Support Vector Machine

Il-Chul Moon
Dept. of Industrial and Systems Engineering
KAIST
icmoon@kaist.ac.kr

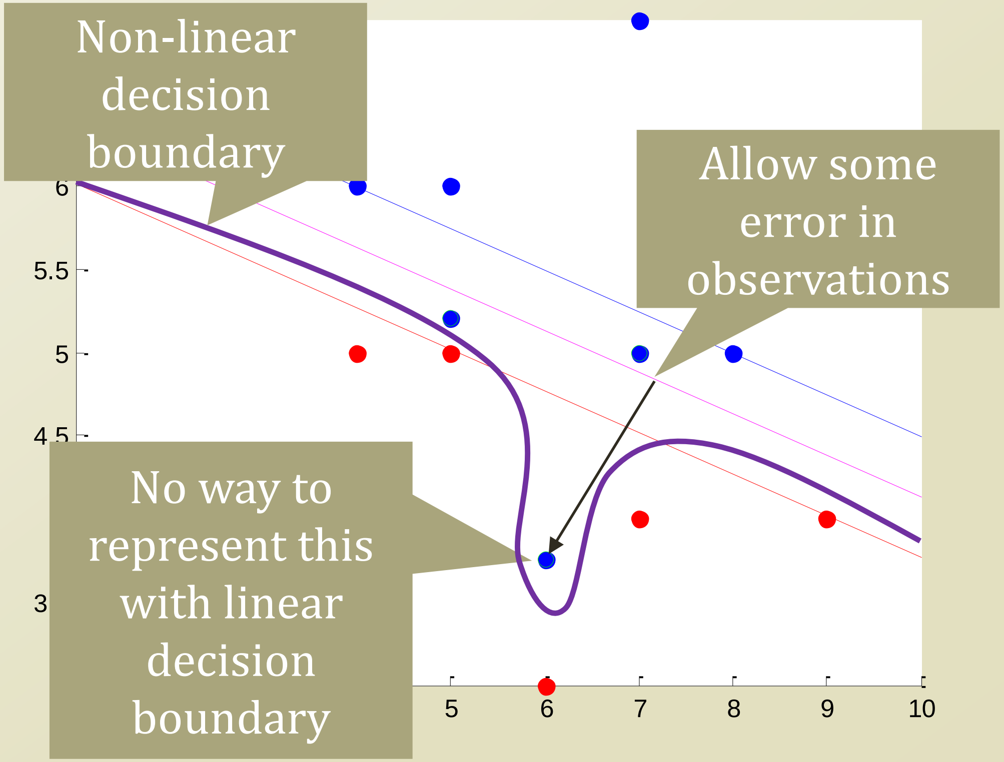
Weekly Objectives

- Learn the support vector machine classifier
  - Understand the maximum margin idea of the SVM
  - Understand the formulation of the optimization problem
- Learn the soft-margin and penalization
  - Know how to add the penalization term
  - Understand the difference between the log-loss and the hinge-loss
- Learn the kernel trick
  - Understand the primal problem and the dual problem of SVM
  - Know the types of kernels
  - Understand how to apply the kernel trick to SVM and logistic regression

SOFT MARGIN

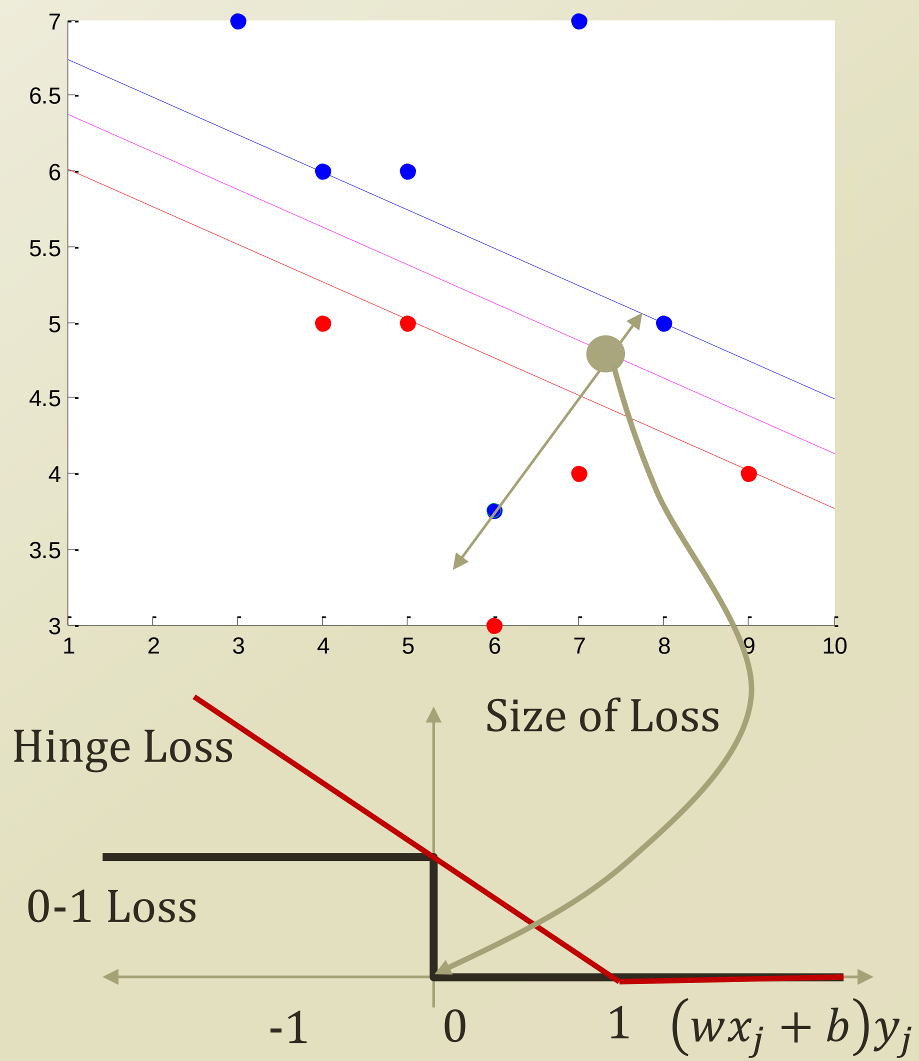
"Error" Cases in SVM

- Data points that are
  - Impossible to classify with a linear decision boundary
- So called, "error" cases…
- How to manage these?
  - Option 1
    - Make decision boundary more complex
    - Go to non-linear
    - Any problem?
  - Option 2
    - Admit there will be an "error"
    - Represent the error in our problem formulation.
    - Try to reduce the error as well.
    - Any problem?



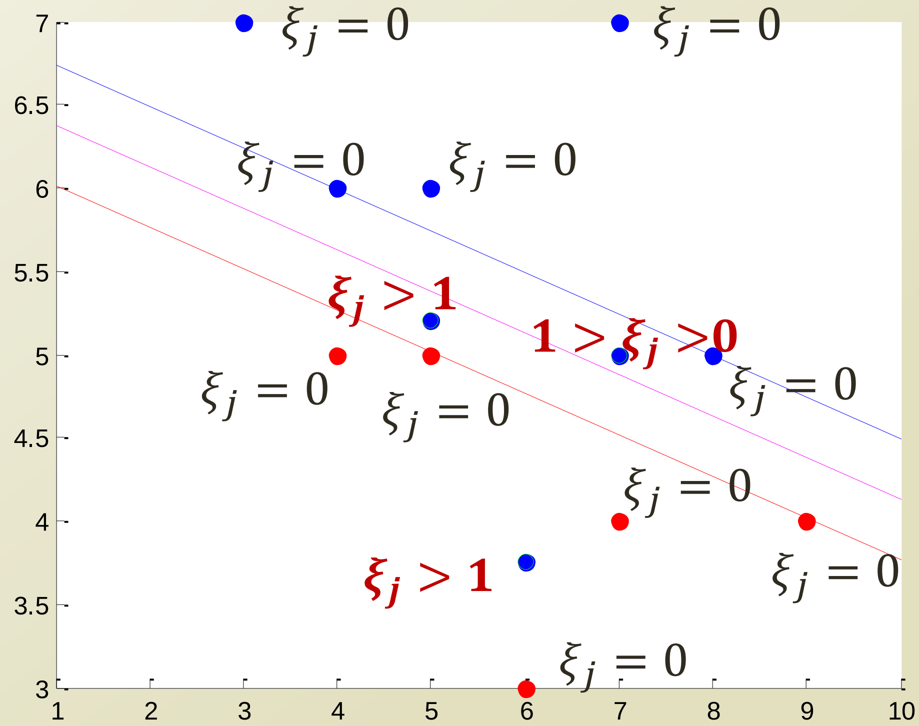
"Error" Handling in SVM

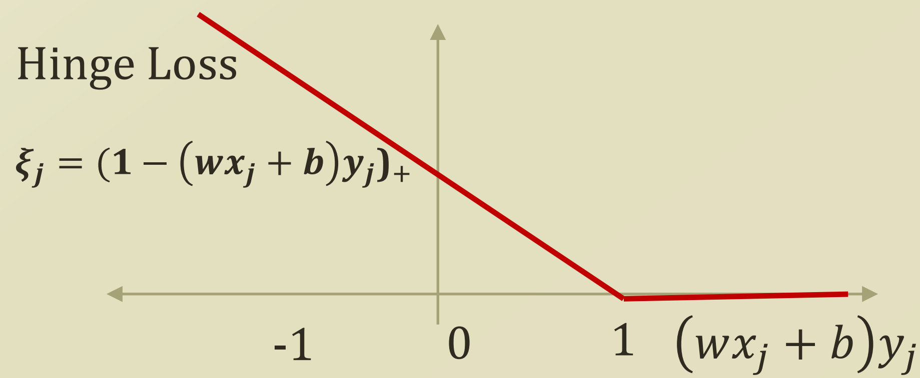
- How to handle
- Option 1)
  - Counting the error cases and reduce the counts
  - $min_{w,b}||w|| + C \times \#_{error}$
    $s.t.(wx_j + b)y_j \geq 1, \forall j$
  - Any problem?
- Option 2)
  - Introduce a slack variable
    - $\xi_j > 1$ when mis-classified
  - $min_{w,b}||w|| + C\sum_j \xi_j$
    $s.t.(wx_j + b)y_j \geq 1 - \xi_j, \forall j$
    $\xi_j \geq 0, \forall j$
  - Any problem?
- C = trade-off parameter



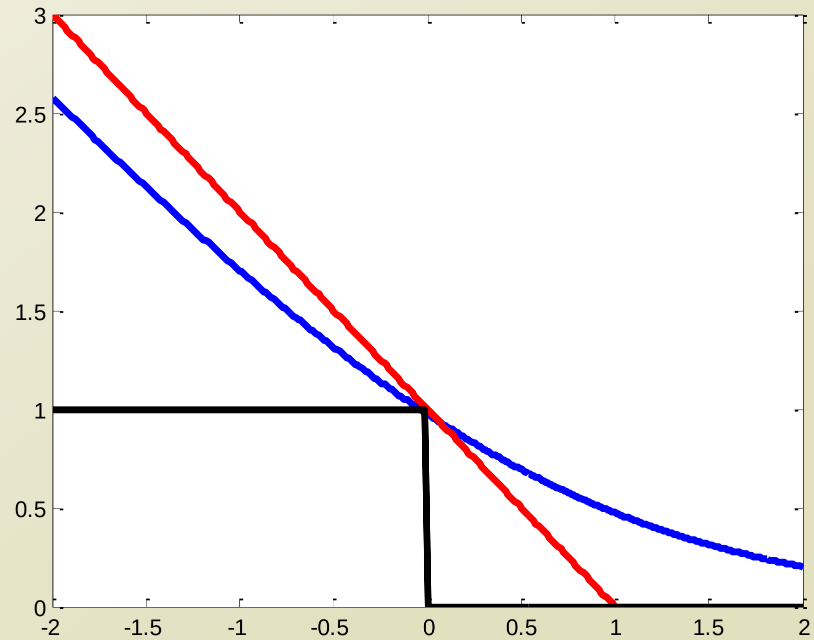
Soft-Margin SVM

- $min_{w,b}||w|| + C\sum_j \xi_j$
  $s.t.$
  $(wx_j + b)y_j \geq 1 - \xi_j, \forall j$
  $\xi_j \geq 0, \forall j$
- We soften the constraints
  - By adding a slack variable
- Instead, we penalize the mis-classification cases in the objective function
  - $C\sum_j \xi_j$
- How to recover the hard-margin SVM?

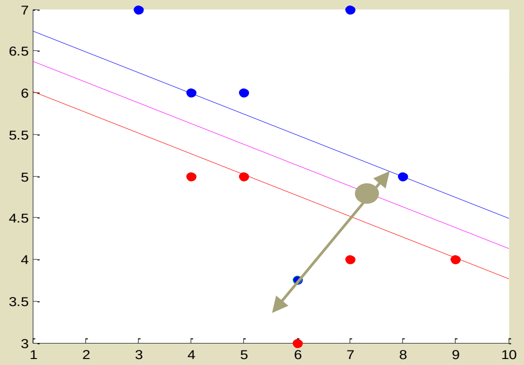




Comparison to Logistic Regression



- Loss function
  - $\xi_j = loss(f(x_j), y_j)$
- SVM loss function: Hinge Loss
  - $\xi_j = (1 - (wx_j + b)y_j)_+$
- Logistic Regression loss function: Log Loss
  - $\hat{\theta} = argmax_\theta \sum_{1 \le i \le N} log(P(Y_i|X_i;\theta))$
    $$= argmax_\theta \sum_{1 \le i \le N} \{Y_i X_i \theta - \log(1 + e^{X_i \theta})\}$$
  - $\xi_j = -\log\left(P\left(Y_j \middle| X_j, w, b\right)\right) = \log\left(1 + e^{(wx_j + b)y_j}\right)$
- Which loss function is preferable?
  - Around the decision boundary?
  - Overall place?



Strength of the Loss Function

- $min_{w,b,\xi_j} ||w|| + C \sum_j \xi_j$

  $s.t.$

  $(wx_j + b)y_j \geq 1 - \xi_j, \forall j$

  $\xi_j \geq 0, \forall j$

- Let's implement the model
- How does the decision boundary evolves over the variations of C?