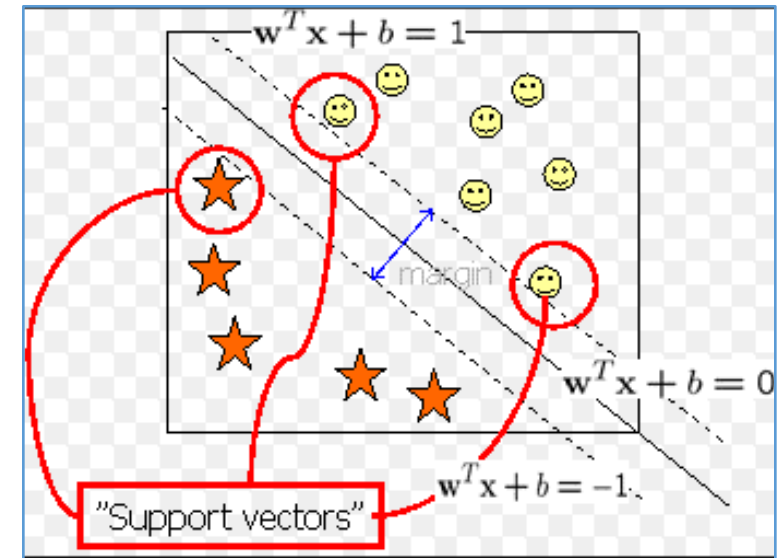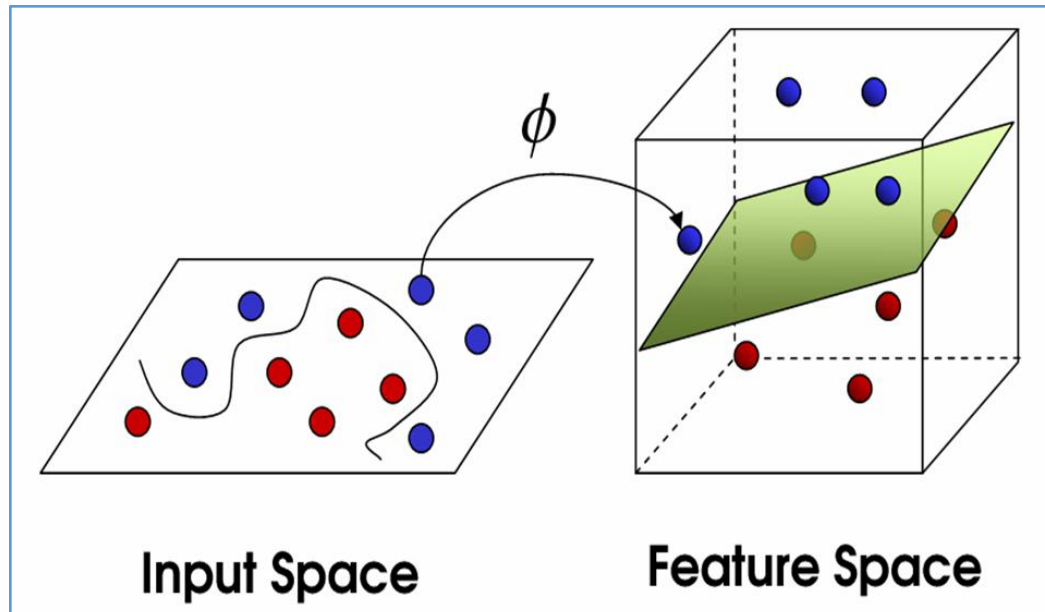# Lab 2: Support Vector Machines

Tianxiao Zhao
Feiyang Liu

Feb 23, 2017

Support Vector Machines

- Goal: Find the best hyperplane which has the largest separation margin
- Problem: Sometimes NOT linearly separable
- Solution: Construct a hyperplane in a higher dimensional space

- Lead to new problems: Huge computational load
- Solution: Define kernel function as dot products in the original space
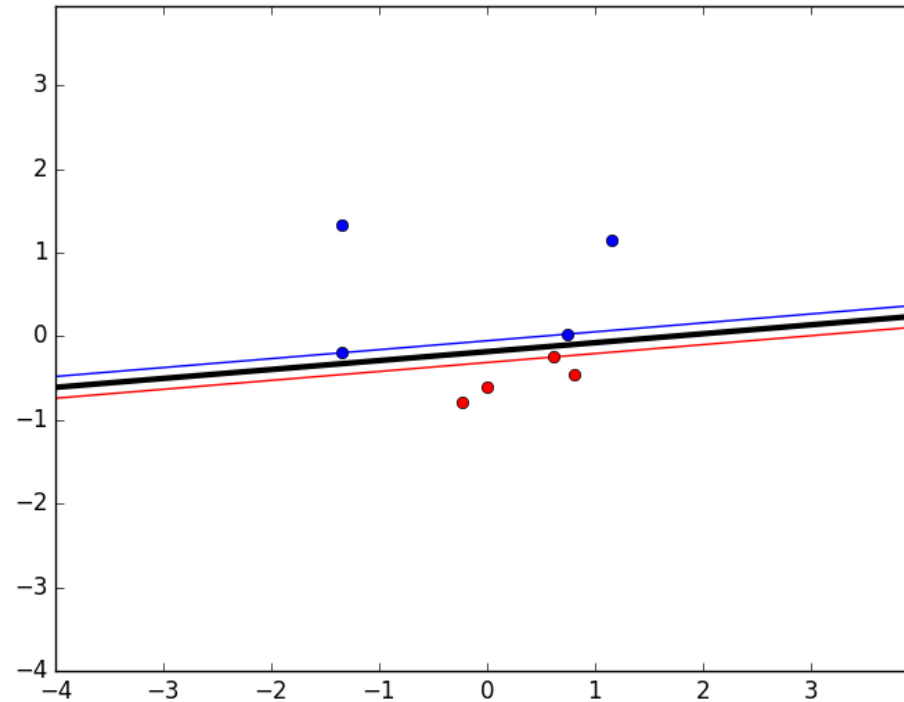
Soft Margin - Adding Slack

- **Pros**
  - Classifier more robust: anti-noise
  - Lower dimensionality
  - Smaller variance

- **Cons**
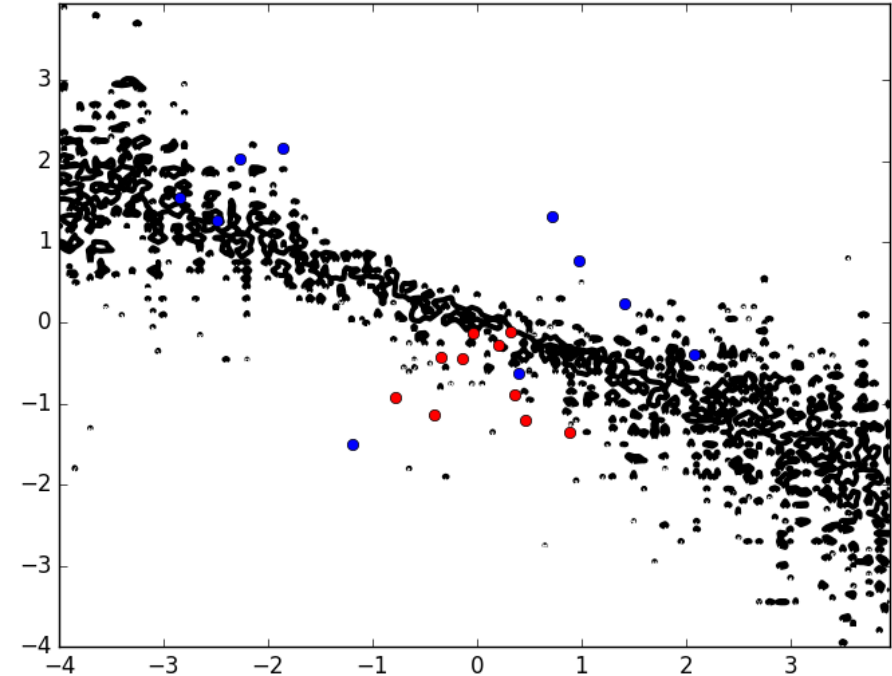 - Results in classification errors
  - Bigger bias
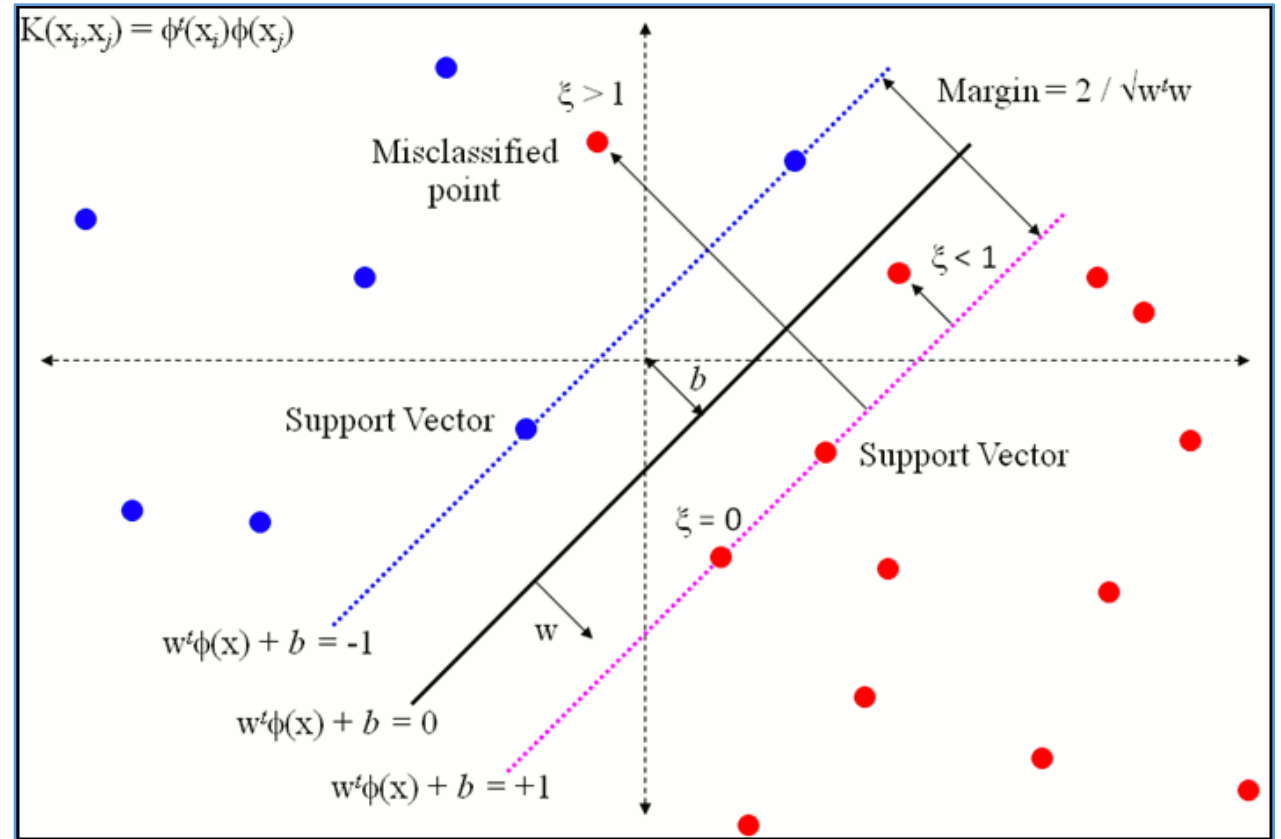
Kernel Functions

- Linear

- Polynomial

- Radial Basis Function

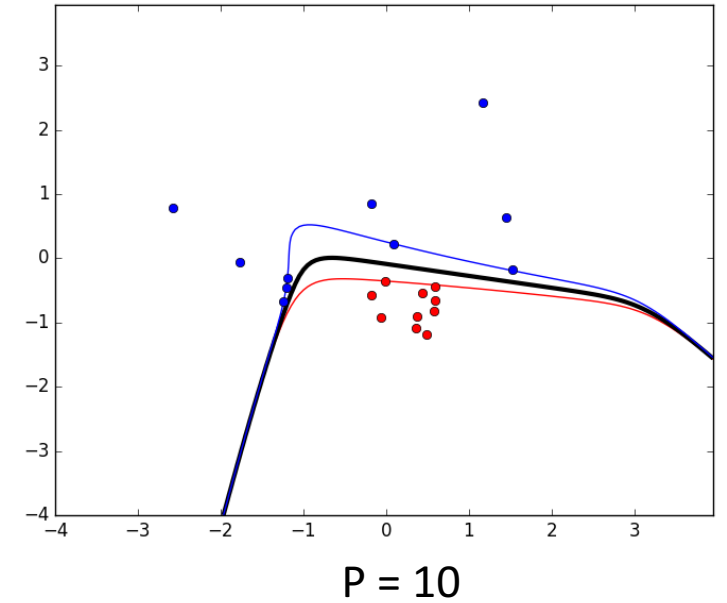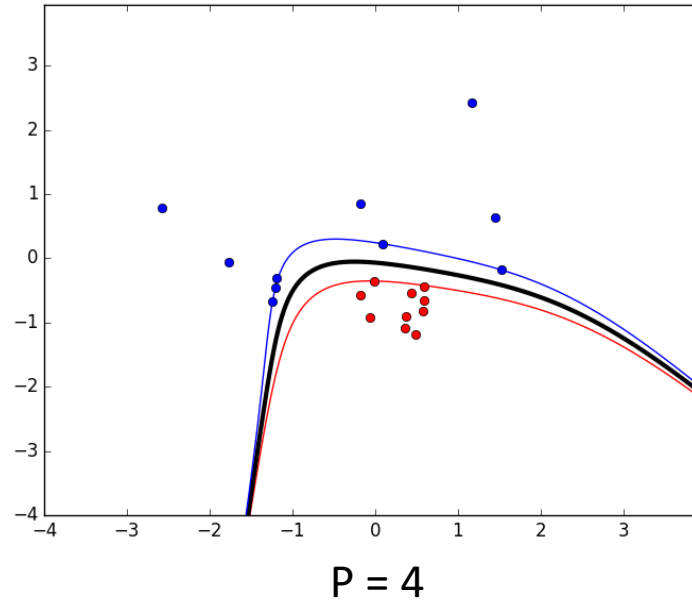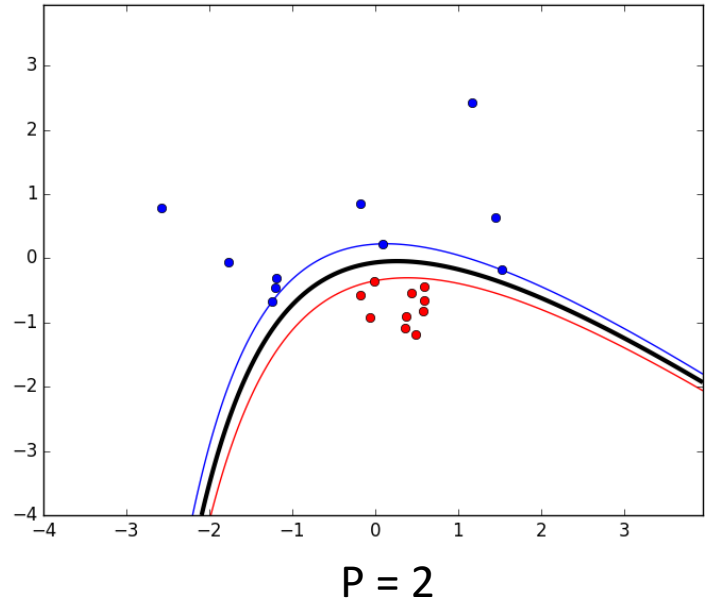# Different Data Points using linear kernel

Units = 8

Units = 20

- Support vectors moves ---> Boundary moves

- Non-linear-separability points ---> No solution for linear kernel

- Linear-separability points ---> might have a solution using this algorithm
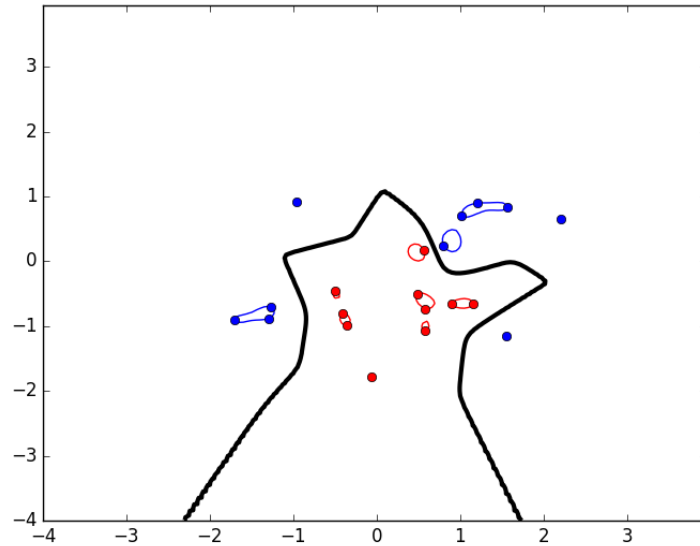
Polynomial Kernel using different p's
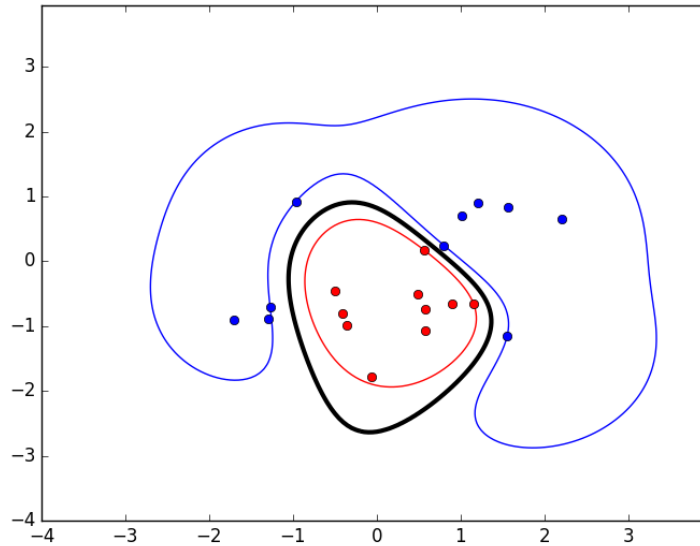
Units = 20

P = 2   P = 4   P = 10

- P increase ---> Higher dimensionalities ---> Smaller bias & Bigger variance
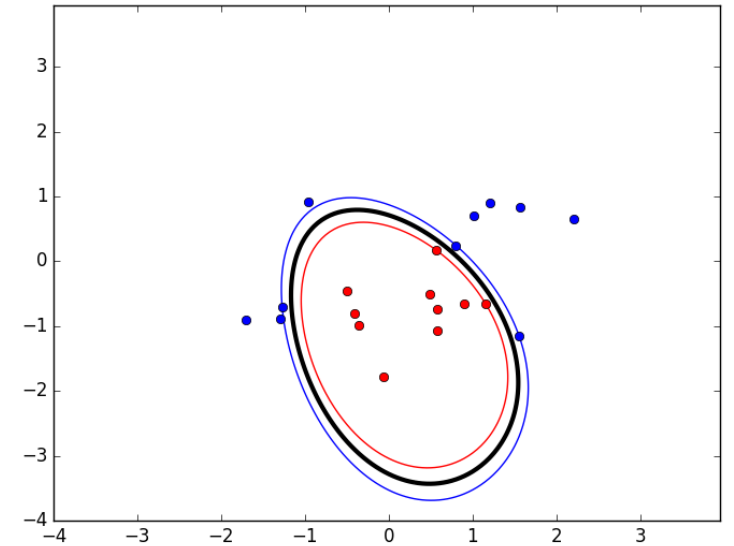
RBF Kernel using different theta
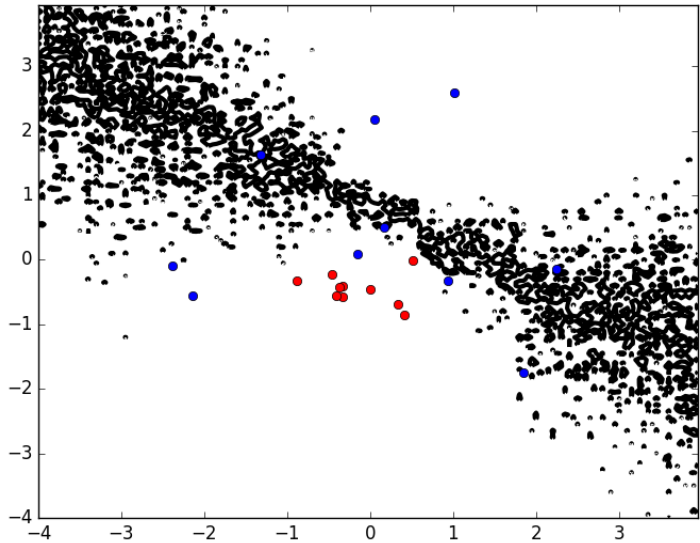

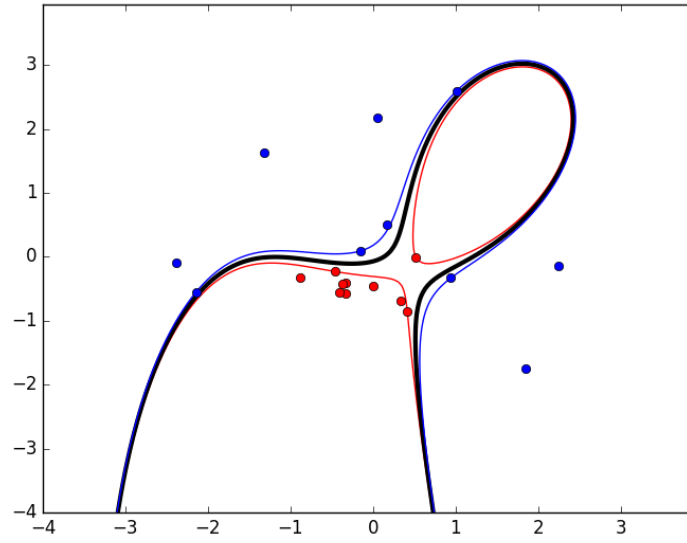theta = 0.2   theta = 1   theta = 3

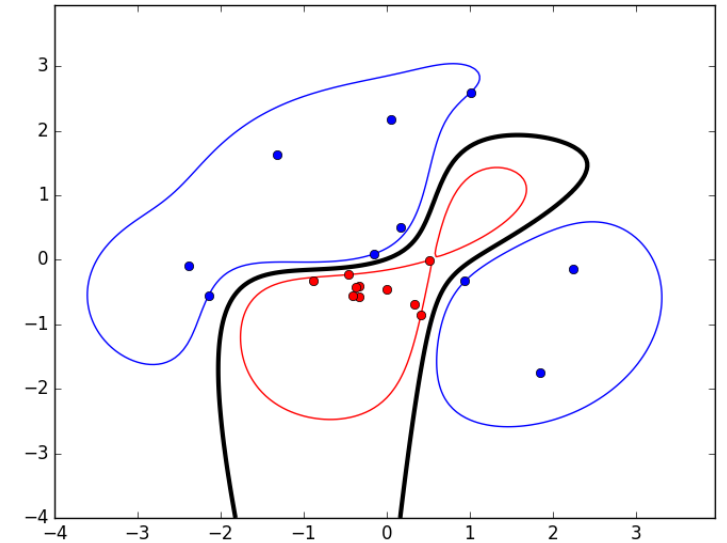- theta increase ---> Boundary smoother ---> Bigger bias & Smaller variance

Performance compare between 3 different Kernels
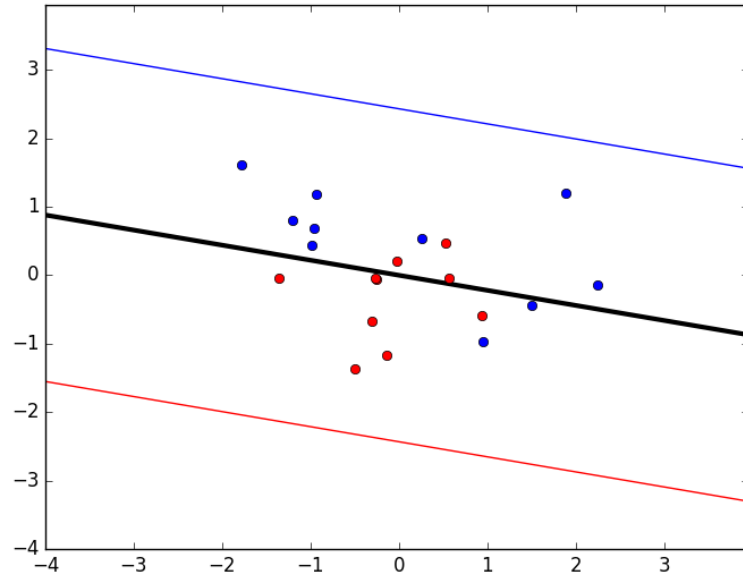

Linear kernel

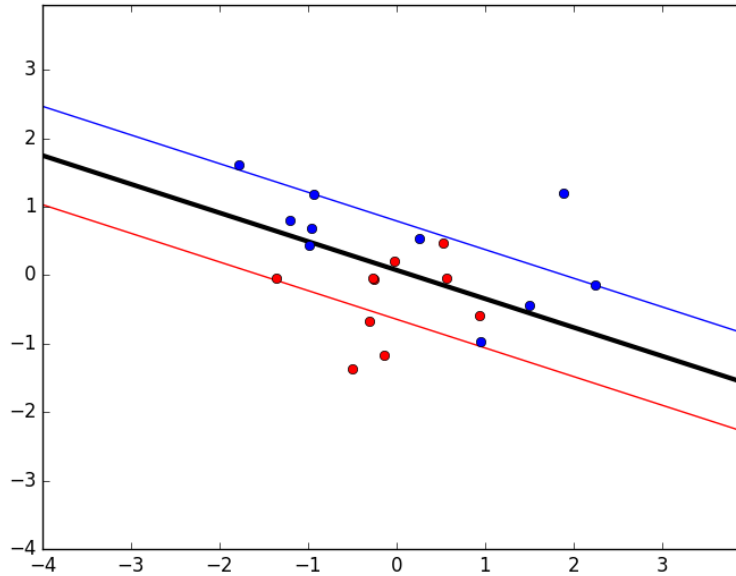Polynomial kernel with p=4

Radial kernel with theta=1

- Linear kernel: No solution due to non-linear-separability
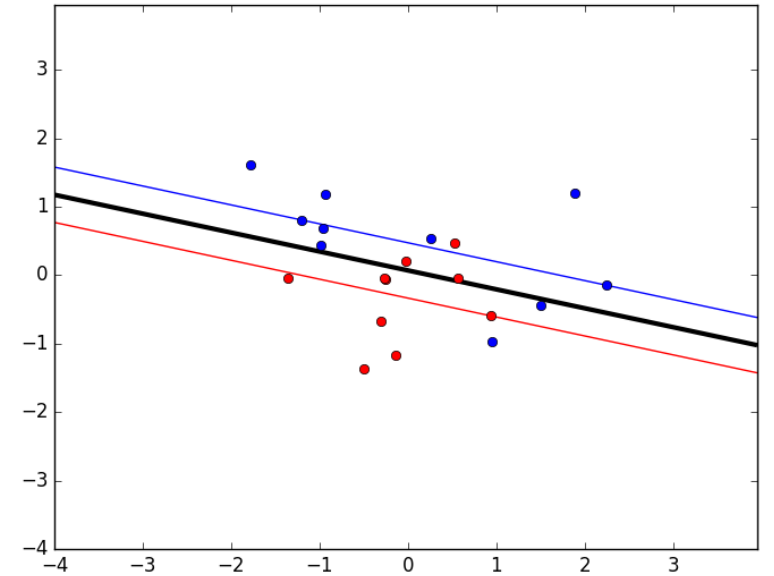
- But polynomial kernel & RBF kernel have solutions

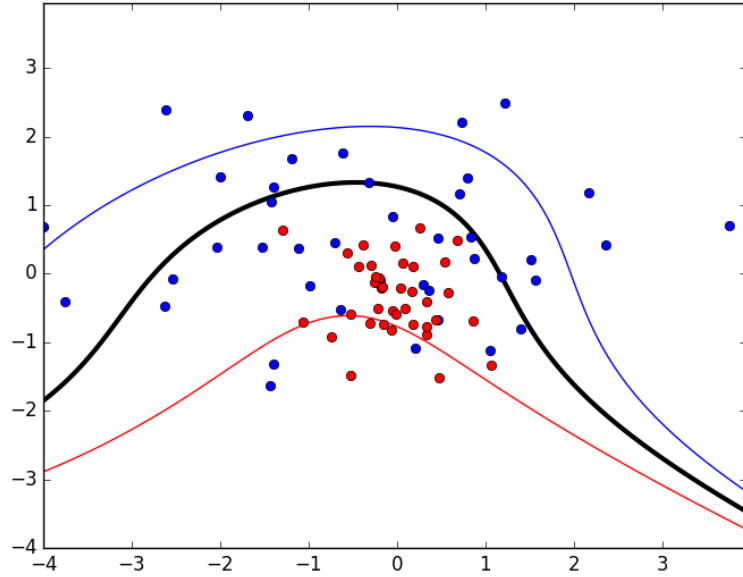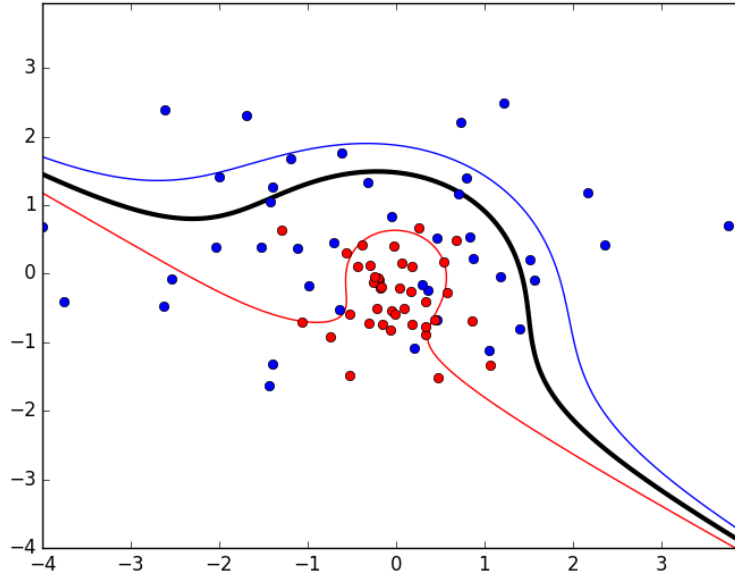# Slack Implementation using different C's (Linear)

C = 0.05   C = 1   C = 1000

- C increase ---> Slack decrease ---> Less margin ---> Less error but less robust

- Too much slack ---> unable to classify points (all data points are thought as outliers)
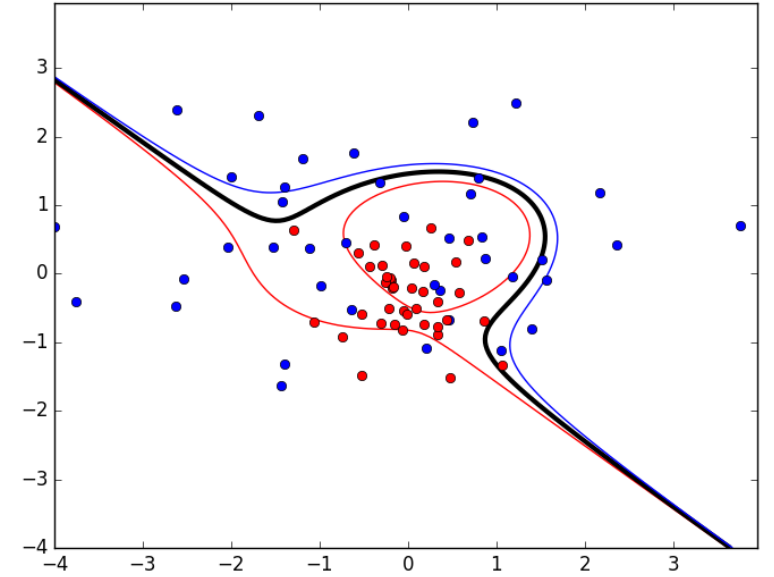
Slack Implementation using different C's (Poly)

C = 0.05   C = 1   C = 1000

- C increase ---> Slack decrease ---> Less margin ---> Less error but less robust

- Too much slack ---> unable to classify points (all data points are thought as outliers)

Slack Implementation using different C's (RBF)

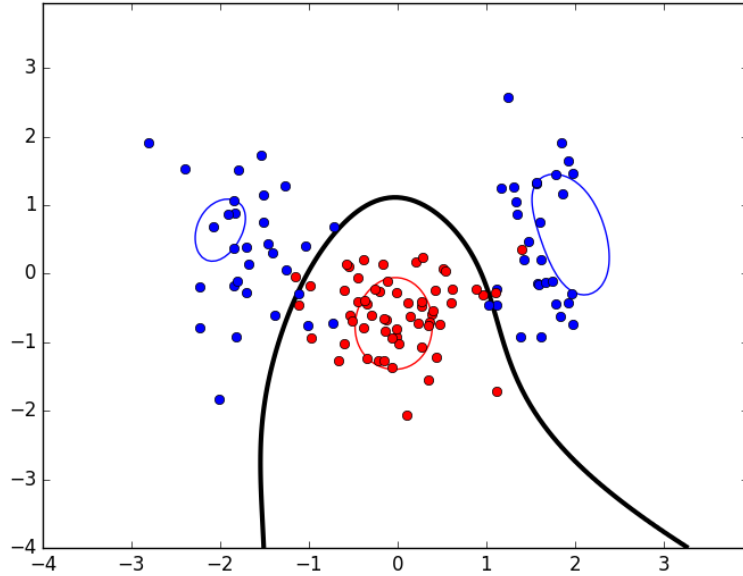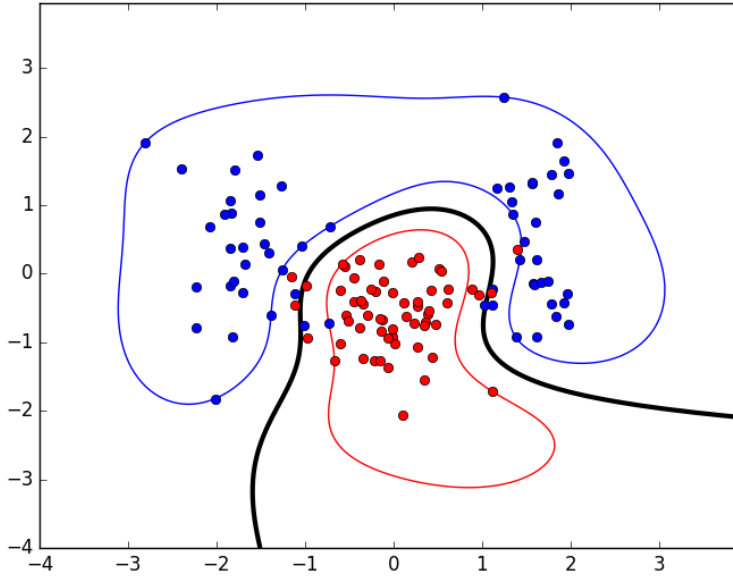C = 0.1   C = 10   C = 1000

- C increase ---> Slack decrease ---> Less margin ---> Less error but less robust

- Too much slack ---> unable to classify points (all data points are thought as outliers)
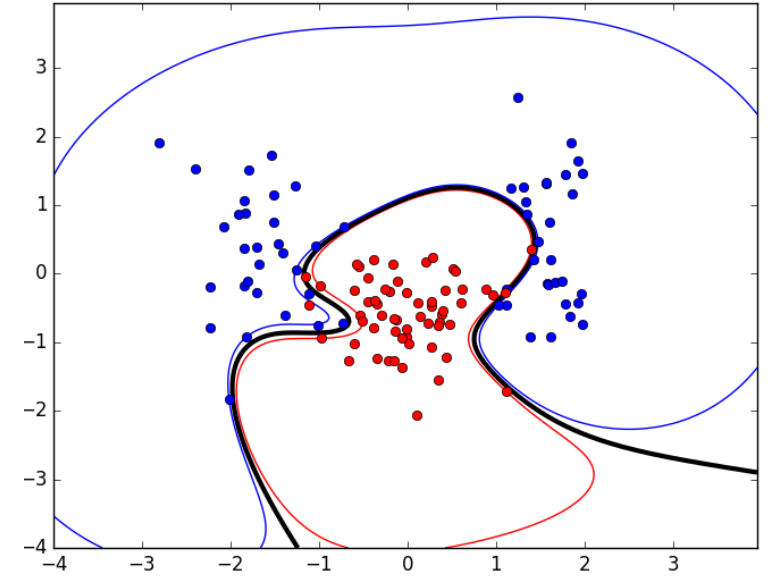
Conclusion about Slack

- More Slack ---> More Margin & More Robust ---> Bigger Bias But Smaller Variance

  ---> Avoid going for a more complex model

- Vice versa