Machine Learning
|Support Vector Classifier (SVC)

Prof. Dr. Mostafa Elhosseini

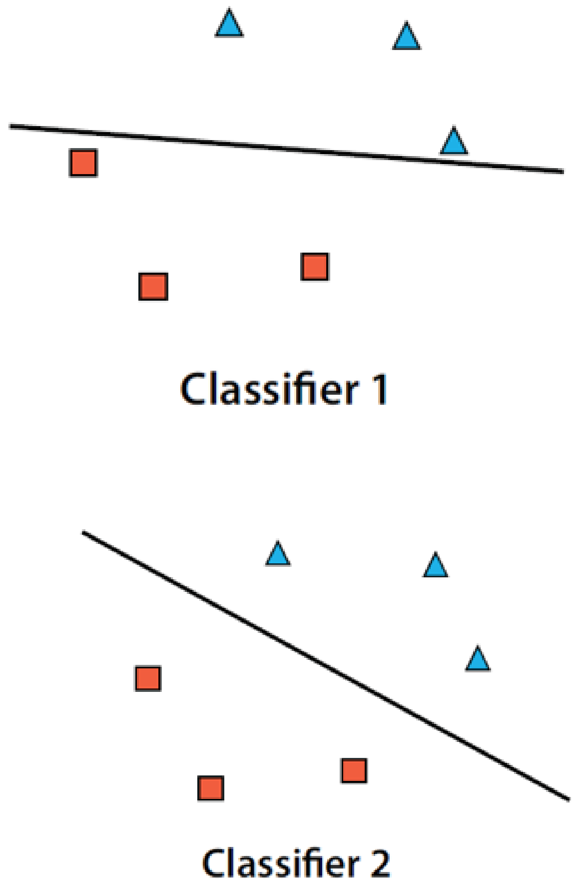
https://youtube.com/ElhosseiniAcademy

Agenda

- Hard Margin
- Large Margin Classification
- Soft Margin
- Support Vector Classifier
- Bias / Variance Tradeoff
- Hyperplanes
- Feature Scaling
- Use Cases

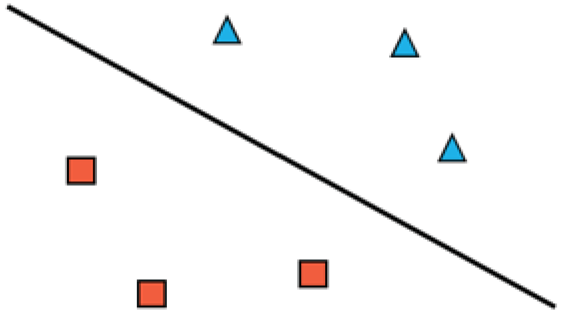
Intro

- An SVM is like a **perceptron**, in that it separates a dataset with two classes using a linear boundary.
- However, the SVM aims to find the linear boundary that is located **as far as possible** from the points in the dataset.
  - Perform as well on new instances
  - Wiggle line (Robust)

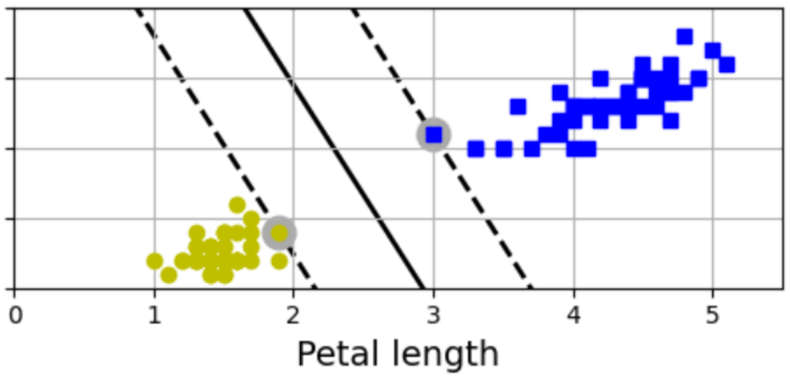


Intro

- We may want to visualize an SVM as the **line** in the middle that tries to stay as far as possible from the points. We can also imagine it as the **two external parallel lines** trying to stay as far away from each other as possible.

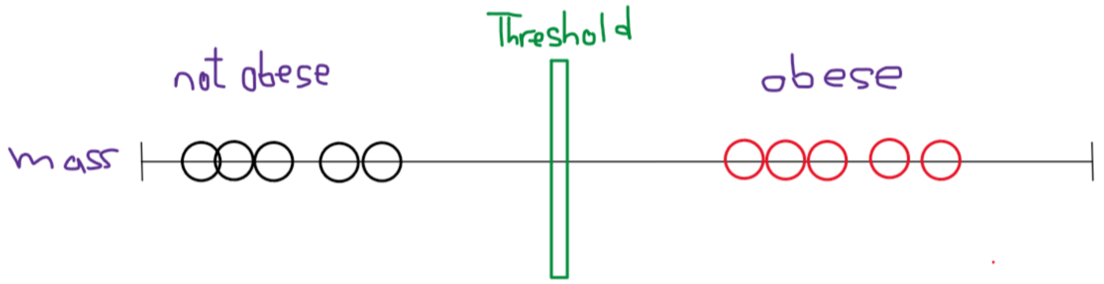


- An SVM classifier uses two parallel lines instead of one line. The goal of the SVM is twofold;
  - It tries to classify the data correctly and also
  - Tries to space the lines as much as possible (fitting the widest possible street)



Large Margin Classification

- You can think of an SVM classifier as fitting the widest possible street between the classes.
- This is called **large margin classification** (**Hard Margin** classification)

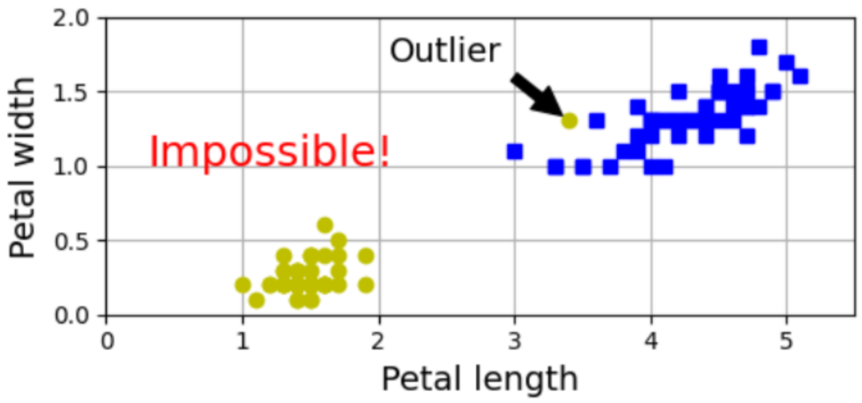


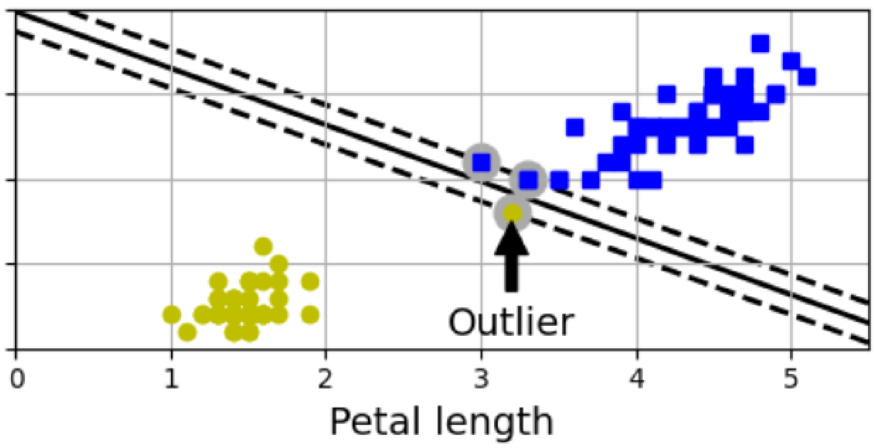
- Shortest distance between the observation and threshold is called **Margin**

- when we use the threshold that gives us the **largest margin** to make classification

Hard Margin Classification

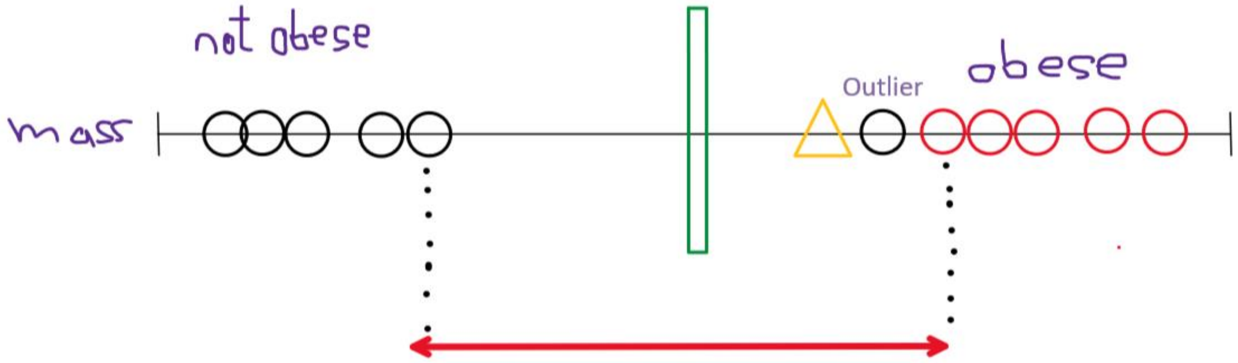
- If we strictly impose that all instances be off the street and on the right side
- There are two main issues with hard margin classification.
  - It only works if the data is linearly separable, and
  - It is quite sensitive to outliers





Soft Margin Classification

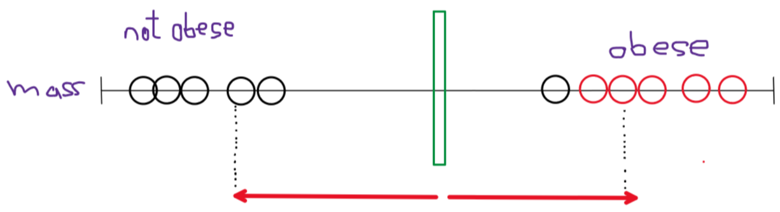
- To make a threshold that is **not so sensitive to outliers**, we must **allow misclassification**

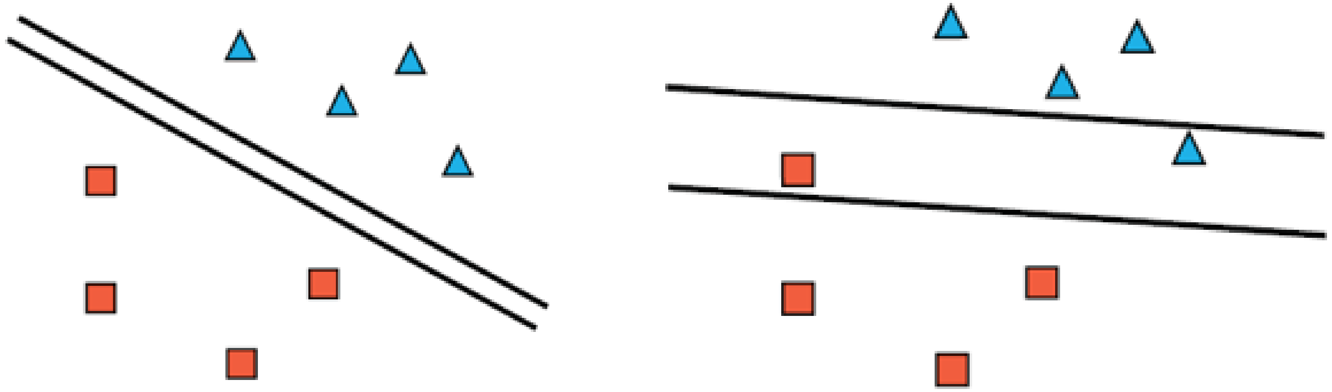


  - Ignore outliers
  - Put the threshold as the midpoint
  - Misclassify this observation “Outlier”

Soft Margin Classification

- The objective is to find a good balance between keeping the street as large as possible and limiting the margin violations
- Violation: Instances that end up in the middle of the street or even on the wrong side).





Soft Margin Classification

- Naming ‘Support Vector Classifier’ comes from observations on the edges and within the soft margin are called support vector.

- Notice that adding more training instances “off the street” will not affect the decision boundary at all: it is fully determined (or “supported”) by the instances located on the edge of the street. These instances are called the support vectors

Soft Margin Classification

- Naming ‘Support Vector Classifier’ comes from observations on the edges and within the soft margin are called support vector.

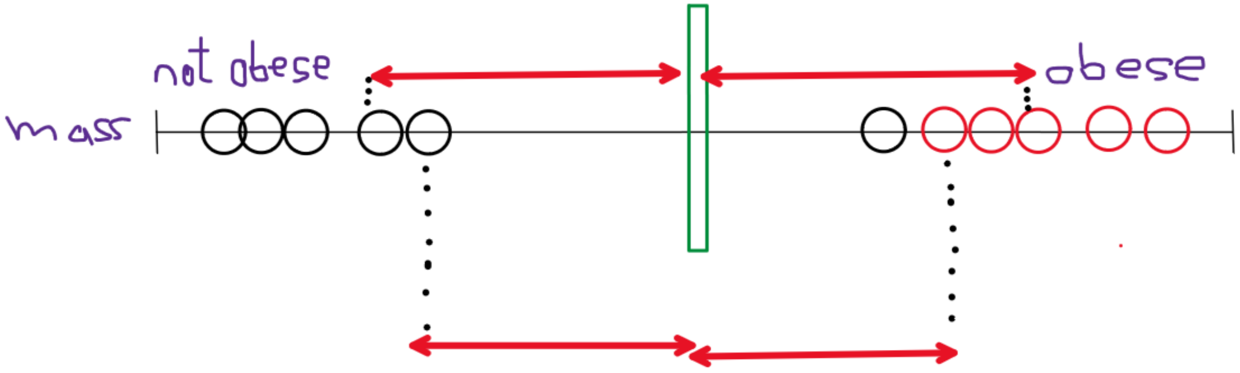
- Notice that adding more training instances “off the street” will not affect the decision boundary at all: it is fully determined (or “supported”) by the instances located on the edge of the street. These instances are called the support vectors

Bias / Variance Tradeoff

- Choosing a threshold that allows misclassification is an example of the bias/variance tradeoff that plagues all of the machine learning
- Before we allowed misclassification – we picked a threshold that was very sensitive to the training data “Low Bias”
- As it performed poorly when we got new data “High variance”
- In contrast
- When we picked a threshold that was less sensitive to the training data and allowed misclassification “Higher bias”
- It performed better when we got new data “Low variance”

Cross validation

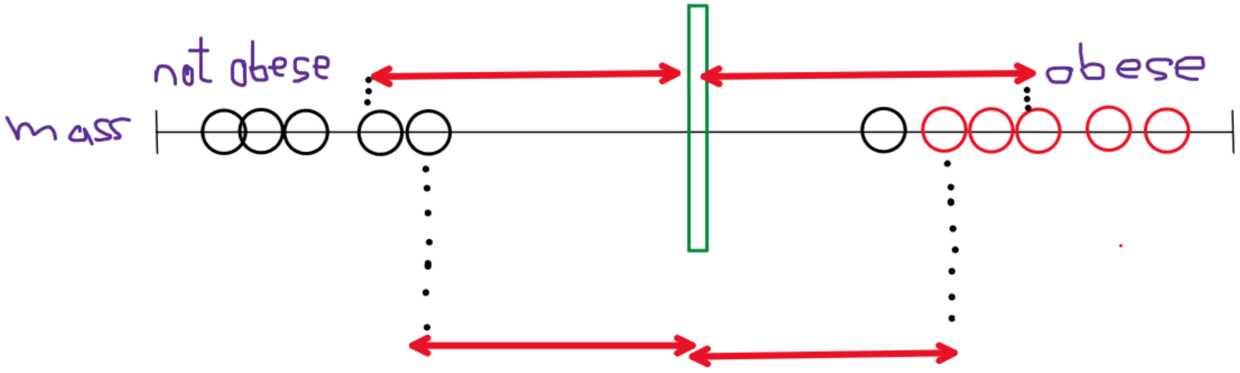
- How do we know that this soft margin is better than this soft margin



- Use cross validation to determine how many misclassification and observations to allow inside the soft margin to get the best classification

Cross validation

- How do we know that this soft margin is better than this soft margin



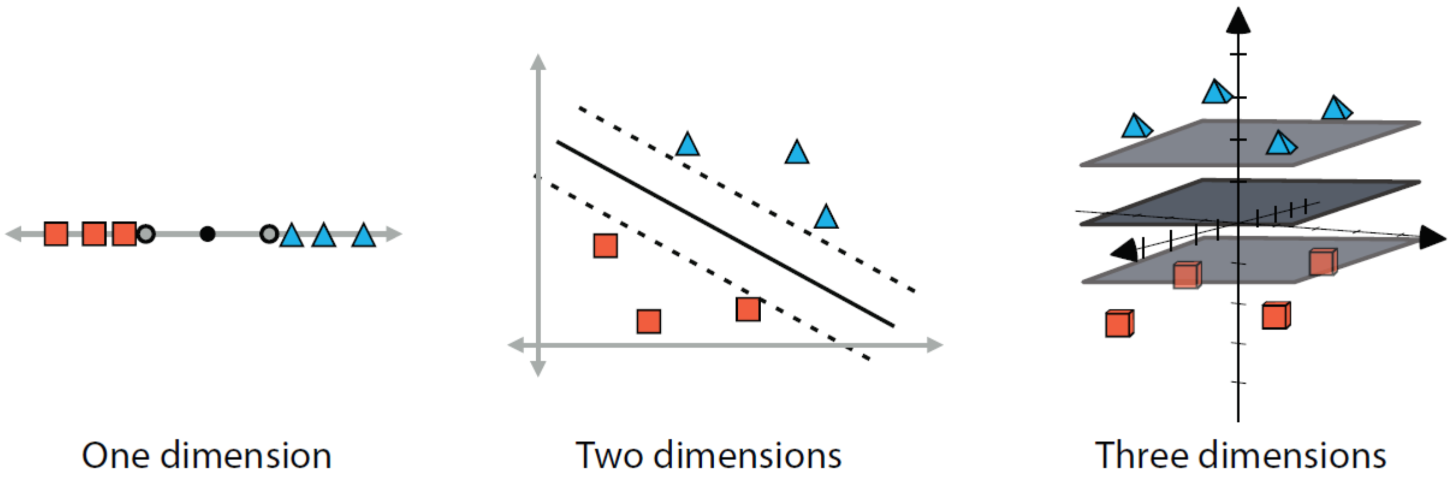
- Use cross validation to determine how many misclassification and observations to allow inside the soft margin to get the best classification

Hyperplanes

- The linear boundaries in one dimension are points and in two dimensions are lines.
- Likewise, the linear boundaries in three dimensions are planes, and in higher dimensions, they are hyperplanes of one dimension less than the space in which the points live.
- Use this term “Hyperplane” when we can not draw it on paper

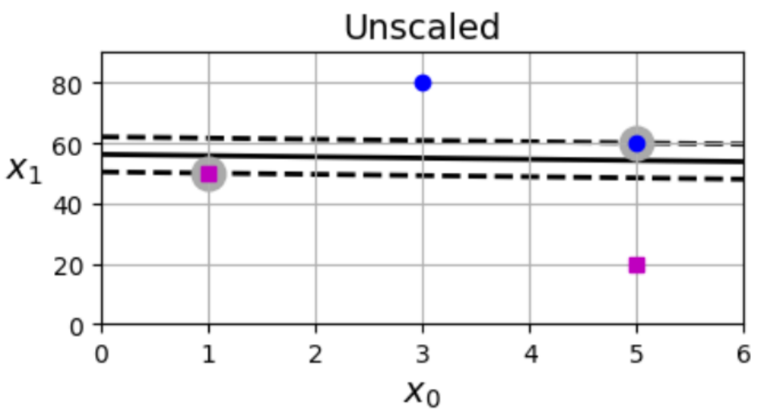
Hyperplanes

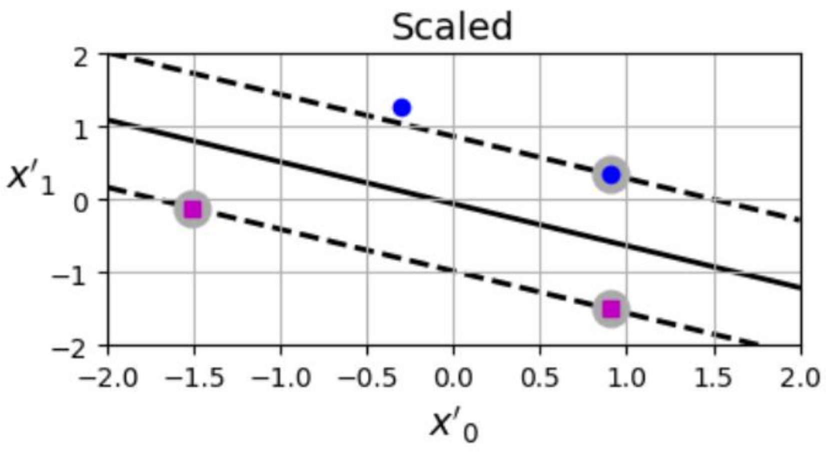
- In each of these cases, we try to find the boundary that is the farthest from the points.



Feature Scaling

- SVMs are sensitive to the feature scales
- the vertical scale is much larger than the horizontal scale, so the widest possible street is close to horizontal
- After feature scaling (e.g., using Scikit-Learn's StandardScaler), the decision boundary looks much better





Use Cases

- A support vector machine (SVM) is a powerful and versatile machine learning model, capable of performing linear or nonlinear classification, regression, and even novelty detection.

- SVMs shine with small to medium-sized nonlinear datasets (i.e., hundreds to thousands of instances), especially for classification tasks.

- However, they don’t scale very well to very large datasets

Error Function

- What do we want the model to achieve?
  - Each of the two lines should classify the points as best as possible.
  - The two lines should be as far away from each other as possible.
- It tries to maximize two things at the same time: the classification of the points and the distance between the lines.
- The error function should penalize any model that doesn’t achieve these things
- Error = Classification Error + Distance Error
- Next Lecture