BM5702 MAKİNE ÖĞRENMESİNE GİRİŞ

Hafta 11

Doç. Dr. Murtaza CİCİOĞLU

Support Vector Machines

- Linear models are also extensively used for classification.

$$\hat{y} = w[0] * x[0] + w[1] * x[1] + \ldots + w[p] * x[p] + b > 0$$

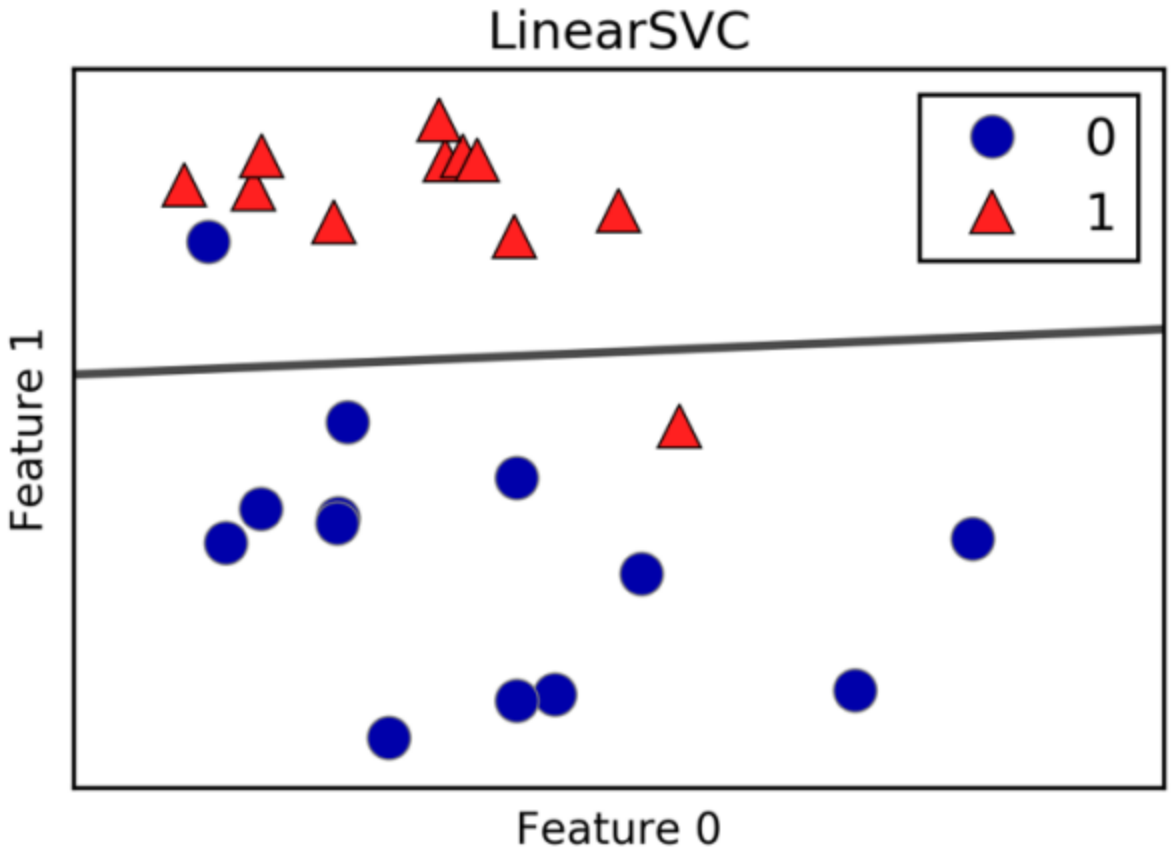
- The formula looks very similar to the one for linear regression, but instead of just returning the weighted sum of the features, we threshold the predicted value at zero.
- If the function is smaller than zero, we predict the class –1; if it is larger than zero, we predict the class +1.
- This prediction rule is common to all linear models for classifica- tion. Again, there are many different ways to find the coefficients (w) and the intercept (b).

Support Vector Machines

- For linear models for regression, the output, $\hat{y}$, is a linear function of the features: a line, plane, or hyperplane (in higher dimensions).
- For linear models for classification, the decision boundary is a linear function of the input. In other words, a (binary) linear classifier is a classifier that separates two classes using a line, a plane, or a hyperplane.
- There are many algorithms for learning linear models. These algorithms all differ in the following two ways:
  - The way in which they measure how well a particular combination of coefficients and intercept fits the training data
  - If and what kind of regularization they use (L1, L2)
- linear support vector machines (linear SVMs), implemented in svm.LinearSVC (SVC stands for support vector classifier).

Support Vector Machines

- By default, both models apply an L2 regularization, in the same way that Ridge does for regression.

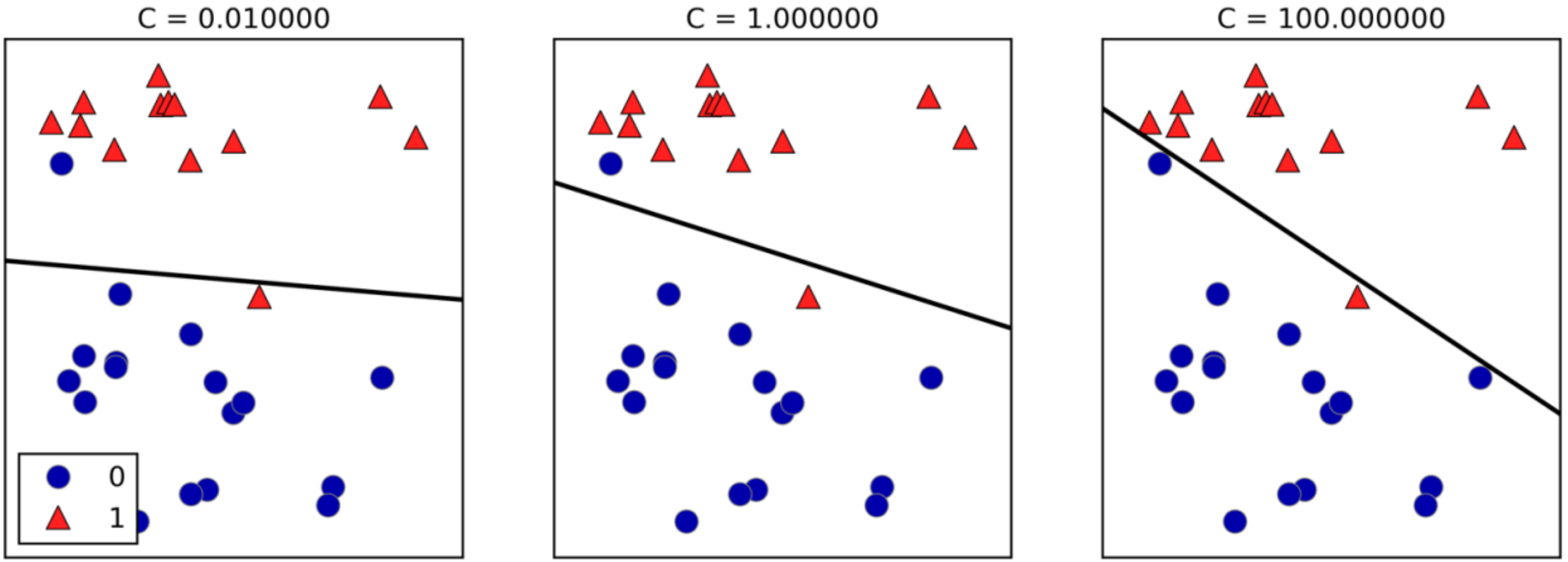


Support Vector Machines

- For LinearSVC the trade-off parameter that determines the strength of the regularization is called C, and higher values of C correspond to less regularization.
- In other words, when you use a high value for the parameter C, LogisticRegression and LinearSVC try to fit the training set as best as possible, while with low values of the parameter C, the models put more emphasis on finding a coefficient vector (w) that is close to zero.

Support Vector Machines

- On the lefthand side, we have a very small C corresponding to a lot of regularization.

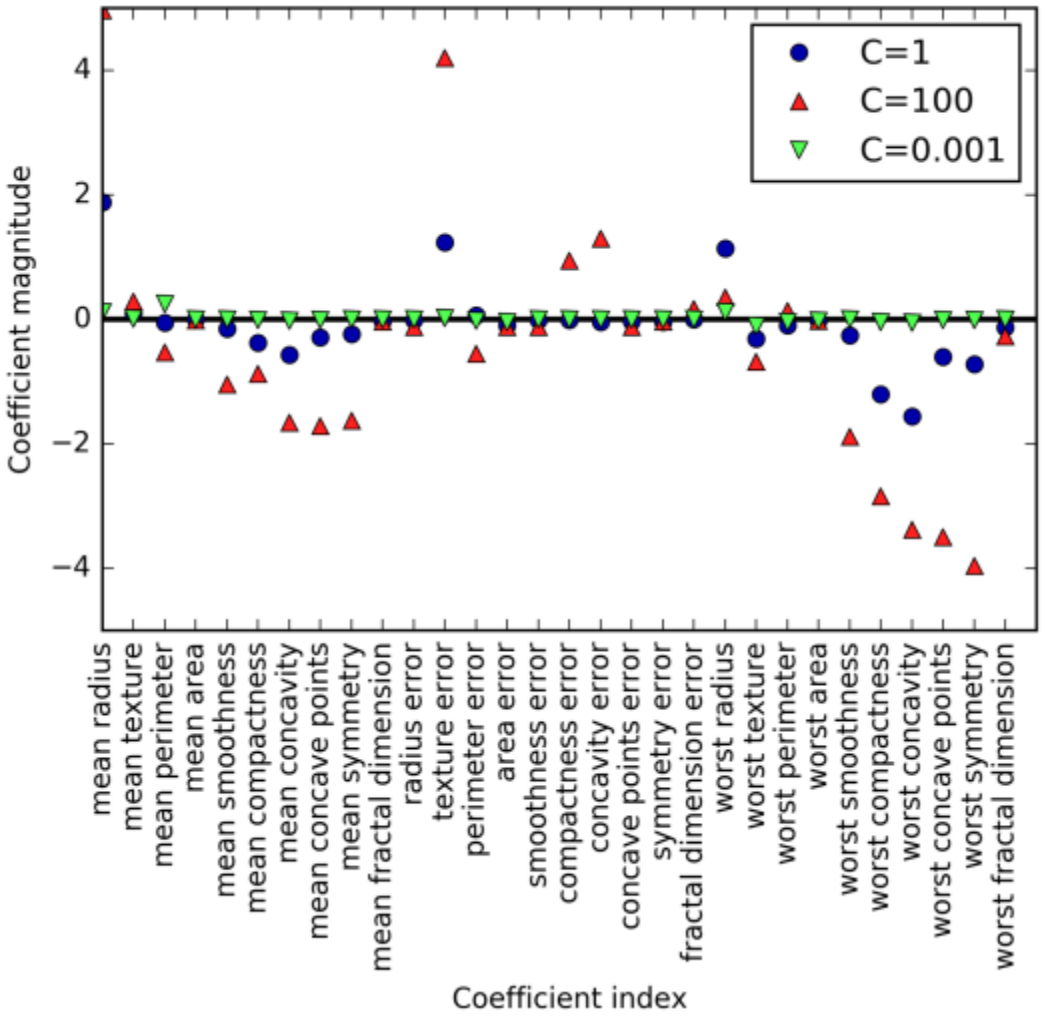


Support Vector Machines

- Similarly to the case of regression, linear models for classification might seem very restrictive in low-dimensional spaces, only allowing for decision boundaries that are straight lines or planes.
- Again, in high dimensions, linear models for classification become very powerful, and guarding against overfitting becomes increasingly important when considering more features.

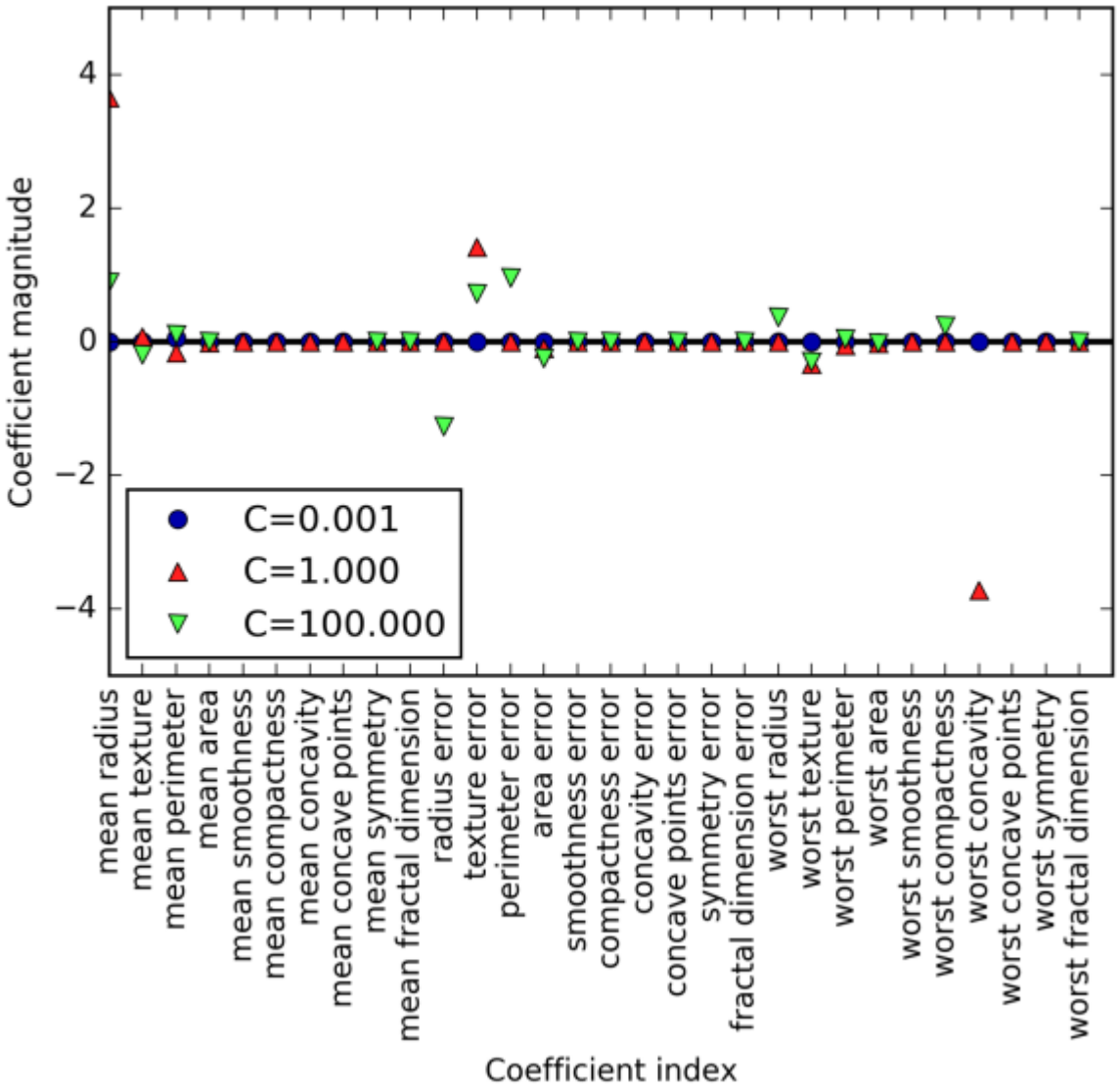
Support Vector Machines

- Coefficients learned by logistic regression on the Breast Cancer dataset for different values of C
- Default L2 regularization
- Stronger regularization pushes coefficients more and more toward zero, though coefficients never become exactly zero.



Support Vector Machines

- If we desire a more interpretable model, using L1 regularization might help, as it limits the model to using only a few features.

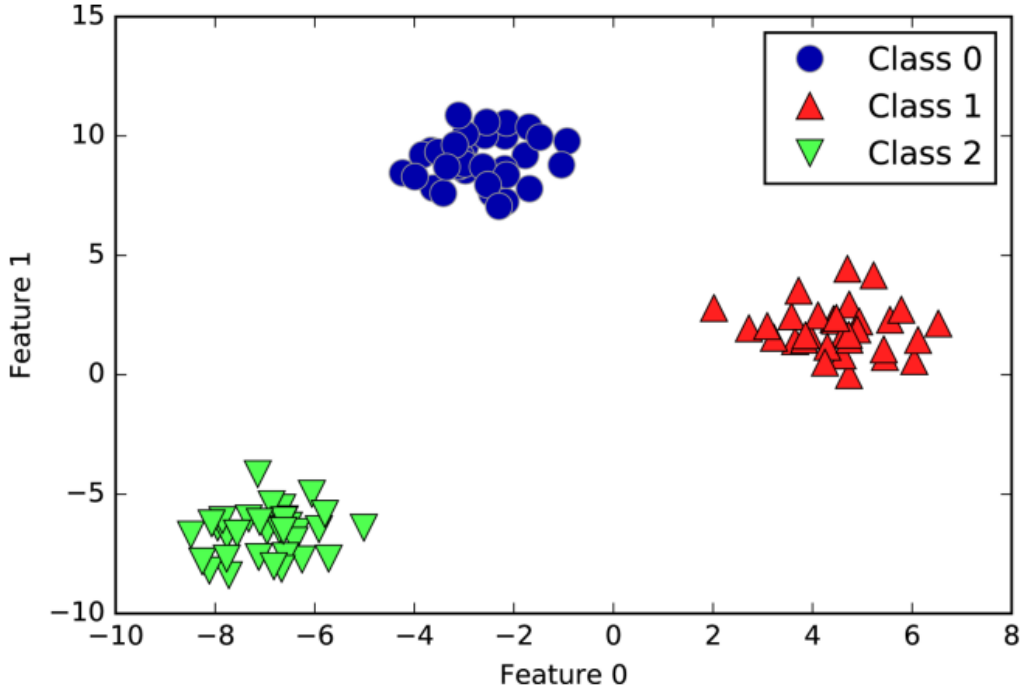


Support Vector Machines

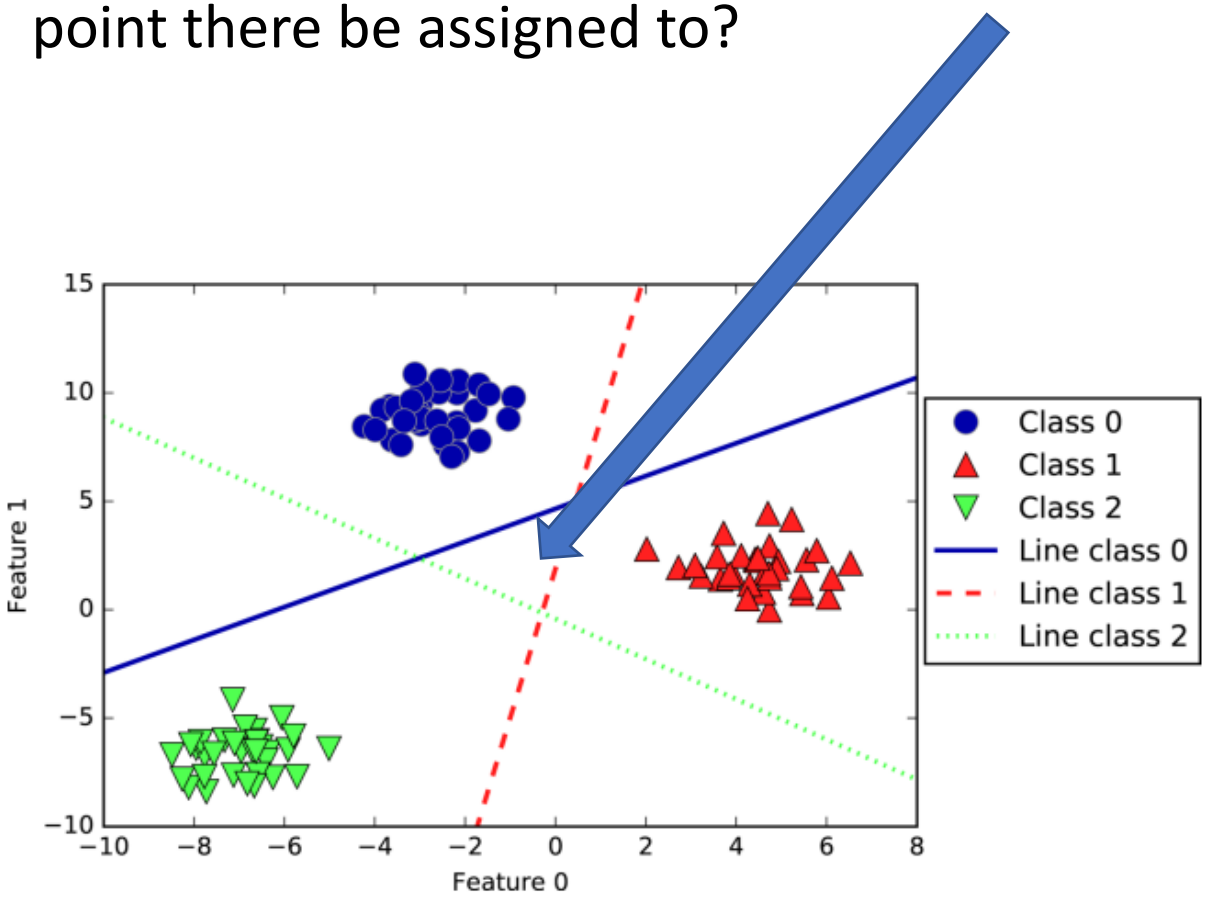
- Linear models for multiclass classification
- Many linear classification models are for binary classification only, and don't extend naturally to the multiclass case (with the exception of logistic regression).
- To make a prediction, all binary classifiers are run on a test point. The classifier that has the highest score on its single class "wins," and this class label is returned as the prediction.
- Having one binary classifier per class results in having one vector of coefficients (w) and one intercept (b) for each class.

Support Vector Machines

- LinearSVC() multiclass classification

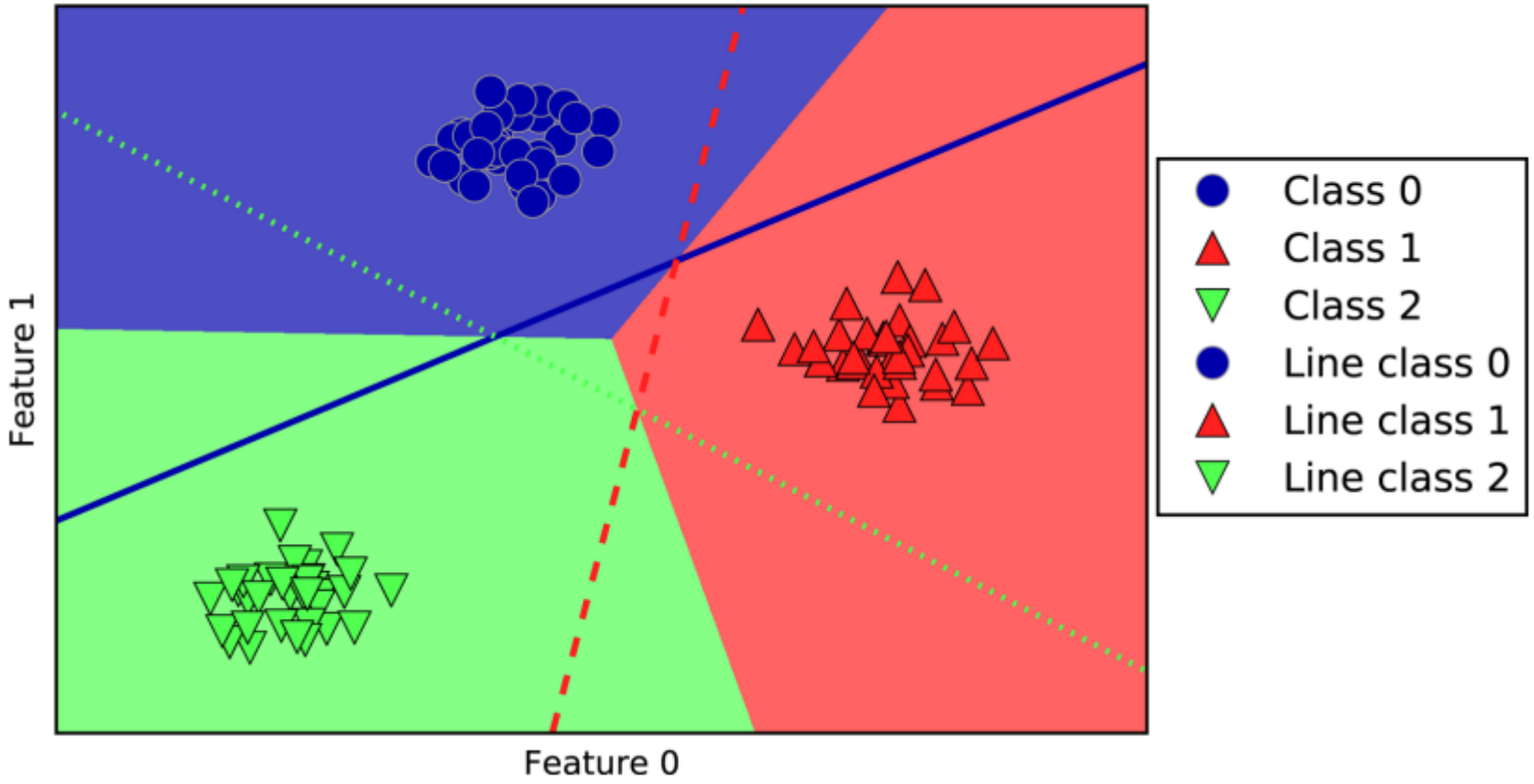


But what about the triangle in the middle of the plot? All three binary classifiers classify points there as “rest.” Which class would a point there be assigned to?



Support Vector Machines

- LinearSVC() multiclass classification



Strengths, weaknesses, and parameters

- The main parameter of linear models is the regularization parameter, called alpha in the regression models and C in LinearSVC and LogisticRegression.
- Large values for alpha or small values for C mean simple models. In particular for the regression models, tuning these parameters is quite important.
- Usually C and alpha are searched for on a logarithmic scale. The other decision you have to make is whether you want to use L1 regularization or L2 regularization.
- If you assume that only a few of your features are actually important, you should use L1. Otherwise, you should default to L2.
- L1 can also be useful if interpretability of the model is important. As L1 will use only a few features, it is easier to explain which features are important to the model, and what the effects of these features are.

Strengths, weaknesses, and parameters

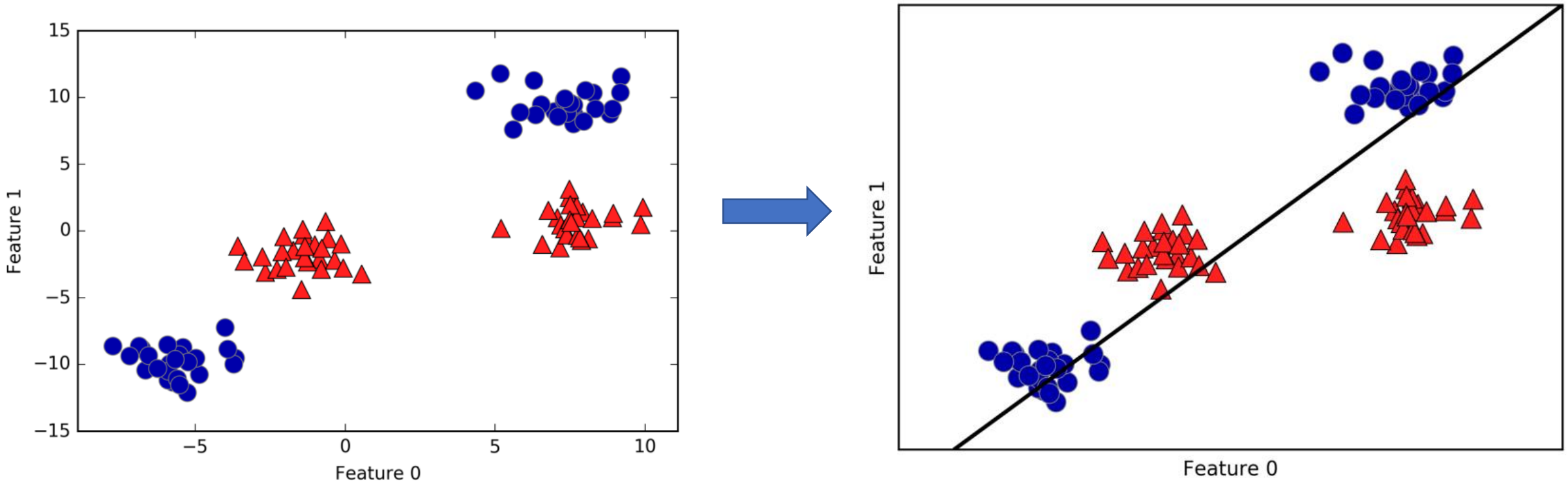
- Linear models are very fast to train, and also fast to predict.
- They scale to very large datasets and work well with sparse data.
- Another strength of linear models is that they make it relatively easy to understand how a prediction is made, using the formulas.
- Unfortunately, it is often not entirely clear why coefficients are the way they are. This is particularly true if your dataset has highly correlated features; in these cases, the coefficients might be hard to interpret.

Kernelized Support Vector Machines

- Kernelized support vector machines (often just referred to as SVMs) are an extension that allows for more complex mod- els that are not defined simply by hyperplanes in the input space.
- Similar concepts apply to support vector regression, as implemented in SVR.
- linear models can be quite limiting in low-dimensional spaces, as lines and hyperplanes have limited flexibility. One way to make a linear model more flexible is by adding more features—for example, by adding interactions or polynomials of the input features.

Kernelized Support Vector Machines

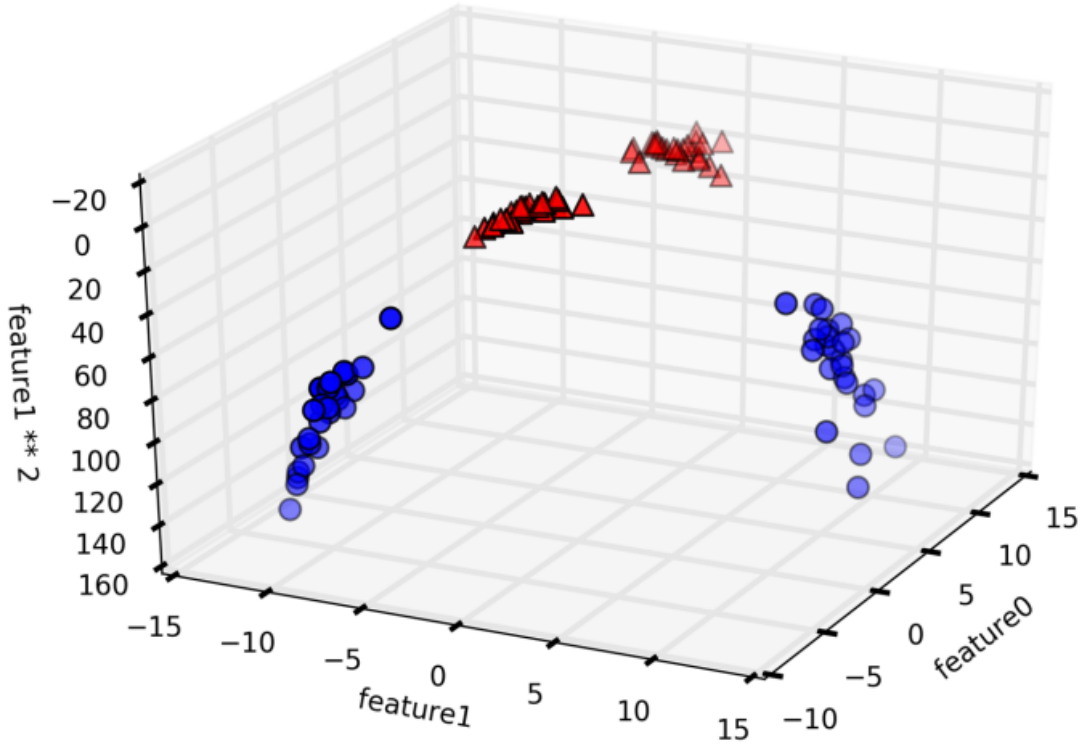
- Two-class classification dataset in which classes are not linearly separable

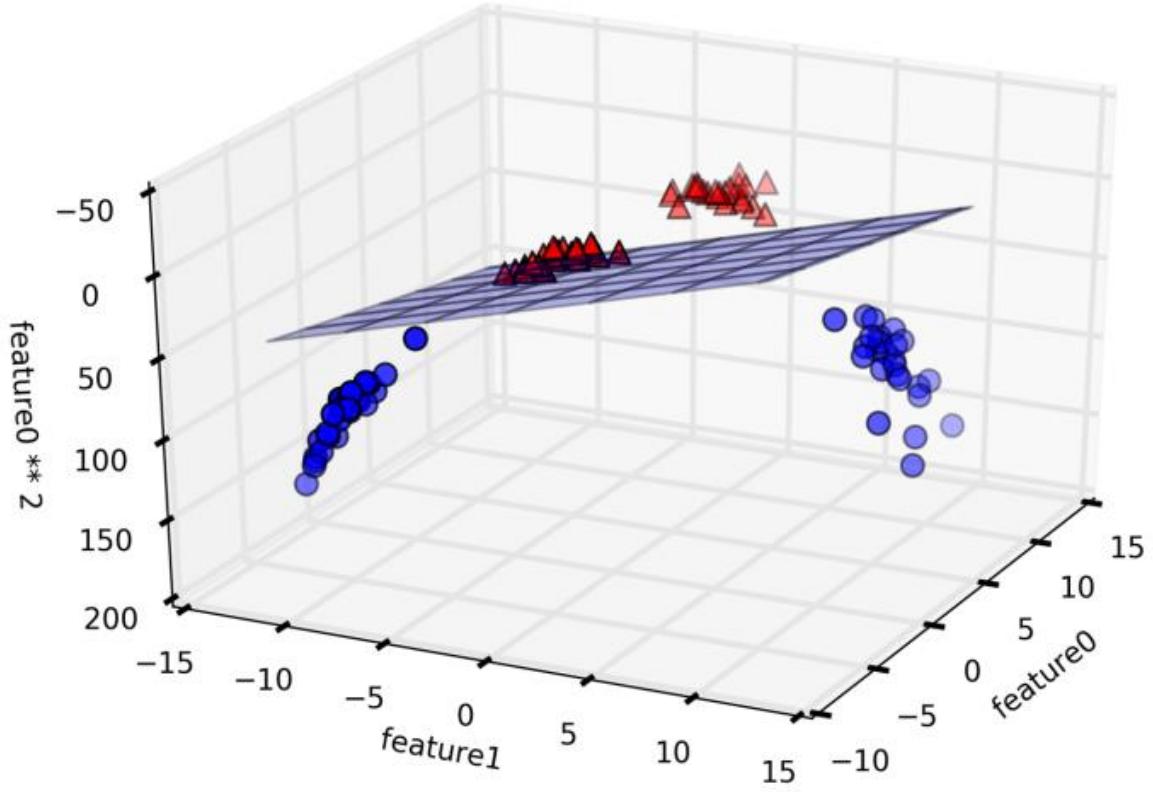


Decision boundary found by a linear SVM

Kernelized Support Vector Machines

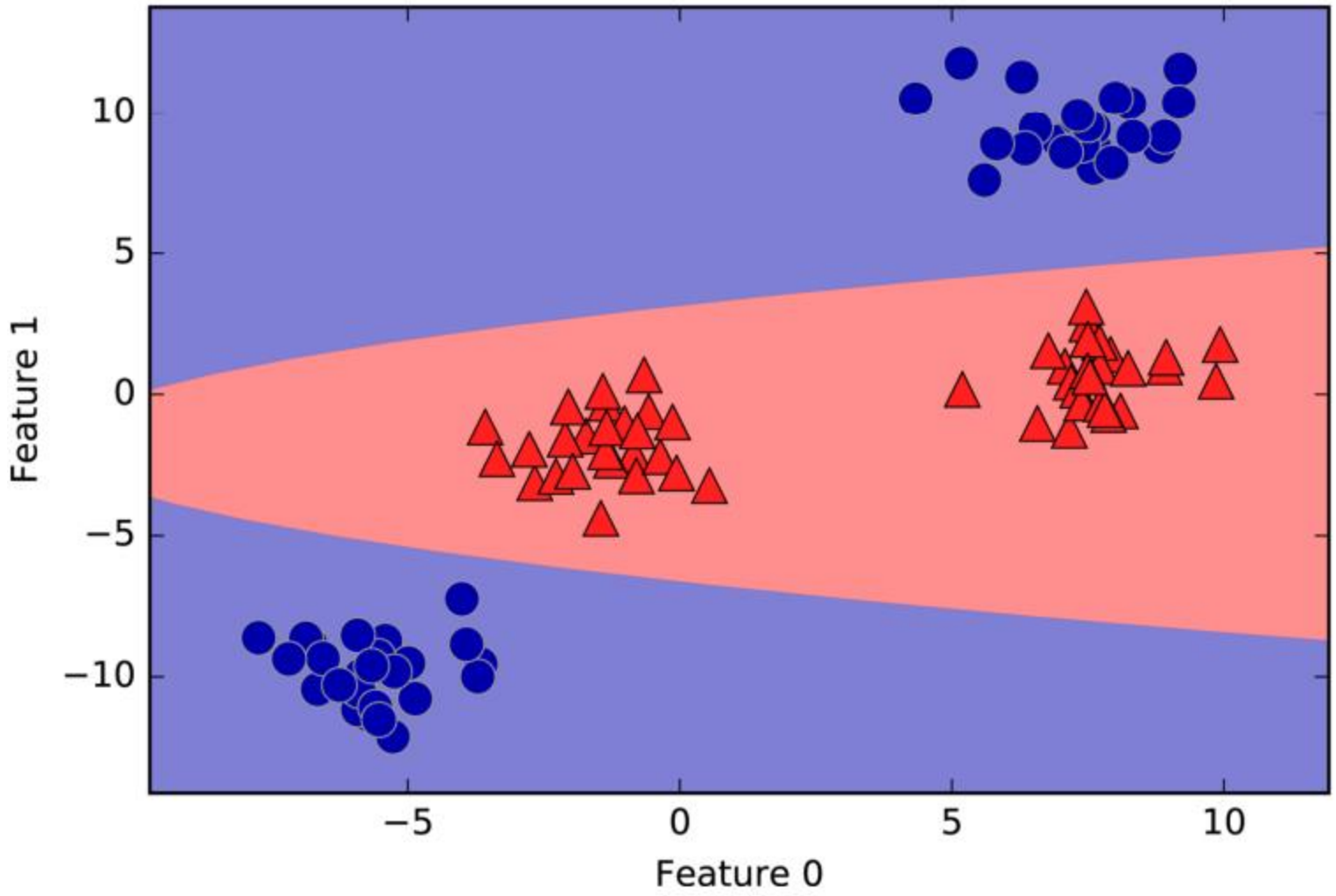
- Decision boundary found by a linear SVM on the expanded threedimensional dataset





Kernelized Support Vector Machines

- Decision boundary



Kernelized Support Vector Machines

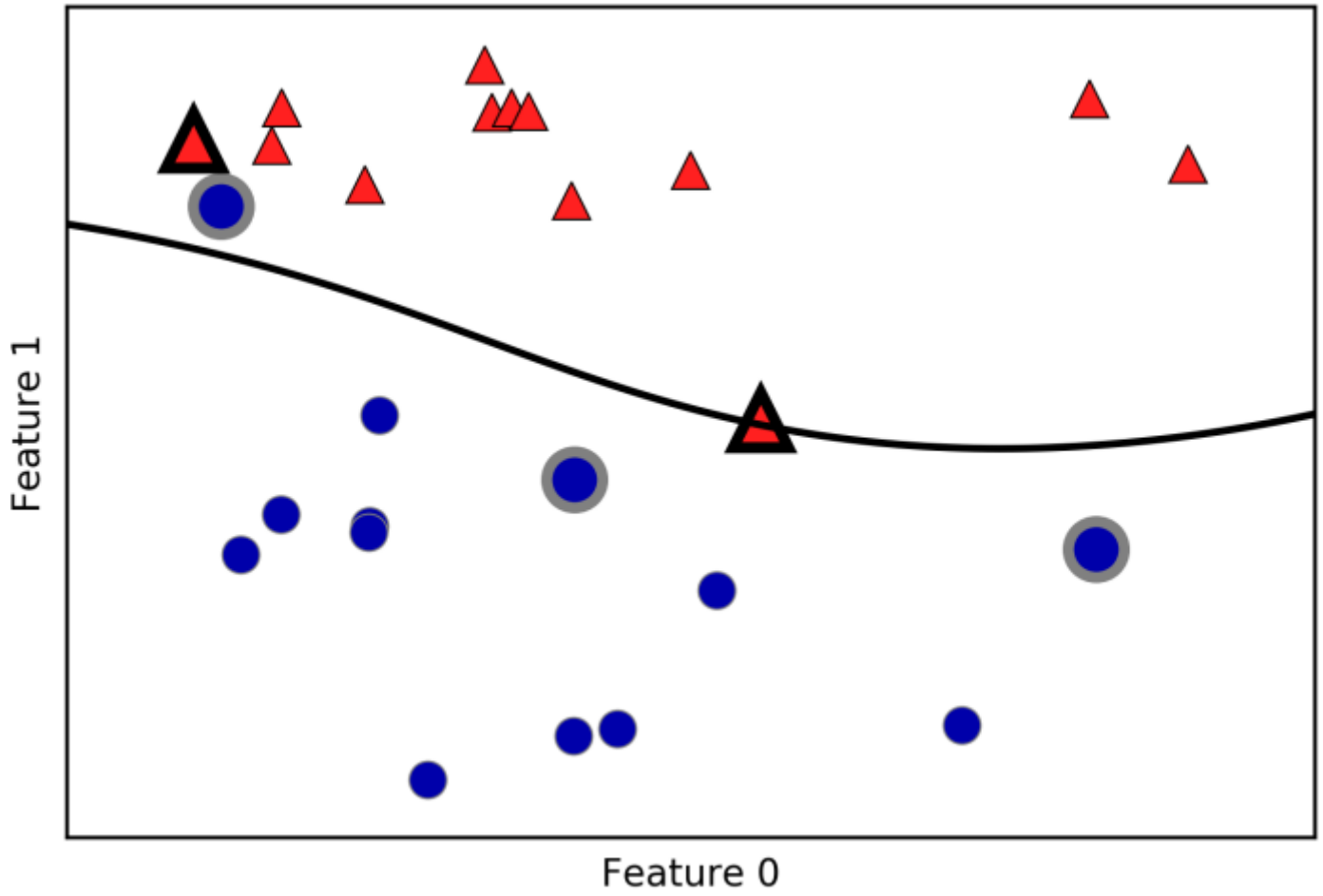
- The kernel trick
- the polynomial kernel, which computes all possible polynomials up to a certain degree of the original features (like feature1 ** 2 * feature2 ** 5); and the radial basis function (RBF) kernel, also known as the Gaussian kernel.
- The distance between data points is measured by the Gaussian kernel:

$$k_{\text{rbf}}(x_1, x_2) = \exp\left(-\gamma \| x_1 - x_2 \|^2\right)$$

- Here, x1 and x2 are data points, ‖ x1 - x2 ‖ denotes Euclidean distance, and ɣ (gamma) is a parameter that controls the width of the Gaussian kernel.

Kernelized Support Vector Machines

- Decision boundary and support vectors found by an SVM with RBF kernel

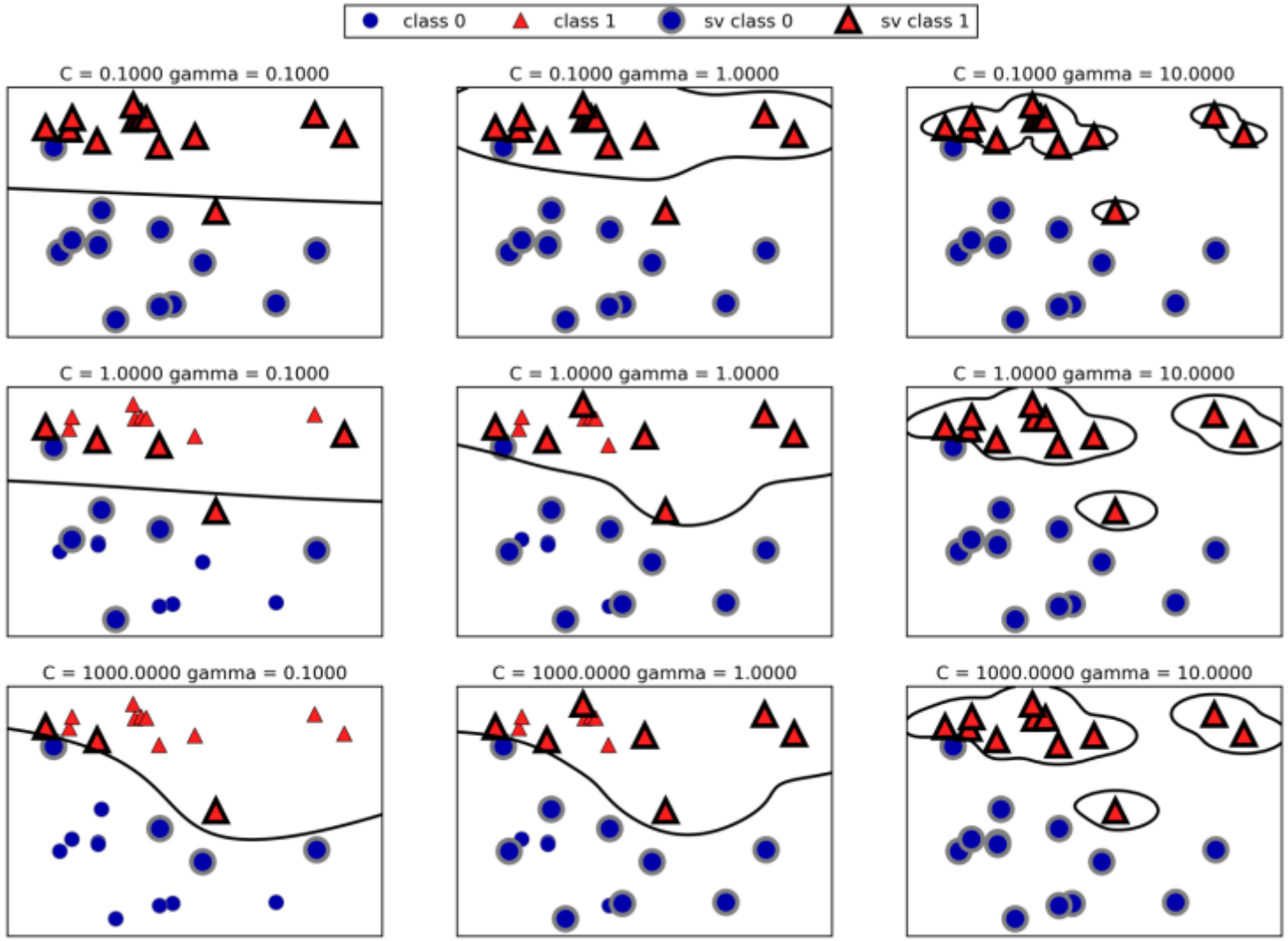


Kernelized Support Vector Machines

- Tuning SVM parameters
- The gamma and C parameter
- Gamma which controls the width of the Gaussian kernel. It determines the scale of what it means for points to be close together.
- The C parameter is a regularization parameter, similar to that used in the linear models. It limits the importance of each point (or more precisely, their dual_coef_).

- https://www.kaggle.com/code/nirajvermafcb/support-vector-machine-detail-analysis/notebook

Kernelized Support Vector Machines



Kernelized Support Vector Machines

- Preprocessing data for SVMs
- Scaling the data made a huge difference
- Kernelized support vector machines are powerful models and perform well on a variety of datasets.
- SVMs allow for complex decision boundaries, even if the data has only a few features.
- They work well on low-dimensional and high-dimensional data (i.e., few and many features), but don’t scale very well with the number of samples.
- Running an SVM on data with up to 10,000 samples might work well, but working with datasets of size 100,000 or more can become challenging in terms of runtime and memory usage.

Kernelized Support Vector Machines

- Another downside of SVMs is that they require careful preprocessing of the data and tuning of the parameters. This is why, these days, most people instead use tree-based models such as random forests or gradient boosting (which require little or no pre- processing) in many applications.
- Furthermore, SVM models are hard to inspect; it can be difficult to understand why a particular prediction was made, and it might be tricky to explain the model to a nonexpert.
- Still, it might be worth trying SVMs, particularly if all of your features represent measurements in similar units (e.g., all are pixel intensities) and they are on similar scales.
- The important parameters in kernel SVMs are the regularization parameter C, the choice of the kernel, and the kernel-specific parameters. RBF -> gamma