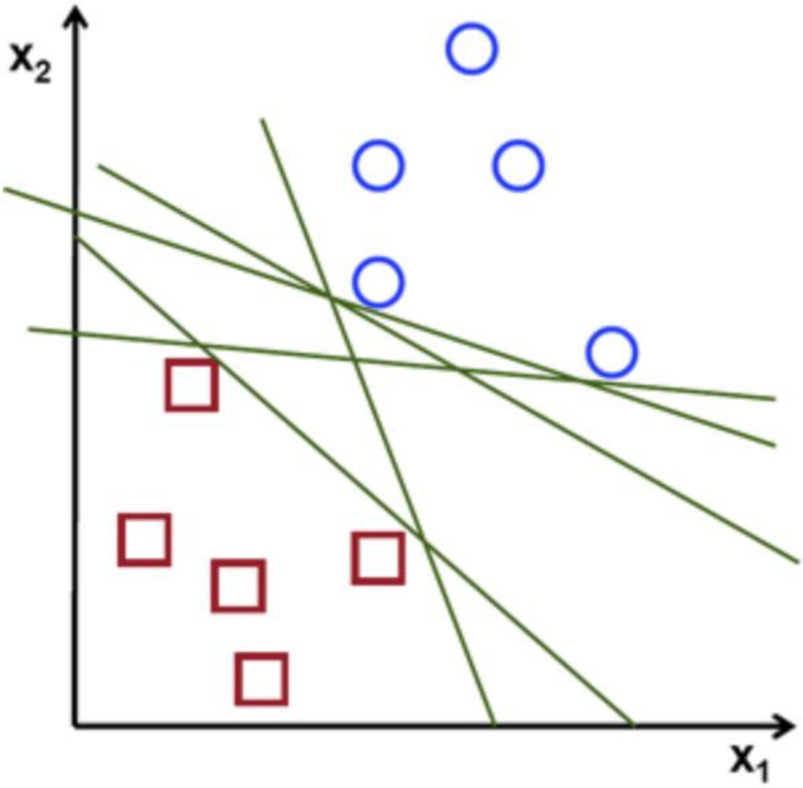
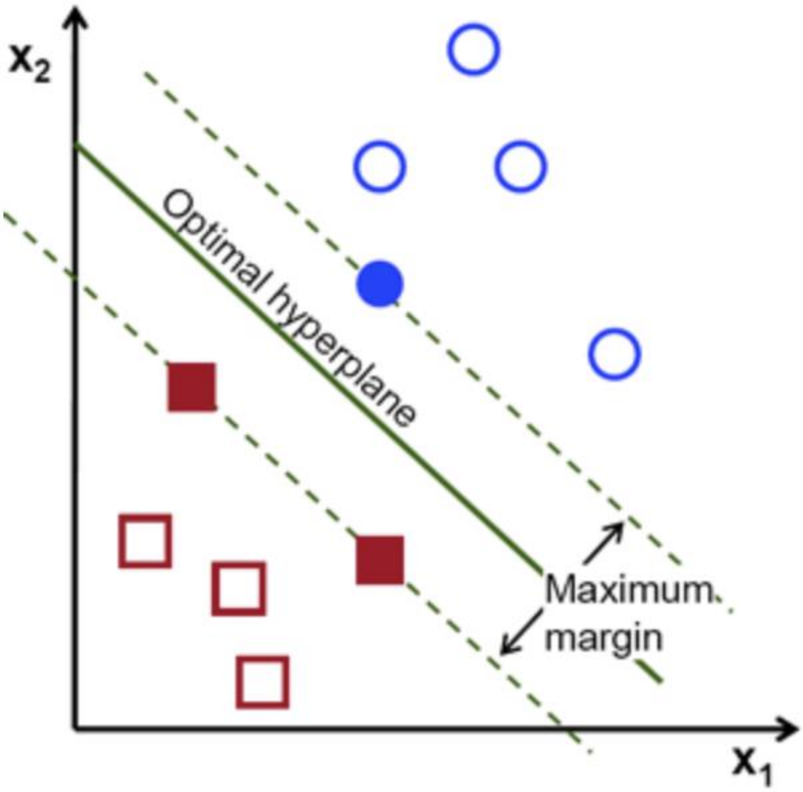
Support Vector Machine

2-D space (2 independent variables)



The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of independent variables) that distinctly classifies the data points.

In other words, find the best line that separates the 2 classes.



To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes.

Which hyperplane to pick?

- Which points should influence optimality?

- only points close to decision boundary

- Support vectors are the elements of the training set that would change the position of the dividing hyperplane if removed.
- Support vectors are the critical elements of the training set
- The problem of finding the optimal hyper plane is an optimization problem and can be solved by optimization techniques

<u>Input</u>: set of (input, output) training pair samples; call the input sample features x_1, x_2...x_n, and the output result y. Typically, there can be lots of input features x_i.

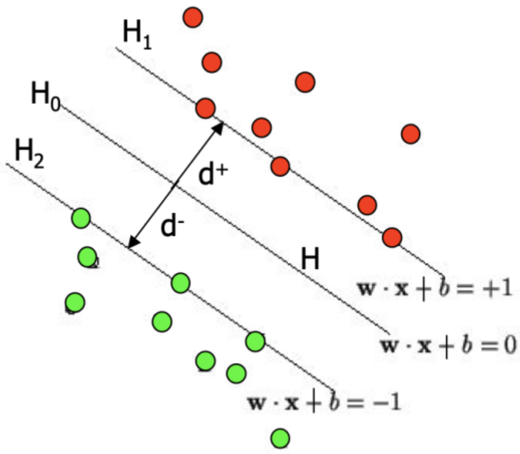
<u>Output</u>: set of weights **w** (or w_i), one for each feature, whose linear combination predicts the value of y

Definitions

Define the hyperplanes H such that:

$w \bullet x_i + b \geq +1$ when $y_i = +1$

$w \bullet x_i + b \leq -1$ when $y_i = -1$



H_1 and H_2 are the planes:

H_1: $w \bullet x_i + b = +1$

H_2: $w \bullet x_i + b = -1$

The points on the planes H_1 and H_2 are the tips of the Support Vectors

The plane H_0 is the median in between, where $w \bullet x_i + b = 0$

d+ = the shortest distance to the closest positive point

d- = the shortest distance to the closest negative point

The margin (gutter) of a separating hyperplane is d+ + d–.

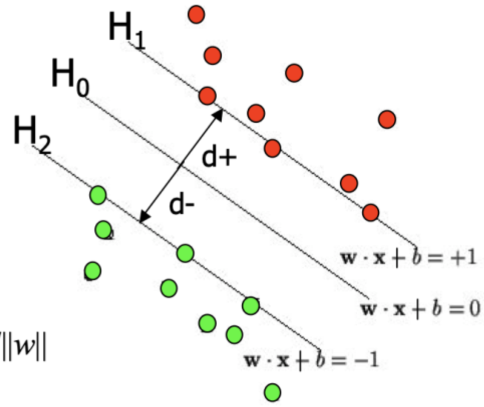
– **w is a weight vector**

– **x is input vector**

– **b is bias**

Maximizing the margin (aka street width)

We want a classifier (linear separator) with as big a margin as possible.



Recall the distance from a point(x_0,y_0) to a line: $Ax+By+c = 0$ is: $|Ax_0 + By_0 + c|/\text{sqrt}(A^2+B^2)$, so,
The distance between H_0 and H_1 is then:
$|w\bullet x+b|/||w||=1/||w||$, so
The total distance between H_1 and H_2 is thus: $2/||w||$

In order to maximize the margin, we thus need to minimize $||w||$. With the condition that there are no datapoints between H_1 and H_2:

$$\left.\begin{array}{l} \mathbf{x}_i\bullet\mathbf{w}+b \geq +1 \text{ when } y_i = +1 \\ \mathbf{x}_i\bullet\mathbf{w}+b \leq -1 \text{ when } y_i = -1 \end{array}\right\}$$

Can be combined into: $y_i(\mathbf{x}_i\bullet\mathbf{w}) \geq 1$

Define classes:

If $x_i * w + b \leq -1$ assign to class green
If $x_i * w + b \geq 1$ assign to class red

Multi-Class Classification

'*red*' '*blue*' and '*green*'

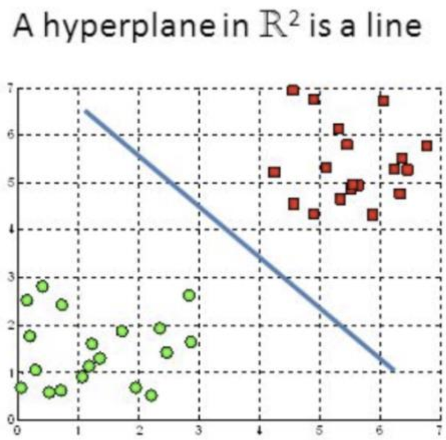
One-Vs-Rest

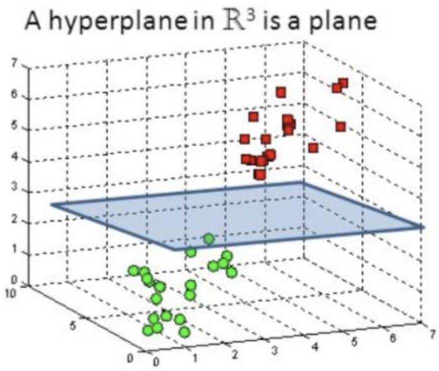
- **Binary Classification Problem 1**: red vs [blue, green]
- **Binary Classification Problem 2**: blue vs [red, green]
- **Binary Classification Problem 3**: green vs [red, blue]

One-Vs-One

- **Binary Classification Problem 1**: red vs. blue
- **Binary Classification Problem 2**: red vs. green
- **Binary Classification Problem 3**: blue vs. green

More than 2 independent variables





The dimension of the hyperplane depends upon the number of independent variables (predictors). If the number of inputs IV is 2, then the hyperplane is just a line. If the number of inputs IV is 3, then the hyperplane becomes a two-dimensional plane. It becomes difficult to imagine when the number of features exceeds 3.

References:

https://www.kaggle.com/faressayah/support-vector-machine-pca-tutorial-for-beginner

http://web.mit.edu/6.034/wwwbob/svm-notes-long-08.pdf

https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/