SUPPORT VECTOR MACHINES

Outline

- Intuition behind support vector machines
- Linear classifiers
- Feature expansion to accommodate non-linear decision boundaries
- Polynomial and Radial Basis Function kernels
- SVM vs. logistic regression
- Implementation of SVM

Support Vector Machines

Support Vector Machines (SVMs) are powerful and versatile supervised learning models used primarily for classification tasks, although they can also be applied to regression and outlier detection. SVMs work by finding the optimal hyperplane that separates data points of different classes with the maximum margin. The idea is to not just classify correctly, but to do so with confidence — maximizing the distance from the nearest data points, known as support vectors.

SVMs can handle both linearly separable and non-linearly separable data. For non-linear cases, they use a technique called the kernel trick (e.g., RBF kernel) to project data into a higher-dimensional space where a linear separator might exist.

Use Cases

- Used for any binary or multi-class classification problem where clear separation between classes is valuable. This includes-
  - *Predicting customer response. E.g., ad clicks, email opens, purchase-*
  - *Email spam detection (spam vs. not spam)-*
  - *Image classification (e.g., handwritten digit recognition)-*
  - *Sentiment analysis (positive vs. negative reviews)-*
  - *Bioinformatics (e.g., disease classification based on gene expression)*

Strengths

- Effective in high-dimensional spaces
- Robust to overfitting in small datasets with proper tuning
- Supports non-linear classification via kernels

Weaknesses

- Training can be slow for large datasets
- Results are less interpretable than tree or linear models
- Requires tuning of kernel and regularization parameters
- Not suitable for overlapping or noisy class boundaries

Support Vector Machines

- Support Vector Machines (SVMs) are powerful and versatile supervised learning models used primarily for classification tasks, although they can also be applied to regression and outlier detection.
- SVMs work by finding the optimal hyperplane that separates data points of different classes with the maximum margin.
- The idea is to not just classify correctly, but to do so with confidence — maximizing the distance from the nearest data points, known as support vectors.
- SVMs can handle both linearly separable and non-linearly separable data. For non-linear cases, they use a technique called the kernel trick (e.g., RBF kernel) to project data into a higher-dimensional space where a linear separator might exist.

Applications

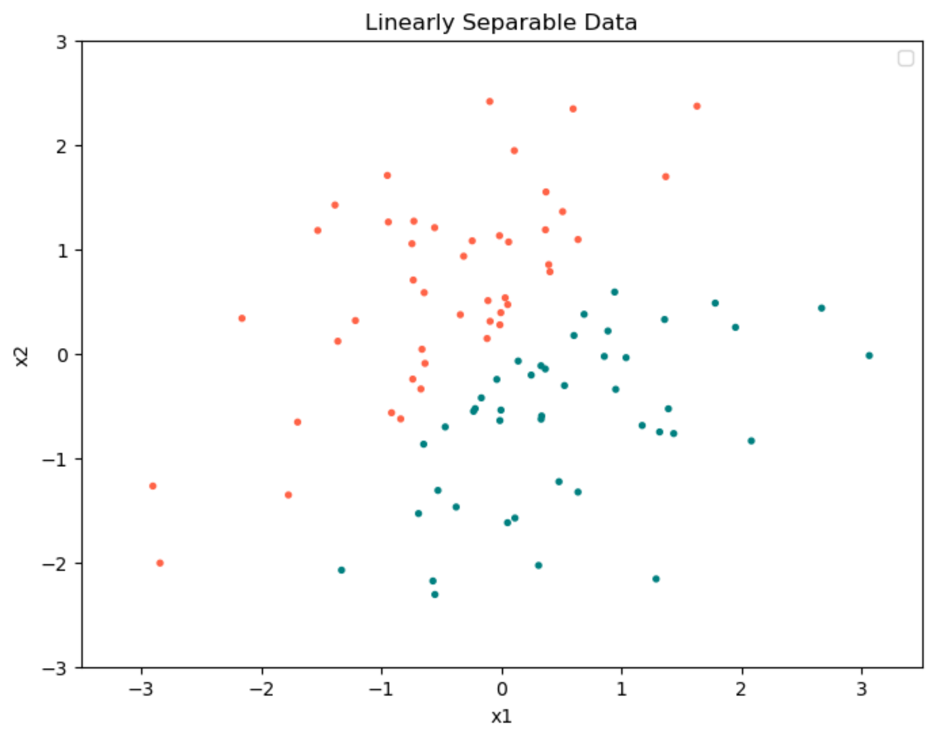
- Used for any binary or multi-class classification problem where clear separation between classes is valuable. This includes-
  - *Predicting customer response. E.g., ad clicks, email opens, purchase-*
  - *Email spam detection (spam vs. not spam)-*
  - *Image classification (e.g., handwritten digit recognition)-*
  - *Sentiment analysis (positive vs. negative reviews)-*
  - *Bioinformatics (e.g., disease classification based on gene expression)*

Support Vector Machines

- Constructs a hyperplane or set of hyperplanes in a high dimensional space which can be use for both classification and regression.
- It is *not that* statistical!
- Flexible and powerful method for making predictions but models are not easy to interpret.

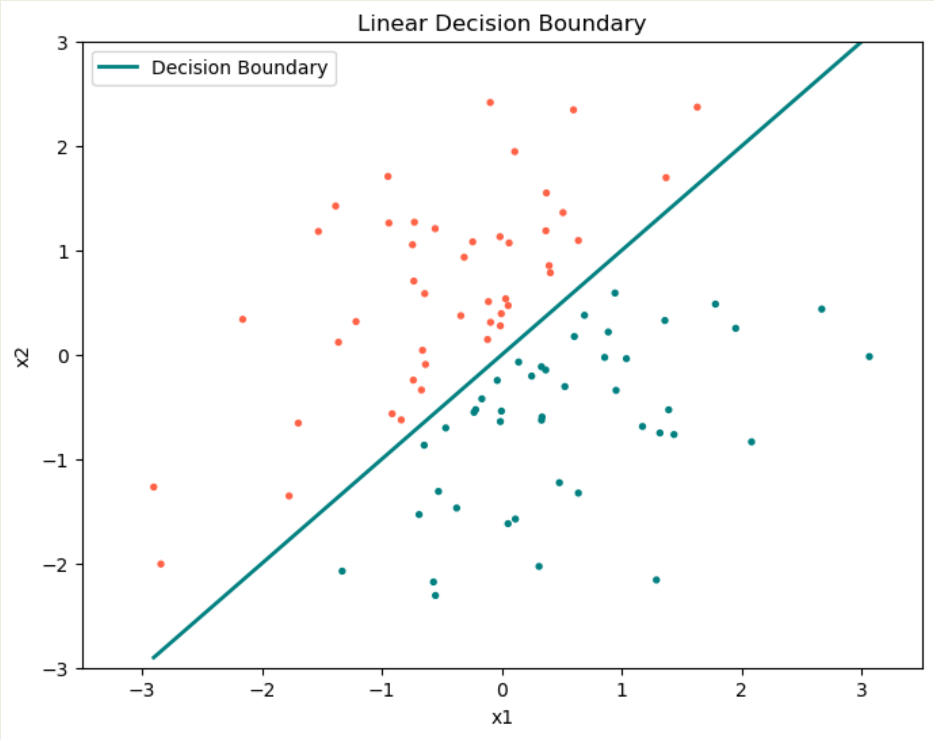
The Intuition

- Begins with trying to find a plane that separates classes in feature space
- Which is great if the classes are linearly separable as seen in the figure.



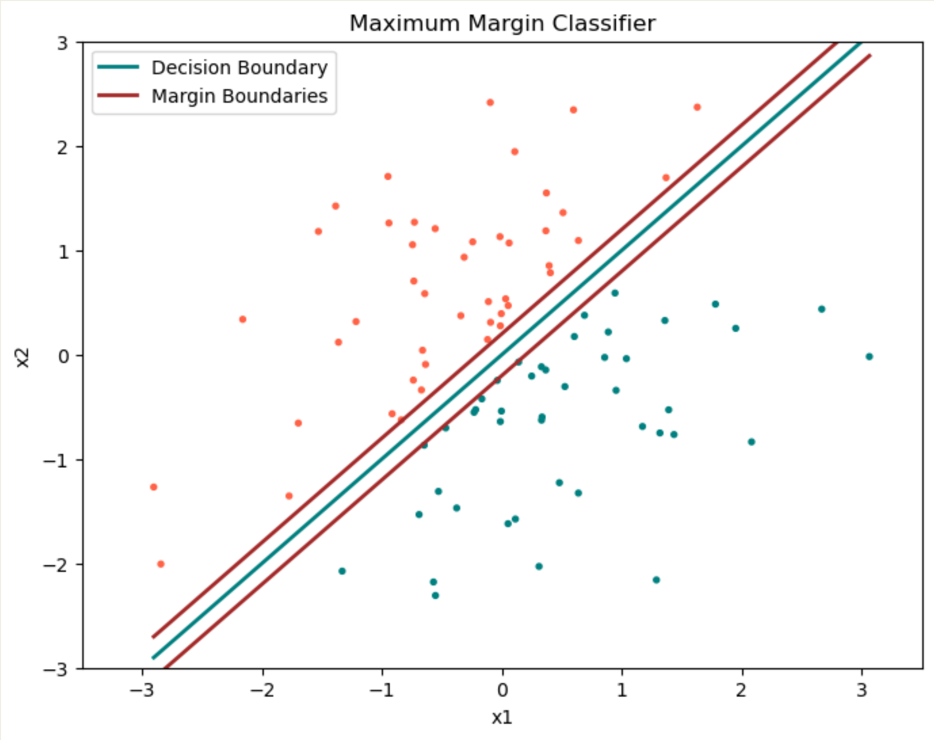
The Intuition

- Fitting a Classifier or Hyperplane to linearly separable classes
- But, these classes can be separated by a very large number of hyperplanes



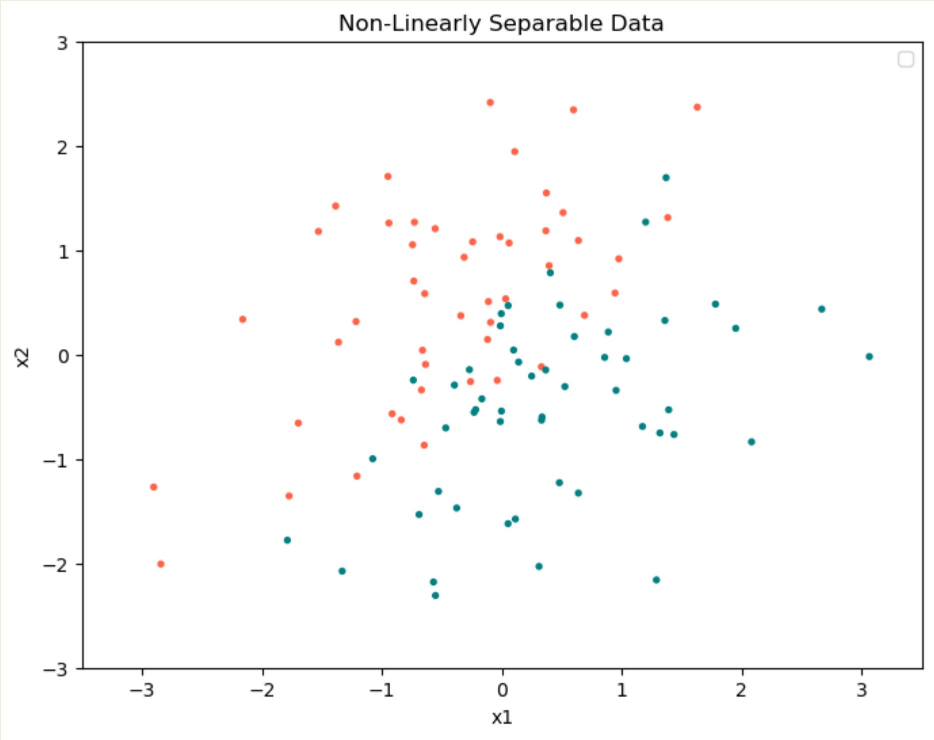
The Intuition

- The decision boundary chosen is the one that has the biggest margin and is accordingly called Maximum Margin Classifier.



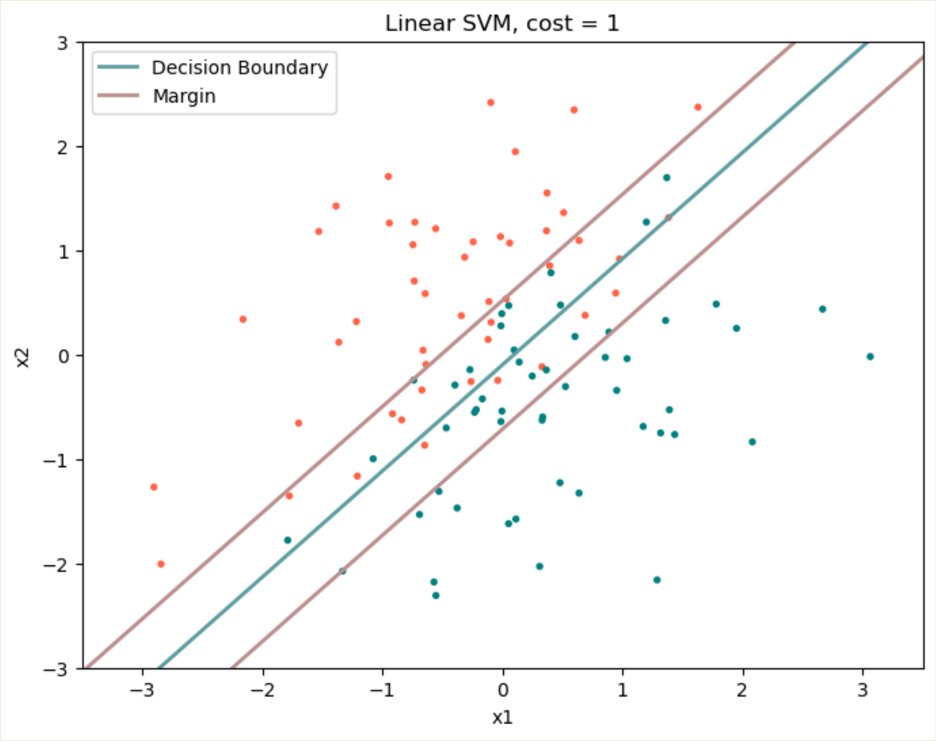
The Intuition

- In practice, classes are seldom linearly separable



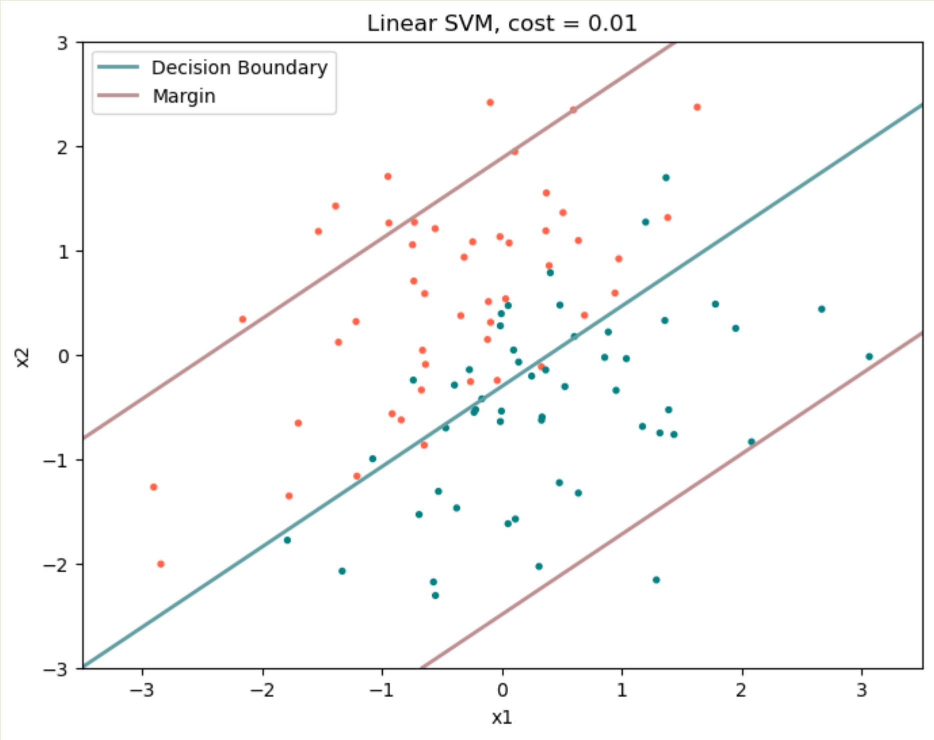
The Intuition

- Therefore, the requirement of a hard margin is relaxed in favor of a soft margin
- The soft margin used is determined by a cost parameter
- Higher cost – narrower margins



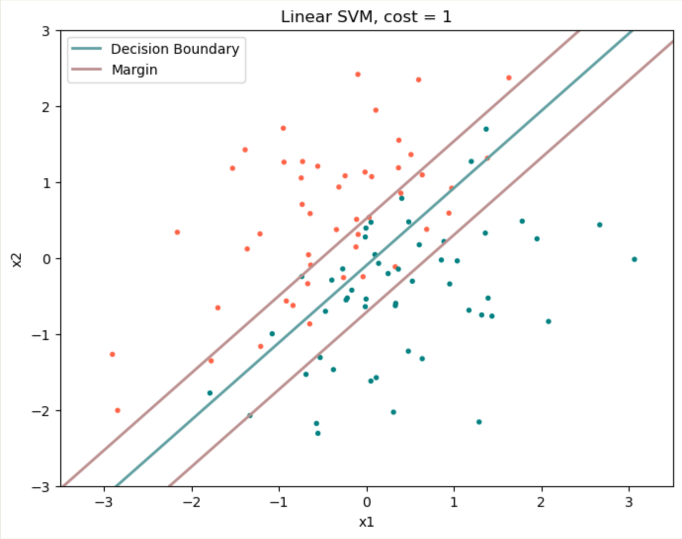
The Intuition

- Therefore, the requirement of a hard margin is relaxed in favor of a soft margin
- The soft margin used is determined by a cost parameter
- Higher cost – narrower margins

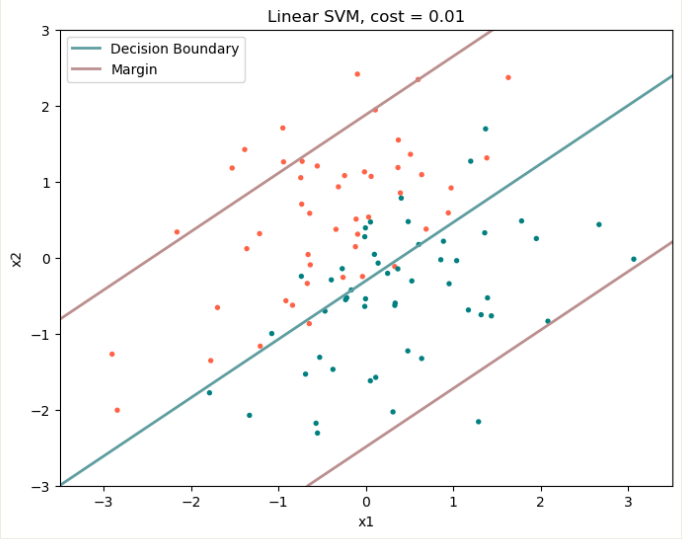


The Intuition

High Cost



Low Cost



Support Vector Machines

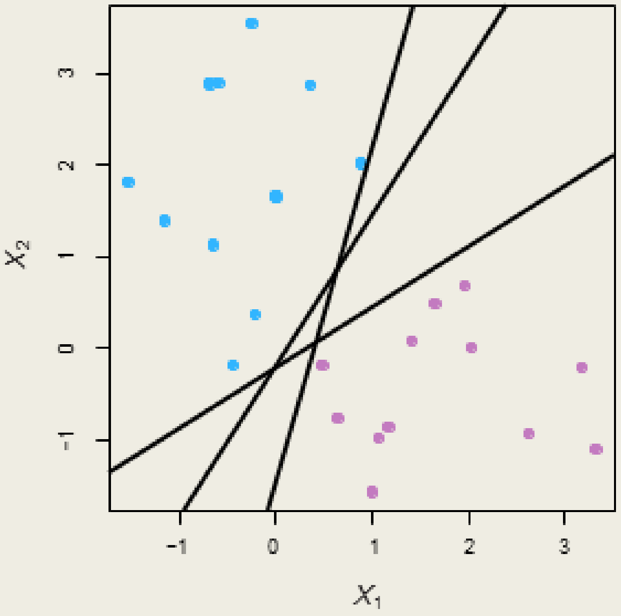
- Begins with trying to find a plane that separates classes in feature space
- Since, in practice this is difficult, the technique
  - *Looks for a soft margin boundary that separates classes*
  - *Enriches and enlarges the features space to make separation possible*

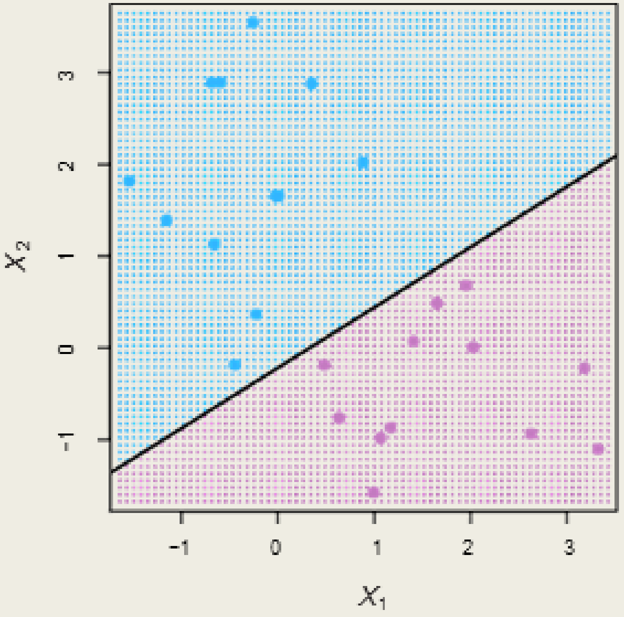
What is a Hyperplane?

- A hyperplane in p dimensions is a flat affine subspace of dimension $p - 1$.
- In general the equation for a hyperplane has the form
- $\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p = 0$
- In $p = 2$ dimensions a hyperplane is a line
- If $\beta_0 = 0$, the hyperplane goes through the origin, otherwise not.
- The vector $\beta = (\beta_1, \beta_2, \cdots, \beta_p)$ is called the normal vector — it points in a direction orthogonal to the surface of a hyperplane

Separating Hyperplanes

- If $f(X) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$, then $f(X) > 0$ for points on one side of the hyperplane, and $f(X) < 0$ for points on the other
- If we code the colored points as $Y_i = +1$ for blue, say, and $Y_i = -1$ for mauve, then if $Y_i \cdot f(X_i) > 0$ for all i, $f(X) = 0$ defines a *separating hyperplane*

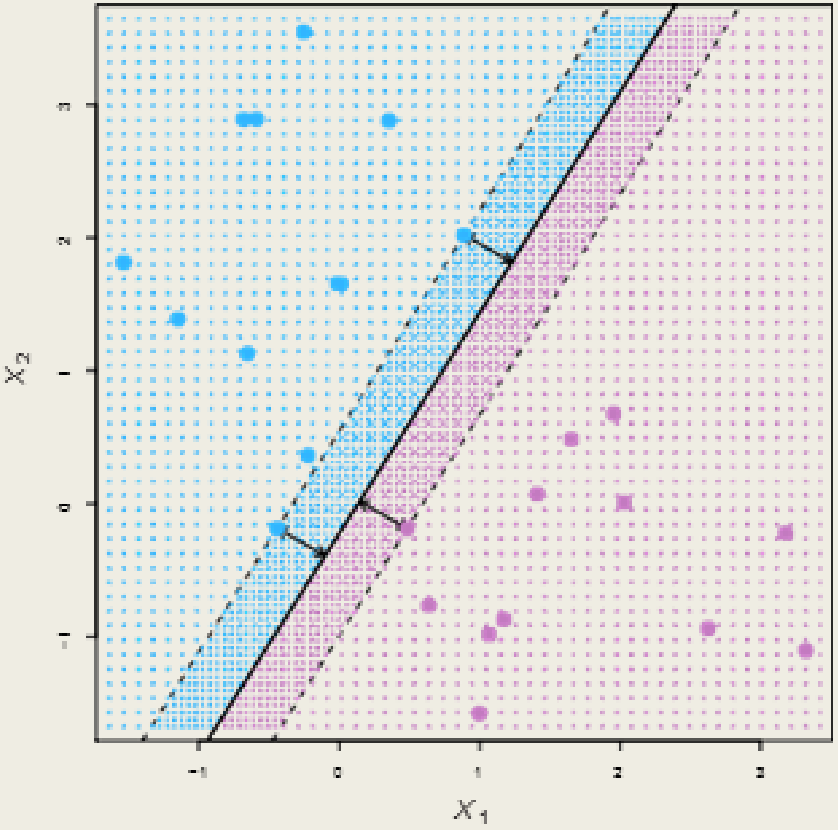




Source: James et al (2023)

Maximum Margin Classifier

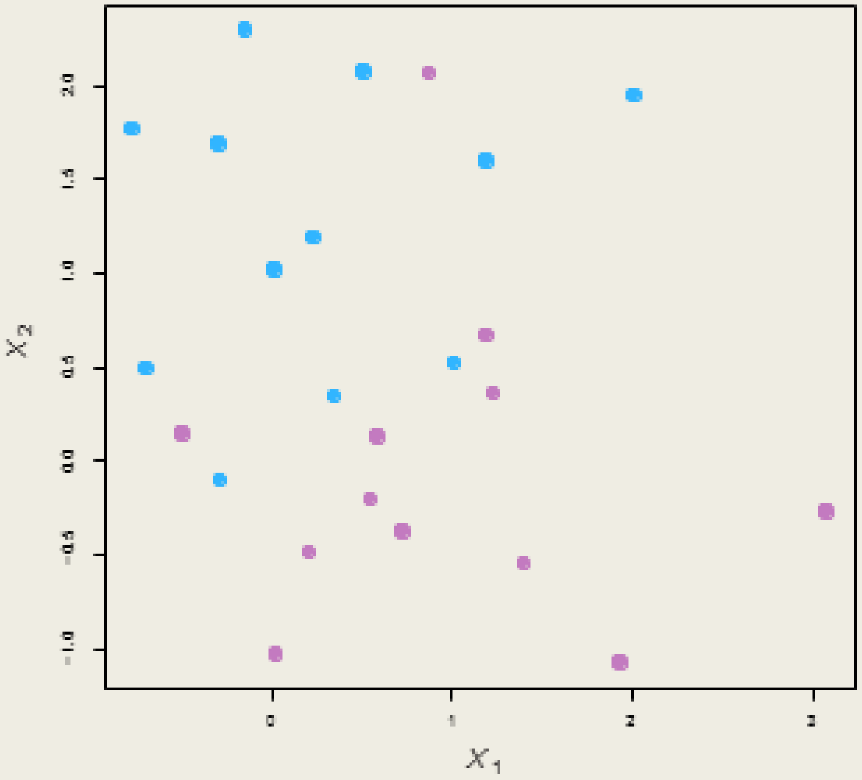
- Among all separating hyperplanes, find the one that makes the biggest gap or margin between the two classes



Source: James et al (2023)

Non-separable Data

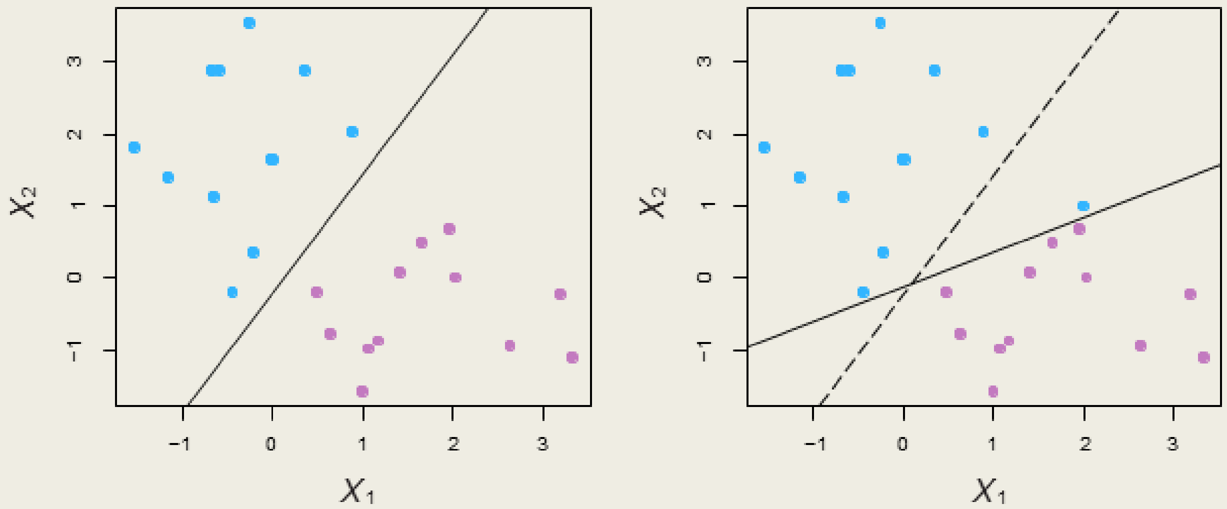
- But, most data is not linearly separable
  - *Exception being when $n < p$*



Source: James et al (2023)

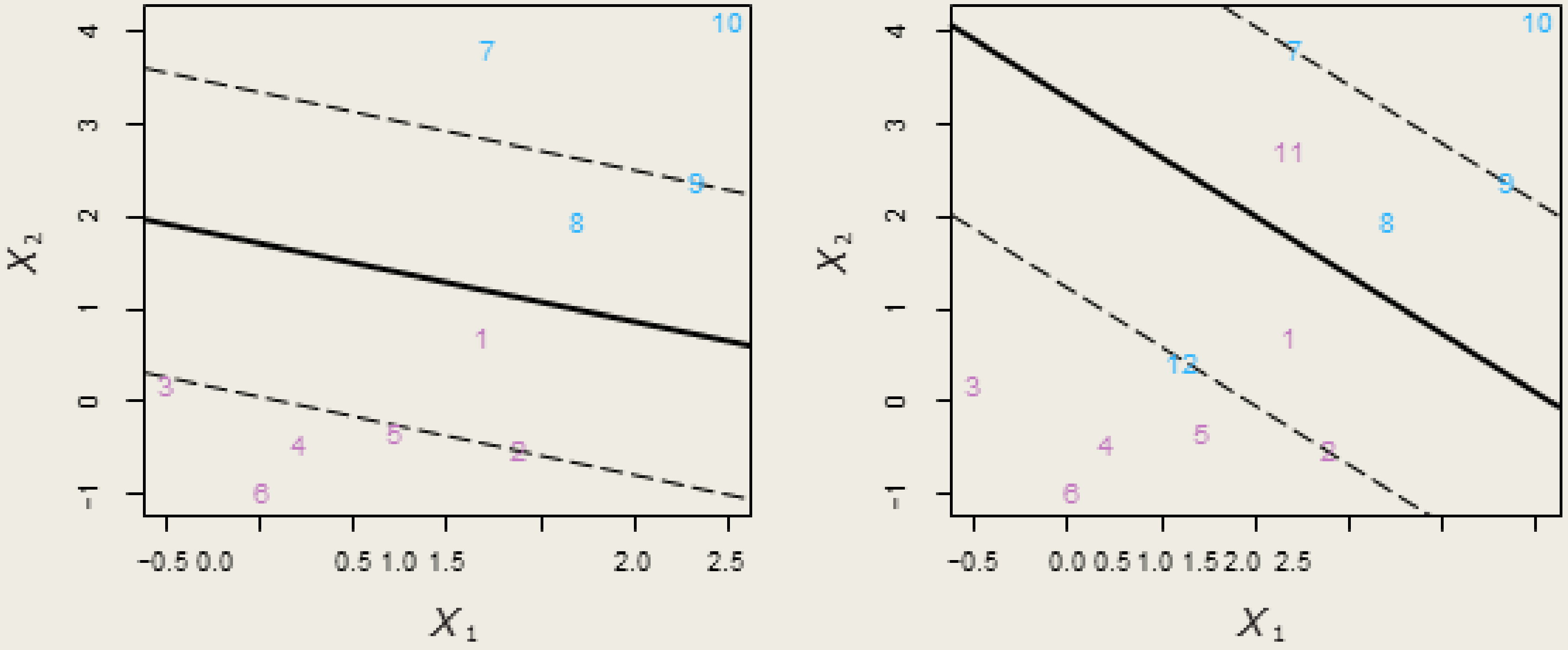
Noisy Data

- Noisy data can lead to a poor solution for the maximal-margin classifier.
- The support vector classifier maximizes a soft margin.



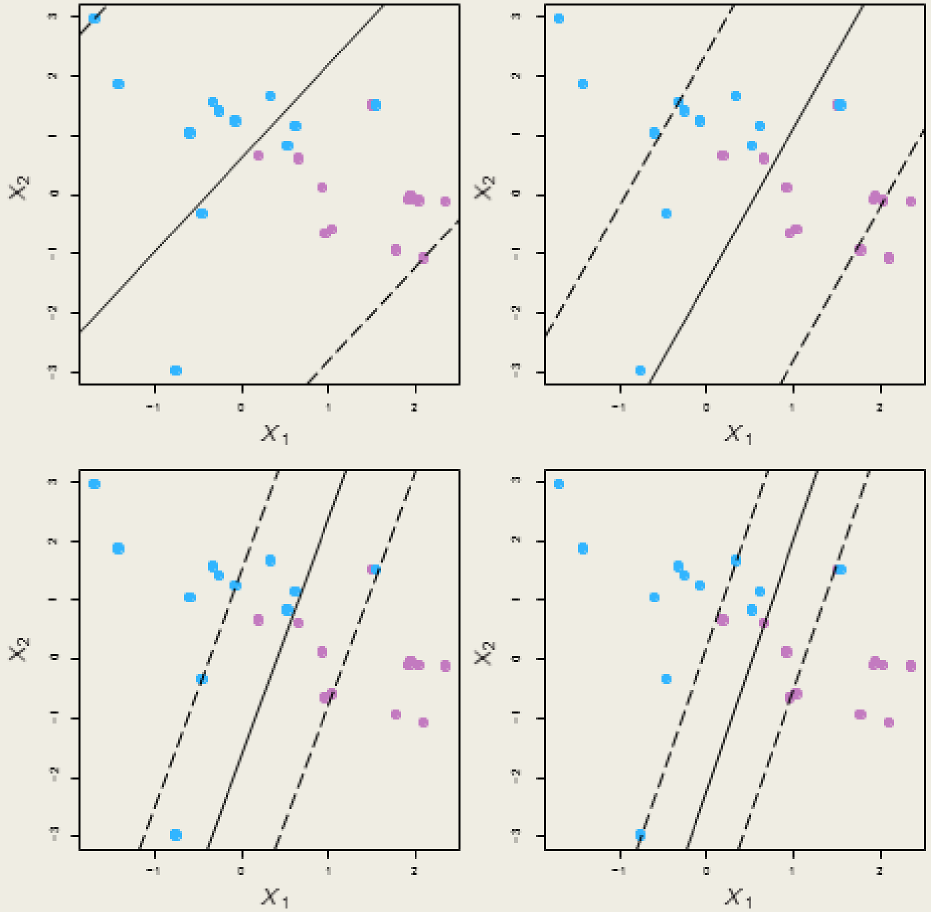
Source: James et al (2023)

Support Vector Classifier



Source: James et al (2017)

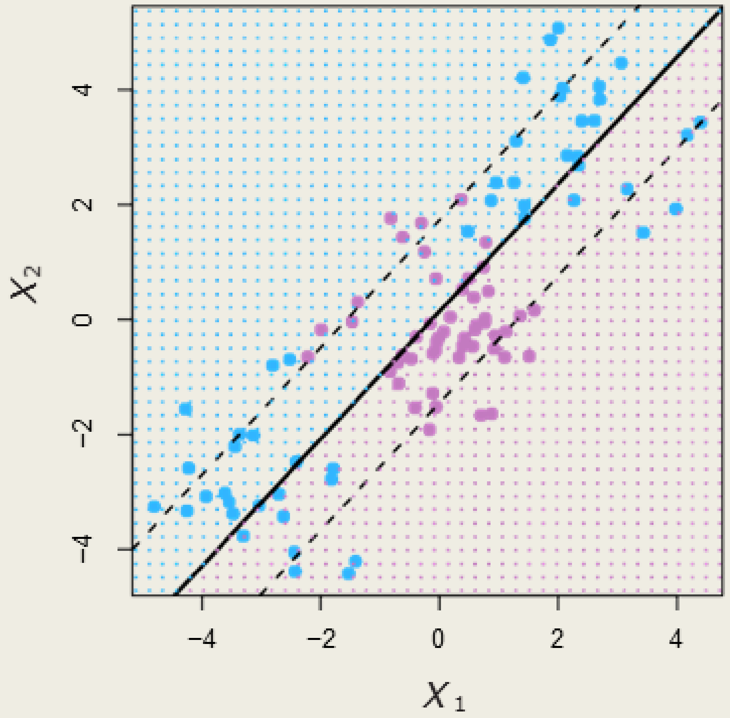
Cost, c, is a Regularization Parameter



Source: James et al (2023)

But, Linear Boundary can Fail

- Sometimes a linear boundary fails, no matter how high the value of C.
- Here is an example.



Source: James et al (2023)

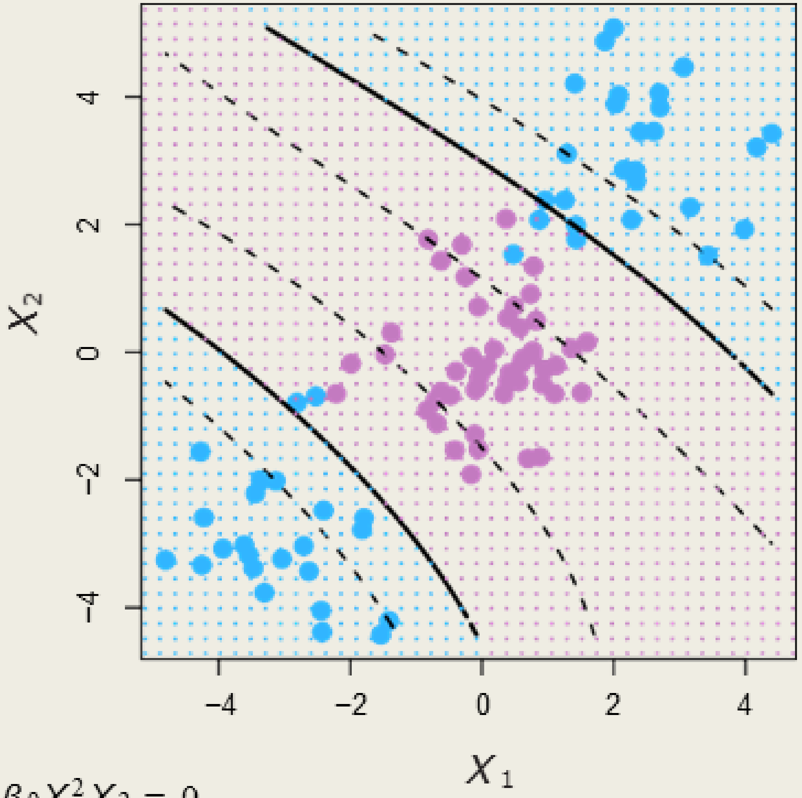
Feature Expansion

- Enlarge the space of features by including transformations, e.g., X_1^2, X_1^3, $X_1X_2^2$. By doing so, we go from a p-dimensional space to an M>p dimensional space.
- Fit a support-vector classifier in the enlarged space
- This results in non-linear decision boundaries in the original space
- E.g., if we use a vector space of $(X_1, X_2, X_1^2, X_2^2, X_1X_2)$ instead of (X_1, X_2)
- Then the decision boundary would be of the form
  - $\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1^2 + \beta_4 X_2^2 + \beta_5 X_1X_2 = 0$
- This leads to nonlinear decision boundaries in the original space (quadratic conic sections).

Source: James et al (2023)

Cubic Polynomials

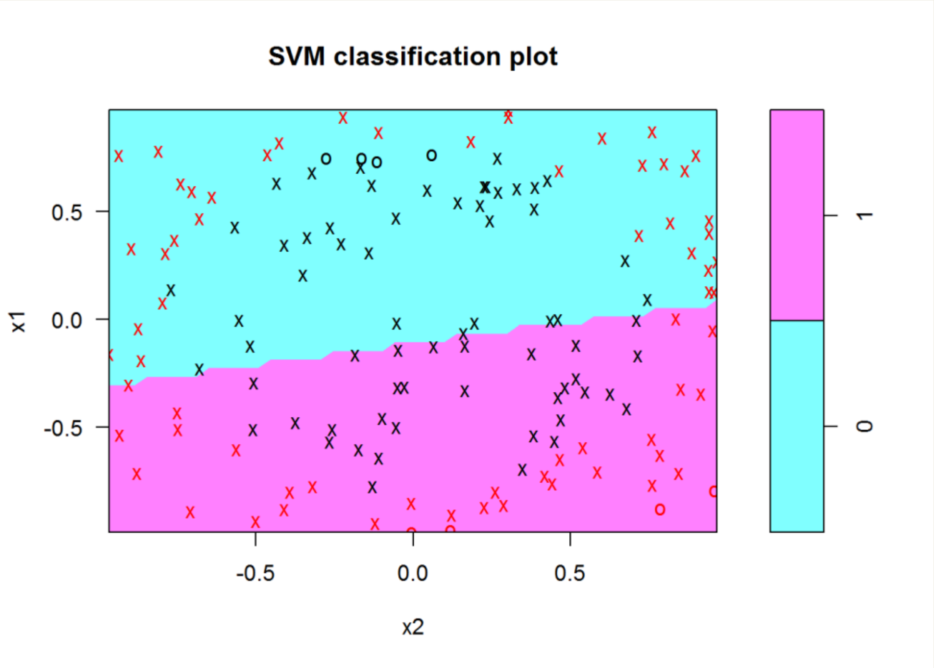
- Basis Expansion of Cubic Polynomials from 2 variables to 9
- The support-vector classifier in the enlarged space solves the problem in the lower-dimensional space

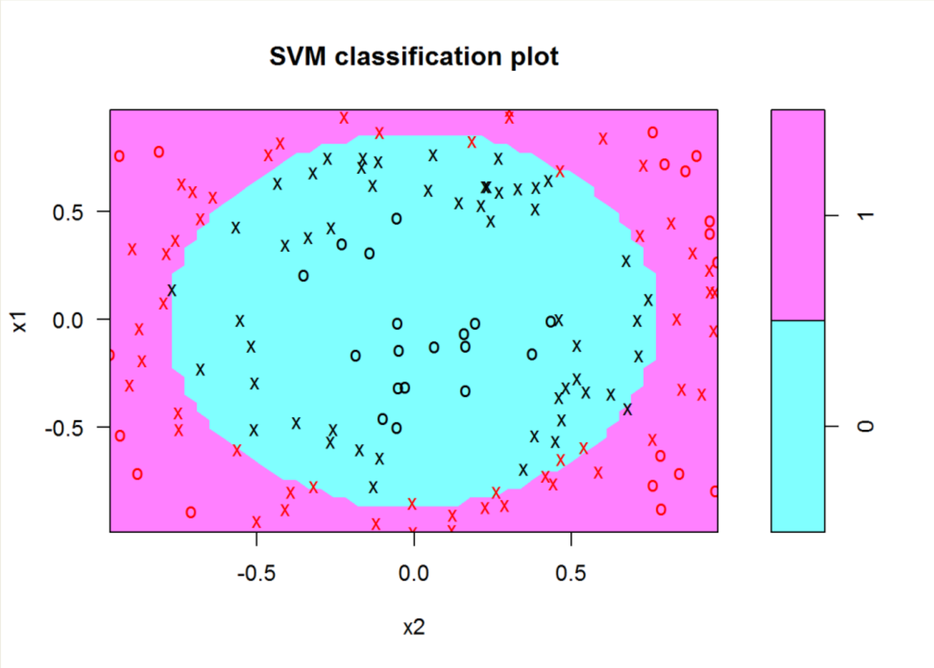


$$\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_1^2+\beta_4X_2^2+\beta_5X_1X_2+\beta_6X_1^3+\beta_7X_2^3+\beta_8X_1X_2^2+\beta_9X_1^2X_2=0$$

Source: James et al (2023)

Linear vs. Polynomial Kernel



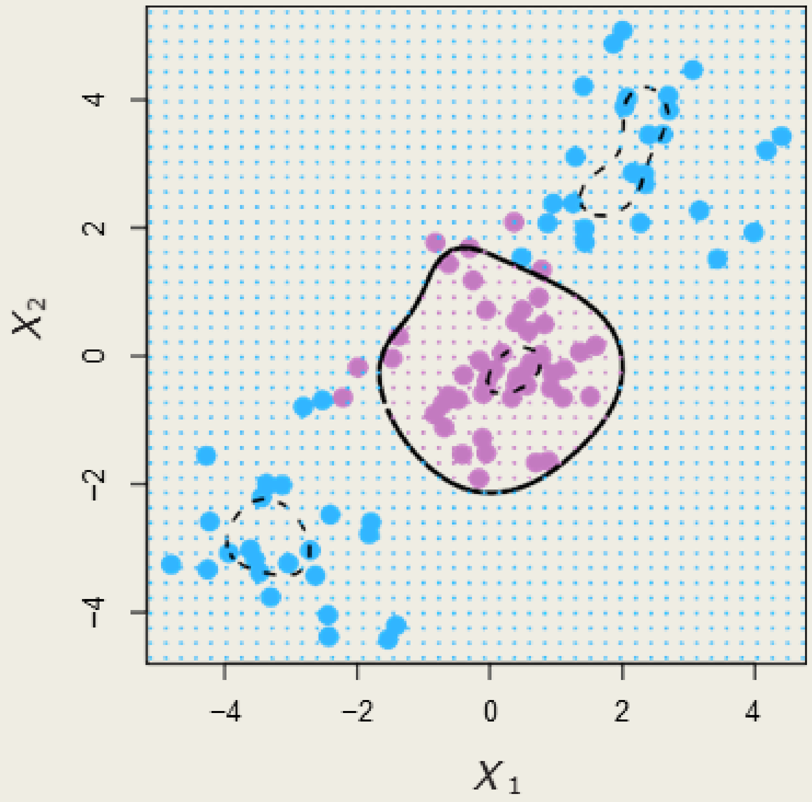


Nonlinearities and Kernels

- 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.
- If we can compute inner products between observations, we can fit a support vector classifier.
- This is made possible by some simple kernel functions like a Radial Basis Kernel

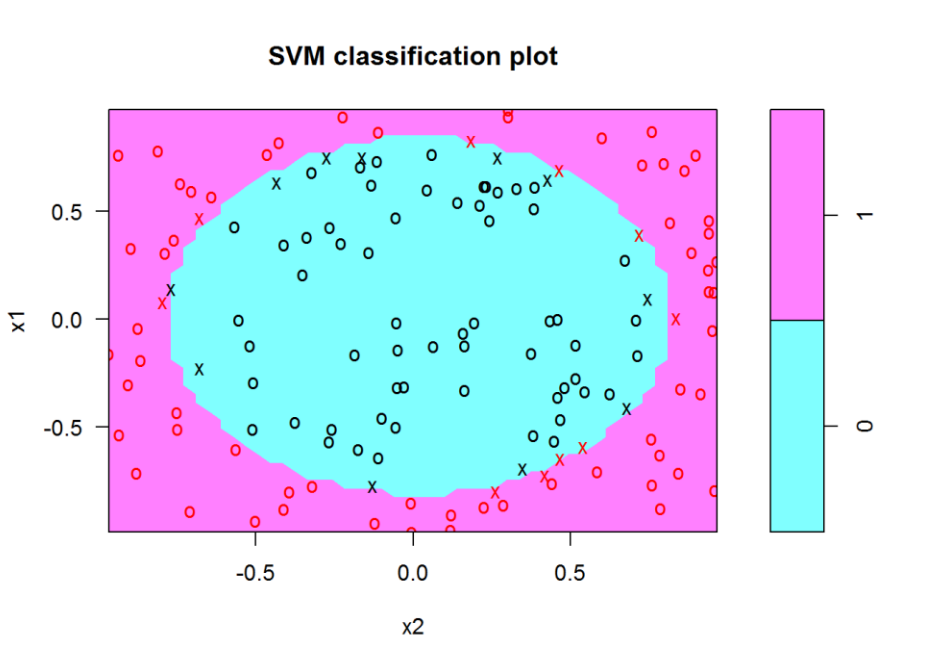
Source: James et al (2023)

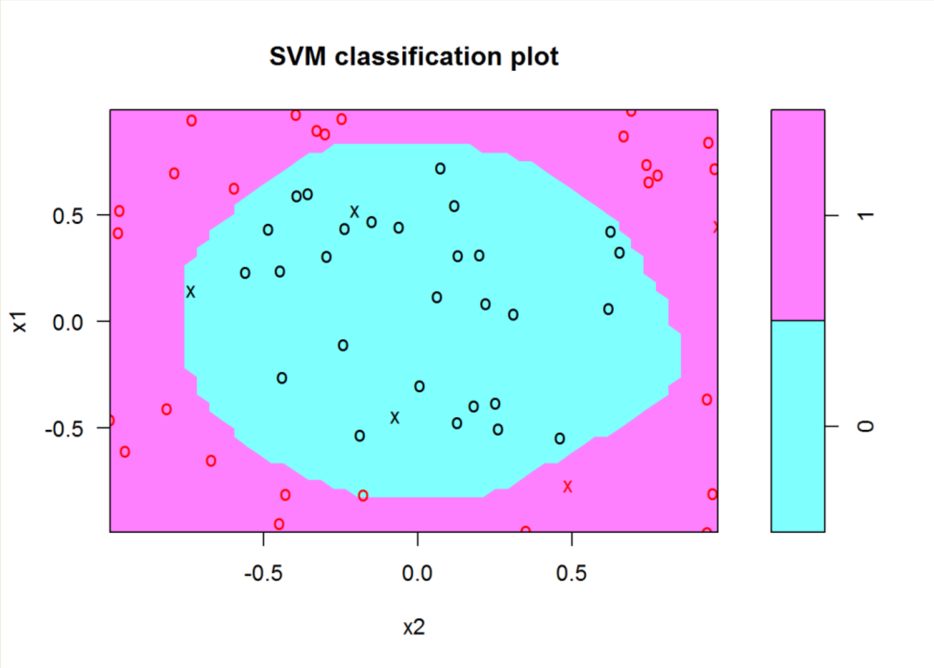
Radial Kernel



Source: James et al (2023)

Polynomial vs. Radial Kernel





SVM for more than 2 classes

- SVM can be extended to a situation with more than two classes. There are two approaches to this
  - *One versus all: Fit k different 2 class SVM classifiers*
  - *One versus One: Fit all pairwise classifiers. Use if k is not too large.*

SVM vs. 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.

Conclusion

- In this session, we
  - *Examined the intuition behind support vector machines*
  - *Discussed linear classifiers*
  - *Looked at feature expansion to accommodate non-linear decision boundaries*
  - *Examined polynomial and radial kernels*
  - *Compared SVM to logistic regression*
  - *Looked at implementation of SVM*