

```

svm_save_model(m, m)
a = m.nSV
c = m.sv_coef

p_label, p_acc, p_val = svm_predict(TransformY(y), x, m)
b = m.rho

w = (m.sv_coef* m.SV)
w_abs = math.sqrt(sum(w**2))
distance = abs((p_val) / (w_abs-b))

#####

#####Q12,13#####
print("")
for i in range(2,7):
    VALUE = i
    prob = svm_problem(TransformY(y), x)
    param =svm_parameter('-s 0 -t 1 -c 10 -d 3 -g 1 -r 1')
    m = svm_train(prob,param)
    #Ein so use train data
    p_label, p_acc, p_val = svm_predict(TransformY(y), x, m)

#####

#####Q14#####
print("")
C= [0.01,0.1,1,10,100]
for C in C:
    VALUE =1
    prob = svm_problem(TransformY(y), x)
    param =svm_parameter('-s 0 -t 2 -c {} -g 10 '.format(C))
    m = svm_train(prob,param)
    p_label, p_acc, p_val = svm_predict(TransformY(y_t), x_t, m)

#####

#####Q15#####
print("")
gamma=[0.1,1,10,100,1000]
for r in gamma:
    VALUE =1
    prob = svm_problem(TransformY(y), x)
    param =svm_parameter('-s 0 -t 2 -c 0.1 -g {} '.format(r))
    m = svm_train(prob,param)
    p_label, p_acc, p_val = svm_predict(TransformY(y_t), x_t, m)

#####

#####Q16#####

print("")
y = TransformY(y)
y, x = np.array(y), np.array(x)
#y_t,x_t =np.array(y_t), np.array(x_t)
q16=[0]*5
for i in range(10):
    print('times: ',i)
    choice= np.random.choice(len(x), 200,False)
    nochoice= np.delete(np.array(np.arange(len(x))), choice)

    x_t1 = x[choice]
    y_t1 = y[choice]

```