CS 435 FALL 2011 ASSIGNMENT-2 / UMUT CAN GENLIK/ 10.25.2011

1. For first question output of a execution process is

umut-can-genliks-MacBook-Pro:libsvm-simple ugenlik1$ make

g++ -Wall -Wconversion -O3 -fPIC -c svm.cpp

svm.cpp:1831: warning: ‘void svm\_binary\_svc\_probability(const svm\_problem\*, const svm\_parameter\*, double, double, double&, double&)’ defined but not used

svm.cpp:1917: warning: ‘double svm\_svr\_probability(const svm\_problem\*, const svm\_parameter\*)’ defined but not used

g++ -Wall -Wconversion -O3 -fPIC svm-train.c svm.o -o svm-train -lm

g++ -Wall -Wconversion -O3 -fPIC svm-predict.c svm.o -o svm-predict -lm

g++ -Wall -Wconversion -O3 -fPIC svm-scale.c -o svm-scale

1. My function is included in my program .
2. Since the program I have created puts selected data in two separate output texts, one training set and one test set, created output files for percentages 30,50 and 70 is look like following

umut-can-genliks-MacBook-Pro:libsvm-simple ugenlik1$ ./svm-train ../output/rands30.txt ../output/model30.txt

.\*

**optimization finished, #iter = 333**

**nu = 0.879017**

**obj = -101.336567, rho = 0.210647**

**nSV = 207, nBSV = 93**

**Total nSV = 207**

umut-can-genliks-MacBook-Pro:libsvm-simple ugenlik1$ ./svm-train ../output/rands50.txt ../output/model50.txt.

**\*optimization finished, #iter = 571**

**nu = 0.885801**

**obj = -167.267778, rho = -0.170485**

**nSV = 345, nBSV = 152**

**Total nSV = 345**

umut-can-genliks-MacBook-Pro:libsvm-simple ugenlik1$ ./svm-train ../output/rands70.txt ../output/model70.txt

.\*

**optimization finished, #iter = 493**

**nu = 0.835290**

**obj = -167.595438, rho = -0.248962**

**nSV = 355, nBSV = 148**

**Total nSV = 355**

Since above results received from “australina.txt” input when we apply svm-predict methods we get results for classification which are look like following

**For %30**

umut-can-genliks-MacBook-Pro:libsvm-simple ugenlik1$ ./svm-predict ../output/other30.txt ../output/model30.txt ../output/output30.txt

**Accuracy = 55.4865% (268/483) (classification)**

**For %50**

umut-can-genliks-MacBook-Pro:libsvm-simple ugenlik1$ ./svm-predict ../output/other50.txt ../output/model50.txt ../output/output50.txt

**Accuracy = 55.0725% (190/345) (classification)**

**For %70**

umut-can-genliks-MacBook-Pro:libsvm-simple ugenlik1$ ./svm-predict ../output/other70.txt ../output/model70.txt ../output/output70.txt

**Accuracy = 51.9403% (174/335) (classification)**

My first observation of accuracy outputs of given data is when percentage increases, accuracy decreases,. Because of predefined classes in classification , amount of the data is crucial ro run such training and predictions. However there is always an error rate in any algorithms ,as well in my algorithm, using more amounts of data can lead us clearer view of accuracy so we can also see how well the algorithm is according to ROC table. But if I assume my algorithm good enough, with more data and less percentage selection , accuracy could be optimum.