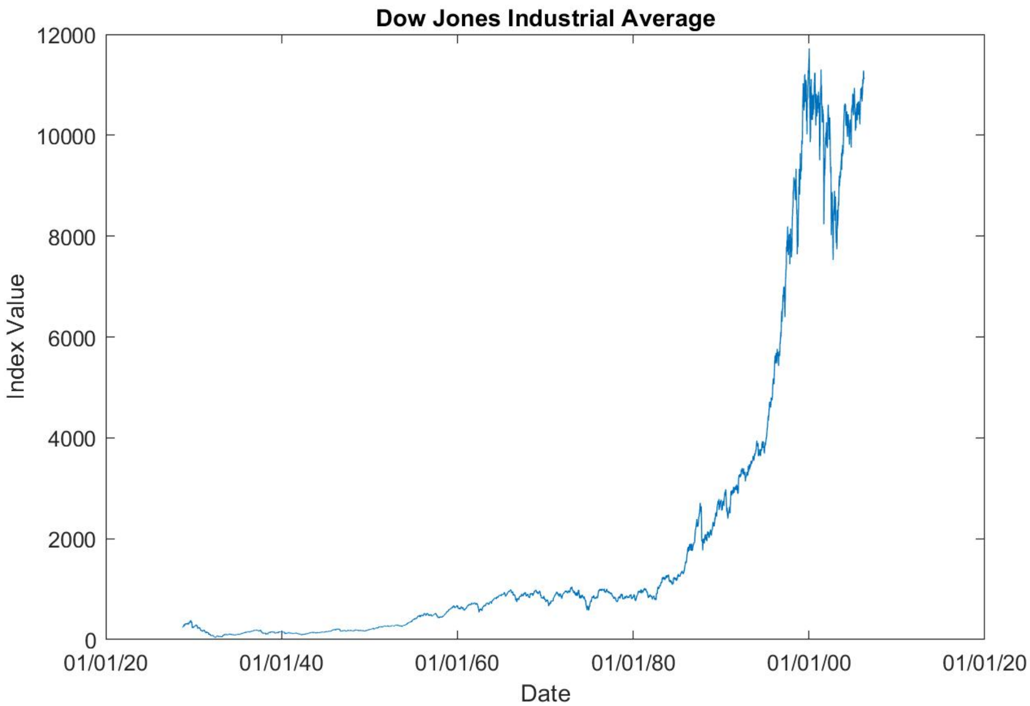

Figure 1. Graph of the Dow Jones Industrial Average on **linear** scale.

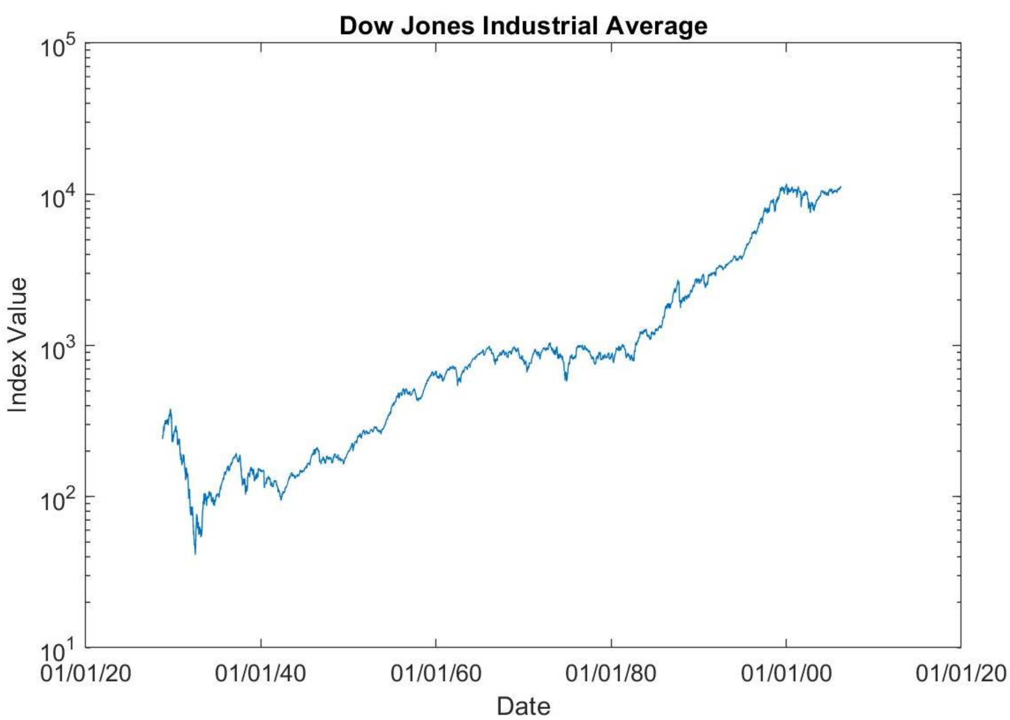

Figure 2. Graph of the Dow Jones Industrial Average on a **semilog** scale.

Amount earned from DJIA over entire span = 46337.25670

Equivalent APR required to achieve same return = 0.04935 (4.935%)

Problem Part (b)

Predictor coefficients over first decade, p = 3:

`a = [0.0267960474, 0.0937731134, -1.1183084409]`

Problem Part (c)

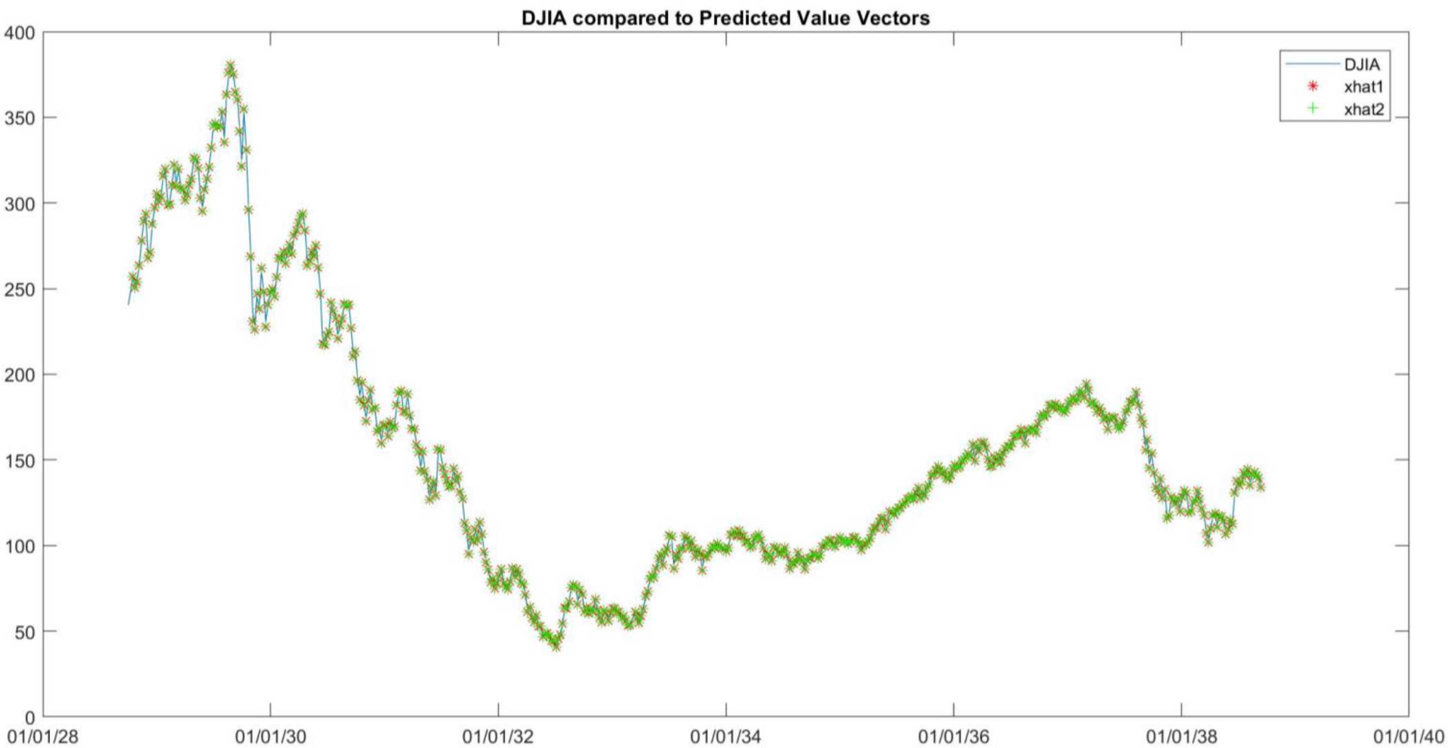

Figure 3. Comparison of DJIA and generated prediction value vectors `xhat1` and `xhat2`, over first decade of data. Vectors `xhat1` and `xhat2` lag the DJIA by p = 3.

Total square error for `xhat1` = 23638.06421

Total square error for `xhat2` = 23638.06421

Problem Part (d)

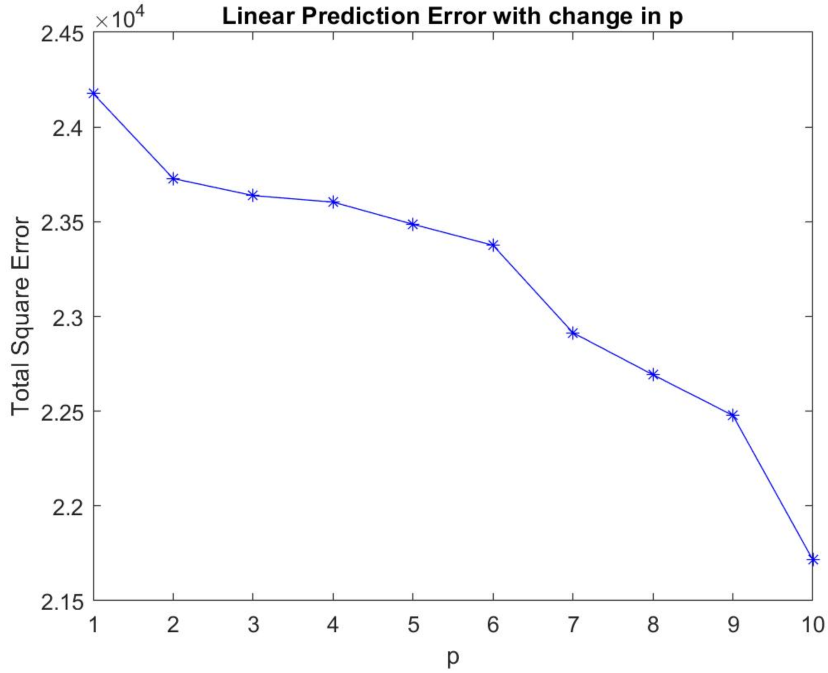

Figure 4. Total square error of linear prediction with increasing order p.

In this range, p = 10 provides the smallest total square error. However, if we extrapolate further, we can see that the error drops off to almost zero around p = 260, and is negligible thereafter:

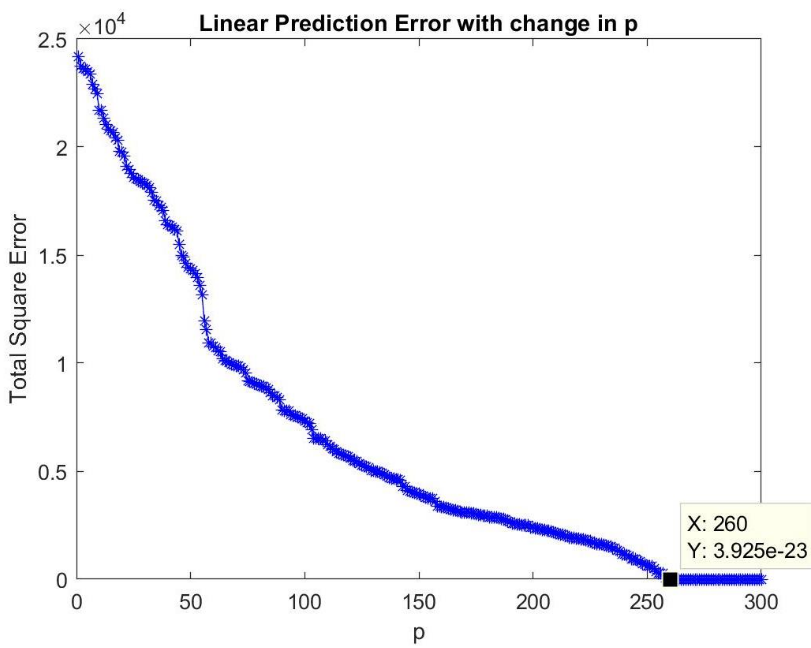

Figure 5. Lowest meaningful total square error of linear prediction at order p = 260.

Problem Part (e)

Predictor coefficients `a` for first decade, p = 10;

```
a = [-0.10268, 0.21325, -0.03380, 0.00952, -0.08588, -0.06323, 0.09255,
0.00763, 0.06856, -1.10356]
```

Upper bound on money made = 4700565.98285

Lower bound – bank interest = 1349.74204

Lower bound – "buy-and-hold" = 544.43830

Money made from predictor = 1422.64823

Equivalent APR to match predictor = 0.03526 (3.526%)

Problem Part (f)

Predictor coefficients `a` for last decade, p = 10;

```
a = [-0.04297, -0.04467, 0.02049, 0.09593, -0.13194, 0.05758, 0.00703,
0.07107, -0.10750, -0.92623]
```

Upper bound on money made = 167340.87340

Lower bound – bank interest = 1349.74204

Lower bound – "buy-and-hold" = 2012.71254

Money made from predictor = 2433.76464

Equivalent APR to match predictor = 0.08902 (8.902%)

Problem Part (g)

Since p = 10; the prediction vector `xhat` only works for 4034 weeks instead of 4044 weeks. Keeping that in mind, the maximum gain and prediction gain were calculated on separate ranges:

Maximum gain over data set = 2.61321e+19

Money made from predictor = 1.29891e+19

The money made using the predictor is roughly 49.7% of the maximum possible gain over the timeframe given. By breaking the DJIA data into blocks and recalculating the prediction coefficients for every block, we are likely to reach an even higher gain with the predictor.

Equivalent APR to match predictor = 3.84586 (384.586%!!!)

Problem Part (h)

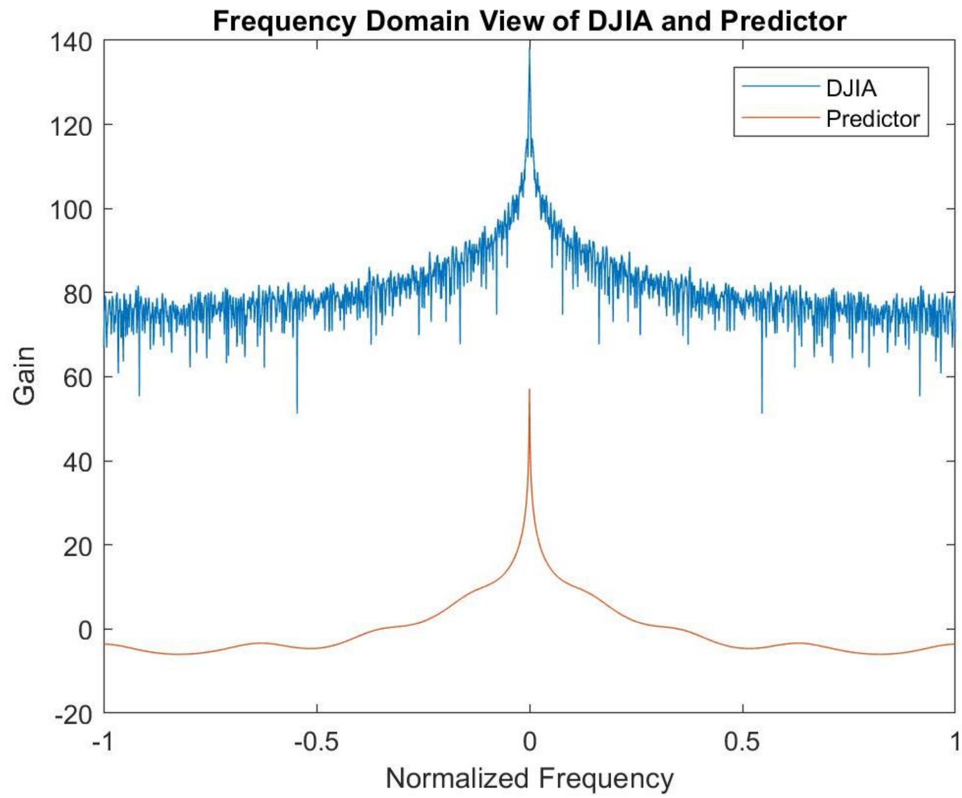

Figure 6. DJIA and predictor plotted in the frequency domain. The predictor is off by a scale factor.

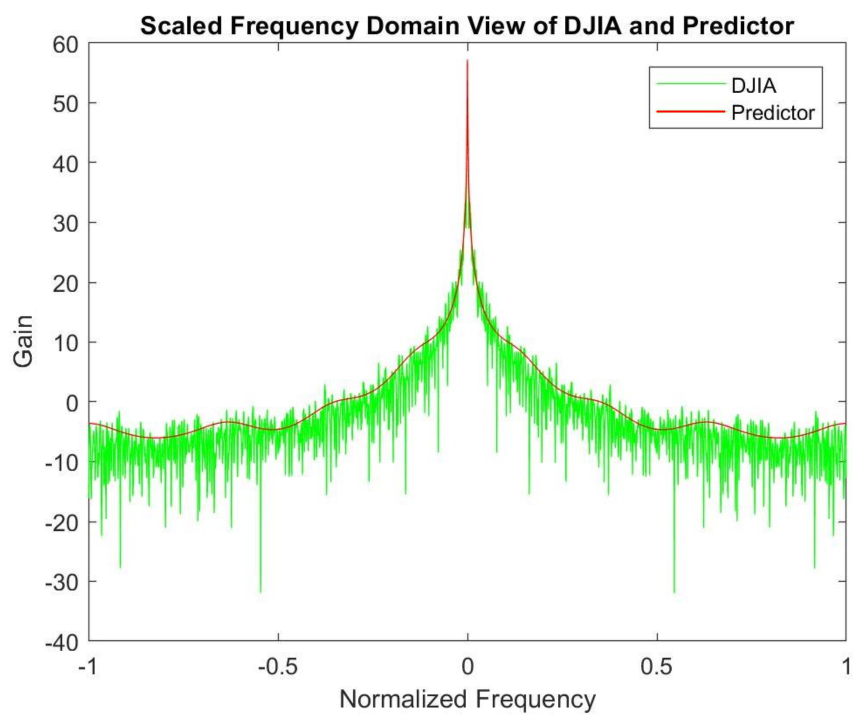

Figure 7. DJIA and predictor plotted in the frequency domain. The DJIA has been scaled by the error vector in the frequency domain.

G = 26909420.63763

We know $-Xa + e = x$, where $Xa = \hat{x}$. In the frequency domain, this is equivalent to $E(e^{j\omega}) = X(e^{j\omega})\hat{X}(e^{j\omega})$. We know $\hat{X}(e^{j\omega}) = 1/1 + \sum_{k=1}^{p} a_k e^{-j\omega k}$. As per Parseval's theorem, the sum of square values in the frequency domain is equivalent to the sum of squared values in the time domain. Hence, the error terms in the time domain are equivalent to the error terms in the frequency domain. Using this knowledge, the predictor coefficients are obtained when $X(e^{j\omega})$ is scaled by $E(e^{j\omega})$, and this is observed when the predictor frequency response matches the DJIA FFT when it is scaled by the error.

MATLAB Code Listing

```
% ECE 4271, Spring 2018
% Project 2 - Stock Market Predictions
% Arjun Sabnis

load('djiaw_2006.mat');

startDate = djiaw(1,1);
endDate = djiaw(end,1);
date = djiaw(:,1);
stock = djiaw(:,2);
Y = length(stock);

% Part (a)
% plot(date,stock), datetick('x',2);
% xlabel('Date'), ylabel('Index Value'), title('Dow Jones Industrial Average');
% figure, semilogy(date,stock), datetick('x',2);
% xlabel('Date'), ylabel('Index Value'), title('Dow Jones Industrial Average');
money = 1000;
for n = 1:Y-1
    money = money*(stock(n+1)/stock(n));
end
apr = 52*((money/1000)^(1/Y) -1);

% Part (b)
p = 3;
N = 520;
X = zeros(N-p,p);
x = zeros(N-p,1);
for i = 1:N-p
    for j = 1:p
        X(i,j) = stock(i+j-1);
    end
    x(i) = stock(p+i);
end

% Part (c)
a = -X\x;
xhat1 = -X*a;
xhat2 = -filter(flip(a),1,stock(1:N));
xhat2 = xhat2(p:N-1);

% plot(date(1:N-1),stock(1:N-1)), datetick('x',2);
```

```
% hold on, plot(date(p:N-1),xhat1,'*r'),plot(date(p:N-1),xhat2,'+g');
% legend({'DJIA','xhat1','xhat2'});
% title('DJIA compared to Predicted Value Vectors'), hold off;

e1 = sum((X*a+x).^2);
e2 = sum((stock(p+1:N)-xhat2).^2);

% Part (d)
e1_arr = zeros(1,10);
e2_arr = zeros(1,10);
for p = 1:10
    X = zeros(N-p,p);
    x = zeros(N-p,1);
    for i = 1:N-p
        for j = 1:p
            X(i,j) = stock(i+j-1);
        end
        x(i) = stock(p+i);
    end
    a = -X\x;
    xhat1 = -X*a;
    xhat2 = -filter(flip(a),1,stock(1:N));
    xhat2 = xhat2(p:N-1);

    e1 = sum((X*a+x).^2);
    e2 = sum((stock(p+1:N)-xhat2).^2);

    e1_arr(p) = e1;
    e2_arr(p) = e2;
end

% plot(1:p,e1_arr,'-*b');
% xlabel('p'), ylabel('Total Square Error');
% title('Linear Prediction Error with change in p')

% Part (e)
p = 10;
r = 0.03;
lbs_market = 1000;
lbs_interest = 1000;
pred_amt_s = 1000;
upper_bound_s = 1000;
xhat_s = zeros(N,1);

for i = 1:N
    for j = 1:p
        xhat_s(i) = xhat_s(i)-(a(j)*stock(i+j-1));
    end
end


for n = p+1:N+p
    lbs_market = lbs_market*(stock(n)/stock(n-1));
    lbs_interest = lbs_interest*(1+(r/52));

    ub1 = upper_bound_s*(stock(n)/stock(n-1));
    ub2 = upper_bound_s*(1+(r/52));
```

```
    if ub1>ub2
        upper_bound_s = ub1;
    else
        upper_bound_s = ub2;
    end

    if (pred_amt_s*(xhat_s(n-p)/stock(n-1)))>(pred_amt_s*(1+(r/52)))
        pred_amt_s = pred_amt_s*(stock(n)/stock(n-1));
    else
        pred_amt_s = pred_amt_s*(1+(r/52));
    end
end

apr_s = 52*((pred_amt_s/1000)^(1/N) -1);

% Part (f)
lbe_market = 1000;
lbe_interest = 1000;
pred_amt_e = 1000;
upper_bound_e = 1000;

X = zeros(N,p);
x = zeros(N,1);
for i = 1:N
    for j = 1:p
        X(i,j) = stock((Y-N-p)+i+j-1);
    end
    x(i) = stock((Y-N-p)+i+j);
end
a = -X\x;
xhat_e = -X*a;

for n = Y-N+1:Y
    lbe_market = lbe_market*(stock(n)/stock(n-1));
    lbe_interest = lbe_interest*(1+(r/52));

    ub1 = upper_bound_e*(stock(n)/stock(n-1));
    ub2 = upper_bound_e*(1+(r/52));
    if ub1>ub2
        upper_bound_e = ub1;
    else
        upper_bound_e = ub2;
    end

    if (pred_amt_e*(xhat_e(n-(Y-N))/stock(n-1)))>(pred_amt_e*(1+(r/52)))
        pred_amt_e = pred_amt_e*(stock(n)/stock(n-1));
    else
        pred_amt_e = pred_amt_e*(1+(r/52));
    end
end

apr_e = 52*((pred_amt_e/1000)^(1/N) -1);

% Part (g)
market = 1000;
interest = 1000;
pred = 1000;
```

```
upper = 1000;

X = zeros(Y-p,p);
x = zeros(Y-p,1);
for i = 1:Y-p
    for j = 1:p
        X(i,j) = stock(i+j-1);
    end
    x(i) = stock(i+j);
end
a = -X\x;
xhat = -X*a;

for n = 1:Y-1
    market = market*(stock(n+1)/stock(n));
    interest = interest*(1+(r/52));

    u1 = upper*(stock(n+1)/stock(n));
    u2 = upper*(1+(r/52));
    if u1>u2
        upper = u1;
    else
        upper = u2;
    end
end

for n = 1:Y-p-1
    if (pred*(xhat(n+1)/stock(n+p-1)))>(pred*(1+(r/52)))
        pred = pred*(stock(n+p)/stock(n+p-1));
    else
        pred = pred*(1+(r/52));
    end
end

apr_total = 52*((pred/1000)^(1/N) -1);

% Part (h)
e = stock(p+1:end)-xhat;
G = sum(e.^2);
G_dft = fftshift(fft(e));
djia_dft = fftshift(fft(stock));
[H,W] = freqz(1,[1; flip(a)],Y/2);
H = [-H(end:-1:1);H];
W = [-W(end:-1:1);W];
% Unscaled
plot(W/pi,20*log10(abs(djia_dft)));
hold on, plot(W/pi,20*log10(abs(H)));
title('Frequency Domain View of DJIA and Pedictor');
xlabel('Normalized Frequency'), ylabel('Gain'), legend({'DJIA','Predictor'});
% Scaled
figure, plot(W/pi,20*log10((abs(djia_dft)/abs(G_dft))),'-g');
hold on, plot(W/pi,20*log10(abs(H)),'-r');
title('Scaled Frequency Domain View of DJIA and Predictor');
xlabel('Normalized Frequency'), ylabel('Gain'), legend({'DJIA','Predictor'});
```