Data Analysis 2021 Spring





**Lecture 12:**

**Support Vector Machines**

May 26 & May 31, 2021

**Taesoo Kwon**

[tskwon@seoultech.ac.kr](mailto:tskwon@seoultech.ac.kr)

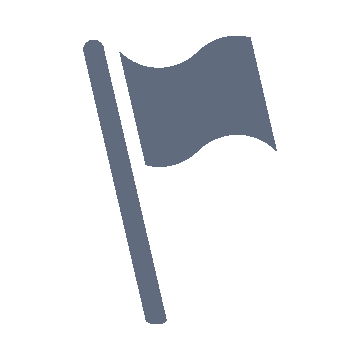
©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# Course Schedule (Tentative)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Topics** | **Note** | **Date (W)** | **Date (M)** |
| 1 | Orientation, Statistical Learning (Ch2) | Online | 03/03 | 03/08 |
| 2 | Statistical Learning (Ch2), Python Programming | Online | 03/10 | 03/15 |
| 3 | Probability & Statistics | Online | 03/17 | 03/22 |
| 4 | Probability & Statistics | Online | 03/24 | 03/29 |
| 5 | Linear Regression (Ch3) | Online | 03/31 | 04/05 |
| 6 | Linear Regression (Ch3) | Online | 04/07 | 04/12 |
| 7 | Classification (Ch4) | Online | 04/14 | 04/19 |
| 8 | **Midterm exam** | **Class hours (W1-W7)** | **04/21** | **04/26** |
| 9 | Resampling Methods (Ch5) | Online | 04/28 | 05/03 |
| 10 | Linear Model Selection and Regularization (Ch6) | Online | 05/05 | 05/10 |
| 11 | Moving Beyond Linearity (Ch7) | Online | 05/12 | 05/17 |
| 12 | Tree-Based Methods (Ch8) | Online | 05/19 | 05/24 |
| **13** | Support Vector Machines (Ch9) | Online | 05/26 | 05/31 |
| 14 | Unsupervised Learning (Ch10) | Online | 06/02 | 06/07 |
| 15 | **Final exam** | **6pm (W9-W14)** | **06/10Th** | **06/10Th** |

©copyright CIEL 2015 **2**





* Support vector machines

**OUTLINES**

* + Maximal margin classifier, support vector classifier
  + Support vector machines
* Python lab
* Summary & Next class

©copyright CIEL 2015



**Support Vector Machines**



**: Ch 9**

## Support vector machines

* Python lab
* Summary & Next class

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# Support Vector Machines

* Here we approach the two-class classification problem in a direct way:
  + We try and find a plane that separates the classes in feature space
* If we cannot, we get creative in two ways:
  + We soften what we mean by “separates", and
  + We enrich and enlarge the feature space so that separation is possible.

©copyright CIEL 2015 **5**

# What is a Hyperplane?

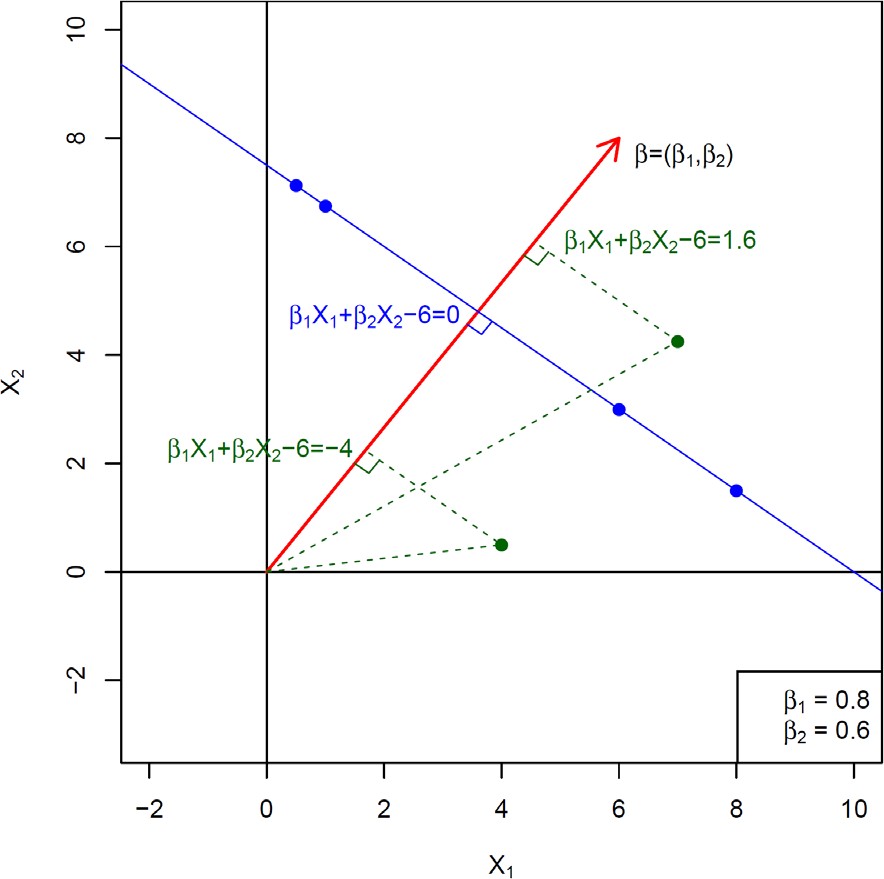
* A hyperplane in 𝑝𝑝 dimensions is a flat affine subspace of dimension 𝑝𝑝 − 1
* In general the equation for a hyperplane has the form



* In 𝑝𝑝 = 2 dimensions a hyperplane is a line
* If 𝑝𝑝 = 0, the hyperplane goes through the origin, otherwise not.
* The vector 𝛽𝛽 = (𝛽𝛽1, 𝛽𝛽2, ⋯ , 𝛽𝛽𝑝𝑝) is called the normal vector
  + It points in a direction orthogonal to the surface of a hyperplane

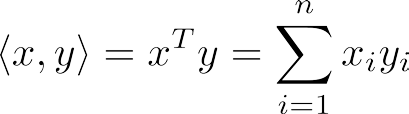
©copyright CIEL 2015 **6**

# Hyperplane in 2 Dimensions



©copyright CIEL 2015 **7**

# [FYI] Inner Product and Norm of Vectors

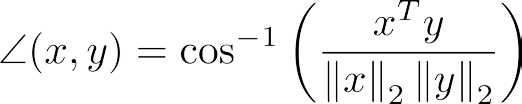
* Standard inner product on
* Euclidean norm, or *l*2-norm



* Cauchy-Schwartz inequality



* Angle between nonzero vectors 

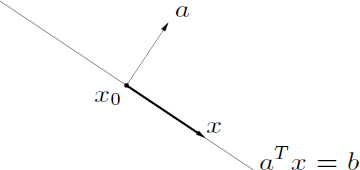


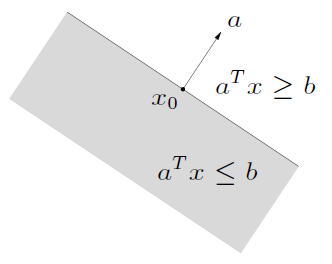
* + and are orthogonal if 

©copyright CIEL 2015 **8**

# [FYI] Hyperplanes and Halfspaces

* Hyperplane: set of the form 

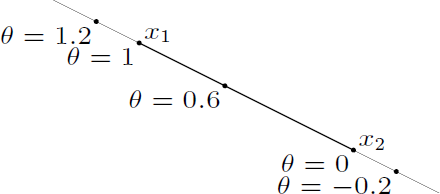


* Halfspace: set of the form 
  + is the normal vector 
  + Hyperplanes are affine and convex; halfspaces are convex

©copyright CIEL 2015 **9**

# [FYI] Affine Set

* Line through : all points



* Affine set: contains the line through any two distinct points in the set
  + As a subspace plus an offset 
* Example: solution set of linear equations

(Conversely, every affine set can be expressed as solution set of system of linear equations)

©copyright CIEL 2015 **10**

# [FYI] Convex Set

* Line segment between and : all points



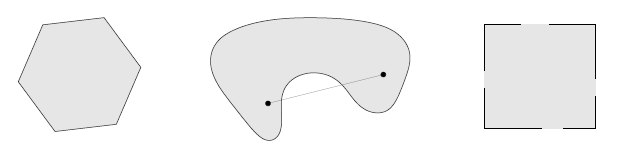
with

* Convex set: contains line segment between any two points in the set



* Examples (one convex, two nonconvex sets)

**11**



©copyright CIEL 2015

# [FYI] Convex Optimization Problem

* Standard form convex optimization problem





  are convex; equality constraints are affine

* + Problem is quasiconvex if  is quasiconvex (and  convex)
* Often written as





important property: feasible set of a convex optimization problem is convex

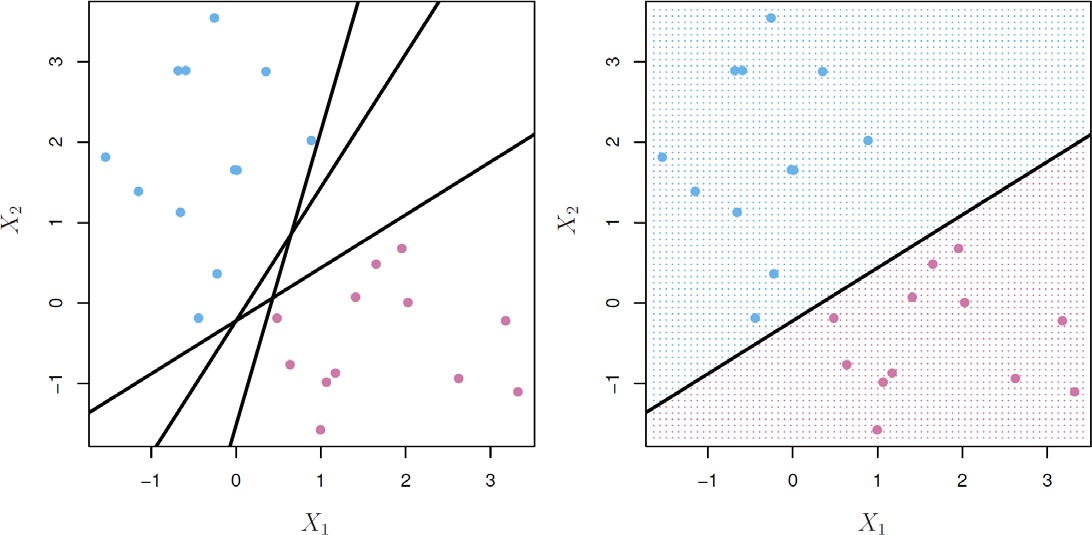
©copyright CIEL 2015 **12**

# Separating Hyperplanes

* If 𝑓𝑓

𝑓𝑓

𝑋𝑋

= 𝛽𝛽0 + 𝛽𝛽1𝑋𝑋1 + ⋯ + 𝛽𝛽𝑝𝑝𝑋𝑋𝑝𝑝, then 𝑓𝑓

< 0 for points on the other

𝑋𝑋

* 0 for points on one side of the hyperplane, and



* If we code the colored points as 𝑌𝑌𝑖𝑖 = +1 for blue, say, and 𝑌𝑌𝑖𝑖 = −1 for mauve, then 𝑌𝑌𝑖𝑖 � 𝑓𝑓 > 0

𝑋𝑋

𝑋𝑋𝑖𝑖

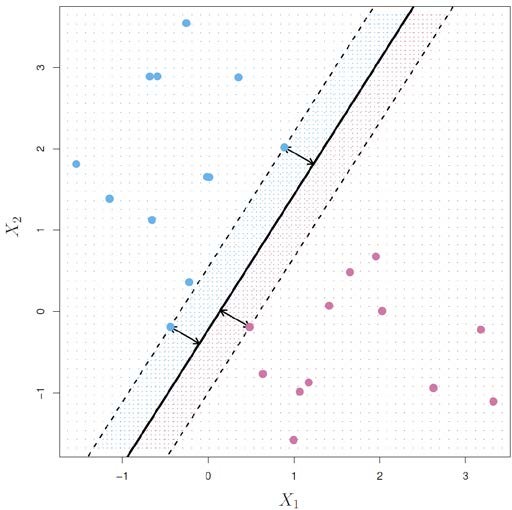
for all 𝑖𝑖, 𝑓𝑓 = 0 defines a separating hyperplane.

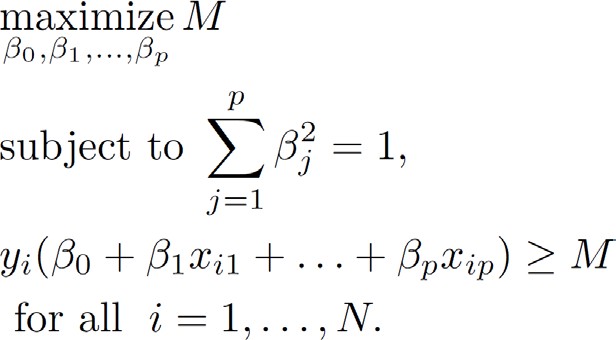
𝑋𝑋

©copyright CIEL 2015 **13**

# Maximal Margin Classifier

* Among all separating hyperplanes, find the one that makes the biggest gap or margin between the two classes

Constrained optimization problem

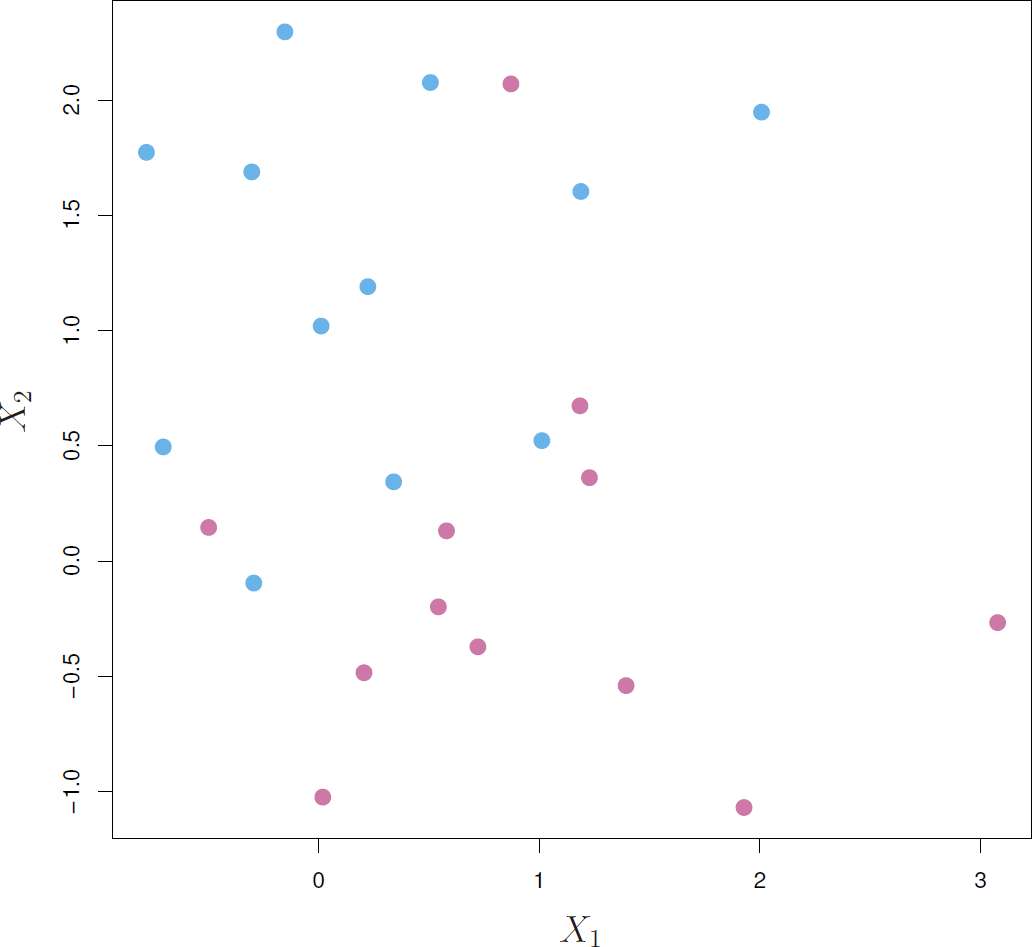


* + This can be rephrased as a convex quadratic program, and solved efficiently

©copyright CIEL 2015 **14**

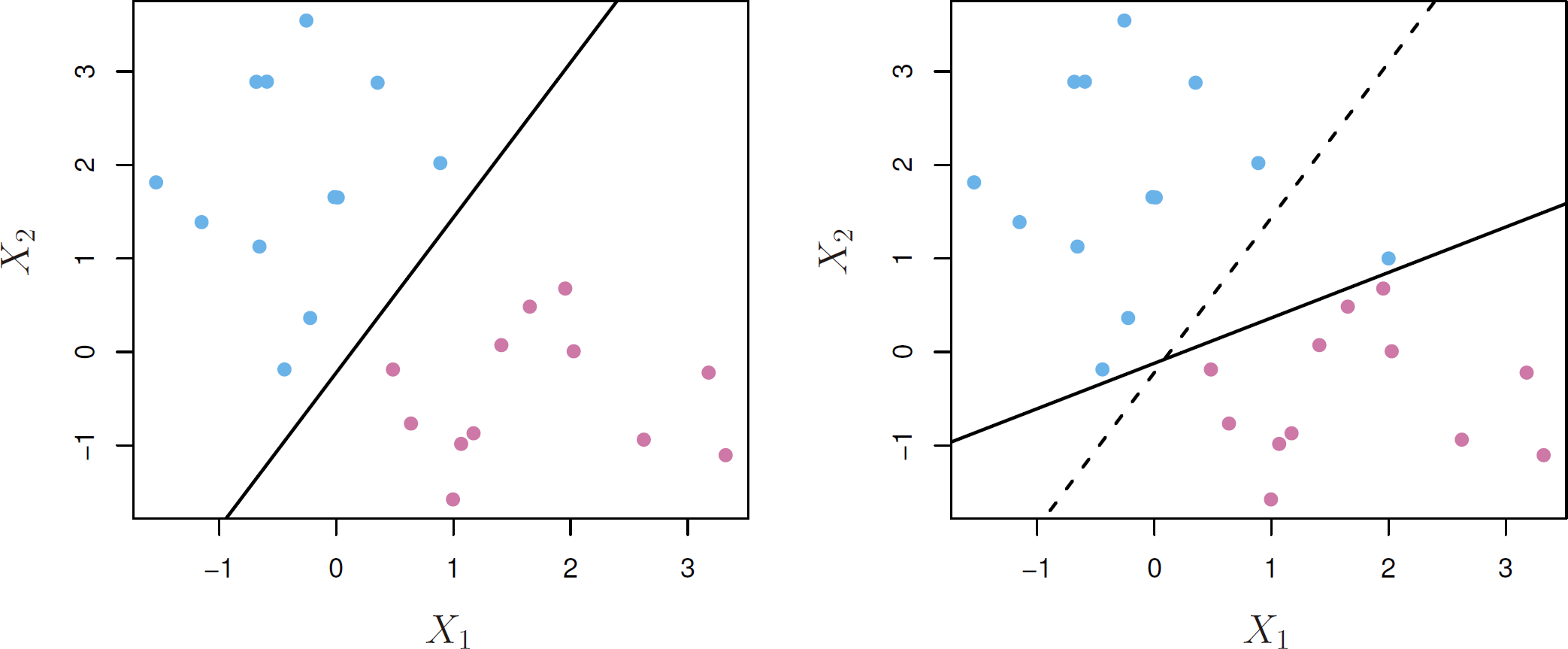
# Non-separable Data

* The data on the left are not separable by a linear boundary
  + This is often the case, unless 𝑁𝑁 < 𝑝𝑝



©copyright CIEL 2015 **15**

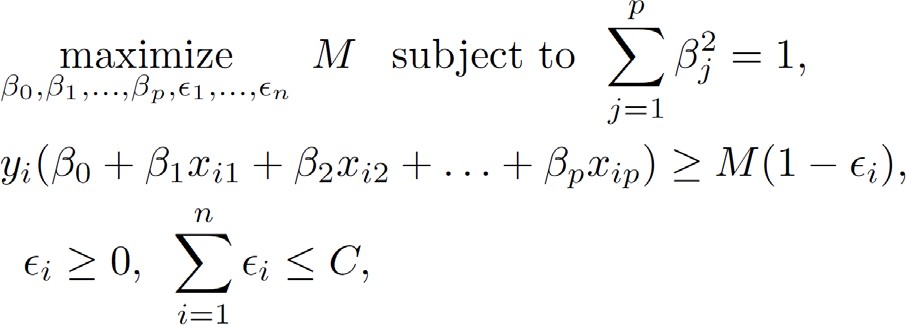
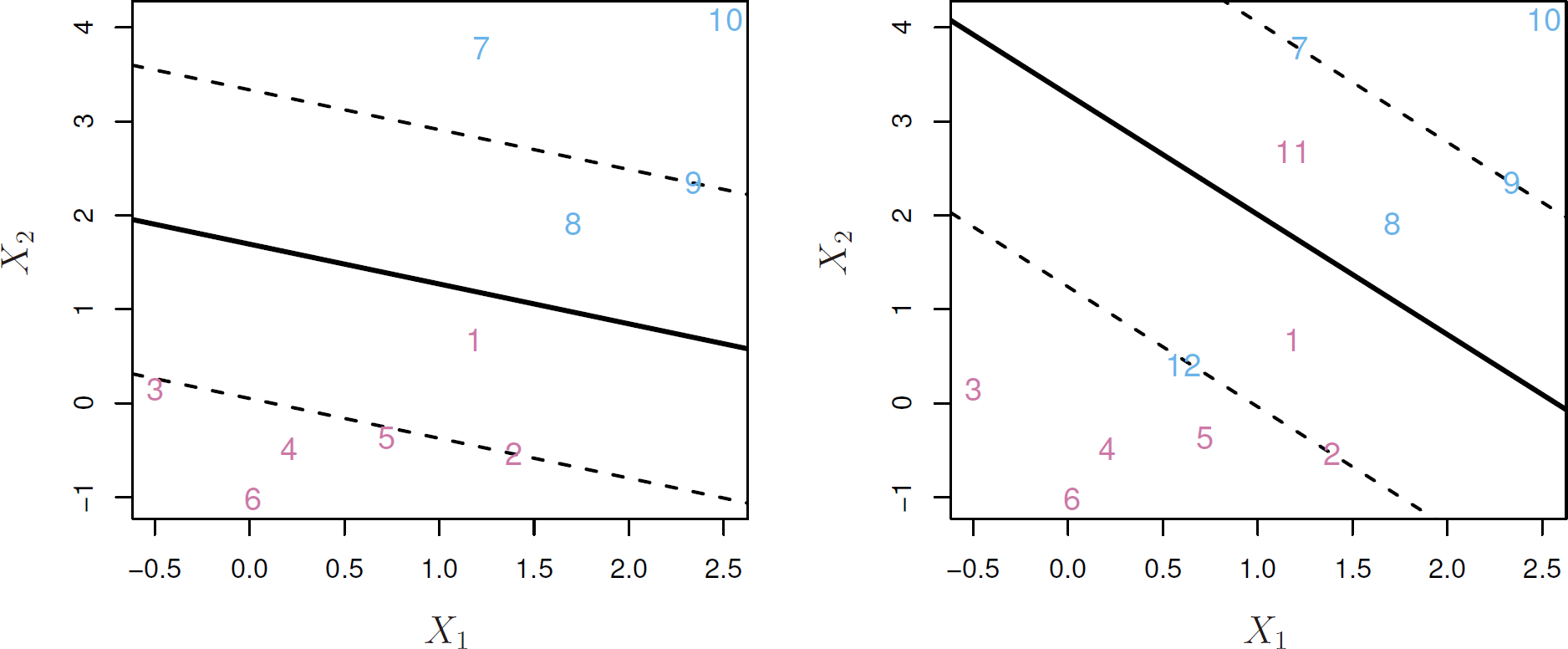
# Noisy Data



* Sometimes the data are separable, but noisy
  + This can lead to a poor solution for the maximal-margin classifier
* The support vector classifier maximizes a soft margin

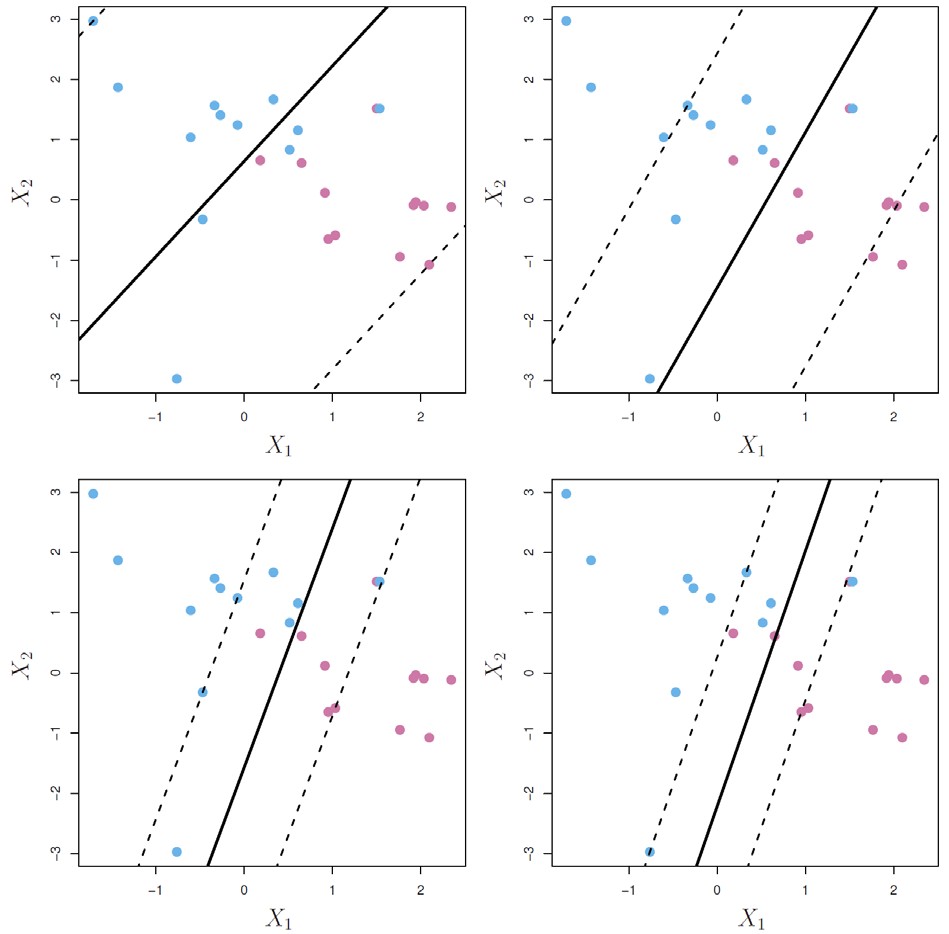
©copyright CIEL 2015 **16**

# Support Vector Classifier



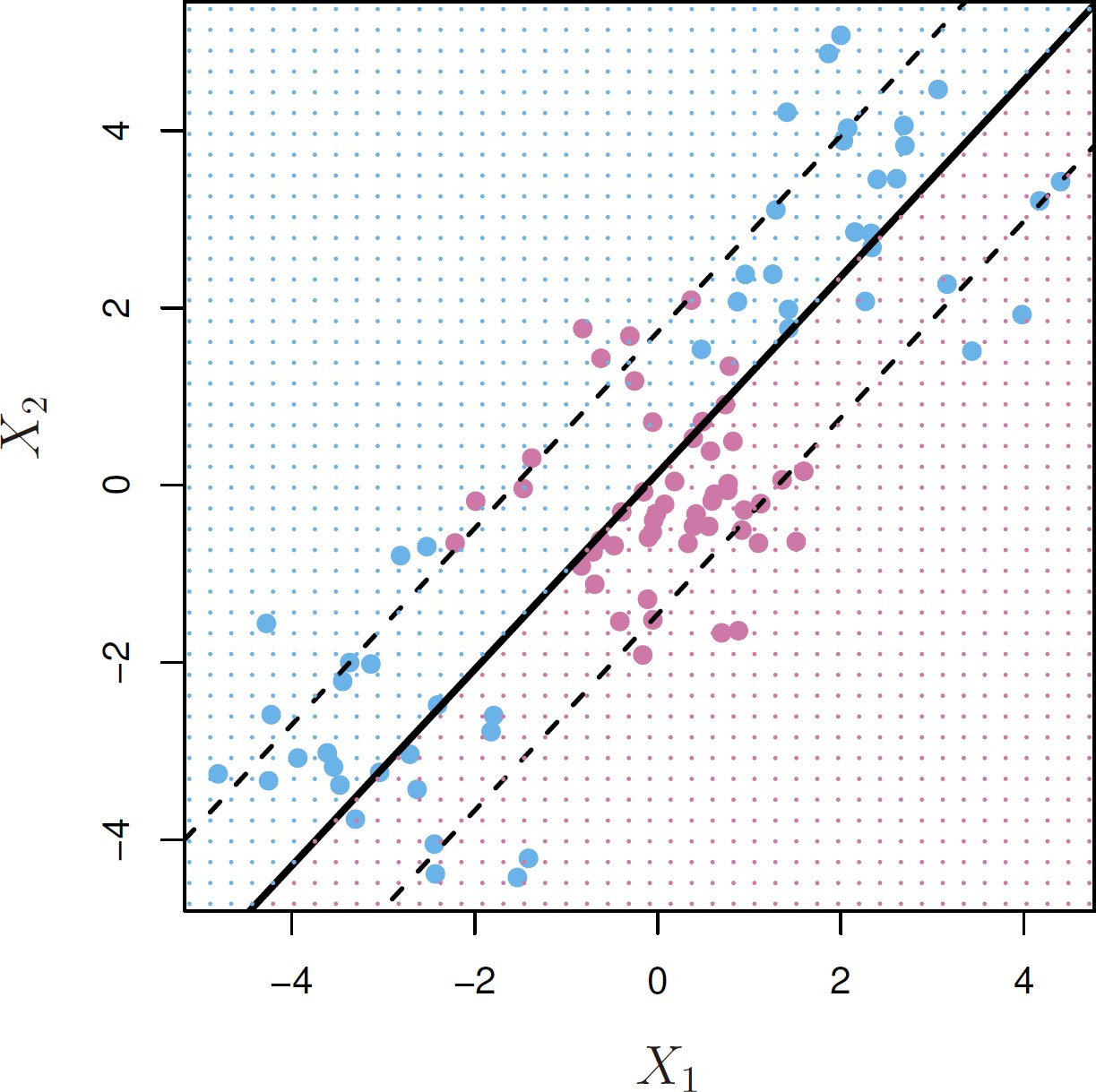
©copyright CIEL 2015 **17**

# Support Vector Classifier [cont.]

* 𝐶𝐶 is a regularization parameter

©copyright CIEL 2015 **18**

# Support Vector Classifier [cont.]

* Linear boundary can fail
  + Sometime a linear boundary simply won't work, no matter what value of 𝐶𝐶
  + The example on the right is such a case

©copyright CIEL 2015 **19**

# Feature Expansion

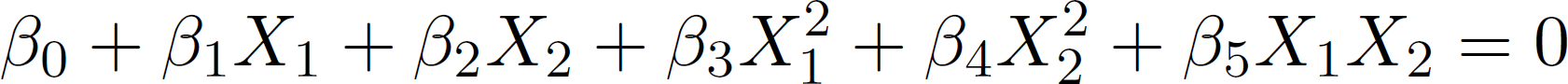
* Enlarge the space of features by including transformations
  + E.g., 𝑋𝑋2, 𝑋𝑋3, 𝑋𝑋1𝑋𝑋2, 𝑋𝑋1𝑋𝑋2, …

1 1 2

* + Hence go from a 𝑝𝑝-dimensional space to a 𝑀𝑀 > 𝑝𝑝 dimensional space
* Fit a support-vector classier in the enlarged space
* This results in non-linear decision boundaries in the original space
* Example
  + Suppose we use (𝑋𝑋2, 𝑋𝑋3, 𝑋𝑋1𝑋𝑋2, 𝑋𝑋1𝑋𝑋2) instead of just (𝑋𝑋1, 𝑋𝑋2)

1 1 2

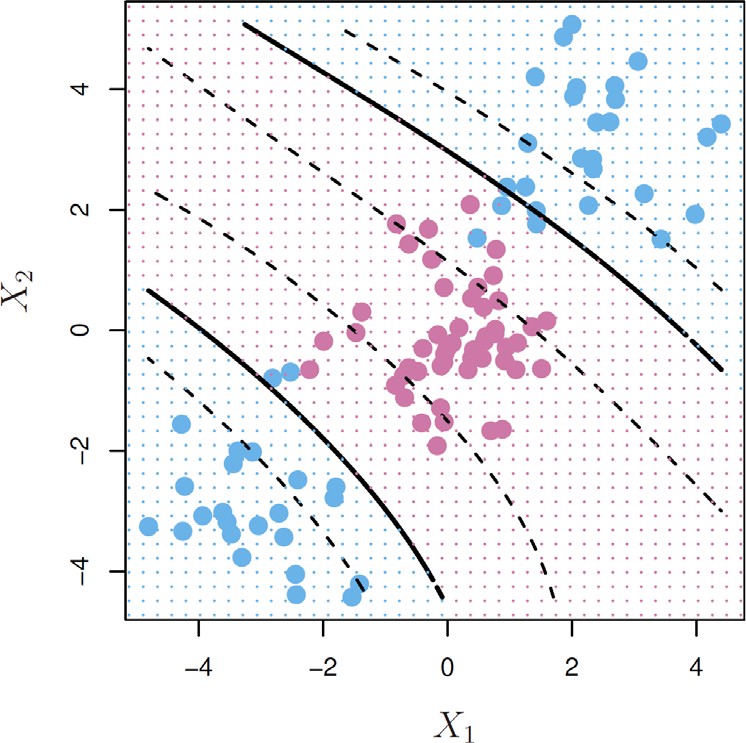
* + Then the decision boundary would be of the form

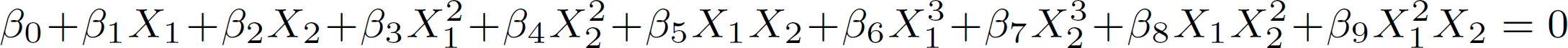


* This leads to nonlinear decision boundaries in the original space (quadratic conic sections)

©copyright CIEL 2015 **20**

# Cubic Polynomials

* Here we use a basis expansion of cubic polynomials
* From 2 variables to 9
* The support-vector classifier in the enlarged space solves the problem in the lower-dimensional space



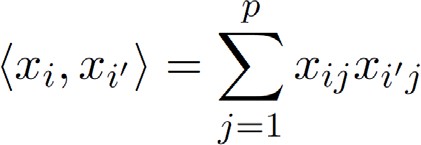
©copyright CIEL 2015 **21**

# Nonlinearities and Kernels

* Polynomials (especially high-dimensional ones) get wild rather fast
* There is a more elegant and controlled way to introduce nonlinearities in support vector classifiers through the use of kernels
* Before we discuss these, we must understand the role of inner products in support-vector classifiers

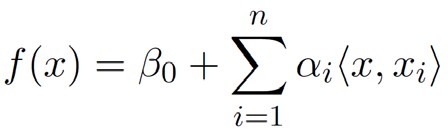
©copyright CIEL 2015 **22**

# Inner Products and Support Vectors

* Inner product between vectors
* The linear support vector classifier can be represented as

𝑛𝑛 parameters

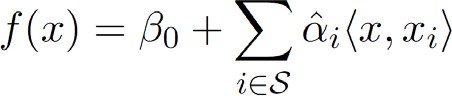
𝑥𝑥𝑖𝑖 , 𝑥𝑥𝑖𝑖′

* To estimate the parameters 𝛼𝛼1, ⋯ , 𝛼𝛼𝑛𝑛 and 𝛽𝛽0, all we need are the between all pairs of training observations

𝑛𝑛

2

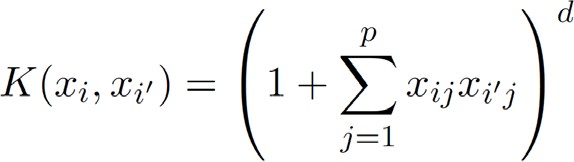
inner products

* It turns out that most of the 𝛼𝛼�𝑖𝑖 can be zero:
  + is the support set of indices 𝑖𝑖 such that

𝛼𝛼�𝑖𝑖 > 0

©copyright CIEL 2015 **23**

# Kernels and Support Vector Machines

* If we can compute inner-products between observations, we can fit a SV classifier
  + Can be quite abstract
* Some special kernel functions can do this for us, e.g.,

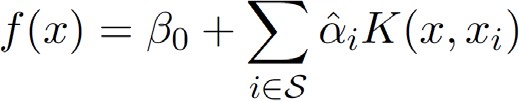
computes the inner-products needed for 𝑑𝑑 dimensional polynomials

o basis functions

𝑝𝑝 + 𝑑𝑑

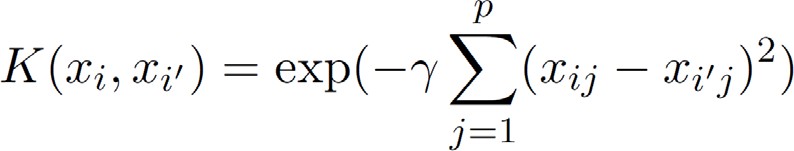
𝑑𝑑

* The solution has the form



©copyright CIEL 2015 **24**

# Radial Kernel





* Implicit feature space
  + Very high dimensional
* Controls variance by squashing down most dimensions severely

©copyright CIEL 2015 **25**

# Example: Heart Data

* ROC curve is obtained by changing the threshold 0 to threshold 𝑡𝑡 in false positive and true positive rates as 𝑡𝑡 varies

𝑥𝑥

* + Here we see ROC curves on training data

𝑓𝑓̂

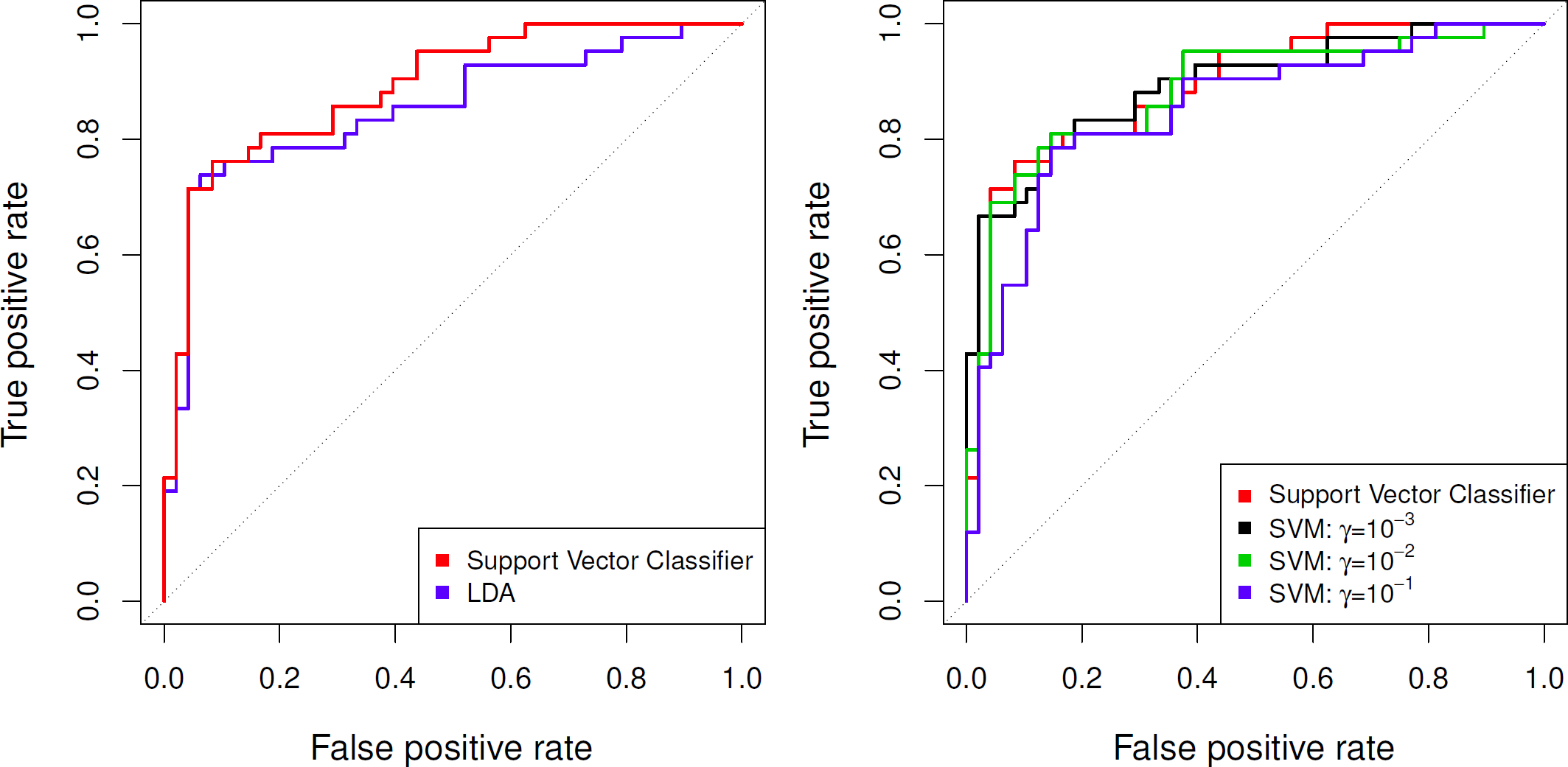
* 𝑡𝑡, and recording



©copyright CIEL 2015 **26**



# Example continued: Heart Test Data



©copyright CIEL 2015 **27**

# SVMs: More than 2 Classes?

* The SVM as defined works for 𝐾𝐾 = 2 classes
* What do we do if we have 𝐾𝐾 > 2 classes?
  + OVA: One versus All
    - Fit 𝐾𝐾 different 2-class SVM classifiers
    - Each class versus the rest

̂

𝑘𝑘

𝑓𝑓

(𝑥𝑥), 𝑘𝑘 = 1, ⋯ , 𝐾𝐾

* + - Classify 𝑥𝑥∗ to the class for which is
  + OVO: One versus One

𝐾𝐾

2

̂

𝑘𝑘

𝑓𝑓

(𝑥𝑥∗) the largest

o Fit all

pairwise classifiers

̂

𝑘𝑘𝑙𝑙



𝑥𝑥

𝑓𝑓

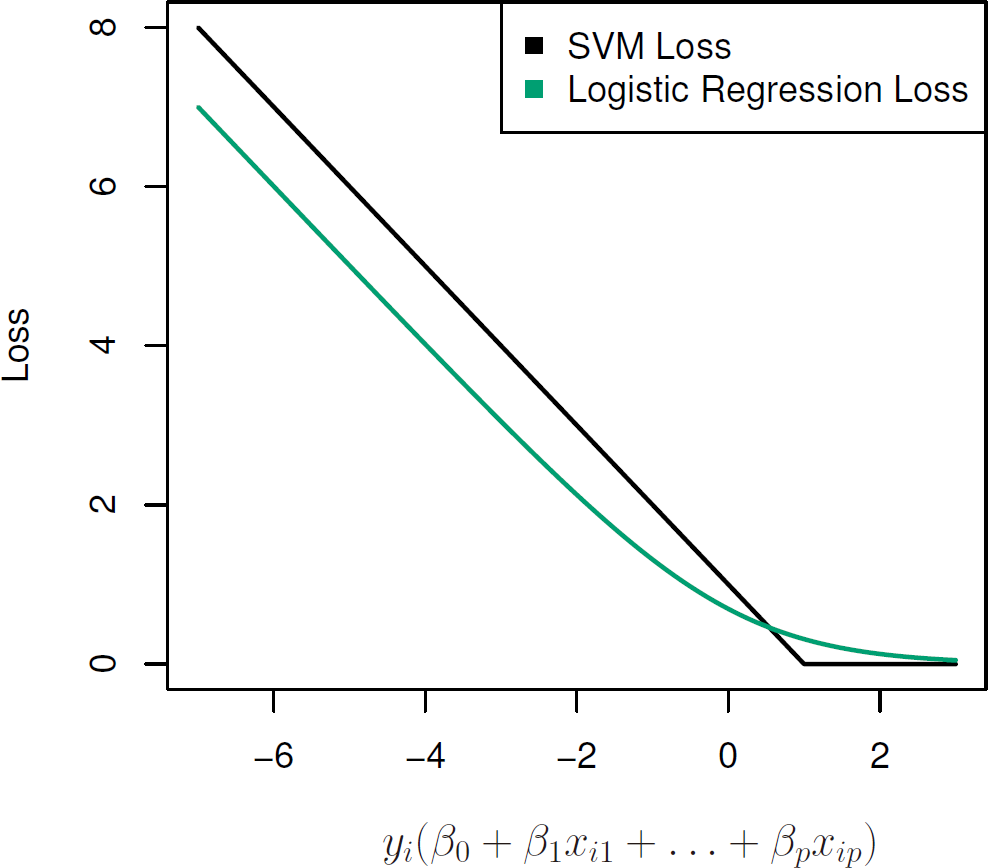
o Classify 𝑥𝑥∗ to the class that wins the most pairwise competitions

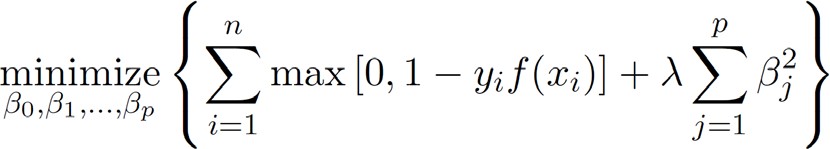
* Which to choose?
  + If 𝐾𝐾 is not too large, use OVO

©copyright CIEL 2015 **28**

# [FYI] Support Vector versus Logistic Regression?

* With 

can rephrase support-vector classifier optimization as



* + This has the form loss plus penalty
  + The loss is known as the hinge loss
  + Very similar to “loss” in logistic regression (negative log-likelihood)

©copyright CIEL 2015 **29**

# Which to Use: SVM or Logistic Regression

* When classes are (nearly) separable, SVM does better than LR. So does LDA
* When not, LR (with ridge penalty) and SVM very similar
* If you wish to estimate probabilities, LR is the choice
* For nonlinear boundaries, kernel SVMs are popular
  + Can use kernels with LR and LDA as well, but computations are more expensive

©copyright CIEL 2015 **30**



**Python Lab**

* Support vector machines

## Python lab

* Summary & Next class

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# 9.6 Lab: Support Vector Machines

* + Using Python Libraries
    - Import the libraries that are often used for data analysis

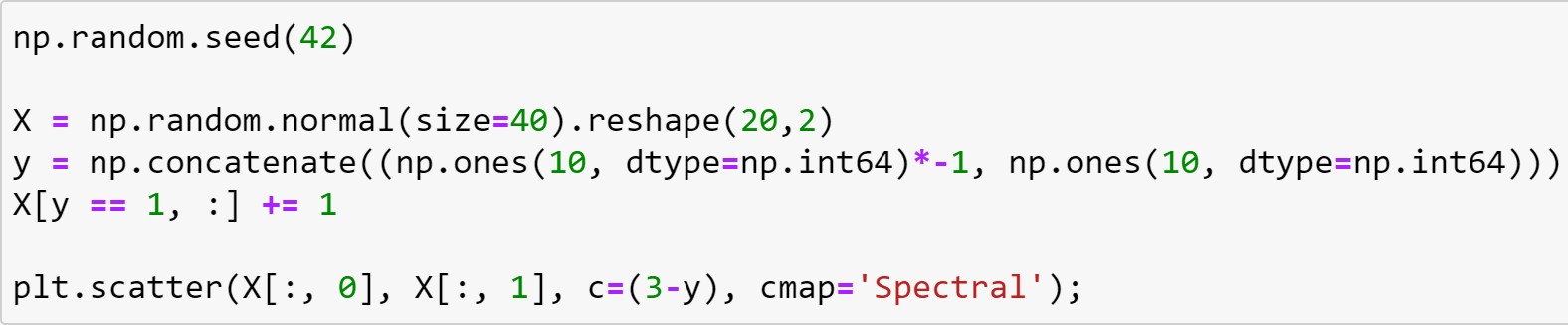
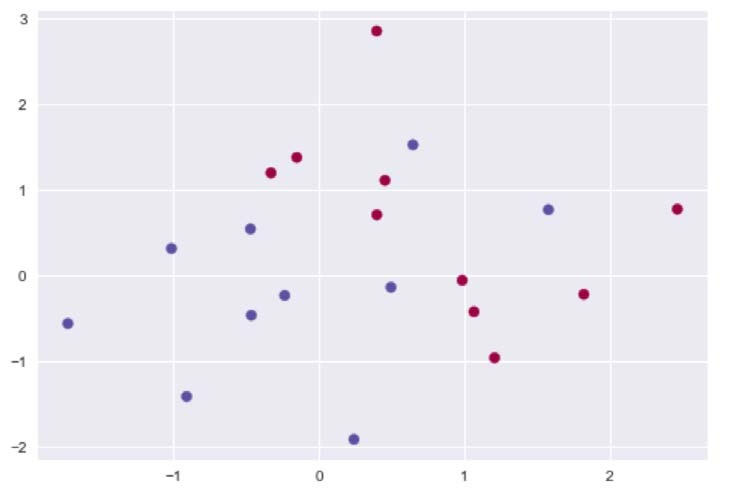


©copyright CIEL 2015 **32**

# 9.6.1 Support Vector Classifier

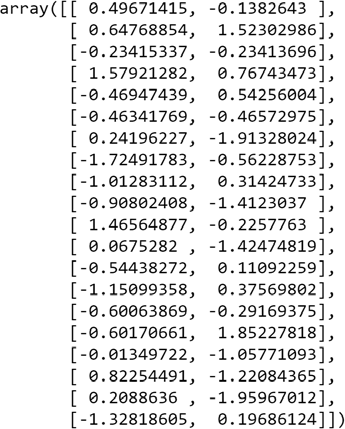
* + Generating data: for checking whether classes are linearly separable

20x2 random numbers

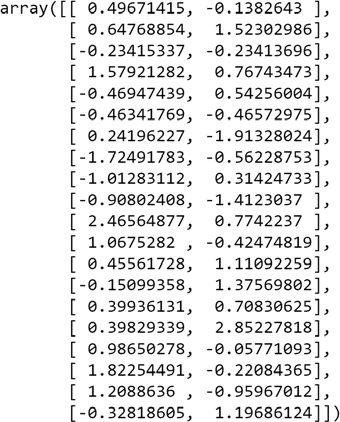


Join a sequence of arrays along an existing axis

Different color for each y

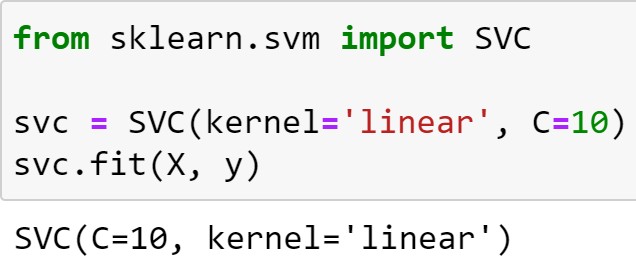


©copyright CIEL 2015 **33**



# 9.6.1 Support Vector Classifier

* + Support vector classifier: using a linear kernel



Performing binary and multi-class classification on a dataset

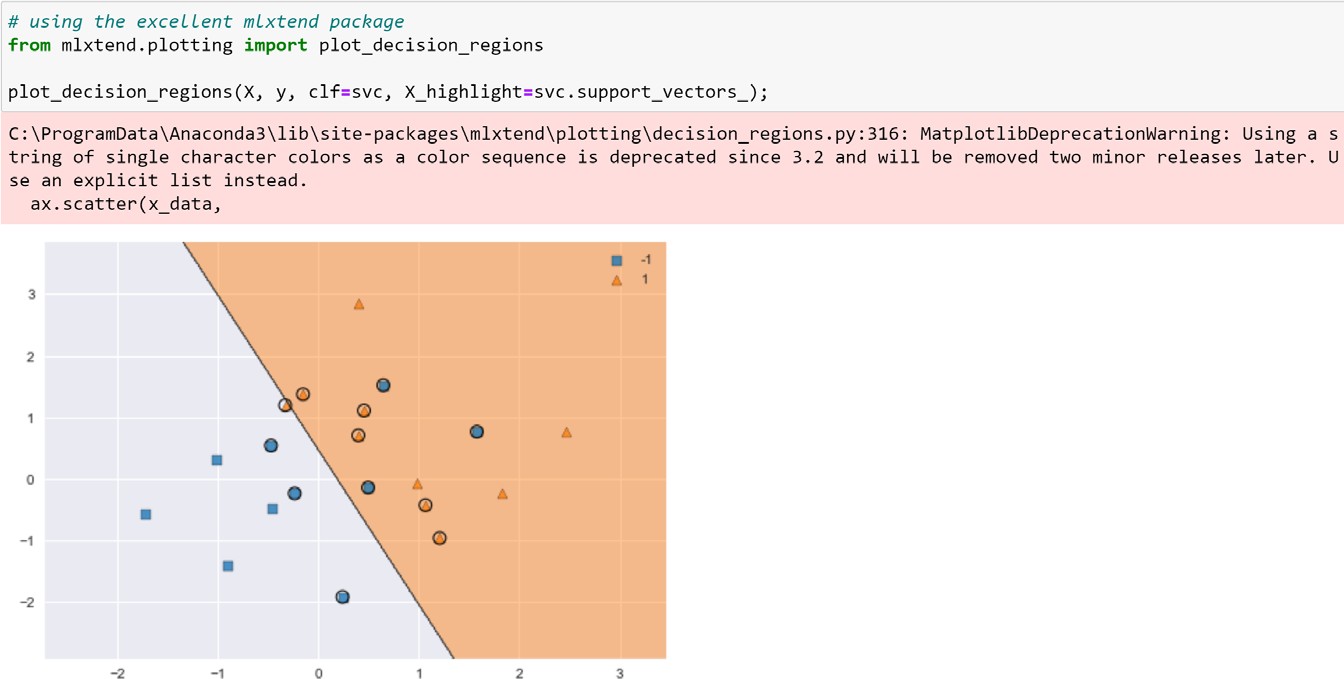
C-support vector classification using linear kernel

* C: Regularization parameter. The strength of the regularization is inversely proportional to C. Must be strictly positive. The penalty is a squared l2 penalty

©copyright CIEL 2015 **34**

# 9.6.1 Support Vector Classifier

* + Visualization for SVC
    - mlxtend (machine learning extensions): a Python library of useful tools for day-to-day data science tasks.



Classifier object

An array with data points that are used to highlight samples in X

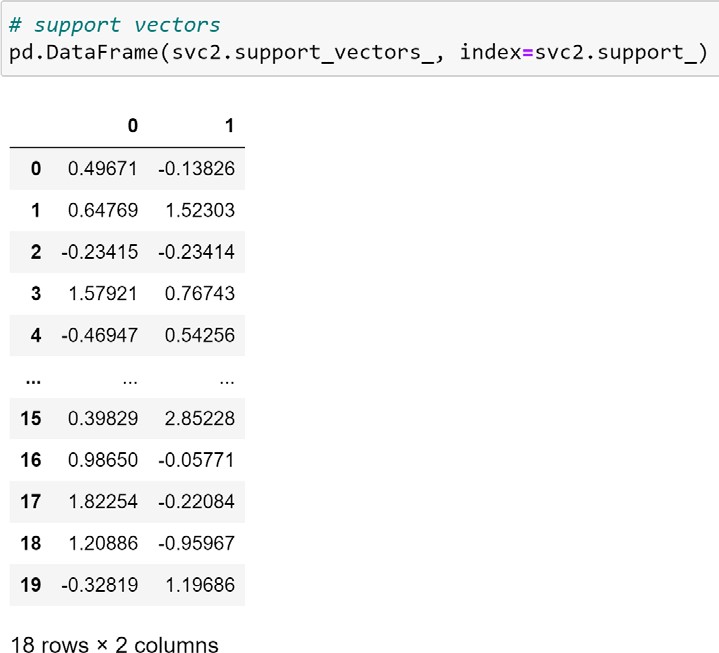
A function for plotting decision regions of classifiers in 1 or 2 dimensions

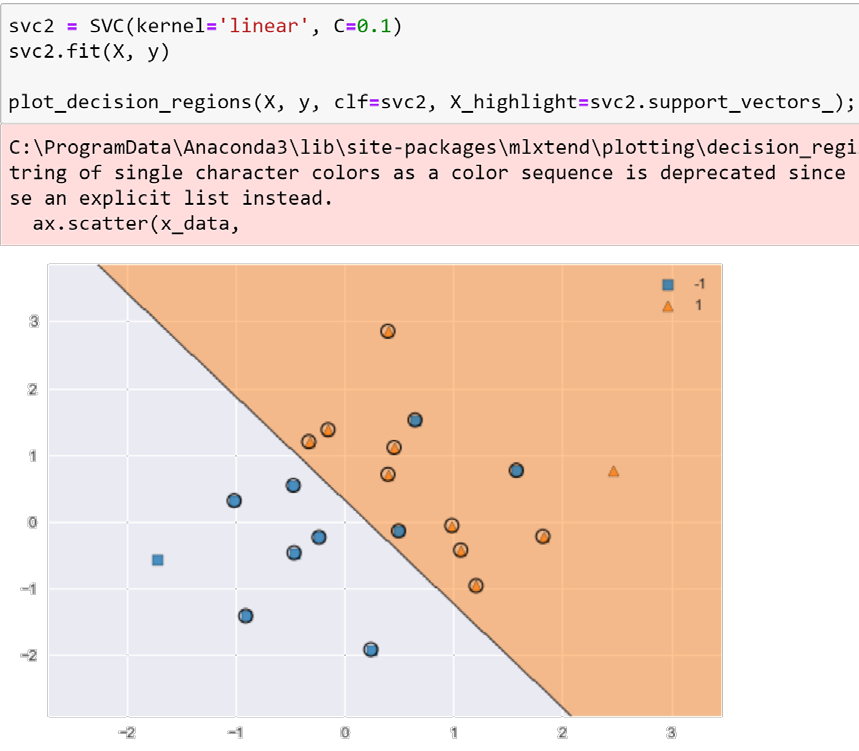
Support vectors

Indices of support vectors

©copyright CIEL 2015 **35**

# 9.6.1 Support Vector Classifier

* + Using a smaller value of cost parameter C
    - A larger number of support vectors because margin is now wider



©copyright CIEL 2015 **36**

# 9.6.1 Support Vector Classifier

* + Comparing SVMs with a linear kernel, using a range of values of C

A range of C values using a dictionary



10-fold CV

Exhaustive search over specified parameter values for an estimator

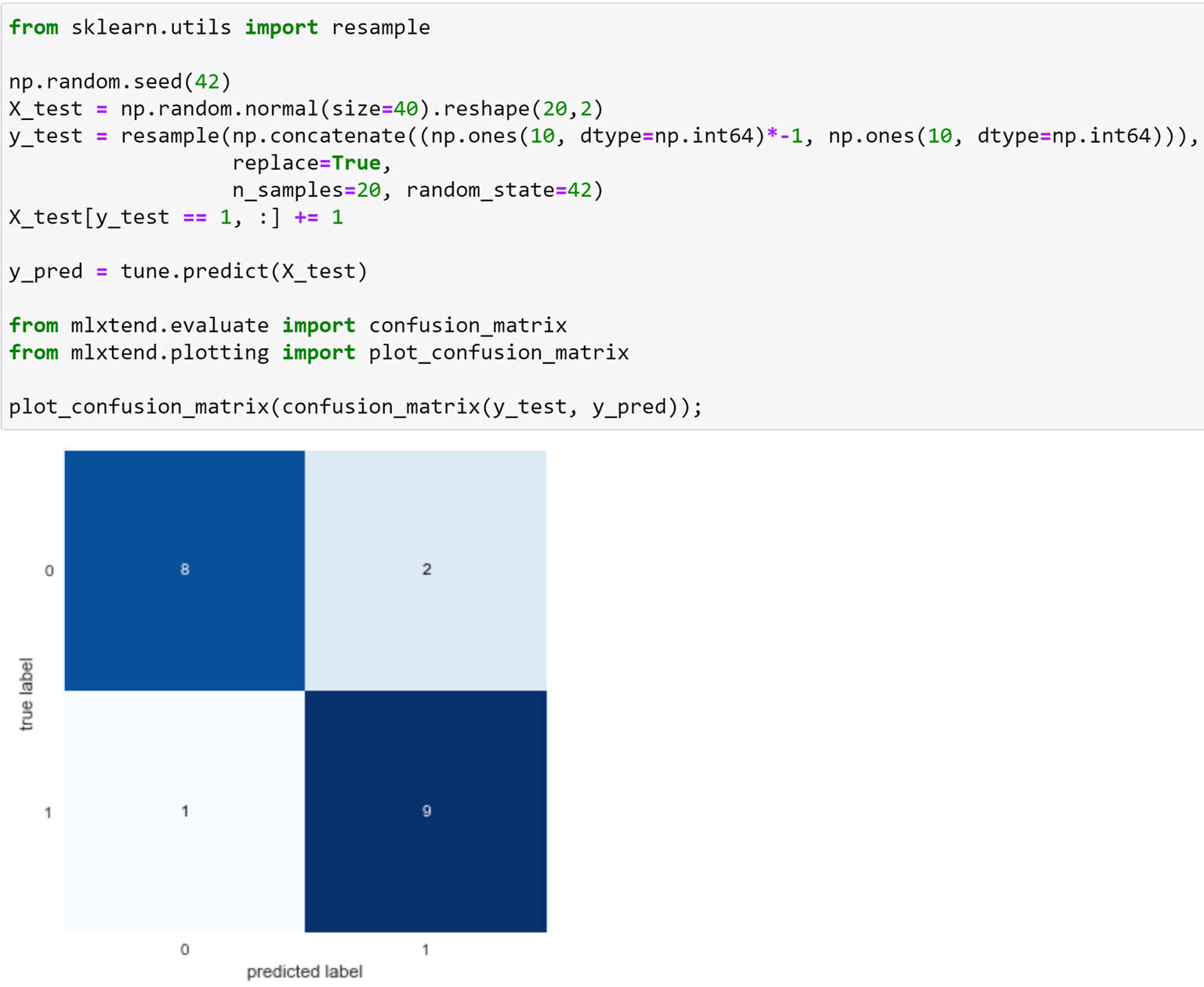
In this example, C=0.001 is best

©copyright CIEL 2015 **37**

# 9.6.1 Support Vector Classifier

* + Test error for SVM with a designed C

Random response



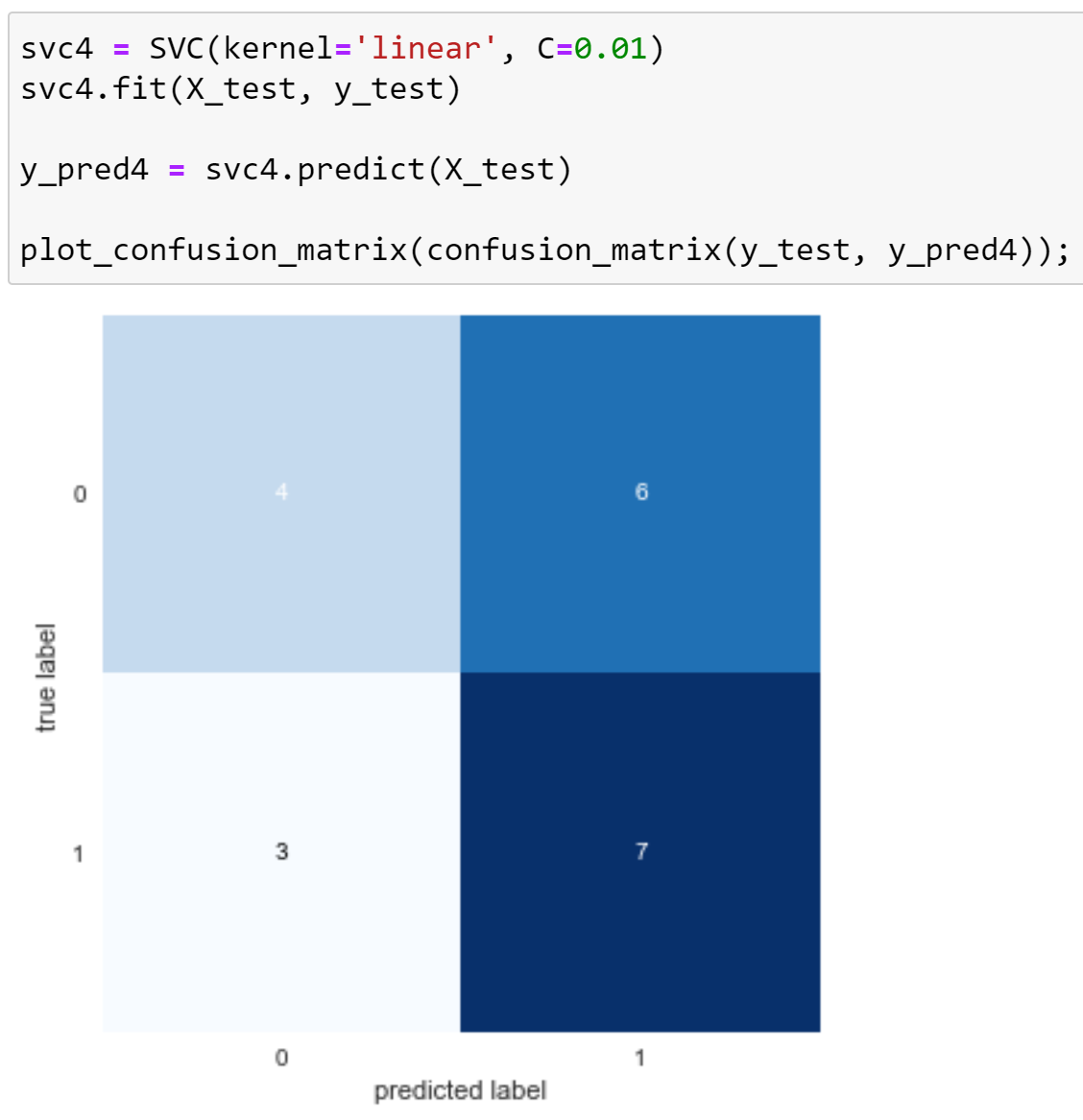
Functions for generating confusion matrices

Utility function for visualizing confusion matrices via matplotlib

©copyright CIEL 2015

# 9.6.1 Support Vector Classifier

* + Test error for SVM with a worse C = 0.01 A smaller C: additionally misclassified

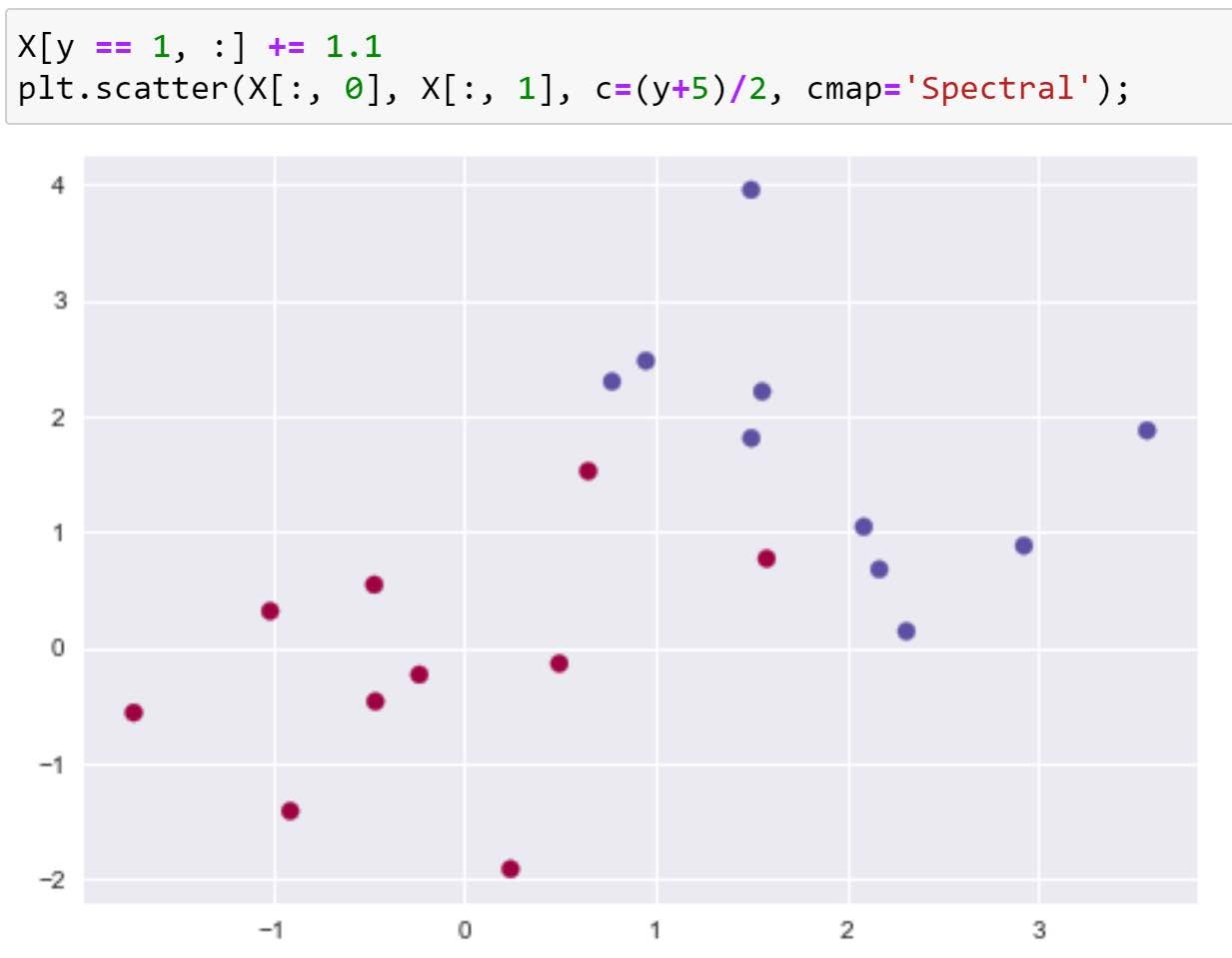


©copyright CIEL 2015

# 9.6.1 Support Vector Classifier

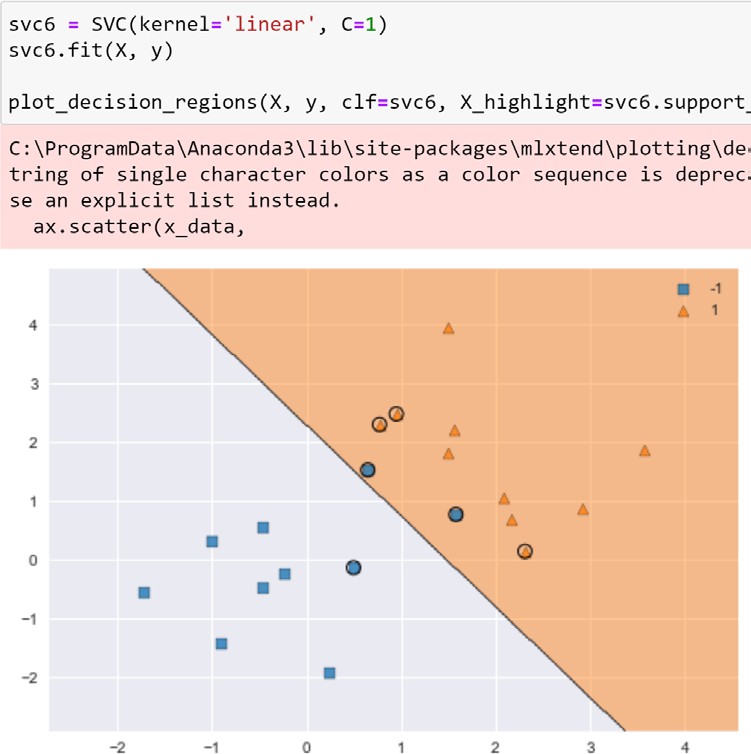
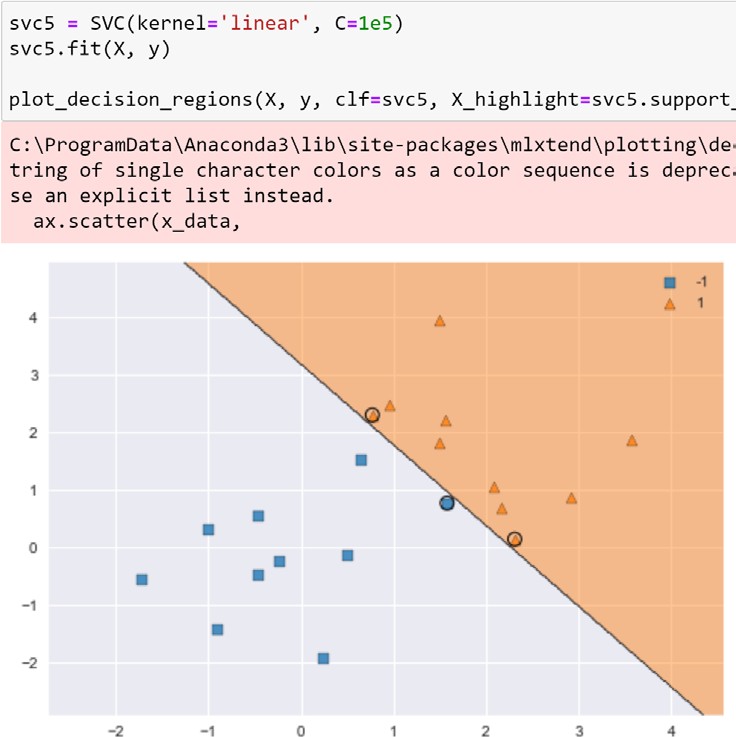
* + Generating data barely linearly separable

©copyright CIEL 2015



# 9.6.1 Support Vector Classifier

* + SVCs for barely linearly separable data



A very large C



A smaller C

Linearly separated

Violated

©copyright CIEL 2015

# 9.6.2 Support Vector Machine

* + Data generation for SVM



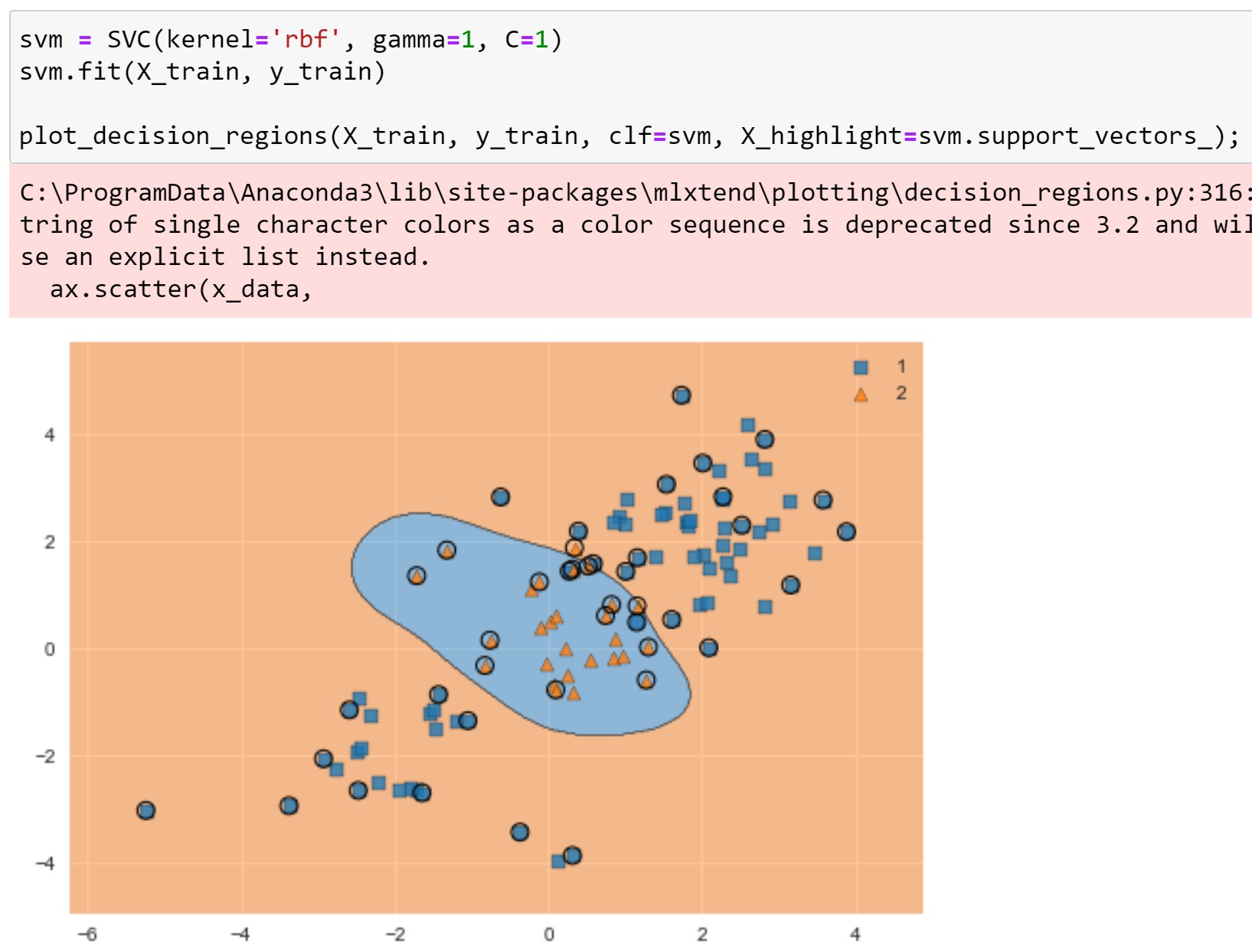
Validation set

©copyright CIEL 2015

# 9.6.2 Support Vector Machine

* + Support vector machine: using a radial kernel with 𝛾𝛾 = 1 and C=1

C-support vector classification using radial kernel



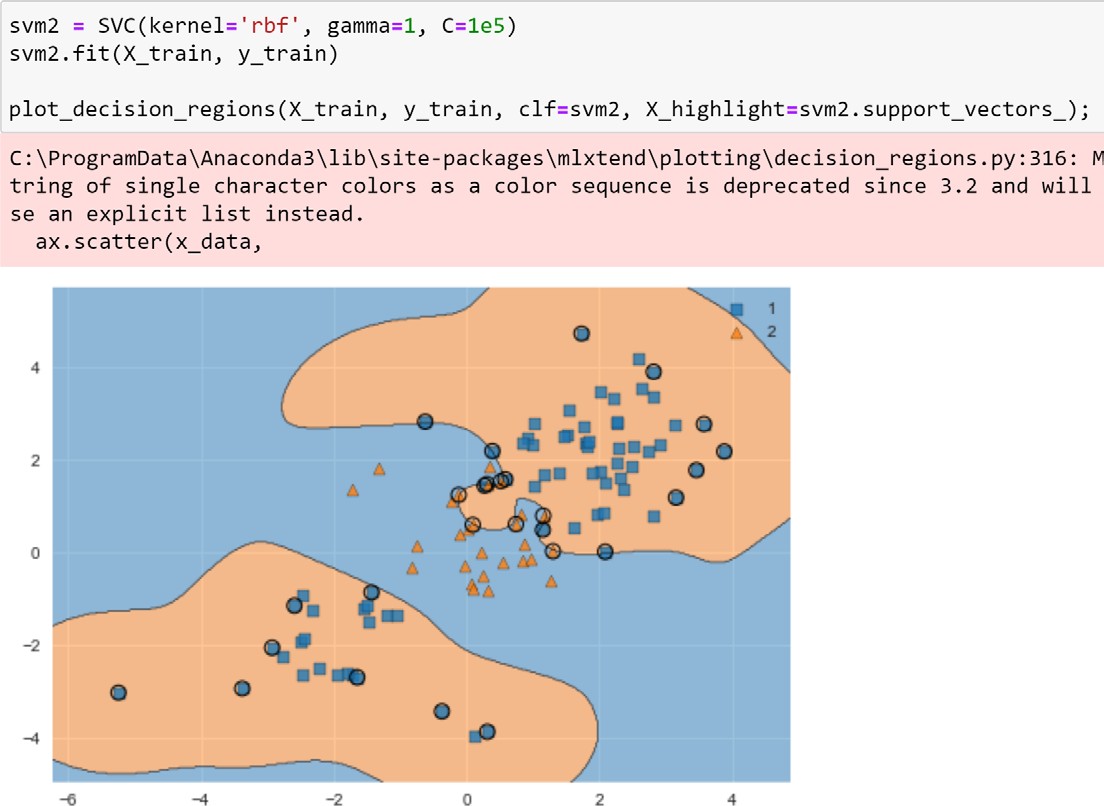
* + - kernel: kernel{‘linear’, ‘poly’, ‘rbf’, ‘sigmoid’, ‘precomputed’}, default=’rbf’

©copyright CIEL 2015 **43**

# 9.6.2 Support Vector Machine

* + Support vector machine: using a radial kernel with 𝛾𝛾 = 1 and a very large C=1e5

Increasing C value

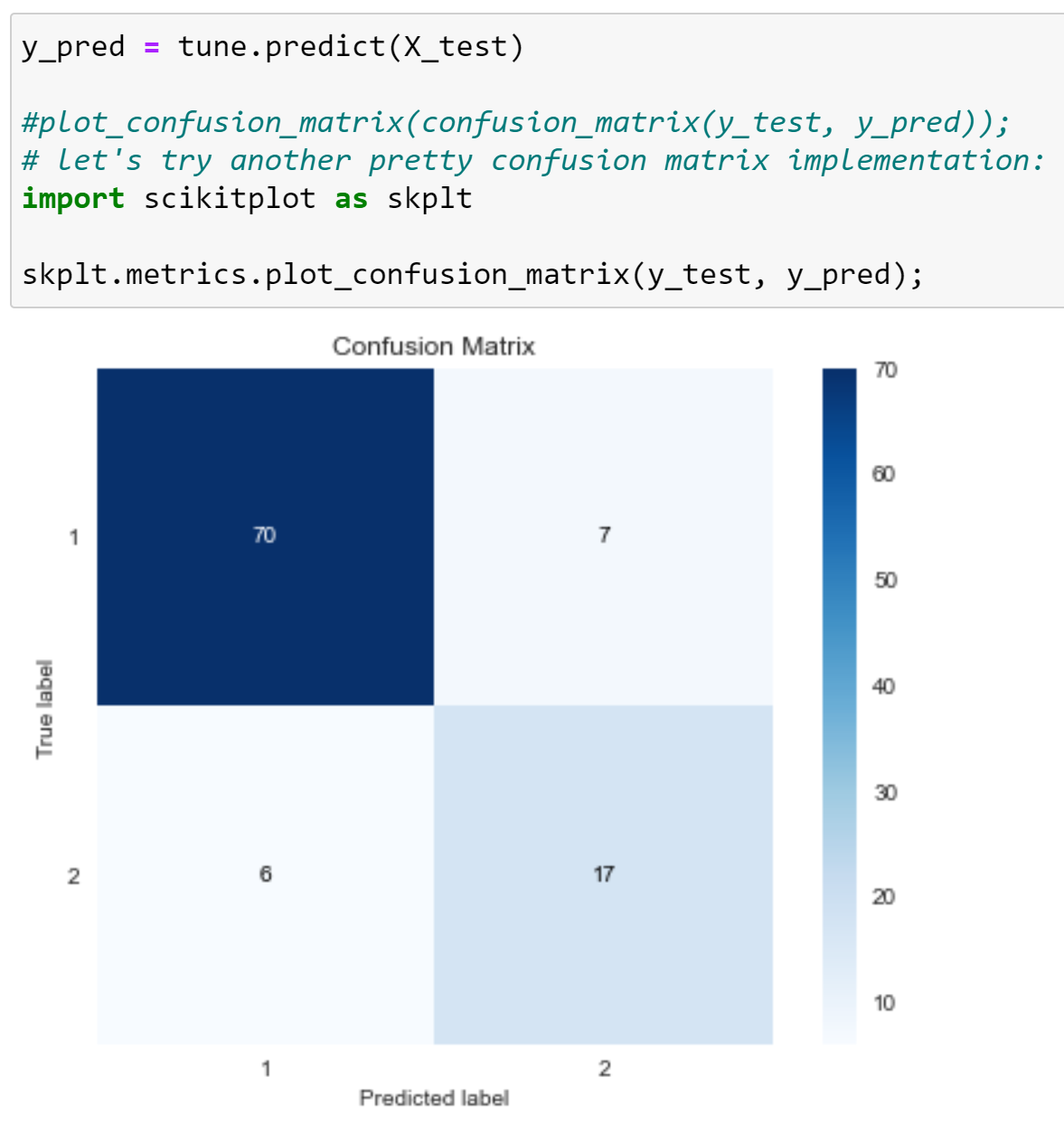


: reducing training errors, but more irregular decision boundary and being at risk of overfitting data

©copyright CIEL 2015 **44**

# 9.6.2 Support Vector Machine

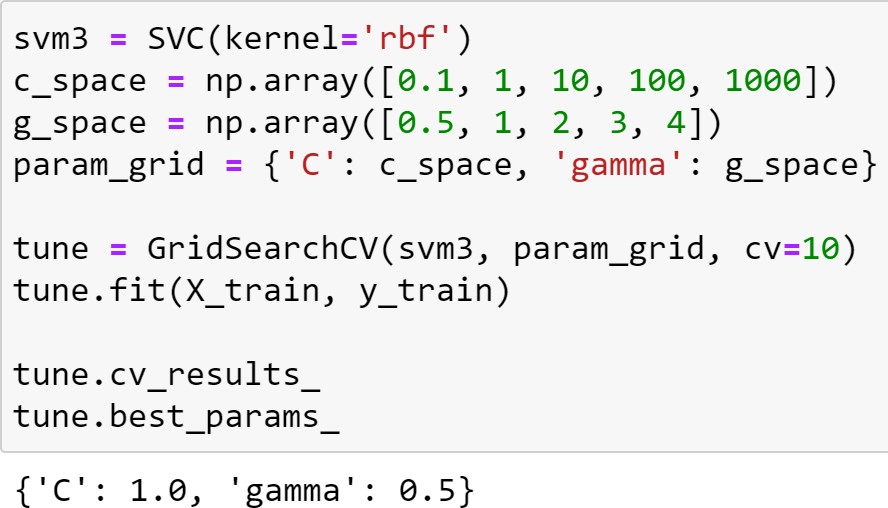
* + Design of C and 𝛾𝛾, and confusion matrix



Confusion matrix using scikitplot

©copyright CIEL 2015

**45**

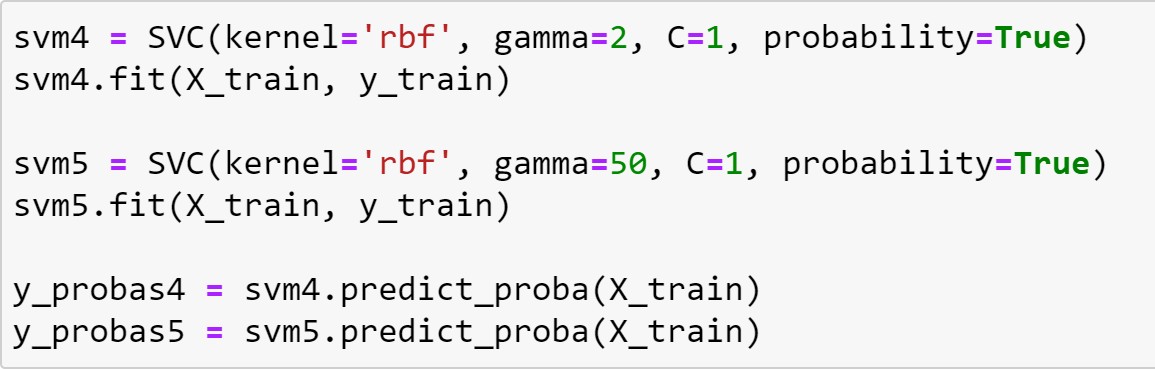


Design of C and 𝛾𝛾



# 9.6.3 ROC Curves

* + ROC curve for different 𝛾𝛾: SVM fitting

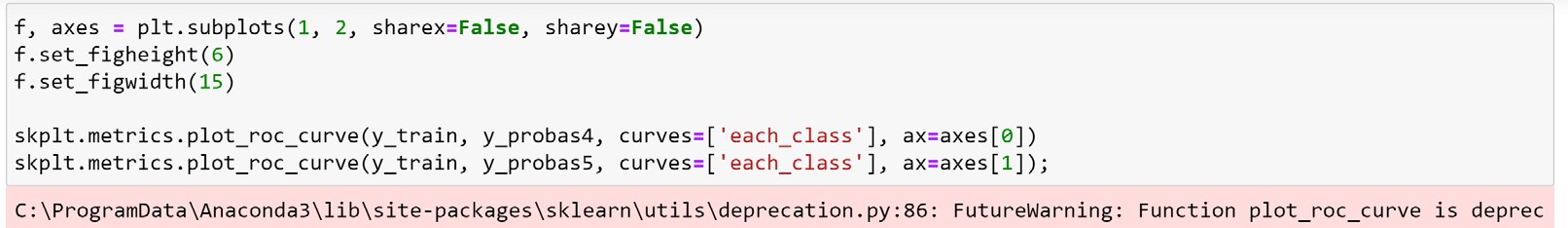


Compute probabilities of possible outcomes for samples in X

©copyright CIEL 2015 **46**

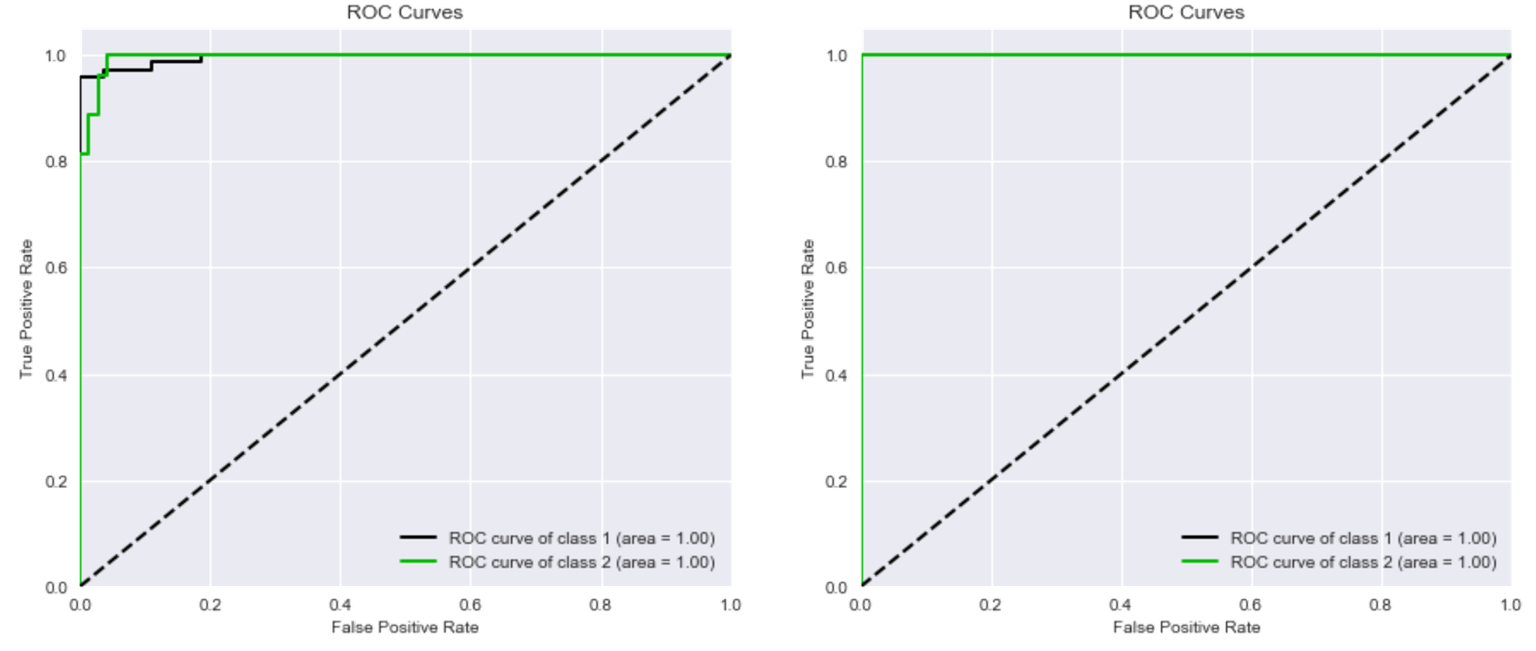
# 9.6.3 ROC Curves

* + ROC curve for different 𝛾𝛾: drawing ROC curves for training errors



Plotting the ROC curve

ROC curve with 𝛾𝛾 = 2



for training data

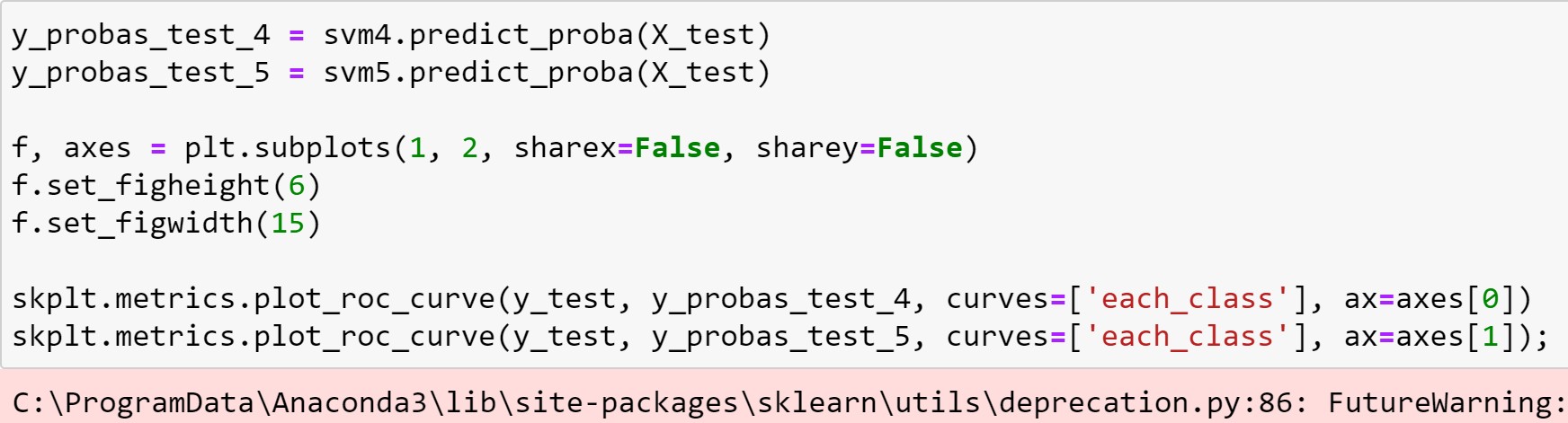
ROC curve with 𝛾𝛾 = 50

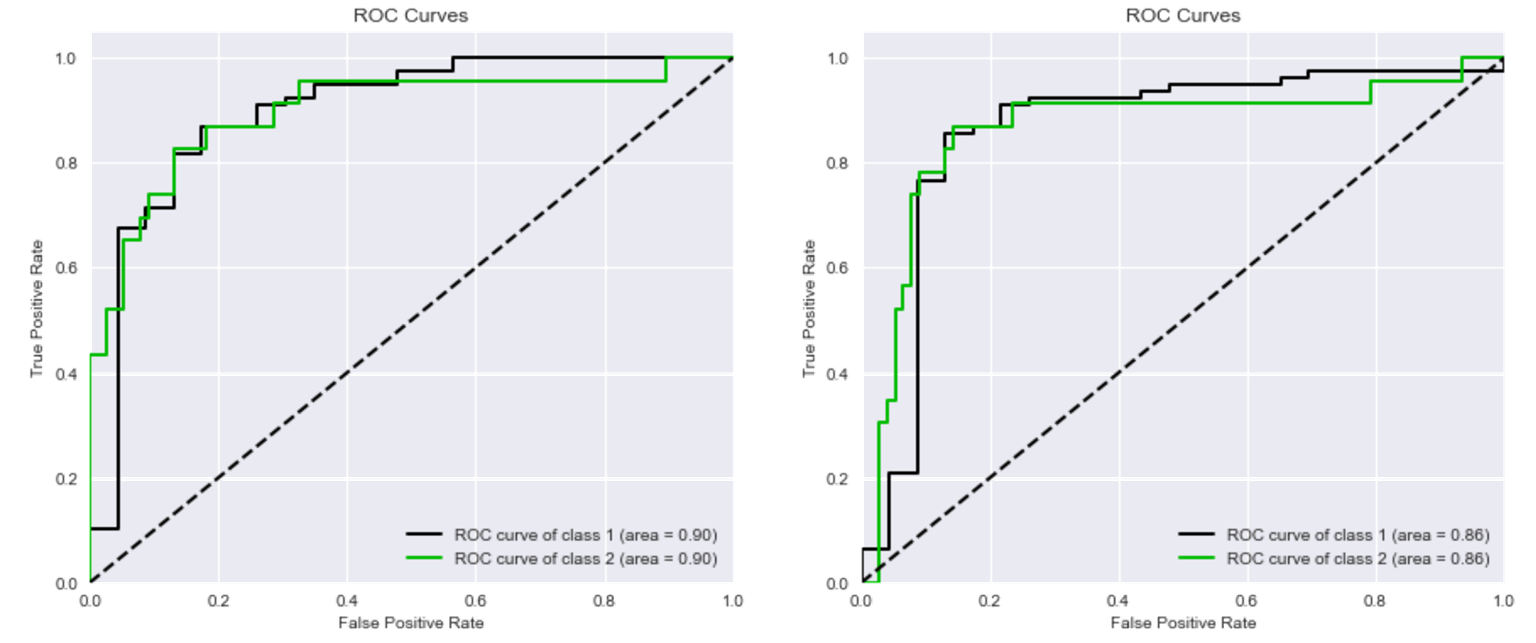
for training data



©copyright CIEL 2015 **47**

# 9.6.3 ROC Curves

* + ROC curve for different 𝛾𝛾: drawing ROC curves for test errors



ROC curve with 𝛾𝛾 = 2

for test data

ROC curve with 𝛾𝛾 = 50

for test data

©copyright CIEL 2015

**48**

# 9.6.4 SVM with Multiple Classes

* + SVM with multiple classes: data generation Return a new array of given shape and type, filled



with fill\_value

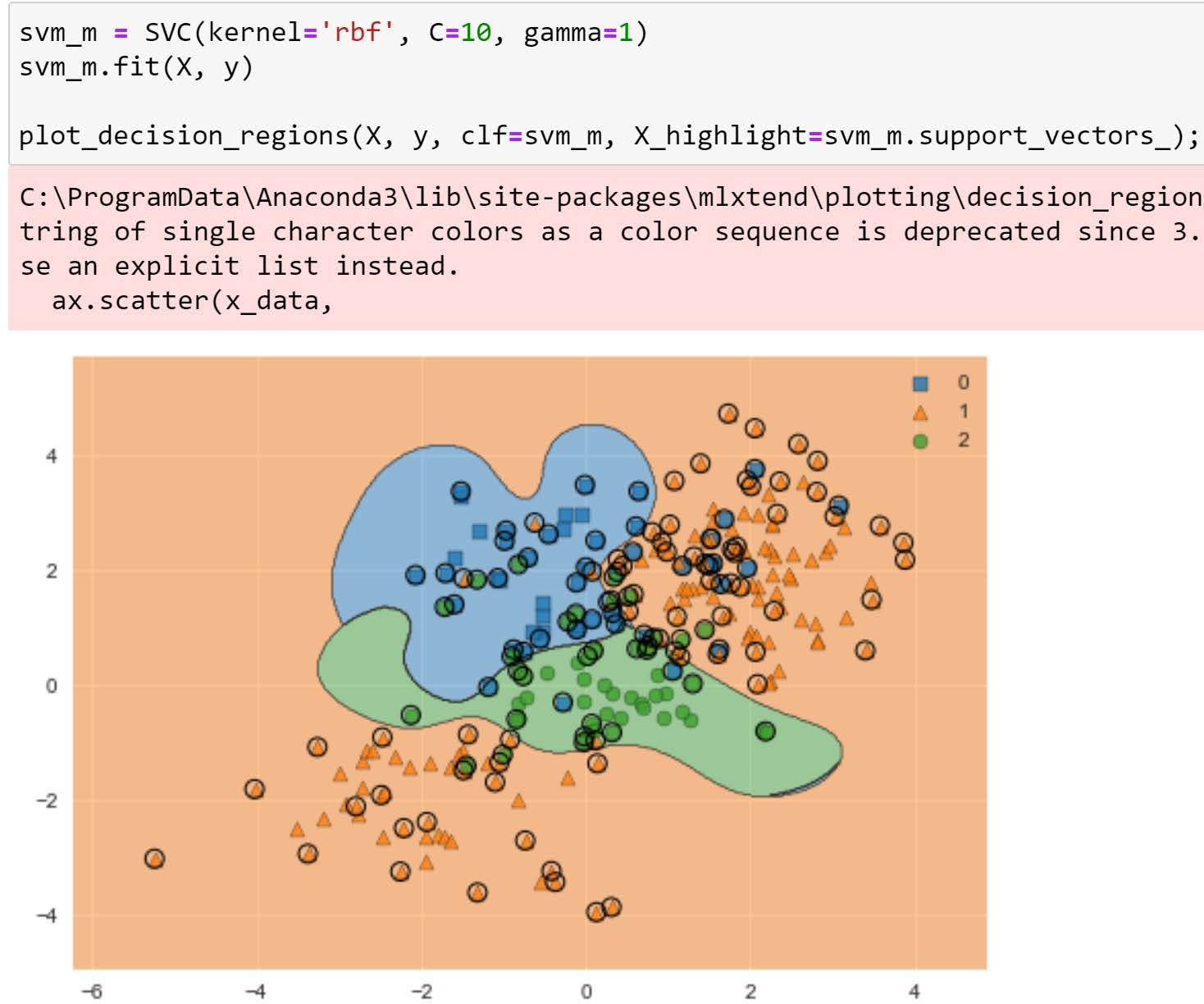
(in this example, shape = 150, fill\_value = 1)

 A total of 3 classes

©copyright CIEL 2015 **49**

# 9.6.4 SVM with Multiple Classes

* + SVM with multiple classes: fitting still using SVC



©copyright CIEL 2015 **50**



**Summary & Next Class**

* + - Support vector machines
    - Python lab

## Summary & Next class

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# Summary

* + Support vector machines
* Maximal margin classifier: separating hyperplane
* Support vector classifier (SVC): soft margin
* Support vector machine (SVM): kernel, e.g., radial kernel

o Decreasing complexity

* + Python lab
* Using scikit-learn
  + SVC: sklearn.svm.svc with kernel=‘linear’
  + Using mlxtend for visualizing SVC decision boundary
  + SVM: also, using sklearn.svm.svc with kernel=‘radial’
  + SVM with multiple classes: fitting still using SVC

©copyright CIEL 2015 **52**

# Assignments

* + eClass > Assignments
* Upload files (do not compress them)
  + Python practices in today’s lecture
* Upload a single ipynb file
* Referring to the lecture slides marked with [P]
* File name: “StudentID” + “\_AssignmentNo w/ 2 digits” + “\_1.ipynb”, e.g., **20211234\_02\_1.ipynb**
  + Textbook exercise problems for today’s lecture
* Conceptual
  + Solving the given problems, then upload your own solution (only docx/hwp formats, not pdf/jpg/bmp etc.)
  + Only include your answers (not need to describe problems)
  + File name: “StudentID” + “\_AssignmentNo w/ 2 digits” + “\_2.ipynb”, e.g., **20211234\_02\_2.docx**
* Applied
  + Implement your Python code for the given problems, then upload another single ipynb file
  + File name: “StudentID” + “\_AssignmentNo w/ 2 digits” + “\_1.ipynb”, e.g., **20211234\_02\_3.ipynb**
  + If not complying with the above policies, some penalty on assignment scores may be given.

©copyright CIEL 2015 **53**

# Course Schedule (Tentative)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Topics** | **Note** | **Date (W)** | **Date (M)** |
| 1 | Orientation, Statistical Learning (Ch2) | Online | 03/03 | 03/08 |
| 2 | Statistical Learning (Ch2), Python Programming | Online | 03/10 | 03/15 |
| 3 | Probability & Statistics | Online | 03/17 | 03/22 |
| 4 | Probability & Statistics | Online | 03/24 | 03/29 |
| 5 | Linear Regression (Ch3) | Online | 03/31 | 04/05 |
| 6 | Linear Regression (Ch3) | Online | 04/07 | 04/12 |
| 7 | Classification (Ch4) | Online | 04/14 | 04/19 |
| 8 | **Midterm exam** | **Class hours (W1-W7)** | **04/21** | **04/26** |
| 9 | Resampling Methods (Ch5) | Online | 04/28 | 05/03 |
| 10 | Linear Model Selection and Regularization (Ch6) | Online | 05/05 | 05/10 |
| 11 | Moving Beyond Linearity (Ch7) | Online | 05/12 | 05/17 |
| 12 | Tree-Based Methods (Ch8) | Online | 05/19 | 05/24 |
| 13 | Support Vector Machines (Ch9) | Online | 05/26 | 05/31 |
| **14** | Unsupervised Learning (Ch10) | Online | 06/02 | 06/07 |
| 15 | **Final exam** | **6pm (W9-W14)** | **06/10Th** | **06/10Th** |

©copyright CIEL 2015 **54**