```
sx1 = zeros(n,1);  num1 = 0;
sKu = zeros(n,1);  Knum1 = 0;
for i = 1:p
       if lambnz(i) > 0
          sx1 = sx1 + u(:,i);
          num1 = num1 + 1;
       end
       if lamK(i) > 0
          sKu = sKu + u(:,i);
          Knum1 = Knum1 + 1;
       end
end
% Knum1
sx2 = zeros(n,1);  num2 = 0;
sKv = zeros(n,1);  Knum2 = 0;
for j = 1:q
       if munz(j) > 0
          sx2 = sx2 + v(:,j);
          num2 = num2 + 1;
       end
       if muK(j) > 0
          sKv = sKv + v(:,j);
          Knum2 = Knum2 + 1;
       end
end
% Knum2
% computes eta from the duality gap
errterm = w'*(sKv - sKu) + (pf - qf)*b;
Pterm = (1/K)*(lam'*P2*lam);
denom = (p+q)*nu - pf -qf;
fprintf('denom =  %.15f \n',denom)
epsilon = 0; xi = 0;
if denom > 0
   eta = (errterm + Pterm)/denom;
   fprintf('eta =  %.15f \n',eta)
   if eta < 10^(-10)
      fprintf('** Warning; eta is too small or negative ** \n')
   end

   tolxi = 10^(-10);
   % tols < lambda_i < K - tolh or K - tolh <= lambda_i and epsilon_i < tolxi
   [lamsv,psf,epsilon] = findsvl2(lamb,w,b,u,eta,K,tols,tolh,tolxi);
```