



Support Vector Machines

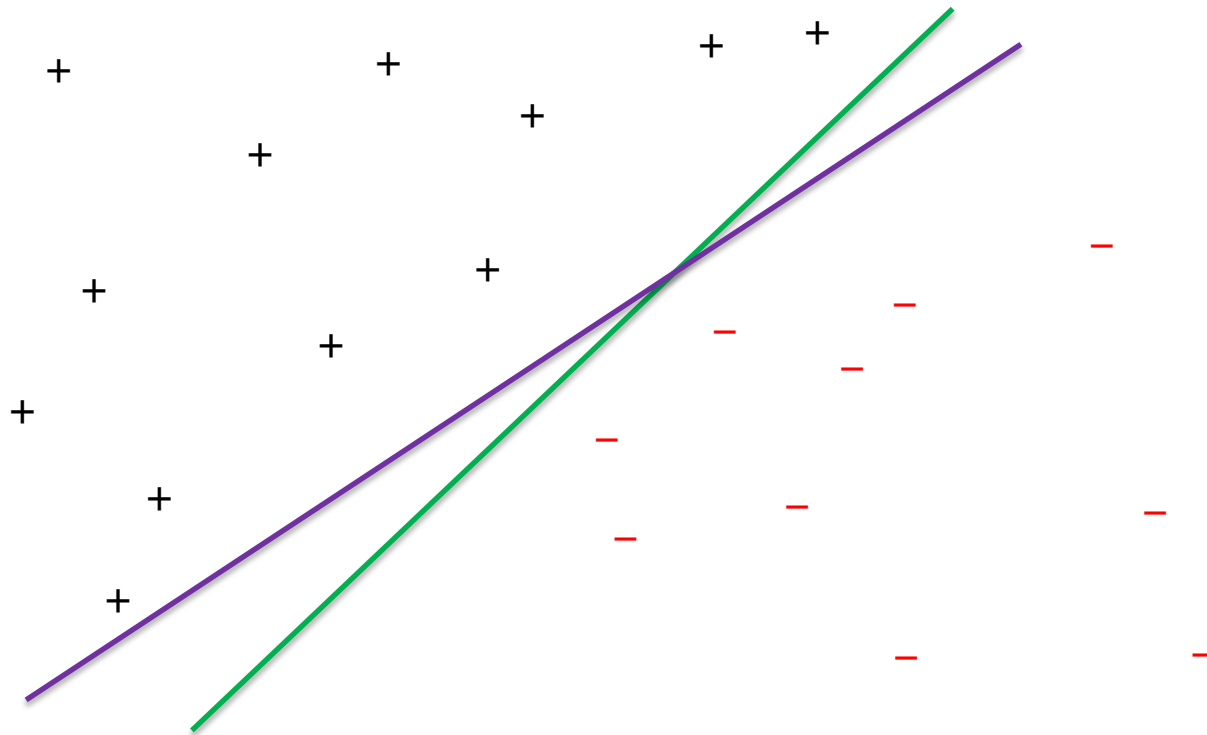
Nicholas Ruozzi

University of Texas at Dallas

Support Vector Machines



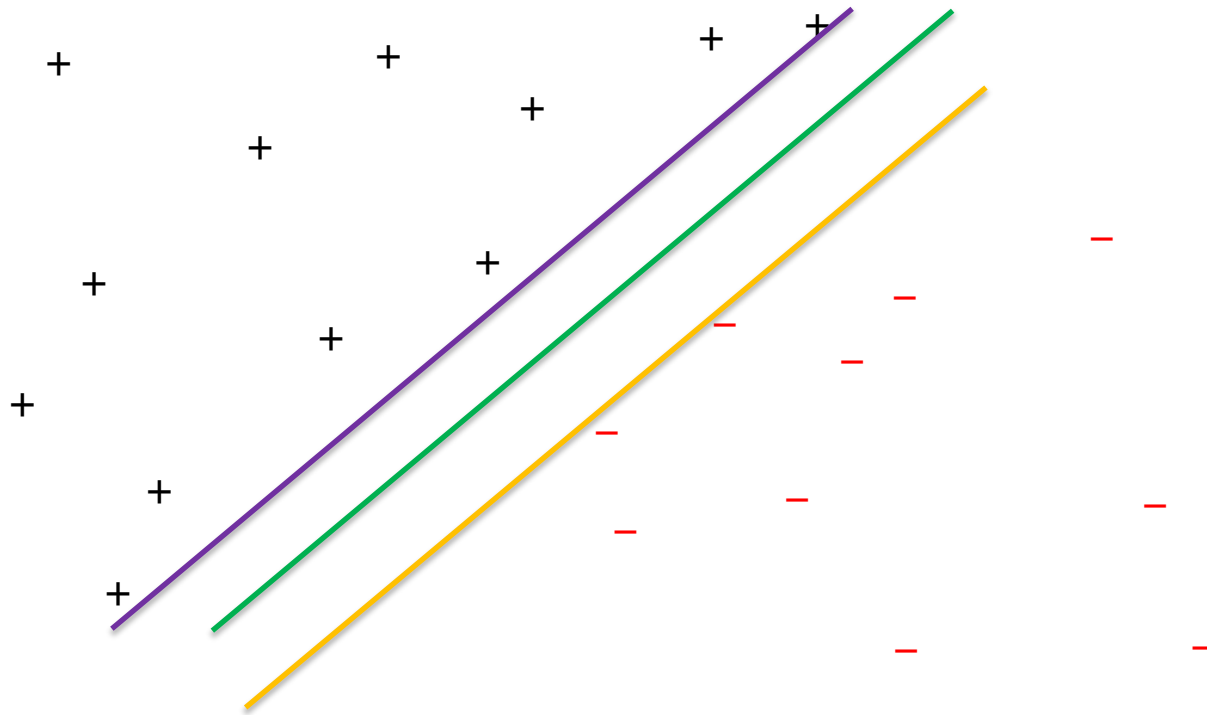
- How can we decide between perfect classifiers?



Support Vector Machines



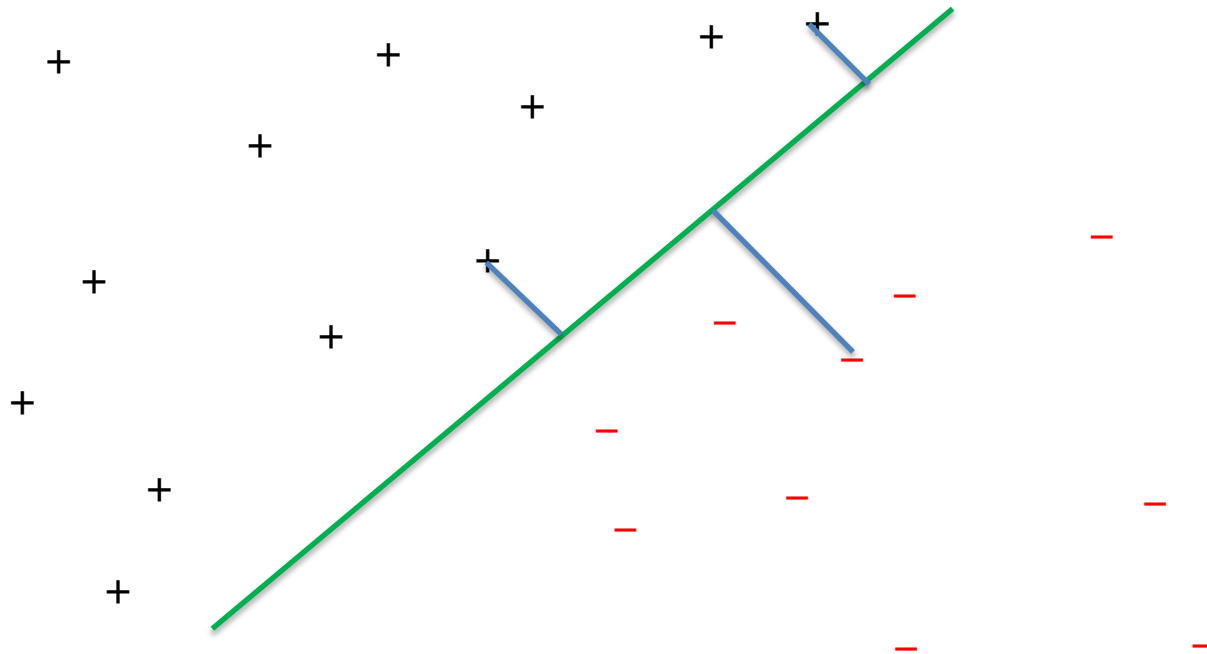
- How can we decide between perfect classifiers?



Support Vector Machines



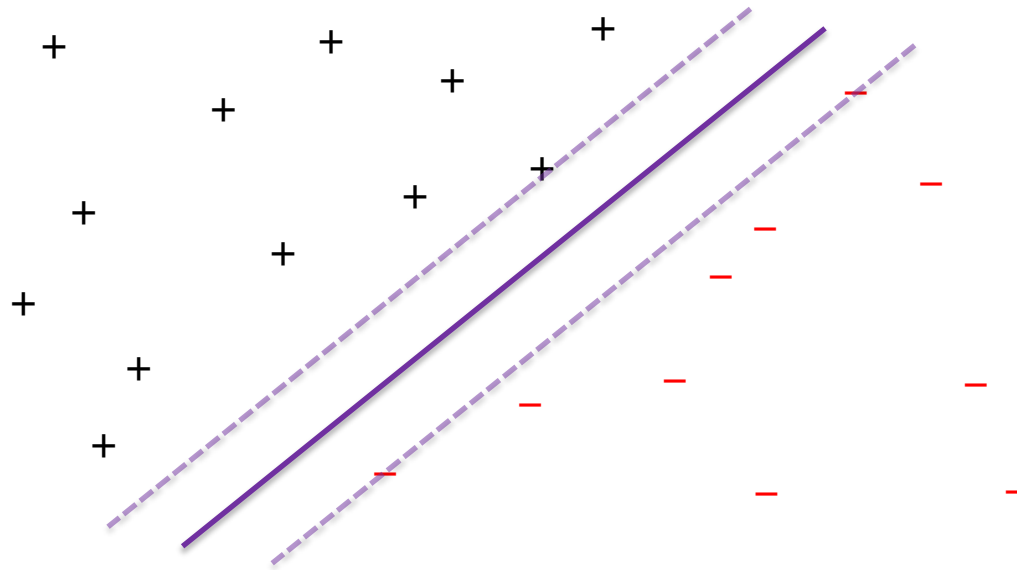
- Define the **margin** to be the distance of the closest data point to the classifier



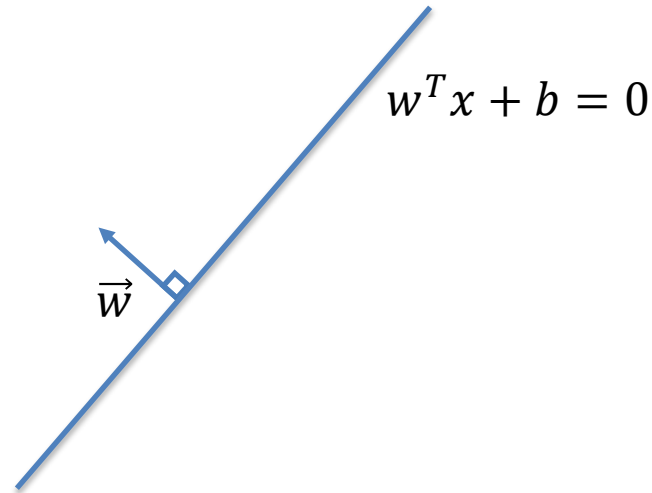
Support Vector Machines



- Support vector machines (SVMs)



- Choose the classifier with the largest margin
 - Has good practical and theoretical performance



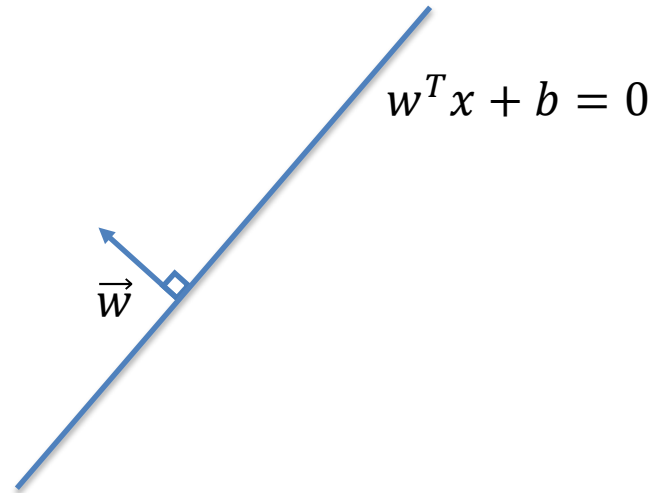
- In n dimensions, a hyperplane is a solution to the equation

$$w^T x + b = 0$$

with $w \in \mathbb{R}^n, b \in \mathbb{R}$

- The vector w is sometimes called the normal vector of the hyperplane

Some Geometry



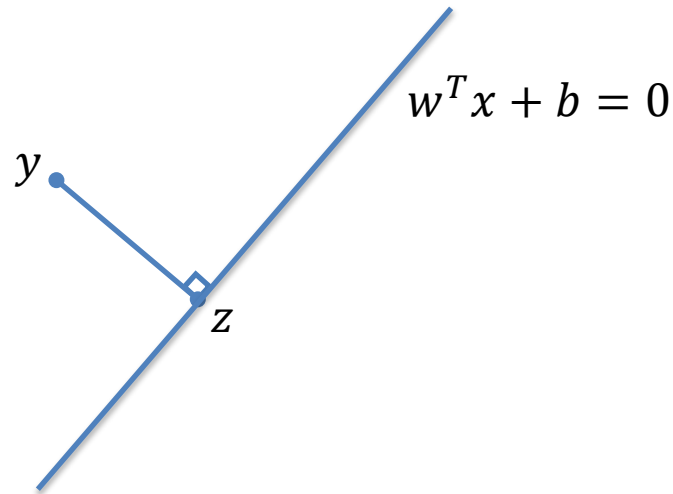
- In n dimensions, a hyperplane is a solution to the equation

$$w^T x + b = 0$$

- Note that this equation is scale invariant for any scalar c

$$c \cdot (w^T x + b) = 0$$

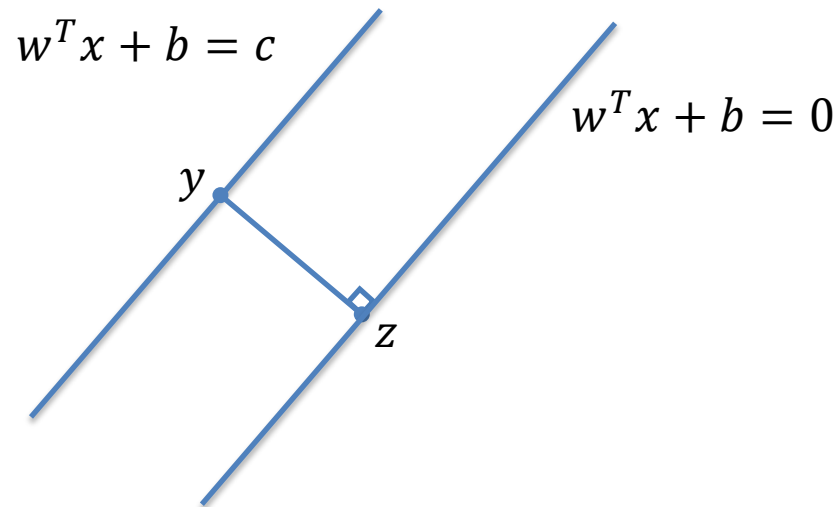
Some Geometry



- The distance between a point y and a hyperplane $w^T x + b = 0$ is the length of the segment perpendicular to the line to the point y
- The vector from y to z is given by

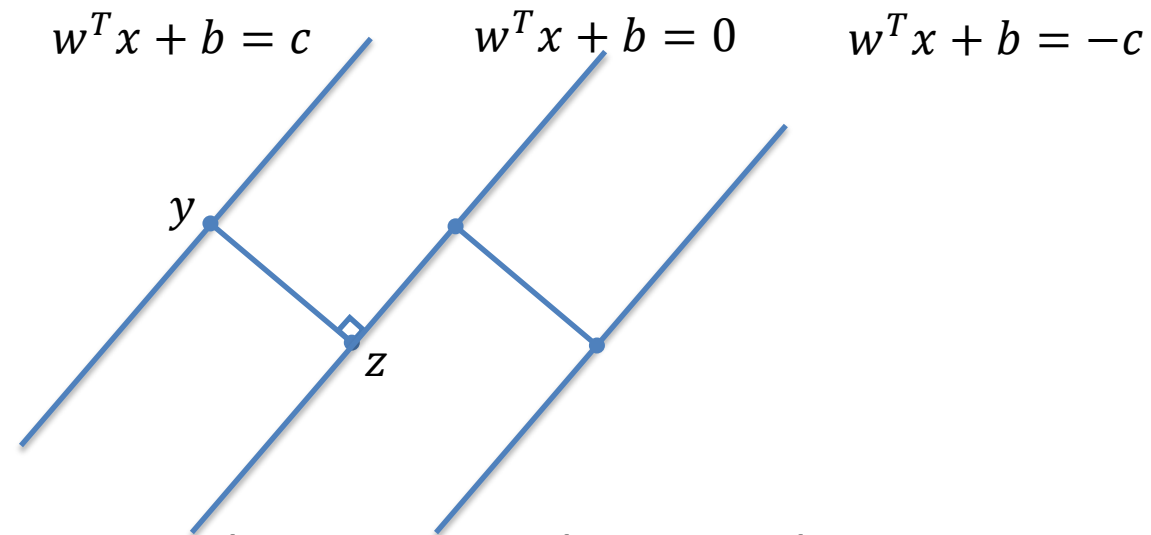
$$y - z = \|y - z\| \frac{w}{\|w\|}$$

Scale Invariance



- By scale invariance, we can assume that $c = 1$
- The maximum margin is always attained by choosing $w^T x + b = 0$ so that it is equidistant from the closest data point classified as +1 and the closest data point classified as -1

Scale Invariance

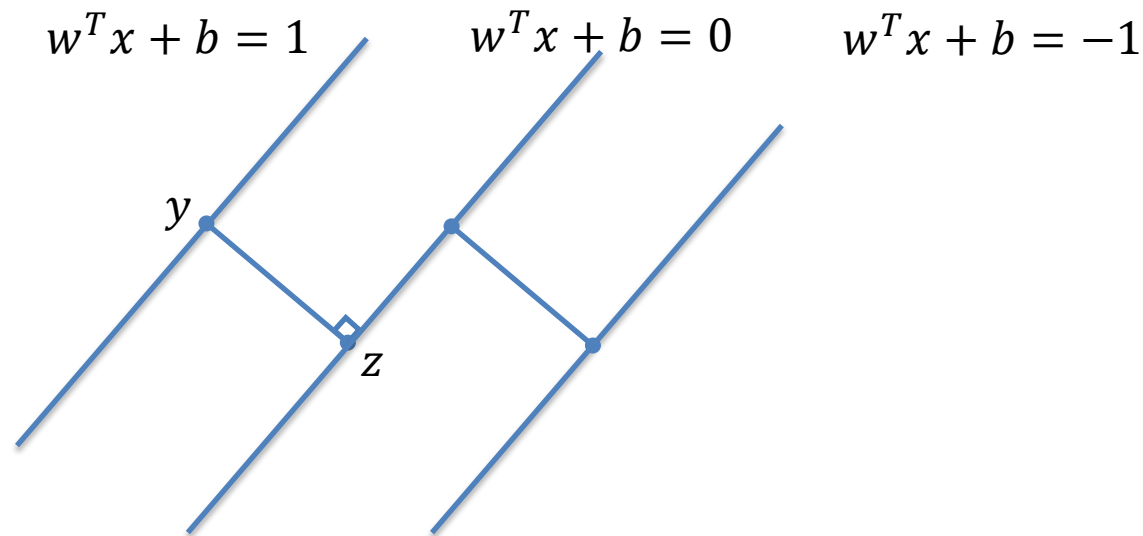


- We want to maximize the margin subject to the constraints that

$$y^{(i)}(w^T x^{(i)} + b) \geq 1$$

- But how do we compute the size of the margin?

Some Geometry



Putting it all together

$$y - z = \|y - z\| \frac{w}{\|w\|}$$

and

$$\begin{aligned} w^T y + b &= 1 \\ w^T z + b &= 0 \end{aligned}$$



$$w^T (y - z) = 1$$

and

$$w^T (y - z) = \|y - z\| \|w\|$$

which gives

$$\|y - z\| = 1 / \|w\|$$

- This analysis yields the following optimization problem

$$\max_{w,b} \frac{1}{\|w\|}$$

such that

$$y^{(i)}(w^T x^{(i)} + b) \geq 1, \text{ for all } i$$

- Or, equivalently,

$$\min_{w,b} \|w\|^2$$

such that

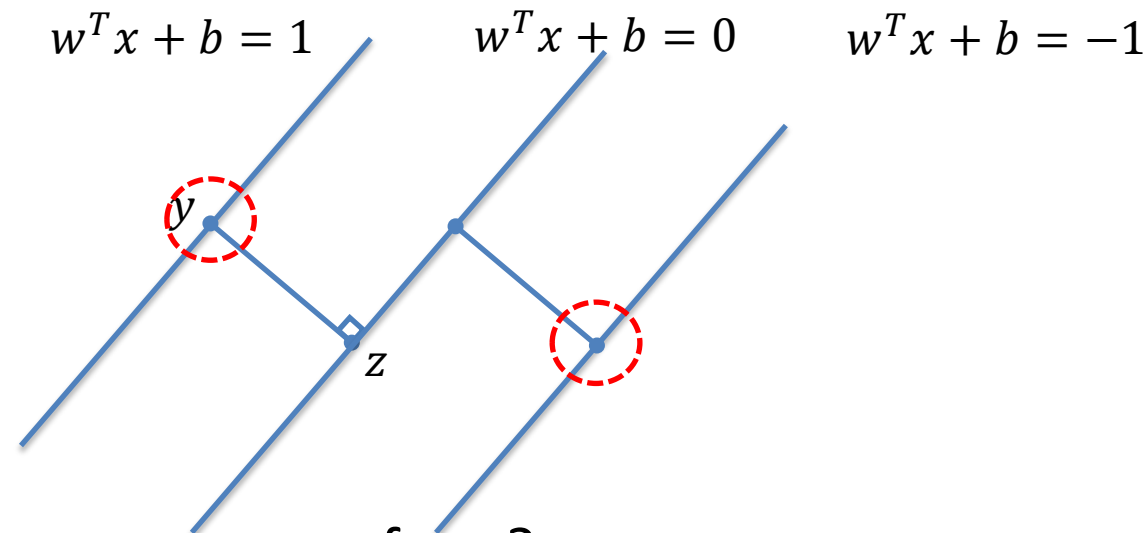
$$y^{(i)}(w^T x^{(i)} + b) \geq 1, \text{ for all } i$$

$$\min_{w,b} \|w\|^2$$

such that

$$y^{(i)}(w^T x^{(i)} + b) \geq 1, \text{ for all } i$$

- This is a standard quadratic programming problem
 - Falls into the class of **convex optimization problems**
 - Can be solved with many specialized optimization tools (e.g., `quadprog()` in MATLAB)

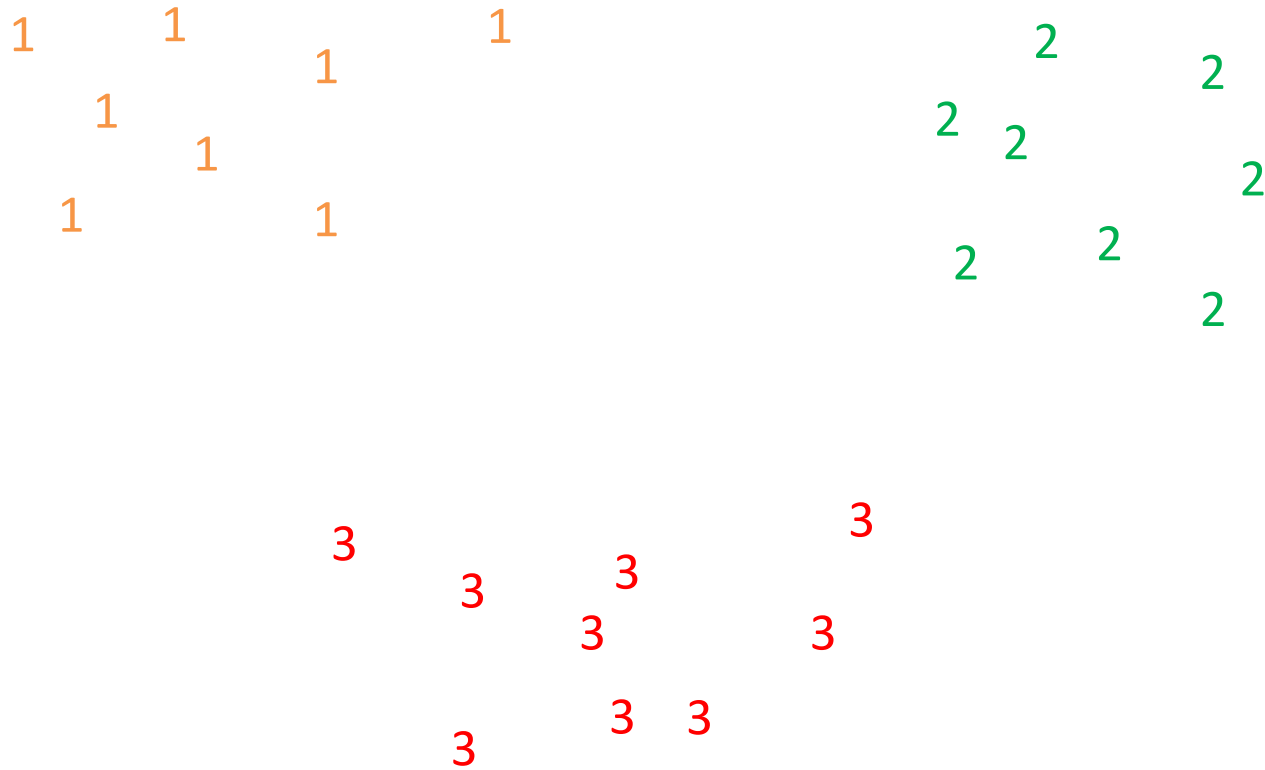


- Where does the name come from?
 - The set of all data points such that $y^{(i)}(w^T x^{(i)} + b) = 1$ are called **support vectors**
 - The SVM classifier is completely determined by the support vectors (you could delete the rest of the data and get the same answer)

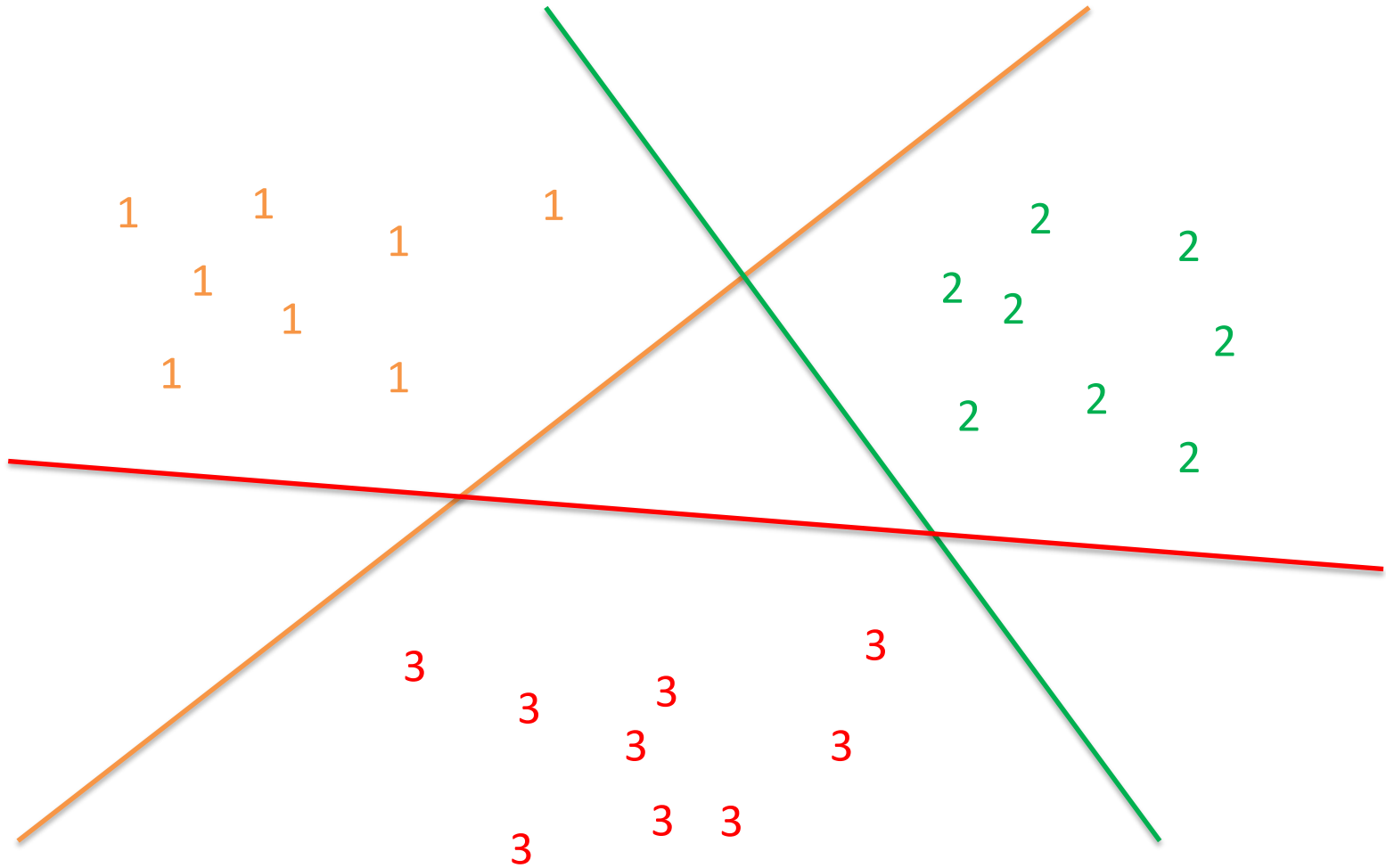
- What if the data isn't linearly separable?
- What if we want to do more than just binary classification (i.e., if $y \in \{1,2,3\}$)?

- What if the data isn't linearly separable?
 - Use feature vectors
 - Relax the constraints (coming soon)
- What if we want to do more than just binary classification (i.e., if $y \in \{1,2,3\}$)?

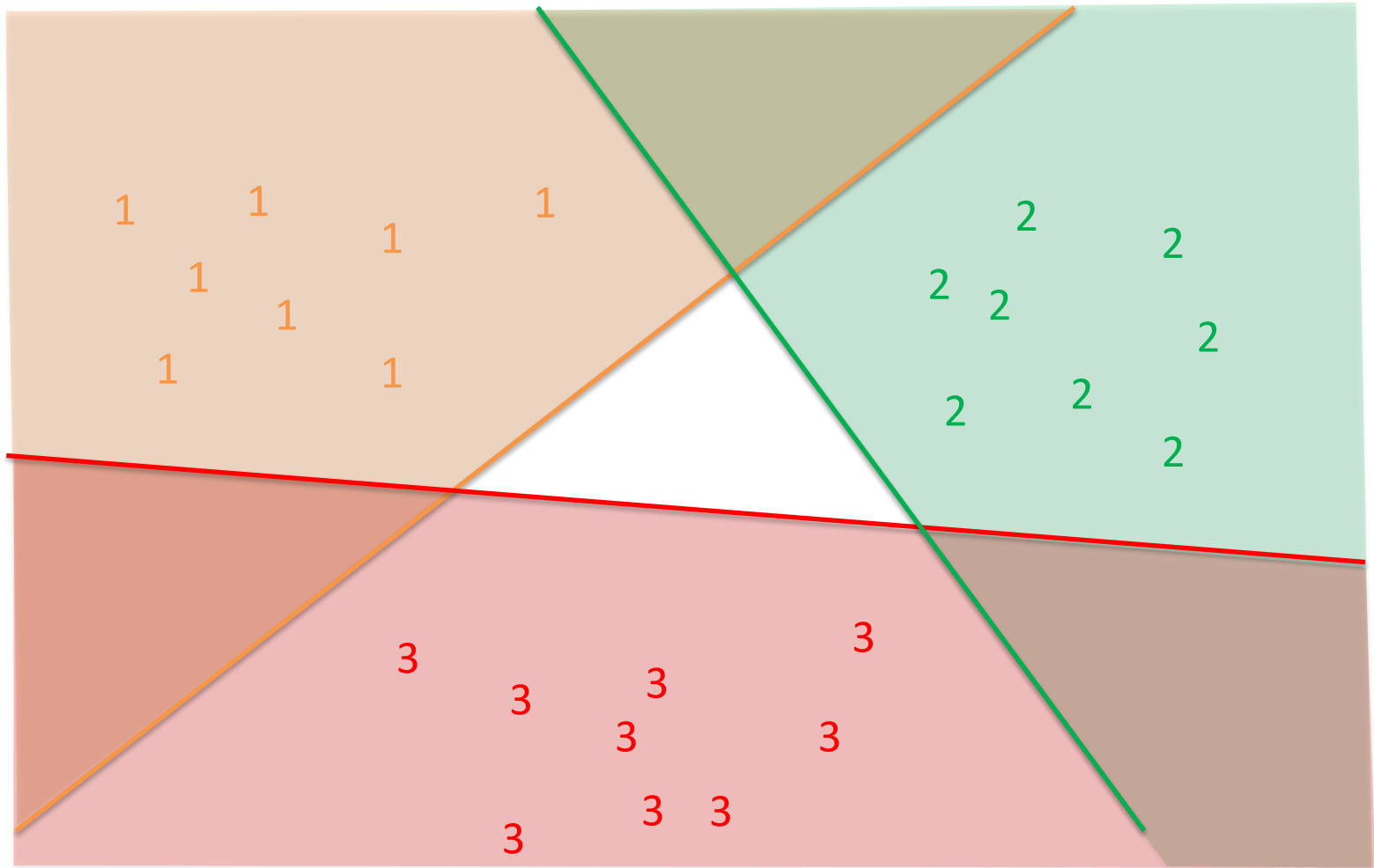
Multiclass Classification



One-Versus-All SVMs



One-Versus-All SVMs



Regions correctly classified by exactly one classifier

- Compute a classifier for each label versus the remaining labels (i.e., an SVM with the selected label as plus and the remaining labels changed to minuses)

- Let $f^k(x) = w^{(k)T}x + b^{(k)}$ be the classifier for the k^{th} label

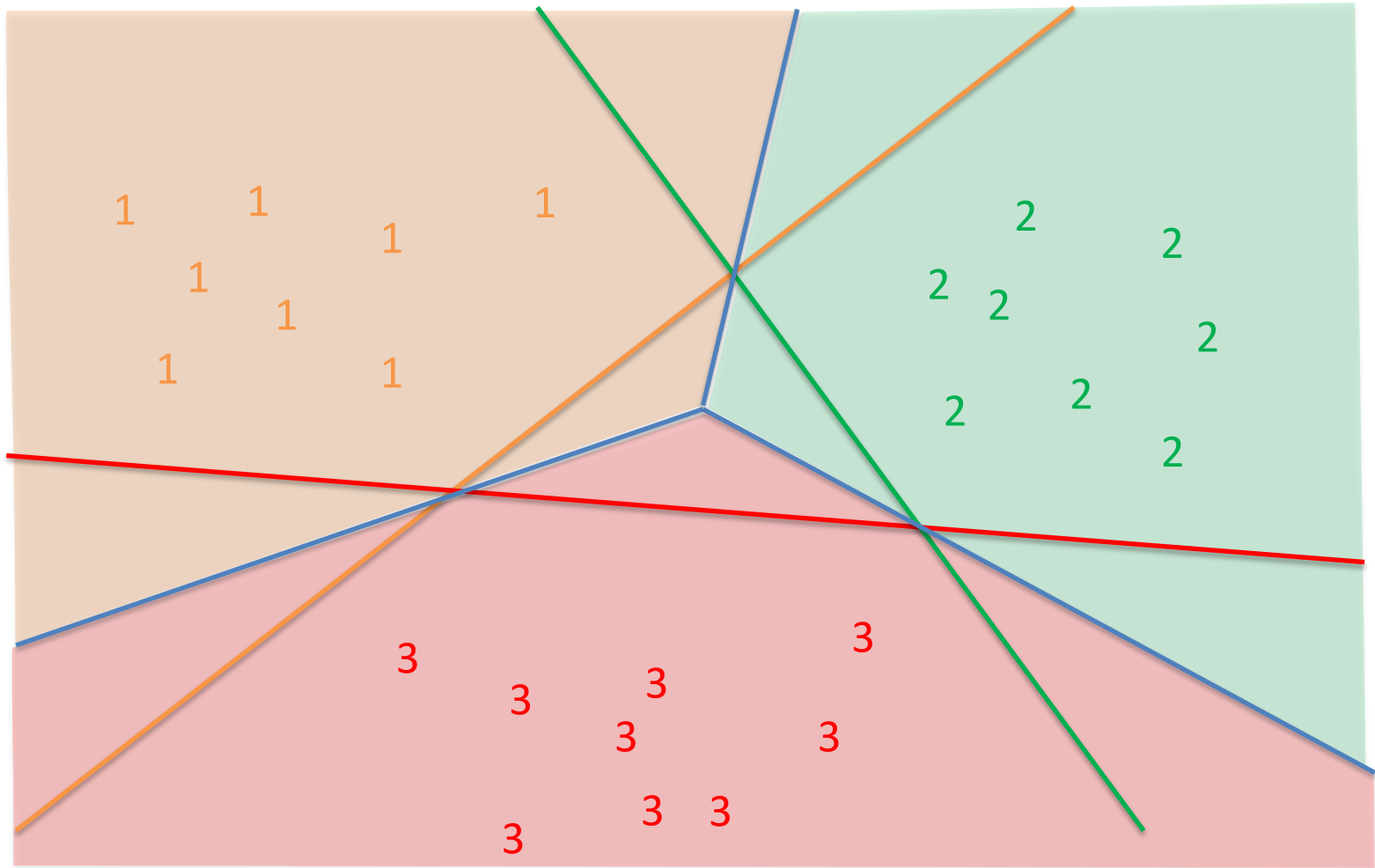
- For a new datapoint x , classify it as

$$k' \in \operatorname{argmax}_k f^k(x)$$

- Drawbacks:

- If there are L possible labels, requires learning L classifiers over the entire data set

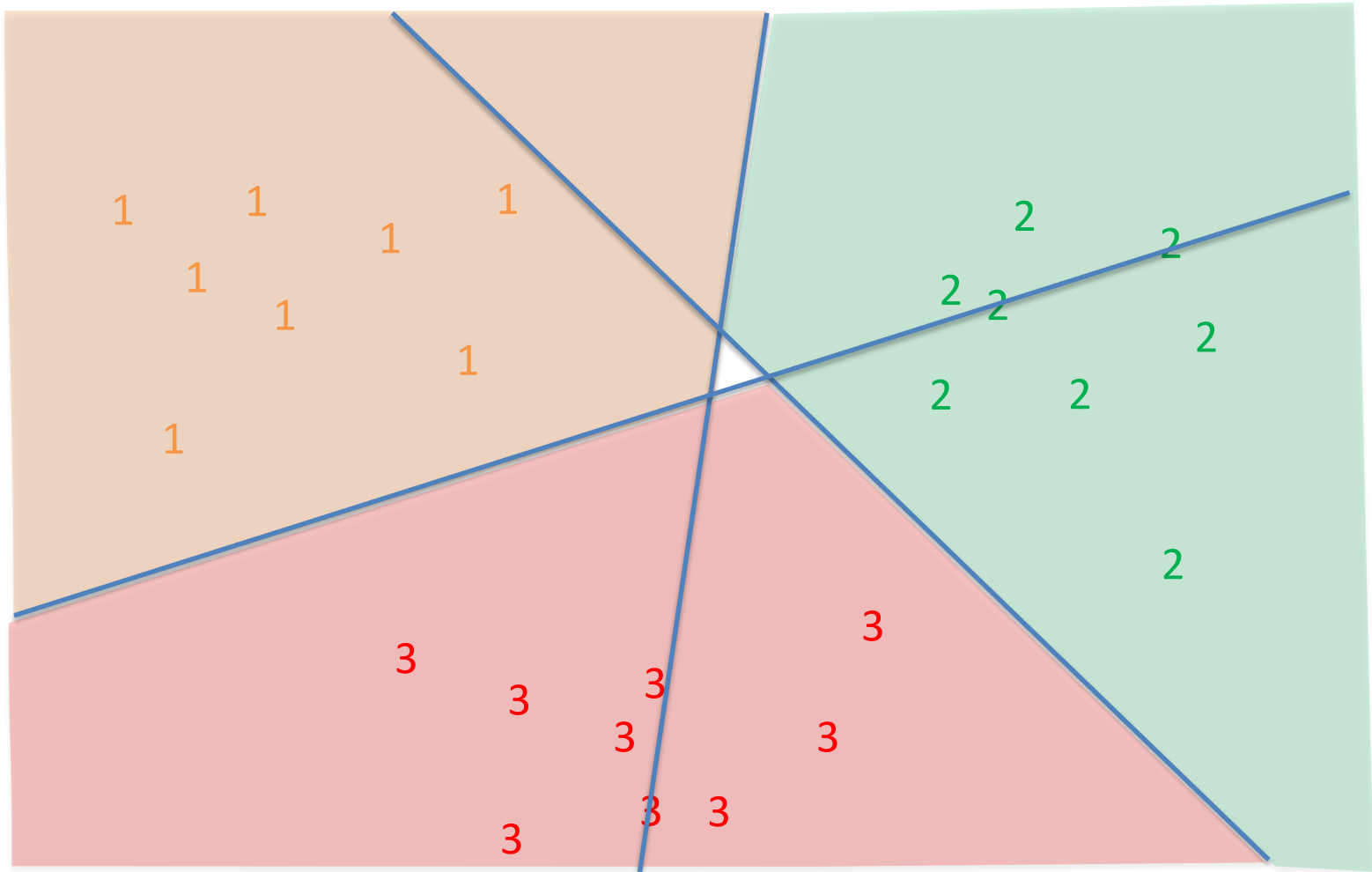
One-Versus-All SVMs



Regions in which points are classified by highest value of $w^T x + b$

- Alternative strategy is to construct a classifier for all possible pairs of labels
- Given a new data point, can classify it by majority vote (i.e., find the most common label among all of the possible classifiers)
- If there are L labels, requires computing $\binom{L}{2}$ different classifiers each of which uses only a fraction of the data
- Drawbacks: Can overfit if some pairs of labels do not have a significant amount of data (plus it can be computationally expensive)

One-Versus-One SVMs



Regions determined by majority vote over the classifiers