hw3

Hongyi Guo 516030910306

March 22, 2019

1

`make` in the libsvm folder and `make` in the python folder.

2

Grid search parameter c and γ in $\{2^{-10}, 2^{-9}, \ldots, 2^{10}\}$. The i^{th} line means c equals to 2^{i-10}. The j^{th} column means γ equals to 2^{j-10}. Note that the results are tested on test set for convenience. I do not use a validation set although that would be better from my perspective.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
1	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
2	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
3	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
4	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
5	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
6	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
7	0.6822...	0.6822...	0.7638...	0.6822...	0.6822...	0.8309...	0.8309...	0.6822...	0.6822...	0.8309...	0.83090...	0.83090...	0.83090...	0.68221...	0.83090...	0.83090...	0.83090...	0.68221...	0.68221...	0.71720...	0.68221...
8	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
9	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
10	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
11	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
12	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
13	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
14	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
15	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
16	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
17	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
18	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
19	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...
20	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...	0.68221...

Figure 1: One-versus-rest with linear kernel

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0.6822...	0.3381...	0.3381...	0.3177...	0.3177...	0.3177...	0.3381...	0.3177...	0.3177...	0.3381...	0.3177...	0.3177...	0.31778...	0.3381...	0.31778...	0.3381...	0.3177...	0.3177...	0.31778...	0.3177...	0.317...
1	0.6618...	0.6618...	0.6822...	0.3381...	0.3177...	0.3381...	0.3440...	0.3440...	0.3440...	0.3440...	0.6822...	0.3381...	0.31778...	0.3177...	0.31778...	0.3440...	0.3381...	0.3381...	0.3440...	0.6822...	0.34...
2	0.6618...	0.6618...	0.6822...	0.5131...	0.5131...	0.6618...	0.3440...	0.6822...	0.5131...	0.3440...	0.6618...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.3440...	0.6822...	0.66180...	0.3440...	0.34...
3	0.6822...	0.6822...	0.6822...	0.6822...	0.6618...	0.6618...	0.6822...	0.6618...	0.6822...	0.6822...	0.6822...	0.6822...	0.6822...	0.6618...	0.6618...	0.6618...	0.6618...	0.6618...	0.66180...	0.6822...	0.66...
4	0.6822...	0.6618...	0.6822...	0.6618...	0.6618...	0.6618...	0.6822...	0.6822...	0.7551...	0.6822...	0.6618...	0.6822...	0.6618...	0.6822...	0.6618...	0.6822...	0.6822...	0.6822...	0.66180...	0.6618...	0.68...
5	0.6618...	0.6618...	0.8309...	0.6822...	0.8309...	0.6822...	0.6618...	0.6618...	0.6618...	0.8309...	0.6618...	0.6618...	0.6618...	0.6822...	0.6618...	0.6618...	0.6822...	0.6822...	0.68221...	0.6822...	0.83...
6	0.6822...	0.8309...	0.8309...	0.6822...	0.6880...	0.8309...	0.6822...	0.6822...	0.6822...	0.8309...	0.6822...	0.8309...	0.6822...	0.8309...	0.6822...	0.6822...	0.8309...	0.6822...	0.8309...	0.6822...	0.68...
7	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
8	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
9	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
10	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
11	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
12	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
13	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
14	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
15	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
16	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
17	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
18	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
19	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...
20	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.8309...	0.83...

Figure 2: One-versus-one with linear kernel

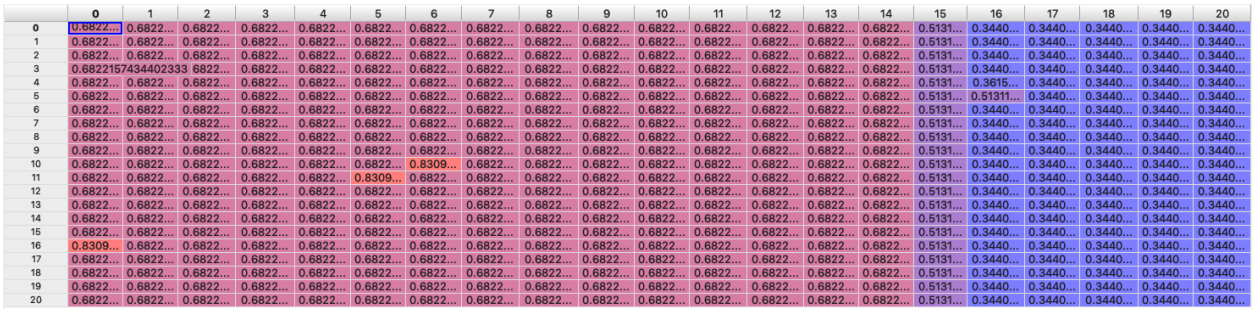

Figure 3: One-versus-rest with RBF kernel

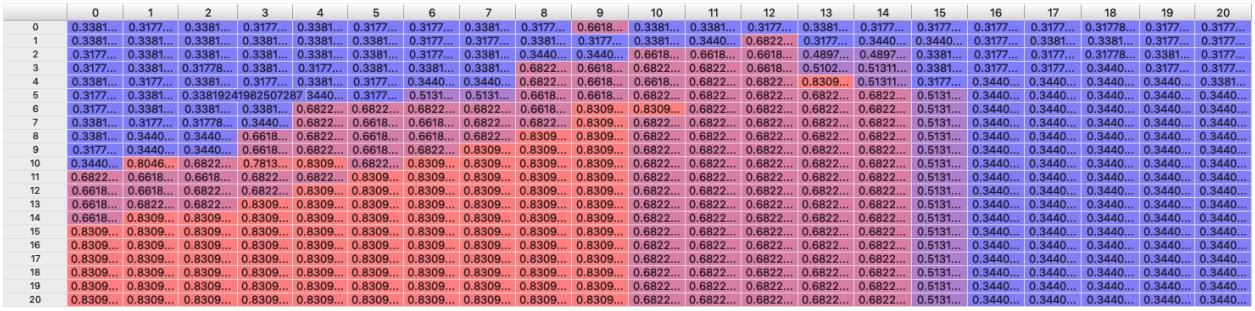

Figure 4: One-versus-one with RBF kernel

3

Results are as in Fig.1, Fig.2, Fig.3, Fig.4

4

4.1 OvR v.s. OvO

One-versus-rest method runs faster than one-versus-one when there are more classes with no doubt. But in our case where only three classes are considered, the total number of models used in both methods are equal (and are 3). But because each model of the one-versus-rest method needs all data from training set. It may trains slightly slower than one-versus-one method.

The biggest problem in OvO is that the voting may cause a tie, which means -1 defeats 0, 0 defeats 1 and 1 defeats -1. This is rare but not impossible. OvR use the probability(confidence score) to vote thus doesn't have to consider ties but suffers from the problem of unbalanced distributions of 'one' and 'rest'.

In my experiment, OvO always outperforms OvR given any kernel.

4.2 Linear v.s. RBF

Typically, RBF kernel should outperform linear kernel because it's more complex and can handle more difficult situation. In my experiment, both linear kernel and RBF kernel can

reach a score of about 83.1%.

From the visualization in Fig.5, -1 and 0 are very close in distribution. After deeper inspectation, I find that both linear and RBF kernel fails to classify -1 and 0 in the test set. I think the training set may not be large enough to build a very rebust classifier.

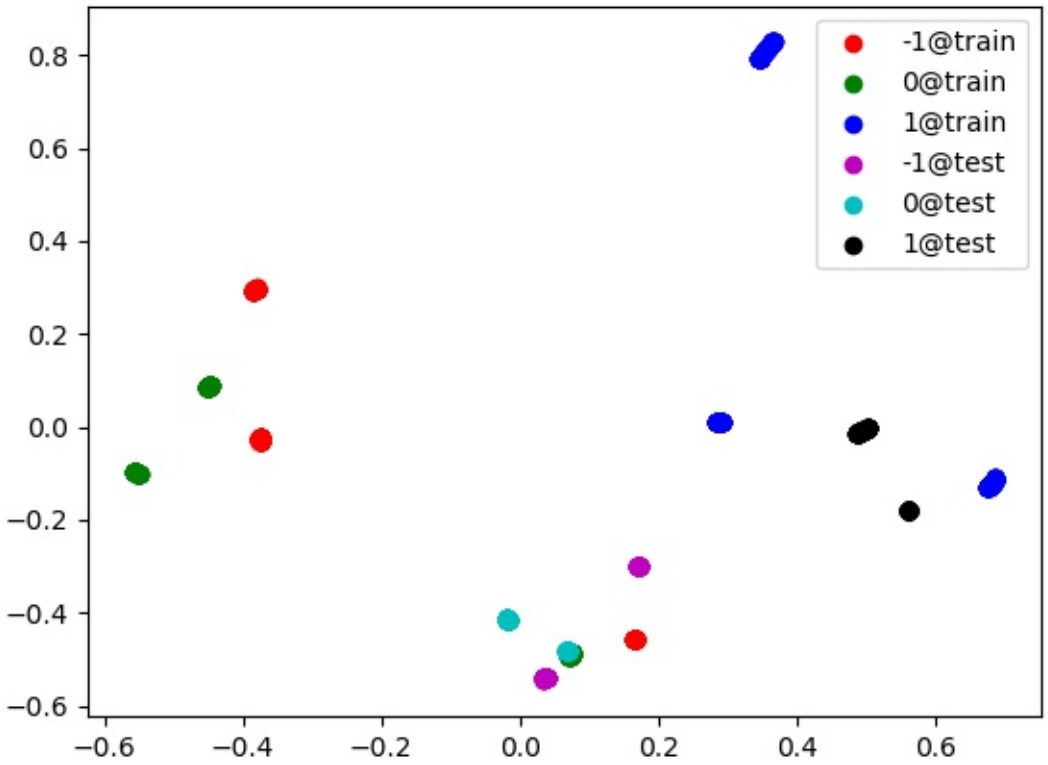

Figure 5: Data distribution