Applied Machine Learning in Health Sciences 2023
-
Support vector machines

Peter Mondrup Rasmussen

CFIN

pmr@cfin.au.dk

Support vector machines

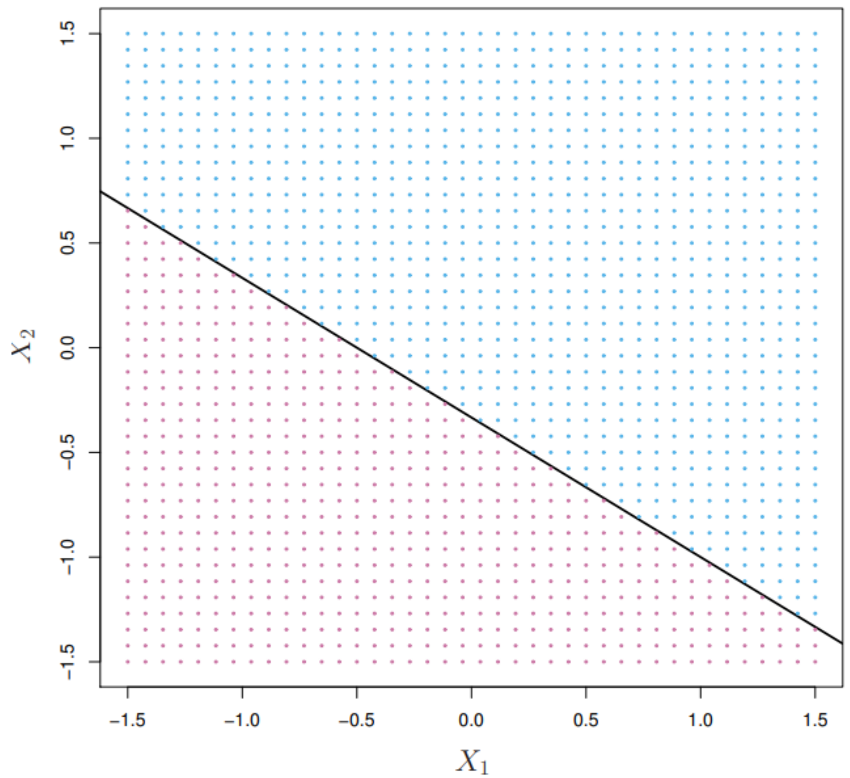
Support vector machines

- Support vector machines (SVMs) are supervised learning models for classification and regression but most often used in classification.
- The ILS distinguishes between *maximal margin classifier, support vector classifier,* and *support vector machine*. In the literature you will often find these under the common term *support vector machines.*

Support vector machines - hyperplanes

- In a p-dimensional space, a *hyperplane* is a $p-1$ flat affine subspace, e.g. in three dimensions a hyperplane will be a plane, and in two dimensions a hyperplane will be a line.
- In p dimensions a hyperplane is defined by
  $$\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p = 0$$
- If a point X lies on one side of the hyperplane
  $$\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p > 0$$
- If the point X lies on the other side
  $$\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p < 0$$

- Example: $1 + 2X_1 + 3X_2 = 0$



Maximal margin classifier

Hard-margin support vector machine

Support vector machines - hyperplanes

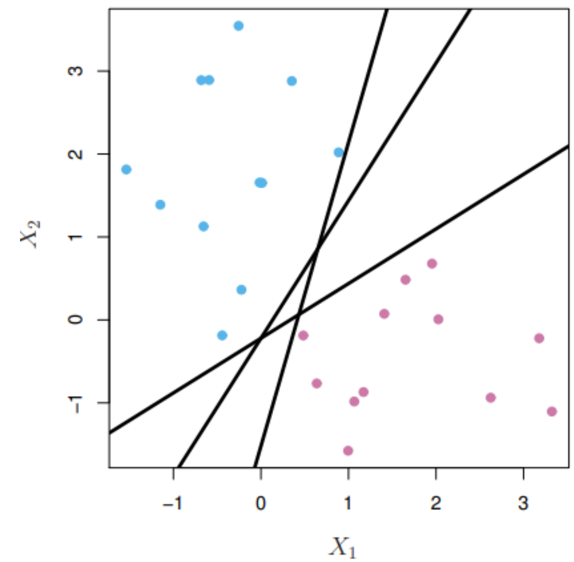
- If we have observations x_i from two classes, if we code the class labels/outputs y_i by -1 and 1, and if the classes are separable, then a separating hyperplane has the property

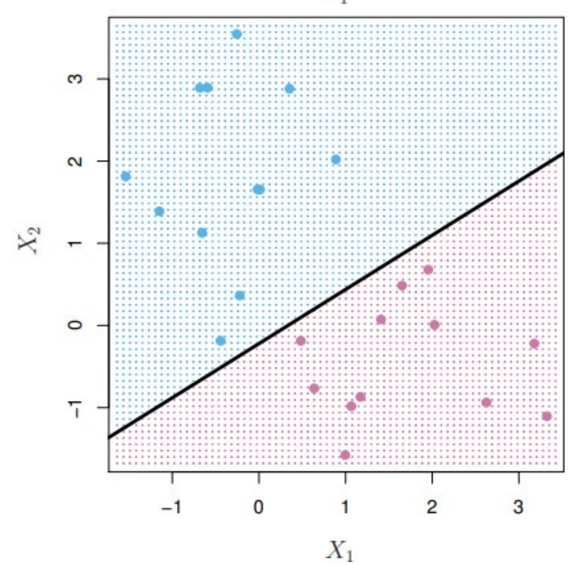
$$y_i(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}) > 0$$

- If we have a test observation x^* we could classify it according to the sign of

$$f(x^*) = \beta_0 + \beta_1 x_1^* + \cdots + \beta_p x_p^*$$

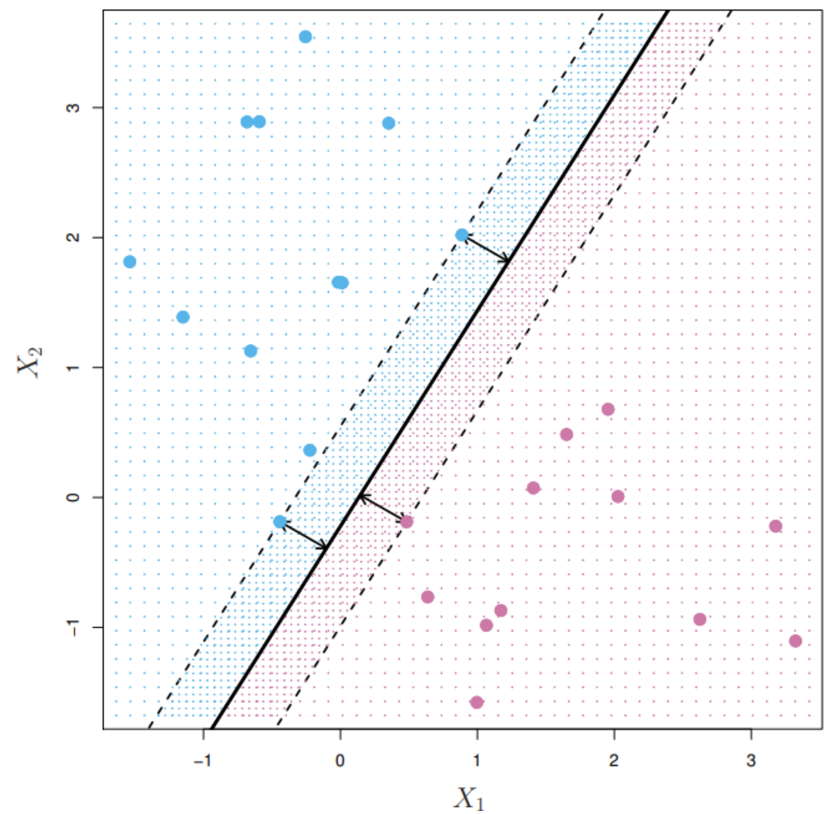
- If our observations are separable there exist an infinite number of such hyperplanes. How to chose a good one?



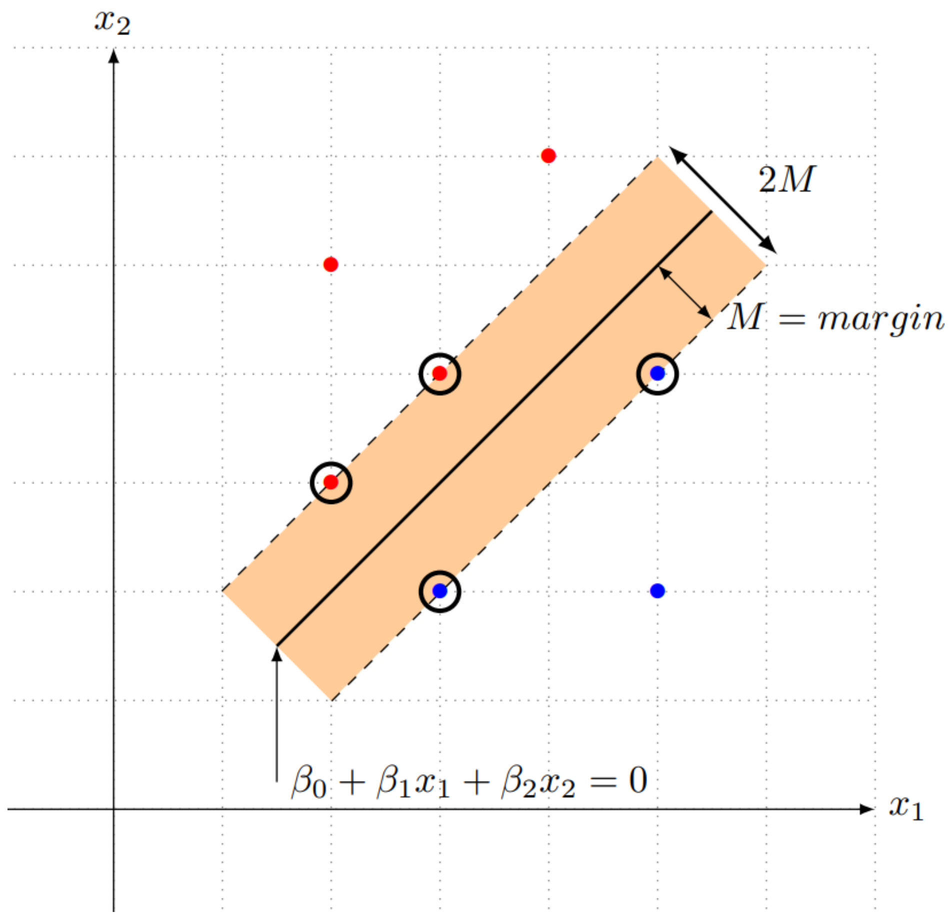


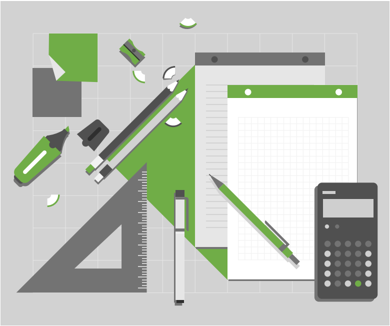
Maximal margin classifier (hard-margin SVM)

- The *margin* is defined as the closest distance from points in either class to the separating hyperplane.
- Finds the hyperplane for which the margin is maximized.
- Parallel lines/planes to the separating hyperplane are called *canonical hyperplanes.* These are located at a distance M to the separating hyperplane.
- Points that lies one the canonical hyperplanes are called *support vectors.*
- Only support vectors has an influence on the location of the separating hyperplane.



Maximal margin classifier (hard-margin SVM)





In-class exercises

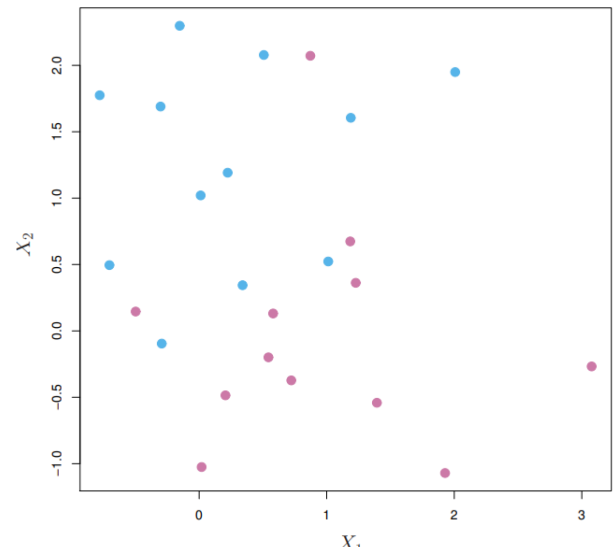
7 Support vector machines
Checkpoint 16

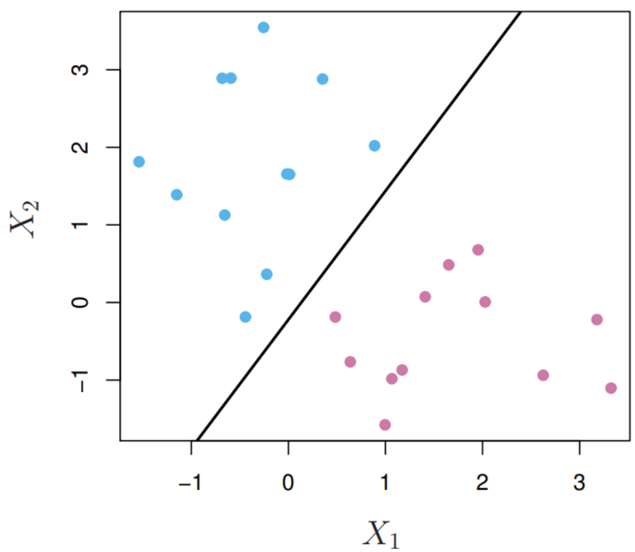
Support vector classifier

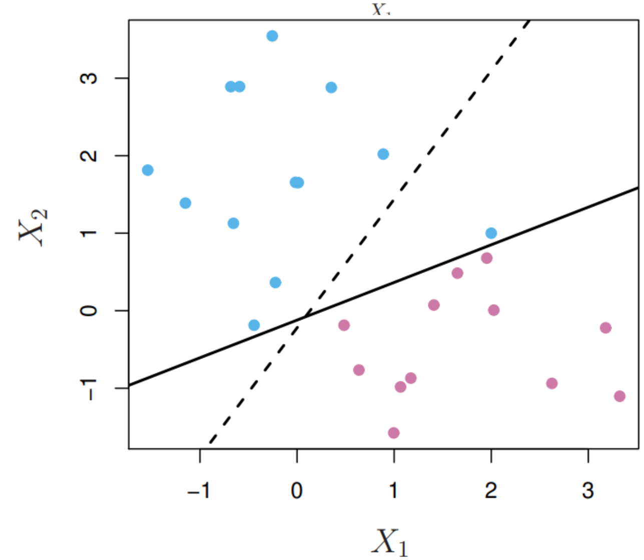
Soft-margin support vector machine

Support vector classifier (soft margin SVM)

- Challenges with maximal margin classifier:
  - Non-separable data.
  - Highly sensitive to specific observation -> risk of overfitting.

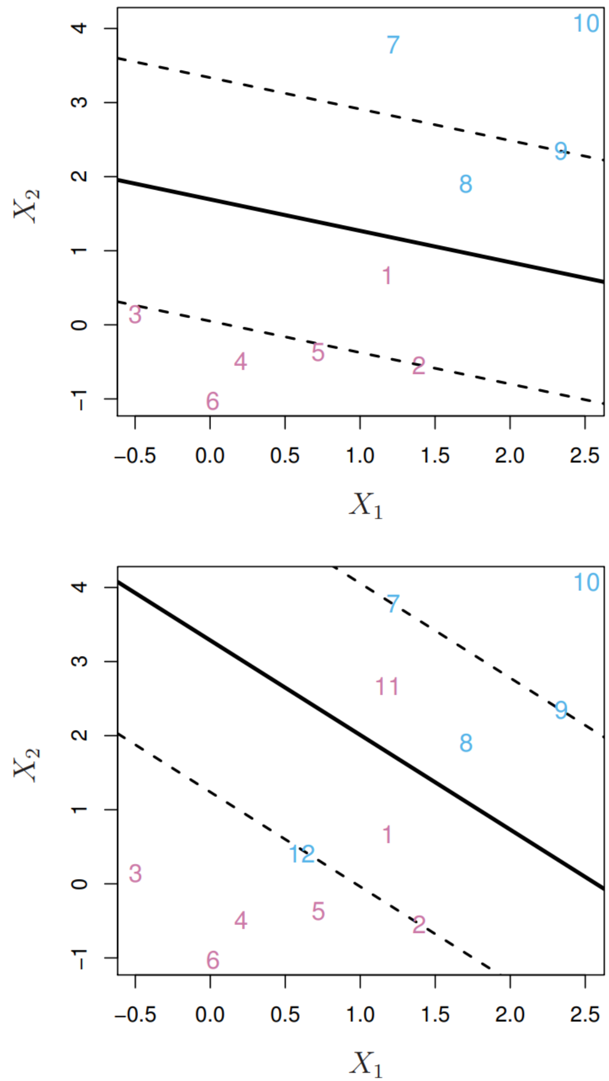




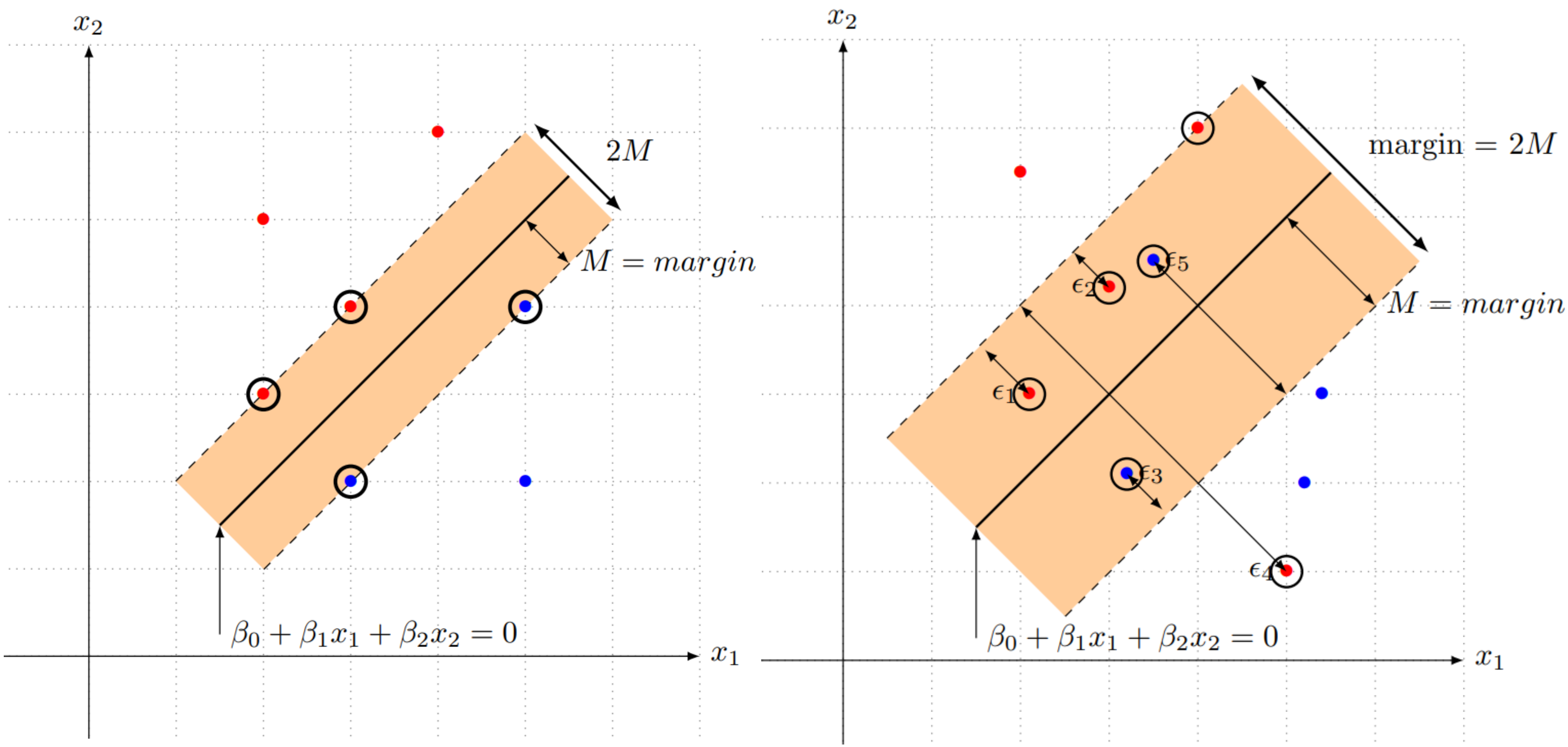


Support vector classifier (soft margin SVM)

- Relaxes constraints and allows observations to be
    - On the wrong side of the canonical hyperplanes (within the margin)
    - On the wrong side of the separating hyperplanes, i.e. mis-classified training observations.
- Goal is to have greater robustness to individual observations and better classification of *most* training observations.
- This model has a hyper parameter (regularization parameter) C that control the extend to which violations of the margin is allowed.
- Observations on the canonical hyperplane, within the margin, or on the wrong side of the separating hyperplane are *support vectors*.



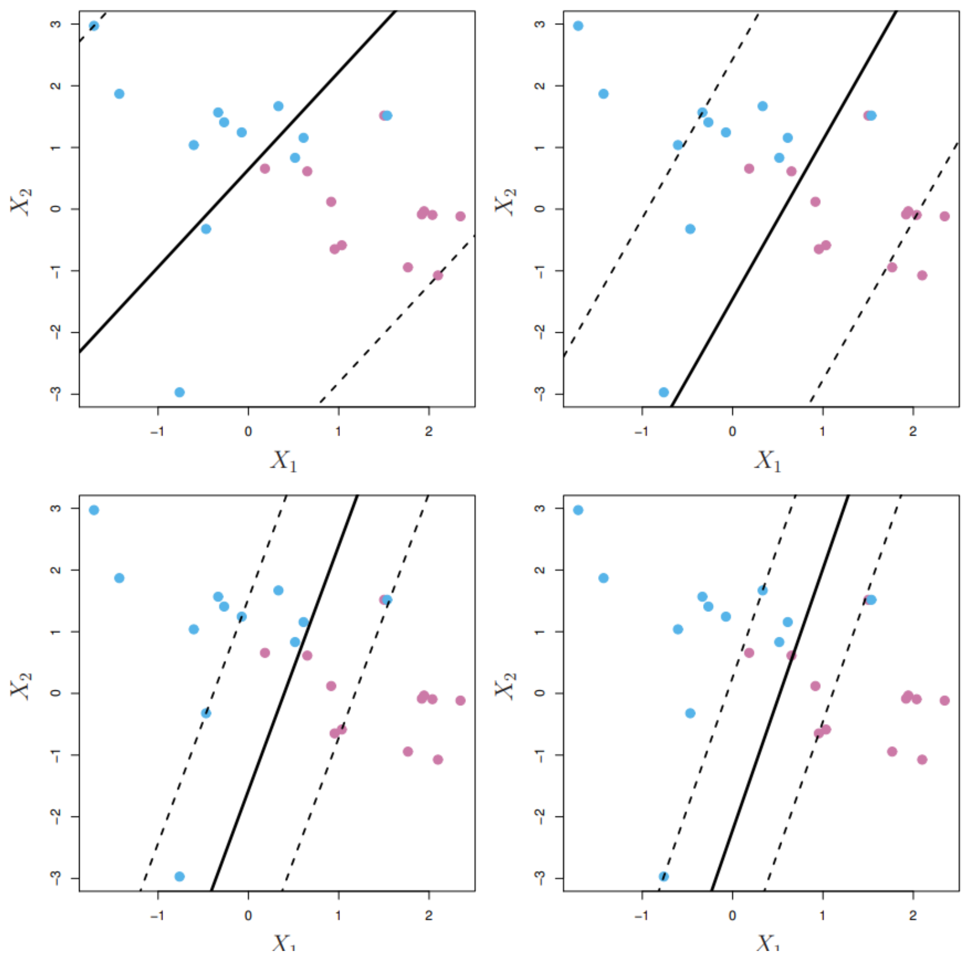
Support vector classifier (soft margin SVM)



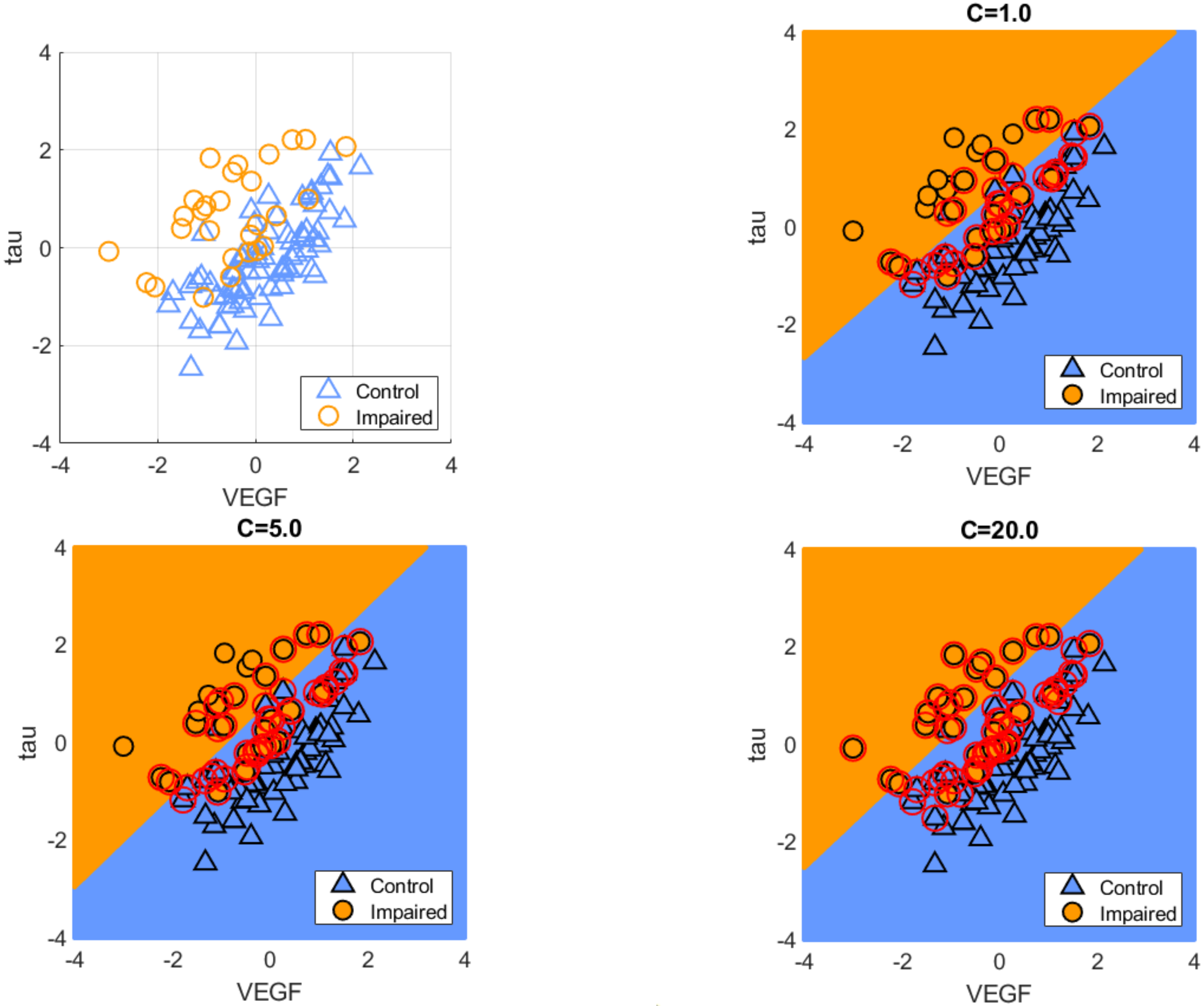
The ϵ's are *slack variables* that allows individual points to be within the margin or on the wrong side of the hyperplane. The distance between the support vector and the corresponding hyperplane is $M\epsilon_i$.
The C parameter places a budget on total amount of such *violations* by $\sum \epsilon_i \leq C$.

Support vector classifier (soft margin SVM)

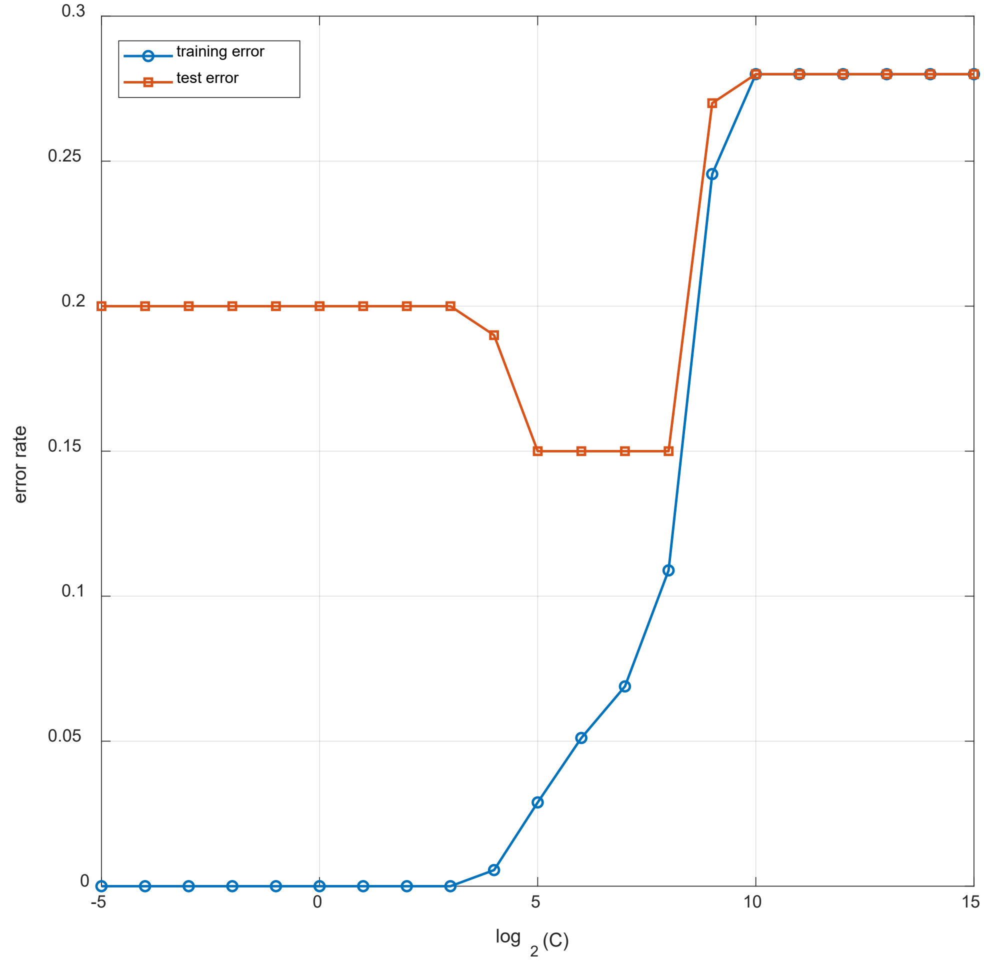
- The regularization parameter C provides means for controlling the Bias-Variance Trade-Off. High: top left, low: bottom right.



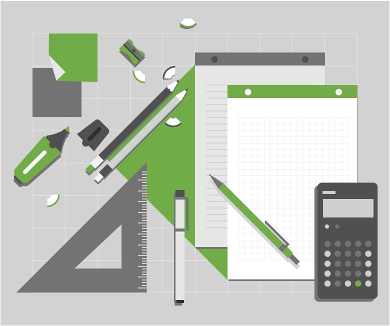
Support vector classifier (soft margin SVM)



Support vector classifier (soft margin SVM)



Using cross-validation to select the regularization parameter C



In-class exercises

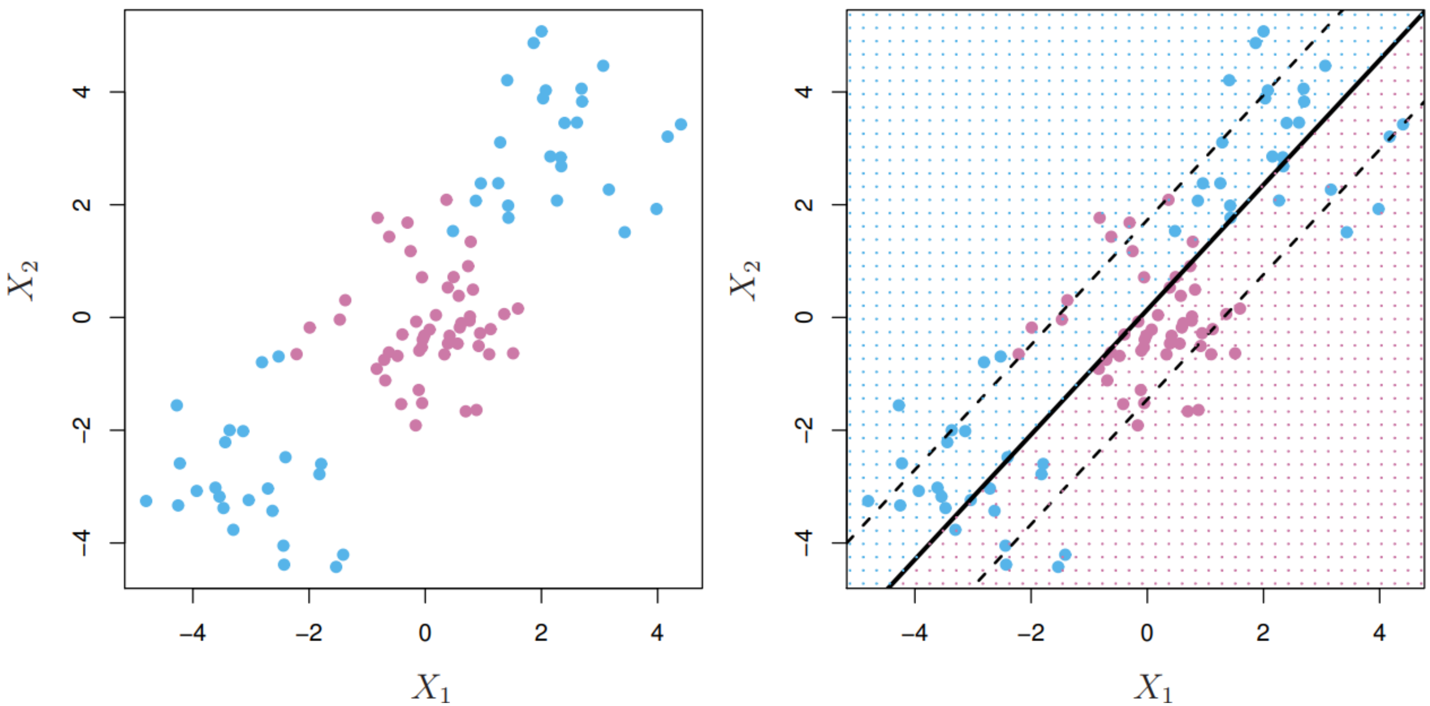
7 Support vector machines
Checkpoints 17, 18

Support vector machine

Non-linear support vector machine

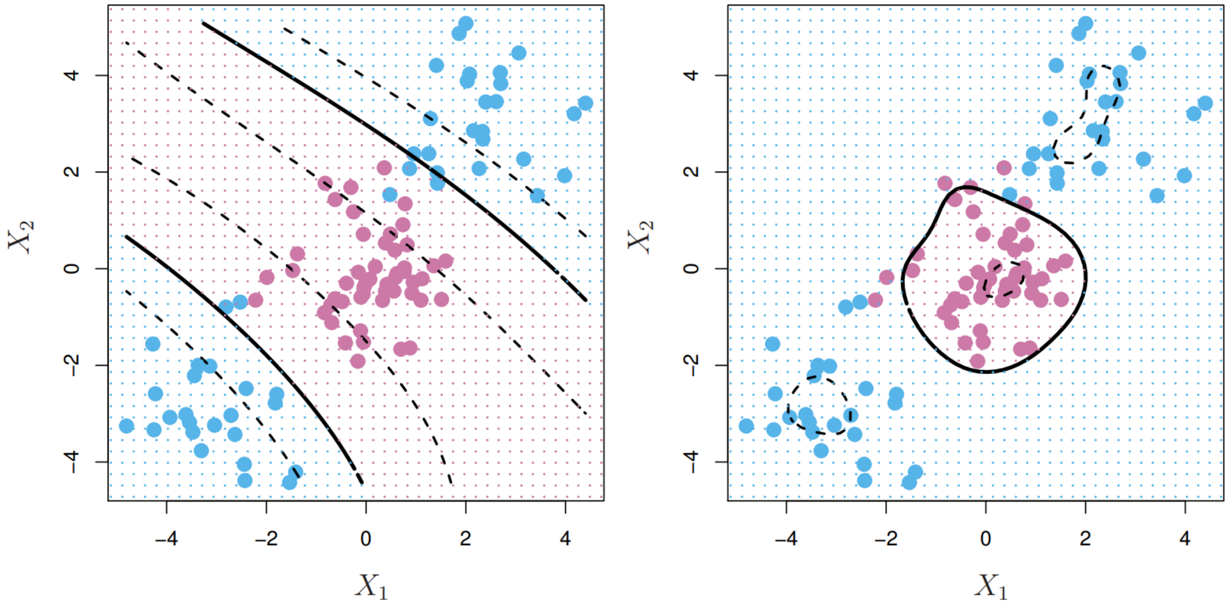
Support vector machines (non-linear SVM)

- Limitation so far: Produces a linear decision boundary



Support vector machines (non-linear SVM)

- Main idea: Enlarge/expand the original feature space to higher dimension and fit a linear model in the expanded feature space. This will produce a non-linear decision boundary in the original feature space.
- Example: Use five features $(X_1, X_2, X_1^2, X_2^2, X_1X_2)$ instead of the original two features and fit a support vector classifier based on these five features.



Support vector machines (non-linear SVM)

- The decision function of the maximal margin classifier and the support vector classifier is

$$f(x^*) = \beta_0 + \beta_1 x_1^* + \cdots + \beta_p x_p^* = \beta_0 + \langle \beta_{\setminus\beta_0}, x^* \rangle$$

where $\beta_{\setminus\beta_0}$ denotes a vector containing all coefficients except β_0.

- It turns out that i) only inner products between observations are needed in computing the coefficients in the SVM, and ii) that the support vector classifier can be represented by

$$f(x^*) = \beta_0 + \sum_{i=1}^{n} \alpha_i \langle x_i x^* \rangle$$

where there are n coefficients α - one for each training observation.

- Only *support vectors* have a non-zero α coefficient, hence

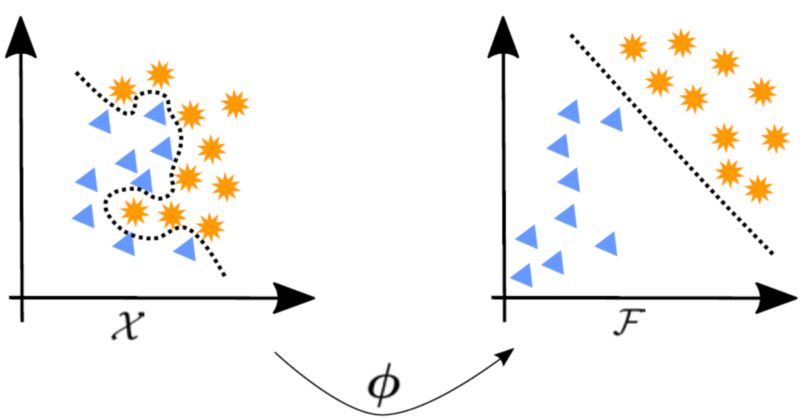
$$f(x^*) = \beta_0 + \sum_{i \in S} \alpha_i \langle x_i x^* \rangle$$

where S is the set of support vectors.

Support vector machines (non-linear SVM)

- A kernel function $K(x_i, x_{i'})$ is a *generalization* of the inner product. The kernel function returns the inner product in some feature space.
- Using an SVM with a kernel amounts to implicitly mapping the original data to a higher-dimensional space and then a support vector classifier in the transformed feature space.
- In practice, a kernel function is used instead of performing the actual mapping.
- The decision function of a support vector machine is

$$f(x) = \beta_0 + \sum_{i=S} \alpha_i K(x_i, x)$$



Support vector machines (non-linear SVM)

- The linear kernel

$$K(x_i, x_{i'}) = \sum_{j=1}^{p} x_{ij} x_{i'j}$$

corresponds to inner products in the original feature space.

- The polynomial kernel

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

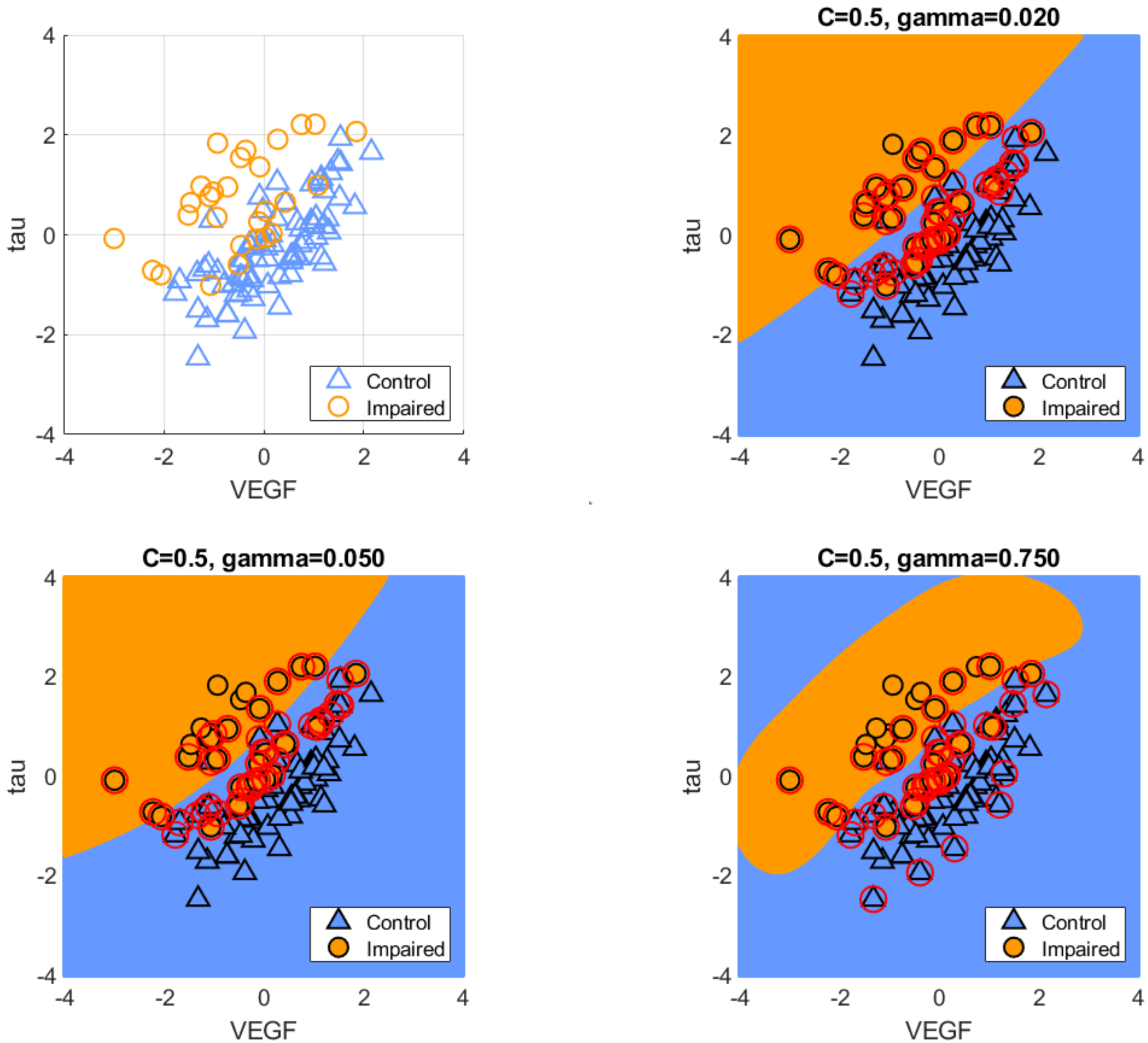
corresponds to inner products in a higher dimensional feature space involving polynomials of degree d.

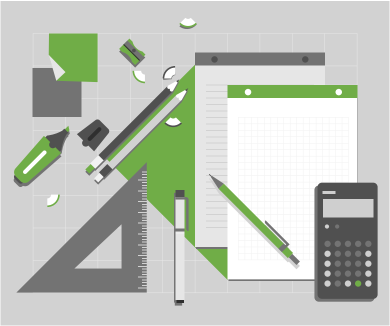
- Another popular kernel is the *radial kernel*

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

where γ is a positive hyper-parameter.

Support vector machine (non-linear SVM)





In-class exercises

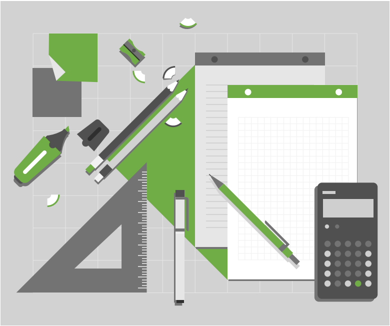
7 Support vector machines
Checkpoint 19

Support vector machines

Practical issues

Support vector machines – practical issues

- The SVM has a regularization parameter C that needs to be selected. This is typically done be cross-validation.
- **Important note**: The definition of the C parameter in the ISL book is inversely related to how the parameter is often defined in other text books and in e.g. Matlab and R. That is, a large C value in the book corresponds to a low C value in Matlab and R, and vice verse.
- Kernels allows for increased model flexibility and non-linear decision boundaries. Select the kernel hyper-parameters by cross-validation.
- (Kernels can also be used together with e.g. logistic- or linear regression to make these models more flexible).
- Originally developed for binary classification tasks, but there are techniques to use SVM with $K > 2$ classes.
- The SVM does not directly provide predicted class probabilities (as in e.g. logistic regression and neural networks) - however there are ways of estimating these also for the SVM.
- Often shows good generalization performance.



In-class exercises

7 Support vector machines
Checkpoint 20

References

Figures from James et al. *An Introduction to Statistical Learning*, second edition, https://www.statlearning.com/resources-second-edition