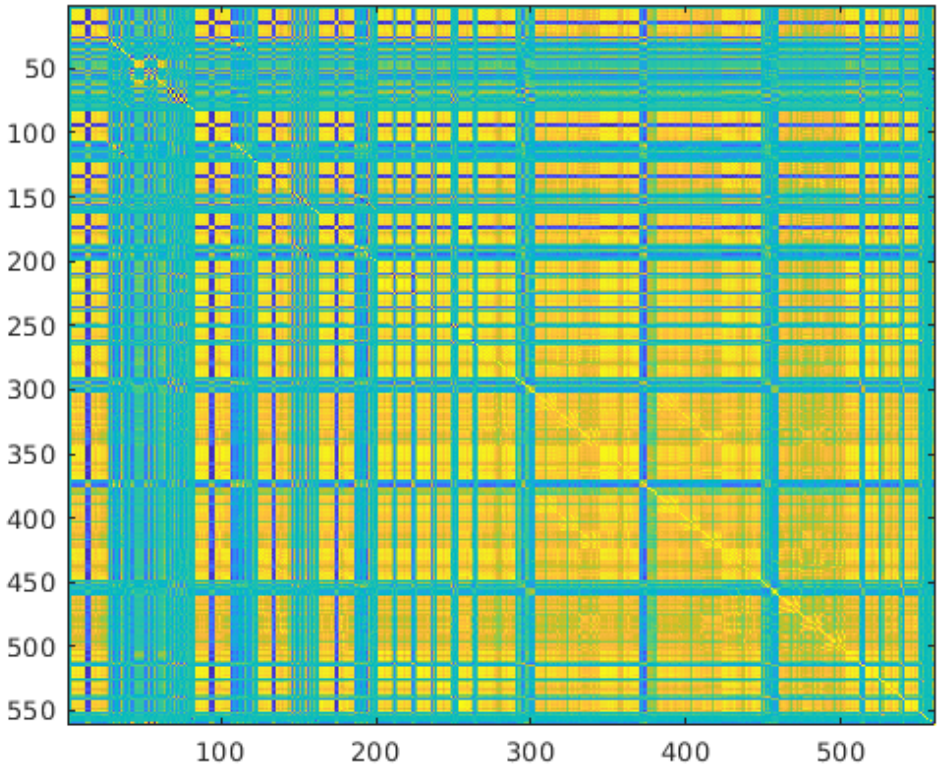
```
clear all
close all

load("HumanActivityRecognition.mat");
imagesc(corrcoef(X_train))

% Séparation en set d'entraînement, validation et test
[Xapp,Yapp,Xval,Yval]=splitdata(X_train,y_train,0.5);
%[Xtest,Ytest,Xval,Yval]=splitdata(Xtemp,Ytemp,0.5);

% Normalisation
[Xapp,Xval,meanapp,stdxapp]=normalizemeanstd(Xapp,Xval);
[Xtest]=normalizemeanstd(X_test,[],meanapp,stdxapp);
```



svm one against all

```
%   Learning and Learning Parameters
c = 0.5;
lambda = 1e-7;
kerneloption= 1;
kernel='poly';
verbose = 1;

%---------------------One Against All algorithms----------------
```

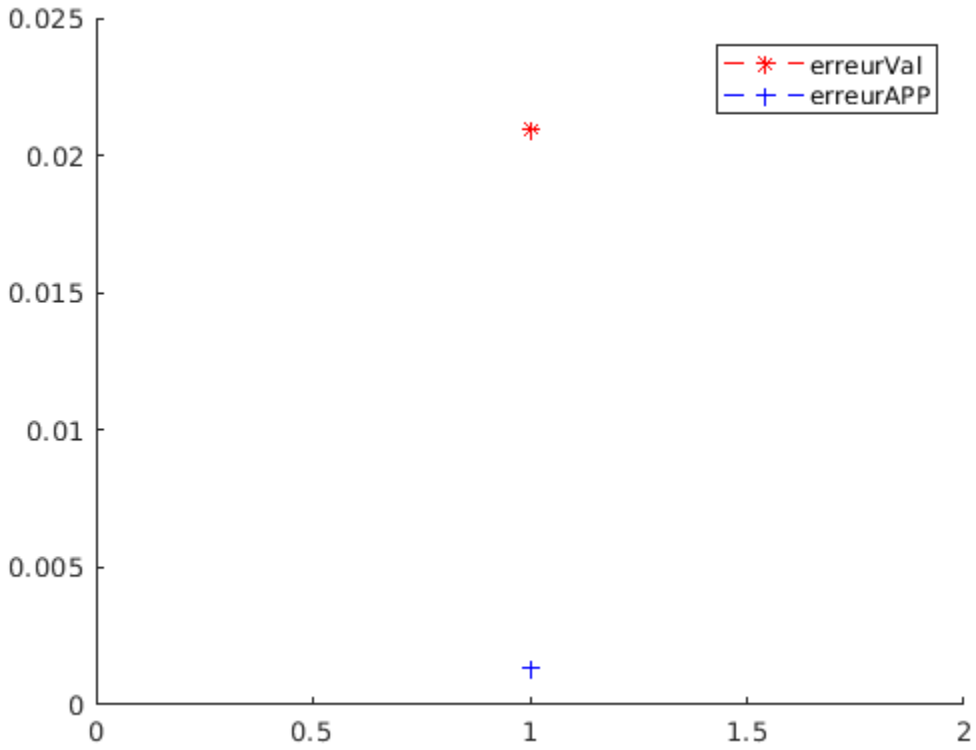
```
nbclass=6;
[xsup,w,b,nbsv,pos,obj]=svmmulticlassoneagainstall(Xapp,Yapp,nbclass,c,lambda,kern
```

Erreur validation

```
    ypred_val=svmmultival(Xval,xsup,w,b,nbsv,kernel,kerneloption);
    erreur_val = mean(Yval~=ypred_val);


    % Erreur apprentissage
    ypred_app=svmmultival(Xapp,xsup,w,b,nbsv,kernel,kerneloption);
    erreur_app = mean(Yapp~=ypred_app);


figure(2)
hold on
plot(erreur_val,'r--*');
plot(erreur_app,'b--+');
legend('erreurVal','erreurAPP')
hold off
```



Test

A faire uniquement si on doit estimer le C optimal

```
%[xsup,w,b,nbsv,pos,obj]=svmmulticlassoneagainstall(Xtest,Yapp,nbclass,c,lambda,ke
```

```
ypred=svmmultival(Xtest,xsup,w,b,nbsv,kernel,kerneloption);

cm = confusionmatrice(y_test, ypred)


cm =

   491     2     3     0     0     0
    19   450     2     0     0     0
     5     6   409     0     0     0
     0     3     0   431    52     5
     0     0     0    28   503     1
     0     0     0     0     3   534
```

Published with MATLAB® R2017b