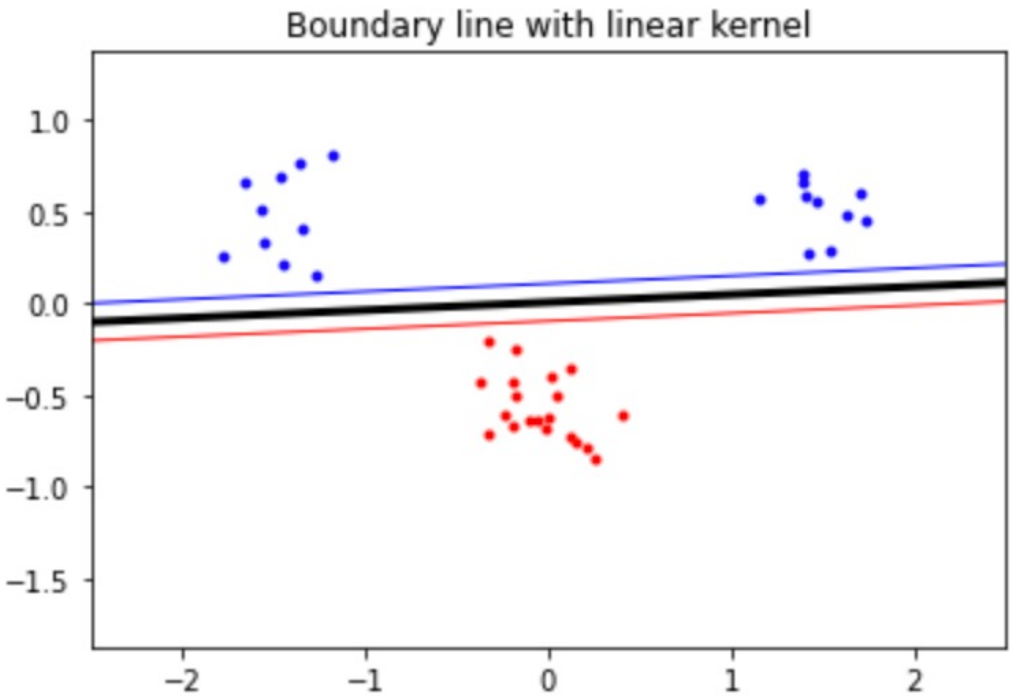
DD2421 Lab2 Support Vector Machines

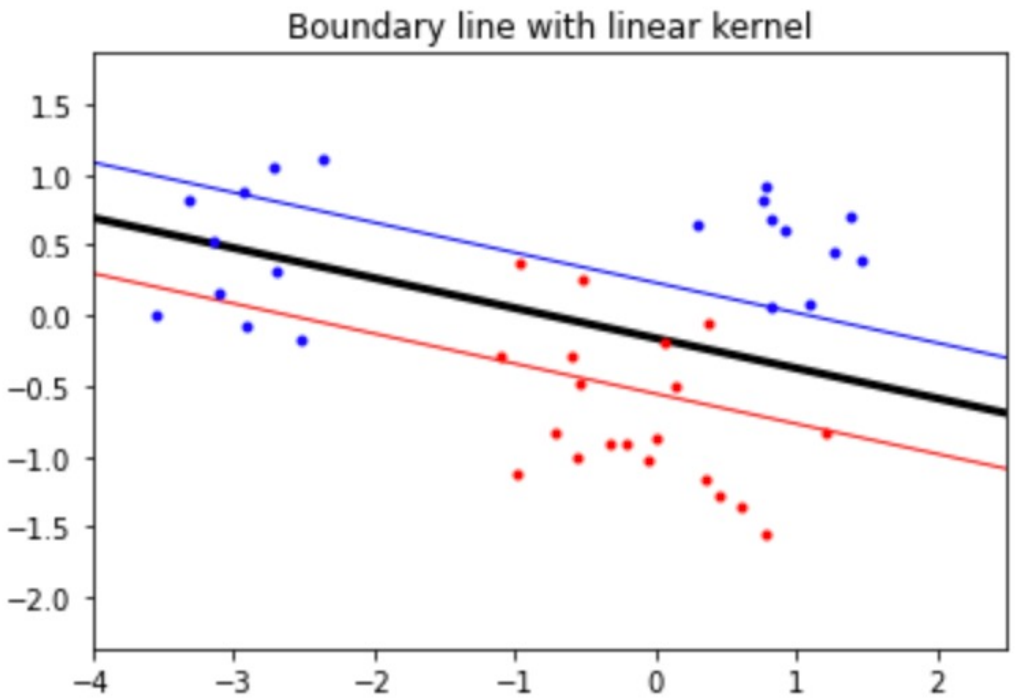
Lab Report

Yuqi Zheng Jingxuan Mao

1. Change the size and position of the data set, classifying with linear kernel and unlimited C



Center of dataset: Class A: [1.5, 0.5], [-1.5, 0.5]
Class B: [0, -0.5]
Variance of dataset: A=B=0.2
Result: Success



Center of dataset: Class A: [1, 0.5], [-3, 0.5]
Class B: [0, -0.5]
Variance of dataset: A=0.4, B=0.6
Result: Failed

2. Implement the Polynomial kernel with different parameter p

$$\mathcal{K}(\vec{x}, \vec{y}) = (\vec{x}^T \cdot \vec{y} + 1)^p$$

Center of dataset:

Result: All Success

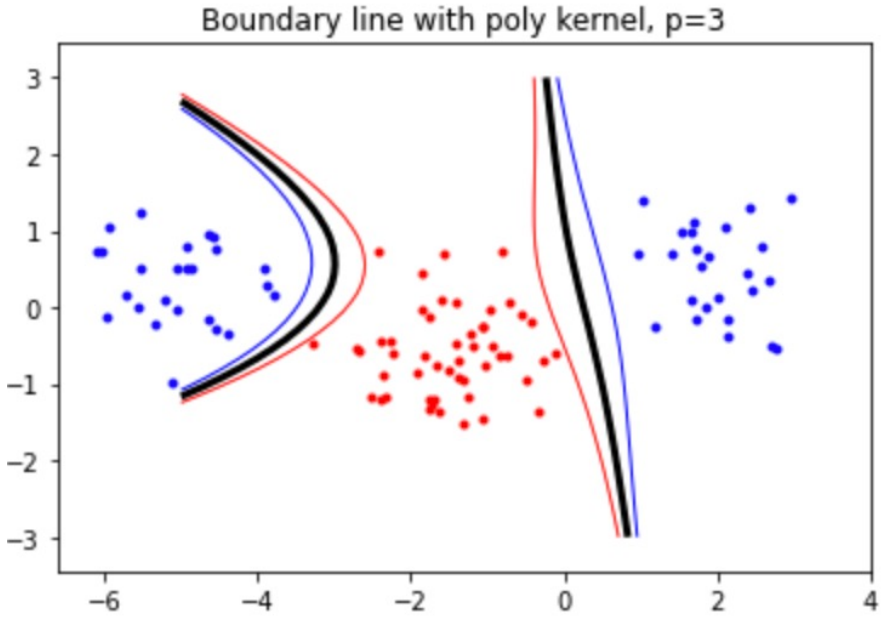
Class A: [2, 0.5], [-5, 0.5]

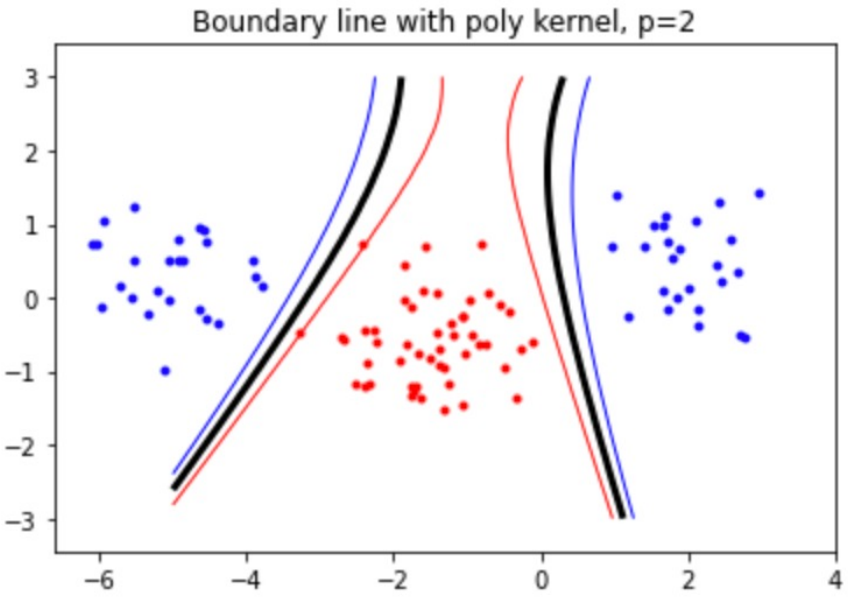
Class B: [-1.5, -0.5]

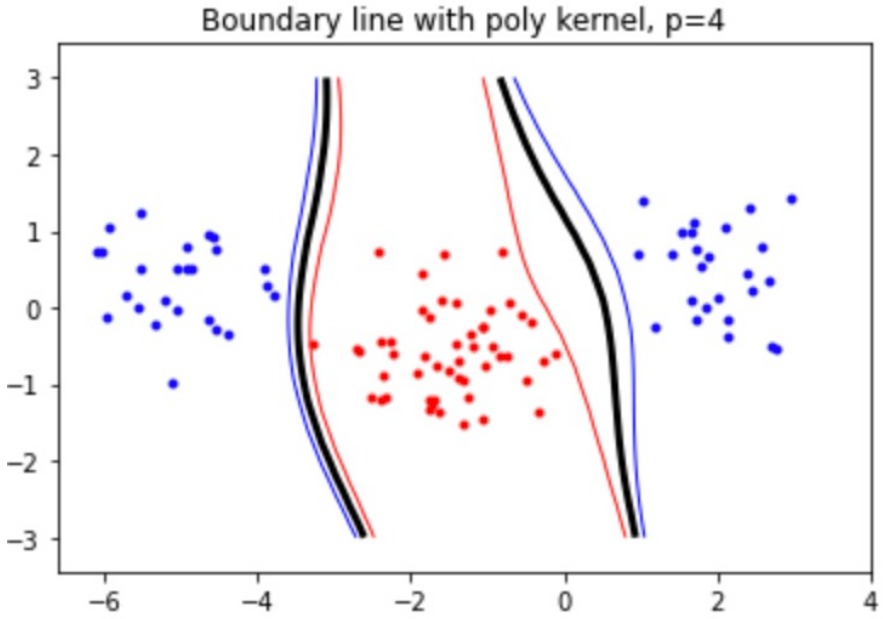
C is unlimited

Variance of dataset: A = B = 0.6

Parameter: p = 2, 3, 4







3. Implement the Radial Basis Function kernel with different parameter sigma

$$\mathcal{K}(\vec{x},\vec{y}) = e^{-\frac{||\vec{x}-\vec{y}||^2}{2\sigma^2}}$$

Center of dataset:

Result: All Success

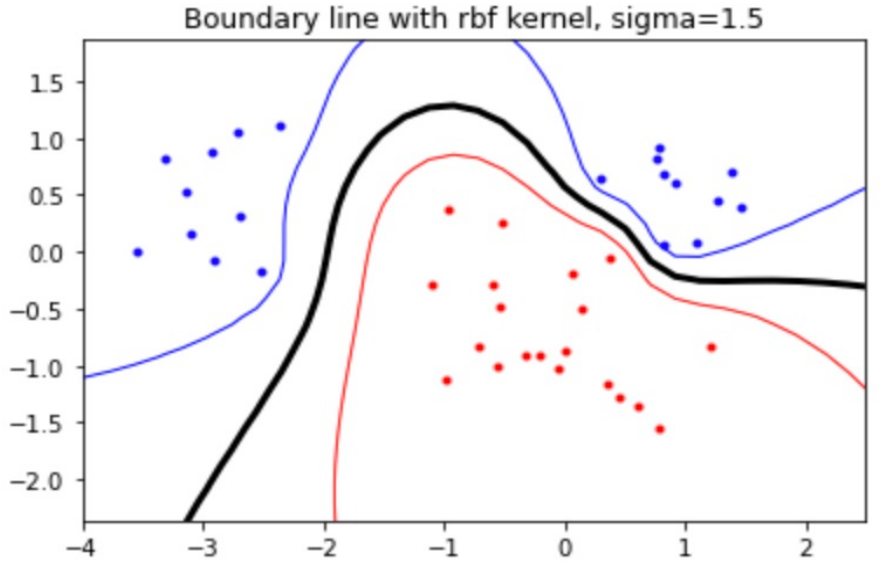
Class A: [1, 0.5], [-3, 0.5]

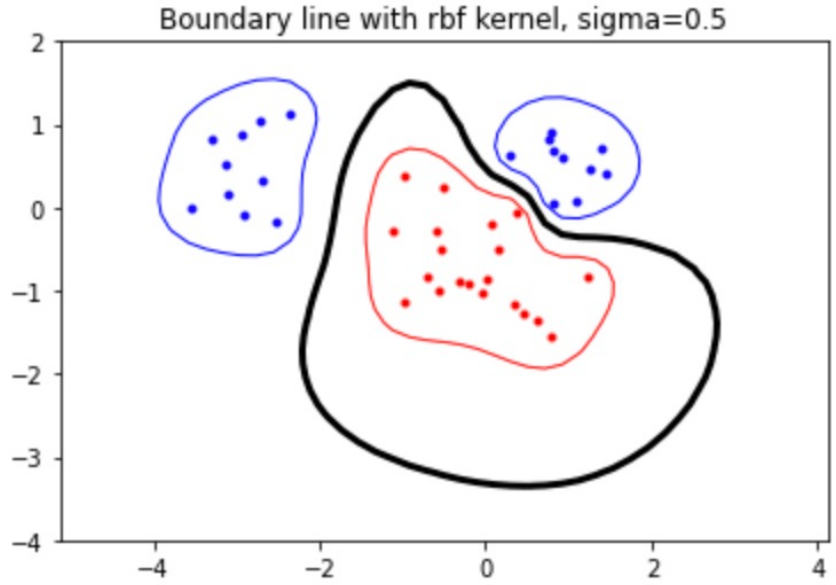
Class B: [0, -0.5]

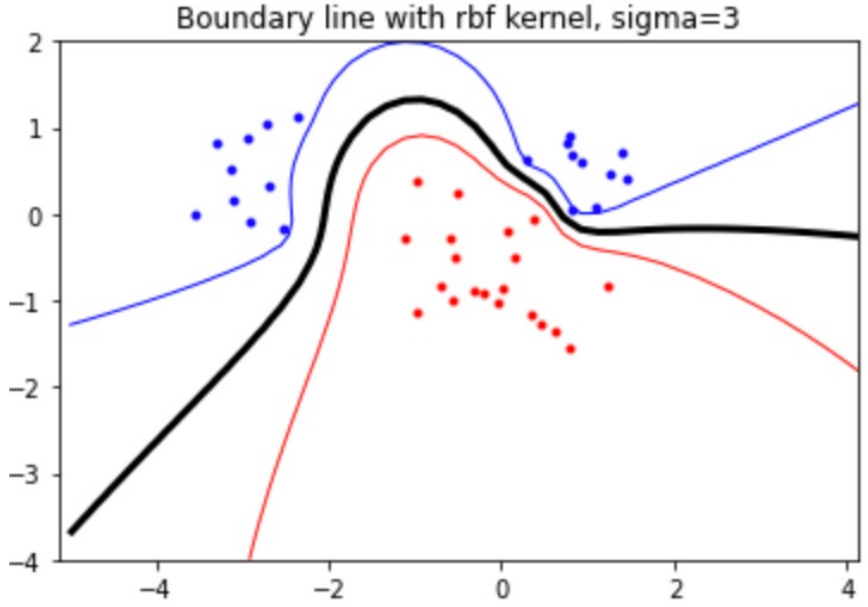
C is unlimited

Variance of dataset: A = 0.4, B = 0.6

Parameter: sigma = 0.5, 1,5, 2







Explanation about the result of different non-linear kernels and parameters:

For harder dataset, data points from different class are mixed together in the linear plane, so the liner kernel had a bad performance. Therefore, we tried non-lineal kernels, projecting the points to hyperplane and try to find possible boundary line within.

For Polynomial kernel, low p value means smoother boundary and higher bias and lower variance.
While higher p value may cause overfitting with lower bias and higher variance.

For RBF kernel, low sigma value can cause the problem of overfitting, since the boundary highly depends on dataset itself with lower bias and higher variance. While high sigma value will lead to smoother boundaries, which shows higher bias and lower variance.

4. Implement the Linear kernel with different C value

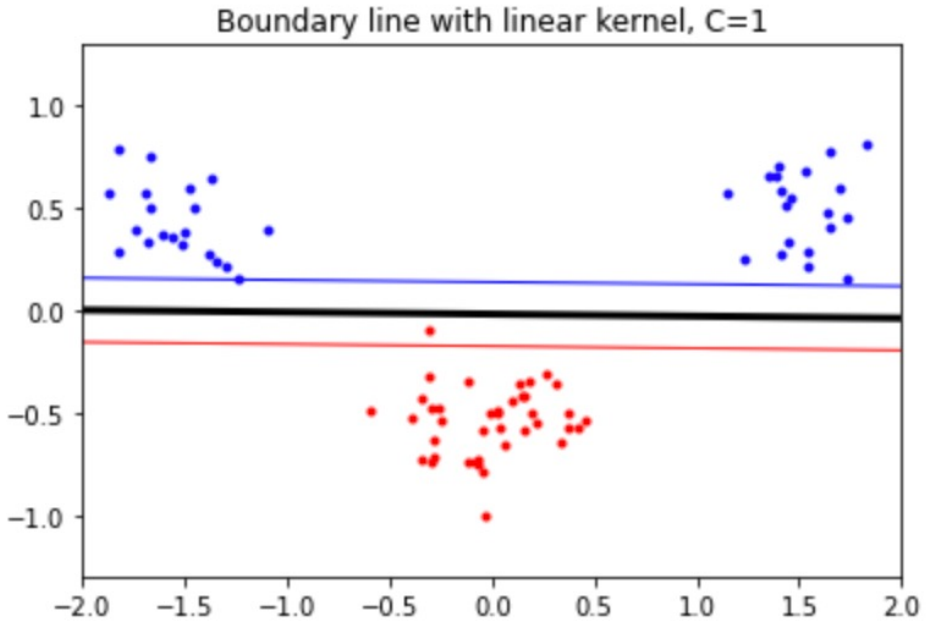
Center of dataset:

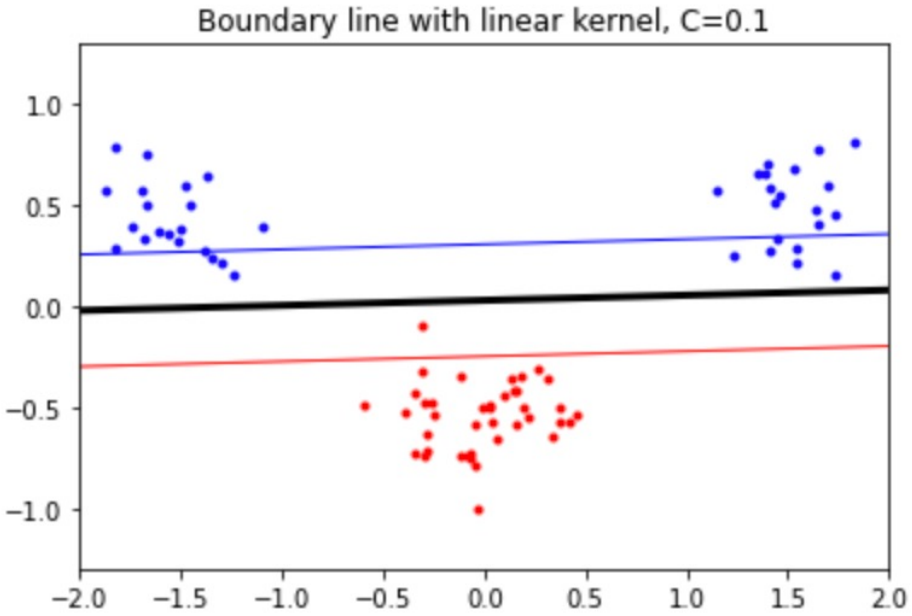
Class A: [1.5, 0.5], [-1.5, 0.5]

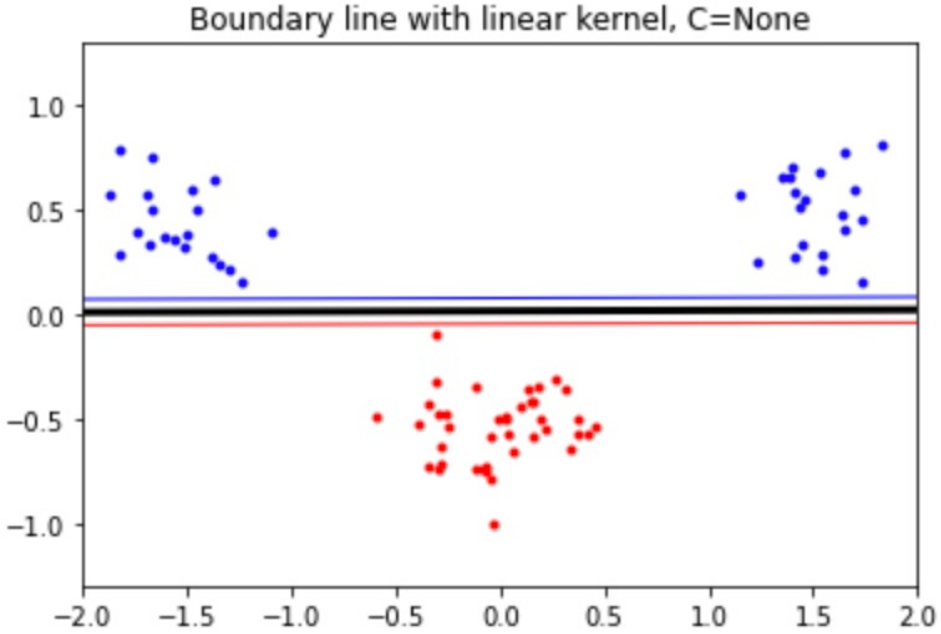
Class B: [0, -0.5]

Variance of dataset: A = B = 0.2

Parameter: C = 0.1, 1, None







Explanation about the result of different C values:

When stack variables are added, the subjective of the algorithm will be to minimize the formula below:

$$\min_{\vec{w},b,\vec{\xi}} ||\vec{w}|| + C\sum_{i} \xi_i$$

When C value is small, larger slack variables are tolerable, since the influence of each slack variable is constrained by the small C value, thus leads to wider margin. However, when C value is large, slack variables will be forced to be small during the optimization process, which means a stricter classification and narrow margin.

Explanation about slack v.s. kernel:

Firstly, we should select approximate kernel according to the characteristics of the dataset. For more complicated dataset where data points of different classes are not linear-separable, we should utilize a more complex kernel to project the data to hyperplane. Then, we could adjust the value of C. For noisy dataset, more slack which means more mistakes allowed will be helpful to improve the classification performance.