CSE 5526 - Autumn 2020 Introduction to Neural Networks Programming Assignment 3

A Preprint

Ibrahim M. Koc
Department of Electrical and Computer Engineering
The Ohio State University
Columbus, OH 43201
koc.15@osu.edu

November 10, 2020

1 Discussion

In this example, we have a RNA sequence that we would like to find out novel non-coding RNA sequences (ncRNA). Since it is difficult to detect novel ncRNA in biochemical screening, we would like to a supervised classification method. SVM is a possible candidate for this task.

The main idea of SVM is to formulate classification problem such that the objective function does not just become a separation between classes, but an optimal separating hyperplane that is equally (optimally) all classes' given instances in training. Training instances closest to the optimal hyperplane are called support vectors. By writing the equations for finding the optimal hyperplane it can be seen that optimal hyperplane can be found by minimizing the both the inner product of weight vector and the classification error, that is, the inner product of weight vector should be minimized for maximizing the margin of separation.

Rewriting the terms constitutes the dual problem. And one can find the optimal hyperplane for linearly separable classification problems using dual problem. However, for linearly inseparable problems, $d_i(w^T wx_i + b) \geq 1$ will be violated for some cases. Therefore, the margin of separation now becomes soft (i.e., cannot be enforced for hold all cases in the optimization problem while still penalizing deviations from it). One can introduce a set of nonnegative variables called slack variables to satisfy the equation $d_i(w^T wx_i + b) \geq 1 - \xi_i$.

Later, a better idea was discovered for linearly inseparable problems. By Cover's theorem, it is stated that a high-dimensional convoluted classification problem can be mapped to a low-dimensional linearly separable problem by an inner product kernel. Furthermore, with the idea called kernel trick, if the inner product kernel is defined, there is no need to train for the optimal weight vector. However, it should be noted that kernel matrix formed by individual kernel inner product mapping should be positive semidefinite.

For the coding part a popular open source library called libsvm will be used. For more information, please use http://www.csie.ntu.edu.tw/~cjlin/libsvm/. As suggested by the authors of the library, for the kernel trick a RBF kernel will be used.

2 Results

First a linear SVM classifier is constructed. Since both the training and the test data are scaled there is no need to scale it. To find the optimal C value C is iterated from 2^{-4} to 2^8 The resulting accuracy characteristics with varying cost value C is demonstrated in Fig. 1 in log scale. As one can see, the accuracy saturates after $C = 2^2 = 4$ since this classification problem is probably a complex high-dimensional classification problem and perfect linear separation is not possible. However, even with a linear-SVM an accuracy of 93.8% is achieved.

Then, the kernel trick with a RBF kernel is used to compare the performances. However, for RBF kernel there is one more additional term, which is γ from equation $\exp(-\gamma||x - x_i||^2)$. Therefore, using the training data, 5-fold

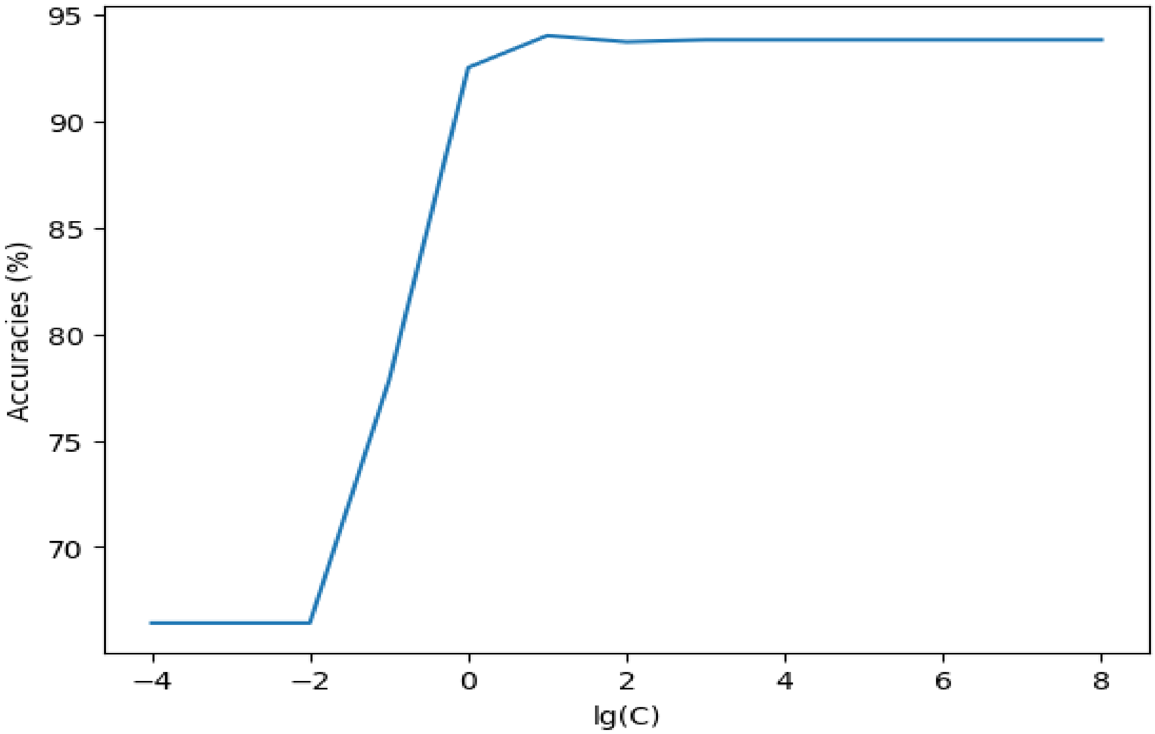

Figure 1: Testing Accuracy with different cost values

cross-validation is used. N-fold cross-validation means separating the training set randomly into N sets and randomly choosing $N-1$ sets to train the SVM and use the remaining set to calculate accuracy. Then, the cross-validation accuracy is calculated by the average of N accuracy values.

Using the specified 5-fold cross-validation, the highest validation accuracy obtained is 95.40% at 12^{th} row and 1^{st} column, which means the best $C = 128.0$ and the best $\gamma = 0.0625$ .Highest accuracy using best gamma and C on test data is 94.2058%. However, as one can see from Fig. 2, which is the graph provided by the built-in algorithm to grid search for the accuracy values that the highest accuracy possible which is 95% can be obtained around the region of $(C, \gamma) = (2^5 \leq C_{best} \leq 2^8, 2^{-2} \leq \gamma_{best} \leq 2^{-4})$. Therefore, it is verified that the algorithm written here gets a similar result to the library's built-in funciton.

Here you can see the resulting accuracy table from cross validation.

$$\begin{bmatrix}
93.80 & 92.10 & 94.70 & 93.60 & 91.60 & 94.40 & 91.90 & 93.40 & 91.50 & 94.80 & 94.30 & 94.50 & 92.20 \\
91.90 & 92.00 & 91.40 & 91.60 & 94.70 & 91.50 & 92.00 & 94.00 & 91.80 & 91.30 & 91.90 & 92.00 & 92.20 \\
91.40 & 92.10 & 94.10 & 92.10 & 91.70 & 94.80 & 91.40 & 91.40 & 93.90 & 94.80 & 92.20 & 92.00 & 94.60 \\
94.40 & 94.10 & 94.60 & 91.60 & 93.80 & 94.40 & 91.60 & 91.60 & 91.30 & 91.30 & 94.50 & 94.20 & 91.80 \\
92.20 & 91.80 & 94.00 & 94.60 & 92.20 & 92.20 & 91.90 & 91.90 & 91.60 & 91.80 & 92.00 & 94.60 & 94.40 \\
94.30 & 93.80 & 92.20 & 91.90 & 91.70 & 92.00 & 91.00 & 93.00 & 94.00 & 94.60 & 94.30 & 91.80 & 92.00 \\
92.00 & 90.80 & 94.90 & 93.30 & 93.40 & 94.50 & 94.70 & 94.50 & 92.40 & 93.50 & 92.10 & 91.70 & 94.60 \\
93.80 & 93.80 & 92.00 & 94.10 & 91.90 & 92.30 & 92.30 & 91.70 & 94.00 & 91.90 & 92.20 & 93.70 & 94.40 \\
92.20 & 94.50 & 92.10 & 94.30 & 91.90 & 91.90 & 94.00 & 93.30 & 94.20 & 93.10 & 94.40 & 92.20 & 91.90 \\
92.00 & 91.50 & 94.30 & 94.30 & 94.80 & 91.90 & 94.70 & 92.00 & 94.60 & 94.50 & 91.80 & 91.60 & 92.00 \\
94.20 & 92.00 & 91.10 & 94.40 & 91.50 & 91.80 & 94.20 & 94.80 & 91.60 & 95.20 & 94.00 & 94.00 & 94.50 \\
95.40 & 94.90 & 91.90 & 91.90 & 94.30 & 94.40 & 94.90 & 93.70 & 92.00 & 91.60 & 92.10 & 94.90 & 94.80 \\
94.50 & 91.60 & 92.10 & 94.40 & 94.20 & 92.20 & 92.10 & 91.70 & 91.70 & 92.20 & 92.00 & 91.90 & 94.90
\end{bmatrix}$$

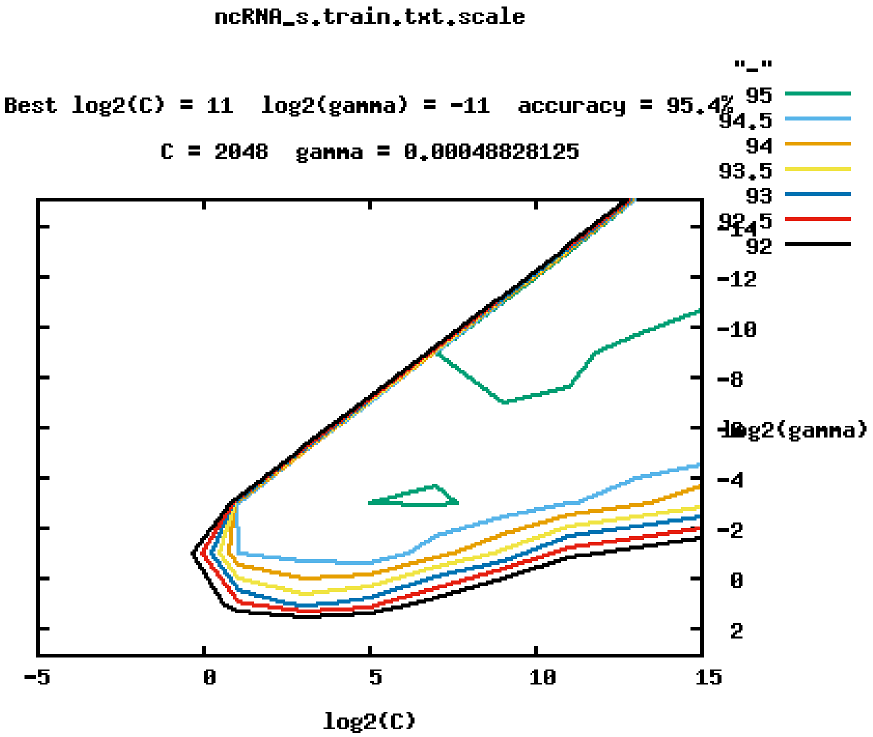

Figure 2: Testing Accuracy using built-in gamma C grid search results

3 Conclusion

As one can from the result for the linear-SVM case, tuning the hyper-parameter indeed helps. However, there is an accuracy limit to the classifier since the problem that is dealt here is most likely to be a nonlinear problem.

On the other hand, since RBF kernel is a nonlinear mapping, it is useful to reduce the dimentionality of the convoluted problem to increase the accuracy. Furthermore, fine-tuninng the hyper-parameters using 5-fold cross-validation also helped to determine the most suitable values for the highest accuracy.

For project's github link: `https://github.com/mertkoc/simpleNeuralNetwork` (I will upload the code to github after the submission due date ends). Furthermore, please note that the provided "lab3.py" python script should be inside "tools" folder of main libsvm library's github repository (e.g., I have used libsvm-3.24 so "lab3.py" should be inside ../libsvm-3.24/tools/).