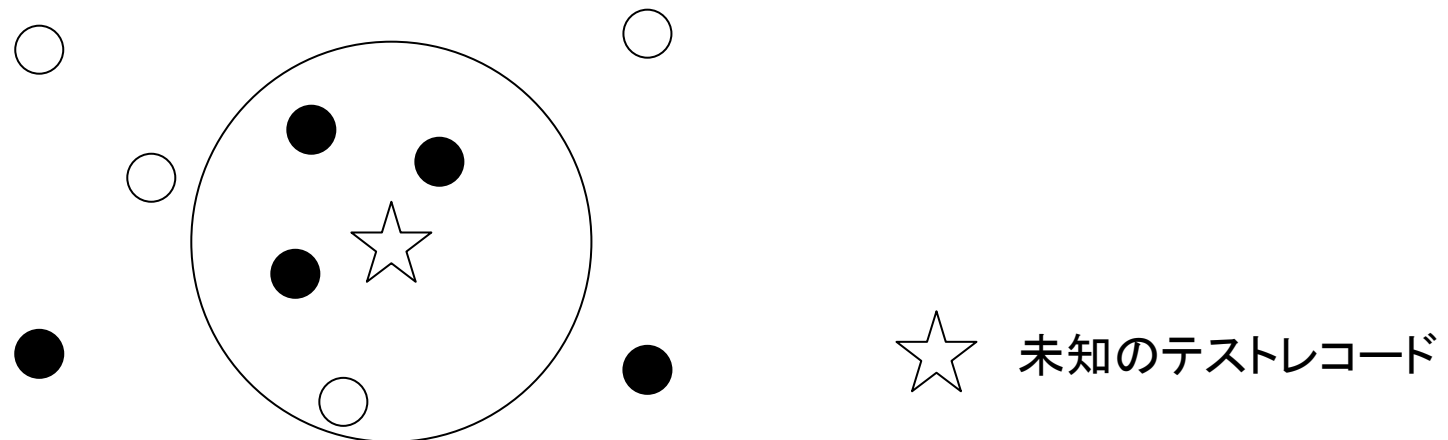


# $k$ nearest neighbors

- 規則を学習しない
- 未知のテストレコードと最も類似した(適切な距離を定義し、近い距離にある) $k$ 個の訓練レコードの集合  $S$  を求める
- 未知のテストレコードの目標属性値を  $S$  から予測  
例 2値の場合:  $S$  の目標属性値の多数決  
数値の場合: 平均値
- レコード間の類似性を適切に定義することが重要で  
ユークリッド距離やハミング距離など
- $k$  の選択も学習能力を左右



# ナイーブベイズ分類

条件付確率

$$P(A | T) = P(A \& T) / P(T)$$

$$P(T | A) = P(A \& T) / P(A)$$

$$P(A \& T) = P(A | T) P(T) = P(T | A) P(A)$$

ベイズの定理

$$P(A | T) = \frac{P(T | A) P(A)}{P(T)}$$

事後確率

(条件TのもとAが観測される確率)

尤度

(Aが観測されたとき、それを説明する条件 T の尤もらしさ)

事前確率

### 訓練データ

T1	T2	T3	T4	目標属性A
1	0	1	1	1
1	0	1	1	1
1	1	1	1	1
1	1	0	0	0
:				

### 未知のテストデータに対して

クラス分類器  $T1=v1, \dots, T4=v4$  が目標属性値  $w$  を予測する(事後)確率をベイズの定理より推定

$$P(A=w \mid T1=v1, \dots, T4=v4) = \frac{P(A=w) P(T1=v1, \dots, T4=v4 \mid A=w)}{P(T1=v1, \dots, T4=v4)}$$

尤度(目標属性Aの値  $w$  が観測されたとき、それを説明する条件  $T1=v1, \dots, T4=v4$  の尤もらしさを示す)

$T1=v1, \dots, T4=v4$  は  $w$  の値として 1 と 0 どちらを推定するか?

事後確率の推定 尤度を条件付確率を利用して変形

$$P(T1 \& T2 \mid A) = P(T1 \mid A) P(T2 \mid T1 \& A)$$

$$\begin{aligned} & P(T1=v1, T2=v2, T3=v3, T4=v4 \mid A=w) \\ &= P(T1=v1 \mid A=w) P(T2=v2, T3=v3, T4=v4 \mid T1=v1, A=w) \\ &= P(T1=v1 \mid A=w) \\ & \quad P(T2=v2 \mid T1=v1, A=w) P(T3=v3, T4=v4 \mid T2=v2, T1=v1, A=w) \\ &= P(T1=v1 \mid A=w) P(T2=v2 \mid T1=v1, A=w) \\ & \quad P(T3=v3 \mid T2=v2, T1=v1, A=w) P(T4=v4 \mid T3=v3, T2=v2, T1=v1, A=w) \\ &\doteq P(T1=v1 \mid A=w) P(T2=v2 \mid A=w) P(T3=v3 \mid A=w) P(T4=v4 \mid A=w) \end{aligned}$$

**条件A=wのもと**  $T1=v1, \dots, T4=v4$  が互いに独立であることを「ナイーブに」仮定 (ナイーブベイズ分類)  
例えば 独立性  $P(T1=v1 \& T2=v2 \mid A=w) = P(T1=v1 \mid A=w) P(T2=v2 \mid A=w)$  と  
 $P(T1=v1 \& T2=v2 \mid A=w) = P(T1=v1 \mid A=w) P(T2=v2 \mid T1=v1 \& A=w)$  より  
 $P(T2=v2 \mid T1=v1, A=w) = P(T2=v2 \mid A=w)$  が導かれる

訓練データ

T1	T2	T3	T4	目標属性A
1	0	1	1	1
1	0	1	1	1
1	1	1	1	1
1	1	1	0	0
1	0	1	0	0
1	1	0	1	0
1	0	0	1	0
1	1	0	1	0
0	1	0	1	1
0	0	1	1	1
0	1	0	1	1
0	1	0	1	1
0	0	0	1	1
0	0	1	0	0
0	1	0	0	0
0	0	1	0	0

条件付確率  $P(T_i=v | A=w)$

		目標属性 A	
		1	0
	$v$	$w$	
T1	1	3/8	5/8
	0	5/8	3/8
T2	1	4/8	4/8
	0	4/8	4/8
T3	1	4/8	4/8
	0	4/8	4/8
T4	1	8/8	3/8
	0	0/8	5/8

尤度

$$\begin{aligned}
 & P(T1=0, T2=0, T3=0, T4=1 | A=1) \\
 &= P(T1=0|A=1) P(T2=0|A=1) P(T3=0|A=1) P(T4=1|A=1) \\
 &= (5/8) (4/8) (4/8) (8/8) \\
 & P(T1=0, T2=0, T3=0, T4=1 | A=0) \\
 &= (3/8) (4/8) (4/8) (3/8)
 \end{aligned}$$

# ナニーブベイズ分類器

クラス分類器  $T_1=v_1, \dots, T_4=v_4$  が推定する目標属性値  $w$  は

事前確率  $\times$  尤度 =

$$P(A=w) P(T_1=v_1 | A=w) P(T_2=v_2 | A=w) P(T_3=v_3 | A=w) P(T_4=v_4 | A=w)$$

を最大にする  $w$  の値と定義する(右辺の値は訓練データから収集する)

例

$$\begin{aligned} & P(A=1) P(T_1=0|A=1) P(T_2=0|A=1) P(T_3=0|A=1) P(T_4=1|A=1) \\ &= (1/2) (5/8) (4/8) (4/8) (8/8) \end{aligned}$$

$$\begin{aligned} & > P(A=0) P(T_1=0|A=0) P(T_2=0|A=0) P(T_3=0|A=0) P(T_4=1|A=0) \\ &= (1/2) (3/8) (4/8) (4/8) (3/8) \end{aligned}$$

$T_1=0, T_2=0, T_3=0, T_4=1$  は  $A=1$  を推定

# Support Vector Machine

## 参考文献

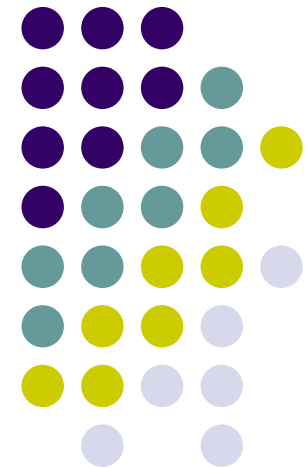
- C. Cortes and V. N. **Vapnik**, *Support-Vector Networks*, *Machine Learning*, 20(3):273-297, September 1995
- Vladimir N. **Vapnik**. *The Nature of Statistical Learning Theory*. Springer, New York, 1995

参考サイト <http://www.kernel-machines.org/>

参考 java applet <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

## 応用例

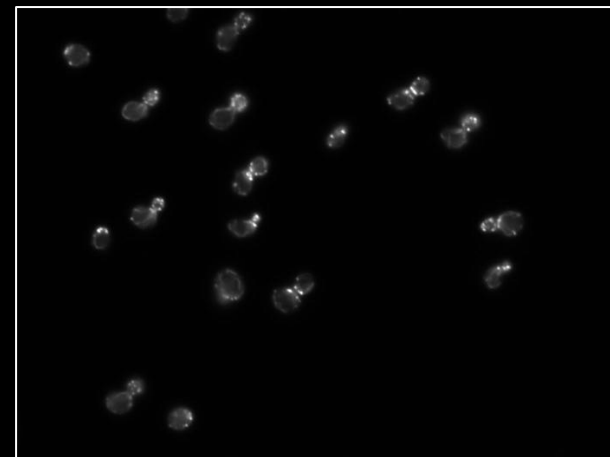
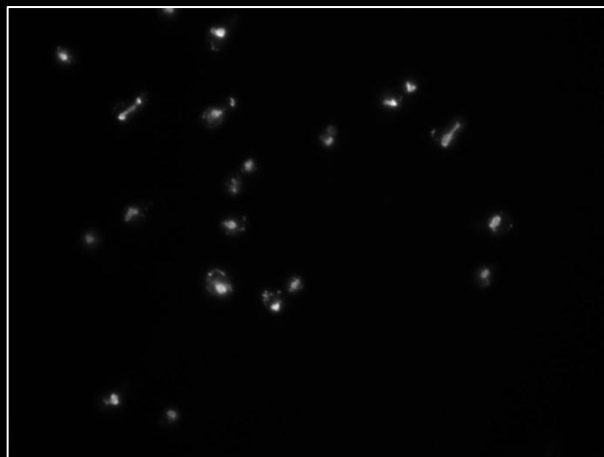
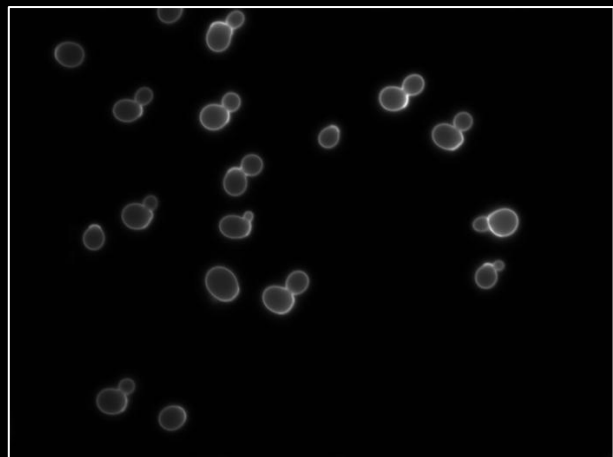
- マイクロアレーデータから遺伝子や組織を分類
- タンパク質の結合関係の予測
- 表現型からの遺伝子機能の予測



細胞壁 FITC-ConA

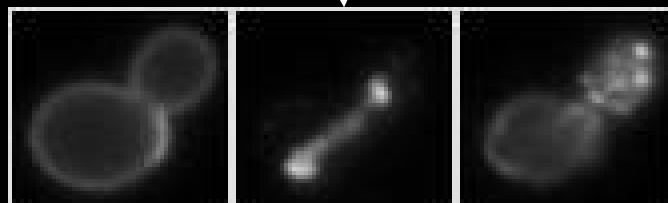
核 DAPI

アクチン Rh-ph

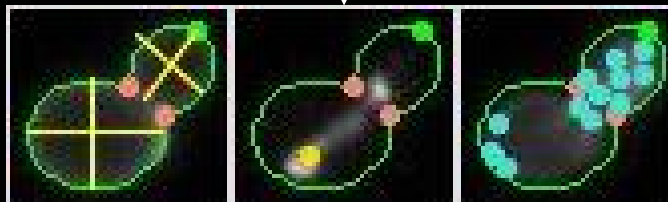


Superimposition

Phenome  
からの例

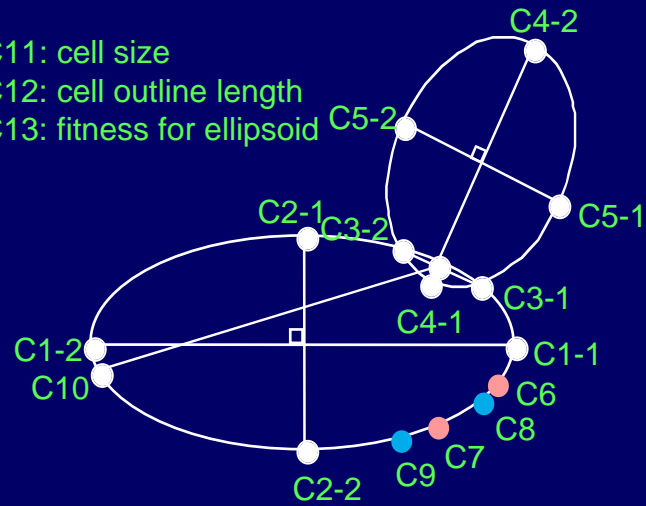


Cell-image Processing



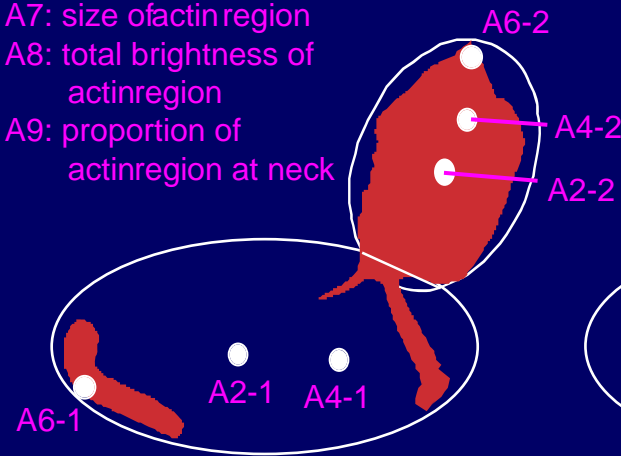
### FITC-ConA画像から

C11: cell size  
 C12: cell outline length  
 C13: fitness for ellipsoid



### Rh-ph画像から

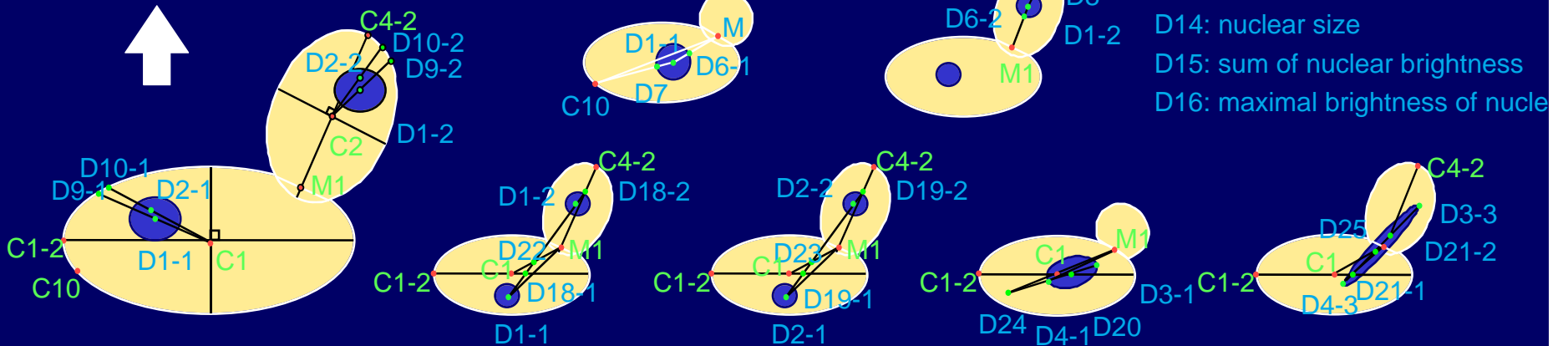
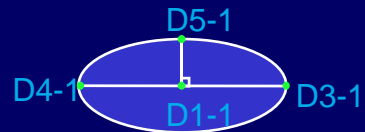
A7: size of actin region  
 A8: total brightness of actin region  
 A9: proportion of actin region at neck



A1: Number, location, size, brightness in each actin patch



### DAPI画像から



D14: nuclear size  
 D15: sum of nuclear brightness  
 D16: maximal brightness of nucle



# DNA修復や複製に関連する遺伝子

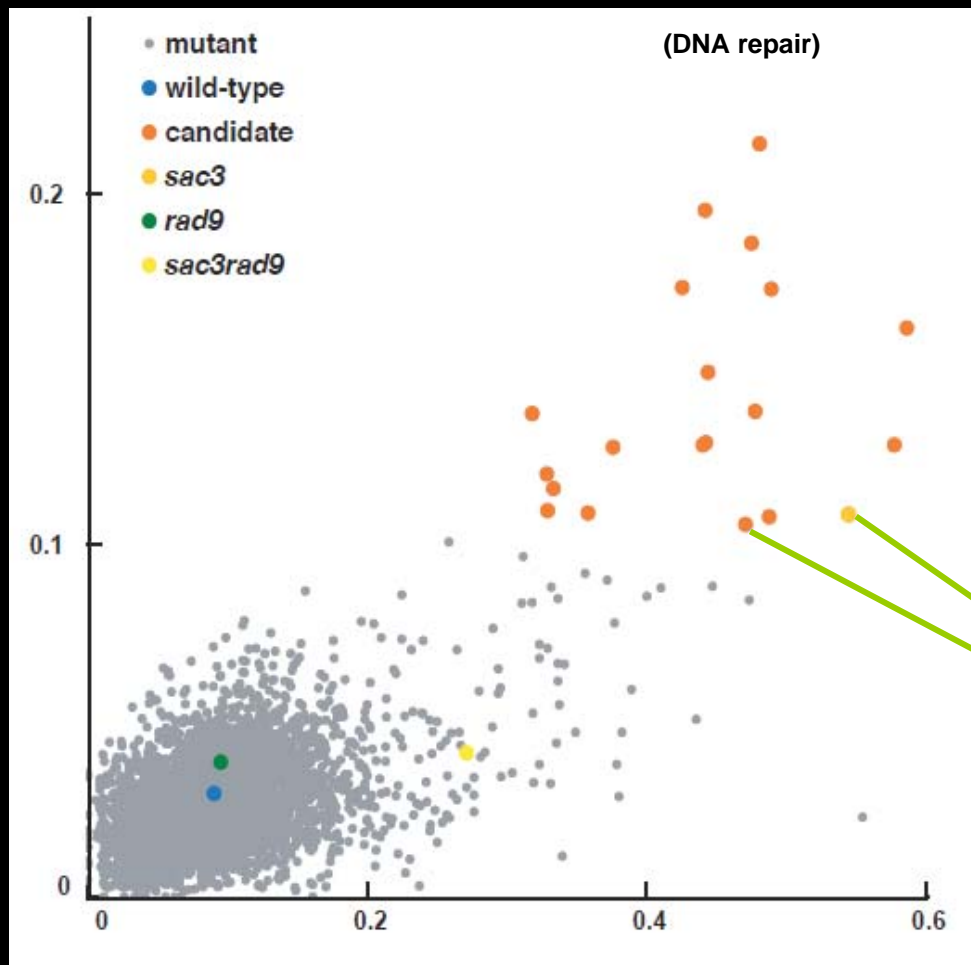
D201

Ohya *et al.* High-dimensional and large-scale phenotyping of yeast mutants. *PNAS* 102(52):19015-20 (2005)

高い



核が分裂中の細胞の比率



<i>ycr066w</i>	<i>rad18</i>	DNA repair
<i>ydI013w</i>	<i>hex3</i>	DNA recombination
* <i>ydr369c</i>	<i>xrs2</i>	Double strand break repair
* <i>yer095w</i>	<i>rad51</i>	DNA recombinase
* <i>ygl163c</i>	<i>rad54</i>	DNA repair
<i>yhr154w</i>	<i>rtt107</i>	DNA transposition
<i>yhr191c</i>	<i>ctf8</i>	DNA replication factor C com
<i>yjl115w</i>	<i>asf1</i>	DNA damage response
<i>yII002w</i>	<i>rtt109</i>	DNA transposition
<i>yIrl233c</i>	<i>est1</i>	Telomere length regulation
* <i>yml032c</i>	<i>rad52</i>	DNA recombinase
<i>yml078c</i>	<i>ctf18</i>	DNA replication factor C com
* <i>yml224c</i>	<i>mre11</i>	Double strand break repair
* <i>ynl250w</i>	<i>rad50</i>	Double strand break repair
<i>ypr135w</i>	<i>ctf4</i>	DNA polymerase a binding p
<i>ypr164w</i>	<i>mms1</i>	DNA repair
<i>ydr159w</i>	<i>sac3</i>	Unknown
<i>yor080w</i>	<i>dia2</i>	Unknown

註: \*印は、予測に用いたGOの  
"recombinational\_repair"に属する遺伝子変異株



核の分裂が完了する前の母細胞に対する芽の大きさ → 大きい

C125\_A1B



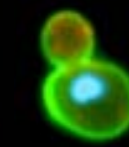
*ctf4*



*est1*

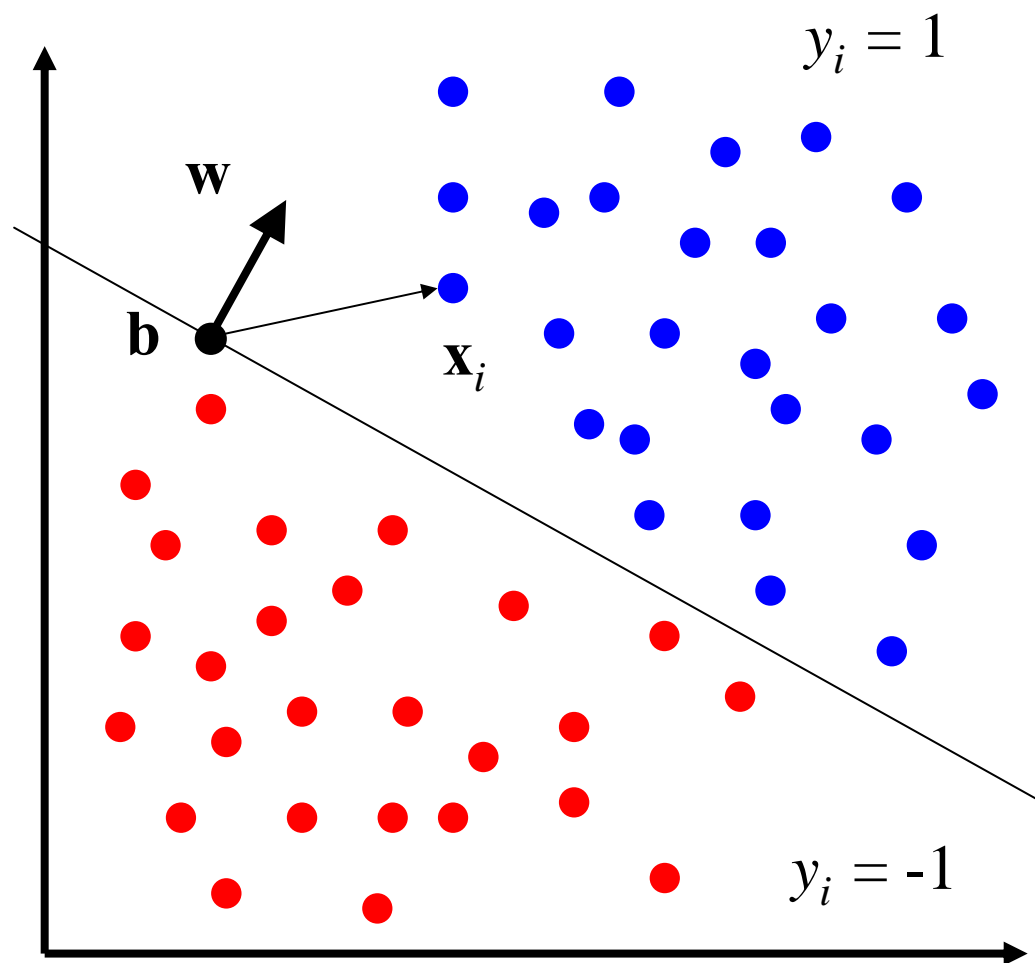


*rad27*



*his3*

# Fisher の線形判別



訓練データ

$$(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l) \quad \mathbf{x}_i \in R^n, y_i \in \{-1, +1\}$$

が線形分離可能とは、  
ベクトル  $\mathbf{w}$  と  $\mathbf{b}$  が存在し

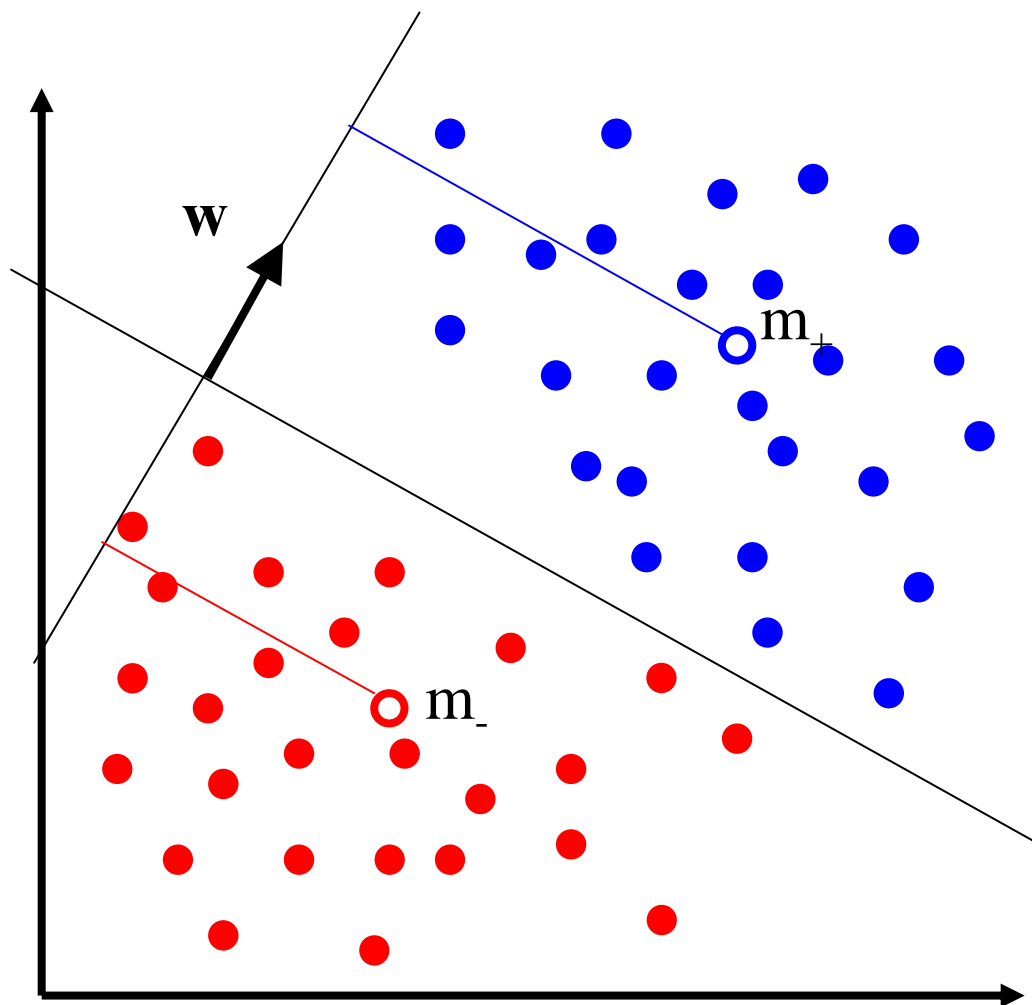
$$y_i(\mathbf{w} \cdot (\mathbf{x}_i - \mathbf{b})) > 0 \quad (i = 1, \dots, l)$$

$\mathbf{b}$  の自由度は不必要に多い  
-  $\mathbf{w} \cdot \mathbf{b}$  を定数  $b$  で表現すれば、

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) > 0 \quad (i = 1, \dots, l)$$

を満たす  $\mathbf{w}$  と  $b$  が存在することと同値

# Fisher の線形判別



線形判別関数をつかったクラス分類  $d(\mathbf{x})$

$$d(\mathbf{x}) = \begin{cases} 1 & \mathbf{w} \cdot \mathbf{x} + b \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

正および負に分類されたデータの平均値

$$\mathbf{m}_+ = \frac{\sum_{d(\mathbf{x})=1} \mathbf{x}}{|\{\mathbf{x} \mid d(\mathbf{x})=1\}|} \quad \mathbf{m}_- = \frac{\sum_{d(\mathbf{x})=-1} \mathbf{x}}{|\{\mathbf{x} \mid d(\mathbf{x})=-1\}|}$$

$\mathbf{m}_+$ ,  $\mathbf{m}_-$  を  $\mathbf{w}$  上に射影した点の距離の二乗

$$((\mathbf{m}_+ - \mathbf{m}_-) \cdot \mathbf{w})^2$$

を大きく、正と負のクラス内での点の分散

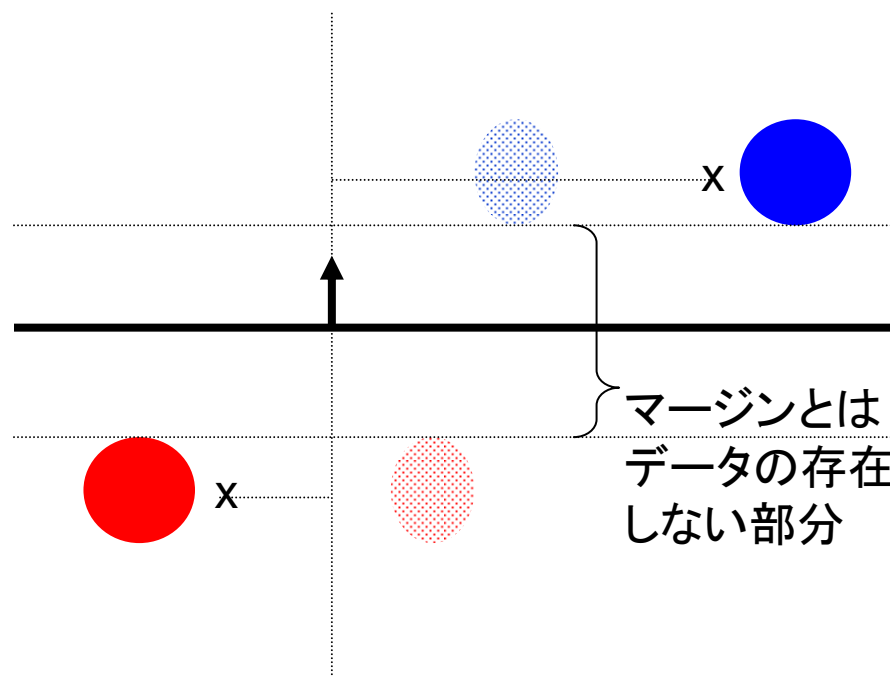
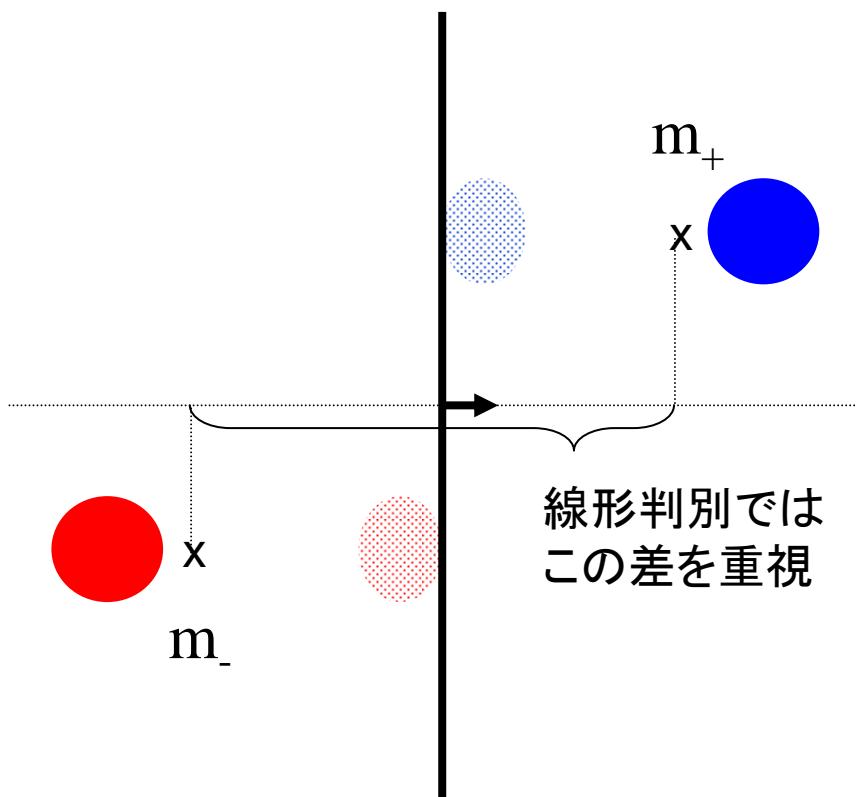
$$\sum_{d(\mathbf{x})=1} ((\mathbf{x} - \mathbf{m}_+) \cdot \mathbf{w})^2 + \sum_{d(\mathbf{x})=-1} ((\mathbf{x} - \mathbf{m}_-) \cdot \mathbf{w})^2$$

を小さくしたい. そこで

$$\frac{((\mathbf{m}_+ - \mathbf{m}_-) \cdot \mathbf{w})^2}{\sum_{d(\mathbf{x})=1} ((\mathbf{x} - \mathbf{m}_+) \cdot \mathbf{w})^2 + \sum_{d(\mathbf{x})=-1} ((\mathbf{x} - \mathbf{m}_-) \cdot \mathbf{w})^2}$$

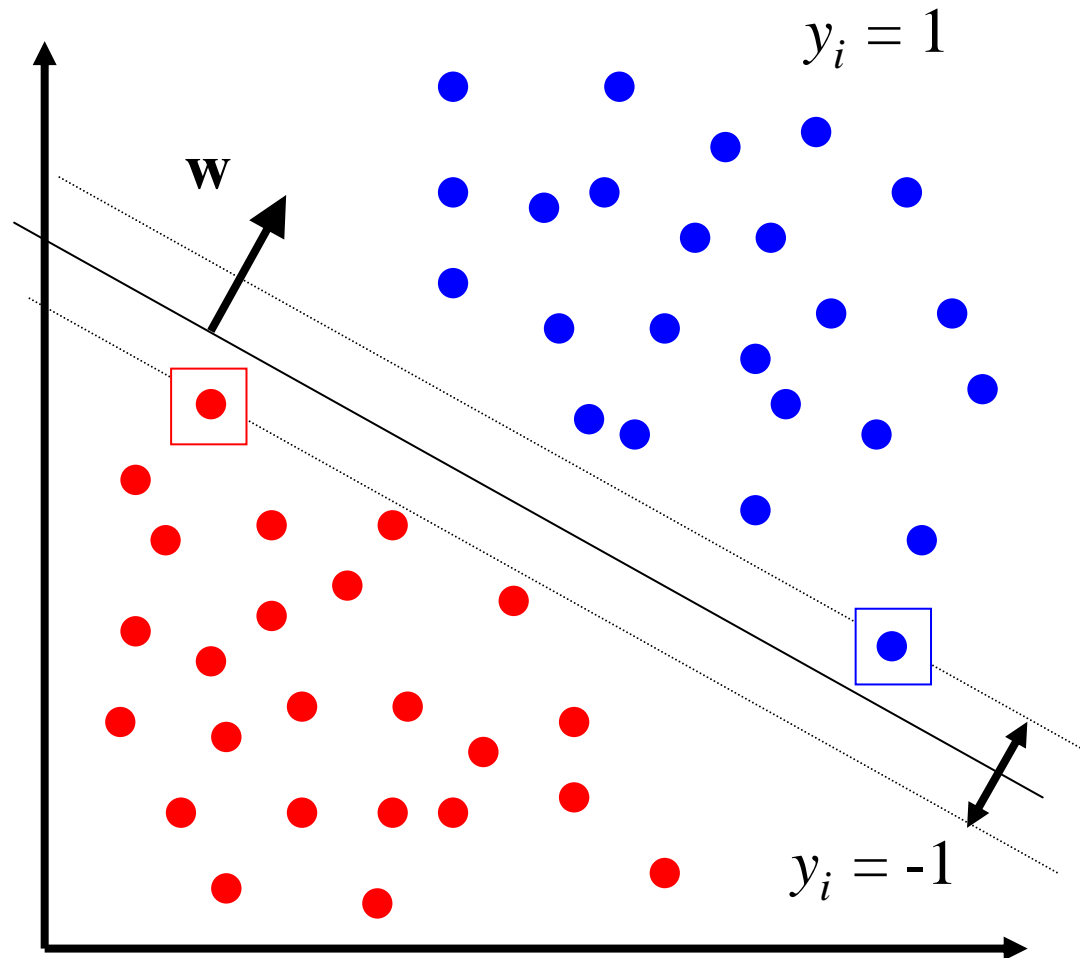
を最大にする  $\mathbf{w}$  を求める

# 線形判別とマージン



# Support Vector Machine

## マージン (Margin)



SVM はマージンを  
最大化するアプローチ

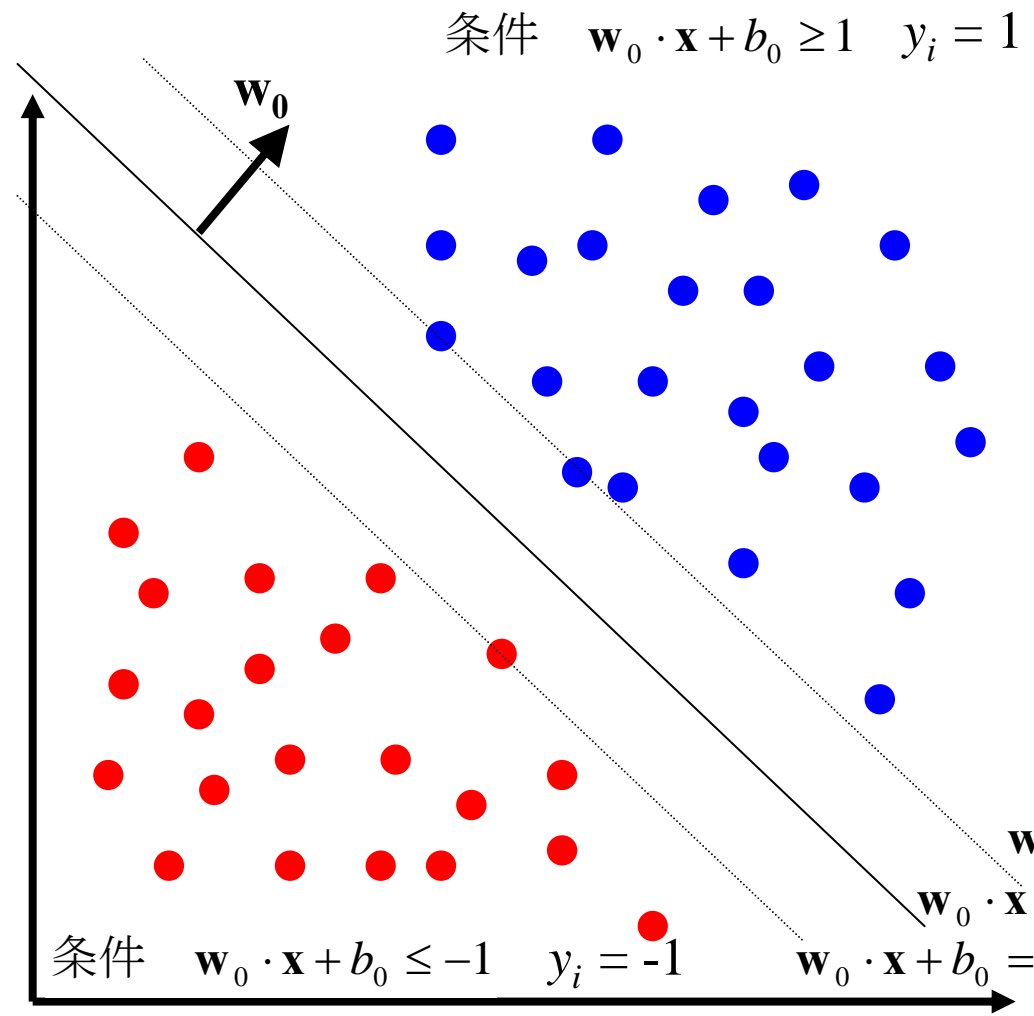
マージン境界上の点  
(support vector) に依存

マージン  $\rho(\mathbf{w}, b) =$

$$\min_{\{x_i | y_i=1\}} \frac{\mathbf{x}_i \cdot \mathbf{w}}{|\mathbf{w}|} - \max_{\{x_i | y_i=-1\}} \frac{\mathbf{x}_i \cdot \mathbf{w}}{|\mathbf{w}|}$$



# 最適超平面 (Optimal Hyperplanes)



境界線上に載る点が存在するように  $w_0$  の大きさを調節

$w_0$  がマージンを最大化する場合

$$\begin{aligned} \rho(w_0, b_0) &= \min_{\{x_i | y_i=1\}} \frac{x_i \cdot w_0}{|w_0|} - \max_{\{x_i | y_i=-1\}} \frac{x_i \cdot w_0}{|w_0|} \\ &= \frac{1-b_0}{|w_0|} - \frac{-1-b_0}{|w_0|} = \frac{2}{|w_0|} \end{aligned}$$

制約  $y_i(w \cdot x_i + b) \geq 1 \quad (i=1, \dots, l)$ のもと  $2/\sqrt{w \cdot w}$  を最大化



# 最適超平面: 2次計画問題へ

[2次計画問題]

制約  $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1$  ( $i = 1, \dots, l$ ) のもと  $2/\sqrt{\mathbf{w} \cdot \mathbf{w}}$  を最大化、  
つまり  $\mathbf{w} \cdot \mathbf{w}$  を最小化する  $\mathbf{w}_0$  が最適超平面をあたえる

ラグランジュの未定乗数法により解く

$\Lambda^T = (\alpha_1, \dots, \alpha_l)$  をラグランジュ乗数  $\alpha_i (\geq 0)$

$$L(\mathbf{w}, b, \Lambda) = \frac{1}{2} \mathbf{w} \cdot \mathbf{w} - \sum_{i=1}^l \alpha_i [y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1] \quad \dots \quad (1)$$

$\mathbf{w}$  と  $b$  について最小、 $\Lambda$  について最大となる saddle point(鞍点)が  $\mathbf{w}_0$  を与える



# 最適超平面： 双対問題へ

$\mathbf{w}, b$  について最小となる条件

$$\left. \frac{\partial L(\mathbf{w}, b, \Lambda)}{\partial \mathbf{w}} \right|_{\mathbf{w}=\mathbf{w}_0} = \mathbf{w}_0 - \sum_{i=1}^l \alpha_i y_i \mathbf{x}_i = 0 \quad \mathbf{w}_0 = \sum_{i=1}^l \alpha_i y_i \mathbf{x}_i \quad \dots (2)$$

$$\left. \frac{\partial L(\mathbf{w}, b, \Lambda)}{\partial b} \right|_{b=b_0} = -\sum_{i=1}^l \alpha_i y_i = 0 \quad \dots (3)$$

(1) へ代入

$$\begin{aligned} L(\mathbf{w}_0, b_0, \Lambda) &= \frac{1}{2} \mathbf{w}_0 \cdot \mathbf{w}_0 - \sum_{i=1}^l \alpha_i [y_i (\mathbf{x}_i \cdot \mathbf{w}_0 + b_0) - 1] \\ &= \frac{1}{2} \mathbf{w}_0 \cdot \mathbf{w}_0 - \boxed{\sum_{i=1}^l \alpha_i y_i \mathbf{x}_i} \cdot \mathbf{w}_0 - \boxed{\sum_{i=1}^l \alpha_i y_i} b_0 + \sum_{i=1}^l \alpha_i \\ &= \frac{1}{2} \mathbf{w}_0 \cdot \mathbf{w}_0 - \mathbf{w}_0 \cdot \mathbf{w}_0 + \sum_{i=1}^l \alpha_i \quad (2), (3) \\ &= \sum_{i=1}^l \alpha_i - \frac{1}{2} \mathbf{w}_0 \cdot \mathbf{w}_0 = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \end{aligned}$$

[双対問題]

制約  $\sum_{i=1}^l \alpha_i y_i = 0, \alpha_i \geq 0$  のもと

$$L(\mathbf{w}_0, b_0, \Lambda) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$$

を最大化する  $\Lambda^T = (\alpha_1, \dots, \alpha_l)$  を  
2次計画法で計算

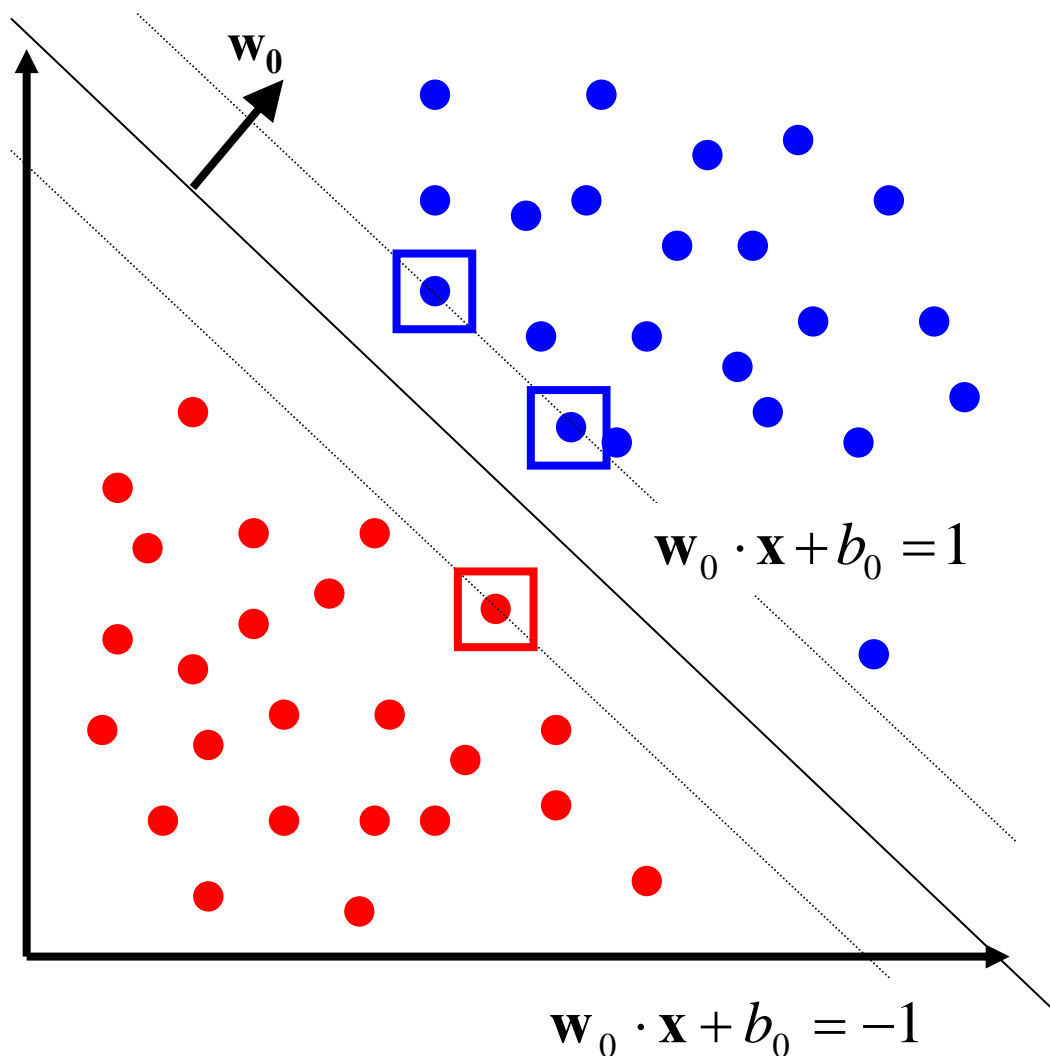
(2)  $\mathbf{w}_0 = \sum_{i=1}^l \alpha_i y_i \mathbf{x}_i$  より  $\alpha_i$  から  $\mathbf{w}_0$  を決定

すべての  $\mathbf{x}_i$  が必要か?





# 最適超平面: $w_0$ の計算



(2) より  $w_0$  は  $x_i$  の線形結合  $\sum_{i=1}^l \alpha_i y_i x_i$

すべての  $x_i$  が必要ではなくて

$\alpha_i \neq 0$  のときの  $x_i$  だけが線形結合に貢献

Kuhn - Tucker の条件より鞍点において

$$\alpha_i [y_i (\mathbf{x}_i \cdot \mathbf{w}_0 + b_0) - 1] = 0$$
$$(i = 1, \dots, l)$$

$\alpha_i \neq 0$  のとき

$$y_i (\mathbf{x}_i \cdot \mathbf{w}_0 + b_0) = 1$$

この条件を満たす  $x_i$  を support vector

最適超平面は support vector で表現可能

# 最適超平面: マージンの計算



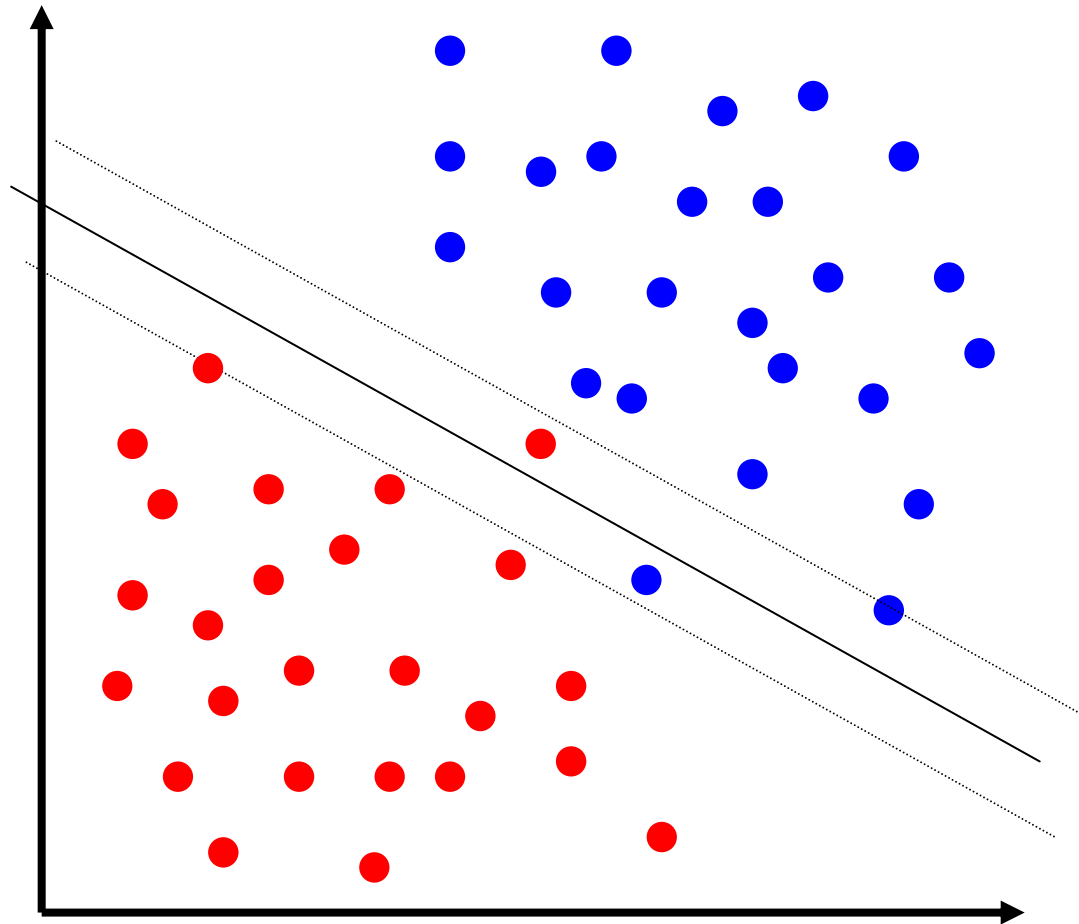
$\mathbf{w}_0$  の定める最適超平面のマージン  $\rho_0$  は  $2/\sqrt{\mathbf{w}_0 \cdot \mathbf{w}_0}$  なので  $\mathbf{w}_0 \cdot \mathbf{w}_0$  を計算.

$\Lambda_0 = (\alpha_1^0, \dots, \alpha_l^0)$  が  $L(\mathbf{w}_0, b_0, \Lambda) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \mathbf{w}_0 \cdot \mathbf{w}_0$  を最大化すると仮定.

$$\begin{aligned} \mathbf{w}_0 \cdot \mathbf{w}_0 &= \sum_{i=1}^l \alpha_i^0 y_i \mathbf{x}_i \cdot \mathbf{w}_0 & \mathbf{w}_0 &= \sum_{i=1}^l \alpha_i y_i \mathbf{x}_i \\ &= \sum_{i=1}^l \alpha_i^0 (1 - y_i b_0) & y_i (\mathbf{x}_i \cdot \mathbf{w}_0 + b_0) &= 1 \\ &= \sum_{i=1}^l \alpha_i^0 & \sum_{i=1}^l y_i \alpha_i &= 0 \end{aligned}$$



# エラーを許す分類へ：ソフトマージン



線形分離が困難な場合、  
各データごとに非負実数

$$\xi_i \geq 0 \quad \dots (1)$$

を用意し制約を緩和

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i \\ (i=1, \dots, l) \quad \dots (2)$$

マージンを広くし、誤差の和  $\sum_{i=1}^l \xi_i$  を  
小さくする超平面で分離したい

$$\frac{1}{2} \mathbf{w}^2 + C \sum_{i=1}^l \xi_i \quad \text{の最小化 (Cは定数)}$$

# ソフトマージン

$$\frac{1}{2} \mathbf{w}^2 + C \sum_{i=1}^l \xi_i \quad \dots \quad (1)$$

を以下の制約のもと最小化

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) \geq 1 - \xi_i, \quad i=1, \dots, l$$

$$\xi_i \geq 0, \quad i=1, \dots, l$$

$\Lambda (\geq 0), \mathbf{R} (\geq 0)$  をラグランジュ乗数

$$L(\mathbf{w}, \xi, b, \Lambda, \mathbf{R}) =$$

$$\frac{1}{2} \mathbf{w}^2 + C \sum_{i=1}^l \xi_i - \sum_{i=1}^l \alpha_i [y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i] - \sum_{i=1}^l r_i \xi_i$$

(1) を最小化する  $\mathbf{w}_0, b_0, \xi_i^0$  は以下の条件をみたす

$$\left. \frac{\partial L}{\partial \mathbf{w}} \right|_{\mathbf{w}=\mathbf{w}_0} = \mathbf{w}_0 - \sum_{i=1}^l \alpha_i y_i \mathbf{x}_i = 0$$

$$\left. \frac{\partial L}{\partial b} \right|_{b=b_0} = \sum_{i=1}^l \alpha_i y_i = 0 \quad \left. \frac{\partial L}{\partial \xi_i} \right|_{\xi_i=\xi_i^0} = C - \alpha_i - r_i = 0$$



$\mathbf{w}_0, b_0, \xi_i^0$  を  $L(\mathbf{w}, \xi, b, \Lambda, \mathbf{R})$  に代入し変形

$$L(\mathbf{w}_0, \xi^0, b_0, \Lambda, \mathbf{R})$$

$$= \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$$

$L(\mathbf{w}_0, \xi^0, b_0, \Lambda, \mathbf{R})$  を最大化する  $\Lambda$  を

2次計画法で計算

Kuhn - Tucker の定理より

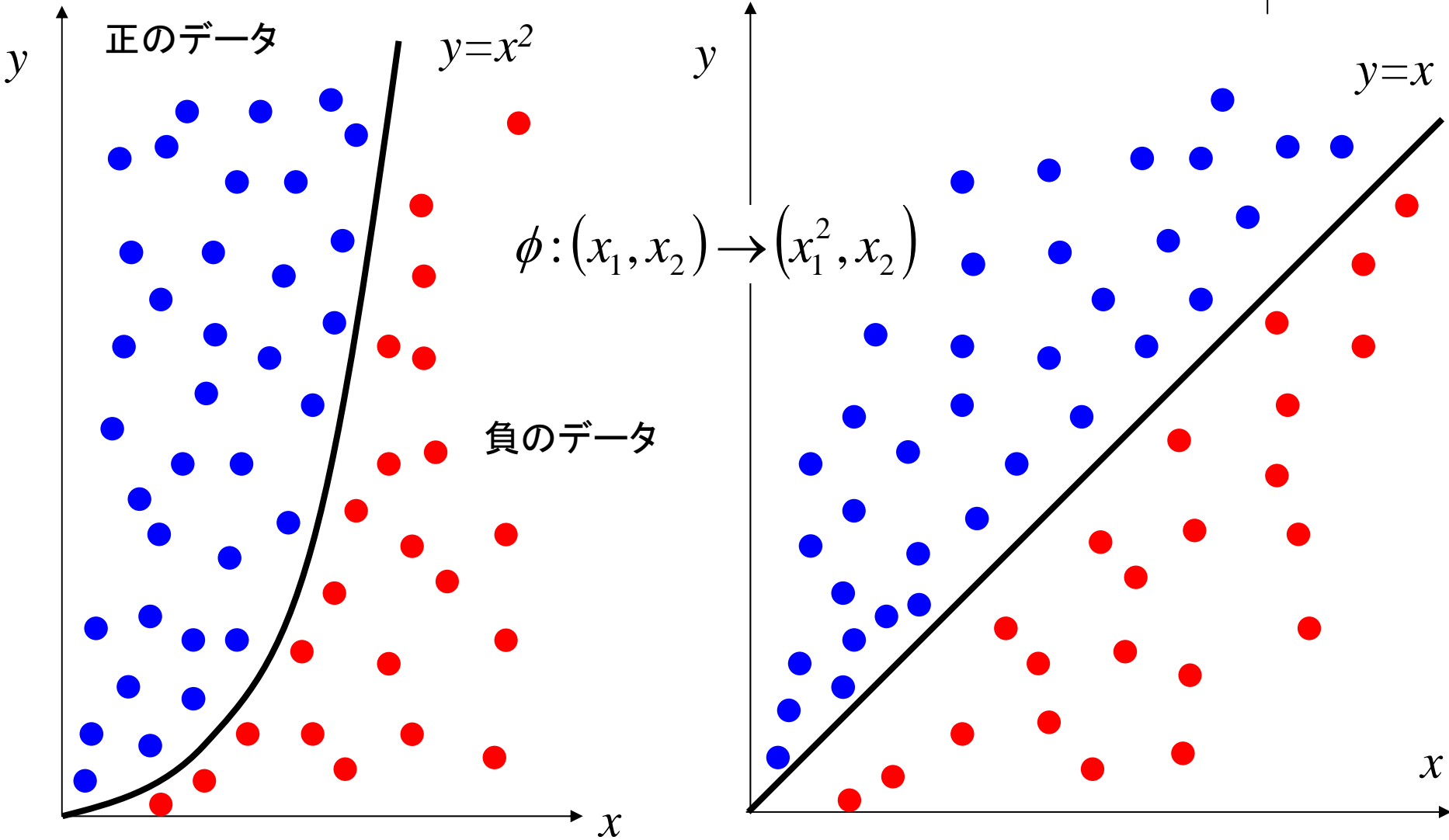
$$\alpha_i [y_i(\mathbf{x}_i \cdot \mathbf{w}_0 + b_0) - 1 + \xi_i^0] = 0$$

$\alpha_i \neq 0$  となる  $\mathbf{x}_i$  を support vector と定義.

$\mathbf{w}_0$  はやはり support vector  $\mathbf{x}_i$  の線形結合

$$\mathbf{w}_0 = \sum_{i=0}^l \alpha_i y_i \mathbf{x}_i$$

# 非線形の判別関数による分類



# 非線形の判別関数： 特徴空間



入力空間  $R^n$  の点を効果的に線形分離できない場合、  
より高い次元の特徴空間  $R^N$  に写像し線形分離を試みる

$$\mathbf{x} \in R^n \rightarrow \phi(\mathbf{x}) = (\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \dots, \phi_N(\mathbf{x})) \in R^N$$

例  $\phi((x_1, x_2)) = (x_1^2, x_1x_2, x_2^2, x_1, x_2)$

特徴空間で最適超平面を形成する  $\mathbf{w}_0$  は

$\phi(\mathbf{x}_i)$  の形をした support vector の線形結合  $\sum_{i=1}^l \alpha_i y_i \phi(\mathbf{x}_i)$



# 非線形の判別関数： 計算量の軽減

入力空間に比べ巨大な 特徴空間 ( $n \ll N$ ) を使えば線形分離は容易 に  
例 訓練データ数より 大きい次元の特徴空間 .  $N \approx 2^n$  の場合も .

入力パラメータ数が  $n = 100$  にたいして  $N = 2^{100} \cong 10^{30}$  次元の特徴空間

巨大な特徴空間では計 算量も増大

双対問題を解くとき

$$L(\mathbf{w}_0, b_0, \Lambda) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$$

テスト  $\mathbf{x}$  を最適超平面  $\mathbf{w}_0 = \sum_{i=1}^l \alpha_i y_i \phi(\mathbf{x}_i)$  で分類するとき

$$\phi(\mathbf{x}) \cdot \mathbf{w}_0 + b_0 = \sum_{i=1}^l \alpha_i y_i \phi(\mathbf{x}) \cdot \phi(\mathbf{x}_i) + b_0 \geq 1? \quad (\leq -1?)$$

$\phi(\mathbf{x})$  と  $\phi(\mathbf{y})$  を陽に計算せずに、 $\phi(\mathbf{x}) \cdot \phi(\mathbf{y})$  を計算可能な関数  $\phi$  の条件とは？



# 非線形の判別関数：カーネルトリック

多項式カーネル

$\mathbf{x}, \mathbf{y} \in R^2$  のとき  $\mathbf{x} = (x_1, x_2), \mathbf{y} = (y_1, y_2)$  と表現

$\phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1x_2, x_2^2, \sqrt{2}x_1, \sqrt{2}x_2, 1)$   $\phi(\mathbf{y}) = (y_1^2, \sqrt{2}y_1y_2, y_2^2, \sqrt{2}y_1, \sqrt{2}y_2, 1)$  ならば

$$\phi(\mathbf{x}) \cdot \phi(\mathbf{y}) = (\mathbf{x} \cdot \mathbf{y} + 1)^2 = (x_1y_1 + x_2y_2 + 1)^2 = (x_1y_1)^2 + 2x_1y_1x_2y_2 + (x_2y_2)^2 + 2x_1y_1 + 2x_2y_2 + 1$$

一般に  $\mathbf{x}, \mathbf{y} \in R^n$  のとき  $\phi(\mathbf{x}) \cdot \phi(\mathbf{y}) = (\mathbf{x} \cdot \mathbf{y} + 1)^d$  の形にするには..

$$(\mathbf{x} \cdot \mathbf{y} + 1)^d = \sum_{c_1, c_2, \dots, c_{n+1}} \frac{d!}{c_1!c_2! \dots c_n!c_{n+1}!} (x_1y_1)^{c_1} (x_2y_2)^{c_2} \dots (x_ny_n)^{c_n} 1^{c_{n+1}} \quad \text{ただし} \quad \sum_{i=1}^{n+1} c_i = d, c_i \geq 0.$$

$\phi(\mathbf{x})$  は  $\sqrt{\frac{d!}{c_1!c_2! \dots c_n!c_{n+1}!}} (x_1)^{c_1} (x_2)^{c_2} \dots (x_n)^{c_n} 1^{c_{n+1}}$  を各要素とするベクトルであればよい。

$\phi(\mathbf{x})$  の次元は  ${}_{n+d}C_n = {}_{n+d}C_d$

他の例：Gauss カーネル  $\phi(\mathbf{x}) \cdot \phi(\mathbf{y}) = \exp\left(-\frac{|\mathbf{x} - \mathbf{y}|^2}{\sigma^2}\right)$

