Homework #4 Solution

Assign date: 2015-07-24

Due date: 2015-08-~~02~~04, 6pm

Submission:

1. **Please submit your results <u>in email</u> to the grader: 130301039@svuca.edu Chi Zhang**
2. **Please separate the written answers from the python code: you should submit 1 file in your email – cs596-29-hw4_yourID#.doc**
3. **30 pts per day will be deducted for late submission**

Problem 1) SVM classifier (15 pts.)

2 linear SVC, SVC-1 & SVC-2, are constructed with C=0.001 and C=1 to classify the data points. The results from svm-gui.py are shown in Figures 1 and 2. Answer the following questions.

a) How many support vectors are identified for SVC-1? How many are for SVC-2?
b) How many data points are miss-classified by SVC-1? How many are by SVC-2?
c) What is most likely the C-value for SVC-1? What is the likely the C-value for SVC-2?

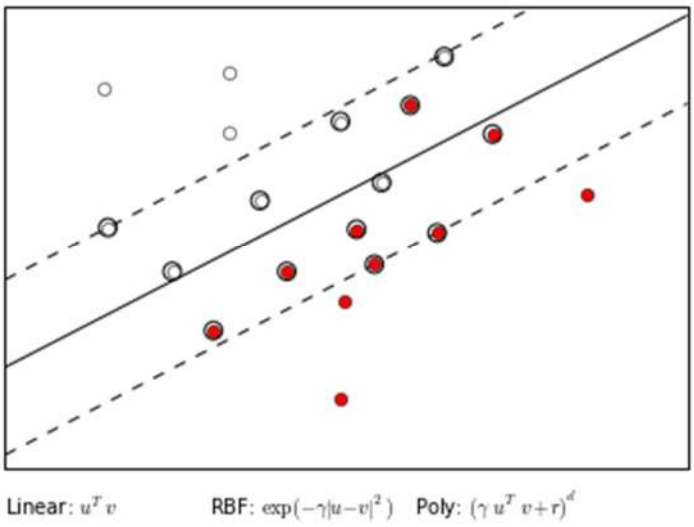

Figure 1. SVC-1

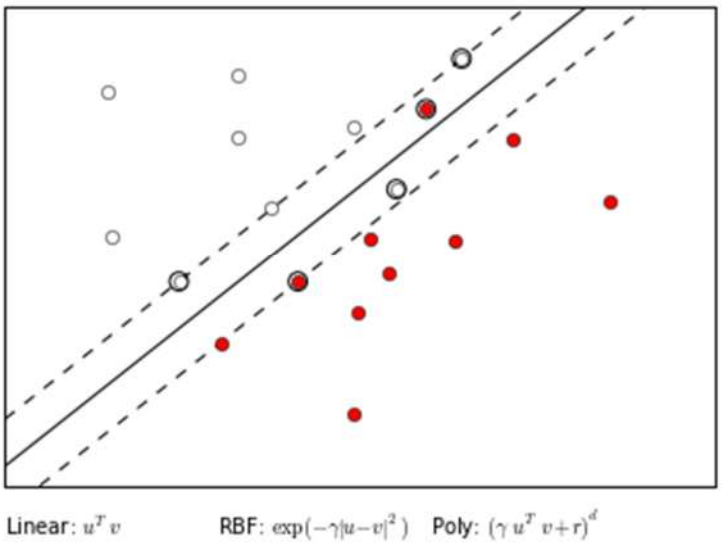

Figure 2. SVC-2

Answers:

a) SVC-1: 13, SVC-2: 5
b) SVC-1: 2, SVC-2: 2
c) SVC-1: 0.001, SVC-2: 1

Problem 2) SVM classifier (10 pts.)

Construct a SVM classifier, either manually or using sk-learn, that can separate the 2 classes defined by the following table. Answer the following questions:

a) First with the linear SVC. What is the best accuracy that can be achieved?
b) Next try with RBF kernel. What is the best accuracy that you can achieve?

X_1	1	-1	-1	1
X_2	1	-1	1	-1
T =	-1	-1	1	1

Answers:

a) 50%
b) 100%

Problem 3) SVM classifier (25 pts.)

Construct a linear SVM classifier, either manually or using sk-learn, that can separate the 2 classes defined by the following table. Based on your classifier, answer the following questions:

a) What are the linear classifier coefficients?
b) What are the support vectors?
c) What is the margin?

X_1	1	3	1	7	9	9
X_2	2	4	4	8	10	8
T =	0	0	0	1	1	1

Answers:

a) [[0.25 0.25]] [-2.75] or 0.25x + 0.25y – 2.75 = 0
b) (3, 4) and (7, 8)
c) $4\sqrt{2}$

Problem 4) Bagging (25 pts.)

Create, either manually or using sk-learn, a Bagging ensemble based on single level decision tree (stump) classifier to classify the training data defined by the following table.

X_1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
X_2	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
y =	1	1	1	0	0	0	1	1	1	1

To construct the ensemble, 9 classifiers will be created using the following sampling sequence:

[[5 0 3 3 7 9 3 5 2 4]
[7 6 8 8 1 6 7 7 8 1]
[5 9 8 9 4 3 0 3 5 0]
[2 3 8 1 3 3 3 7 0 1]
[9 9 0 4 7 3 2 7 2 0]
[0 4 5 5 6 8 4 1 4 9]
[8 1 1 7 9 9 3 6 7 2]
[0 3 5 9 4 4 6 4 4 3]
[4 4 8 4 3 7 5 5 0 1]]

Each row corresponds to an iteration, and each item in a sequence represents the data point. For example, the 1st row [5 0 3 3 7 9 3 5 2 4] means that points 5 and 3 would be used more than once, and points 1, 6, 8 would not be used in the 1st iteration.

a) Please report the array of targets predicted by the ensemble based on the training data. An example table is given below.

[[1 1 1 1 1 1 0 0 0 0]

[1 1 1 1 1 1 0 0 0 0]

[0 0 0 0 0 0 0 0 1 1]

[1 1 1 1 1 1 1 0 0 0]

[1 1 1 1 1 1 1 0 0 0]

[1 1 1 1 1 0 0 0 0 0]

[0 0 1 1 1 1 1 1 1 1]

[1 1 1 1 1 1 0 0 0 0]

[0 0 0 1 1 1 1 1 1 1]]

b) Assuming majority vote, what is the final accuracy of your bagging ensemble for the training data?

c) Change the target vector to [1 1 1 0 0 0 0 0 1 1] but keep the X values the same. Rerun your bagging ensemble. What is the final accuracy of your bagging ensemble for the new training data?

Answers:

a) Bagging decision tree with stump (depth=1)

Sampling sequence:

[[5 0 3 3 7 9 3 5 2 4]

[7 6 8 8 1 6 7 7 8 1]

[5 9 8 9 4 3 0 3 5 0]

[2 3 8 1 3 3 3 7 0 1]

[9 9 0 4 7 3 2 7 2 0]

[0 4 5 5 6 8 4 1 4 9]

[8 1 1 7 9 9 3 6 7 2]

[0 3 5 9 4 4 6 4 4 3]

[4 4 8 4 3 7 5 5 0 1]]

Iteration: 0

Samples: [5 0 3 3 7 9 3 5 2 4]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [1 1 1 0 0 0 0 0 0 0]

Tree score = 0.6

Iteration: 1

Samples: [7 6 8 8 1 6 7 7 8 1]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [1 1 1 1 1 1 1 1 1 1]

Tree score = 0.7

Iteration: 2

Samples: [5 9 8 9 4 3 0 3 5 0]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [0 0 0 0 0 0 0 1 1 1]

Tree score = 0.6

Iteration: 3

Samples: [2 3 8 1 3 3 3 7 0 1]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [1 1 1 0 0 0 0 0 0 0]

Tree score = 0.6

Iteration: 4

Samples: [9 9 0 4 7 3 2 7 2 0]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [1 1 1 1 1 1 1 1 1 1]

Tree score = 0.7

Iteration: 5

Samples: [0 4 5 5 6 8 4 1 4 9]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [0 0 0 0 0 0 0 1 1 1]

Tree score = 0.7

Iteration: 6

Samples: [8 1 1 7 9 9 3 6 7 2]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [1 1 1 1 1 1 1 1 1 1]

Tree score = 0.7

Iteration: 7

Samples: [0 3 5 9 4 4 6 4 4 3]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [0 0 0 0 0 0 1 1 1 1]

Tree score = 0.7

Iteration: 8

Samples: [4 4 8 4 3 7 5 5 0 1]

target: [1 1 1 0 0 0 1 1 1 1]

Tree prediction: [1 1 1 0 0 0 0 0 0 0]

Tree score = 0.6

Tree predictions:

[[1 1 1 0 0 0 0 0 0 0]

[1 1 1 1 1 1 1 1 1 1]

[0 0 0 0 0 0 0 1 1 1]

[1 1 1 0 0 0 0 0 0 0]

[1 1 1 1 1 1 1 1 1 1]

[0 0 0 0 0 0 1 1 1 1]

[1 1 1 1 1 1 1 1 1 1]

[0 0 0 0 0 0 1 1 1 1]

[1 1 1 0 0 0 0 0 0 0]]

b) Bagging: [1 1 1 0 0 0 1 1 1 1]

Target: [1 1 1 0 0 0 1 1 1 1]

Score = 1.0

c) Score = 0.8

Problem 5) Random Forest (25 pts.)

Use the same training data as in Problem-4. Create a Random Forest classifier using sklearn that has 9 estimators with all default values. Answer the following questions.

a) For targets = [1 1 1 0 0 0 1 1 1 1], what is the accuracy of the random forest?
b) For targets = [1 1 1 0 0 0 0 0 1 1], what is the accuracy of the random forest?
c) Give your opinions on why Random Forest is doing better or worse than Bagging.
d) Given the test data as follows, what are the prediction from the random forest classifier trained using the targets in (a)?

X_1	0.9	0.8	0.7	0.6	0.4	0.3	0.2	0.1
X_2	0.1	0.2	0.3	0.4	0.6	0.7	0.8	0.9

Answers:

a) Score = 1.0
b) Score = 1.0, or 0.8 if you set *max_depth=1*
c) Random Forest is doing better in this case. The main reason is the random forest with default values uses a normal decision tree without depth constraint, as compared to the stump used in the previous problem. Or, the other argument is that even if stump is used in RandomForestClassifier, it may still performs better in some cases because the feature selection is randomized.
d) [1 1 1 0 0 1 1 1]