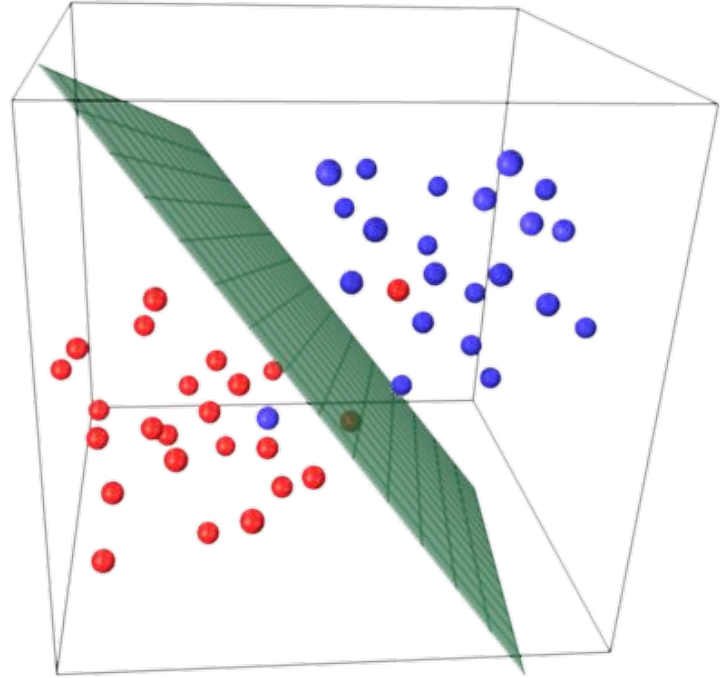


# Support Vector Machines

DSI SEA5, jf.omhover, Sep 30 2016

*a posteriori version (spoiler)*



# Support Vector Machines

DSI SEA5, [jf.omhover](http://jf.omhover.com), Sep 30 2016



## STANDARDS

- Compute a hyperplane as a decision boundary in SVC
- Explain what a support vector is in plain english
- Tune a SVC or SVM using their hyperparameters
- State what happens to bias and variance if we tune these hyperparameters
- State how “one-vs-one” and “one-vs-rest” approaches for multi-class problems are implemented.

# Support Vector Machines

DSI SEA5, [jf.omhover](http://jf.omhover), Sep 30 2016



## OBJECTIVES

- **Understand** the notion of decision boundaries
- **Describe** the function and parameters of SVMs
- **Investigate** some of the maths behind SVMs
- **Extend** SVMs by soft margins and kernel tricks
- **Investigate** how SVMs perform in terms of Bias-Variance
- Get your **mind blown**

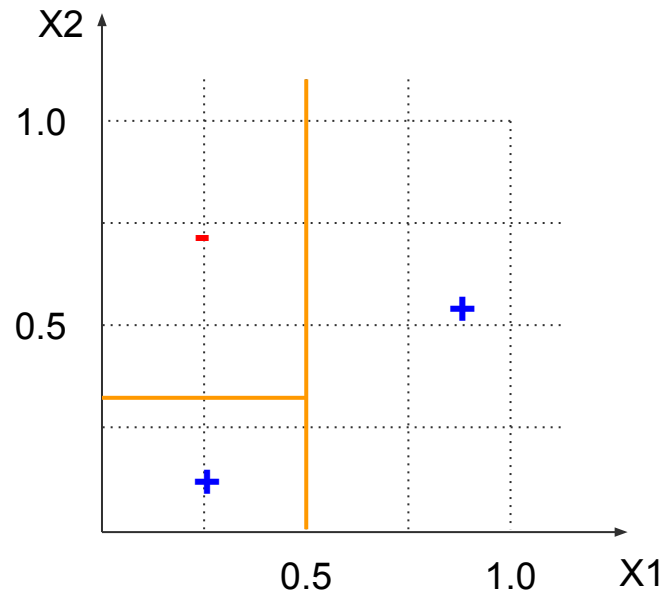
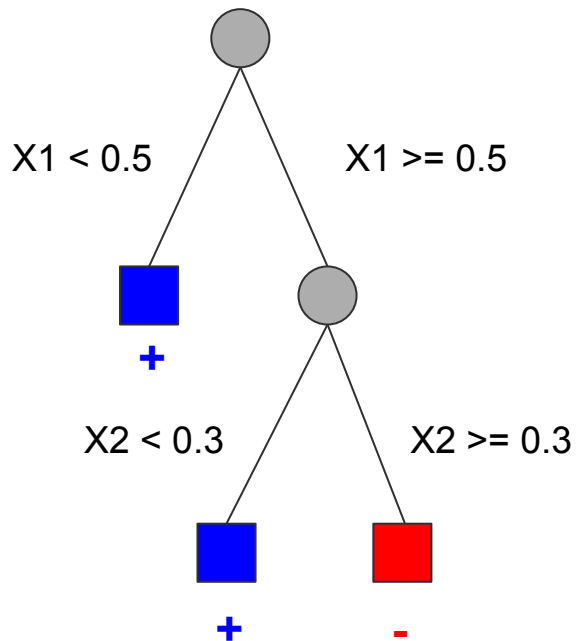


# Decision Boundaries (review)

# Draw the decision boundaries for... DT



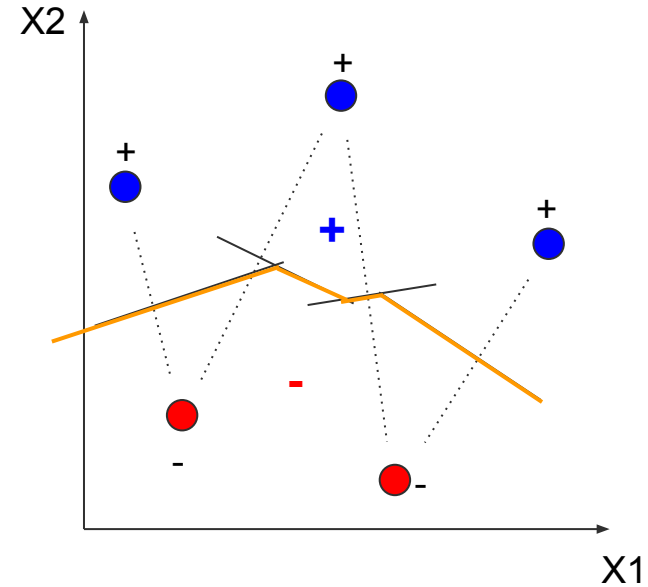
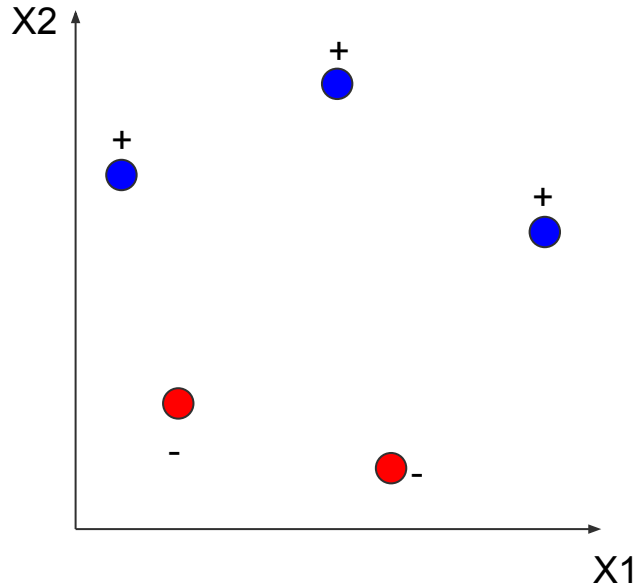
*find this slide in the a-posteriori version*



# Draw the decision boundaries for... 1-NN



*find this slide in the a-posteriori version*



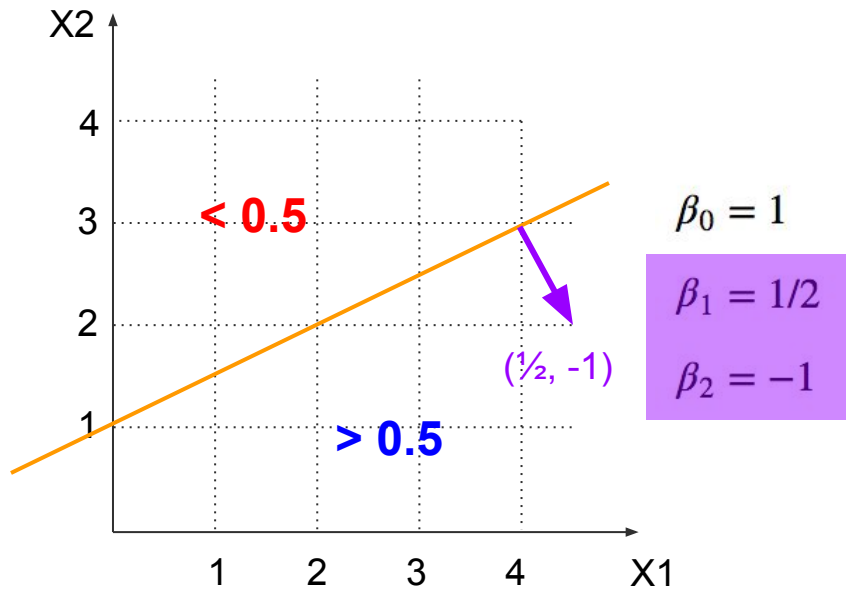
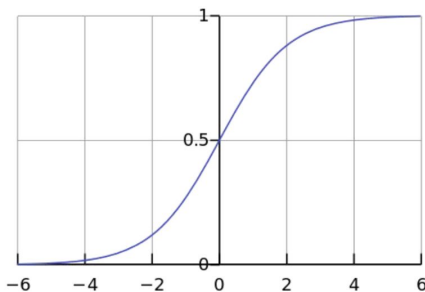
# Draw the decision boundaries for... LogReg



$$p(X) = h(\beta_0 + \beta_1 \cdot x_1 + \dots + \beta_p \cdot x_p)$$

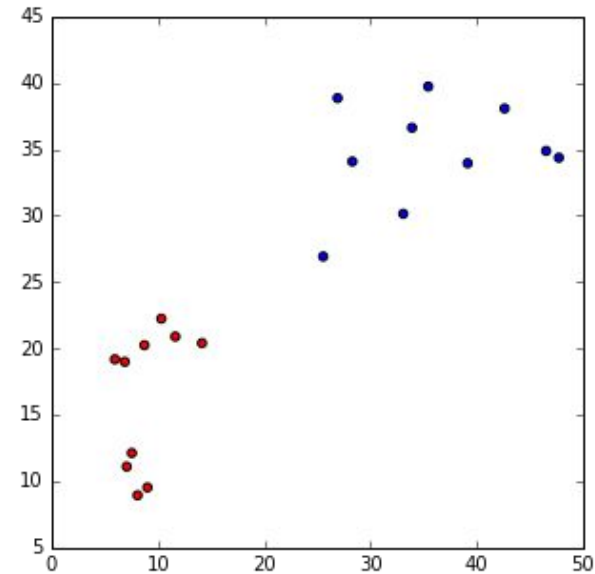
$$h : \mathbb{R} \rightarrow [0, 1]$$

$$h(t) = \frac{1}{1+e^{-t}}$$



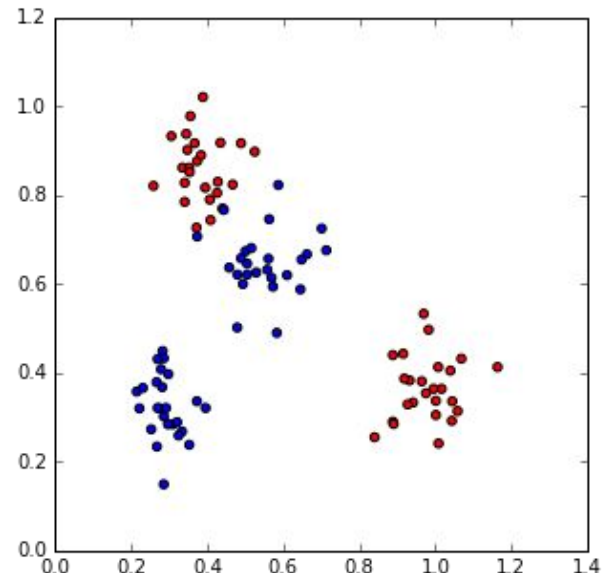
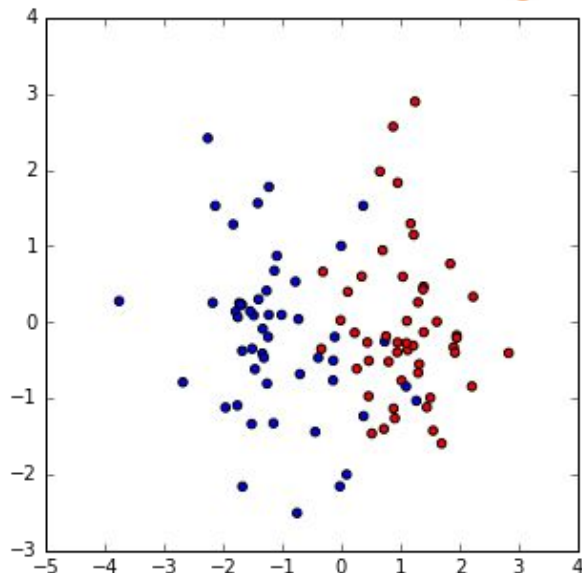
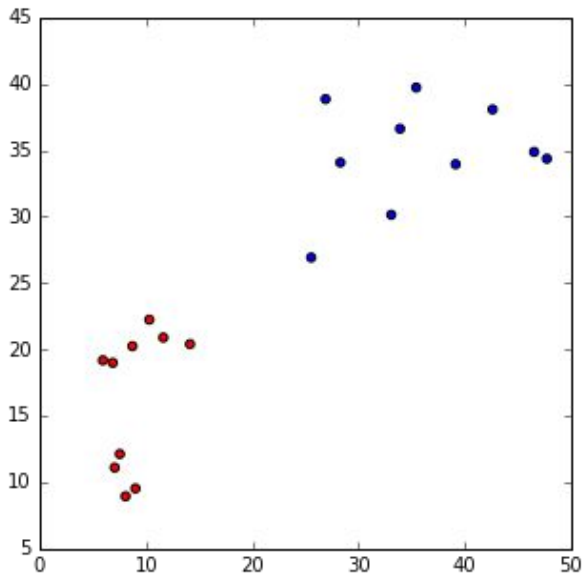
$$\beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 = 0 \implies x_2 = -\frac{\beta_0}{\beta_2} - \frac{\beta_1}{\beta_2} \cdot x_1$$

# Brainstorm : what's a good decision boundary ?





# Brainstorm : what's a good decision boundary ?





# Re-Formalizing Classification as a separation problem

# Reality VS Model



## REALITY

	type	income	education	prestige
accountant	prof	62	86	82
pilot	prof	72	76	83
architect	prof	75	92	90
author	prof	55	90	76
chemist	prof	64	86	90
minister	prof	21	84	87
professor	prof	64	93	93
dentist	prof	80	100	90
reporter	wc	67	87	52
engineer	prof	72	86	88
undertaker	prof	42	74	57
lawyer	prof	76	98	89

data

$(x_1, y_1)$

...

$(x_n, y_n)$

$x \ y$

**OBJECTIVE:**  
descriptive  
predictive  
normative

...

COST FUNCTION

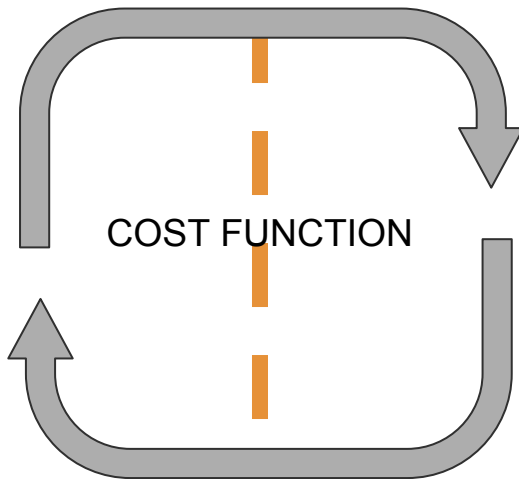
## MODEL

$$y = f(x) + \epsilon$$

take a function as  
an assumption

$$\hat{y} = \hat{f}(x)$$

Estimator  
of the function



# Supervised Learning : Classification



## REALITY

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{2,1} & \cdots & x_{2,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = y$$

Categorical output  
(classes)

## OBJECTIVE:

descriptive  
predictive  
normative  
...

COST FUNCTION

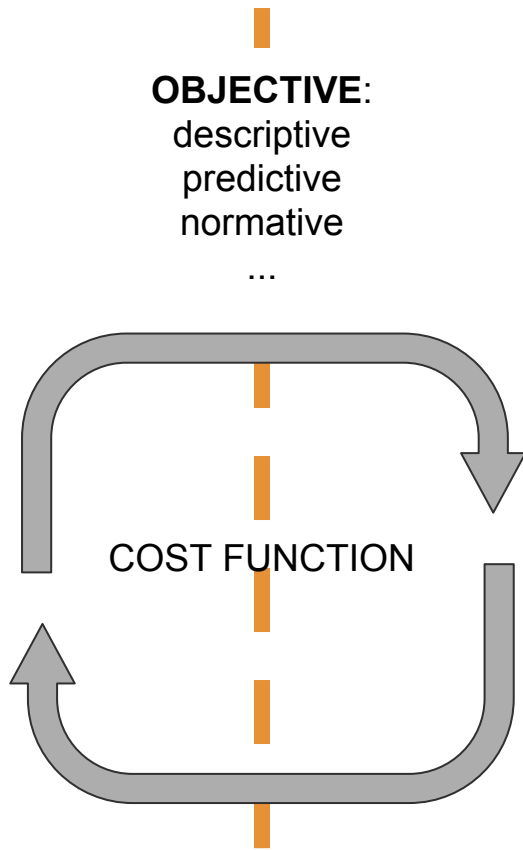
## MODEL

$$y = f(x) + \epsilon$$

take a function as  
an assumption

$$\hat{y} = \hat{f}(x)$$

Estimator  
of the function



# Classification : how LogReg does it



## REALITY

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{2,1} & \cdots & x_{2,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = y$$

$$\forall i, y_i \in \{0, 1\}$$

Binary output  
(two classes)

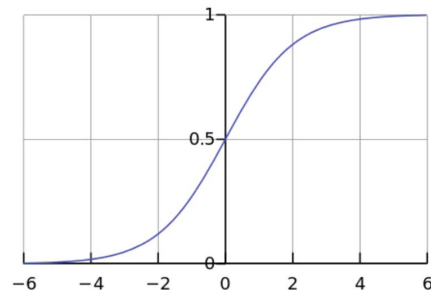
## MODEL

find/estimate betas such as

$$p(X) = h(\beta_0 + \beta_1 \cdot x_1 + \cdots + \beta_p \cdot x_p)$$

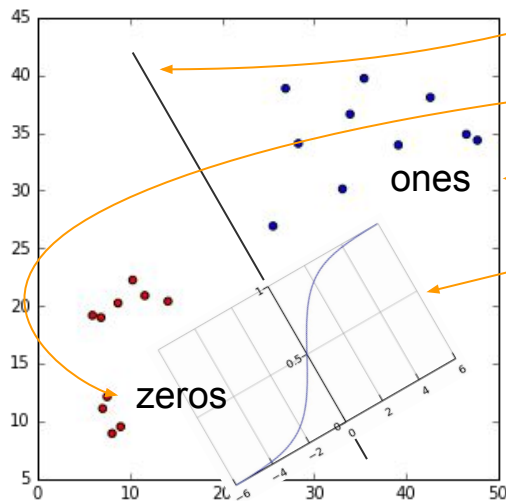
$$h : \mathbb{R} \rightarrow [0, 1]$$

$$h(t) = \frac{1}{1+e^{-t}}$$



# Classification : how LogReg shows in sample space

## REALITY



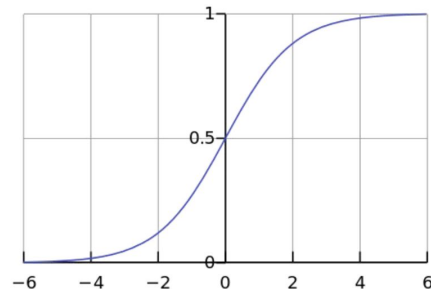
*It (badly) translates as :  
computes the probability  
of being in one of the two  
classes  
depending on of the side  
and distance of the plan*

## MODEL

$$p(X) = h(\beta_0 + \beta_1 \cdot x_1 + \dots + \beta_p \cdot x_p)$$

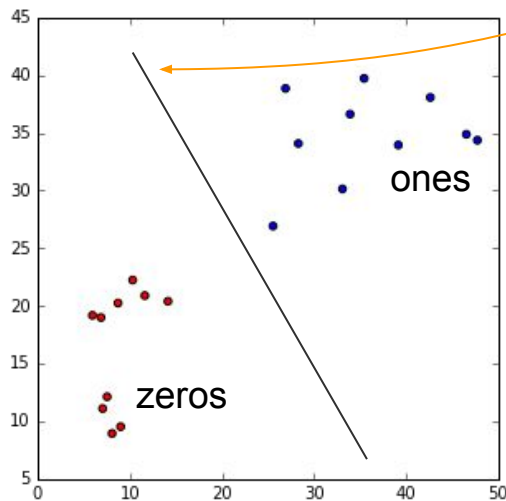
$$h : \mathbb{R} \rightarrow [0, 1]$$

$$h(t) = \frac{1}{1+e^{-t}}$$



# Classification : let's strip LogReg from probabilities

**REALITY**

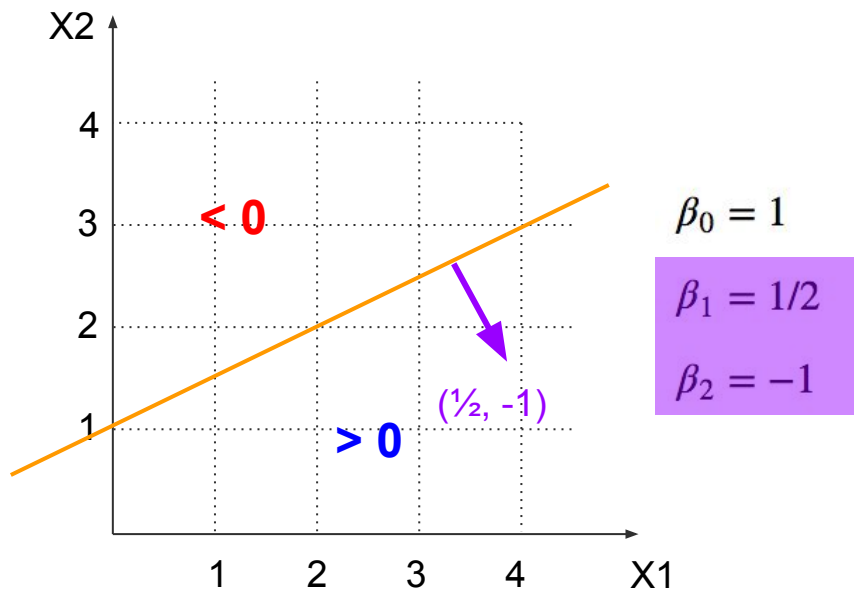


*It (badly) translates as :  
you're in one class or the  
other  
depending on of the side  
and distance of the plan*

**MODEL**

$$(\beta_0 + \beta_1 \cdot x_1 + \dots + \beta_p \cdot x_p) > 0$$

# Solutions to a linear equations (2D)



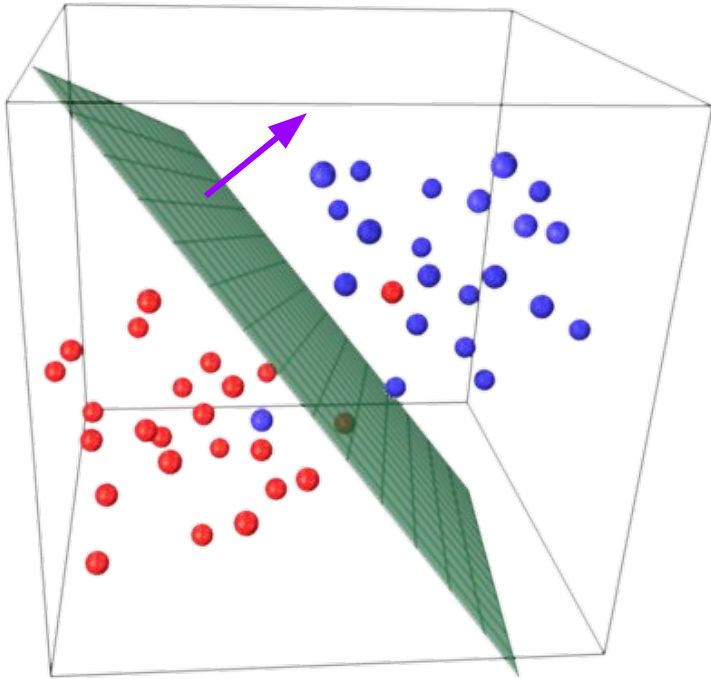
$$\beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 = 0 \implies x_2 = -\frac{\beta_0}{\beta_2} - \frac{\beta_1}{\beta_2} \cdot x_1$$

$$(\beta_0 + \beta_1 \cdot x_1 + \dots + \beta_p \cdot x_p) > 0$$

# Hyperplane !

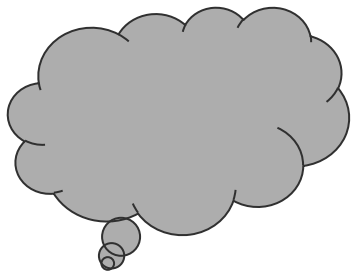


# Solutions to a linear equations (3D)



$$(\beta_0 + \beta_1 \cdot x_1 + \dots + \beta_p \cdot x_p) > 0$$

**Hyperplane !**



$$(\beta_0 + \beta_1 \cdot x_1 + \dots + \beta_p \cdot x_p) > 0$$

# Hyperplane !

(Has dimension N-1)

# Classification as a hyperplane pb

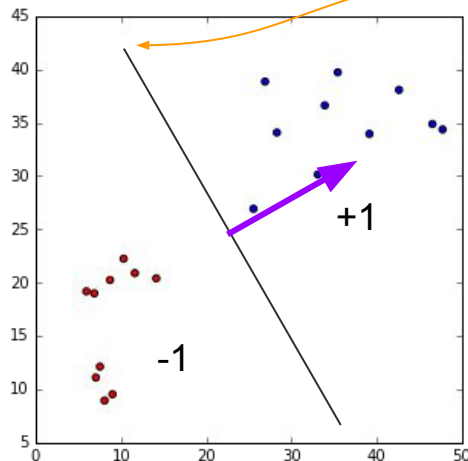


## REALITY

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{2,1} & \cdots & x_{2,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = y$$

$$\forall i, y_i \in \{-1, 1\}$$

Binary output  
(two classes)



## MODEL

find/estimate betas such as

$$y_i = +1, \beta_0 + \beta_1 \cdot x_{i,1} + \cdots + \beta_p \cdot x_{i,p} \geq 0$$

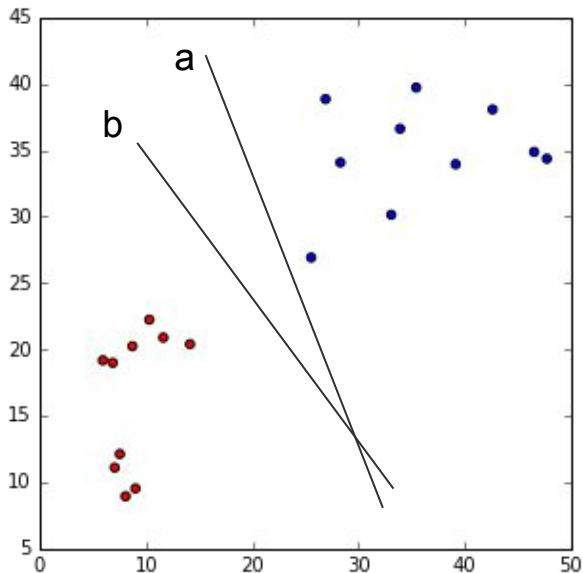
$$y_i = -1, \beta_0 + \beta_1 \cdot x_{i,1} + \cdots + \beta_p \cdot x_{i,p} < 0$$

or, simply put...

$$y_i \cdot (\beta_0 + \beta_1 \cdot x_{i,1} + \cdots + \beta_p \cdot x_{i,p}) \geq 0$$

$$y_i \cdot (\beta_0 + x_i^T \cdot \beta) \geq 0$$

# Brainstorm : what's a best decision boundary ?



Between boundary a and b, I choose b because...

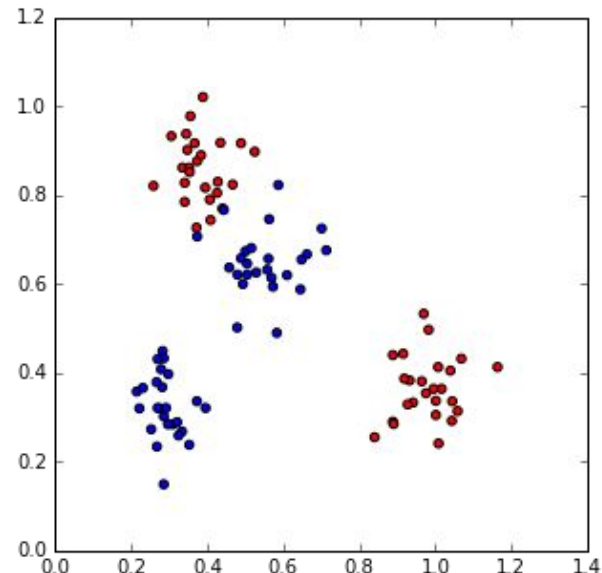
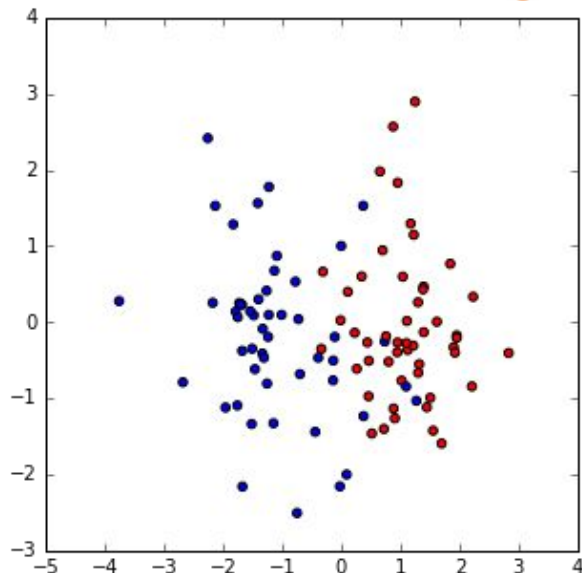
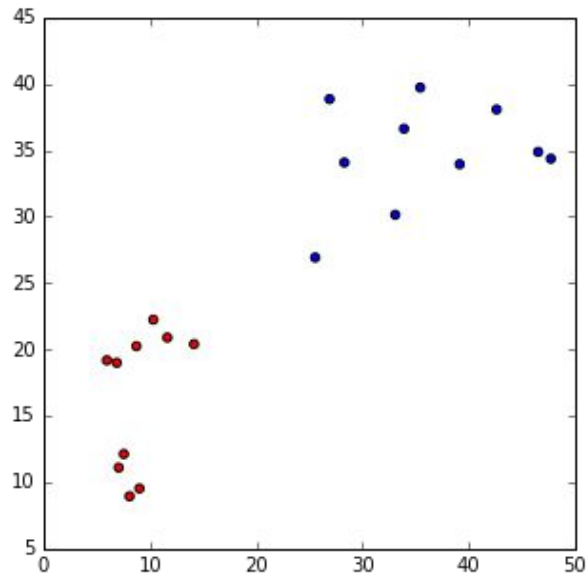
(or)

I bet b would win over a in a k-fold contest because...

# Brainstorm : what's a good decision boundary ?



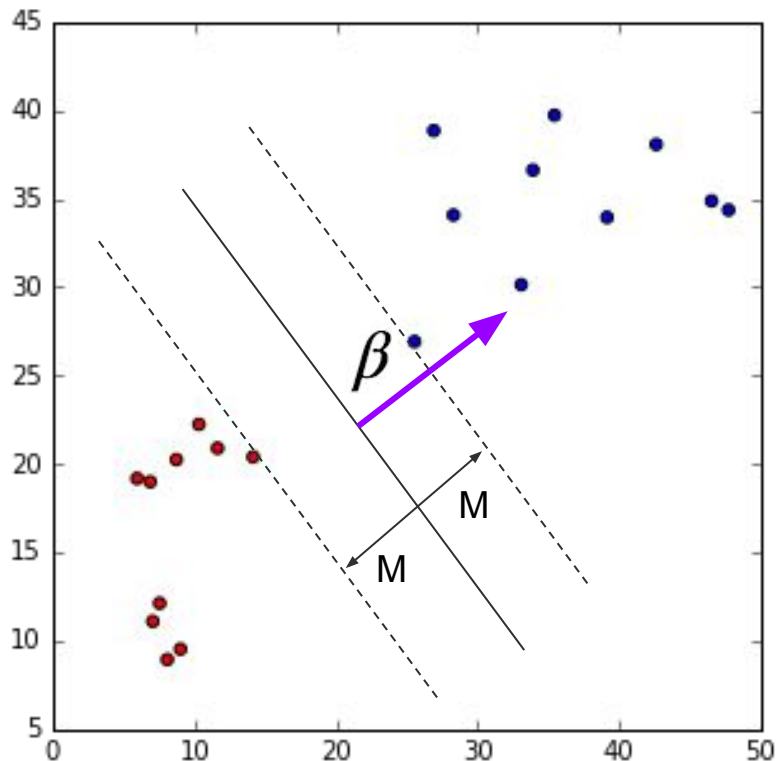
MMC





# Maximum Margin Classification

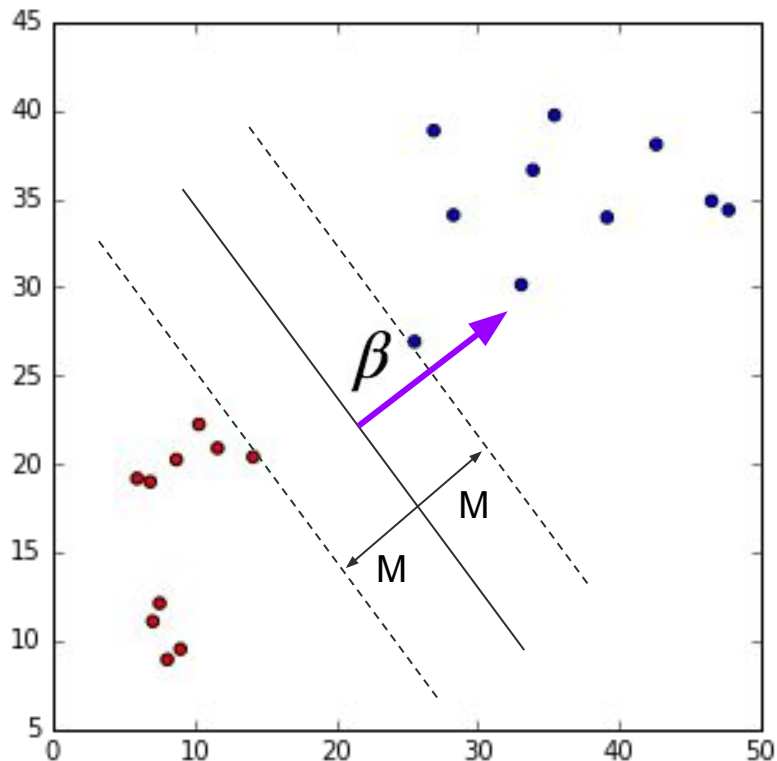
# What's Margin ?



The distance from the hyperplane to the nearest training data point.

We'd like to find a hyperplane that maximizes that margin !

# Maximum Margin Classification



$$\max_{\beta_0, \dots, \beta_p} M$$

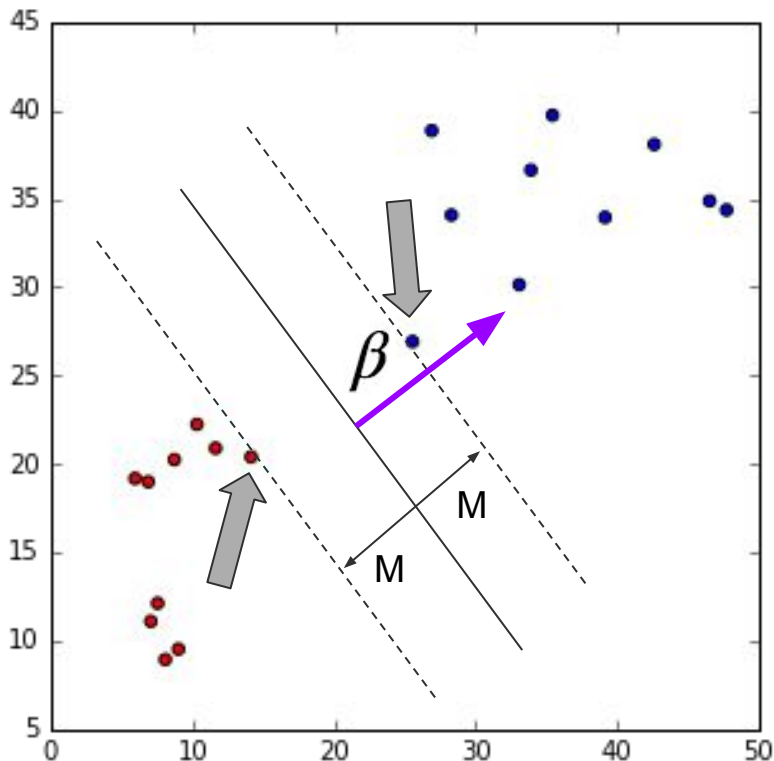
$$\text{subject to } \sum_{i=1}^p \beta_i^2 = 1$$

$$y_i \cdot (\beta_0 + \beta_1 \cdot x_{i,1} + \dots + \beta_p \cdot x_{i,p}) \geq M$$

$$y_i \cdot (\beta_0 + x_i^T \cdot \beta) \geq M$$



# Maximum Margin Classification / Support Vectors



$$\max_{\beta_0, \dots, \beta_p} M$$

$$\text{subject to } \sum_{i=1}^p \beta_i^2 = 1$$

$$y_i \cdot (\beta_0 + \beta_1 \cdot x_{i,1} + \dots + \beta_p \cdot x_{i,p}) \geq M$$

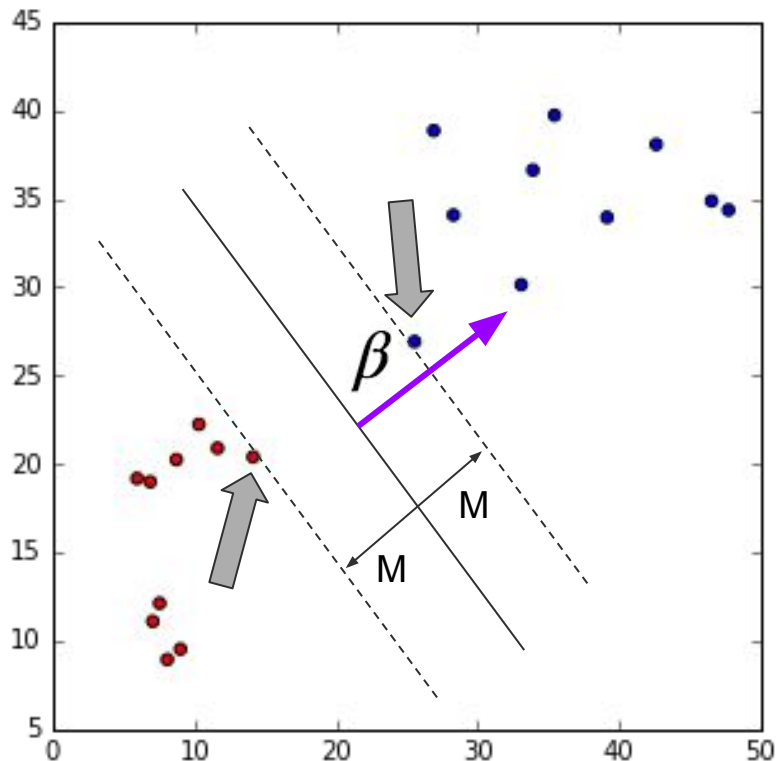
$$y_i \cdot (\beta_0 + x_i^T \cdot \beta) \geq M$$

Points that condition the margin.

Points that have a direct influence on the margin.

Points that end up being the closest to the hyperplane.

# MMC and Scaling...



$$\max_{\beta_0, \dots, \beta_p} M$$

subject to  $\sum_{i=1}^p \beta_i^2 = 1$

$$y_i \cdot (\beta_0 + \beta_1 \cdot x_{i,1} + \dots + \beta_p \cdot x_{i,p}) \geq M$$

$$y_i \cdot (\beta_0 + x_i^T \cdot \beta) \geq M$$

**Pre-Scaling of the data is necessary**



# Individual Assignment

# Support Vector Machines

DSI SEA5, [jf.omhover](http://jf.omhover), Sep 30 2016



## OBJECTIVES

- **Understand** the notion of decision boundaries
- **Describe** the function and parameters of SVMs
- **Investigate** some of the maths behind SVMs
- **Extend** SVMs by soft margins and kernel tricks
- **Investigate** how SVMs perform in terms of Bias-Variance
- Get your **mind blown**