Machine Learning

CS 539

Worcester Polytechnic Institute
Department of Computer Science
Instructor: Prof. Kyumin Lee

Upcoming Schedule

- HW4
  - https://canvas.wpi.edu/courses/58900/assignments/357384
  - Due date is July 16

- Online Quiz3 will be taken on July 9
  - Coverage: from neural network to SVM

Support Vector Machines & Kernels

Doing *really* well with linear decision surfaces

Strengths of SVMs

- Good generalization
  - in theory
  - in practice
- Works well with few training instances
- Find globally best model
- Efficient algorithms
- Amenable to the kernel trick

Minor Notation Change

To better match notation used in SVMs

...and to make matrix formulas simpler

We will drop using superscripts for the i^{th} instance

i^{th} instance $\boldsymbol{x}^{(i)} \rightarrow \mathbf{x}_i$

Bold denotes vector

i^{th} instance label $y^{(i)} \rightarrow y_i$

Non-bold denotes scalar

j^{th} feature of i^{th} instance $x_j^{(i)} \rightarrow x_{ij}$

Linear Separators

- Training instances

$$\mathbf{x} \in \mathbb{R}^{d+1}, x_0 = 1$$

$$y \in \{-1, 1\}$$

- Model parameters

$$\boldsymbol{\theta} \in \mathbb{R}^{d+1}$$

- Hyperplane

$$\boldsymbol{\theta}^\intercal \mathbf{x} = \langle \boldsymbol{\theta}, \mathbf{x} \rangle = 0$$

- Decision function

$$h(\mathbf{x}) = \text{sign}(\boldsymbol{\theta}^\intercal \mathbf{x}) = \text{sign}(\langle \boldsymbol{\theta}, \mathbf{x} \rangle)$$

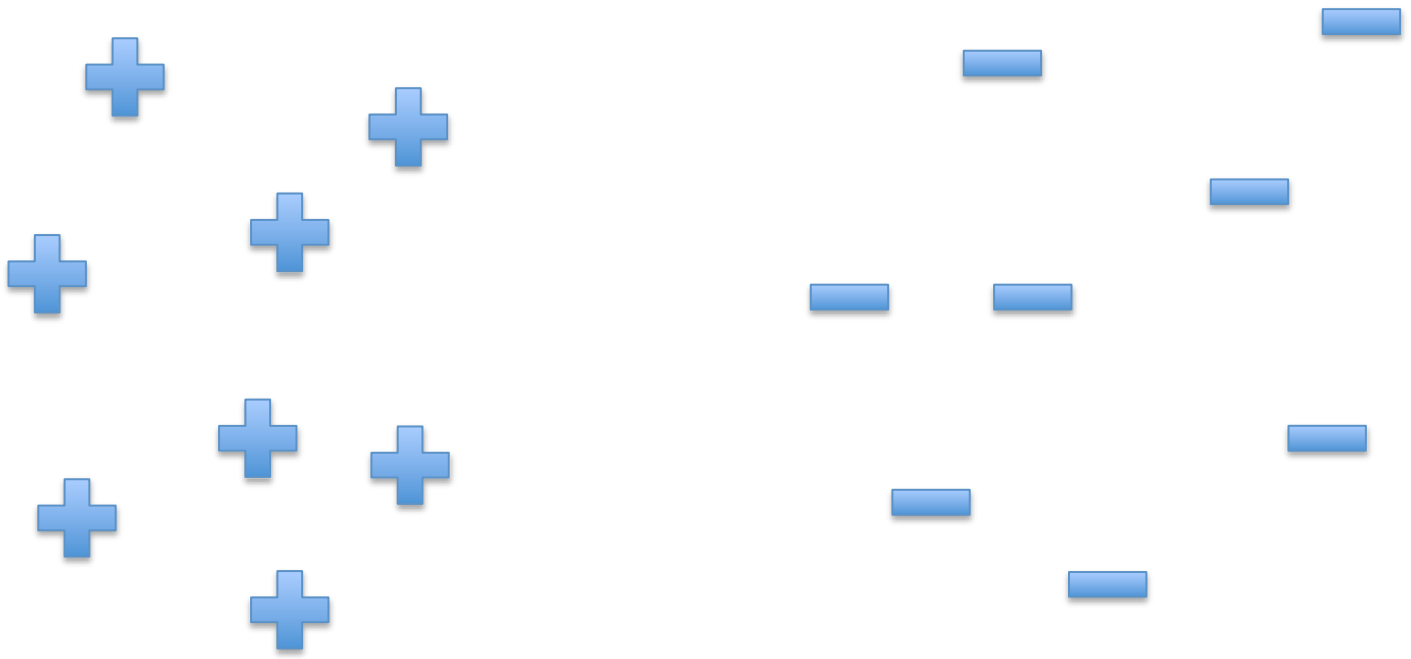
Recall:

Inner (dot) product:

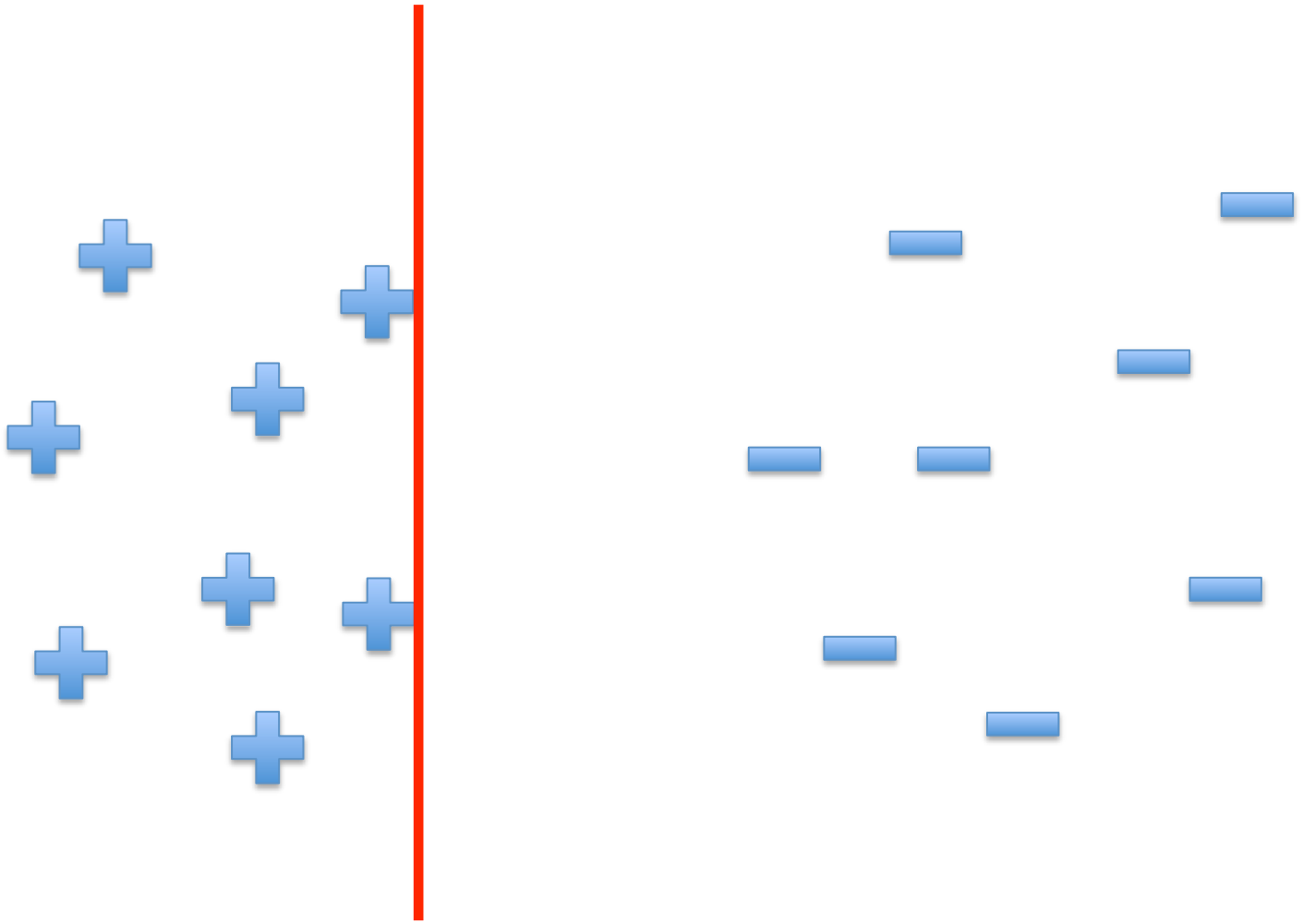
$$\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u} \cdot \mathbf{v} = \mathbf{u}^\intercal \mathbf{v}$$

$$= \sum_i u_i v_i$$

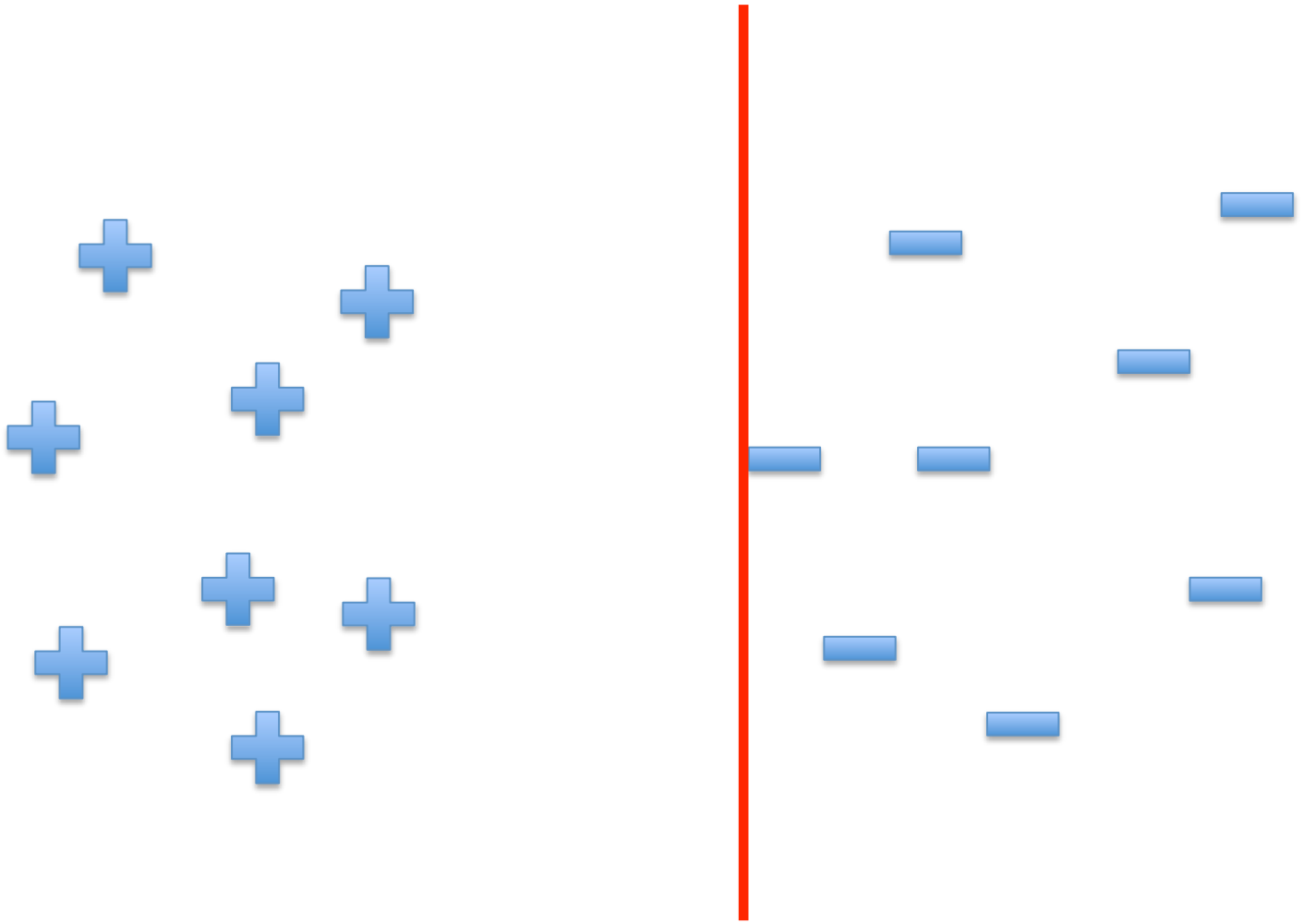
Intuitions



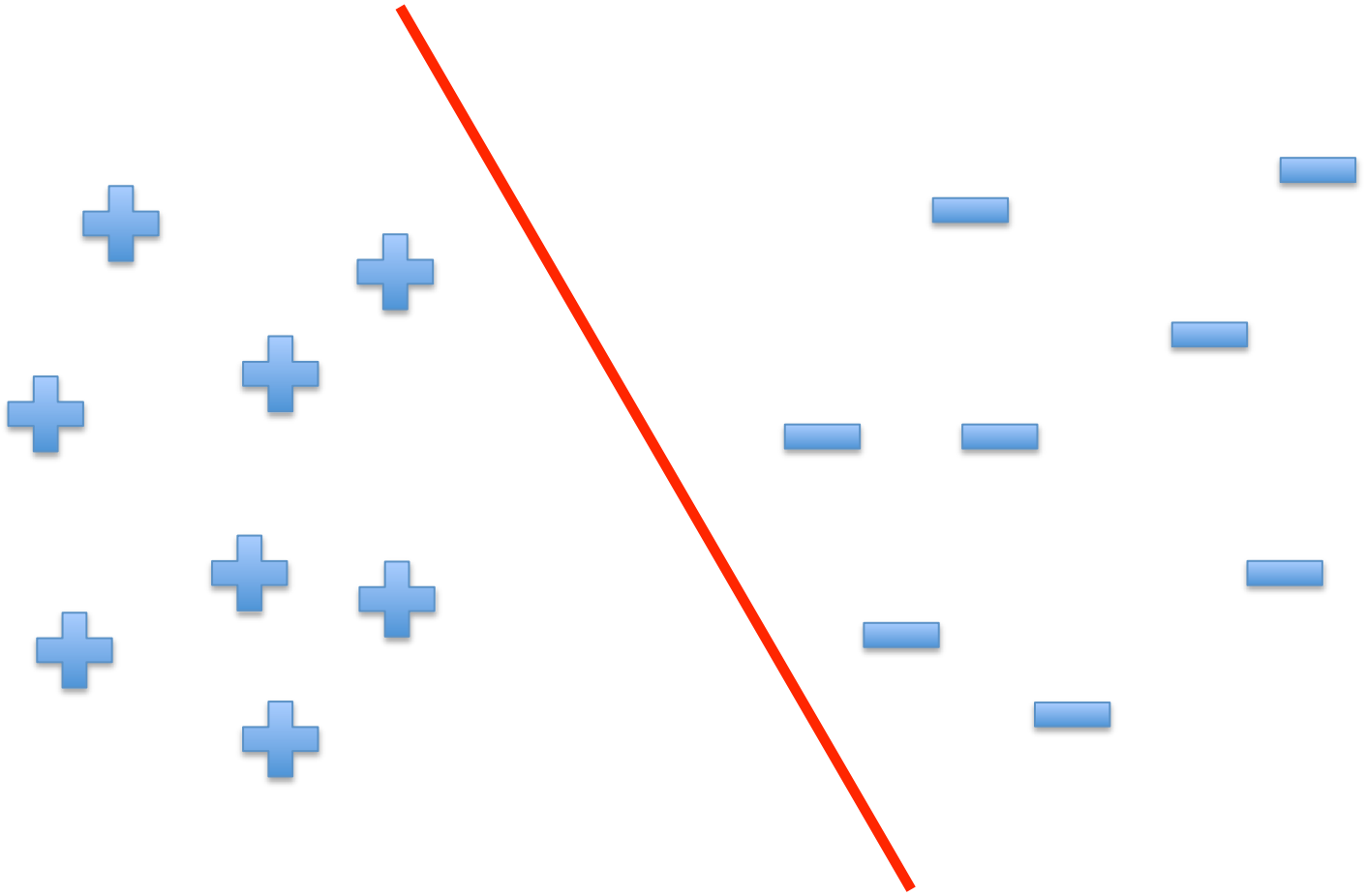
Intuitions



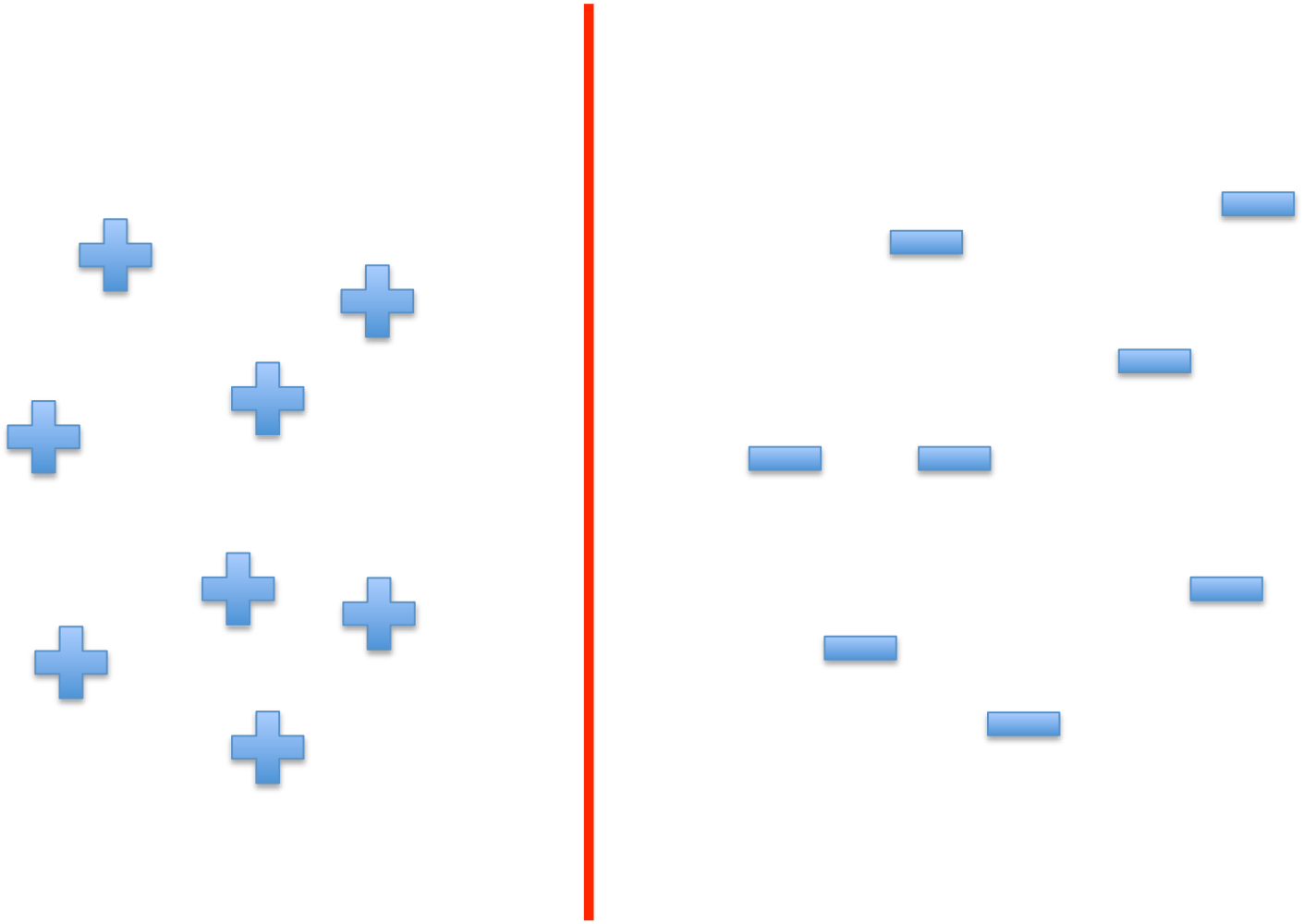
Intuitions



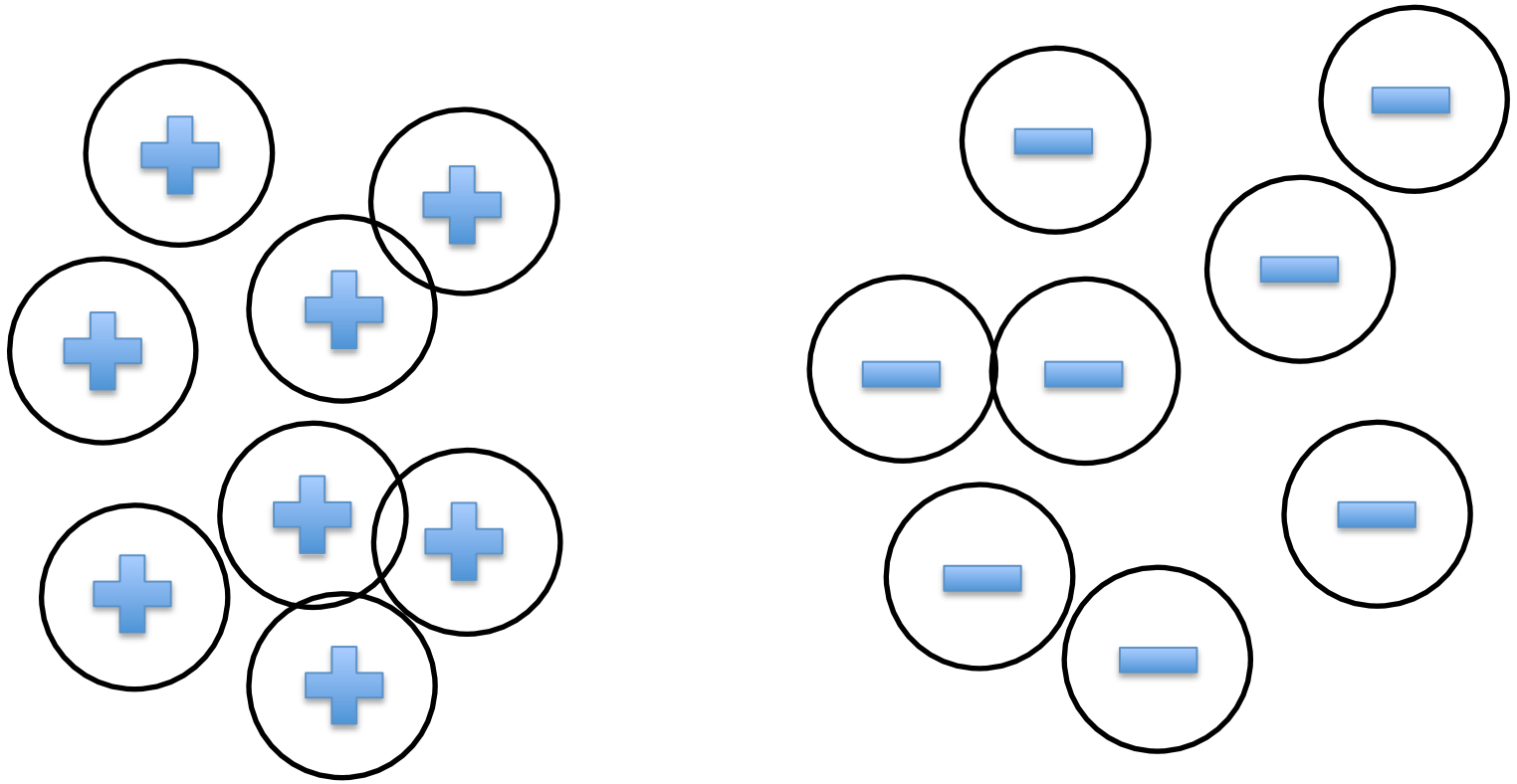
Intuitions



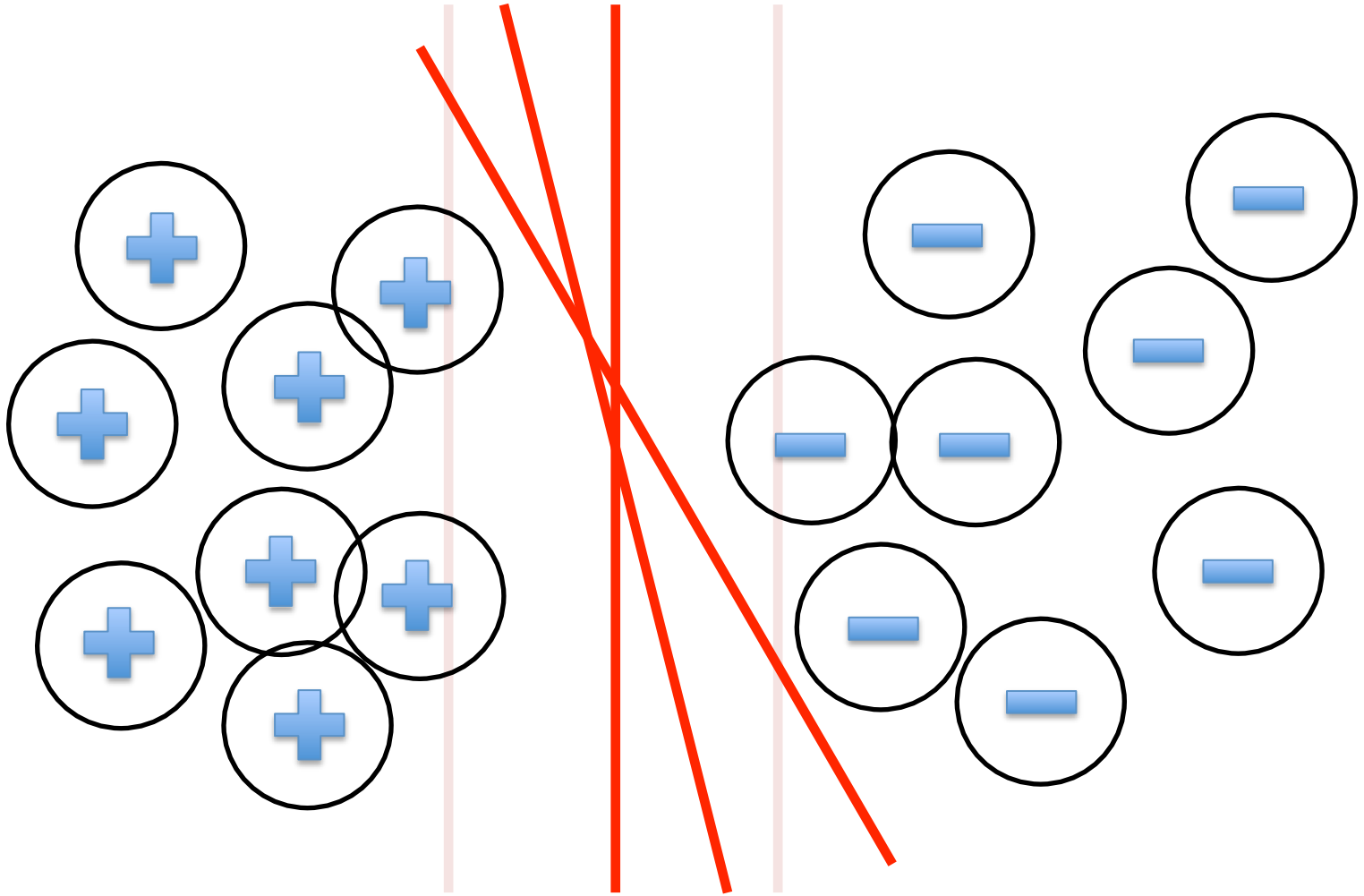
A “Good” Separator



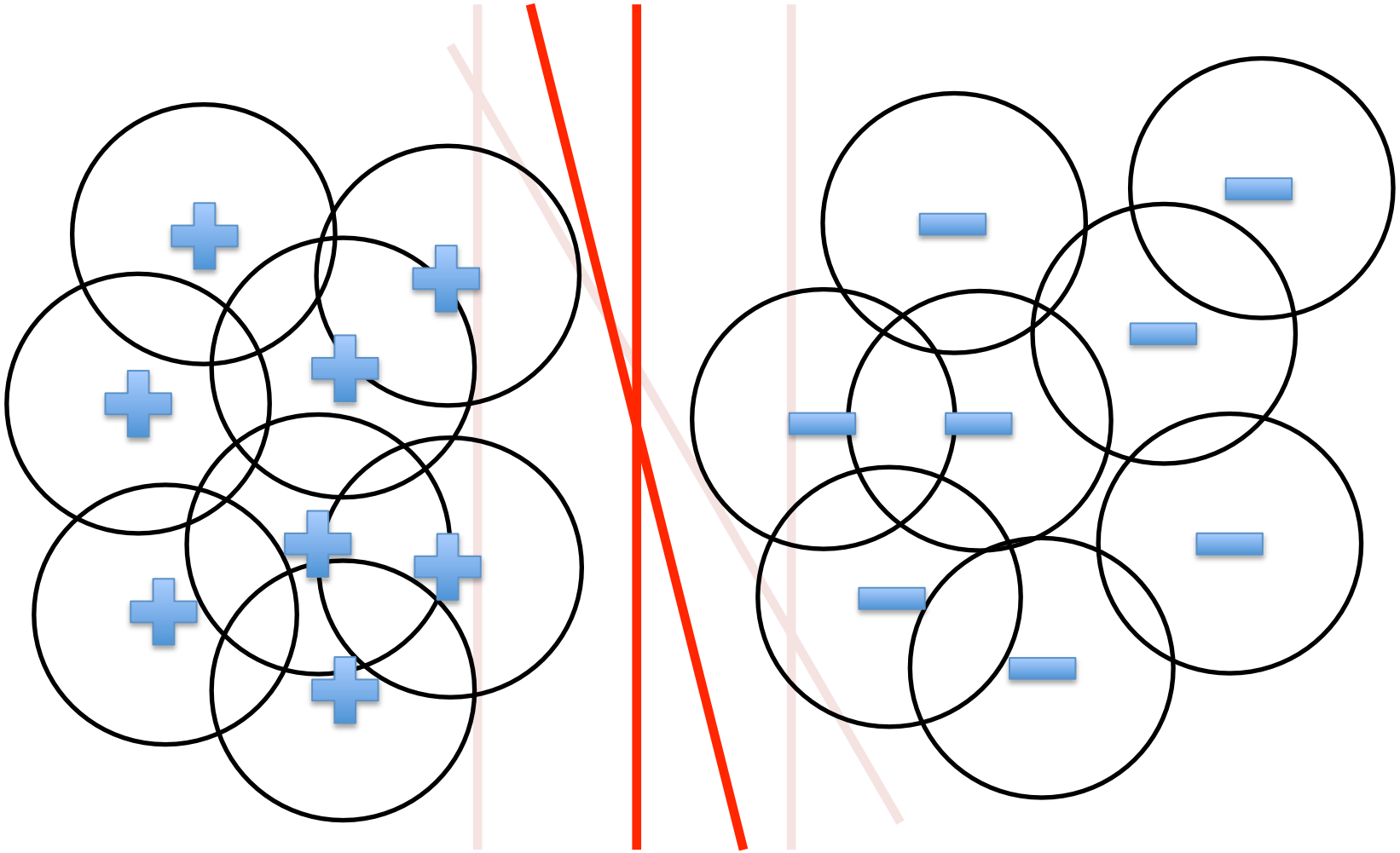
Noise in the Observations



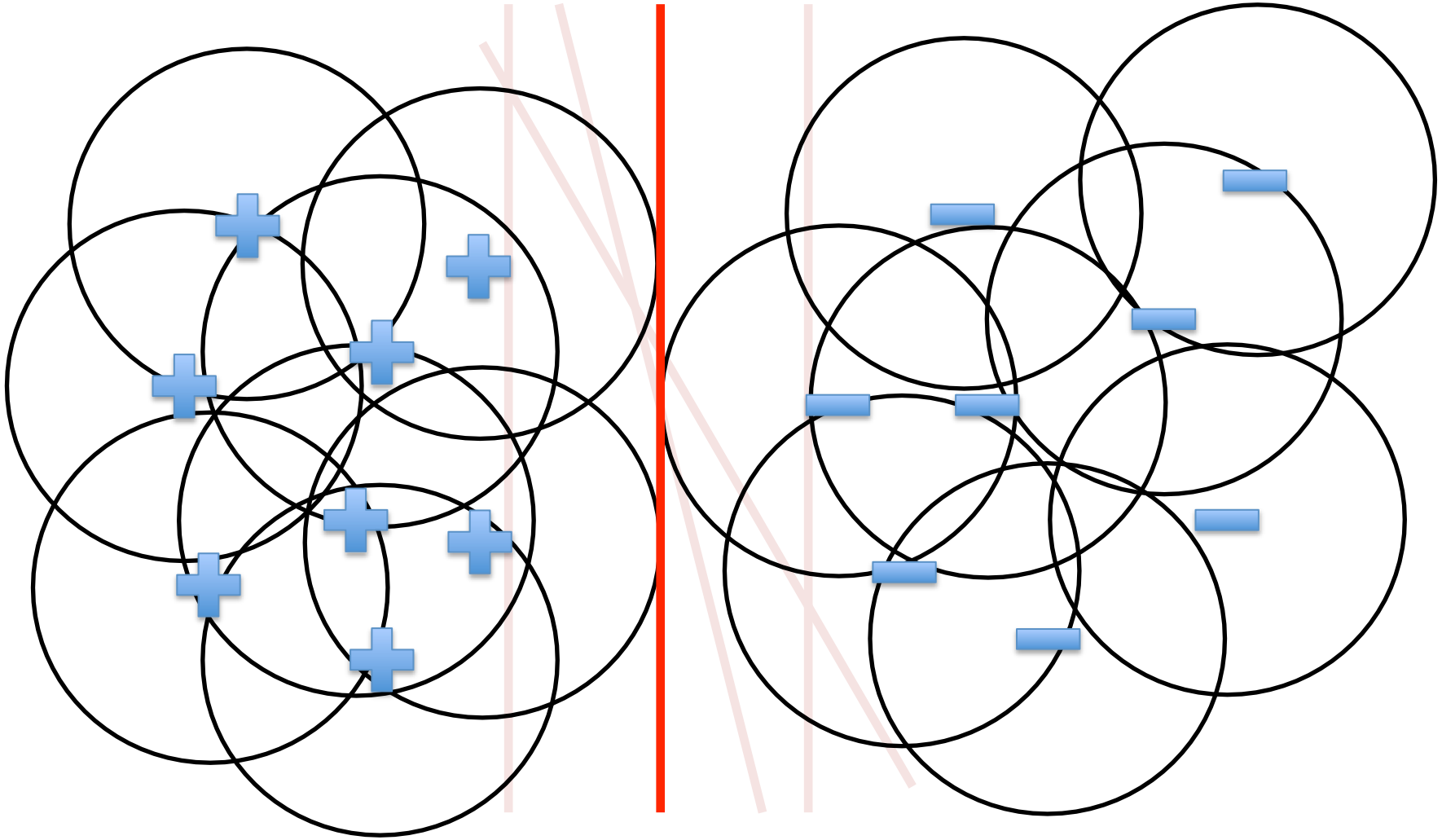
Ruling Out Some Separators



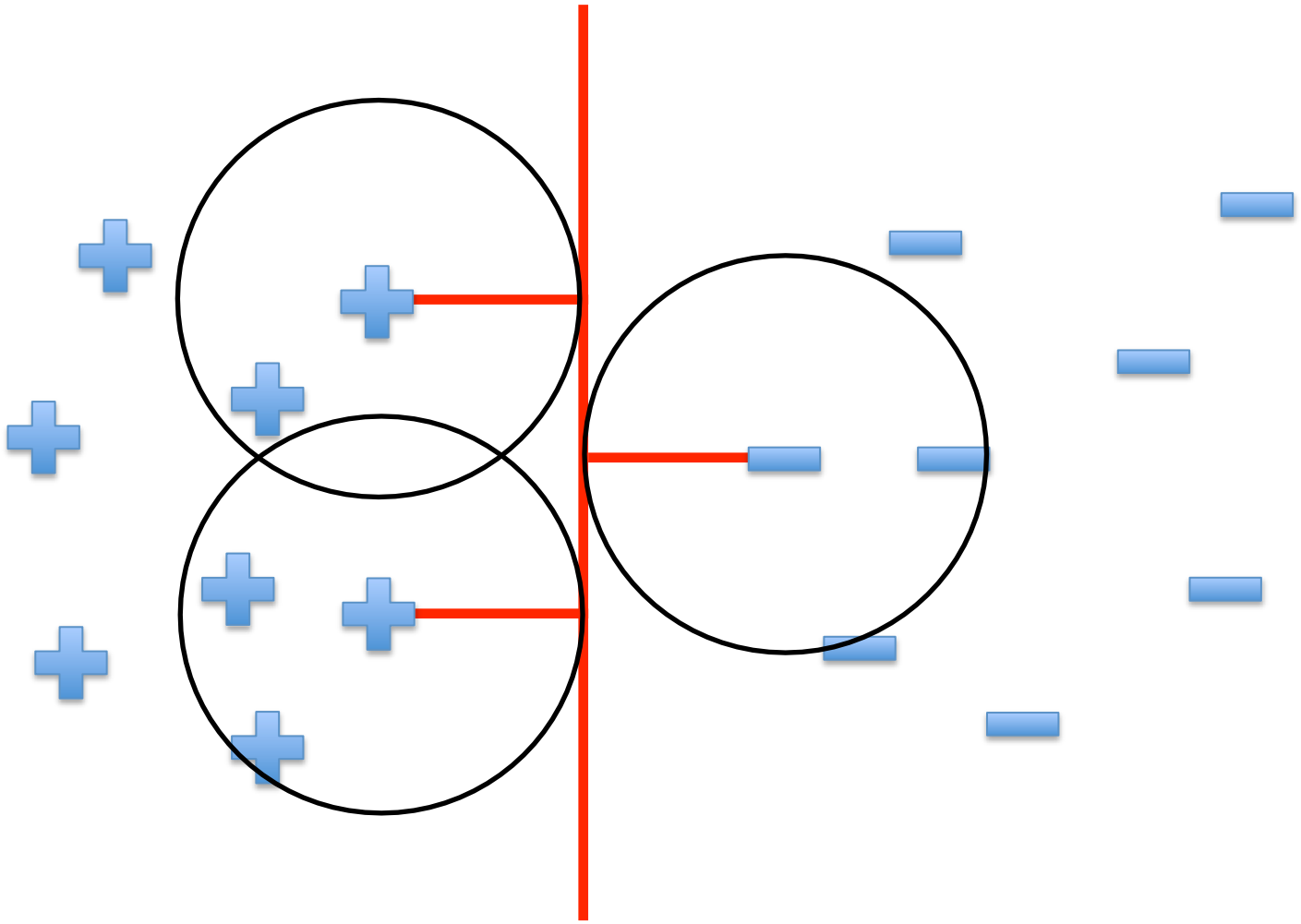
Lots of Noise



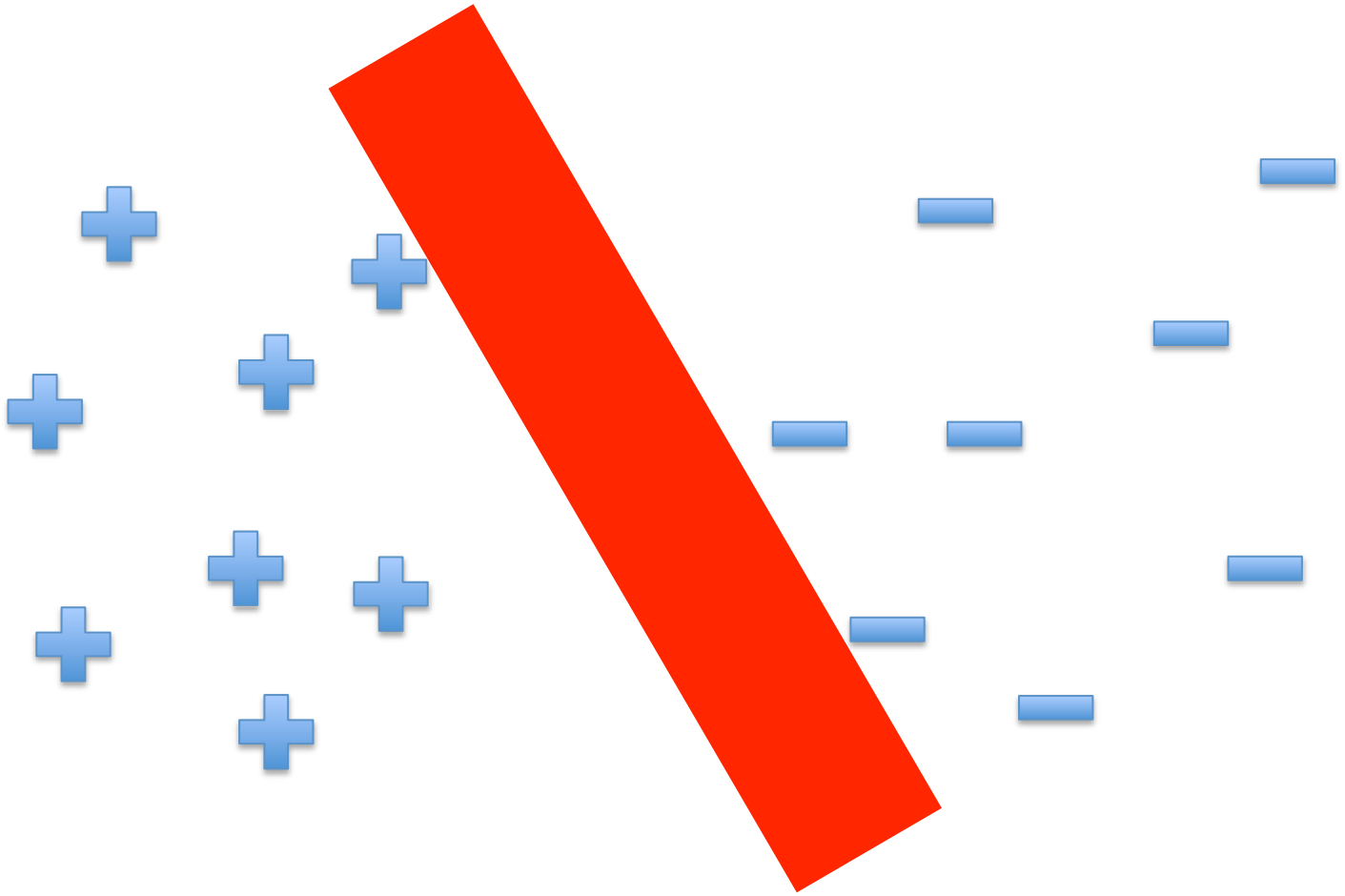
Intuitions



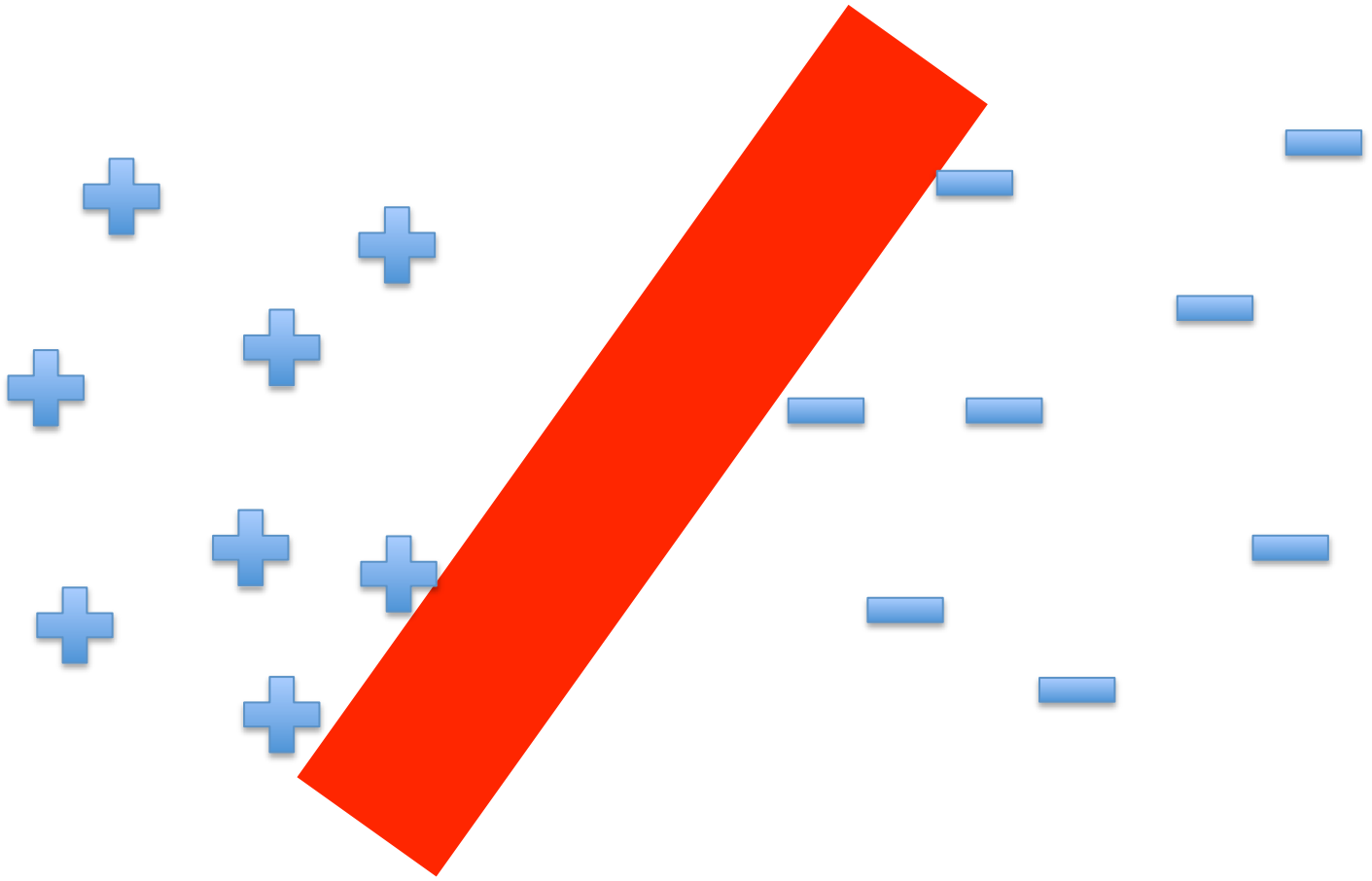
Maximizing the Margin



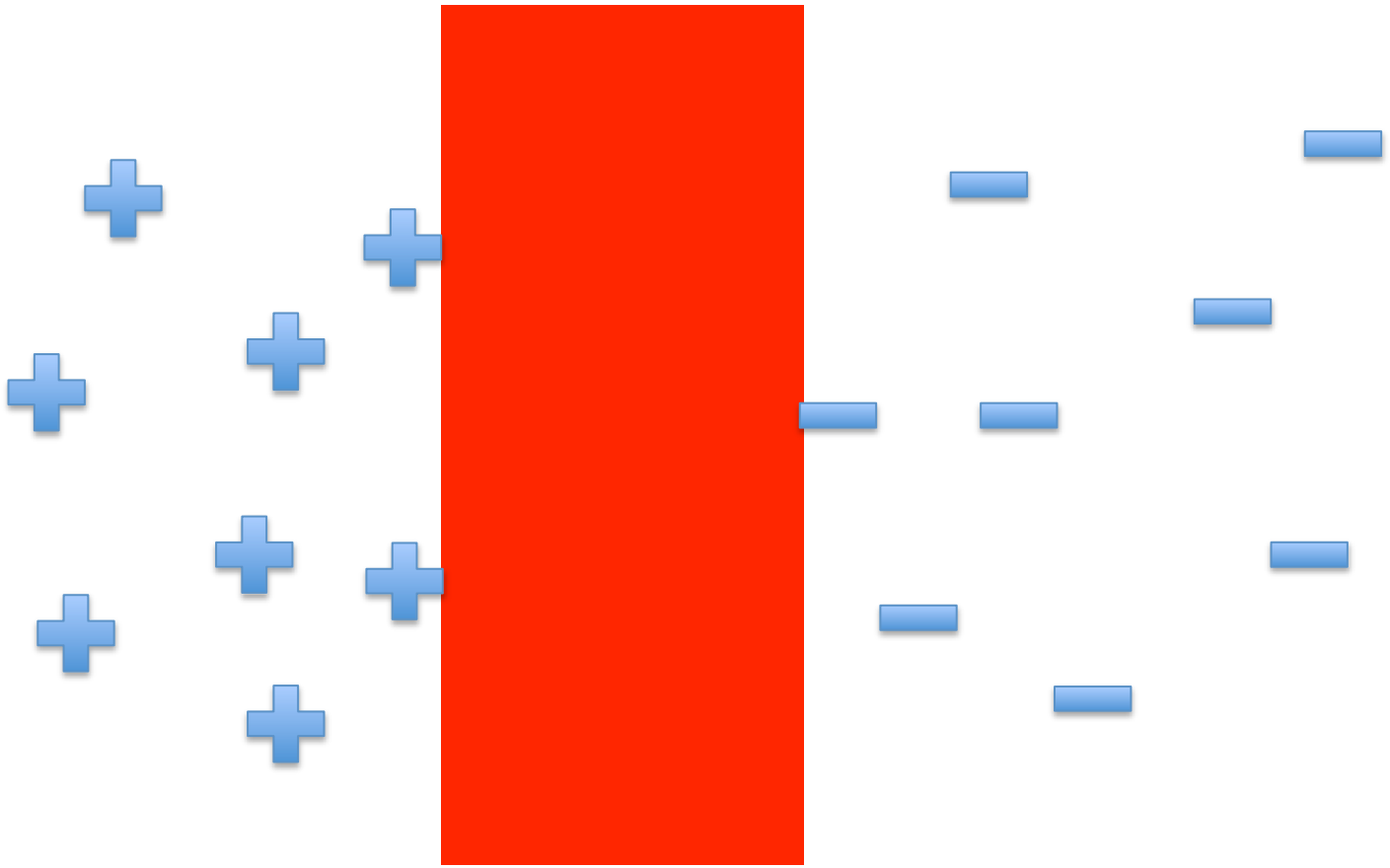
“Fat” Separators



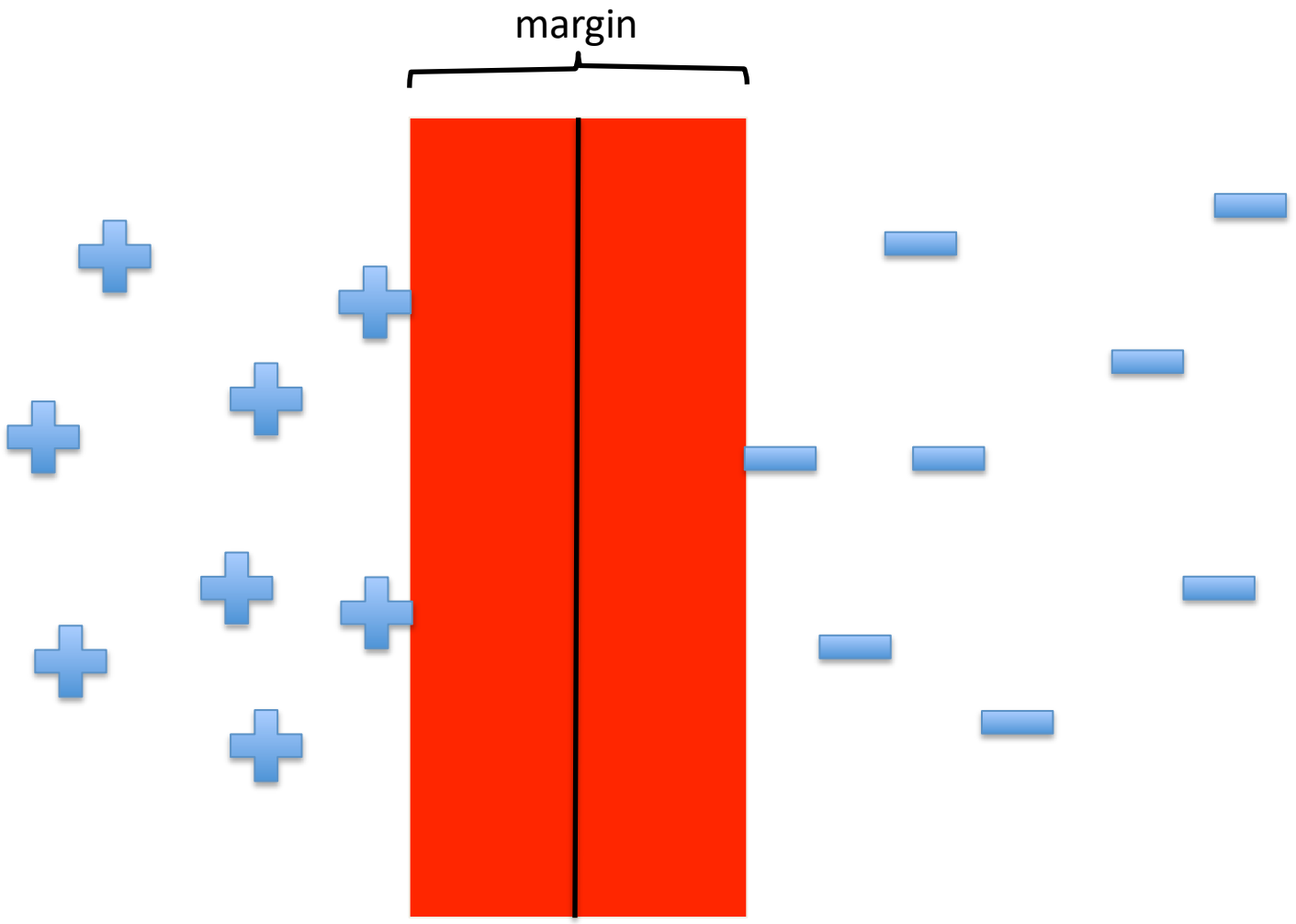
“Fat” Separators



“Fat” Separators



“Fat” Separators

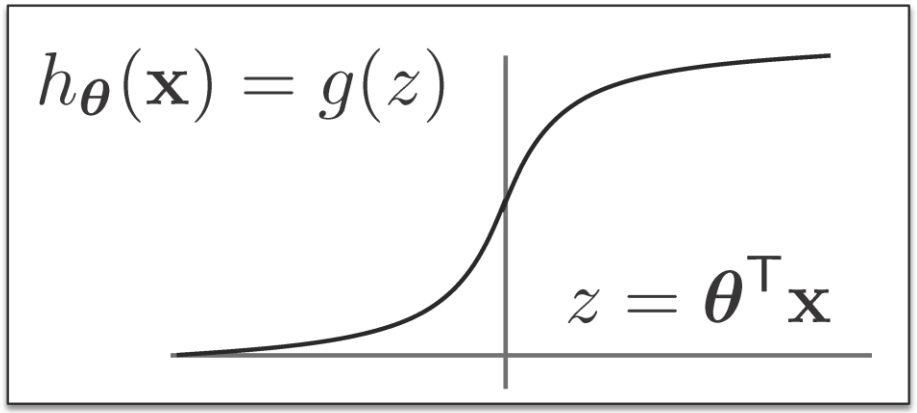


Why Maximize Margin

- Increasing margin reduces *capacity*
  - i.e., fewer possible models
  - Add a safe margin which helps avoid overfitting (noise-resilient)
  - Produce relatively low error on test data

Alternative View of Logistic Regression

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x}}}$$



If $y = 1$, we want $h_{\boldsymbol{\theta}}(\mathbf{x}) \approx 1, \quad \boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \gg 0$

If $y = 0$, we want $h_{\boldsymbol{\theta}}(\mathbf{x}) \approx 0, \quad \boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \ll 0$

$$J(\boldsymbol{\theta}) = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log h_{\boldsymbol{\theta}}(\mathbf{x}_i) + (1 - y_i) \log (1 - h_{\boldsymbol{\theta}}(\mathbf{x}_i))]$$

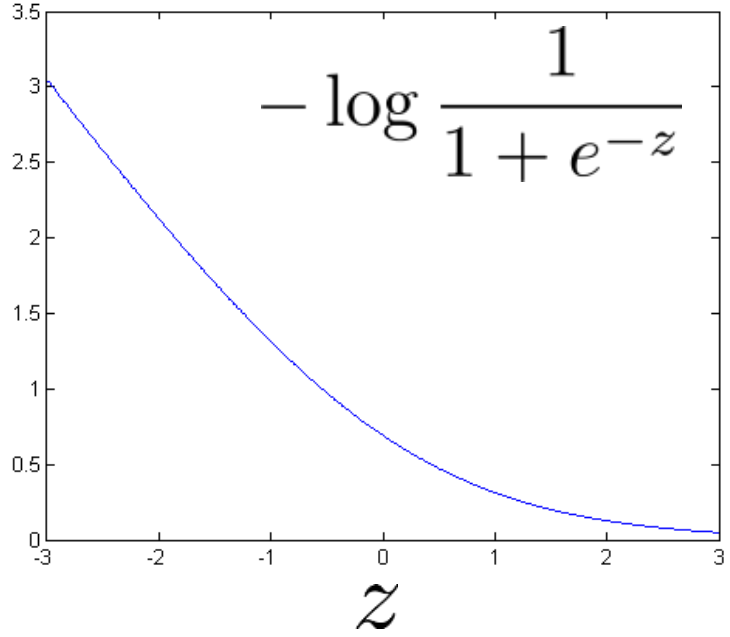
$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$

Alternative View of Logistic Regression

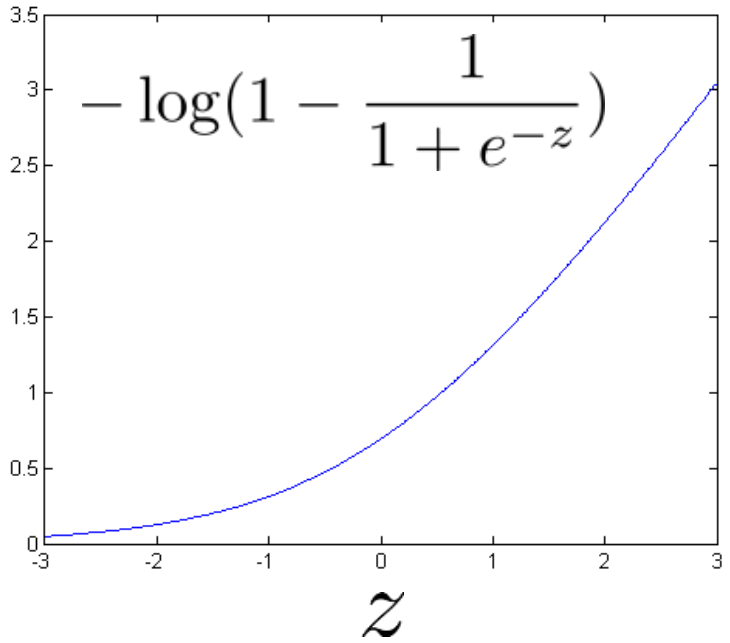
Cost of example: $-y_i \log h_{\boldsymbol{\theta}}(\mathbf{x}_i) - (1 - y_i) \log (1 - h_{\boldsymbol{\theta}}(\mathbf{x}_i))$

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x}}} \qquad z = \boldsymbol{\theta}^{\mathsf{T}}\mathbf{x}$$

If $y = 1$ (want $\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \gg 0$):



If $y = 0$ (want $\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \ll 0$):



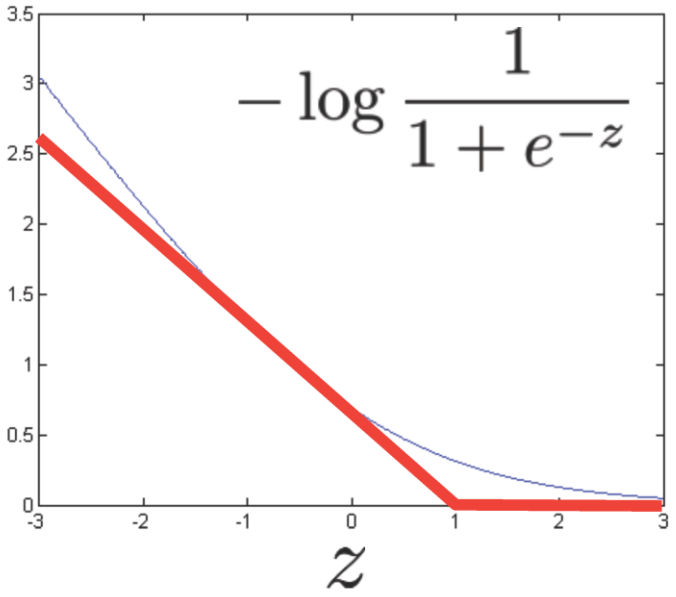
https://www.desmos.com/calculator/kreo2ssqj8

Cost of Example in SVMs

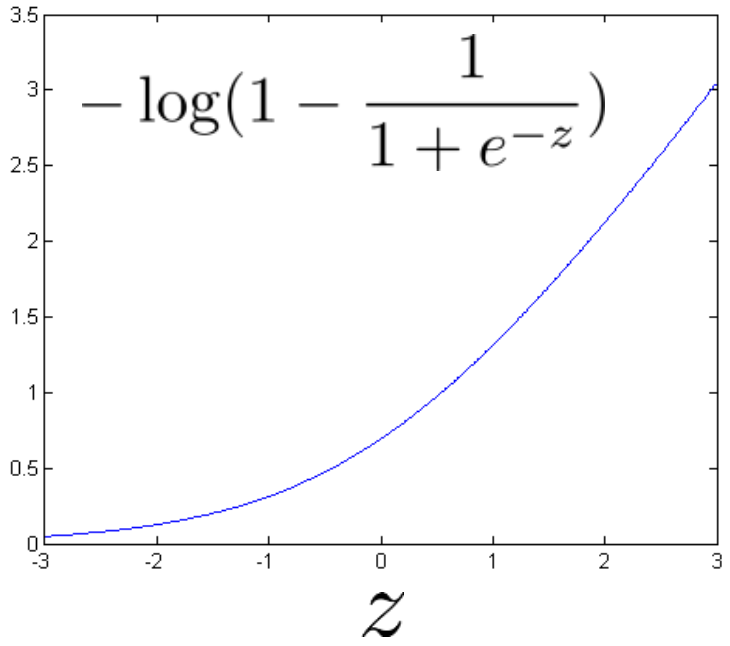
Cost of example: $-y_i \log h_{\boldsymbol{\theta}}(\mathbf{x}_i) - (1 - y_i) \log (1 - h_{\boldsymbol{\theta}}(\mathbf{x}_i))$

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x}}} \qquad z = \boldsymbol{\theta}^{\mathsf{T}}\mathbf{x}$$

If $y = 1$ (want $\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \gg 0$):



If $y = 0$ (want $\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \ll 0$):

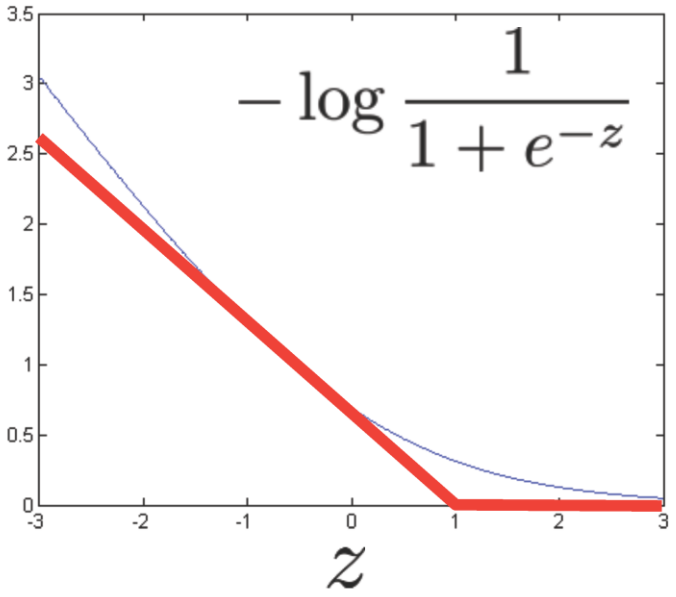


Cost of Example in SVMs

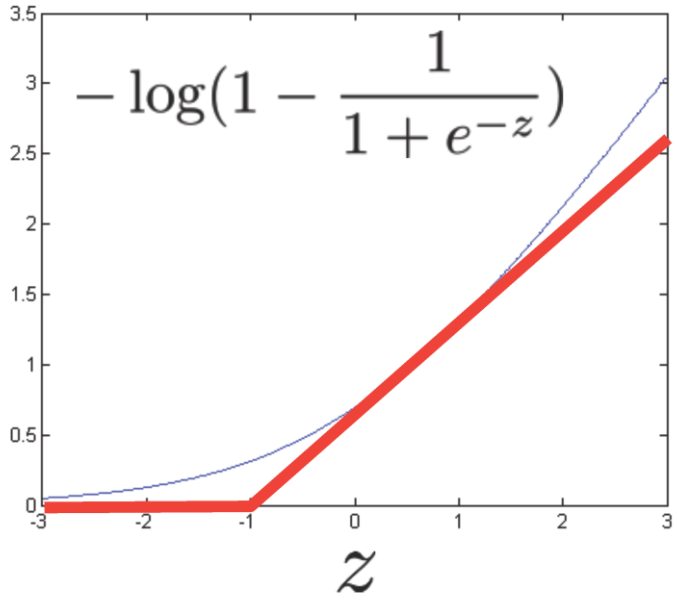
Cost of example: $-y_i \log h_{\boldsymbol{\theta}}(\mathbf{x}_i) - (1 - y_i) \log (1 - h_{\boldsymbol{\theta}}(\mathbf{x}_i))$

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x}}} \qquad z = \boldsymbol{\theta}^{\mathsf{T}}\mathbf{x}$$

If $y = 1$ (want $\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \gg 0$):



If $y = 0$ (want $\boldsymbol{\theta}^{\mathsf{T}}\mathbf{x} \ll 0$):



Logistic Regression to SVMs

Regularized Logistic Regression:

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} [y_i \underbrace{(-\log h_{\boldsymbol{\theta}}(\mathbf{x}_i))}_{\text{cost}_1(\boldsymbol{\theta}^\intercal \mathbf{x}_i)} + (1 - y_i) \underbrace{(-\log (1 - h_{\boldsymbol{\theta}}(\mathbf{x}_i)))}_{\text{cost}_0(\boldsymbol{\theta}^\intercal \mathbf{x}_i)}] + \frac{\lambda}{2} \sum_{j=1}^{d} \theta_j^2$$

Support Vector Machines:

$$\min_{\boldsymbol{\theta}} \; C \sum_{i=1}^{n} [y_i \text{cost}_1(\boldsymbol{\theta}^\intercal \mathbf{x}_i) + (1 - y_i) \, \text{cost}_0(\boldsymbol{\theta}^\intercal \mathbf{x}_i)] + \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

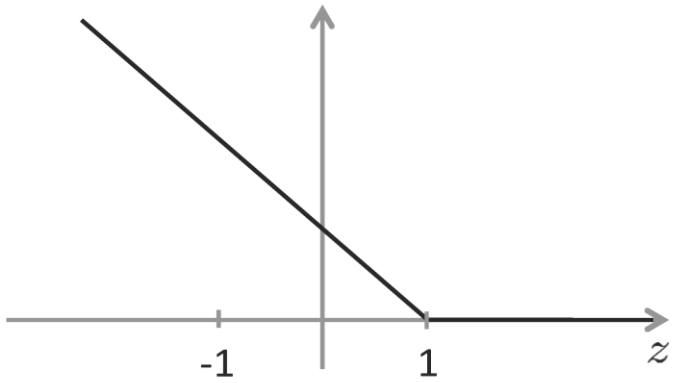
You can think of C as similar to $\frac{1}{\lambda}$

Support Vector Machine

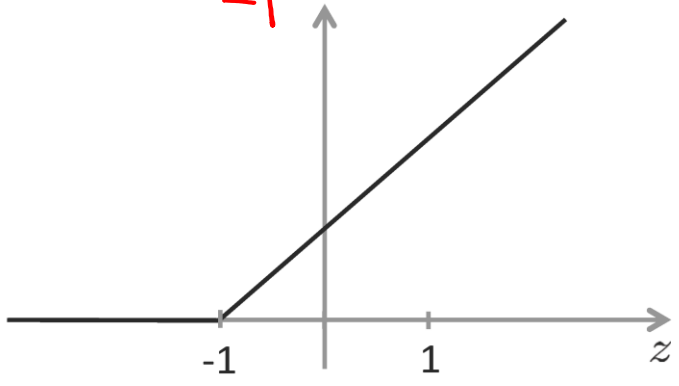
$$\min_{\boldsymbol{\theta}} \; C \sum_{i=1}^{n} [y_i \text{cost}_1(\boldsymbol{\theta}^\intercal \mathbf{x}_i) + (1 - y_i)\, \text{cost}_0(\boldsymbol{\theta}^\intercal \mathbf{x}_i)] + \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

piecewise function

If $y = 1$ (want $\boldsymbol{\theta}^\intercal \mathbf{x} \geq 1$):



If $y = 0$ (want $\boldsymbol{\theta}^\intercal \mathbf{x} \leq -1$):



When Z = $\boldsymbol{\theta}^\intercal \mathbf{x}_i$

$$\ell_{\text{hinge}}(h(\mathbf{x})) = \max(0, 1 - y\,(\boldsymbol{\theta}^\intercal \mathbf{x}_i))$$

y = +1 / -1

$$\min_{\boldsymbol{\theta}} \; C \sum_{i=1}^{n} [y_i \text{cost}_1(\boldsymbol{\theta}^\intercal \mathbf{x}_i) + (1 - y_i)\, \text{cost}_0(\boldsymbol{\theta}^\intercal \mathbf{x}_i)] + \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

y = 1 / 0

y = +1 / -1

with $C = 1$

Hard margin linear SVM classifier objective (constrained optimization problem)

$$\min_{\boldsymbol{\theta}} \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

$$\text{s.t.} \;\; \boldsymbol{\theta}^\intercal \mathbf{x}_i \geq 1 \quad \text{if } y_i = 1$$

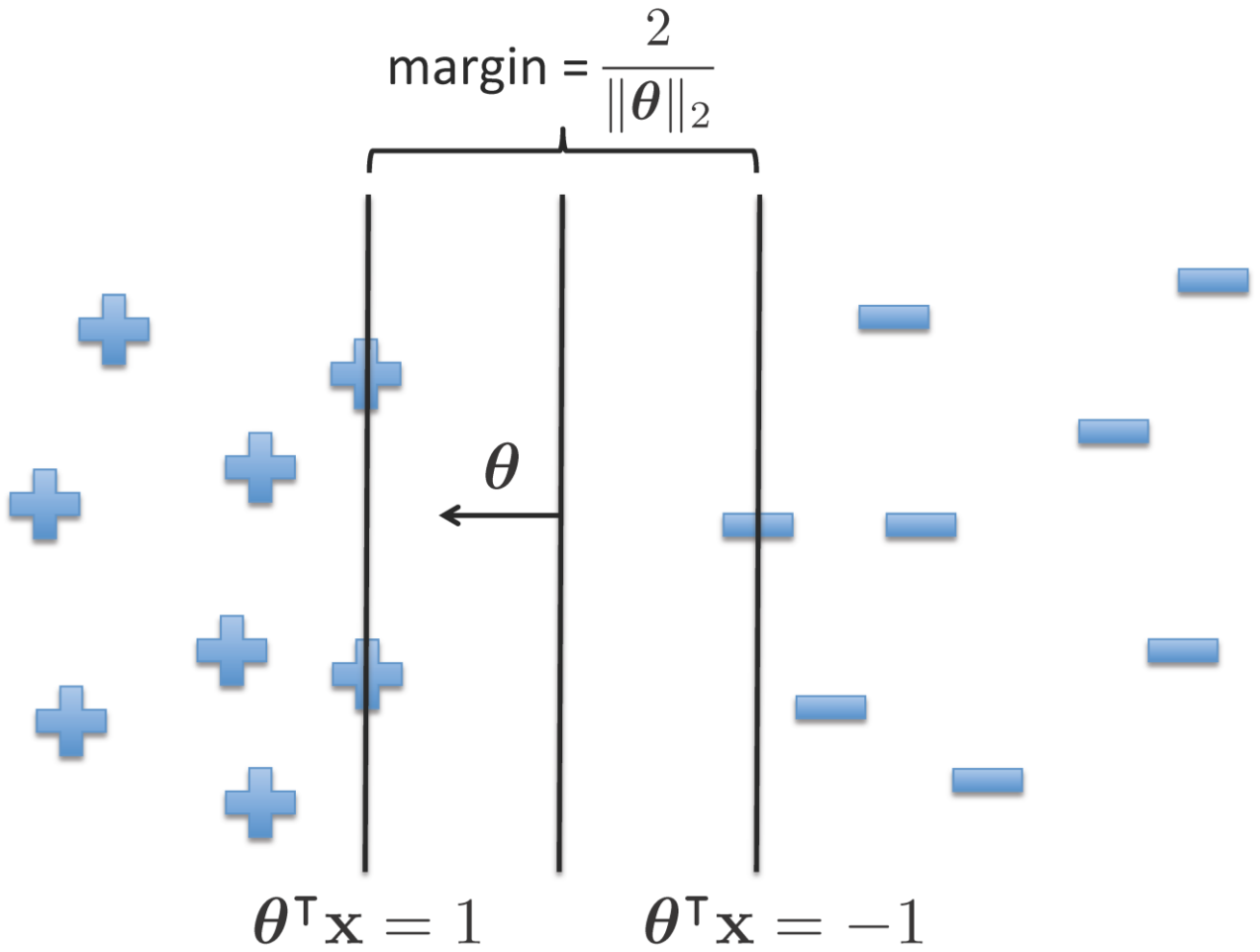
$$\boldsymbol{\theta}^\intercal \mathbf{x}_i \leq -1 \quad \text{if } y_i = -1$$

$$\min_{\boldsymbol{\theta}} \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

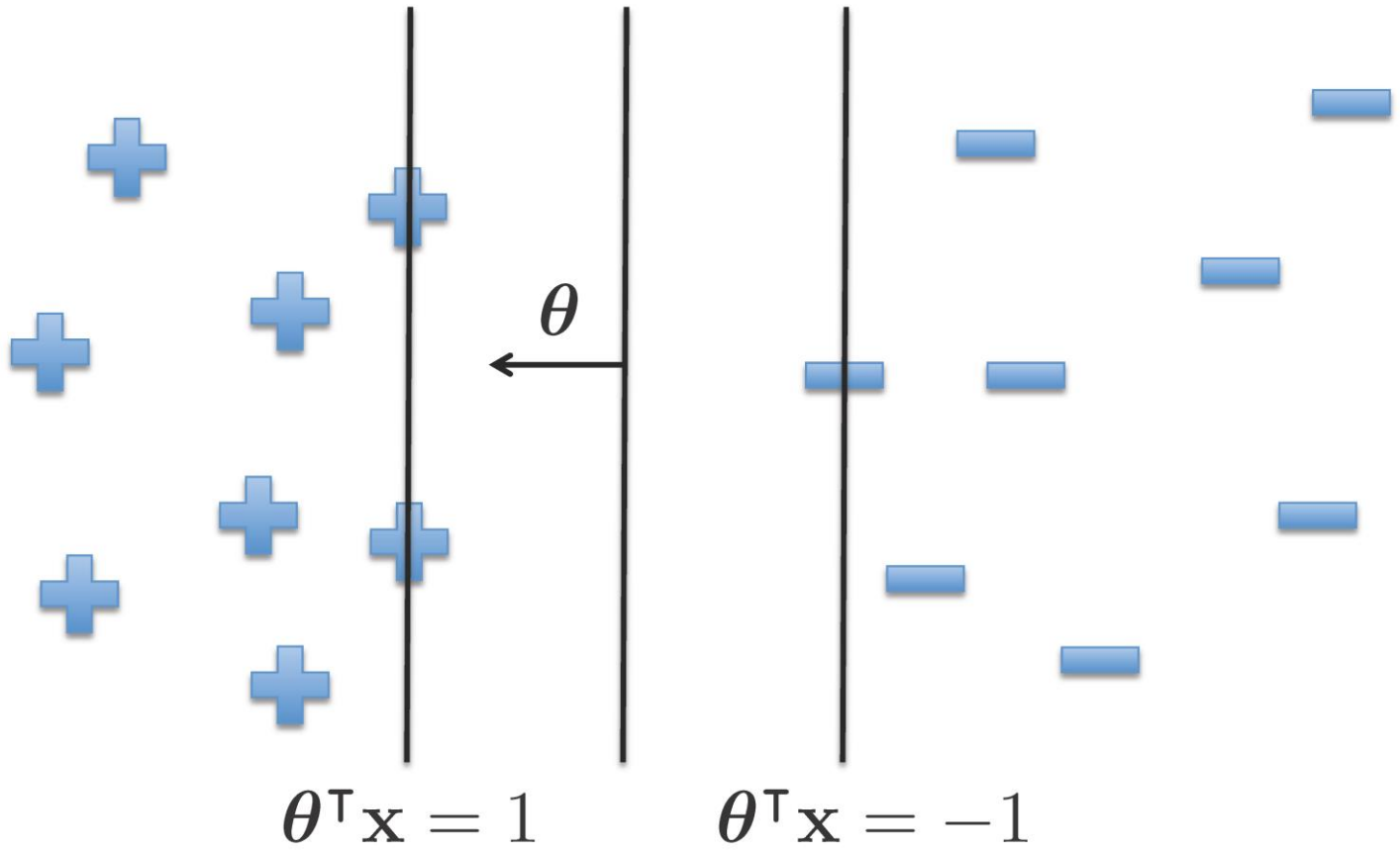
$$\text{s.t.} \;\; y_i(\boldsymbol{\theta}^\intercal \mathbf{x}_i) \geq 1$$

This approach only works when the training data is **linearly separable** without error

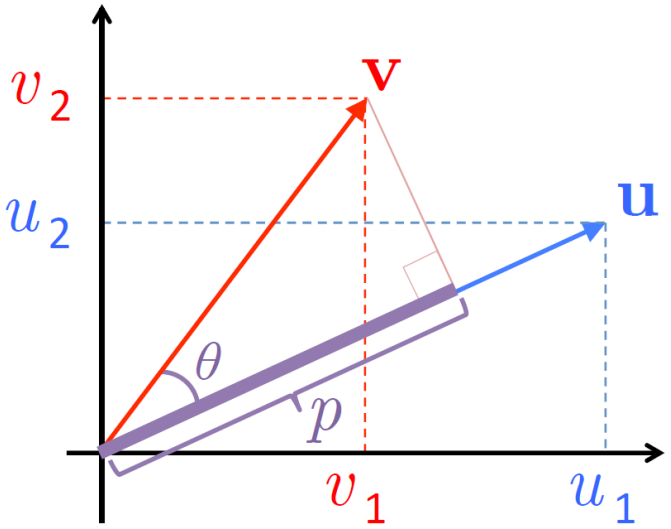
Maximum Margin Hyperplane



Support Vectors



Vector Inner Product



$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \qquad v = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$\|\mathbf{u}\|_2 = \text{length}(\mathbf{u}) \in \mathbb{R}$$
$$= \sqrt{u_1^2 + u_2^2}$$

$$\mathbf{u}^\intercal \mathbf{v} = \mathbf{v}^\intercal \mathbf{u}$$
$$= u_1 v_1 + u_2 v_2$$
$$= \|\mathbf{u}\|_2 \ \|\mathbf{v}\|_2 \cos\theta$$
$$= p\|\mathbf{u}\|_2 \quad \text{where } p = \|\mathbf{v}\|_2 \cos\theta$$

https://www.mathsisfun.com/sine-cosine-tangent.html

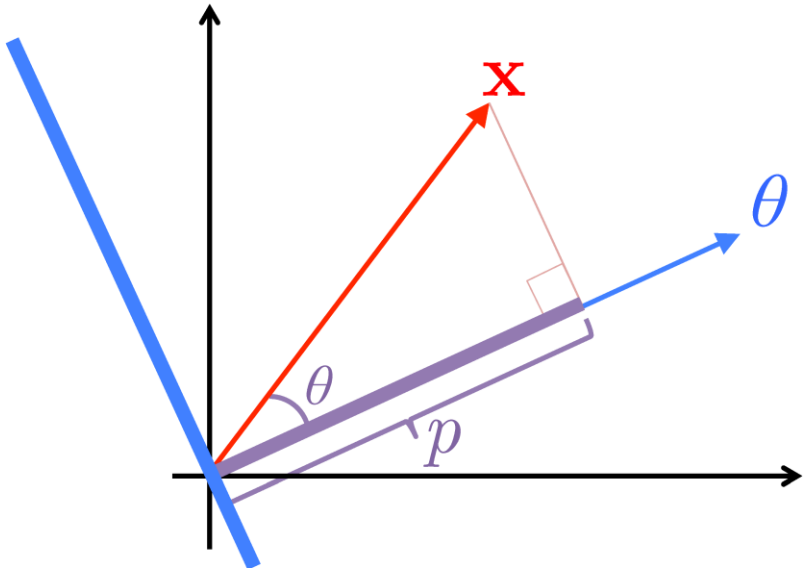
Understanding the Hyperplane

$$\min_{\boldsymbol{\theta}} \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

$$\text{s.t. } \boldsymbol{\theta}^\mathsf{T}\mathbf{x}_i \geq 1 \quad \text{if } y_i = 1$$

$$\boldsymbol{\theta}^\mathsf{T}\mathbf{x}_i \leq -1 \quad \text{if } y_i = -1$$

Assume $\theta_0 = 0$ so that the hyperplane is centered at the origin, and that $d = 2$



$$\boldsymbol{\theta}^\mathsf{T}\mathbf{x} = \|\boldsymbol{\theta}\|_2 \underbrace{\|\mathbf{x}\|_2 \cos\theta}_{p}$$

$$= p\|\boldsymbol{\theta}\|_2$$

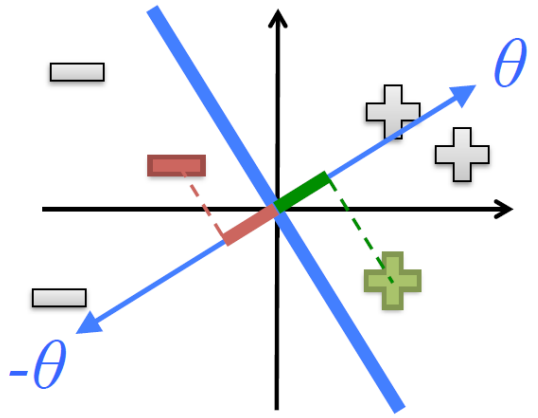
Maximizing the Margin

$$\min_{\boldsymbol{\theta}} \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

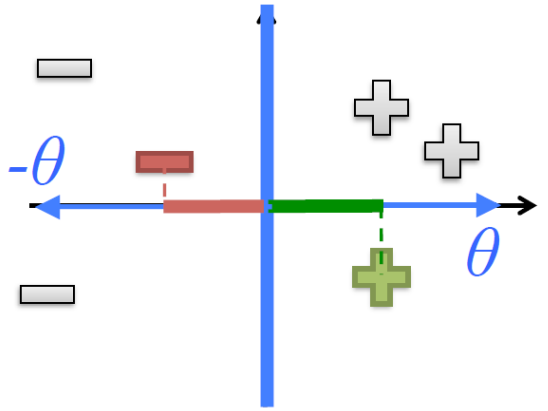
$$\text{s.t. } p\|\boldsymbol{\theta}\|_2 \geq 1 \quad \text{if } y_i = 1$$
$$p\|\boldsymbol{\theta}\|_2 \leq -1 \quad \text{if } y_i = -1$$

Assume $\theta_0 = 0$ so that the hyperplane is centered at the origin, and that $d = 2$

Let p_i be the projection of $\mathbf{x}_i$ onto the vector $\boldsymbol{\theta}$



Since p is small, therefore $\|\boldsymbol{\theta}\|_2$ must be large to have $p\|\boldsymbol{\theta}\|_2 \geq 1$ (or ≤ -1)

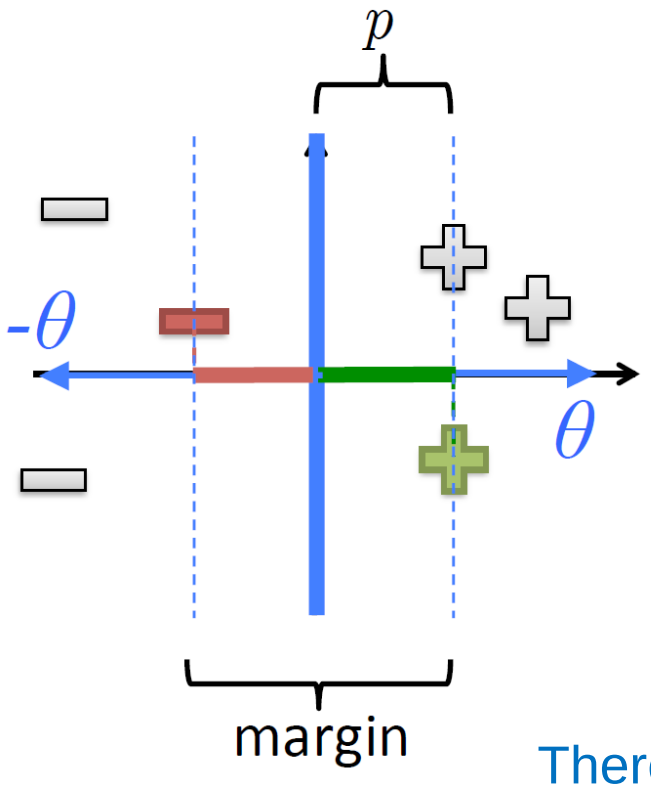


Since p is larger, $\|\boldsymbol{\theta}\|_2$ can be smaller in order to have $p\|\boldsymbol{\theta}\|_2 \geq 1$ (or ≤ -1)

Size of the Margin

For the support vectors, we have $p\|\boldsymbol{\theta}\|_2 = \pm 1$

- p is the length of the projection of the SVs onto $\boldsymbol{\theta}$



Therefore,

$$p = \frac{1}{\|\boldsymbol{\theta}\|_2}$$

$$\text{margin} = 2p = \frac{2}{\|\boldsymbol{\theta}\|_2}$$

Therefore, minimizing theta means maximizing the margin (2p)

The SVM Dual Problem

The primal SVM problem was given as

$$\min_{\boldsymbol{\theta}} \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

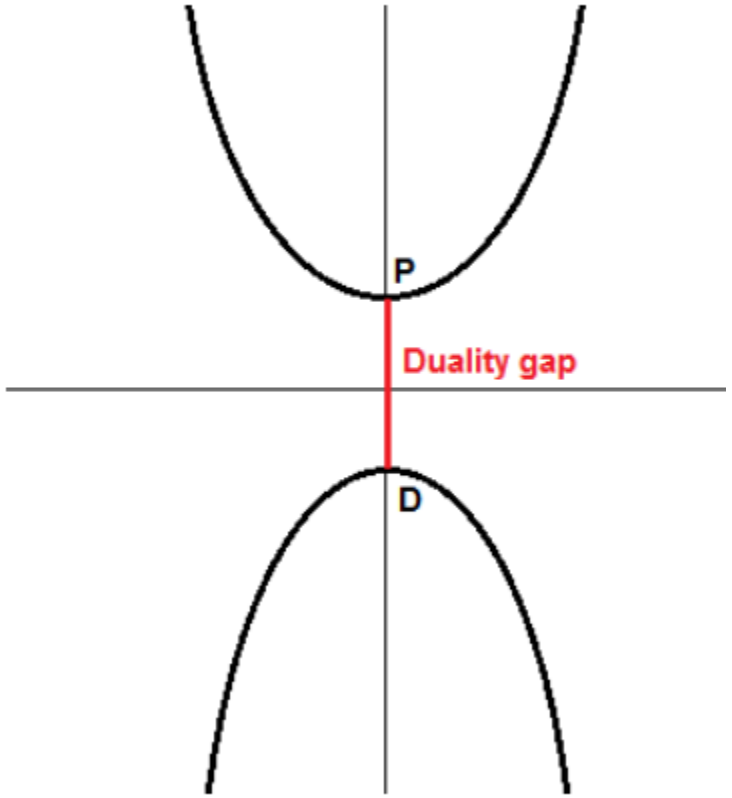
$$\text{s.t. } y_i(\boldsymbol{\theta}^\intercal \mathbf{x}_i) \geq 1 \quad \forall i$$

Can solve it more efficiently by taking the Lagrangian dual

- Duality is a common idea in optimization
- It transforms a difficult optimization problem into a simpler one
- Key idea: introduce Lagrange multiplier α_i for each constraint
  - α_i indicates how important a particular constraint is to the solution

In mathematical optimization, the method of Lagrange multipliers is a strategy for finding the local maxima and minima of a function subject to equality constraints. (Wikipedia)

Duality



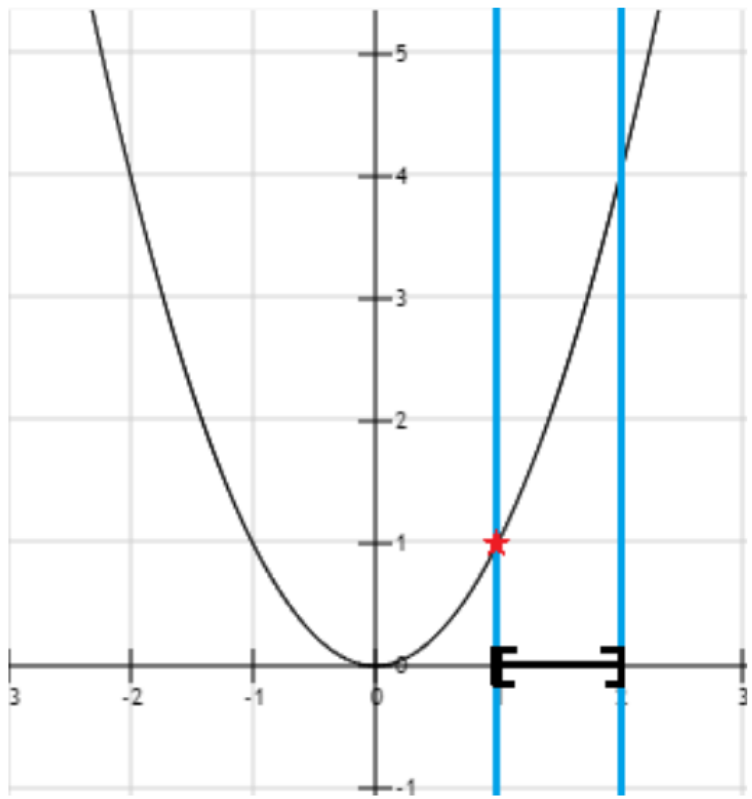
- In mathematical optimization theory, **duality** means that optimization problems may be viewed from either of two perspectives, the primal problem or the dual problem (**the duality principle**). The solution to the dual problem provides a lower bound to the solution of the primal (minimization) problem.

Adopted content of the following slides from https://www.svm-tutorial.com/2016/09/duality-lagrange-multipliers/

Combining constraints

It is possible to add several constraints to an optimization problem. Here is an example with two inequality constraints and its visual representation:

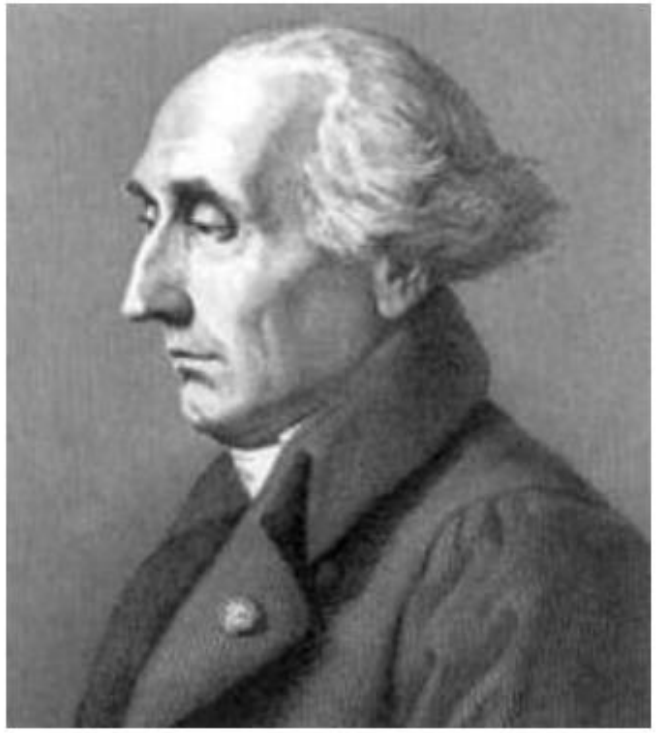
$$\begin{aligned} \underset{x}{\text{minimize}} \quad & x^2 \\ \text{subject to} \quad & x >= 1 \\ & x <= 2 \end{aligned}$$



Combining constraints restrict the feasible region

How do we find the solution to an optimization problem with constraints?

We will be using the Lagrange Multipliers. It is a method invented by the Italian mathematician, Joseph-Louis Lagrange around 1806.



Joseph Louis Lagrange

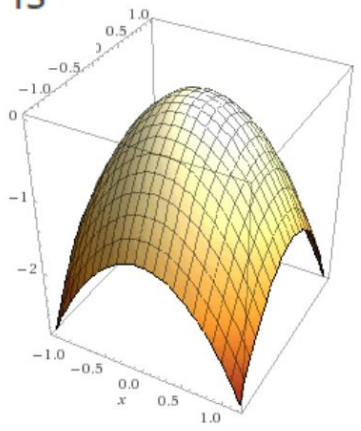
Lagrange multipliers

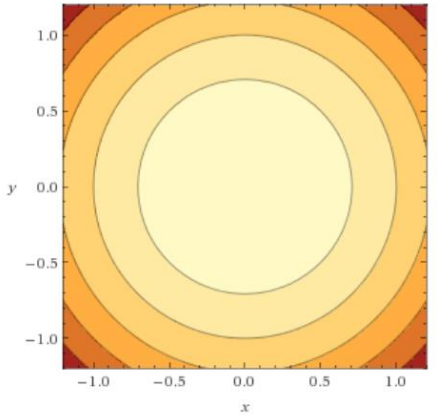
- As often, we can find a pretty clear definition on Wikipedia:
  - In mathematical optimization, the method of Lagrange multipliers is a strategy for finding the local maxima and minima of a function subject to equality constraints.

Contour lines

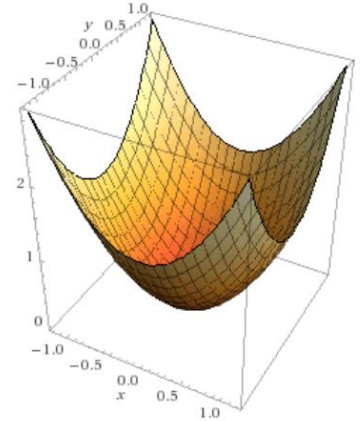
Key concepts regarding contour lines:

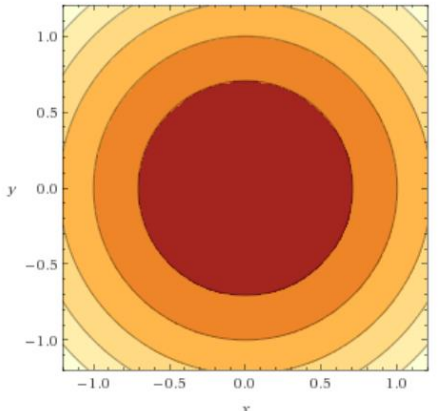
- for each point on a line, the function returns the same value
- the darker the area is, the smallest the value of the function is





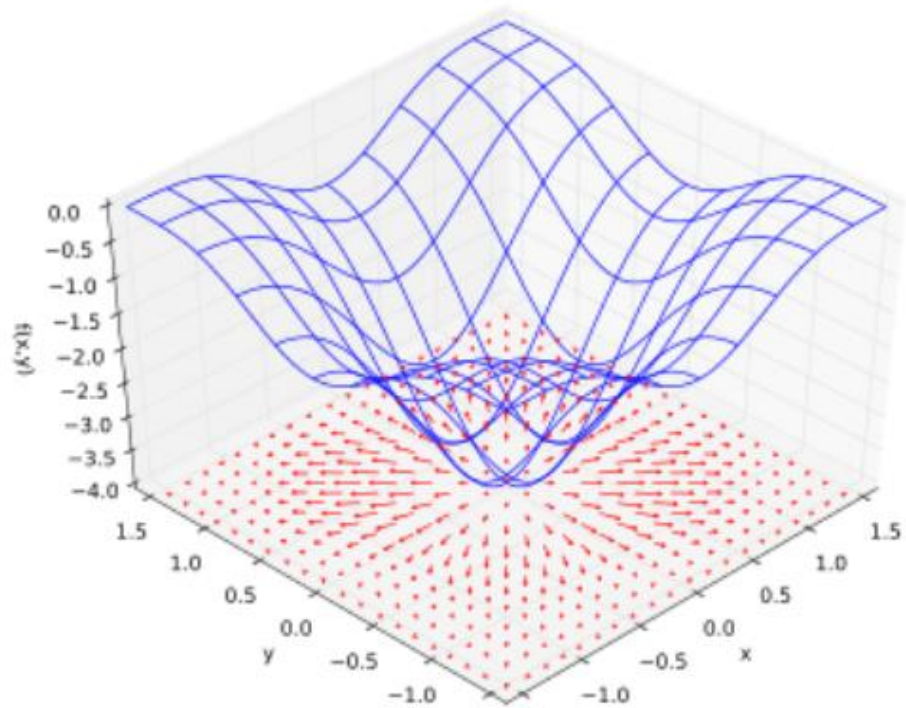
$-x^2 - y^2$





$x^2 + y^2$

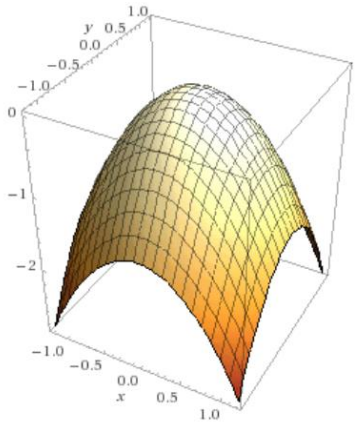
Moreover, the gradient of a function can be visualized as a vector field, with the arrow pointing in the direction where the function is increasing:

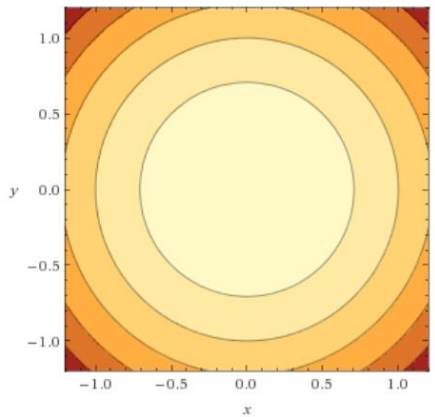


The gradient of the function is projected as a vector field (Wikipedia)

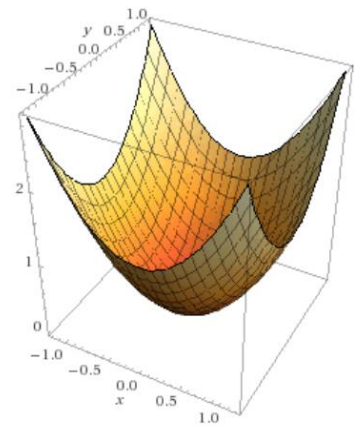
It turns out we can easily draw gradient vectors on a contour plot:

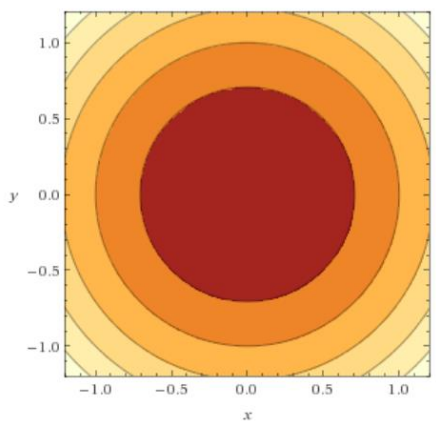
- they are perpendicular to a contour line
- they should point in the direction where the function is increasing





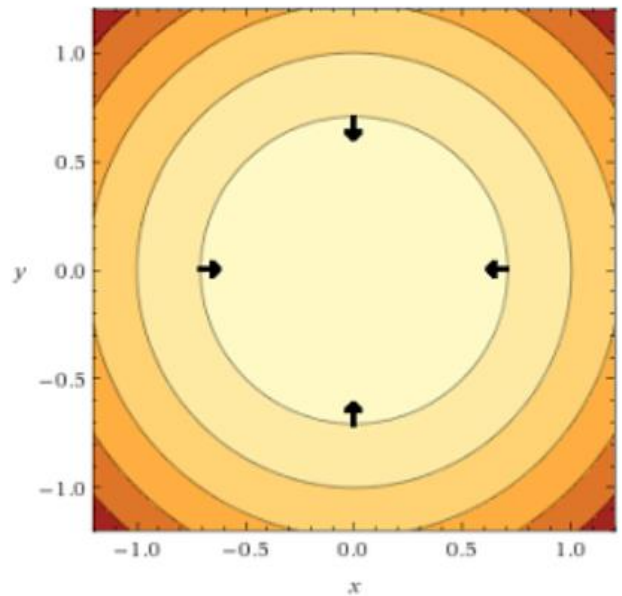
$-x^2 - y^2$

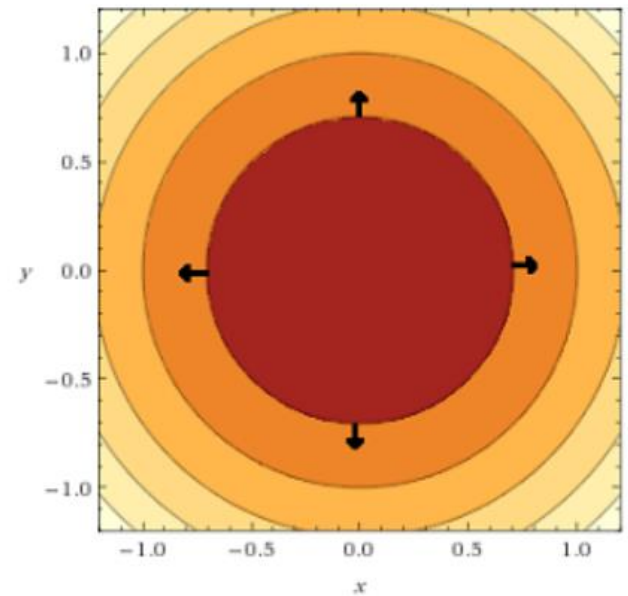




$x^2 + y^2$

Here is a representation on two contours plots:



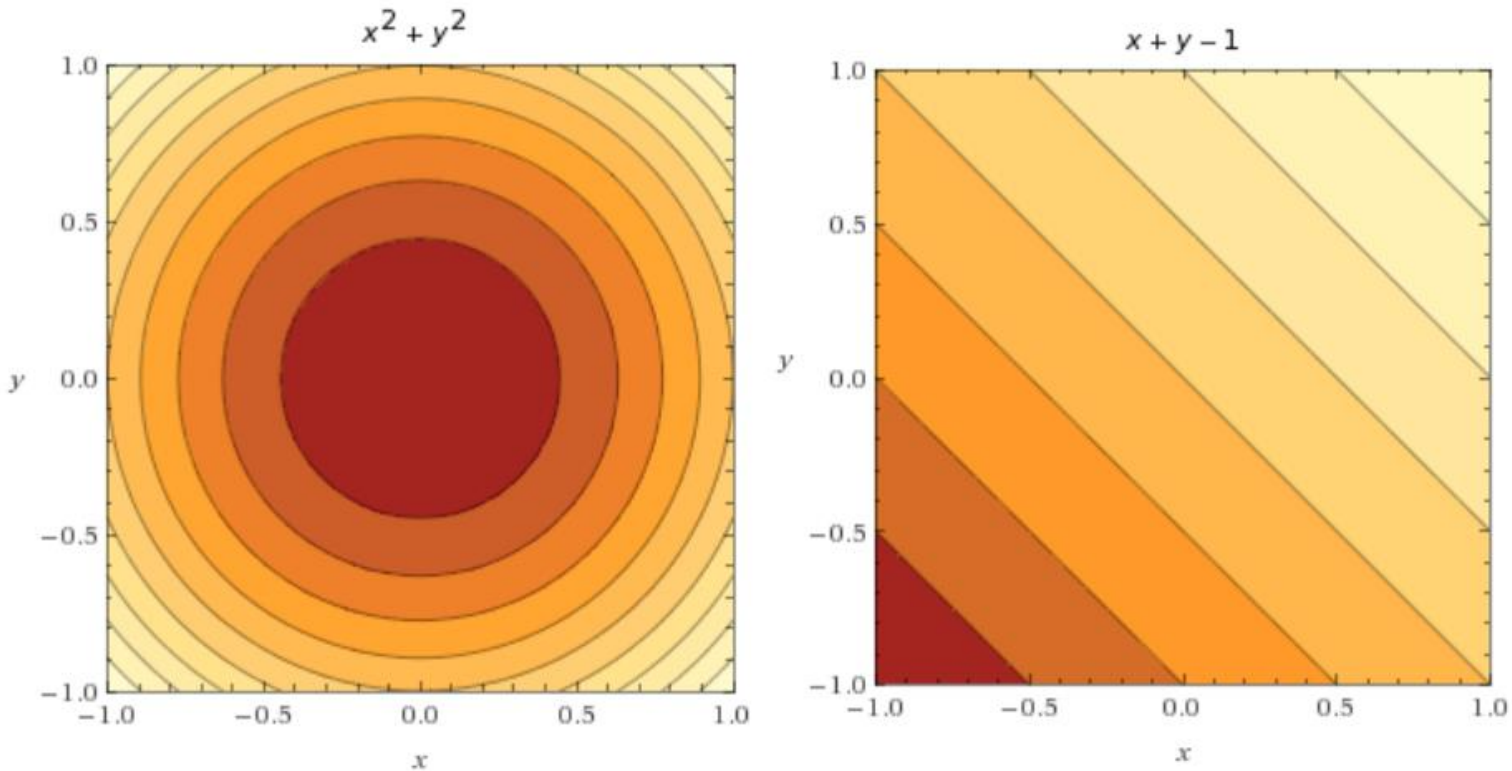


Back to Lagrangian multipliers

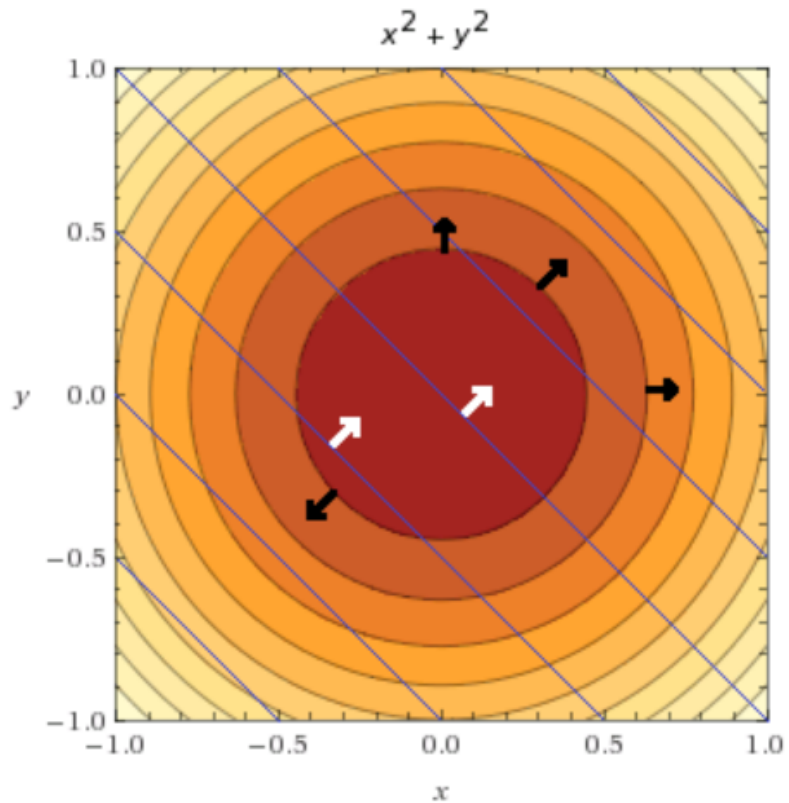
Let us consider the following optimization problem:

$$\begin{aligned} \underset{x,y}{\text{minimize}} \quad & f(x,y) = x^2 + y^2 \\ \text{subject to} \quad & g_i(x,y) = x + y - 1 = 0 \end{aligned}$$

The objective function $f(x,y)$ and the constraint function $g(x,y)$ can be visualized as contour in the figure below:



It is interesting to note that we can combine both contour plot to visualize how the two functions behave on one plot. Below you can see the constraint function depicted by blue lines.
Also, I draw some gradient vectors of the objective function (in black) and some gradient vectors of the constraint function (in white).



However, we are not interested in the whole constraint function. We are only interested in points where the constraint is satisfied, i.e. $g(x, y) = 0$.

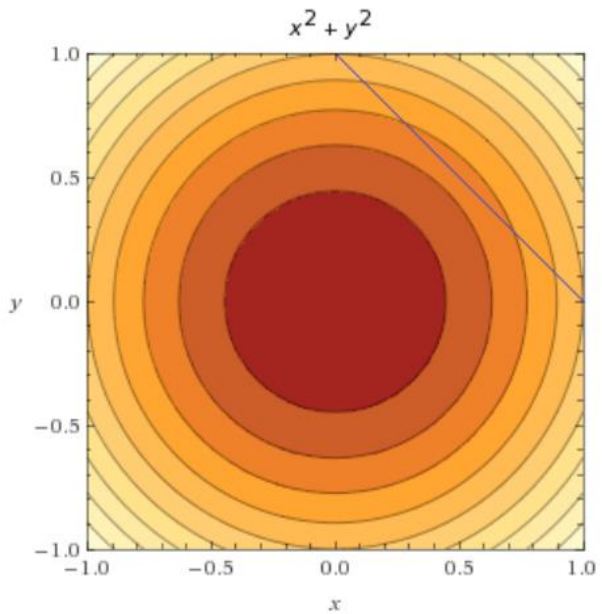
It means that we want points where:

$$x + y - 1 = 0$$

$$x + y = 1$$

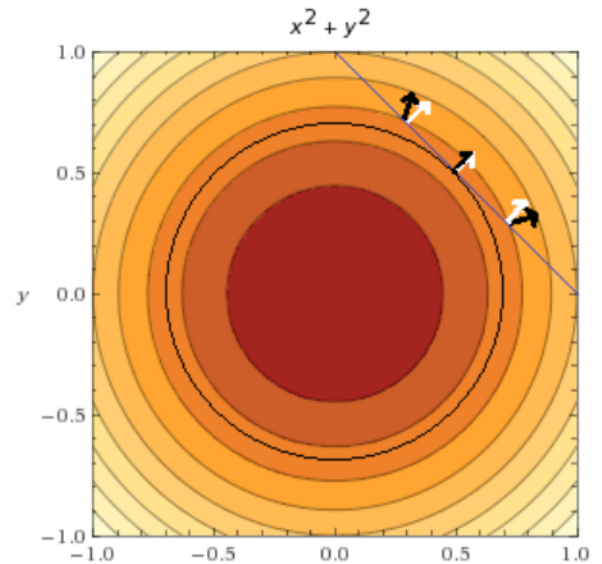
$$y = 1 - x$$

In the graph below we plot the line $y = 1 - x$ on top of the objective function. We earlier that this line is also called the **feasible set**.



What did Lagrange find? He found that the minimum of $f(x, y)$ under the constraint $g(x, y) = 0$ is obtained **when their gradients point in the same direction**. Let us look at how we can find a similar conclusion.

In the figure below, we can see the point where the objective function and the feasible set are tangent. I also added some objective function gradient vectors in black and some constraint gradient vectors in white.



Lagrange told us that to find the minimum of a constrained function, we need to look or points were $\nabla f(x, y) = \lambda \nabla g(x, y)$.

But what is λ and where does it come from?

It is what we call a **Lagrange multiplier**. Indeed, even if the two gradient vectors point in the same direction, they might not have the same length, so there must be some factor λ allowing to transform one in the other.

How do we find points for which $\nabla f(x,y) = \lambda \nabla g(x,y)$?

Note that, $\nabla f(x,y) = \lambda \nabla g(x,y)$ is equivalent to:

$$\nabla f(x,y) - \lambda \nabla g(x,y) = 0$$

To make things a little easier, we notice that if we define a function:

$L(x,y,\lambda) = f(x,y) - \lambda g(x,y)$ then its gradient is:

$$\nabla L(x,y,\lambda) = \nabla f(x,y) - \lambda \nabla g(x,y)$$

This function L is called the Lagrangian, and solving for the gradient of the Lagrangian (solving $\nabla L(x,y,\lambda) = 0$) means finding the points where the gradient of f and g are parallels.

In the following slides, we use α instead of λ as Lagrange multiplier

(remember) The SVM Dual Problem

The primal SVM problem was given as

$$\min_{\boldsymbol{\theta}} \frac{1}{2} \sum_{j=1}^{d} \theta_j^2$$

$$\text{s.t. } y_i(\boldsymbol{\theta}^\intercal \mathbf{x}_i) \geq 1 \quad \forall i$$

Can solve it more efficiently by taking the Lagrangian dual

- Duality is a common idea in optimization
- It transforms a difficult optimization problem into a simpler one
- Key idea: introduce Lagrange multiplier α_i for each constraint
  - α_i indicates how important a particular constraint is to the solution

In mathematical optimization, the method of Lagrange multipliers is a strategy for finding the local maxima and minima of a function subject to equality constraints. (Wikipedia)

The SVM Dual Problem

- The Lagrangian is given by

$$L(\boldsymbol{\theta}, \boldsymbol{\alpha}) = \frac{1}{2}\sum_{j=1}^{d}\theta_j^2 - \sum_{i=1}^{n}\alpha_i(y_i\boldsymbol{\theta}^\intercal \mathbf{x}_i - 1)$$

$$\text{s.t. } \alpha_i \geq 0 \quad \forall i$$

- We must minimize over $\boldsymbol{\theta}$ and maximize over $\boldsymbol{\alpha}$
- At optimal solution, partials w.r.t $\boldsymbol{\theta}$'s are 0

$$\sum_i \alpha_i y_i = 0 \qquad \boldsymbol{\theta} = \sum_{i=1}^{n}\alpha_i y_i x_i$$

- Plug the above two functions into the Lagrangian…

https://www.svm-tutorial.com/2016/09/duality-lagrange-multipliers/

SVM Dual Representation

Maximize $$J(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

$$\text{s.t. } \alpha_i \geq 0 \quad \forall i$$

$$\sum_i \alpha_i y_i = 0 \qquad \boldsymbol{\theta} = \sum_{i=1}^{n} \alpha_i y_i x_i$$

Understanding the Dual

Maximize $$J(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

s.t. $\alpha_i \geq 0 \quad \forall i$

$$\sum_i \alpha_i y_i = 0$$

Balances between the weight of constraints for different classes

Constraint weights (α_i's) cannot be negative

Understanding the Dual

Maximize $$J(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

$$\text{s.t. } \alpha_i \geq 0 \quad \forall i$$

$$\sum_i \alpha_i y_i = 0$$

Points with different labels increase the sum

Points with same label decrease the sum

Measures the similarity between points

Intuitively, we should be more careful around points near the margin

Understanding the Dual

Maximize $$J(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

$$\text{s.t. } \alpha_i \geq 0 \quad \forall i$$

$$\sum_i \alpha_i y_i = 0$$

In the solution, either:

- $\alpha_i > 0$ and the constraint is tight $(y_i(\boldsymbol{\theta}^\intercal \mathbf{x}_i) = 1)$
  - ➢ point is a support vector
- $\alpha_i = 0$
  - ➢ point is not a support vector

Example of Solving the Dual

- Refer to Flach: 7.3: pages 215 - 216

Employing the Solution

- Given the optimal solution **α***, optimal weights are

$$\boldsymbol{\theta}^\star = \sum_{i \in SVs} \alpha_i^\star y_i \mathbf{x}_i$$

  - In this formulation, have *not* added $x_0 = 1$

- Therefore, we can solve one of the SV constraints

$$y_i(\boldsymbol{\theta}^\star \cdot \mathbf{x}_i + \theta_0) = 1$$

  to obtain θ_0

  - Or, more commonly, take the average solution over all support vectors