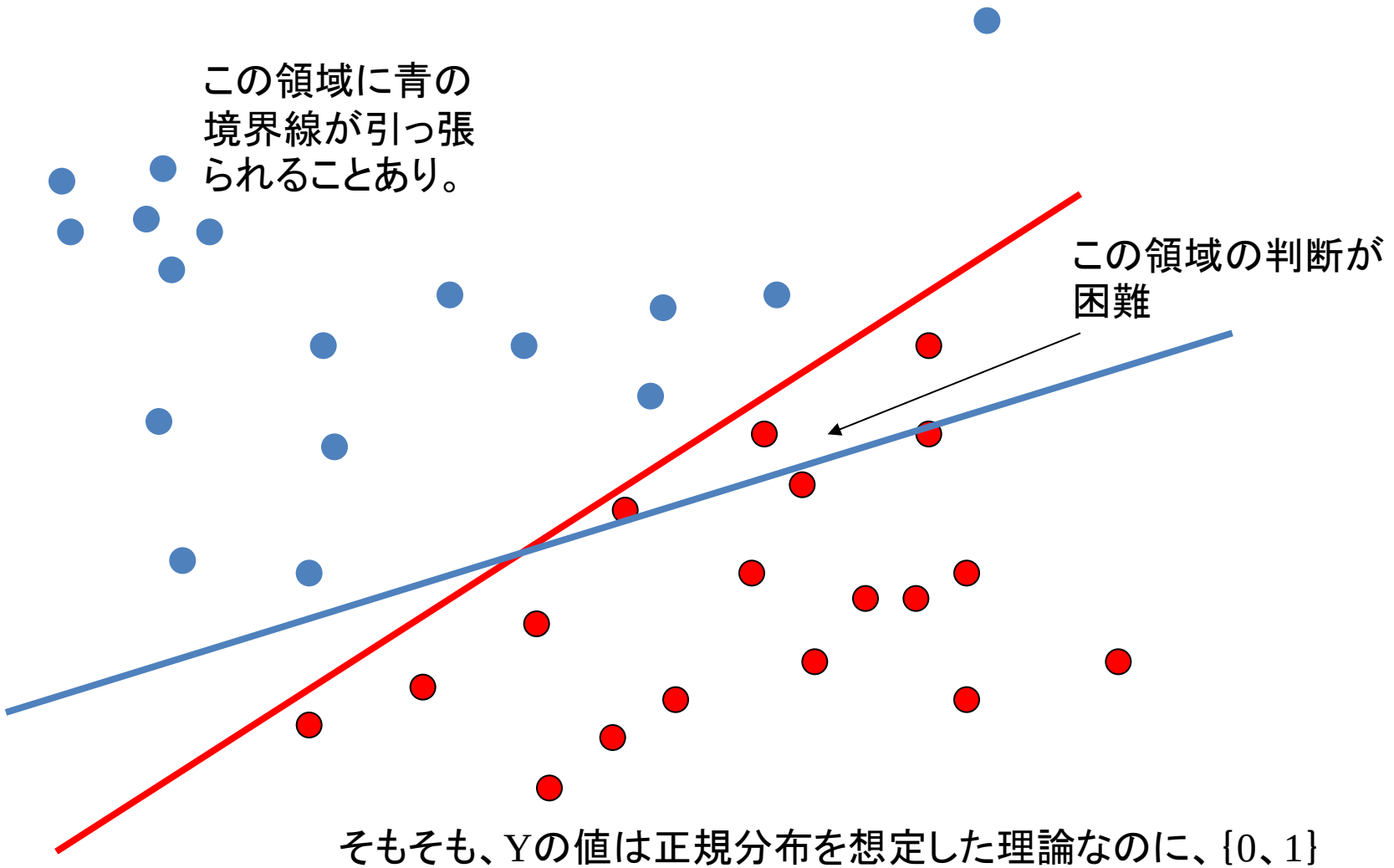
# 5. サポートベクターマシン

SVMの概念
双対化によるSVMの定式化：線形分離可能な場合
KKT条件とサポートベクトル
双対化の御利益
ソフトマージンSVM：線形分離できない場合
SVM実装のためのアルゴリズム（ワーキング集合、SMO)
SVMによる回帰
カーネル関数

by 中川裕志（東京大学）

# 再掲:2乗誤差最小化の線形識別の問題点




そもそも、Yの値は正規分布を想定した理論なのに、{0、1}の2値しかとらないとして2乗誤差最小化を当てはめたところに無理がある。

# SVMの定式化

- 学習データ:
  - N個の入力ベクトル$\mathbf{x}_1,\ldots,\mathbf{x}_N$と
  - 各々に対応するクラス分け結果 $y_1,\ldots,y_N$

    ただし、$y_i$ は＋1（正例）か−1（負例）をとる。
  - 新規のデータ$\mathbf{x}$に対して、yが $y(\mathbf{x})>0$なら正、$y(\mathbf{x})<0$なら負になるようにしたい

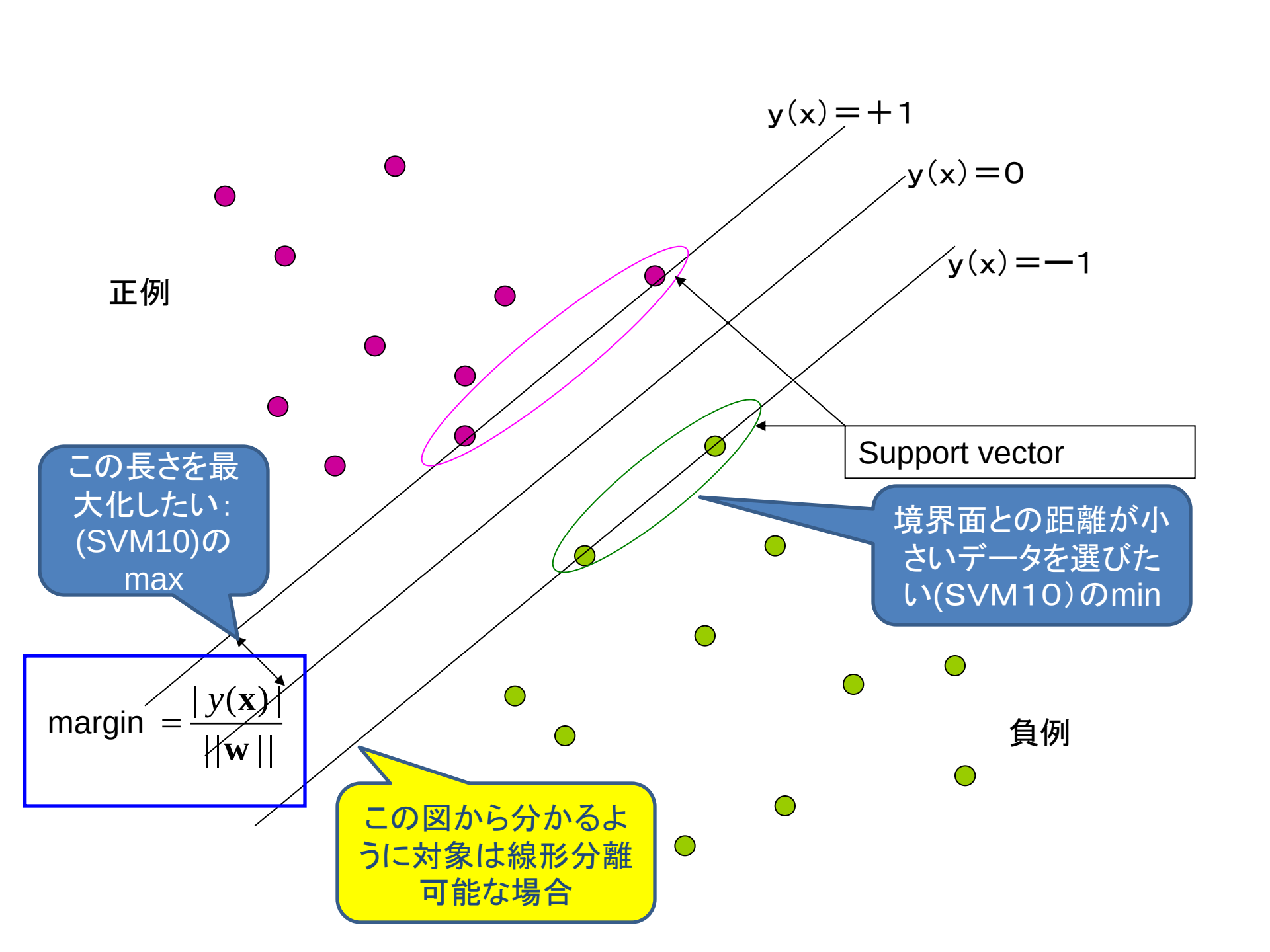
$$y(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + w_0$$

正しい分類ができた場合は、

$y(\mathbf{x}) > 0$かつ$y = +1$　あるいは　$y(\mathbf{x}) < 0$かつ$y = -1$

すなわち、 $y(\mathbf{x})y > 0$

正しく分類できなかった場合は、当然ながら　$y(\mathbf{x})y < 0$


y(x)＝＋1
y(x)＝0
y(x)＝－1
正例
負例
Support vector
この長さを最大化したい：(SVM10)のmax
境界面との距離が小さいデータを選びたい(SVM10)のmin
margin ＝ $\frac{|y(\mathbf{x})|}{||\mathbf{w}||}$
この図から分かるように対象は線形分離可能な場合

# マージン最大化

- SVMの狙いは識別境界面:y(x)=0から一番近いデータ点までの距離（マージン:margin）を最大化すること。以下でその考えを定式化する。
- 識別境界面:y(x)=0からあるxまでの距離は、線形識別の幾何学的解釈で述べたようにy(**x**)/||**w**||
- 我々は正しく識別されたデータ、すなわち $y_n y(\mathbf{x}_n)>0$ のデータにだけ焦点を当てる。
- すると、点$\mathbf{x}_n$から境界面までの距離は次式。

$$\frac{y_n y(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0)}{\|\mathbf{w}\|}$$

- **w**を定数倍してc**w**と変換しても、境界面までの距離

  $y_n y(\mathbf{x_n})/\|\mathbf{w}\|$の値は分母子でcが相殺するので不変。

- よって、境界面に一番近い点で次式が成立しているとする。

$$y_n\left(\left\langle \mathbf{w}, \mathbf{x}_n \right\rangle + w_0\right) = 1$$

- したがって、最適な$\mathbf{w}, w_0$を求めることは、境界面までの距離が最小の点の距離（margin)を最大化するという問題に定式化でき、以下の式となる。

$$\arg\max_{\mathbf{w}, w_0}\left\{\frac{1}{\|\mathbf{w}\|}\min_n\left[y_n\left(\left\langle \mathbf{w}, \mathbf{x}_n \right\rangle + w_0\right)\right]\right\} \qquad \cdots(SVM10)$$

- この最適化はそのままでは複雑なので、等価な問題に変換する。

# 双対問題化

- 元の問題では、argmax{ min }という一般的な枠組みの問題なので、内点法などの効率の良い最適化アルゴリズムが良く研究されている「線形制約凸2次計画問題」に変換する方向に進める。
参考文献：工系数学講座17「最適化法」（共立出版）

- 境界面に一番近いデータでは $y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0) = 1$

- したがって、正しく識別された全てのデータは次式の制約を満たす。

$$y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0) \geq 1 \qquad n = 1, \cdots, N$$

- ここで、等号が成り立つデータをactive、そうでないデータをinactiveと呼ぶ。

制約条件 : $y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0) \geq 1 \qquad n = 1, \cdots, N \qquad \cdots (SVM\,20)$

- ここで、等号が成り立つデータをactive、そうでないデータをinactiveと呼ぶ。
- 定義より、activeなデータは境界面に一番近いデータであり、これがsupport vectorとなる。
- このとき、marginを最大化する式(SVM10)から、$\|\mathbf{w}\|^{-1}$を最大化するのだが、これは等価的に$\|\mathbf{w}\|^2$を最小化する問題となる。すなわち、以下の定式化。

$$\arg\min_{\mathbf{w},b} \frac{1}{2} \| \mathbf{w} \|^2 \equiv \frac{1}{2} \mathbf{w}^T \mathbf{w}$$

$$\text{subject to} \qquad y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0) \geq 1 \qquad n = 1, \cdots, N \qquad \cdots (SVM\,30)$$

➢（SVM30)のような不等式制約を持つ最適化問題は、Lagrange未定乗数ベクトル**a**を導入したLagrange関数: $L(\mathbf{w}, w_0, \mathbf{a})$を

**w**,$w_0$（最小化)→**a**(最大化) という最適化問題に双対化する

まず、**w**, $w_0$について、$L(\mathbf{w}, w_0, \mathbf{a})$の最適化を行う。

$$L(\mathbf{w}, w_0), \mathbf{a}) = \frac{1}{2} \| \mathbf{w} \|^2 + \sum_{n=1}^{N} a_n \{1 - y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0)\} \quad \cdots (SVM\,40)$$

$\mathbf{w}, w_0$に関して微分すると以下の条件が出る。

$$\mathbf{w} = \sum_{n=1}^{N} a_n y_n \mathbf{x}_n \quad \cdots (SVM\,50)$$

$$0 = \sum_{n=1}^{N} a_n y_n \quad \cdots (SVM\,60)$$

➢(SVM50)(SVM60)を（SVM40)に代入して、**w**と$w_0$を消すと、次のように双対問題としての定式化が得られる

# SVMの双対問題ー境界面で完全に分離できる場合

$$\max \tilde{L}(\mathbf{a}) = \max\left[\sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m y_n y_m k(\mathbf{x}_n, \mathbf{x}_m)\right] \quad \cdots(SVM70)$$

$$\text{subject to} \quad a_n \geq 0 \quad n = 1,.., N \quad \cdots(SVM80)$$

$$\sum_{n=1}^{N} a_n y_n = 0 \quad \cdots(SVM90)$$

$$where \quad k(\mathbf{x}_n, \mathbf{x}_m) = \langle \mathbf{x}_n, \mathbf{x}_m \rangle$$

これがカーネル関数（データ$\mathbf{x}_n$,$\mathbf{x}_m$の対だけによる）後で説明する

また、新規のデータに対する予測は次式の$y(\mathbf{x})$で行う

$$y(\mathbf{x}) = \sum_{n=1}^{N} a_n y_n k(\mathbf{x}, \mathbf{x}_n) + w_0 \quad \cdots(\text{SVM100})$$

上記(SVM70,80,90)を解くアルゴリズムは後に説明する。また、（SVM100）で良い理由はカーネルの関する記述で述べる（表現定理）

# 双対化を最適化の観点から見なおそう

- 最適化問題　P

$$\min f(x)$$
$$\text{subject to } g_i(x) \le 0 \qquad i = 1,..,m$$

- ラグランジュ関数

$$L(x,\lambda) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x)$$
$$= f(x) + \lambda^T g(x) \quad (\lambda, g \text{はベクトルで書く})$$

- 双対問題　Q

$$q(\lambda) = \min_x L(x,\lambda)$$
$$\max q(\lambda)$$
$$\text{subject to } \lambda \ge 0$$

# 双対定理

- 弱双対定理
  - 最適化問題Pにおける$f$の最適値$=f^*$

    双対問題Qにおける$q$の最適値$=q^*$

    $$q^* \leq f^*$$
- 強双対定理
  - 目的関数$f$が凸で、制約条件が線形の場合は　$q^* = f^*$であり、対応するラグランジュ乗数$\lambda^*$が存在する。
  - Pは制約条件が線形なので、 $f$が凸なら強双対定理が成立

# 双対化を最適化の観点から見なおそう

- 元の問題（再掲）

$$\arg\min_{\mathbf{w},w_0} \frac{1}{2}\|\mathbf{w}\|^2$$

$$\text{subject to} \quad y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0) \geq 1 \qquad n = 1, \cdots, N \qquad \cdots (SVM\,30)$$

- この問題では目的関数は2乗ノルムなので凸であり、制約条件が線形な式なので、強双対定理が成立し、双対問題を解けば、この問題（主問題）の解が得られる。

# 鞍点定理からみると

- 元の問題（再掲）

$$\arg\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2$$

$$\text{subject to} \quad y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0) \geq 1 \qquad n = 1, \cdots, N \qquad \cdots(SVM\,30)$$

$$L(\mathbf{w}, w_0, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} a_n \left\{1 - y_n\left(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0\right)\right\} \qquad \cdots(SVM\,40)$$

$\mathbf{w}, w_0$に関して微分すると以下の条件が出る。

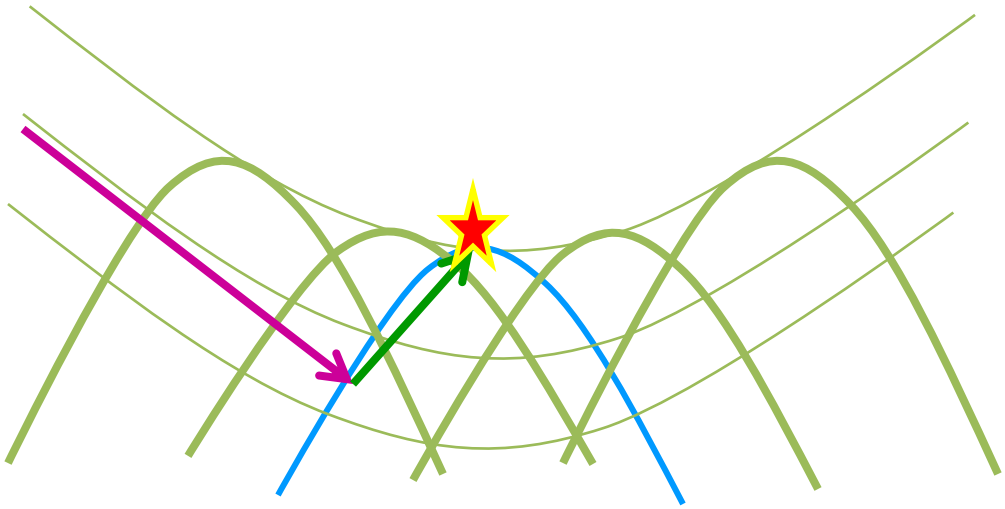
$$\mathbf{w} = \sum_{n=1}^{N} a_n y_n \mathbf{x}_n \qquad \cdots(SVM\,50)$$

$$0 = \sum_{n=1}^{N} a_n y_n \qquad \cdots(SVM\,60)$$

- 上記のLagrange関数$L(\mathbf{w},w_0,a)$の最小化の意味は次のページの図

- Lagrange関数$L(\mathbf{w}, w_0, a)$の最小化の意味は下の図で鞍点にかかる曲線に上から近づく処理であり、

  となる$\mathbf{w}, w_0$を代入して ↘ のように動く。 $\frac{\partial L}{\partial \mathbf{w}} = 0, \frac{\partial L}{\partial w_0} = 0$



- この曲線に沿って最適点 ★ に $a$ を動かす処理が双対問題であり、図から分かるように最大化となる
- つまり $\max_{a} \min_{\mathbf{w}, w_0} L(\mathbf{w}, w_0, a)$ という問題

## 鞍点定理

$$\exists x^*, \lambda^* \quad \min_{x} \max_{\lambda \geq 0} L(x, \lambda) = L(x^*, \lambda^*) = \max_{\lambda \geq 0} \min_{x} L(x, \lambda)$$

$\Rightarrow x^*$は主問題, $\lambda^*$は双対問題の解

## 前のページとの対応は$\boldsymbol{w}w_0 = x,\ a = \lambda$

# スパースなデータに対する識別器

- $k(\mathbf{x}_n, \mathbf{x}_m) = \langle \mathbf{x}_n, \mathbf{x}_m \rangle$ ;カーネル関数を利用して、回帰や識別を行うことは $k(x,y)$において、$\{x,y\}$ のペアの観測データ(＝教師データ)が膨大だと、正規方程式に現れた $\mathrm{X}^{\mathrm{T}}\mathrm{X}$
  - ($\mathrm{X}^{\mathrm{T}}\mathrm{X}$がちょうど$k(x_n,x_m)$を$(n,m)$成分とする行列)
- の逆行列の計算が計算量的に困難！

- すべての観測データを使うと、重要な境界線が観測データの集中的に集まっている部分に強い影響を受けてしまう。

- 限定された観測データを使って効率よく計算できないものだろうか。

- 正例データと負例データのうち、両者の境界にあるもの(これをSupport Vectorという)だけを使えば、つまりスパースなデータによるカーネルならずいぶん楽になる。
- → Support Vector Machine

# *KKT*条件

minimize $f(\mathbf{x})$

subject to $g_i(\mathbf{x}) \le 0 \quad (i = 1,..,m)$ は

*Lagrangian* $L(\mathbf{x}, \lambda) \equiv f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x})$

を最適化する以下の条件で得られる

$$\nabla f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}) = 0 \qquad (KKT-1)$$

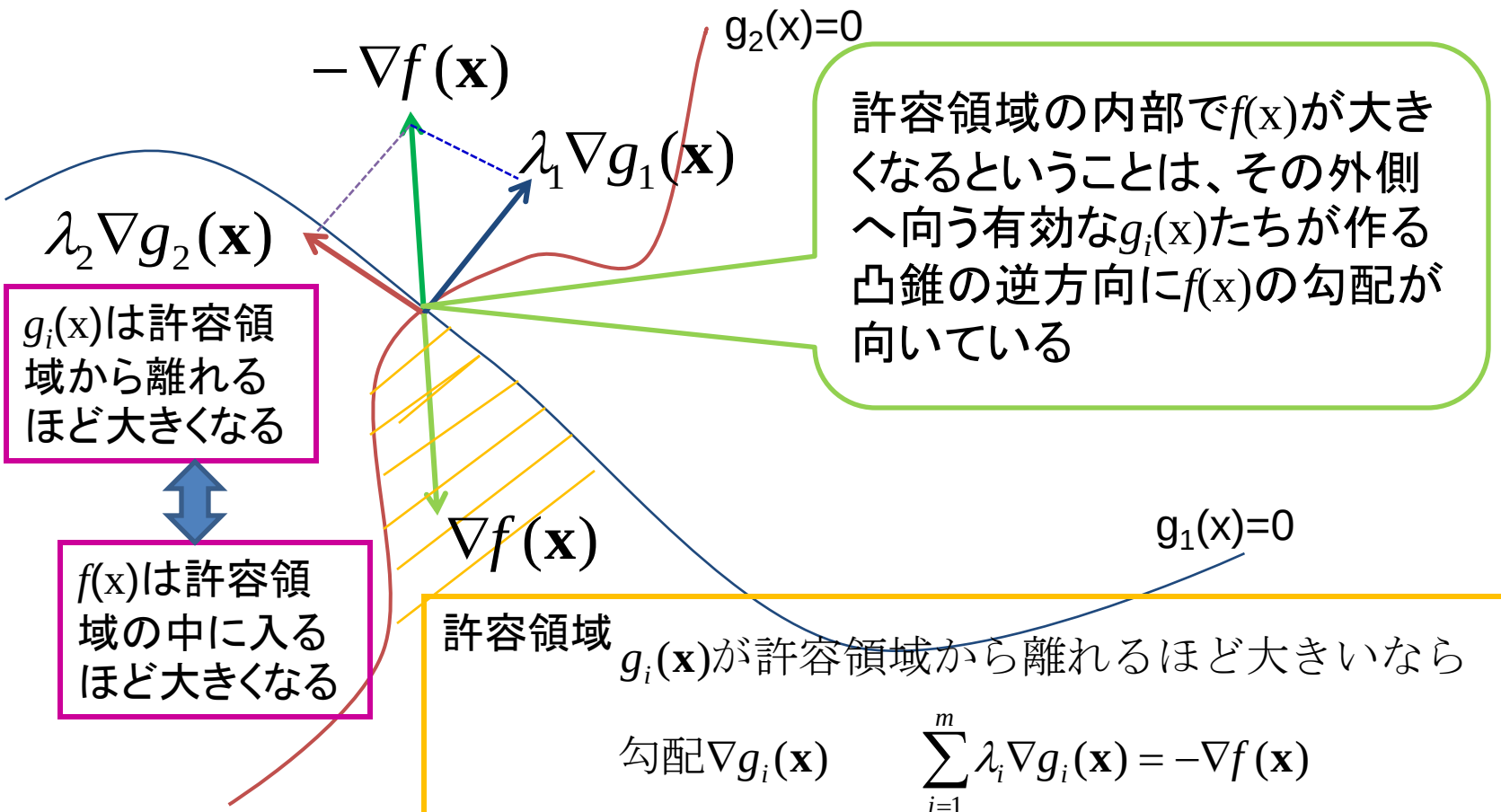
$$g_i(\mathbf{x}) \le 0 \qquad (KKT-2)$$

$$\lambda_i \ge 0 \qquad (KKT-3)$$

$$\lambda_i g_i(\mathbf{x}) = 0 \qquad (KKT-4)$$

これを*KKT*条件と呼ぶ。なお$\lambda_i > 0$なら$g_i(\mathbf{x}) = 0$なので、このような$g_i$を有効な制約と呼ぶ。

$$\nabla f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}) = 0$$ の解釈




許容領域の内部で$f(x)$が大きくなるということは、その外側へ向う有効な$g_i(x)$たちが作る凸錐の逆方向に$f(x)$の勾配が向いている

$g_i(\mathbf{x})$が許容領域から離れるほど大きいなら

勾配$\nabla g_i(\mathbf{x})$ $\sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}) = -\nabla f(\mathbf{x})$

なので、許容領域に入り込むほど大きいので

$f(\mathbf{x})$は許容領域の端で最小

なお、この問題における*KKT*条件は以下のようになる。

$$1 - y_n y(\mathbf{x}_n) \le 0 \quad \leftarrow (\mathrm{KKT}-2)$$

$$a_n \ge 0 \quad \leftarrow (\mathrm{KKT}-3)$$

$$a_n(1 - y_n y(\mathbf{x}_n)) = 0 \leftarrow (\mathrm{KKT}-4)$$

$$L(\mathbf{x},\lambda) \equiv f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x}) \qquad g_i(\mathbf{x}) \le 0$$

だったが、ここでの定式化では、

$$L(\mathbf{w}, w_0, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} a_n \left\{1 - t_n\left(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0\right)\right\}$$

よって、全てのデータは、$a_n = 0$　か$y_n y(\mathbf{x}_n) = 1$となる。

$a_n = 0$の点は$y(\mathbf{x})\cdots(SVM100)$に寄与しない。

寄与するのは$y_n y(\mathbf{x}_n) = 1$の点だけ。

この点たちが support vector である。

# $w_0$の求め方

support vector Sに含まれるデータnにおいては　$y_m y(\mathbf{x}_m) = 1$

よって、$y_m \left( \sum_{n=1}^{N} a_n y_n \langle \mathbf{x}_m, \mathbf{x}_n \rangle + w_0 \right) = 1$

両辺に$y_m$を掛け、${y_m}^2 = 1$に注意すると、

$w_0$は以下の式で与えられる。

なお、１個の$m$だけではなく、$\sum_{m \in S}$しているのは解の安定性のため

$$w_0 = \frac{1}{|S|} \sum_{m \in S} \left( y_m - \sum_{n \in S} a_n y_n \langle \mathbf{x}_m, \mathbf{x}_n \rangle \right)$$

# 双対化の御利益:
# 教師データアクセスの観点から

- 主問題と双対問題は最適化するパラメター数が違う。
  - 主問題パラメター数 >>双対問題パラメター数 なら双対問題を解くほうが楽　→教科書的
- SVMの場合:
  - 主問題のパラメターは重みベクトル:w
  - 双対問題にパラメターは個別データ:$x_i$
  - →必ずしも教科書的なお得感ではない。

# 双対化の御利益

- SVMの場合：
  - 主問題のパラメターは重みベクトル:w
  - 下の定式化なので、全教師データ$\{y_n, \mathbf{x}_n\}$が同時に必要

$$\arg\min_{\mathbf{w}, w_0} \frac{1}{2} \| \mathbf{w} \|^2$$

高次元ベクトル

$$\text{subject to} \quad y_n(\langle \mathbf{w}, \mathbf{x}_n \rangle + w_0) \geq 1 \qquad n = 1, \cdots, N \qquad \cdots(SVM\,30)$$

- データ量が大きくメモリにロード仕切れない場合に困ったことになる。
  - データ量は最近、増加傾向

# 双対化の御利益

- →必ずしも教科書的なお得感ではない。
- 一方、双対問題では入力データ$\mathbf{x}_i y_i$のと最適化する$a_i$が対応する形で最適化式に現れるので、どのデータを学習で使うか制御しやすい。（下の式参照）
  - 例えば、$a_i (i \neq j)$を固定して、$a_j$を最適化する操作を$j$を動かして繰り返すなど。そのときには $k(\mathbf{x}_i, \mathbf{x}_j)\ j = 1,\ldots,N$ だけしか使わない。

スカラー

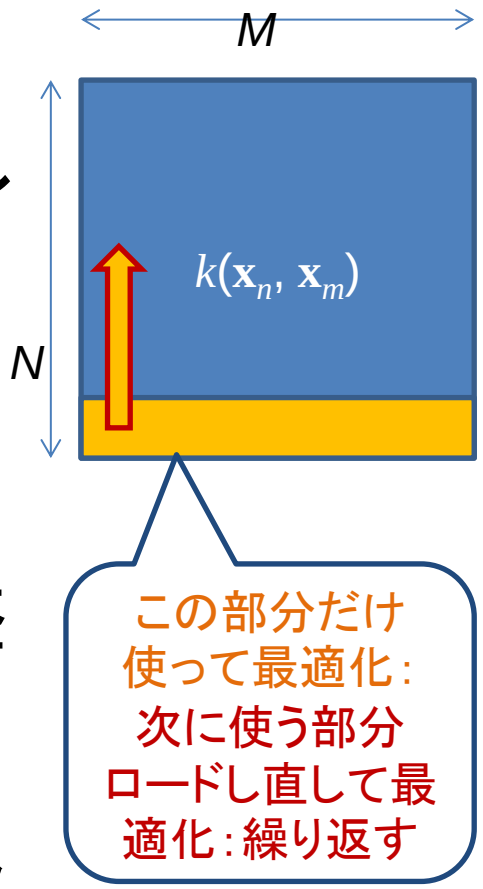
$$\max \tilde{L}(\mathbf{a}) = \max\left[\sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m y_n y_m \langle \mathbf{x}_n, \mathbf{x}_m \rangle\right] \quad \cdots (SVM\,70)$$

$$\text{subject to} \quad a_n \geq 0 \quad n = 1,..,N \quad \cdots (SVM\,80)$$

$$\sum_{n=1}^{N} a_n y_n = 0 \quad \cdots (SVM\,90) \quad where \quad k(\mathbf{x}_n, \mathbf{x}_m) = \langle \mathbf{x}_n, \mathbf{x}_m \rangle$$ とも書く

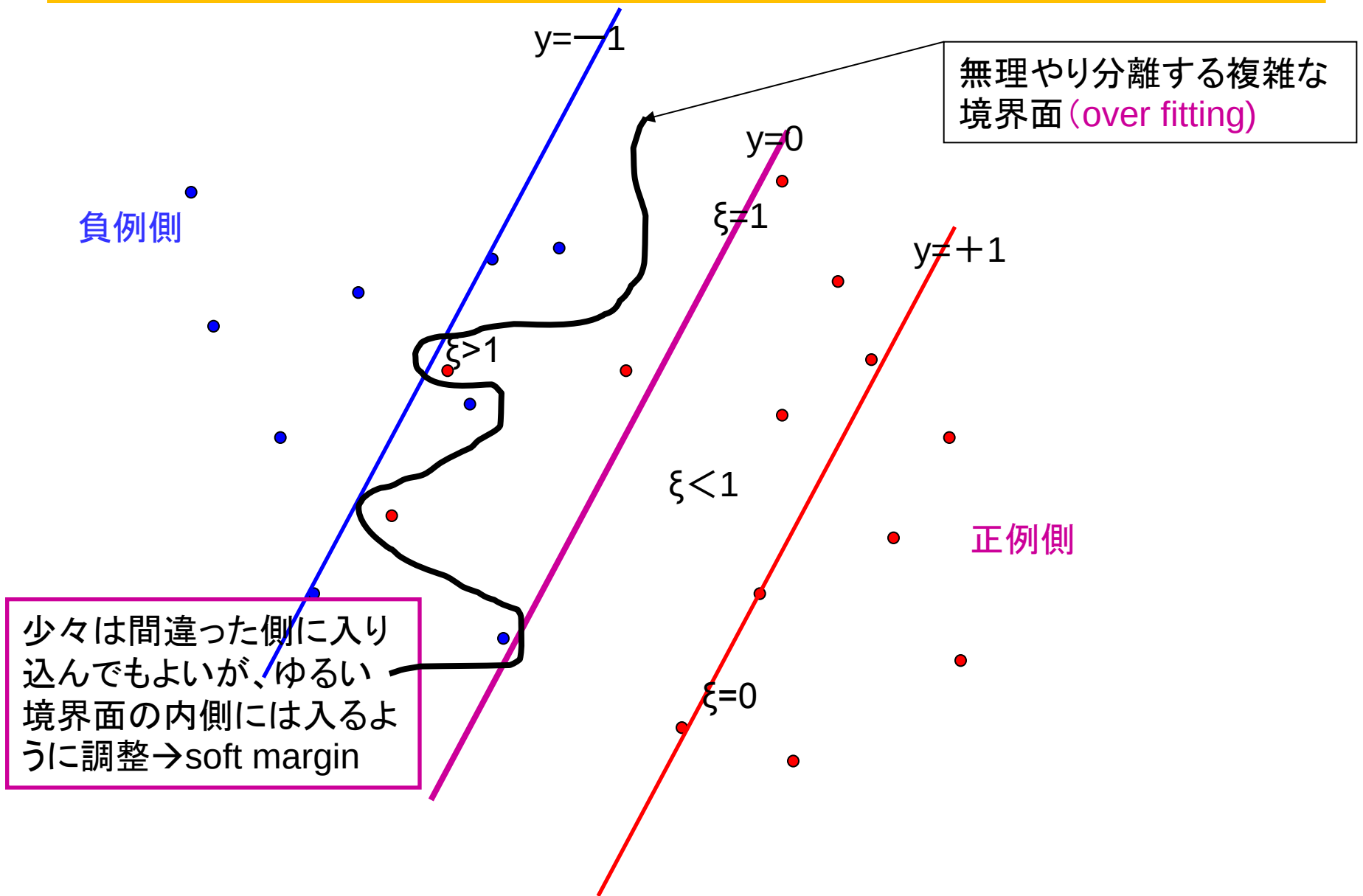
# 双対化の御利益

- 入力データ、あるいはカーネル行列全体がメモリに乗り切らないビッグデータを扱うために、入力（すなわちカーネル$k(\mathbf{x}_n, \mathbf{x}_m)$の一部を取捨選択してメモリにロードして使う方法が、この双対化で可能になっている。




- →ビッグデータ時代における御利益
  - cf. 台湾大学のLIBSVM （SVMの有名な実装）
  - 全データからどのようにメモリにロードする部分を切り出すかが重要な研究課題

# SVMの定式化　境界面で完全に分離できない場合



- スラック変数ξ≥0を導入
  - 正しいsoft marginの境界面の上ないし内側の点ではξ＝0
  - その他の点$x_n$では $\xi_n = |y_n - y(x_n)|$
  - 中央の識別境界面　$y(x_n)=0$　では、$\xi_n = 1$
  - 間違って識別された点は$\xi_n > 1$
- まとめると線形分離できない場合の制約条件のξによる緩和:

$$1 \le y_n y(\mathbf{x}_n) \Rightarrow 1 - y_n y(\mathbf{x}_n) \le 0 \Rightarrow 1 - y_n y(\mathbf{x}_n) \le \xi_n$$
$$\Rightarrow 1 - y_n y(\mathbf{x}_n) - \xi_n \le 0 \quad where \quad \xi_n \ge 0 \qquad ...(SF10)$$

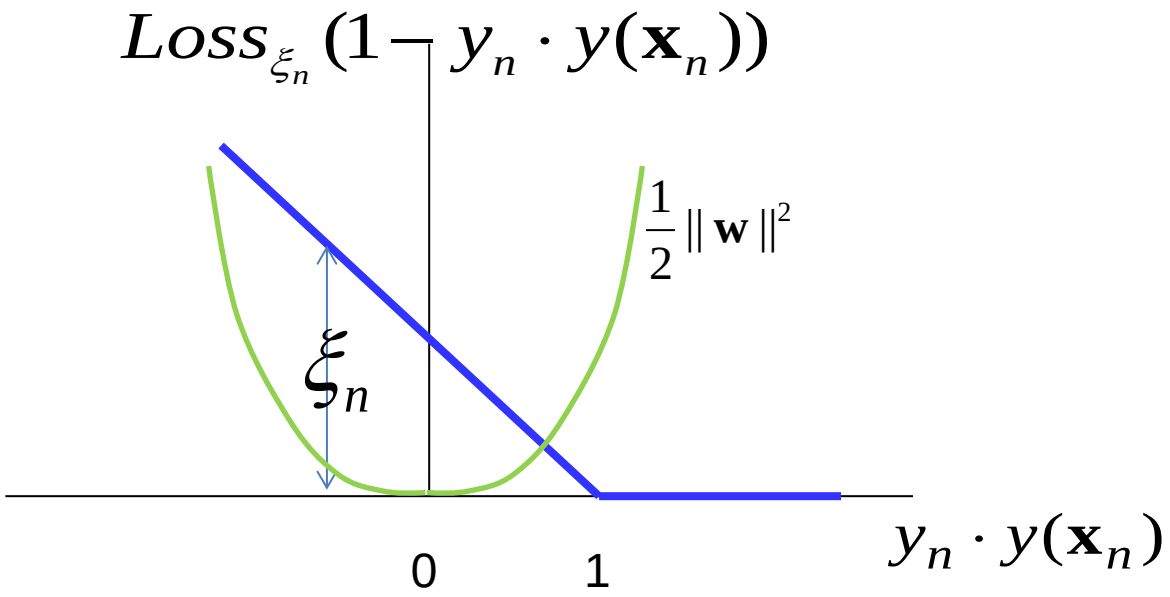
  - $\xi_n > 1$ が許容されるが、できるだけ小さく押さえたい！
- 最適化は以下のように形式化される。ただし、$C$はスラック変数ξのペナルティとmargin **w**の按配を制御するパラメター

$$minmize\left[ C\sum_{n=1}^{N} \xi_n + \frac{1}{2}||\mathbf{w}||^2 \right] \qquad where \quad C > 0 \qquad \ldots(SF20)$$

$$Loss_{\xi}(1-y\cdot y(\mathbf{x}))=\begin{cases}0 & \text{if } 1-y\cdot y(\mathbf{x})<0\\ \xi & \text{if } 1-y\cdot y(\mathbf{x})\geq\xi>0\end{cases}$$

この関数を用いると次の
関数の最小化問題となる。

$$\min\left\{C\sum_{n=1}^{N}Loss_{\xi_n}(1-y_n\cdot y(\mathbf{x}_n))+\frac{1}{2}\|\mathbf{w}\|^2\right\}=\min\left\{C\sum_{n=1}^{N}\xi_n+\frac{1}{2}\|\mathbf{w}\|^2\right\}$$



- この最適化問題を解くためのLagrangianは以下のようになる。
  - 最後の項は$\xi \geqq 0$ を表す項。

$$L(\mathbf{w}, w_0, \mathbf{a}) = \frac{1}{2} \| \mathbf{w} \|^2 + C \sum_{n=1}^{N} \xi_n + \sum_{n=1}^{N} a_n \{1 - y_n y(\mathbf{x}_n) - \xi_n\} - \sum_{n=1}^{N} \mu_n \xi_n$$

$$where \quad a_n \geq 0 \qquad \mu_n \geq 0 \qquad y(\mathbf{x}_n) = \langle \mathbf{w}, \mathbf{x}_n \rangle + w_0 \qquad ...(SF30)$$

- KKT条件は以下の通り。

$$a_n \geq 0$$

$$1 - y_n y(\mathbf{x}_n) - \xi_n \leq 0$$

$$a_n \left(1 - y_n y(\mathbf{x}_n) - \xi_n\right) = 0$$

$$\mu_n \geq 0$$

$$\xi_n \geq 0$$

$$\mu_n \xi_n = 0$$

$$where \quad n = 1, ..., N$$

- w、b、ξを最適化するためにLagrangianを各々で微分すると右下の結果が得られる。
- 右をLagrangianに代入すると下の双対問題が得られ線形制約凸2次計画問題となる。

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{n=1}^{N} a_n y_n \mathbf{x}_n$$

$$\frac{\partial L}{\partial w_0} = 0 \quad \Rightarrow \quad \sum_{n=1}^{N} a_n y_n = 0$$

$$\frac{\partial L}{\partial \xi_n} = 0 \quad \Rightarrow \quad a_n = C - \mu_n$$

$$\tilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a_n a_m y_n y_m \langle \mathbf{x}_n, \mathbf{x}_m \rangle$$

$$k(\mathbf{x}_n, \mathbf{x}_m) = \langle \mathbf{x}_n, \mathbf{x}_m \rangle$$

制約条件は　$\mu_n \geq 0$　$a_n \geq 0$　$\Rightarrow$　$a_n \leq C$

以上をまとめると制約条件は全体で以下のようになる

$$0 \leq a_n \leq C$$

$$\sum_{n=1}^{N} a_n y_n = 0$$

# SVM実装上のアルゴリズムの工夫

- さて、いよいよ $a_i$ を求める段階になった。
- SVMは「線形制約を持つ凸2次計画問題」なので、標準的な実装方法が使える。
- ただし、次元が高い場合には、カーネルの行列をメモリに乗せるだけで大変。
- 独自の工夫がなされているので、ポイントを紹介
- 最適解の探索は、素朴なgradient ascent法でも解けるが、効率は良くない。

# ワーキング集合法

- 教師データSを分割して部分的に解くことを繰り返す。

教師データSに対して
$a_i \leftarrow 0$
Sの適当な部分集合S’を選ぶ
repeat
　S’に対する最適化問題を解く
　KKT条件を満たさないデータから新たなS’を選択
until 停止条件を満たす
return $\{a_i\}$

# 分解アルゴリズム

- 変数$\{a_i\}$の集合全体ではなく、ある大きさの部分集合（ワーキング集合）のみを更新する。
- この更新の後、ワーキング集合に新たな点を加え、他の点は捨てる。
- 上記の$\{a_i\}$の選択における極端な方法として、2個の$a_i$だけを使って更新する方法を逐次最小最適化アルゴリズム(Sequential Minimal Optimization algorithm:SMO algorithm)と言う。

- なぜ2点か？
- 復習：右のような最適化だった。

$$\tilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m y_n y_m \langle \mathbf{x}_n, \mathbf{x}_m \rangle$$

$$k(\mathbf{x}_n, \mathbf{x}_m) = \langle \mathbf{x}_n, \mathbf{x}_m \rangle$$

制約条件は　$\mu_n \geq 0$　$a_n \geq 0$　$\Rightarrow$　$a_n \leq C$

以上をまとめると制約条件は全体で次式

$$0 \leq a_n \leq C \qquad (SMO-1)$$

$$\sum_{n=1}^{N} a_n y_n = 0 \qquad (SMO-2)$$

- （SMO－2）より、最適化の各ステップで更新される$a_i$の個数の最小値は2。なぜなら、1個の$a_i$を更新したときは、（SMO－2）を満たすために、最低でもう1個の$a_i$を調整のために更新する必要があるから。

# S’の2点を最適化する更新式

$S'$＝更新の対象となる2点＝$\mathbf{x}_1, \mathbf{x}_2$　　とする。

$$\sum_{j=1}^{N} y_j a_j k(\mathbf{x}_i, \mathbf{x}_j) = \sum_{j=1}^{N} y_j a_j K_{i,j} = f(\mathbf{x}_i)$$

$$a_2^{new} = a_2^{old} + \frac{y_2\left(\left(f(\mathbf{x}_1) - y_1\right) - \left(f(\mathbf{x}_2) - y_2\right)\right)}{K_{11} + K_{22} - 2K_{12}} \qquad (SMO8)$$

$$a_1^{new} = a_1^{old} + y_1 y_2 (a_2^{old} - a_2^{new}) \qquad (SMO9)$$

# 2点の更新式の導出

- 対象とする2点を$a_1$ 、$a_2$とする。
- 動かすのが2点を$a_1$ 、$a_2$だけなので次式が成立

$$a_1^{new} y_1 + a_2^{new} y_2 = a_1^{old} y_1 + a_2^{old} y_2 = \text{定数} \qquad (SMO-3)$$

ただし　$y_i = -1$か$+1 (i = 1,2)$

- まず、$a_2$を$a_2^{old}$から$a_2^{new}$に変えることにする。

- $a_2$の取る範囲の制約　$0 \leq a_2 \leq C$から当然　$0 \leq a_2^{new} \leq C$
- ただし、(SMO-3)から次の制約が加わる。

$y_1 = y_2$の場合

$(SMO-3)$より　$a_2^{new} = a_1^{old} + a_2^{old} - a_1^{new}$

$a_2^{new}$は最大になっても$-a_1^{new}$が最大値0だから、　$a_2^{new} \le a_1^{old} + a_2^{old}$

　は最小になっても$-a_1^{new}$が最小値$-C$だから、　$a_2^{new} \ge a_1^{old} + a_2^{old} - C$

よって、

$U = \max(0, a_1^{old} + a_2^{old} - C)$,　$V = \min(C, a_1^{old} + a_2^{old})$　とおくと

$$U \le a_2^{new} \le V \qquad (SMO-4)$$

$y_1 = -y_2$の場合

$(SMO-3)$より　$a_2^{new} = a_2^{old} - a_1^{old} + a_1^{new}$

$a_2^{new}$は最大になっても$+a_1^{new}$が最大値$C$だから、　$a_2^{new} \le C - a_1^{old} + a_2^{old}$

　は最小になっても$+a_1^{new}$が最小値0だから、　$a_2^{new} \ge a_2^{old} - a_1^{old}$

よって、

$U = \max(0, C - a_1^{old} + a_2^{old})$,　$V = \min(C, a_2^{old} - a_1^{old})$　とおくと

$$U \le a_2^{new} \le V \qquad (SMO-5)$$

# $a_2^{new}$の更新式の導出

$$\max \tilde{L}(\mathbf{a}) = \max\left[\sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m y_n y_m k(\mathbf{x}_n, \mathbf{x}_m)\right] \quad \cdots(SVM\,70)$$

を、$a_1, a_2$に関連する部分だけに注目して最適化することにする。

$$v_i = \sum_{j=3}^{N} y_j a_j k(\mathbf{x}_i, \mathbf{x}_j) = f(\mathbf{x}_i) - \sum_{j=1}^{2} y_j a_j k(\mathbf{x}_i, \mathbf{x}_j) \quad for\ i = 1,2$$

と定義する。

すると$(SVM\,70)$の$a_1, a_2$に関連する部分$W(a_1, a_2)$は次のように書ける。

ただし、$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$と略記する。

$$W(a_1, a_2) = a_1 + a_2 - \frac{1}{2}K_{11}a_1^2 - \frac{1}{2}K_{22}a_2^2 - y_1 y_2 a_1 a_2 K_{12} - y_1 a_1 v_1 - y_2 a_2 v_2 + 定数$$

ここで$a_1^{new} y_1 + a_2^{new} y_2 = a_1^{old} y_1 + a_2^{old} y_2 = a_1 y_1 + a_2 y_2$の両辺に$y_1$を掛け、

$y_1^2 = 1$に注意し、$s = y_1 y_2$とおくと上の式は

$$a_1^{new} + s a_2^{new} = a_1 + s a_2 = \gamma$$

$$W(a_2) = \gamma - s a_2 + a_2 - \frac{1}{2}K_{11}(\gamma - s a_2)^2 - \frac{1}{2}K_{22}a_2^2$$
$$- s(\gamma - s a_2)a_2 K_{12} - y_1(\gamma - s a_2)v_1 - y_2 a_2 v_2 + 定数 \quad (SMO6)$$

$W(a_2)$の最大化のために$a_2$で微分して$=0$とおく。

$$\frac{\partial W(a_2)}{\partial a_2}=0$$

$$\rightarrow 1-s+sK_{11}(\gamma-sa_2)-K_{22}a_2+K_{12}a_2-sK_{12}(\gamma-sa_2)+y_2v_1-y_2v_2=0$$

$s^2=1$　　$sy_1=y_1y_2y_1=y_1^{\,2}y_2=y_2$に注意。

またこの式の$a_2$が更新された$a_2^{\,new}$であるから

$$\rightarrow \quad a_2^{\,new}(K_{11}+K_{22}-2K_{12})=1-s+\gamma s(K_{11}-K_{12})+y_2(v_1-v_2)$$

$$=y_2(y_2-y_1+\gamma y_1(K_{11}-K_{12})+v_1-v_2) \qquad (SMO7)$$

ここで　$v_i=f(\mathbf{x}_i)-\sum_{j=1}^{2}y_j\,a_jK_{ij}$　に入っている$a_1,a_2$は古い値なので、

上記$(SMO7)$が更新式となる。

(*SMO*7)の両辺に$y_2$を掛け、もう少し書き直して整理してみよう。

$$a_2^{new} y_2 (K_{11} + K_{22} - 2K_{12})$$

$$= y_2 - y_1 + \gamma y_1 (K_{11} - K_{12}) + f(\mathbf{x}_1) - \sum_{j=1}^{2} y_j a_j K_{1j} - f(\mathbf{x}_2) + \sum_{j=1}^{2} y_j a_j K_{2j}$$

$\gamma = a_1 + s a_2$　および　$\gamma y_1 = y_1 a_1 + s y_1 a_2 = y_1 a_1 + y_2 a_2$に注意し書き直すと

$$= y_2 - y_1 + f(\mathbf{x}_1) - f(\mathbf{x}_2) + (y_1 a_1 + y_2 a_2)(K_{11} - K_{12}) - y_1 a_1 K_{11} - y_2 a_2 K_{12} + y_1 a_1 K_{21} + y_2 a_2 K_{22}$$

$$= y_2 - y_1 + f(\mathbf{x}_1) - f(\mathbf{x}_2) + y_2 a_2 (K_{11} + K_{22} - 2K_{12}) \quad \because K_{12} = K_{21}$$

$\therefore a_2^{new}$の更新式は、両辺に$y_2$を掛け、$(K_{11} + K_{22} - 2K_{12})$で割れば、$a_2 = a_2^{old}$として

$$a_2^{new} = a_2^{old} + \frac{y_2 \left( (f(\mathbf{x}_1) - y_1) - (f(\mathbf{x}_2) - y_2) \right)}{K_{11} + K_{22} - 2K_{12}} \qquad (SMO8)$$

この結果の$a_2^{new}$の値に対して(*SMO*4)(*SMO*5)

の条件で制約したものを$a_2^{new}$の更新値とする。

$a_1^{new}$は$a_1^{new} y_1 + a_2^{new} y_2 = a_1^{old} y_1 + a_2^{old} y_2$の両辺に$y_1$を掛け、今更新した$a_2^{new}$によって

$$a_1^{new} = a_1^{old} + y_1 y_2 (a_2^{old} - a_2^{new}) \qquad (SMO9)$$

# SVMによる回帰

- SVMは本来、2クラス分類器であり、識別モデルである。
- しかし、回帰すなわち生成モデルにも使える。
- 線形回帰では次の式を最小化した。

$$\sum_{n=1}^{N}\left(y_n - y(\mathbf{x}_n)\right)^2 + \lambda \| w \|^2$$

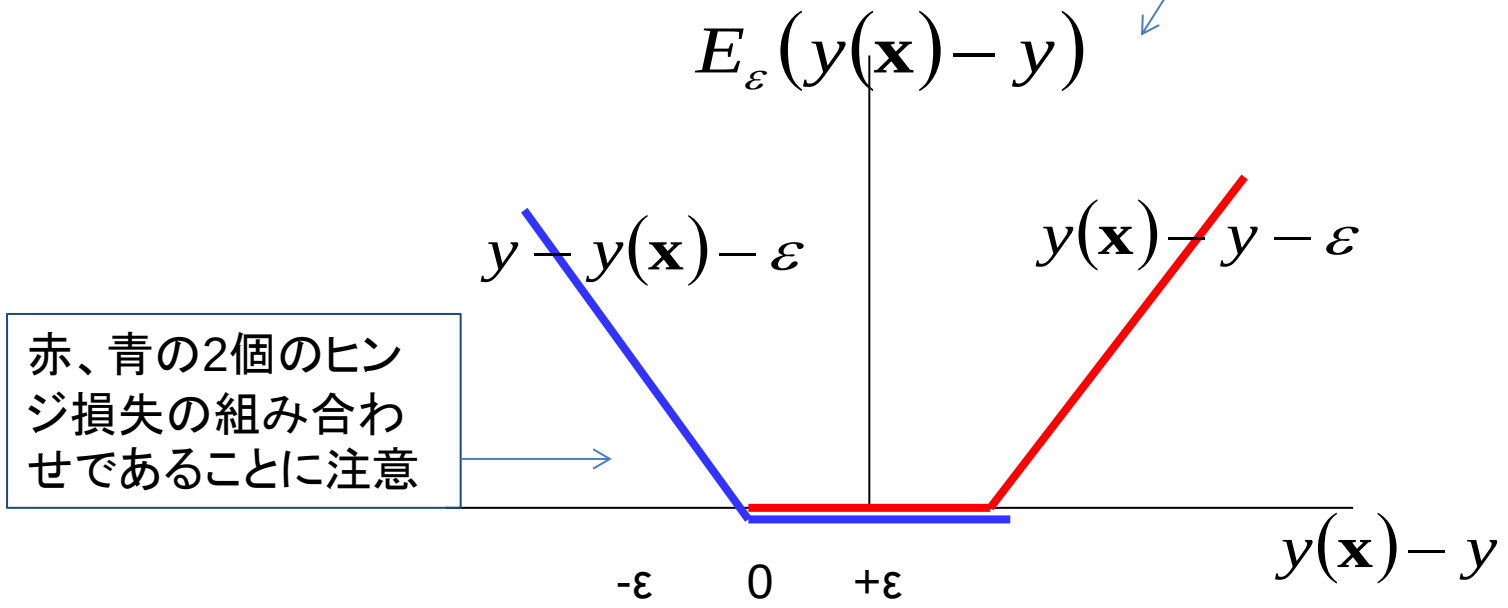
- この考え方を拡張する。

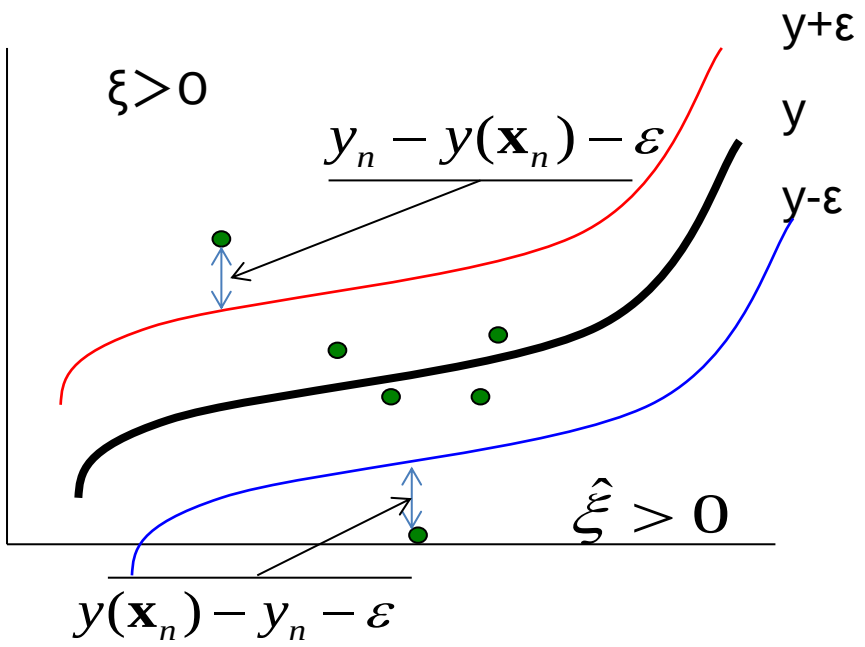
$$E_\varepsilon(y(\mathbf{x})-y)=\begin{cases}0 & \text{if } |y(x)-y|<\varepsilon \\ |y(x)-y|-\varepsilon & \text{otherwise}\end{cases}$$

下図参照

この関数を用いると回帰は、次の関数の最小化問題となる。

$$\min\left\{C\sum_{n=1}^{N}E_\varepsilon(y(\mathbf{x}_n)-y_n)+\frac{1}{2}\|\mathbf{w}\|^2\right\}$$






ここで、上図の赤い線の上で正となるスラック変数$\xi$と、

青い線の下で正となるスラック変数$\hat{\xi}$を導入する。

すなわち

$y(\mathbf{x}_n) - \varepsilon \le y_n \le y(\mathbf{x}_n) + \varepsilon$　　で０、その外側で＞0という条件は下記。

$$y_n - y(\mathbf{x}_n) - \varepsilon \le \xi_n \qquad \cdots(SVM100)$$

$$y(\mathbf{x}_n) - y_n - \varepsilon \le \hat{\xi}_n \qquad \cdots(SVM101)$$

こうすると最適化問題は

$$\min_{\mathbf{w},\xi_n,\hat{\xi}_n} C\sum_{n=1}^{N}\left(\xi_n+\hat{\xi}_n\right)+\frac{1}{2}\|\mathbf{w}\|^2$$

subject to

$$y_n - y(\mathbf{x}_n) - \varepsilon \le \xi_n \qquad \cdots(SVM100)$$

$$y(\mathbf{x}_n) - y_n - \varepsilon \le \hat{\xi}_n \qquad \cdots(SVM101)$$

$$\xi_n \ge 0 \qquad \cdots(SVM102)$$

$$\hat{\xi}_n \ge 0 \qquad \cdots(SVM103)$$

Lagrangianは

$$L = C\sum_{n=1}^{N}\left(\xi_n+\hat{\xi}_n\right)+\frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N}\left(\mu_n\xi_n+\hat{\mu}_n\hat{\xi}_n\right)$$

$$+\sum_{n=1}^{N}a_n\left(y_n-\varepsilon-\xi_n-y(\mathbf{x}_n)\right)+\sum_{n=1}^{N}\hat{a}_n\left(-y_n-\varepsilon-\hat{\xi}_n+y(\mathbf{x}_n)\right) \qquad \cdots(SVM110)$$

$$L = C\sum_{n=1}^{N}\left(\xi_n + \hat{\xi}_n\right) + \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N}\left(\mu_n\xi_n + \hat{\mu}_n\hat{\xi}_n\right)$$

$$-\sum_{n=1}^{N} a_n\left(\varepsilon + \xi_n + y(\mathbf{x}_n) - y_n\right) - \sum_{n=1}^{N}\hat{a}_n\left(\varepsilon + \hat{\xi}_n - y(\mathbf{x}_n) + y_n\right) \quad \cdots(SVM110)$$

$y(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + w_0$ を代入すると

$$L = C\sum_{n=1}^{N}\left(\xi_n + \hat{\xi}_n\right) + \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N}\left(\mu_n\xi_n + \hat{\mu}_n\hat{\xi}_n\right)$$

$$+\sum_{n=1}^{N} a_n\left(y_n - \varepsilon - \xi_n - \langle \mathbf{w}, \mathbf{x}_n \rangle - w_0\right) + \sum_{n=1}^{N}\hat{a}_n\left(-y_n - \varepsilon - \hat{\xi}_n + \langle \mathbf{w}, \mathbf{x}_n \rangle + w_0\right)$$

その上で$\mathbf{w}, w_0, \xi_n, \hat{\xi}_n$で微分

$L$の最小化のために微分

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{n=1}^{N} (a_n - \hat{a}_n)\mathbf{x}_n$$

$$\frac{\partial L}{\partial w_0} = 0 \quad \Rightarrow \quad \sum_{n=1}^{N} (a_n - \hat{a}_n) = 0$$

$$\frac{\partial L}{\partial \xi_n} = 0 \quad \Rightarrow \quad a_n + \mu_n = C \qquad \frac{\partial L}{\partial \hat{\xi}_n} = 0 \quad \Rightarrow \quad \hat{a}_n + \hat{\mu}_n = C$$

この結果を$L$に代入し最大化すべき$\tilde{L}$を求めると

$$\tilde{L}(\mathbf{a},\hat{\mathbf{a}}) = -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}(a_n - \hat{a}_n)(a_m - \hat{a}_m)\langle \mathbf{x}_n, \mathbf{x}_m \rangle$$

$$-\varepsilon \sum_{n=1}^{N}(a_n + \hat{a}_n) + \sum_{n=1}^{N}(a_n - \hat{a}_n) y_n \qquad -(SVM120)$$

$k(\mathbf{x}_n, \mathbf{x}_m) = \langle \mathbf{x}_n, \mathbf{x}_m \rangle$とも書く

この$\tilde{L}$を最適化する$a_n, \hat{a}_n$を求める問題に帰着した。

この式の導出は次々ページ以降に記述

上の導出過程から次の制約が得られる。

$0 \le a_n \le C \qquad 0 \le \hat{a}_n \le C$

また、回帰モデルは以下のようになる。

$$y(\mathbf{x}) = \sum_{n=1}^{N} (a_n - \hat{a}_n) \langle \mathbf{x}, \mathbf{x}_n \rangle + w_0 \qquad \cdots (SVM120)$$

$(SVM110)$ より $a_n \neq 0, \quad \hat{a}_n \neq 0$ のときは

$w_0 = y_n - \varepsilon - \langle \mathbf{w}, \mathbf{x}_n \rangle \quad \mathbf{w} = \sum_{n=1}^{N} (a_n - \hat{a}_n) \mathbf{x}_n$ を代入し

$$= y_n - \varepsilon - \sum_{m=1}^{N} (a_m - \hat{a}_m) \langle \mathbf{x}_n, \mathbf{x}_m \rangle$$

# （SVM120）の導出

*KKT*条件は以下のようになり、導入された変数と制約条件を消せる。

$$a_n\left(y_n-\varepsilon-\xi_n-y(\mathbf{x}_n)\right)=0 \qquad (a-1)$$

$$\hat{a}_n\left(-y_n-\varepsilon-\hat{\xi}_n+y(\mathbf{x}_n)\right)=0 \qquad (a-2)$$

ε+ξ>0, ε+ξ^>0

$$\mu_n\xi_n=(C-a_n)\xi_n=0 \qquad \hat{\mu}_n\hat{\xi}_n=(C-\hat{a}_n)\hat{\xi}_n=0$$

$$not\left((y_n-\varepsilon-\xi_n-y(\mathbf{x}_n))=0\text{かつ}(-y_n-\varepsilon-\hat{\xi}_n+y(\mathbf{x}_n))=0\right) \Rightarrow a_n\hat{a}_n=0$$

$$L=C\sum_{n=1}^{N}\left(\xi_n+\hat{\xi}_n\right)+\frac{1}{2}\|\mathbf{w}\|^2-\sum_{n=1}^{N}\left(\mu_n\xi_n+\hat{\mu}_n\hat{\xi}_n\right)$$

$$+\sum_{n=1}^{N}a_n\left(y_n-\varepsilon-\xi_n-\langle\mathbf{w},x_n\rangle-w_0\right)+\sum_{n=1}^{N}\hat{a}_n\left(-y_n-\varepsilon-\hat{\xi}_n+\langle\mathbf{w},x_n\rangle+w_0\right)$$

$$\mu_n=C-a_n \quad \hat{\mu}_n=C-\hat{a}_n \qquad \because \frac{\partial L}{\xi_n}=\frac{\partial L}{\hat{\xi}_n}=0$$

$$=\sum_{n=1}^{N}\left((C-a_n)\xi_n+(C-\hat{a}_n)\hat{\xi}_n\right)-\sum_{n=1}^{N}\left((C-a_n)\xi_n+(C-\hat{a}_n)\hat{\xi}_n\right)$$

$$+\frac{1}{2}\|\mathbf{w}\|^2+\sum_{n=1}^{N}a_n\left(y_n-\varepsilon-\langle\mathbf{w},x_n\rangle-w_0\right)+\sum_{n=1}^{N}\hat{a}_n\left(-y_n-\varepsilon+\langle\mathbf{w},x_n\rangle+w_0\right)$$

$$= \sum_{n=1}^{N} \left( (C - a_n)\xi_n + (C - \hat{a}_n)\hat{\xi}_n \right) - \sum_{n=1}^{N} \left( (C - a_n)\xi_n + (C - \hat{a}_n)\hat{\xi}_n \right)$$

$$+ \frac{1}{2} \| \mathbf{w} \|^2 + \sum_{n=1}^{N} a_n \left( y_n - \varepsilon - \langle \mathbf{w}, \mathbf{x}_n \rangle - w_0 \right) + \sum_{n=1}^{N} \hat{a}_n \left( - y_n - \varepsilon + \langle \mathbf{w}, \mathbf{x}_n \rangle + w_0 \right)$$

$$\mathbf{w} = \sum_{n=1}^{N} (a_n - \hat{a}_n)\mathbf{x}_n \quad と \quad \frac{\partial L}{\partial w_0} = 0 \rightarrow \quad \sum_{n=1}^{N} (a_n - \hat{a}_n) = 0 \qquad により$$

$$= -\frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} (a_n - \hat{a}_n)(a_m - \hat{a}_m) \langle \mathbf{x}_n, \mathbf{x}_m \rangle - \varepsilon \sum_{n=1}^{N} (a_n + \hat{a}_n) + \sum_{n=1}^{N} (a_n - \hat{a}_n) y_n$$

$$= -\frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} (a_n - \hat{a}_n)(a_m - \hat{a}_m) \langle \mathbf{x}_n, \mathbf{x}_m \rangle - \varepsilon \sum_{n=1}^{N} (a_n + \hat{a}_n) + \sum_{n=1}^{N} (a_n - \hat{a}_n) y_n$$

# カーネル

($M$は教師データの次元数, $N$は教師データ数)

$$\mathbf{\Phi} = \left(\phi(\mathbf{x}_1), ..., \phi(\mathbf{x}_N)\right)^T = \begin{pmatrix} \phi_1(\mathbf{x}_1) & \cdots & \phi_M(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_1(\mathbf{x}_N) & \cdots & \phi_M(\mathbf{x}_N) \end{pmatrix}$$

計画行列
Design Matrix

$$\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^T = \begin{pmatrix} \phi_1(\mathbf{x}_1) & \cdots & \phi_M(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_1(\mathbf{x}_N) & \cdots & \phi_M(\mathbf{x}_N) \end{pmatrix} \begin{pmatrix} \phi_1(\mathbf{x}_1) & \cdots & \phi_1(\mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ \phi_M(\mathbf{x}_1) & \cdots & \phi_M(\mathbf{x}_N) \end{pmatrix}$$

カーネル関数:$k(x_i, y_j)$を要素とするグラム行列

$$= \begin{pmatrix} \sum_{i=1}^{M} \phi_i(\mathbf{x}_1)\phi_i(\mathbf{x}_1) & \cdots & \sum_{i=1}^{M} \phi_i(\mathbf{x}_1)\phi_i(\mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^{M} \phi_i(\mathbf{x}_N)\phi_i(\mathbf{x}_1) & \cdots & \sum_{i=1}^{M} \phi_i(\mathbf{x}_N)\phi_i(\mathbf{x}_N) \end{pmatrix} = \begin{pmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \cdots & k(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_N, \mathbf{x}_1) & \cdots & k(\mathbf{x}_N, \mathbf{x}_N) \end{pmatrix}$$

$\phi_i(\mathbf{x}_k) = \mathbf{x}_k$の第$i$成分 : $x_{ki}$なら　$k(\mathbf{x}_k, \mathbf{x}_l) = \sum_{i=1}^{M} x_{ki}x_{li} = \langle \mathbf{x}_k, \mathbf{x}_l \rangle$ : 内積

- 正定値カーネル： 対称行列k(xi,xj)が半正定値：グラム行列:
  既存のカーネル関数から別のカーネル関数を構成する方法

$$k(\mathbf{x},\mathbf{y}) = ck_1(\mathbf{x},\mathbf{y}) \quad \cdots(k-1)$$

$$k(\mathbf{x},\mathbf{y}) = f(\mathbf{x})k_1(\mathbf{x},\mathbf{y})f(\mathbf{y}) \quad f\text{は任意の関数} \quad \cdots(k-2)$$

$$k(\mathbf{x},\mathbf{y}) = q(k_1(\mathbf{x},\mathbf{y})) \quad q\text{は係数正の多項式} \quad \cdots(k-3)$$

$$k(\mathbf{x},\mathbf{y}) = \exp(k_1(\mathbf{x},\mathbf{y})) \quad \cdots(k-4)$$

$$k(\mathbf{x},\mathbf{y}) = k_1(\mathbf{x},\mathbf{y}) + k_2(\mathbf{x},\mathbf{y}) \quad \cdots(k-5)$$

$$k(\mathbf{x},\mathbf{y}) = k_1(\mathbf{x},\mathbf{y})k_2(\mathbf{x},\mathbf{y}) \quad \cdots(k-6)$$

$$k(\mathbf{x},\mathbf{y}) = k_1(\phi(\mathbf{x}),\phi(\mathbf{y})) \quad \cdots(k-7)$$

$$k(\mathbf{x},\mathbf{y}) = \mathbf{x}^T\mathbf{A}\mathbf{y} \quad \mathbf{A}\text{は対称半正定値行列} \quad \cdots(k-8)$$

$\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_b)$ $\mathbf{y} = (\mathbf{y}_a, \mathbf{y}_b)$ ($\mathbf{x},\mathbf{y}$とも同じ次元に分割）のとき

$$k(\mathbf{x},\mathbf{y}) = k_1(\mathbf{x}_a,\mathbf{y}_a) + k_2(\mathbf{x}_b,\mathbf{y}_b) \quad \cdots(k-9)$$

$$k(\mathbf{x},\mathbf{y}) = k_1(\mathbf{x}_a,\mathbf{y}_a)k_2(\mathbf{x}_b,\mathbf{y}_b) \quad \cdots(k-10)$$

# カーネルの例

- 問題の非線形性あるいは高次性を非線形なカーネルで表すことになる

- 線形カーネル $k(\mathbf{x},\mathbf{y}) = \mathbf{x}^T\mathbf{y}$
- 多項式カーネル $k(\mathbf{x},\mathbf{y}) = (\mathbf{x}^T\mathbf{y}+1)^M \quad M = 1,2,...$
- Gaussianカーネル:RBF

$$k(\mathbf{x},\mathbf{y}) = \exp\left(-\frac{\| x-y \|^2}{2\sigma^2}\right)$$

  - 以下の分解によりGaussianカーネルがカーネル関数だといえる

by (k-1)(k-2)(k-4)

$$\| \mathbf{x}-\mathbf{y} \|^2 = \mathbf{x}^T\mathbf{x} + \mathbf{y}^T\mathbf{y} - 2\mathbf{x}^T\mathbf{y} \Rightarrow$$

$$k(\mathbf{x},\mathbf{y}) = \exp(-\frac{\mathbf{x}^T\mathbf{x}}{2\sigma^2})\exp(-\frac{\mathbf{y}^T\mathbf{y}}{2\sigma^2})\exp(\frac{\mathbf{x}^T\mathbf{y}}{\sigma^2})$$

# 表現定理

- SVMなどの最適化は下のような形

$$\min \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} a_n \{1 - t_n y(\mathbf{x}_n)\} \quad where \quad y(\mathbf{x}_n) = \langle \mathbf{w}, \mathbf{x}_n \rangle + w_0$$

$$\Rightarrow \min_{f \in \mathcal{H}} \Psi(\|f\|) + loss_{\{t_n, \mathbf{x}_n\}}(f)$$

正の単調増加関数である正則化項

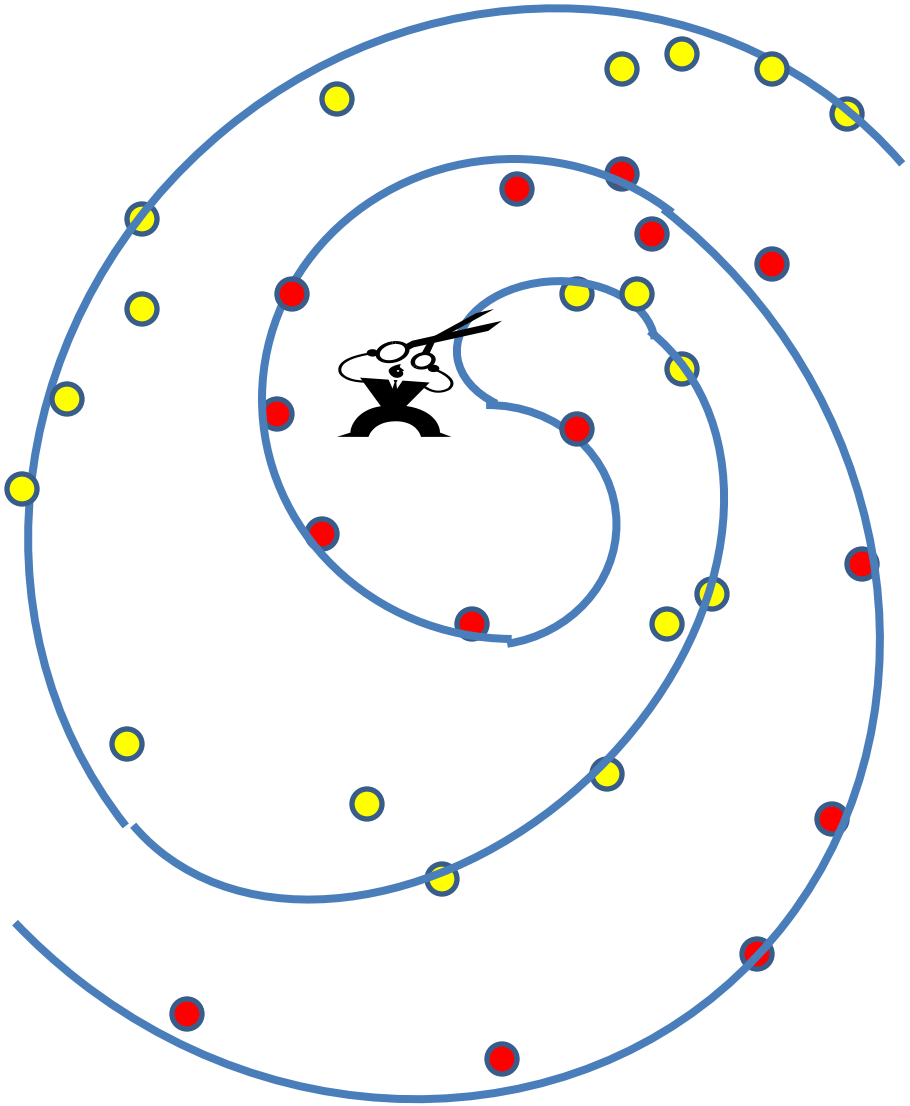
教師データに対する損失

- このとき上の最適化問題の解$f$は下記の形

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i k(\mathbf{x}, \mathbf{x}_n)$$

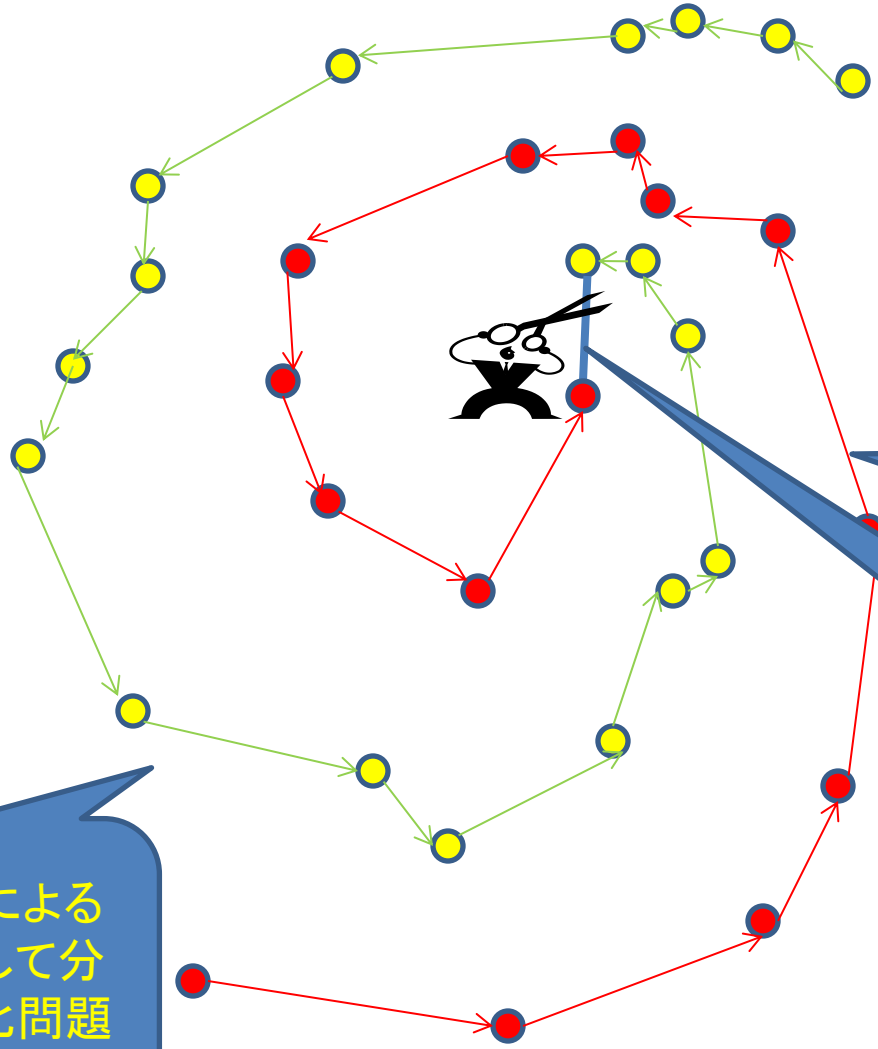
- よって、一般的なカーネルであっても(SVM100)の形の解を想定できる。

# よもやま話：スイスロール－１



多様体に写像
してから識別
境界を切る

# よもやま話：スイスロール－２



同じ色のデータのうち近接するものを繋いで、多様体を近似的に求め

色が違うデータを繋ぐところで切る

この多様体による制約を考慮して分類する最適化問題の定式化は？

# 捕捉：線形回帰、識別からカーネル関数へ

$y(\mathbf{w}) = \mathbf{w}^T \phi(\mathbf{x})$　という一般化した線形回帰式に対して

$$J(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left\{\mathbf{w}^T\phi(\mathbf{x}_n) - t_n\right\}^2 + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$$

$t_n$は$\mathbf{w}^T\phi(\mathbf{x}_n)$がとるべき値。

ただし　$\phi(\mathbf{x}_n) = \begin{pmatrix} \phi_1(\mathbf{x}_n) \\ \vdots \\ \phi_M(\mathbf{x}_n) \end{pmatrix}$

($M$は教師データの次元数
, $N$は教師データ数)

という正規化項つきの２乗誤差を考えるとき、

$\phi(\mathbf{x})$についてもう少し組織的に考えてみよう。

➢カーネル関数と呼ばれる$k(\phi(x),\phi(y))$で回帰や識別を考え直すことにより、より効率の良い方法が見えてくる。

# 双対表現

- まず、L2-正規化項つきのL2-損失（＝2乗誤差関数）を考える。

$$J(\mathbf{w})=\frac{1}{2}\sum_{n=1}^{N}\left\{\mathbf{w}^T\phi(\mathbf{x}_n)-t_n\right\}^2+\frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$$

$t_n$は$\mathbf{w}^T\phi(\mathbf{x}_n)$がとるべき値。　$\lambda\geq 0$

- $\frac{\partial J(w)}{\partial w}=0$ から$J(w)$を最小化するwを求め、その右辺をaとΦで表す。$N$は教師データ数

$$w=-\frac{1}{\lambda}\sum_{n=1}^{N}\left\{w^T\phi(\mathbf{x}_n)-t_n\right\}\cdot\phi(\mathbf{x}_n)=\sum_{n=1}^{N}a_n\phi(\mathbf{x}_n)=\mathbf{\Phi}^T\mathbf{a}$$

wが$\phi(\mathbf{x}_n)$の$a_n$を重みとする線形結合で書けることに注目

$$\mathbf{a}=(a_1,\cdots,a_N)^T \qquad a_n=-\frac{1}{\lambda}\left\{w^T\phi(\mathbf{x}_n)-t_n\right\}$$

- 以上の表記を用い、$J(w)$を$J(a)$として書き直すと

$$J(\mathbf{a})=\frac{1}{2}\mathbf{a}^T\mathbf{\Phi}\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{\Phi}^T\mathbf{a}-\mathbf{a}^T\mathbf{\Phi}\mathbf{\Phi}^T\mathbf{t}+\frac{1}{2}\mathbf{t}^T\mathbf{t}+\frac{\lambda}{2}\mathbf{a}^T\mathbf{\Phi}\mathbf{\Phi}^T\mathbf{a}$$

$$\mathbf{t}=(t_1,\cdots,t_N)^T$$

ここで下記の要素を持つ$N\times N$の$Gram$行列　$\mathbf{K}=\mathbf{\Phi}\mathbf{\Phi}^T$　を定義する。

$$K_{nm}=\phi(\mathbf{x}_n)^T\phi(\mathbf{x}_m)=k(\mathbf{x}_n,\mathbf{x}_m)$$　：カーネル関数という。

- カーネル関数を用いると、*J(a)*、*a*、*y(w)*は次のように書ける

$$J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{K}\mathbf{a} - \mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{t} + \frac{1}{2}\mathbf{t}^{T}\mathbf{t} + \frac{\lambda}{2}\mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{a}$$

ここで $\dfrac{\partial J(\mathbf{a})}{\partial \mathbf{a}} = 0$ より $\mathbf{a} = \left(\mathbf{K} + \lambda \mathbf{I}_N\right)^{-1}\mathbf{t}$

$$y(\mathbf{x}) = \mathbf{w}^{T}\phi(\mathbf{x}) = \mathbf{a}^{\mathrm{T}}\mathbf{\Phi}\phi(\mathbf{x})$$

- $M$個の基底関数$\varphi i$があったとすると、カーネル関数は、内積の形で以下のように書ける。

$$k(\mathbf{x},\mathbf{y}) = \phi(\mathbf{x})^{T}\phi(\mathbf{y}) = \sum_{i=1}^{M}\phi_i(\mathbf{x})\phi_i(\mathbf{y}) \qquad \phi(\mathbf{x}) = (\phi_1(\mathbf{x}), \cdots \phi_M(\mathbf{x}))^{T}$$