```
input_output = importdata("input_output.mat");

x = input_output.x;
y = input_output.y;

lag = 50;
lag_plot = -lag:lag;

max_M = 20;

Rxx = xcorr(x, lag);
Ryy = xcorr(y, lag);
Rxy = xcorr(x, y, lag);

figure(1)
stem(lag_plot,Rxx)
title("Auto-correlation Rxx[k]")
figure(2)
stem(lag_plot,Ryy)
title("Auto-correlation Ryy[k]")
figure(3)
stem(lag_plot,Rxy)
title("Cross-correlation Rxy[k]")

fftRxx = fft(Rxx);
fftRyy = fft(Ryy);
fftRxy = fft(Rxy);

figure(4)
stem(lag_plot,abs(fftRxx))
title("Power spectral density Rxx[k]")
figure(5)
stem(lag_plot,abs(fftRyy))
title("Power spectral density Ryy[k]")

Ryx = xcorr(y, x, lag);

figure(9)
stem(lag_plot,Ryx)
title("Cross-correlation Ryx[k]")


MSE = zeros(1, max_M-1);

for M = 2:max_M
    Rxx_m = toeplitz(Rxx(lag+1:lag+1+M)); % As the function in matlab gives both sides↙
we need to skip to lag 50 to be at 1
    Ryx_v = Ryx(lag+1:lag+M+1);

    h_est = Rxx_m \ Ryx_v;
```

```
    y_est = filter(h_est, 1, x)

    MSE(M-1) = immse(y_est, y)
end

plot_M = 2:max_M
figure(6)
plot(plot_M,MSE)
title('M vs MSE')
```