6. Support Vector Machines

Jesper Armouti-Hansen

University of Cologne

January 28, 2019

Course website

jeshan49.github.io/eemp2/

Today

- Lecture[1]:
  - Maximal Margin Classification
  - Linear Support Vector Classification
  - General Support Vector Classification
  - Support vector Regression

[1] Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani

Methods covered

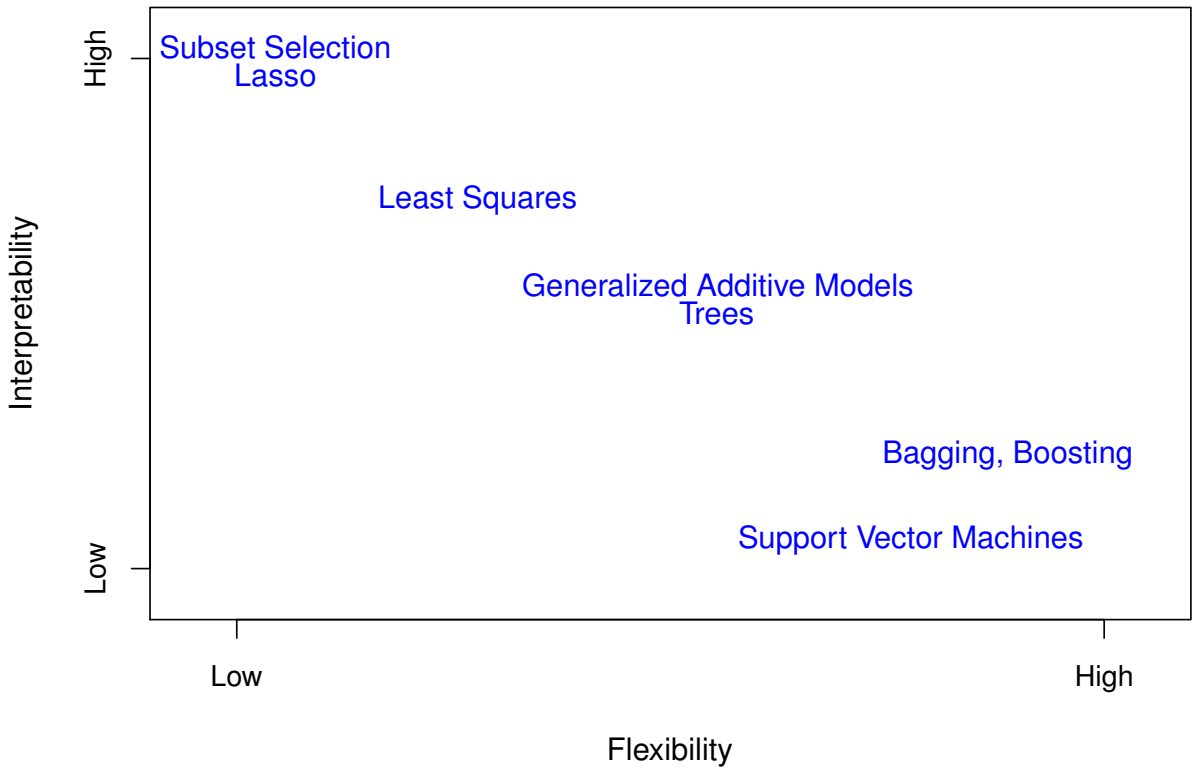


Figure: Tradeoff between flexibility and interpretability, using different learning methods (See ISLR p. 25)

Introduction to Support Vector Machines (SVMs)

- The SVM is an approach developed for classification in the computer science community in the 1990s
- It is capable of performing
  - linear and nonlinear classification;
  - linear and nonlinear regression
- We will mostly see SVMs for classification today, and only shortly discuss its approach for regression
- In general, the concept of SVMs for classification applies only to binary classification, but we will also consider generalizations
- It has gained popularity due to its performance on small- and medium-sized complexed datasets

Maximal Margin Classification

- The fundamental idea behind SVMs is the concept of an *optimal separating hyperplane*

- In a p-dimensional space, a *hyperplane* is a flat affine subspace of dimension $p - 1$
  - $p = 2 \rightarrow$ a line
  - $p = 3 \rightarrow$ a plane
- In a p-dimensional space, a *hyperplane* is the set of points $X = (X_1, \ldots, X_p)^T$ which solves

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p = 0 \tag{1}$$

  for some $\beta_0, \ldots, \beta_p$
- Thus, if (1) is positive for some X^*, then we know that X^* lies to one side of the hyperplane (and analogously if (1) is negative)

Example of a hyperplane with $p = 2$

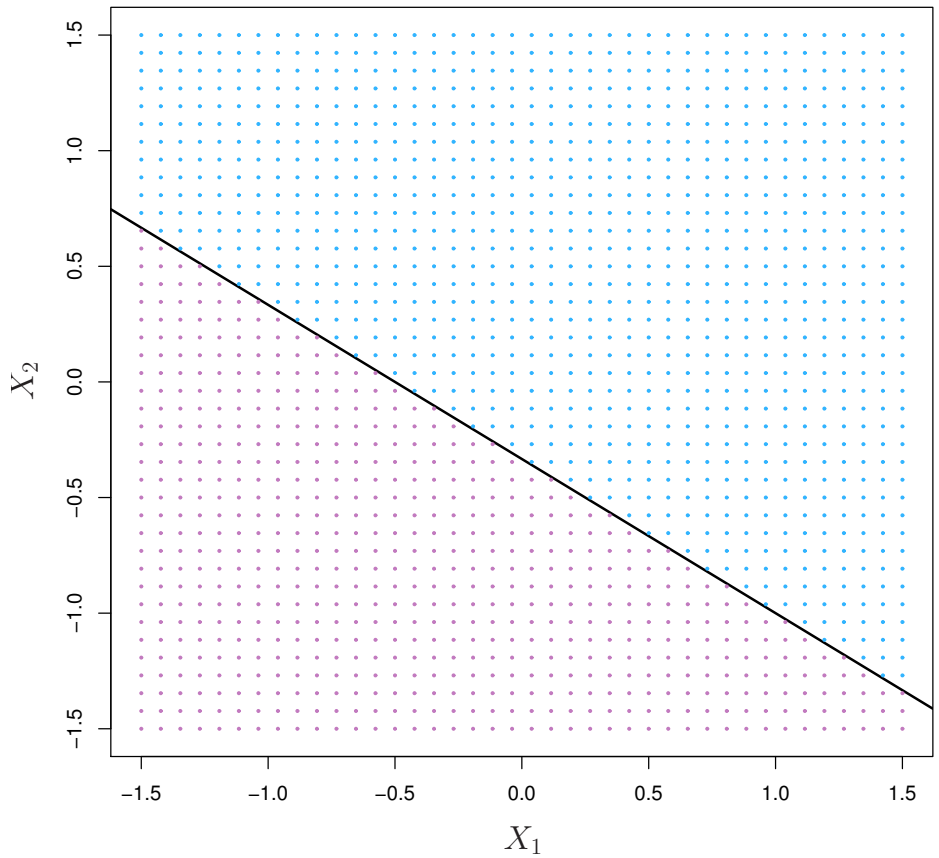


Figure: The hyperplane $1 + 2X_1 + 3X_2 = 0$ is shown. The blue region is the set of points for which $1 + 2X_1 + 3X_2 > 0$, and the purple region the set of points for which $1 + 2X_1 + 3X_2 < 0$ (See ISLR p. 339)

Classification Using a Separating Hyperplane

- Suppose we have a training set consisting on N observations:

$$y_1, \ldots, y_N \in \{-1, 1\} \text{ and } x_1, \ldots, x_N \tag{2}$$

where

$$x_i = (x_{i1}, \ldots, x_{ip})^T \text{ for } i = 1, \ldots, N \tag{3}$$

- **Our goal**: Develop a classifier that will correctly classify a test observation $x^* = (x_1^*, \ldots x_p^*)^T$ based on its feature measurements

- Suppose that it is possible to construct a hyperplane that seperates the training observations according to their class labels

- How many such hyperplanes exist?

- Let x_i be any training observation. A separating hyperplane has the property that

$$\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} > 0 \text{ if } y_i = 1 \tag{4}$$

and

$$\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} < 0 \text{ if } y_i = -1 \tag{5}$$

or, equivalently

$$y_i(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}) > 0 \tag{6}$$

- Thus, conditional on having constructed the separating hyperplane, we can use it as a classifier in a straightforward way:

$$C(x^*) = sng\left(f(x^*)\right), \text{ where } f(x^*) = \beta_0 + \beta_1 x_1^* + \cdots + \beta_p x_p^* \tag{7}$$

- Also the magnitude of $f(x^*)$ is meaningful: The further from zero it is, the more certain we are in our classification

Example of seperating hyperplanes with $p = 2$

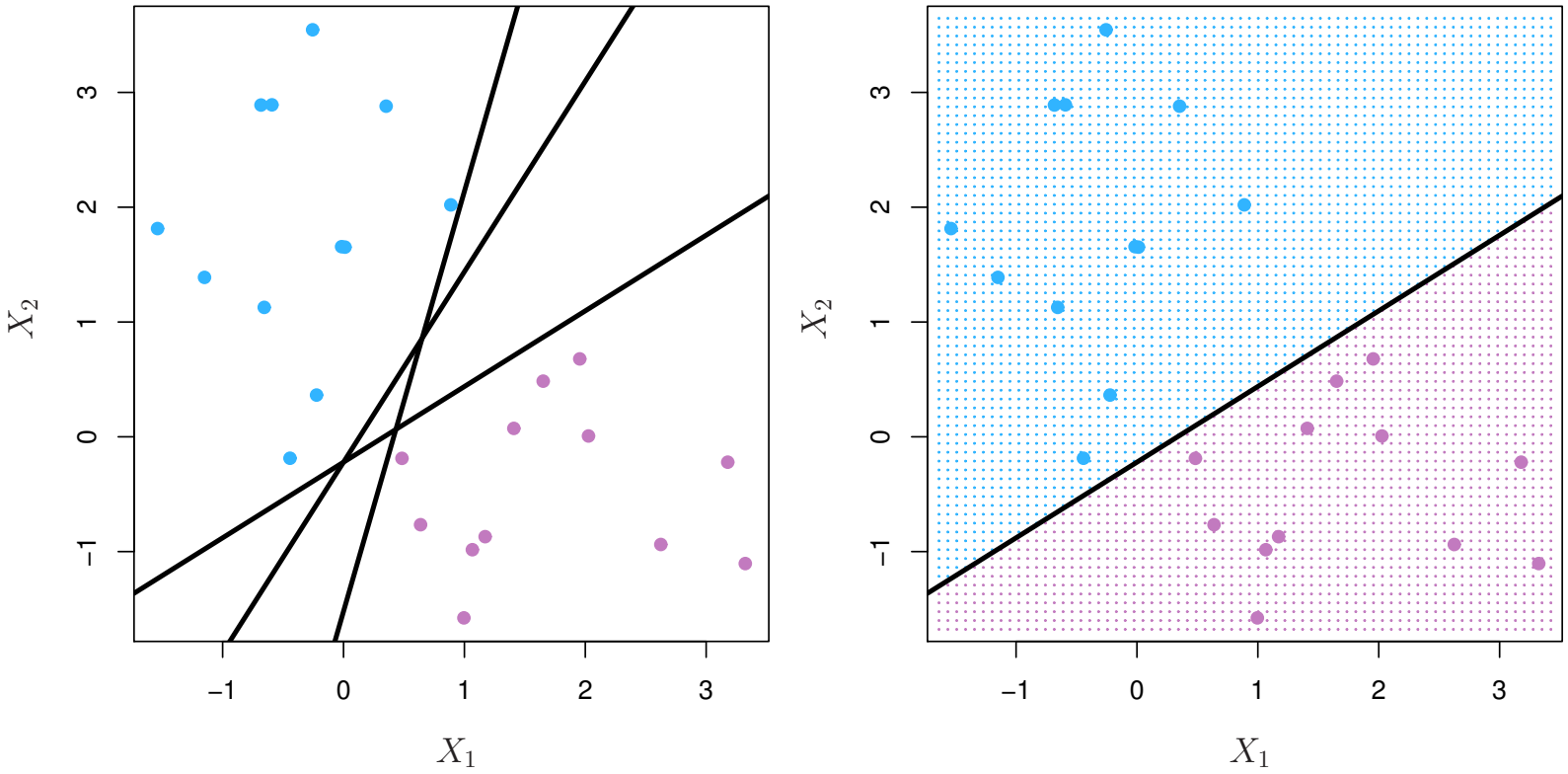


Figure: Left: An illustration that conditional on existence, infinitely many separating hyperplanes exist. Right: An illustration of the separating hyperplane as a decision boundary (See ISLR p. 340)

- If there are infinitely many separating hyperplanes, we must use a concept or metric in order to select one
- We have already argued that the distance between an observation and the hyperplane provides certainty
- Thus, a natural choice is the *maximal margin hyperplane* or the *optimal separating hyperplane*
- Simply described, it is the separating hyperplane that is farthest from the training observations
- We compute the perpendicular distance from each training observation to a given separating hyperplane – the smallest such distance is the minimal distance, i.e. the margin
- The maximal margin hyperplane is the separating hyperplane that has the maximal margin

Example of the maximal margin hyperplane with $p = 2$

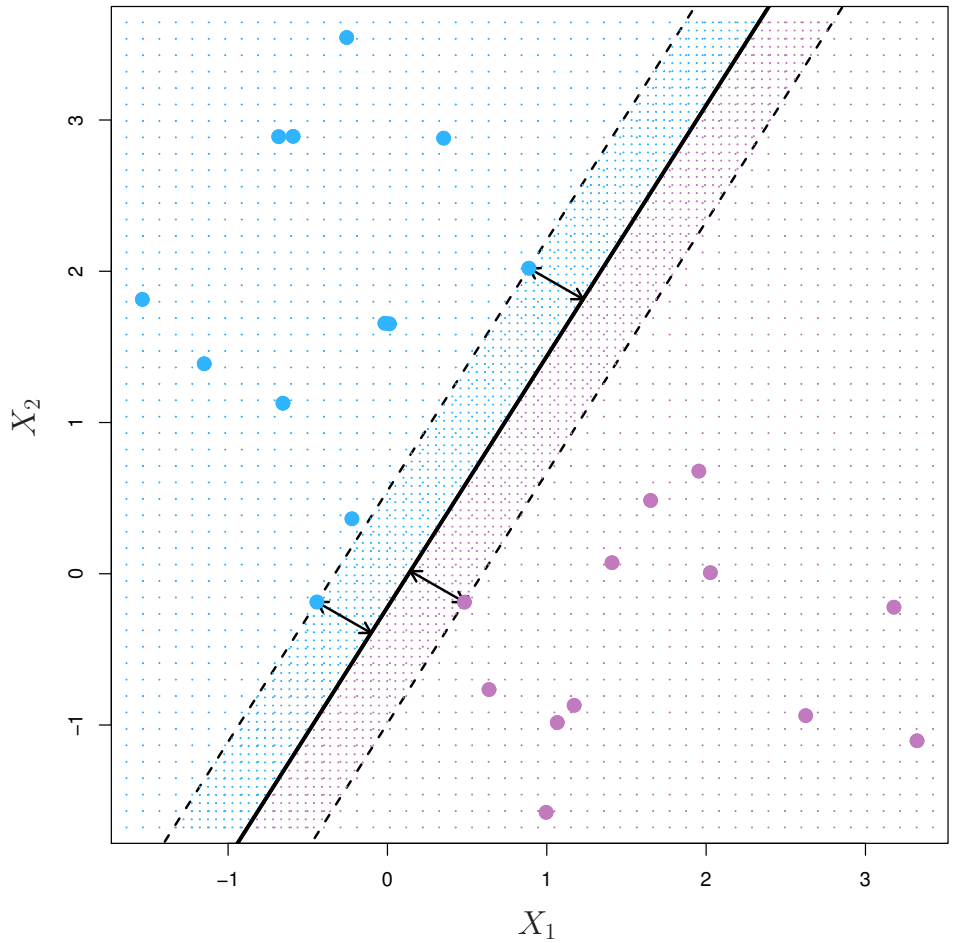


Figure: The maximal margin hyperplane is shown as a solid line. The margin is the distance from the solid line to either of the dashed lines. The two blue points and the purple point that lie on the margin boundary are the support vectors (See ISLR p. 342)

The Support Vectors

- Using the maximal margin hyperplane, the maximal margin classifier follows straightforwardly:

$$C(x^*) = sng\left(f(x^*)\right), \text{ where } f(x^*) = \beta_0 + \beta_1 x_1^* + \cdots + \beta_p x_p^* \quad (8)$$

- The support vectors are the training observations that lie are the margin's outer boundary

- The hyperplane depends directly on the support vectors:
  - If one of the support vectors are moved, then the hyperplane moves as well
  - If any of the other observations are moved, then the hyperplane stays, provided that they do not cross the margin

- Thus, the hyperplane depends directly only on a small subset of the observations

Construction of the Maximal Margin Classifier

- The maximal margin hyperplane is the solution to the optimization problem

$$\max_{\beta_0,\ldots,\beta_p,M} M \tag{9}$$

subject to

$$\sum_{j=1}^{p} \beta_j^2 = 1 \tag{10}$$

and

$$y_i(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}) \geq M, \forall i = 1, \ldots, N \tag{11}$$

- Hence, M represents the margin of our hyperplane, and the optimization problem chooses $\beta_0, \ldots, \beta_p$ to maximize M

- Thus, it follows the exact definition of the maximal margin hyperplane

Linear Support Vector Classifier (SVC)

- We have seen that the maximal margin hyperplane provides a natural way of performing classification – but only if a separating hyperplane exists
- This will often not be the case, so that the preceding optimization problem do not have a solution for $M \geq 0$
- Even if we can perfectly separate observations, this might not be beneficial because we might be overfitting the training data
- The Linear SVC is a generalization of the maximal margin classifier using a *soft margin*
- Basically, we create hyperplanes which *almost* separates the classes

Example of sensitive maximal margin hyperplanes

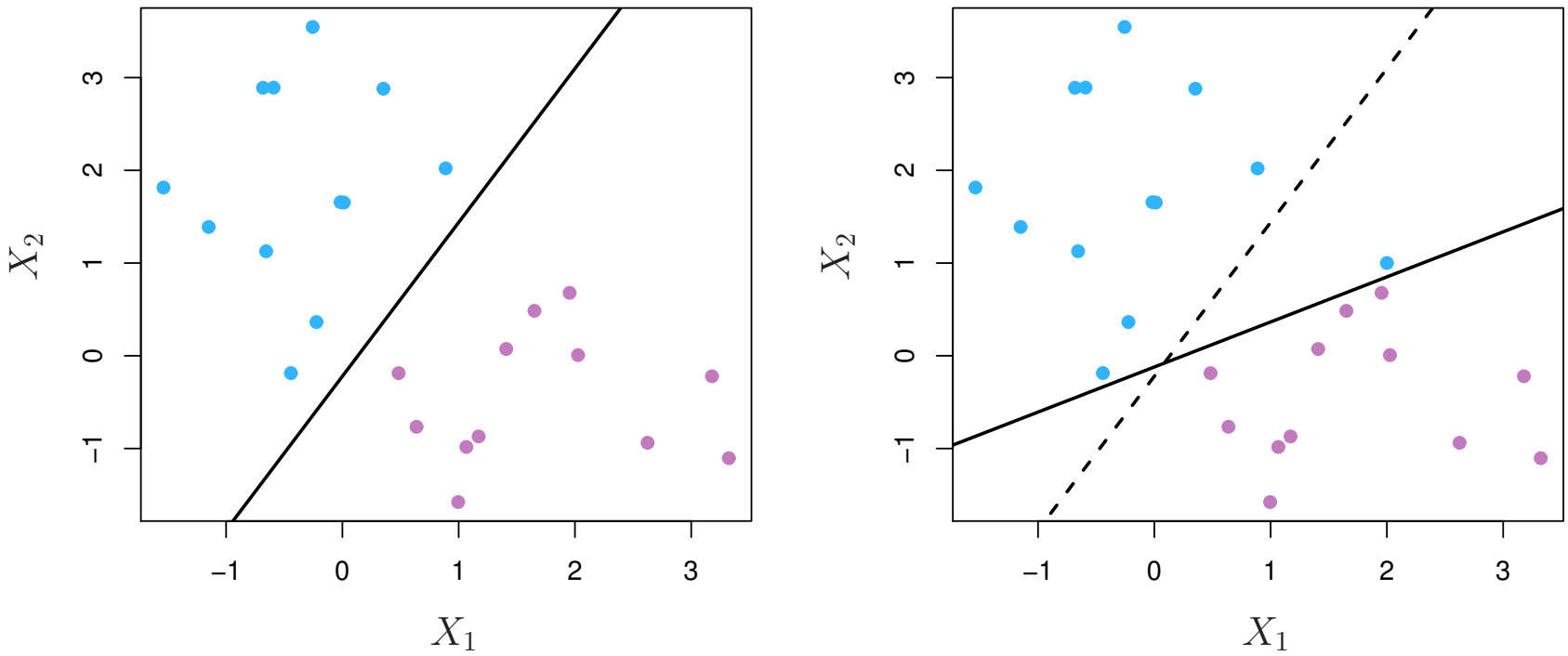


Figure: Left: the maximal margin hyperplane for a given training set. Right: The maximal margin hyperplane by adding one blue observation (See ISLR p. 345)

- The soft-margin hyperplane is the solution to the optimization problem

$$\max_{\beta_0,\ldots,\beta_p,\epsilon_1,\ldots,\epsilon_N,M} M \tag{12}$$

subject to

$$\sum_{j=1}^{p} \beta_j^2 = 1 \tag{13}$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}) \geq M(1 - \epsilon_i), \forall i = 1, \ldots, N \tag{14}$$

and

$$\epsilon_i \geq 0, \quad \sum_{i=1}^{N} \epsilon_i \leq C \tag{15}$$

where C is a nonnegative hyperparameter, and $\epsilon_1, \ldots, \epsilon_N$ slack variables allowing individual observations to be on the wrong side of the margin or hyperplane

- Note: if $\epsilon_i = 0$, then the ith observation is on the correct side of the margin
- Note: No more than C observations can be on the wrong side of the hyperplane

- Classification then follows as in the previous case
- In practice, C is a hyperparameter that is generally chosen via CV
- C controls the bias-variance trade-off:
  - **Small** C: we seek narrow margins that are rarely violated $\rightarrow$ we highly fit the training data $\rightarrow$ we may have less bias and more variance
  - **Large** C: wider margin, and more violations $\rightarrow$ we fit the trainig data less $\rightarrow$ more bias and less variance
- Only observations that lie on the margin or violate it determined the hyperplane
  - Thus, these observations are the *support vectors*
- When C is large, more observations will violate the margin, and thus we will have more support vectors that determine the hyperplane

The role of C in the soft-margin hyperplane

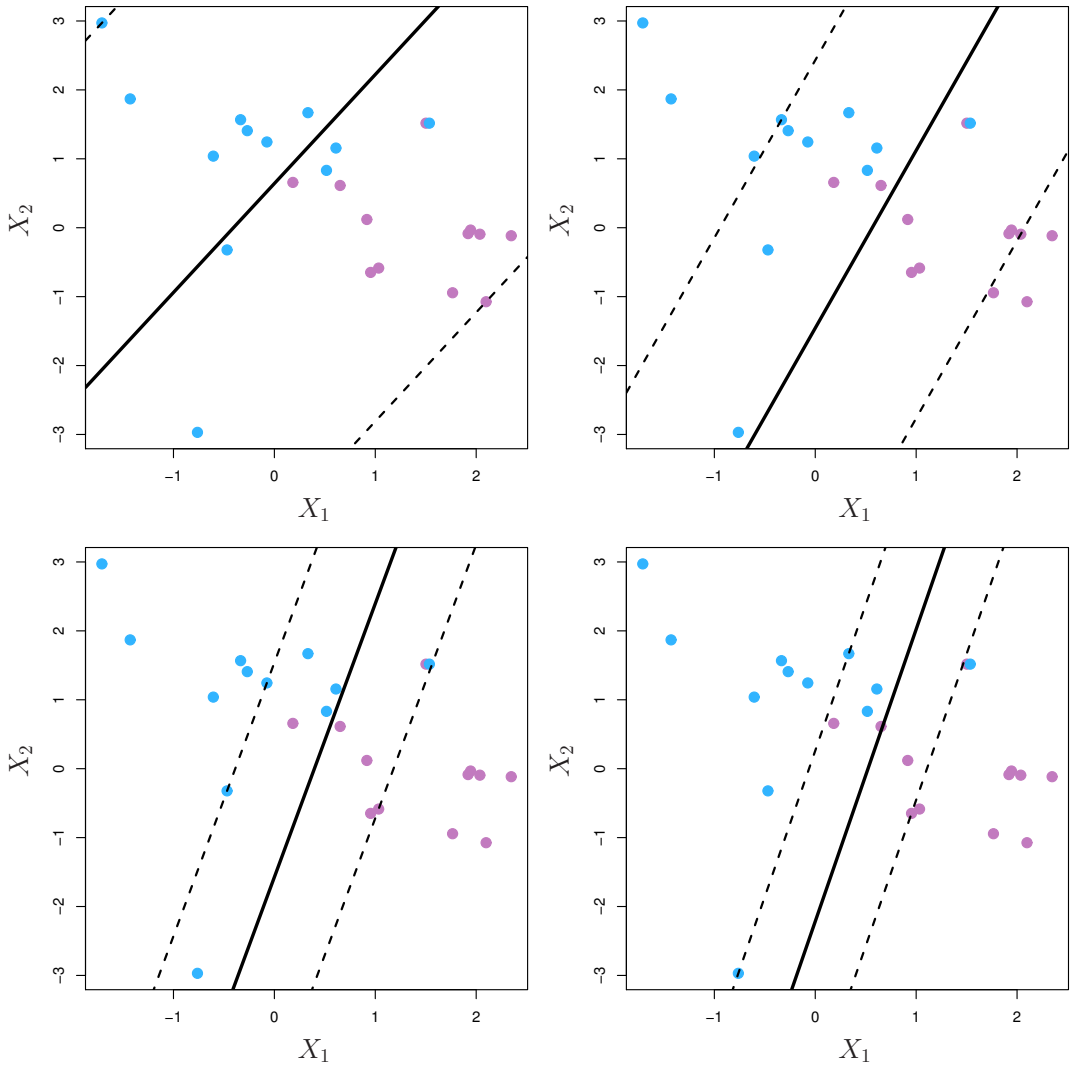


Figure: The effect of sequentially decreasing C (See ISLR p. 348)

A different, but equivalent representation of linear SVC

- One can rewrite (12) – (15) for fitting the support vector classifier as

$$\min_{\beta_0,\ldots,\beta_p} \left\{ \sum_{i=1}^{N} \max\{0, 1 - y_i f(x_i)\} + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} \tag{16}$$

- A small λ amounts to a small C (Ridge penalty)
- The first term is known as the hinge loss function
  - In this formulation, the margin corresponds to the value 1
  - Thus, an observation on the correct side of the margin will not have a loss
- With this formulation, one can establish a familiarity between logistic regression and support vector machines
  - SVCs tend to perform better with well separable classes

General Support Vector Classifier (SVC)

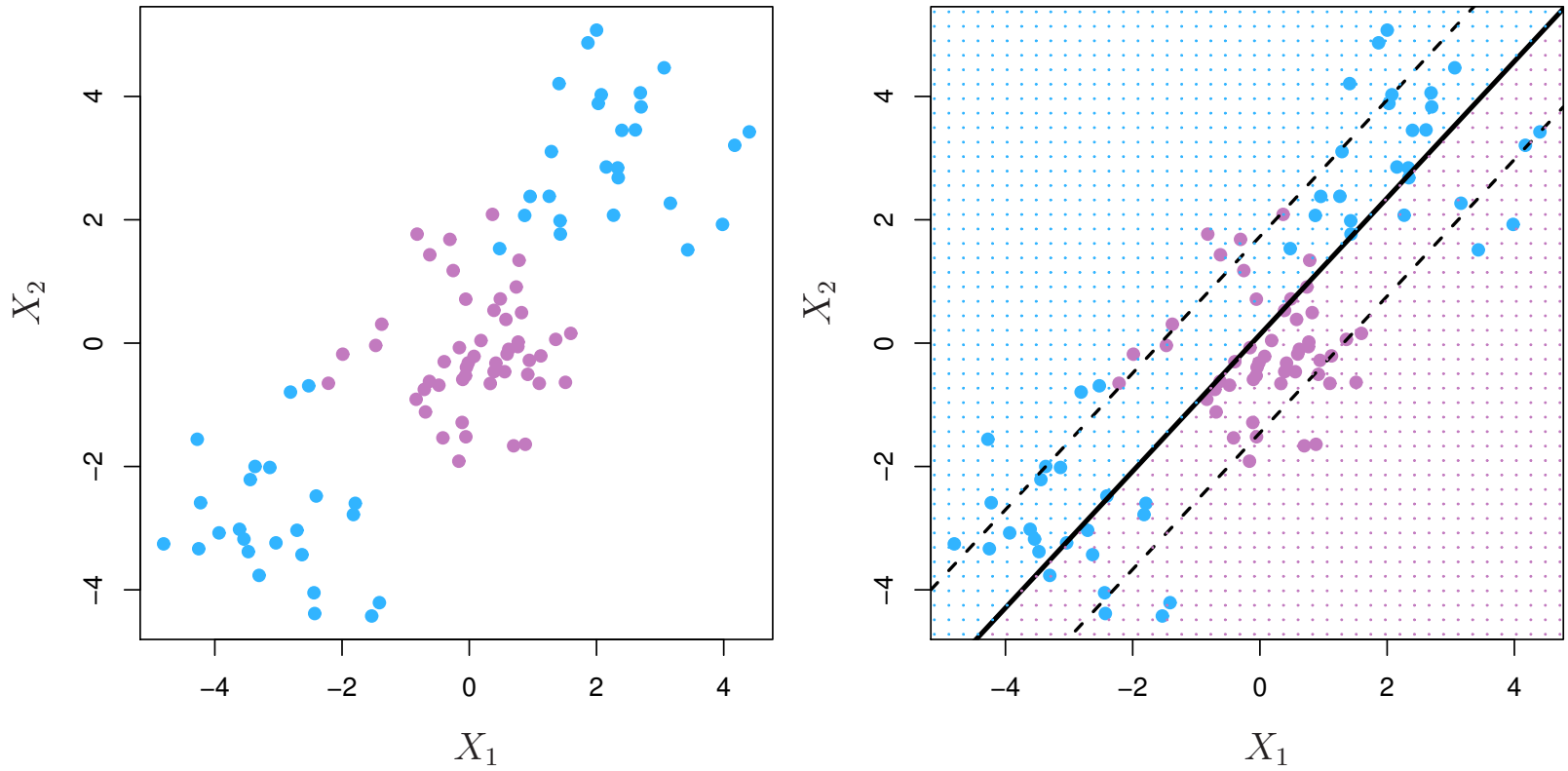


Figure: The observations fall into two classes, with a non-linear boundary between them. The SVC seeks a linear boundary (See ISLR p. 349)

- We could address the case of possible non-linearity by enlarging the feature space
  - e.g. by adding second and third degree polynomials
- Then the hyperplane would be linear in the transformed feature space, but most likely non-linear in the original
  - It would simply be the solution to (12) – (15) for the transformed feature space
- Alternatively, we could use other functions to transform our feature space
- Hence, we may end up with a huge feature space with computational infeasibility as the result
- The SVC is a generalization of the linear SVC, that enlarge the feature space in an efficient way by using kernels

- The solution to the support vector classifier problem (12) — (15) involves only the inner products of the observations (as opposed to the observations themselves)
- The inner product of two observations $x_i, x_{i'}$ is given by

$$\langle x_i, x_{i'} \rangle = \sum_{j=1}^{p} x_{ij} x_{i'j} \tag{17}$$

- The support vector classifier can then be represented as

$$f(x) = \beta_0 + \sum_{i=1}^{N} \alpha_i \langle x, x_i \rangle \tag{18}$$

- We estimate β_0 and $\alpha_1, \ldots, \alpha_N$ by using the $\binom{N}{2}$ pairwise inner products of the training observations
- α_i is non-zero only if the ith observation is a support vector. Thus, to classify x, we only need to evaluate

$$f(x) = \beta_0 + \sum_{i \in \mathcal{S}} \alpha_i \langle x, x_i \rangle \tag{19}$$

- Suppose we replace the inner product in (17) with a kernel function $K(x_i, x_{i'})$
- The support vector classifier then becomes

$$f(x) = \beta_0 + \sum_{i=1}^{N} \alpha_i K(x, x_i) \tag{20}$$

- If we set K equal to the inner product, we will return to (17), which is the linear SVC – It has a linear kernel
- Alternatively, to accommodate non-linearity, we could use a polynomial kernel of degree d

$$K(x_i, x_{i'}) = \left(1 + \sum_{j=1}^{p} x_{ij} x_{i'j}\right)^d \tag{21}$$

- Another popular choice is the Gaussian basis radial kernel, which takes the form

$$K(x_i, x_{i'}) = exp\left(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2\right) \tag{22}$$

- The radial kernel has very local behavior, in the sense that only nearby training observations have an effect on the class label of a test observation

Examples of non-linear kernels

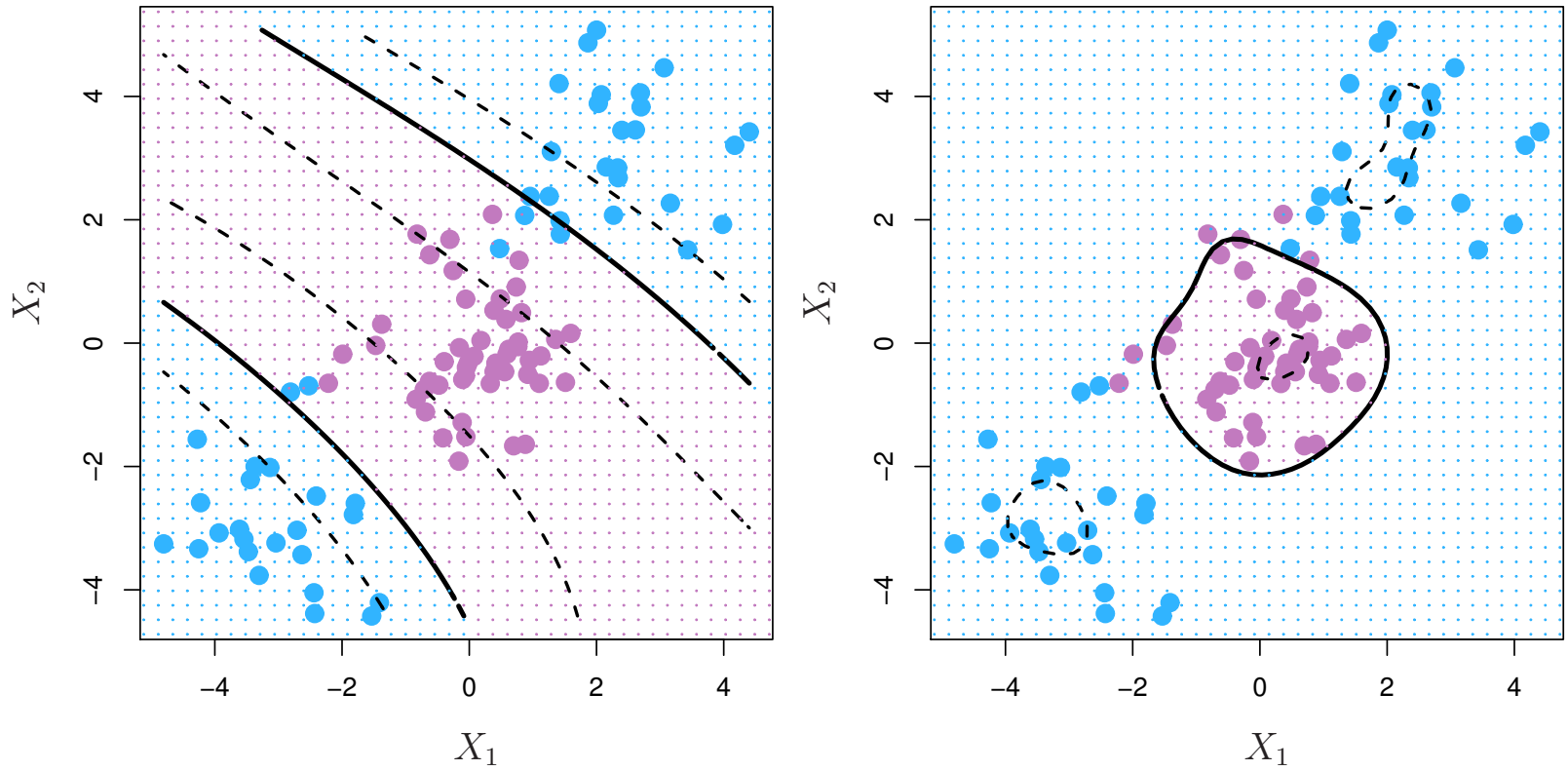


Figure: Left: An SVM with a polynomial kernel of degree 3. Right: An SVM with a radial kernel is applied (See ISLR p. 353)

Advantages of using non-linear kernels

- One advantage is computational:
  - When using kernels, one need only compute $K(x_i, x_{i'})$ for all $\binom{N}{2}$ pairs
  - This can be done without explicitly working in the enlarged feature space

- Another advantage is that we can work with implicit transformations of the feature space that we would never be able to explicitly create
  - For example, it can be shown that using the Gaussian radial basis function, the feature space is infinite-dimensional

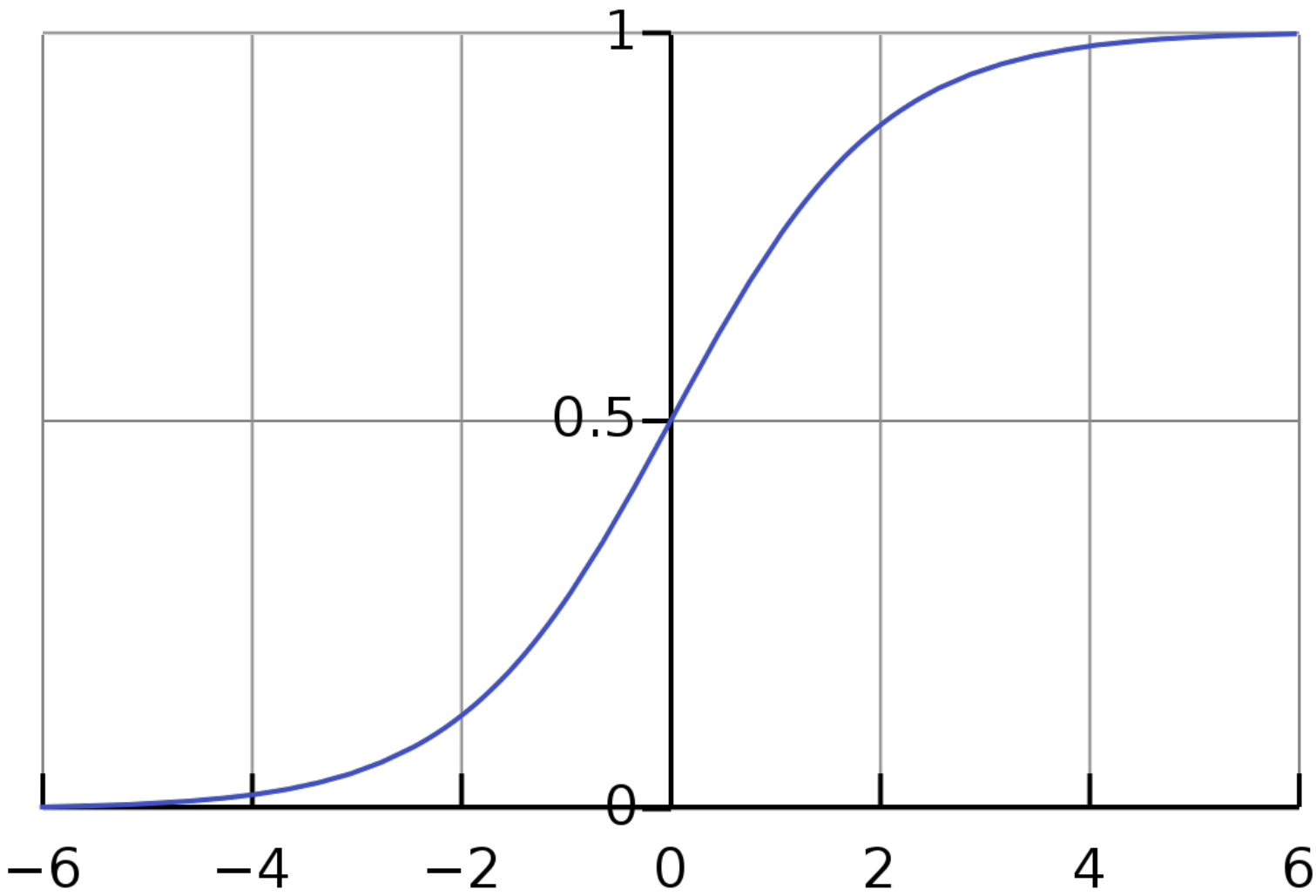
Probabilities with SVC

- Note that nothing in the construction of SVC was an estimate of a probability
- Furthermore, the output that we get for a certain test observation tells us on which side of the hyperplane it lies, and how far away it is
- We previously argued that a higher distance comes with higher certainty in our classification, but since the output is unbounded, there is no naturally probabilistic interpretation
- A method to get probabilities is the so-called Platt scaling:
  - Roughly speaking, we use the properties of the logistic/sigmoid function:

$$P(y = 1|x) = \frac{1}{1 + exp(Af(x) + B)} \tag{23}$$

    Where A, B are estimated using maximum likelihood

Probabilities with SVC



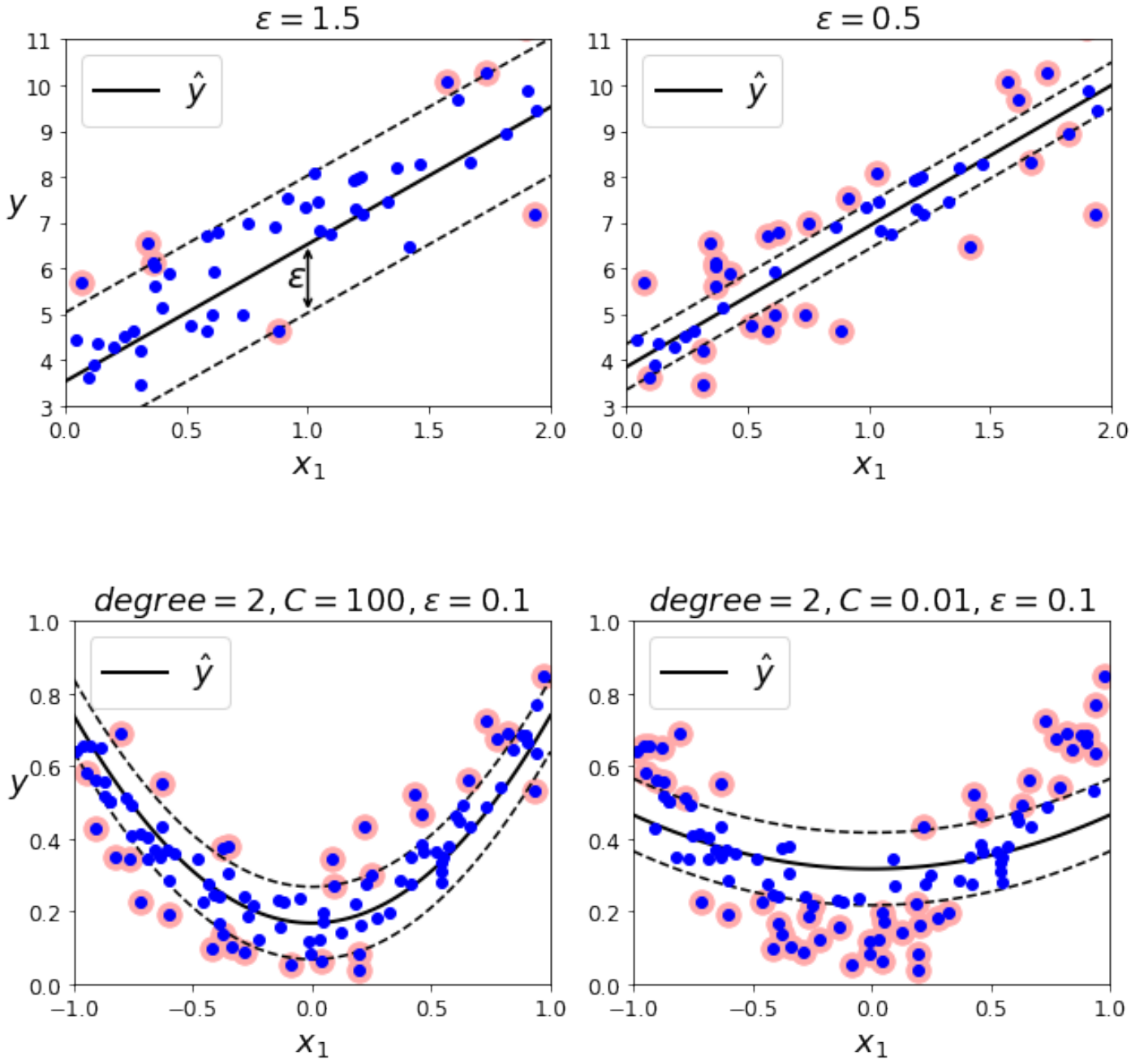
SVCs with more than two classes

- The concept of separating hyperplanes upon which SVCs are based does not lend itself naturally to more than two classes
- A number of proposals for extending SVMs to the K-class case have been made, the two most popular are
  - *one-vs.-one*
  - *one-vs.-all*
- *one-vs.-one*: Constructing $\binom{K}{2}$ SVCs, each of which compares a pair of classes
  - we classify a test observation using each of the $\binom{K}{2}$ classifiers
  - an observation is then classified to the class for which it got the most "votes" (i.e. hard-voting)
  - alternatively, we may use soft-voting if using platt-scaling, but this may be computationally expensive

- *one-vs.-all*: We fit K SVCs, each time comparing one of all the K classes to the remaining $K-1$ classes
- We then get K "confidence" scores $f(x)$, each informing us how certain it is that x belong to a class
- Then, we simply pick the class with the highest score
- This is somewhat similar to other classifiers, in which we get the class probabilities of an observation, and classify it to the class with the highest one

Support Vector Regression

- Even though SVMs were developed for classification, the approach has been extended to the case of regression
- The details are somewhat complex and beyond the scope of this course, but the idea is as follows:
- We reverse the objective:
  - We no longer try to fit the largest area between two classes, while limiting margin violations
  - instead, we try to fit as many observations within the margin, while limiting margin violations
- The width of the margin is controlled by the hyperparameter ϵ
  - The larger the ϵ, the larger the margin


$\varepsilon = 1.5$
$\varepsilon = 0.5$
$degree = 2, C = 100, \varepsilon = 0.1$
$degree = 2, C = 0.01, \varepsilon = 0.1$
$\hat{y}$
ε
y
x_1

References

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112). **Chapter 9**

Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems. **Chapter 5**