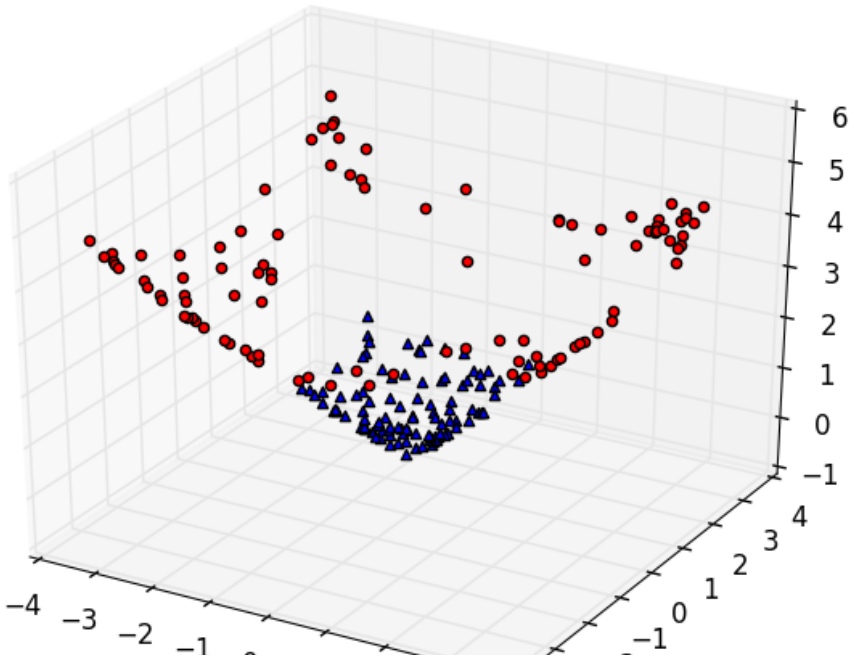
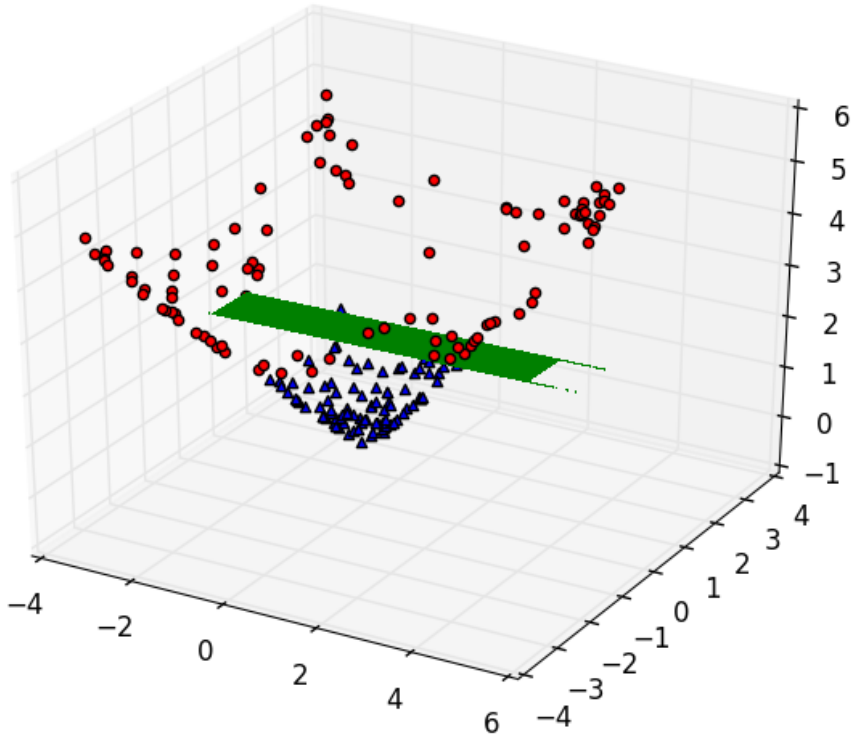
Kernel Trick for separating linearly inseparable data using Perceptron Let x,y be 2d data Kernels used are:
1)(X,Y,Z)=(x,y,x^2+y^2)
Accuracy is 1.000 and the plots are: 1)Data after changed and 2)The final hyperplane





English letter classification using Support Vector Machines

The kernels used were:
1)Linear 2)Polynomial and 3)RBF
And the hyperparameters tested were :
C, Gamma, degree and coef0

C is the parameter for the soft margin cost function.

Gamma is the free parameter of the Gaussian radial basis function.

1)RBF: RBG implicitly maps every point to an infinite dimensional space and the parameters used here are C, Gamma , degree and coef0.

Variation with C:

A low C makes the decision surface smooth, while a high C aims at classifying all training examples correctly by giving the model freedom to select more samples as support vectors. So we have to choose C which is not too high or too low.

Results:
for c= 1 --accuracy: 0.9563 , precision: 0.9563 , recall: 0.9563 , f1: 0.9563
for c= 10 --accuracy: 0.9734 , precision: 0.9734 , recall: 0.9734 , f1:0.9734
for c= 100 --accuracy: 0.9697 , precision: 0.9697 , recall: 0.9697 , f1:0.9697
for c= 1000 --accuracy: 0.971 , precision: 0.971 , recall: 0.971 , f1: 0.971

Variation with Gamma:If gamma is too large, the radius of the area of influence of the support vectors only includes the support vector itself and no amount of regularization with C will be able to prevent overfitting. When gamma is very small, the model is too constrained and cannot capture the complexity or “shape” of the data.

So we have to choose Gamma which is not too high or too low. Results:
for Gamma= 0.001 --accuracy: 0.8256 , precision: 0.8256 , recall: 0.8256 , f1: 0.8256
for Gamma= 0.01 --accuracy: 0.9204 , precision: 0.9204 , recall: 0.9204 , f1: 0.9204
for Gamma= 0.1 --accuracy: 0.9731 , precision: 0.9731 , recall: 0.9731 , f1: 0.9731
for Gamma= 1 --accuracy: 0.9415 , precision: 0.9415 , recall: 0.9415 , f1: 0.9415
for Gamma= 10 --accuracy: 0.288 , precision: 0.288 , recall: 0.288 , f1: 0.288

. 2)Polynomial:
from the formula we can say that the kernel depends on all the above parameters. Polynomial: (gamma*u'*v + coef0)^degree (using libsvm's nomenclature)

Variation with C:
Results:
for C= 1 --accuracy: 0.9292 , precision: 0.9292 , recall: 0.9292 , f1: 0.9292
for C= 10 --accuracy: 0.952 , precision: 0.952 , recall: 0.952 , f1: 0.952
for C= 100 --accuracy: 0.948 , precision: 0.948 , recall: 0.948 , f1: 0.948
for C= 1000 --accuracy: 0.942 , precision: 0.942 , recall: 0.942 , f1: 0.942
So intermediate c values gives best results. Variation with Gamma:

Results:
for Gamma= 0.01 --accuracy: 0.6059 , precision: 0.6059 , recall: 0.6059 , f1: 0.6059
for Gamma= 0.1 --accuracy: 0.9494 , precision: 0.9494 , recall: 0.9494 , f1: 0.9494
for Gamma= 1 --accuracy: 0.9418 , precision: 0.9418 , recall: 0.9418 , f1: 0.9418
for Gamma= 10 --accuracy: 0.9453 , precision: 0.9453 , recall: 0.9453 , f1: 0.9453 similarly intermediate gamma values gives best results.

Variation with Degree:
Variation with degree is data specific, in our data it decreases with degree.
Results:

Degree 3 --accuracy: 0.9526 , precision: 0.9526 , recall: 0.9526 , f1: 0.9526
for Degree= 5 --accuracy: 0.9062 , precision: 0.9062 , recall: 0.9062 , f1: 0.9062
for Degree= 7 --accuracy: 0.8516 , precision: 0.8516 , recall: 0.8516 , f1: 0.8516
for Degree= 10 --accuracy: 0.7764 , precision: 0.7764 , recall: 0.7764 , f1: 0.7764

Variation with coef0: For this dataset, intermediate coef0 values gives best results.
Results:
for Coef0= -1 --accuracy: 0.2187 , precision: 0.2187 , recall: 0.2187 , f1: 0.2187
for Coef0= 0.01 --accuracy: 0.9532 , precision: 0.9532 , recall: 0.9532 , f1: 0.9532
for Coef0= 0 --accuracy: 0.9525 , precision: 0.9525 , recall: 0.9525 , f1: 0.9525
for Coef0= 0.1 --accuracy: 0.9599 , precision: 0.9599 , recall: 0.9599 , f1: 0.9599
for Coef0= 1 --accuracy: 0.9621 , precision: 0.9621 , recall: 0.9621 , f1: 0.9621
for Coef0= 5 --accuracy: 0.9565 , precision: 0.9565 , recall: 0.9565 , f1: 0.9565
for Coef0= 10 --accuracy: 0.9537 , precision: 0.9537 , recall: 0.9537 , f1: 0.9537

3)Linear: For linear kernel the parameter is C as linear: . It tries to seperate data linearly. Variation with C.
Results:
for C= 1 --accuracy: 0.8573 , precision: 0.8573 , recall: 0.8573 , f1: 0.8573
for C= 10 --accuracy: 0.8531 , precision: 0.8531 , recall: 0.8531 , f1: 0.8531
for C= 100 --accuracy: 0.8503 , precision: 0.8503 , recall: 0.8503 , f1: 0.8503
So here accuracy decreases with C.