Programming Report:

We are solving the Support Vector Machine problem given in the form:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2$$
$$\text{s.t. } 1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \leq 0, \forall i$$

Where w is the weight put on each component of x, x_i's are training data, here the sample number N is 120, and each x_i is a 4d array representing an Iris plant with floating number attributes "Sepal length", "Sepal width", "Petal length", "Petal width", and y_i is an integer which is the class attribute of the plant (taking value 0,1,2) for each training data.

In the programming part, we want to build a classification model that separates the 3 classes of plants based on their length and width attributes and can predict the class of a newly given Iris plant if its corresponding length and width attributes are given. We adopted the Support Vector Machine method for the whole question and realized it based on sklearn, where we built the SVC model using its OneVsRestClassifier. In Q1 to Q3, we want to test the accuracy of different kernels with or without slack variables using 30 testing data.

In Q1, we adopted linear kernel without slack variables, the training error is 0.042 and testing error is 0, which means that the linear model performs well on testing set. However, the error in training set implies that the 3 types of data is not linearly separable. By **checking the performance of each SVC** in the OneVsRestClassifier, we **recorded the correctly predicted number of plants of each type in the training data set**, and used them to determine whether a type is linearly separable from the other two. The result was:

```
Class setosa linearly separable: True
Class versicolor linearly separable: False
Class virginica linearly separable: False
```

That is, only the type setosa (or class 0) is linearly separable from the other 2 types.

In Q2, we adopted linear kernel with slack variables, where we set parameter C to 0,1 * [1,10], i.e. the degree of slackness decreases. We then calculated the training and testing error for each C, the results were:

C=0.100

training_error:0.125

testing_error:0.233

C=0.200

training_error:0.058

testing_error:0.167

C=0.300

training_error:0.050

testing_error:0.133

C=0.400

training_error:0.050

testing_error:0.100

C=0.500

training_error:0.050

testing_error:0.100

C=0.600

training_error:0.050

testing_error:0.100

C=0.700

training_error:0.050

testing_error:0.100

C=0.800

training_error:0.050

testing_error:0.100

C=0.900
training_error:0.050
testing_error:0.067

C=1.000
training_error:0.050
testing_error:0.067

We observed that overall the linear model with slack variables performed well, and as the slackness decreased, both the training error and testing error gradually decreased. This implies that the 3 types of plants tended to be linearly separated. Notice that Q1 as a limit case of Q2 (with C set to 1e5), the model performed better than all models in Q2, so we could claim that a strict linear model is the best for linear kernel case.

In Q3, we adopted multiple kernels that are not linear, and set the slackness parameter C to 1 in the whole sub-question. We first adopted a 2^{nd}-order polynomial kernel, the training and testing error are both 0.033, we then adopted a 3^{rd}-order polynomial, the training error is 0.025 and testing error is 0.000, which is quite satisfactory. Comparing the two, we claim that a 3^{rd}-order polynomial would be a good choice, even better than the strictly linear one provided in Q1.

We then adopted a Radial Basis Function kernel with sigma = 1, i.e. the parameter gamma = 0.5 in SVC. Both the training and testing error are 0.033. Lastly, we adopted a sigmoid kernel with the same gamma, both the training and testing error are 0.667. Comparing the two, we found that RBF is better than sigmoid.