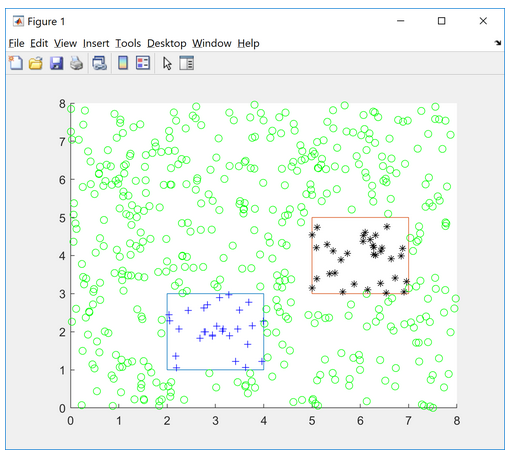
**Multivariate Parametric Classification**

1. Use code below to generate datasets for two classes C1 and C2. The true classes are within the bounding rectangles. The rest of the data points do not belong to any class.

2. From the generated data, train a parametric classifier assuming multivariate Gaussian distribution. Here you'll need to select the appropriate complexity of the model as discussed in Sections 5.5. and 5.6 of the book. I highly advise thinking on your own: in the past wrong ideas propagated through the class resulting in almost everyone getting a lower grade.

3. Discuss how the number of data points impacts parametric classifier's accuracy.

4. Visualize results obtained by the classifier on an independent test set drawn from the same classes C1 and C2 and discuss possible reasons for results being different (or not different)



**MATLAB code to generate data:**

xa=2; xb=4; ya=1; yb=3;         % coordinates of the rectangle C1

xa2=2; xb2=7; ya2=3; yb2=5;         % coordinates of the rectangle C2  
hold on; plot([xa xb xb xa xa],[ya ya yb yb ya],'-');    % draw it

plot([xa2 xb2 xb2 xa2 xa2],[ya2 ya2 yb2 yb2 ya2],'-');    % draw it

% generate positive and negative examples

N=500;   % no of data points

ds=zeros(N,2); ls=zeros(N,1);       % labels

for i=1:N

x=rand(1,1)\*8; y=rand(1,1)\*8;ds(i,1)=x; ds(i,2)=y;

% +ve if falls in the rectangle, -ve otherwise

if ((x > xa) && (y > ya) && (y < yb) && ( x < xb)) ls(i)=1; plot(x,y,'b+');

elseif ((x > xa2) && (y > ya2) && (y < yb2) && ( x < xb2)) ls(i)=2; plot(x,y,'k\*');

else ls(i)=0; plot(x,y,'go'); end;

end;