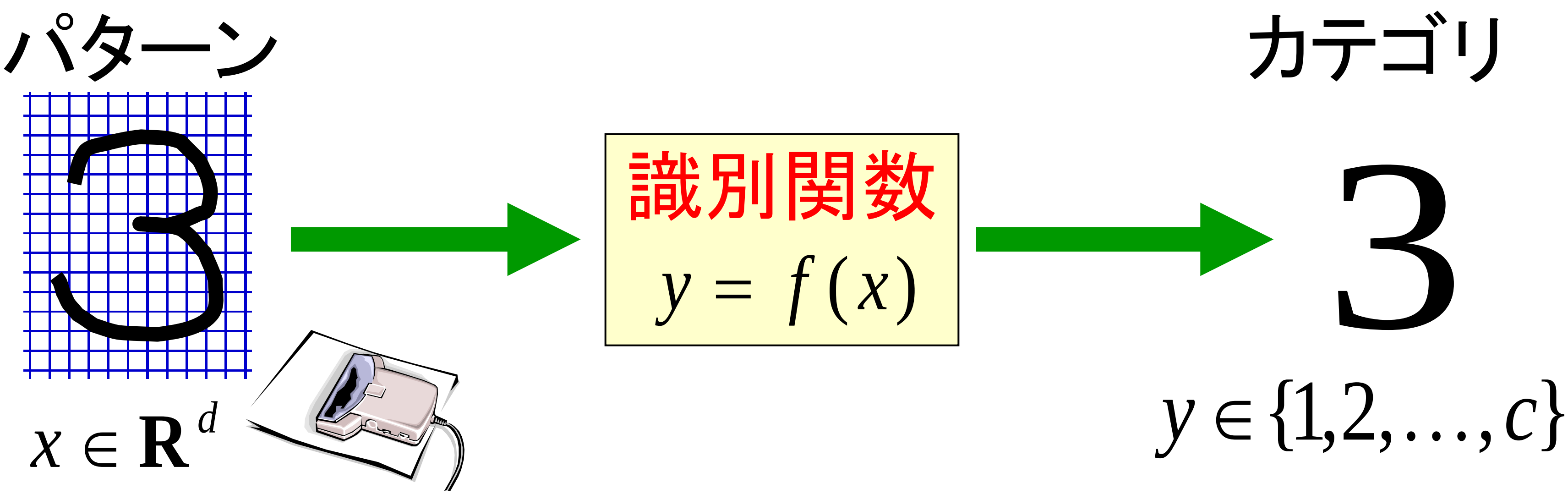
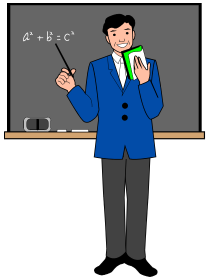
最小二乗分類
（７章）

杉山将・本多淳也
sugi@k.u-tokyo.ac.jp, jhonda@k.u-tokyo.ac.jp
http://www.ms.k.u-tokyo.ac.jp

パターン認識

- 入力パターンをカテゴリに割り当てる**識別関数**を構成する問題
  - 問題に合わせて人間が識別関数を設計
  - データから自動的に識別関数を学習

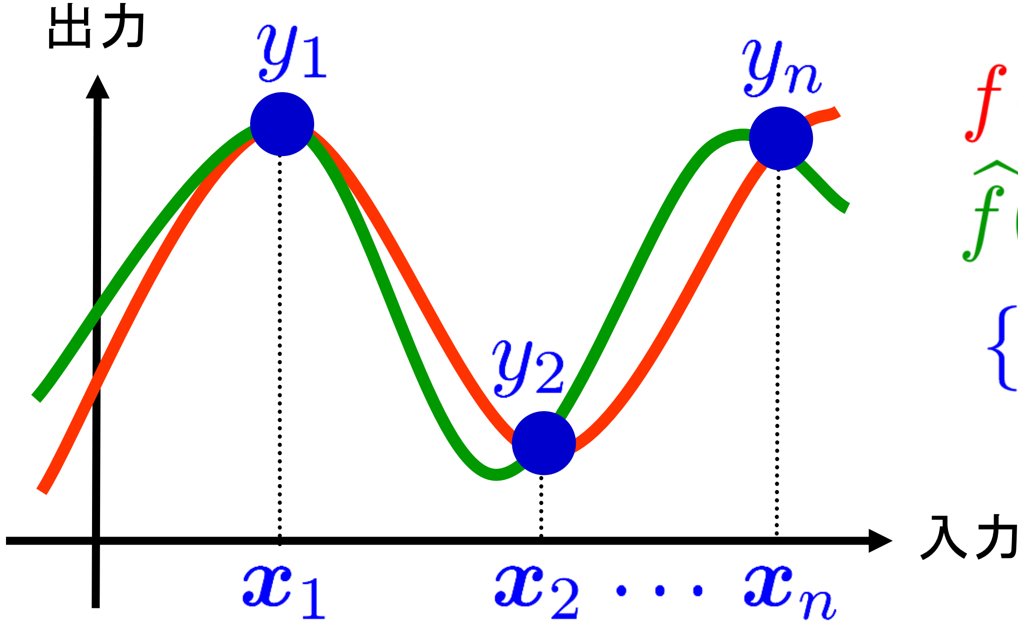




講義の流れ

1. 最小二乗回帰による分類
2. フィッシャー判別分析
3. 多クラスの分類問題
4. 0/1-損失とマージン

回帰の復習



$f(\boldsymbol{x})$:学習したい真の関数

$\widehat{f}(\boldsymbol{x})$:学習結果の関数

$\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$:訓練標本

$$y_i = f(\boldsymbol{x}_i)\ (+\mathrm{noise})$$

訓練標本から真の関数にできるだけ近い関数を求める

パラメータに関する線形モデル

- 線形モデル: $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sum_{j=1}^{b} \theta_j \phi_j(\boldsymbol{x})$

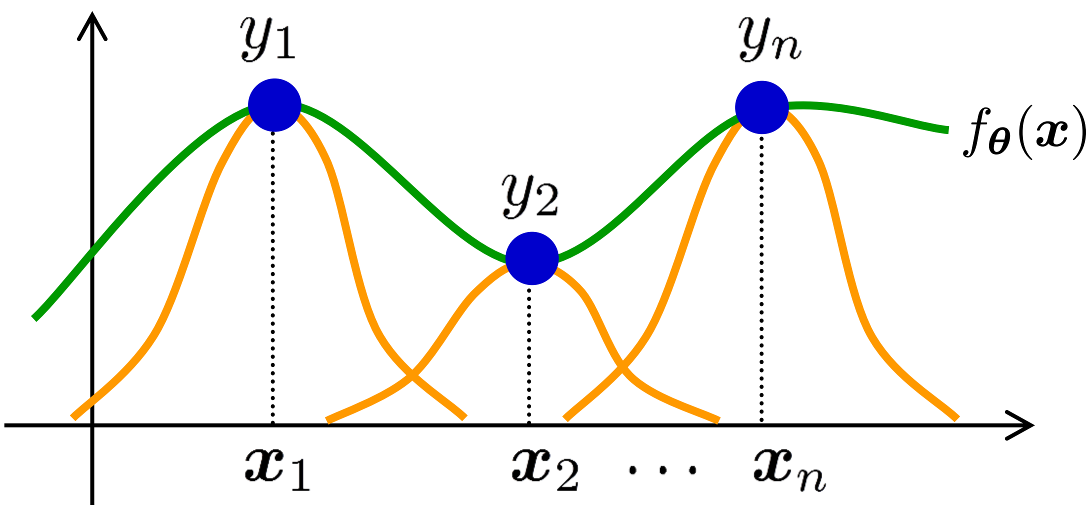
$\{\phi_j(\boldsymbol{x})\}_{j=1}^{b}$:基底関数

- カーネルモデル:

$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sum_{j=1}^{n} \theta_j K(\boldsymbol{x}, \boldsymbol{x}_j)$$

ガウスカーネル

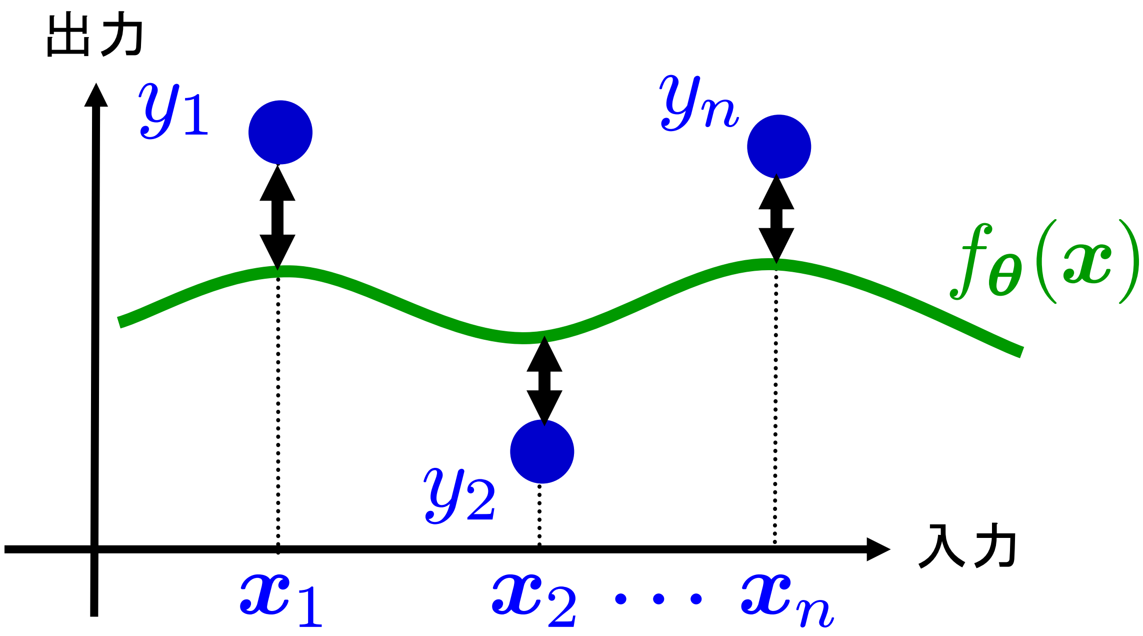
$$K(\boldsymbol{x}, \boldsymbol{c}) = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{c}\|^2}{2h^2}\right)$$



最小二乗回帰

- 訓練出力との二乗誤差を最小にする:

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - y_i \right)^2$$

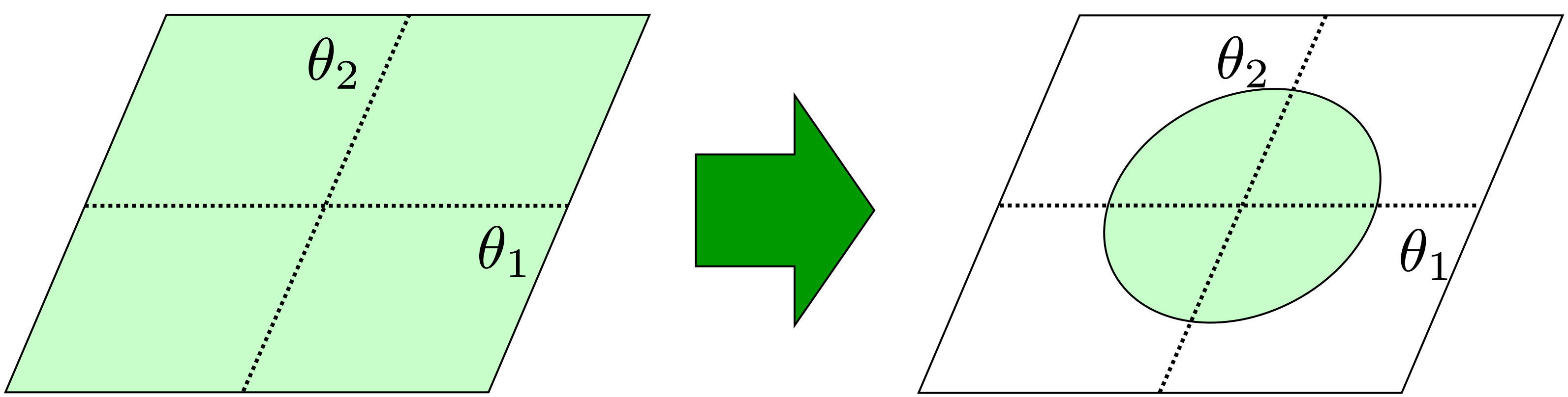


ℓ_2-拘束付き最小二乗回帰

- モデルを**超球**に限定することにより過適合を回避

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - y_i \Big)^2 \quad \text{subject to} \quad \|\boldsymbol{\theta}\|^2 \leq R$$

$$R \geq 0$$



通常の最小二乗回帰 → ℓ_2-拘束付き最小二乗回帰

様々な損失と拘束の組み合わせ

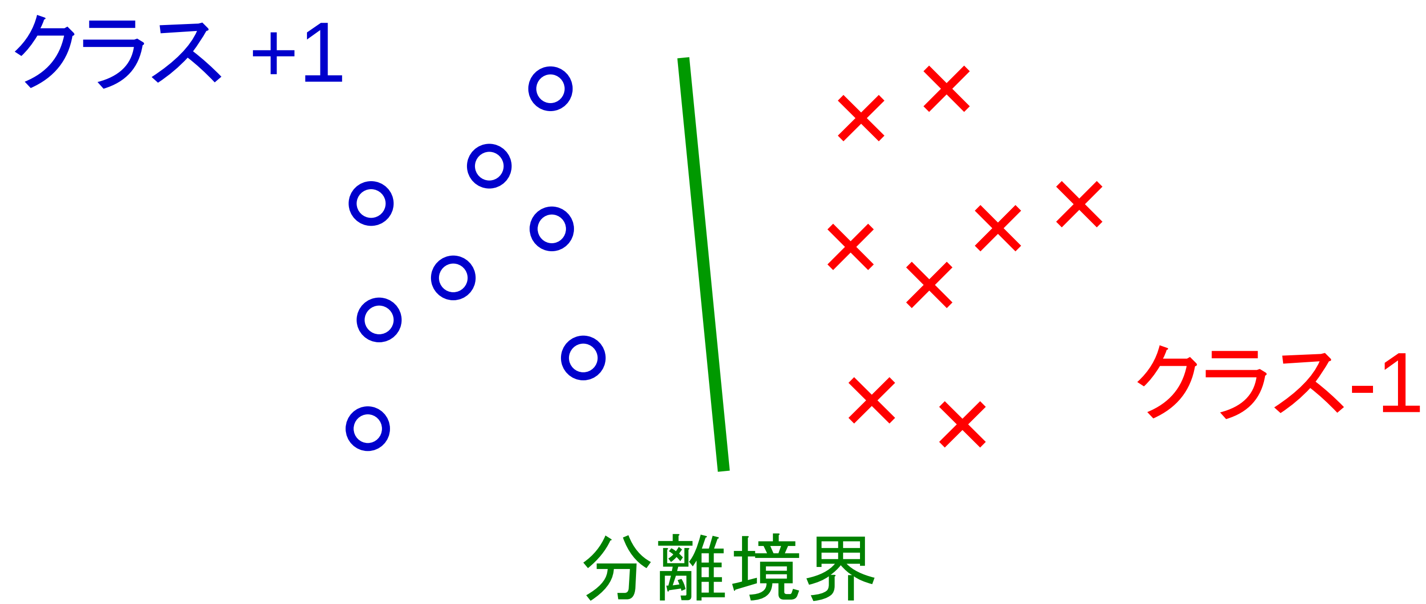
- 線形モデル／カーネルモデルに対する学習法:

拘束条件 損失関数	無し	ℓ_2-拘束 正則化	ℓ_1-拘束 正則化＆スパース
ℓ_2-損失　有効	解析解	解析解	二次計画
フーバー損失	二次計画	二次計画	二次計画
ℓ_1-損失　ロバスト	線形計画	二次計画	線形計画

- モデルや正則化パラメータは交差確認法で決定.

2クラスの分類問題

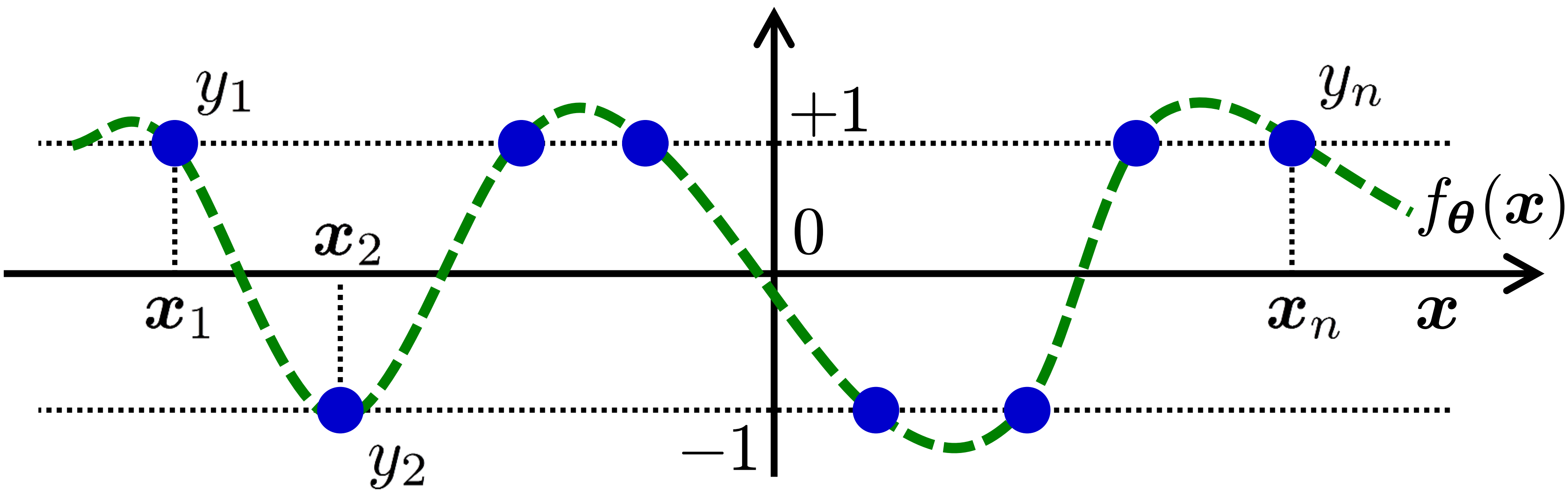
- ラベル付き訓練データ: $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$
  - 入力$\boldsymbol{x}$はd次元の実ベクトル $\boldsymbol{x} \in \mathbb{R}^d$
  - 出力yは2値のクラスラベル $y \in \{+1, -1\}$



- クラス間の分離境界を求めたい

2クラスの分類問題

- 2クラス分類問題は**2値関数の近似問題**として表現可能：



- 回帰における学習手法が分類にも使える！

回帰学習による分類

- パラメータを正則化最小二乗回帰で学習

$$\widehat{\boldsymbol{\theta}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\frac{1}{2} \sum_{i=1}^{n} \Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - y_i \Big)^2 + \frac{\lambda}{2} \|\boldsymbol{\theta}\|^2 \right]$$

$\lambda \ (\geq 0)$: 正則化パラメータ

- テストパターンの分類:

$$\widehat{y} = \mathrm{sign}\left(f_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{x}) \right) = \begin{cases} +1 & (f_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{x}) > 0) \\ 0 & (f_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{x}) = 0) \\ -1 & (f_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{x}) < 0) \end{cases}$$

実行例

- ガウスカーネルモデル

$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sum_{j=1}^{n} \theta_j K(\boldsymbol{x}, \boldsymbol{x}_j)$$

$$K(\boldsymbol{x}, \boldsymbol{c}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}\|^2}{2h^2}\right)$$

に対して，正則化最小二乗回帰を用いた分類アルゴリズムを実装する

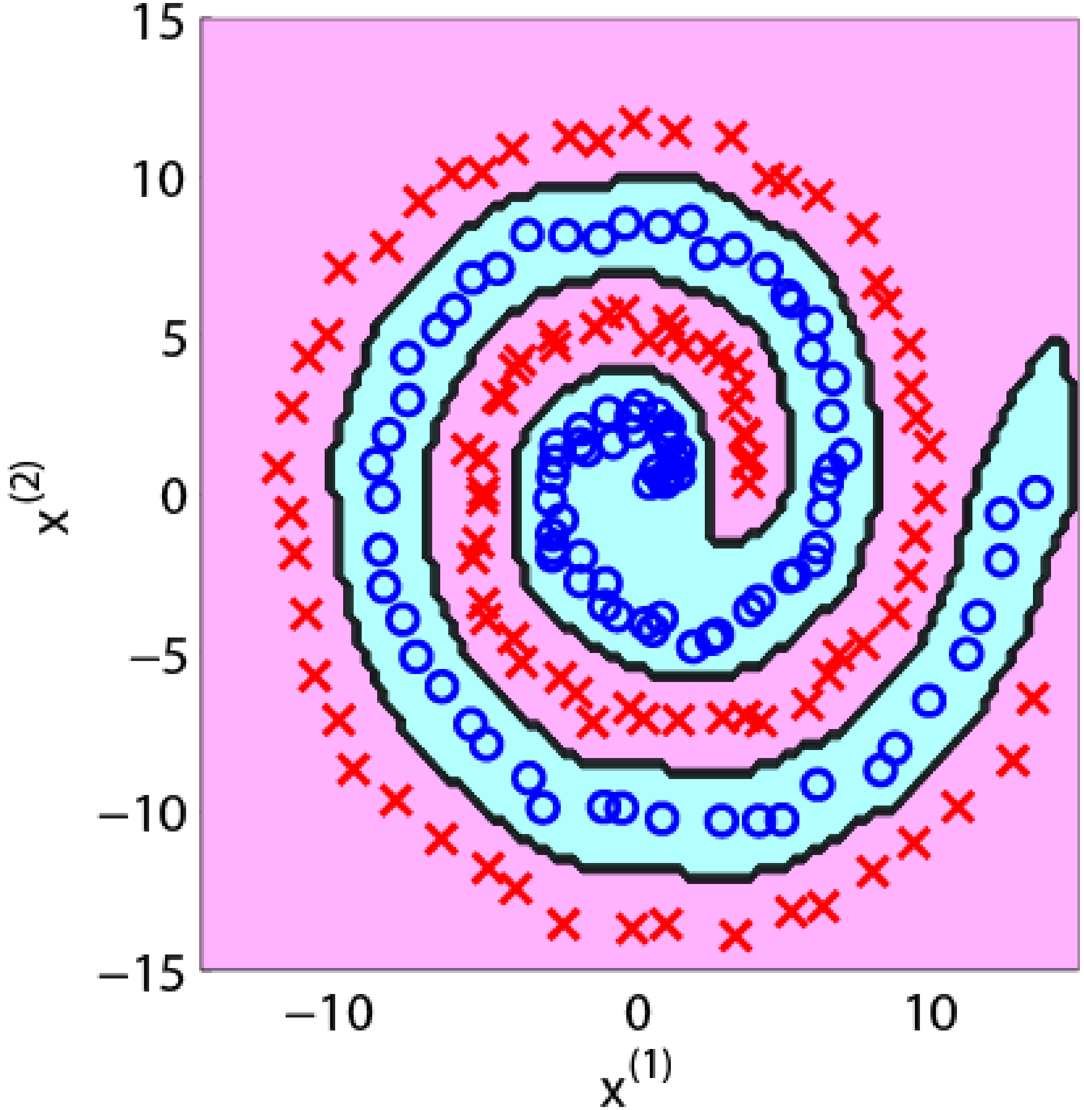
実行例(続き)


```
clear all; rand('state',0); randn('state',0);
n=200; a=linspace(0,4*pi,n/2);
u=[a.*cos(a) (a+pi).*cos(a)]'+1*rand(n,1);
v=[a.*sin(a) (a+pi).*sin(a)]'+1*rand(n,1);
x=[u v]; y=[ones(1,n/2) -ones(1,n/2)]';
x2=sum(x.^2,2); hh=2*1^2; l=0.01;
k=exp(-(repmat(x2,1,n)+repmat(x2',n,1)-2*x*x')/hh);
t=(k^2+l*eye(n))\(k*y);

m=100; X=linspace(-15,15,m)'; X2=X.^2;
U=exp(-(repmat(u.^2,1,m)+repmat(X2',n,1)-2*u*X')/hh);
V=exp(-(repmat(v.^2,1,m)+repmat(X2',n,1)-2*v*X')/hh);
figure(1); clf; hold on; axis([-15 15 -15 15]);
contourf(X,X,sign(V'*(U.*repmat(t,1,m))));
plot(x(y==1,1),x(y==1,2),'bo');
plot(x(y==-1,1),x(y==-1,2),'rx');
colormap([1 0.7 1; 0.7 1 1]);
```

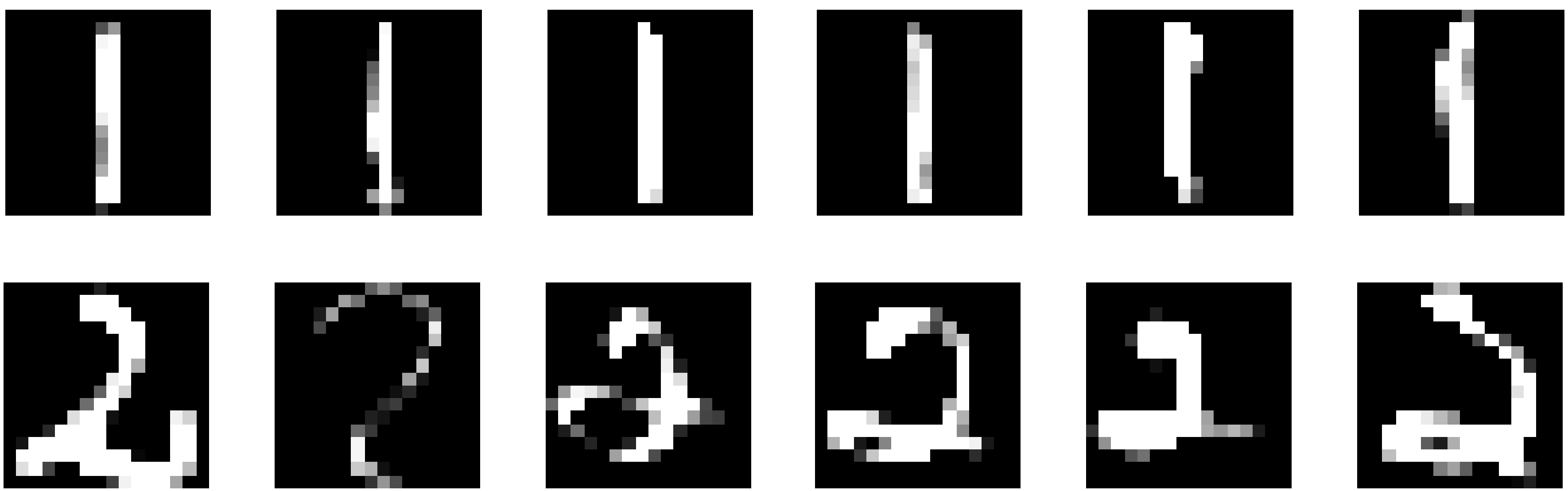
実行例(続き)

- 複雑な分離境界もうまく求められる



ガウスカーネル最小二乗分類による手書き数字認識

- 16×16画素，各画素の濃度は0から255
  - 訓練標本:1と2の各500文字ずつ計1000文字
  - テスト標本:1と2の各200文字ずつ計400文字



手書き数字データの読み込み

- 以下からファイルをダウンロードする:

http://www.ms.k.u-tokyo.ac.jp/software/SML.zip

- 次のようにして手書き数字のデータを読み込む.

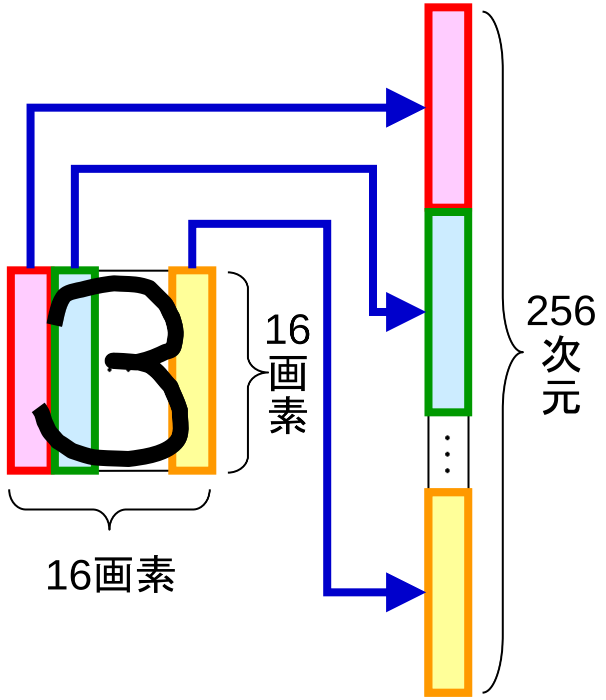
```
load digit.mat
```

- そうすると, XとTという変数が読み込まれる.
  - X: 訓練用の文字データ
  - T: テスト用の文字データ
- 読み込まれた変数はwhosコマンドで確認できる.

```
> whos
  Name            Size                Bytes  Class
  T               256x200x10        4096000  double
  X               256x500x10       10240000  double
```

手書き数字データの詳細

- XとTは3次元の配列.
- 一つの手書き文字データは256次元のベクトルで表されている.
- これは16×16画素の画像データを一列に並べたベクトルであり，それぞれの要素は-1から1までの実数を取る.



手書き数字データの詳細

- 値が-1のとき：画素は黒
- 値が1のとき：画素は白
- X：0から9からまでの各数字が500文字ずつ
- T：0から9からまでの各数字が200文字ずつ
- 例えば，23番目の訓練用の手書き数字5のデータを変数xに取り出すときは

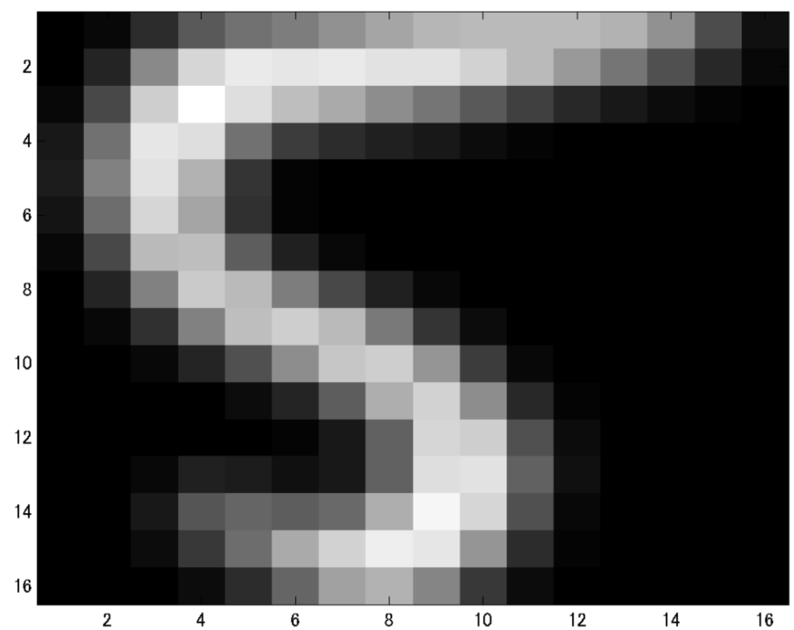
```
x=X(:,23,5);
```

  とすればよい．
- 手書き数字0のデータは，3番目の引数が10の場合に対応するので注意すること．

手書き数字データの表示

- 取り出した手書き文字のデータの画像は，以下のようにして表示することができる．

```
imagesc(reshape(x,[16 16])')
colormap(gray)
```



実行例(続き)


```
clear all; rand('state',0); randn('state',0);
load digit.mat
x=[X(:,:,1) X(:,:,2)]; y=[ones(500,1); -ones(500,1)];
n=length(y); x2=sum(x.^2,1); hh=2*10^2; l=1;
k=exp(-(repmat(x2,n,1)+repmat(x2',1,n)-2*x'*x)/hh);
t=(k^2+l*eye(n))\(k*y);

u=T(:,:,1); % Test patterns 1
v=exp(-(repmat(x2,200,1)+repmat(sum(u.^2,1)',1,n)-
2*u'*x)/hh)*t;
C(1,1)=sum(sign(v)>=0); C(1,2)=sum(sign(v)<0);
u=T(:,:,2); % Test patterns 2
v=exp(-(repmat(x2,200,1)+repmat(sum(u.^2,1)',1,n)-
2*u'*x)/hh)*t;
C(2,1)=sum(sign(v)>=0); C(2,2)=sum(sign(v)<0);

C
```

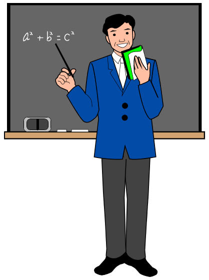
実験結果

- ３９９／４００＝９９．７５％の正解率

予測したカテゴリ

正解のカテゴリ

	1	2
1	199	1
2	0	200



講義の流れ

1. 最小二乗回帰による分類
2. フィッシャー判別分析
3. 多クラスの分類問題
4. 0/1-損失とマージン

統計的パターン認識

- 訓練標本：属するカテゴリが既知のパターン

$$\{(x_i, y_i)\}_{i=1}^{n}$$

$$x_i \in \mathbf{R}^d$$

$$y_i \in \{1, 2, \ldots, c\}$$

- 統計的パターン認識：訓練標本の統計的な性質を利用して識別関数を学習する
- 仮定：

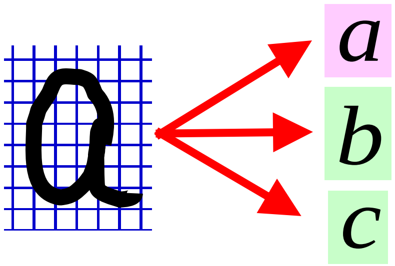
$$(x_i, y_i) \overset{\text{i.i.d.}}{\sim} p(x, y)$$

i.i.d. (independent and identically distributed)

独立に同一の分布に従う

理想的なパターン分類法

- 事後確率 $p(y\mid x)$：与えられたパターン x がクラス y に属する確率



- 事後確率を最大にするカテゴリにパターンを分類すれば，パターンの誤識別率が最小になる.

$$f(x)=\arg\max_{y}\ p(y\mid x)$$

- 実際には事後確率は未知なので，訓練標本から推定しなければならない.

生成モデルに基づく分類

- クラス事後確率を最大にするように分類:

$$\arg\max_{y}\ p(y \mid x)$$

- ベイズの公式により，識別モデルを生成モデルで表現し，分解:

$$\underbrace{p(y|x)}_{\text{識別モデル}} = \frac{p(x,y)}{p(x)} \propto \underbrace{p(x,y)}_{\text{生成モデル}} = \underbrace{p(x|y)}_{\text{条件付き確率}}\underbrace{p(y)}_{\text{事前確率}}$$

- 対数をとると

$$\log p(y \mid x) = \log p(x \mid y) + \log p(y) + C$$

$C = -\log p(x)$: 定数

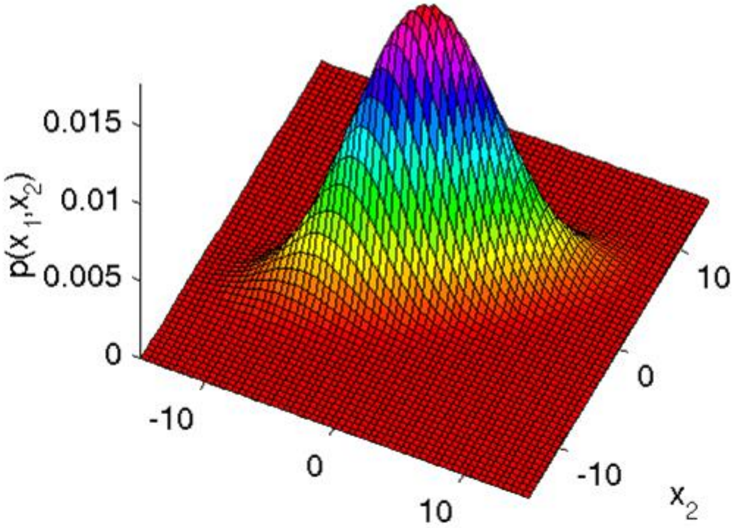
事前確率と条件付き確率の推定

- **事前確率** $p(y)$: クラス y に含まれる訓練標本の割合で推定

$$\hat{p}(y) = \frac{n_y}{n}$$

n_y :クラス y に属する訓練標本数

- **条件付き確率** $p(x|y)$: ガウスモデルを仮定し, 期待値と共分散行列を**最尤推定**



$$q(x;\mu_y,\Sigma_y) = \frac{1}{(2\pi)^{d/2}\det(\Sigma_y)^{1/2}}\exp\left(-\frac{1}{2}(x-\mu_y)^T\Sigma_y^{-1}(x-\mu_y)\right)$$

最尤推定法

- 最尤推定法(maximum likelihood estimation): 手元にある訓練標本が最も生起しやすいようにパラメータ値を決める方法
  - “最も尤もらしいようにパラメータの値を決める”
- 訓練標本 $\{x_i\}_{i=1}^{n}$ がモデル $q(x;\theta)$ から生起する確率密度:

$$p(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} q(x_i;\theta)$$

- 尤度(likelihood): これを θ の関数とみたもの

$$L(\theta) = \prod_{i=1}^{n} q(x_i;\theta)$$

最尤推定法(続き)

- 尤度を最大にするようにパラメータの値を決定

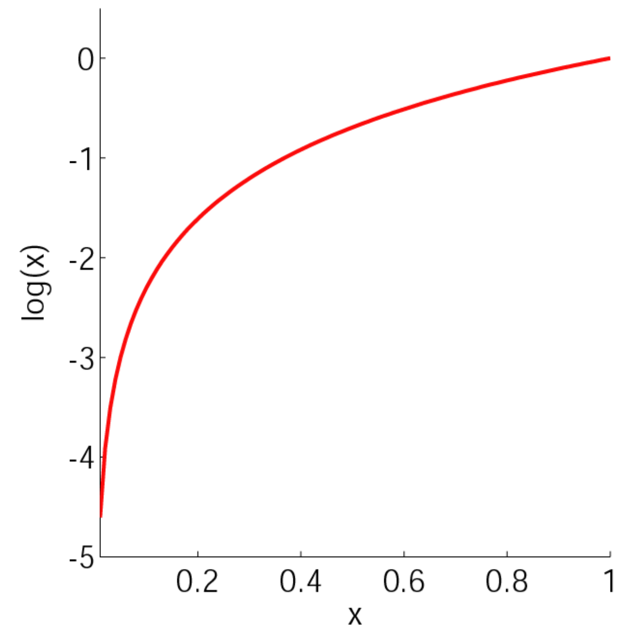
$$\hat{\theta}_{ML} = \mathop{\arg\max}_{\theta \in \Theta} L(\theta)$$

$$L(\theta) = \prod_{i=1}^{n} q(x_i; \theta)$$

- 対数尤度を用いるのが便利なことが多い:

$$\hat{\theta}_{ML} = \mathop{\arg\max}_{\theta \in \Theta} \log L(\theta)$$

$$\log L(\theta) = \sum_{i=1}^{n} \log q(x_i; \theta)$$



数学演習

- ガウスモデル $q(x;\mu,\Sigma)=\dfrac{1}{(2\pi)^{d/2}\det(\Sigma)^{1/2}}\exp\left(-\dfrac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$

  の訓練標本 $\{x_i\}_{i=1}^n$ に対する最尤推定量は

$$\hat{\mu}_{ML}=\frac{1}{n}\sum_{i=1}^n x_i \qquad \hat{\Sigma}_{ML}=\frac{1}{n}\sum_{i=1}^n (x_i-\hat{\mu}_{ML})(x_i-\hat{\mu}_{ML})^T$$

  で与えられることを示せ:

- ヒント:ベクトルや対称行列に関する微分公式

$$\frac{\partial}{\partial\mu}\mu^T\Sigma\mu=2\Sigma\mu \qquad \frac{\partial}{\partial\mu}\mu^T\Sigma x=\Sigma x$$

$$\frac{\partial}{\partial\Sigma}x^T\Sigma^{-1}x=-\Sigma^{-1}xx^T\Sigma^{-1} \qquad \frac{\partial}{\partial\Sigma}\log\det(\Sigma)=\Sigma^{-1}$$

解答例

対数尤度:

$$\log L(\mu,\Sigma) = \sum_{i=1}^{n}\left(-\frac{d}{2}\log 2\pi - \frac{1}{2}\log\det(\Sigma) - \frac{1}{2}(x_i-\mu)^T\Sigma^{-1}(x_i-\mu)\right)$$
$$= -\frac{nd}{2}\log 2\pi - \frac{n}{2}\log\det(\Sigma) - \frac{1}{2}\sum_{i=1}^{n}(x_i-\mu)^T\Sigma^{-1}(x_i-\mu)$$

尤度方程式:

$$\frac{\partial}{\partial\mu}\log L(\mu,\Sigma) = -n\Sigma^{-1}\mu + \Sigma^{-1}\sum_{i=1}^{n}x_i = 0$$
$$\frac{\partial}{\partial\Sigma}\log L(\mu,\Sigma) = -\frac{n}{2}\Sigma^{-1} + \frac{1}{2}\sum_{i=1}^{n}\Sigma^{-1}(x_i-\mu)(x_i-\mu)^T\Sigma^{-1} = 0$$

これを解けば答えが得られる.

事前確率と条件付き確率の推定

- クラス y の期待値ベクトルの最尤推定量:

$$\hat{\mu}_y = \frac{1}{n_y} \sum_{i:y_i=y} x_i$$

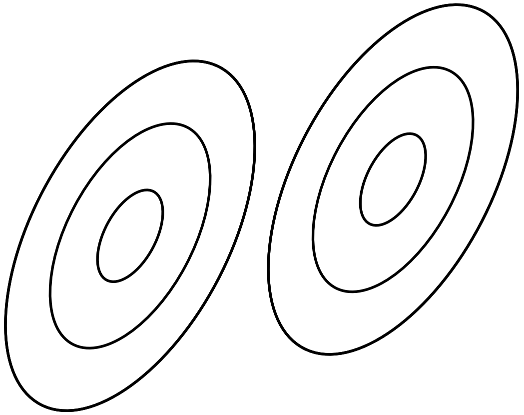
$\sum_{i:y_i=y}$: $y_i = y$ を満たす i に関する和

- 各クラスの分散共分散行列が等しいと仮定したときの共通の分散共分散行列 Σ の最尤推定量は次式で与えられる:

$$\boldsymbol{\Sigma}_+ = \boldsymbol{\Sigma}_- = \boldsymbol{\Sigma}$$

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{y=-1,+1} \sum_{i:y_i=y} (\boldsymbol{x}_i - \widehat{\boldsymbol{\mu}}_y)(\boldsymbol{x}_i - \widehat{\boldsymbol{\mu}}_y)^\top$$

$$= \frac{1}{n}\left(n_+ \widehat{\boldsymbol{\Sigma}}_+ + n_- \widehat{\boldsymbol{\Sigma}}_-\right)$$



フィッシャー判別分析

$$\log p(y|x) = \log p(x|y) + \log p(y) + C$$

■ 対数事後確率：

$$\log \hat{p}(y|x) = \underbrace{-\frac{1}{2}x^T\hat{\Sigma}^{-1}x + \hat{\mu}_y^T\hat{\Sigma}^{-1}x - \frac{1}{2}\hat{\mu}_y^T\hat{\Sigma}^{-1}\hat{\mu}_y - \frac{1}{2}\log\det(\hat{\Sigma})}_{\log p(x|y)} + \underbrace{\log n_y}_{\log p(y)} + C'$$

$$= \hat{\mu}_y^T\hat{\Sigma}^{-1}x - \frac{1}{2}\hat{\mu}_y^T\hat{\Sigma}^{-1}\hat{\mu}_y + \log n_y + C''$$

■ 決定境界 $\hat{p}(y=+1|x) = \hat{p}(y=-1|x)$ は一次形式：

$$a^T x + b = 0 \qquad a = \hat{\Sigma}^{-1}(\hat{\mu}_{+1} - \hat{\mu}_{-1})$$

$$b = -\tfrac{1}{2}(\hat{\mu}_{+1}^T\hat{\Sigma}^{-1}\hat{\mu}_{+1} - \hat{\mu}_{-1}^T\hat{\Sigma}^{-1}\hat{\mu}_{-1}) + \log(n_{+1}/n_{-1})$$

最小二乗分類とフィッシャー判別分析の関係

- 設定:
  - 訓練標本入力の平均がゼロとする:

$$\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i=\boldsymbol{0}$$

  - 入力 $\boldsymbol{x}$ に関する線形モデルを使用:

$$f_{\boldsymbol{\theta}}(\boldsymbol{x})=\boldsymbol{\theta}^\top\boldsymbol{x}$$

最小二乗分類とフィッシャー判別分析の関係

- 前ページの設定のもと，最小二乗分類

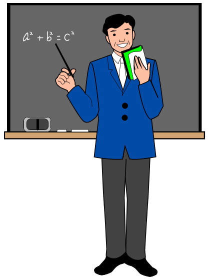
$$\widehat{\boldsymbol{\theta}} = \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}} \frac{1}{2}\sum_{i=1}^{n}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - y_i\Big)^2$$

によって得られる識別境界の向きは

$$\widehat{\boldsymbol{\Sigma}}^{-1}(\widehat{\boldsymbol{\mu}}_+ - \widehat{\boldsymbol{\mu}}_-)$$

- これはフィッシャー判別分析と同じ！

証明は宿題



講義の流れ

1. 最小二乗回帰による分類
2. フィッシャー判別分析
3. 多クラスの分類問題
4. 0/1-損失とマージン

多クラスの分類問題

- 3つ以上のクラスを扱う分類問題
  - 数字認識：10クラス
  - アルファベット認識：26クラス
  - 漢字認識：数千クラス
- 最小二乗分類といった方法はそのままでは多クラス分類に適用できない
- 多クラスの分類問題を複数の2クラス問題に分解する
  - 一対他法，一対一法

一対他法

- 「一つのクラス」と「残り」の２クラス問題に分解
- 最大のスコアを与えるクラスに分類

$$\widehat{y} = \mathop{\mathrm{argmax}}_{y=1,\ldots,c} \widehat{f}_y(\boldsymbol{x})$$

一対一法

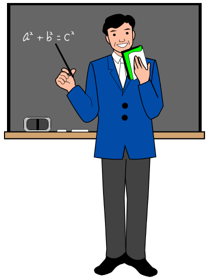
- 「一つのクラス」と「別の一つのクラス」の２クラス問題に分解
- 投票による多数決で分類するクラスを決める

$$\mathrm{sign}\left(\widehat{f}_{y,y'}(\boldsymbol{x})\right) = \begin{cases} +1 & \text{クラス } y \text{ に一票} \\ 0 & \text{投票しない} \\ -1 & \text{クラス } y' \text{ に一票} \end{cases}$$

	クラス1	クラス2	クラス3	$\cdots$	クラスc
クラス1		$\widehat{f}_{1,2}$	$\widehat{f}_{1,3}$	$\cdots$	$\widehat{f}_{1,c}$
クラス2			$\widehat{f}_{2,3}$	$\cdots$	$\widehat{f}_{2,c}$
クラス3				$\cdots$	$\widehat{f}_{3,c}$
$\vdots$					$\vdots$
クラスc					

多クラスの分類問題

- 多クラス分類問題を直接解く
  - 方法によっては不可能なことがある
- 一対他法により2クラス問題に分解
  - c個の２クラス問題を解くだけでよい
  - １個の２クラス問題に全ての訓練標本が含まれる
  - ２クラスの訓練標本数がアンバランス
- 一対一法により2クラス問題に分解
  - c（c＋１）／２個の２クラス問題を解く必要がある
  - １個の２クラス問題の訓練標本数は少ない
  - 投票の仕方に恣意性がある
- どの方法が良いかは状況による



講義の流れ

1. 最小二乗回帰による分類
2. フィッシャー判別分析
3. 多クラスの分類問題
4. 0/1-損失とマージン

0/1-損失関数とマージン

- 分類問題では，学習した関数の符号だけが必要

$$\widehat{y} = \mathrm{sign}\left(f_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{x})\right)$$

- ℓ_2-損失でなく，0/1-損失の方が自然

$$l_{0/1}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i) = \begin{cases} 0 & \text{if } \mathrm{sign}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)\right) = y_i \\ 1 & \text{otherwise} \end{cases}$$

$$= \frac{1 - \mathrm{sign}\left(m_i\right)}{2}$$

マージン

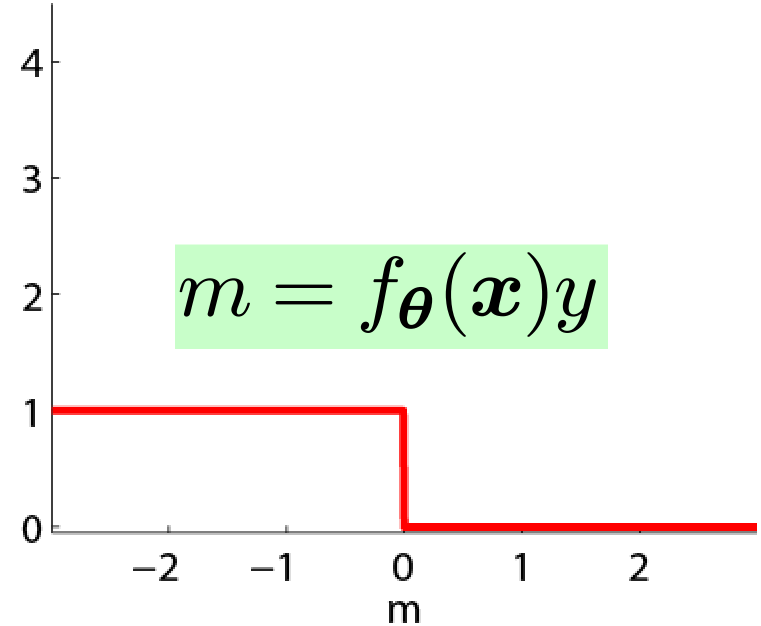
$$m_i = f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) y_i$$

- $J_{0/1}(\boldsymbol{\theta}) = \sum_{i=1}^{n} l_{0/1}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i)$ は誤分類数に相当

0/1-損失関数とマージン

$$J_{0/1}(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^{n}(1 - \mathrm{sign}\,(m_i))$$

- 0/1-損失は誤分類標本数に対応
  - マージンが正なら誤差0
  - マージンが負なら誤差1
- 分類の損失としては理想的
  - しかし傾きを持たない離散的な関数
- 0/1-損失の最小化はNP困難
  - 現実的な時間では不可能



$$m = f_{\boldsymbol{\theta}}(\boldsymbol{x})y$$

数学演習

- ℓ_2-損失関数をマージンを用いて表せ

$$\frac{1}{2}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - y_i\Big)^2 \qquad m_i = f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)y_i$$

- ヒント: $y_i = \pm 1$ を用いる

解答例

$$\frac{1}{2}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - y_i\right)^2 = \frac{1}{2}\left(\frac{m_i}{y_i} - y_i\right)^2$$

$$m_i = f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)y_i$$

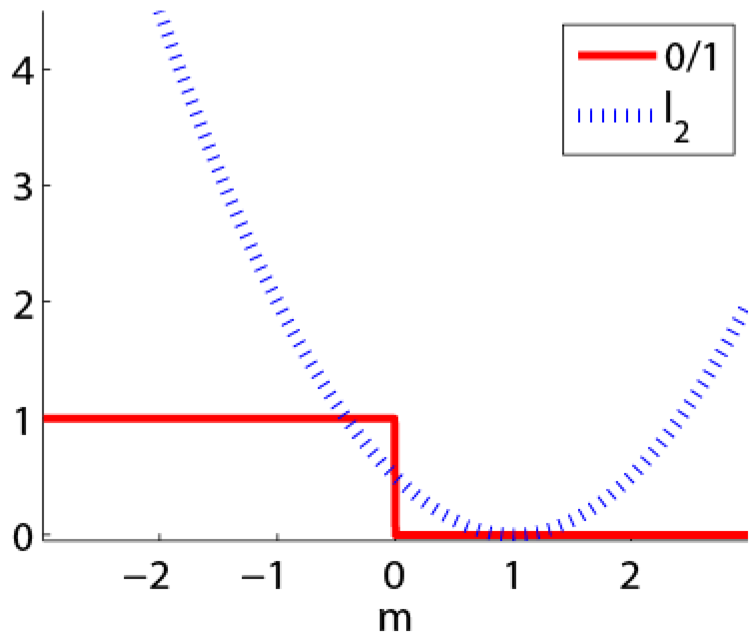
$$= \frac{1}{2}\left(\frac{m_i^2}{y_i^2} - 2m_i + y_i^2\right)$$

$$= \frac{1}{2}\left(m_i^2 - 2m_i + 1\right)$$

$$y_i^2 = 1$$

$$= \frac{1}{2}\left(m_i - 1\right)^2$$

ℓ_2-損失とマージン

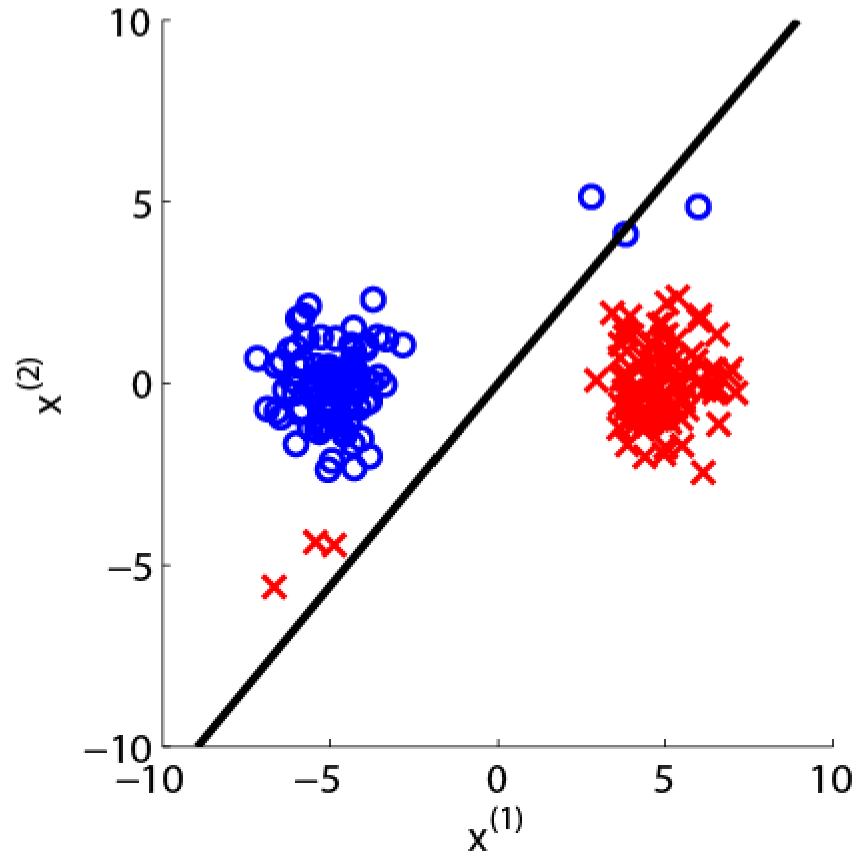


$$m = f_{\boldsymbol{\theta}}(\boldsymbol{x})y$$

- ℓ_2-損失関数は連続関数で扱いやすい
- 負のマージンを正（＋1）にしようとする
- 正の大きいマージンを＋1に減らそうとする

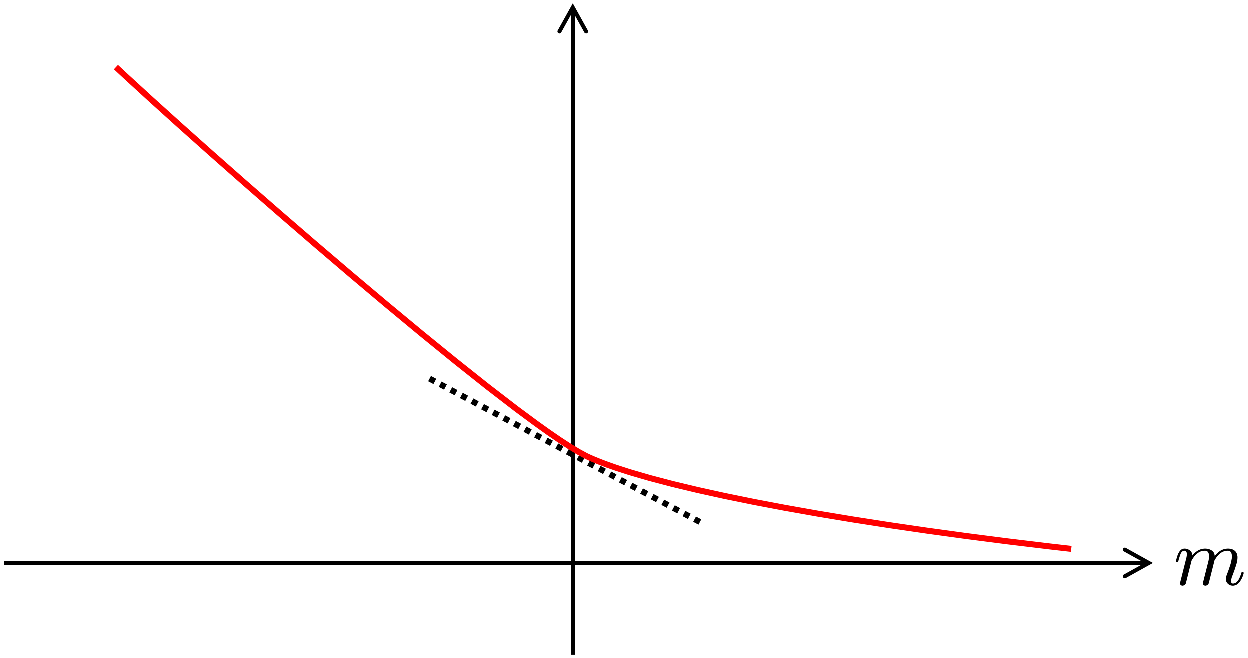
ℓ_2-損失の問題点

- 正のマージンを＋1に減らそうとすることにより，下記のデータを正しく分離できない



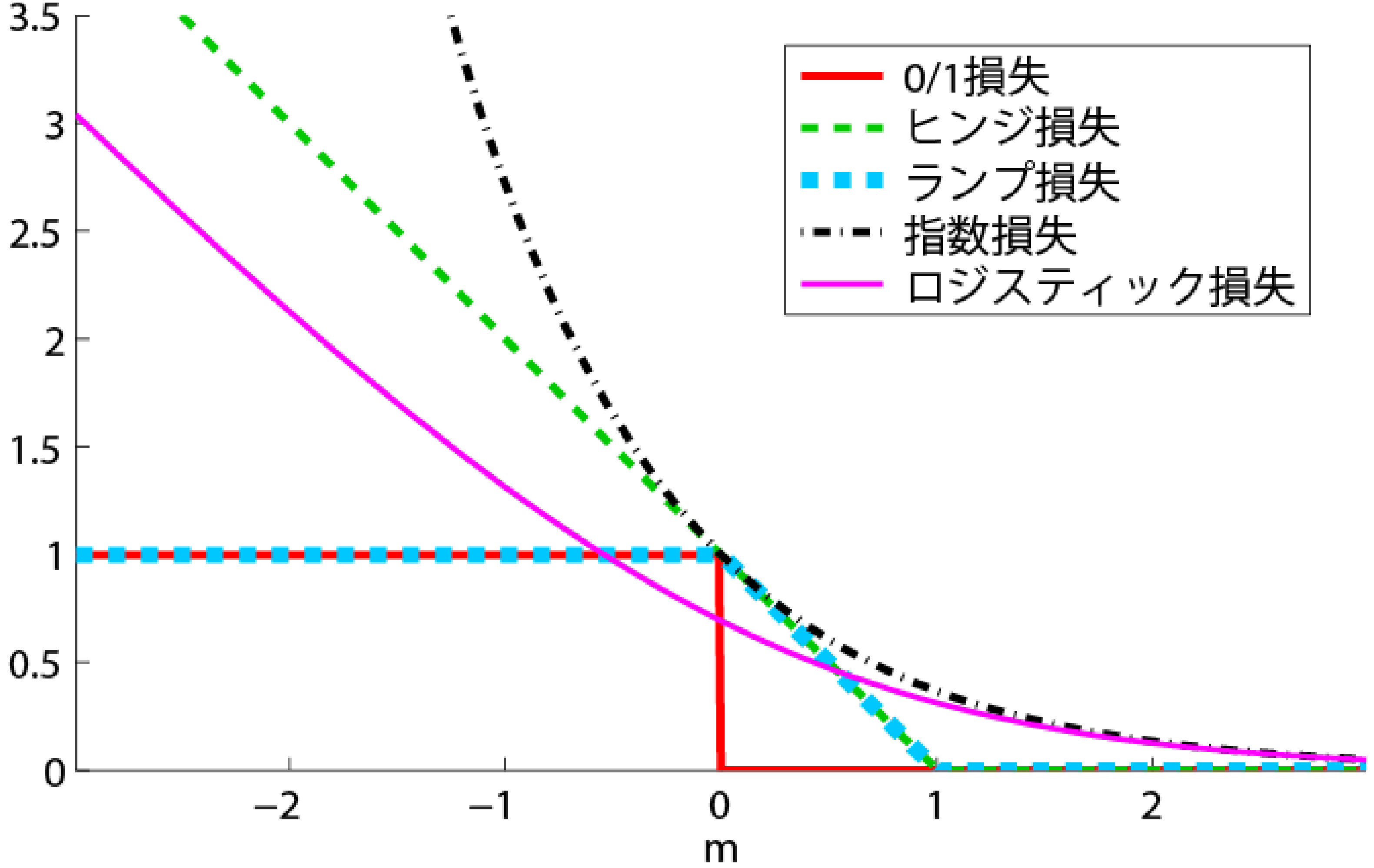
代理損失

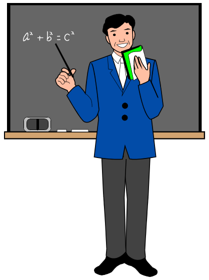
- 0/1-損失の代理として使う損失は，単調非増加で $m=0$ での傾きが負のものがよい
  - 負のマージンを正にしようとする
  - 正のマージンは減らさない



代理損失

- 機械学習では，様々な代理損失が用いられる



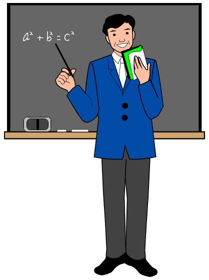


講義の流れ

1. 最小二乗回帰による分類
2. フィッシャー判別分析
3. 多クラスの分類問題
4. 0/1-損失とマージン

まとめ

- 最小二乗回帰によって分類問題が解ける
- 入力に関する線形モデルに対しては, フィッシャー判別分析と同等
- 多クラスの場合は, 一対他か一対一により複数の2クラス問題に分解
- ℓ_2-損失は0/1-損失の代理としてはあまり良い近似ではない



次回の予告

- サポートベクトル分類（８章）

宿題1

- 以下の条件のもと，最小二乗分類

$$\widehat{\boldsymbol{\theta}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \frac{1}{2}\sum_{i=1}^{n}\Big(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - y_i\Big)^2$$

により得られる識別境界の垂線 $\widehat{\boldsymbol{\theta}}$ が，フィッシャー判別分析による境界の垂線 $\widehat{\boldsymbol{\Sigma}}^{-1}(\widehat{\boldsymbol{\mu}}_{+} - \widehat{\boldsymbol{\mu}}_{-})$ とと同じ方向(⇔スカラー倍となる)ことを示せ

- 訓練標本の平均がゼロ: $\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i = \mathbf{0}$
- 入力 x に関する線形モデルを使用: $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{\theta}^\top \boldsymbol{x}$

宿題1のヒント

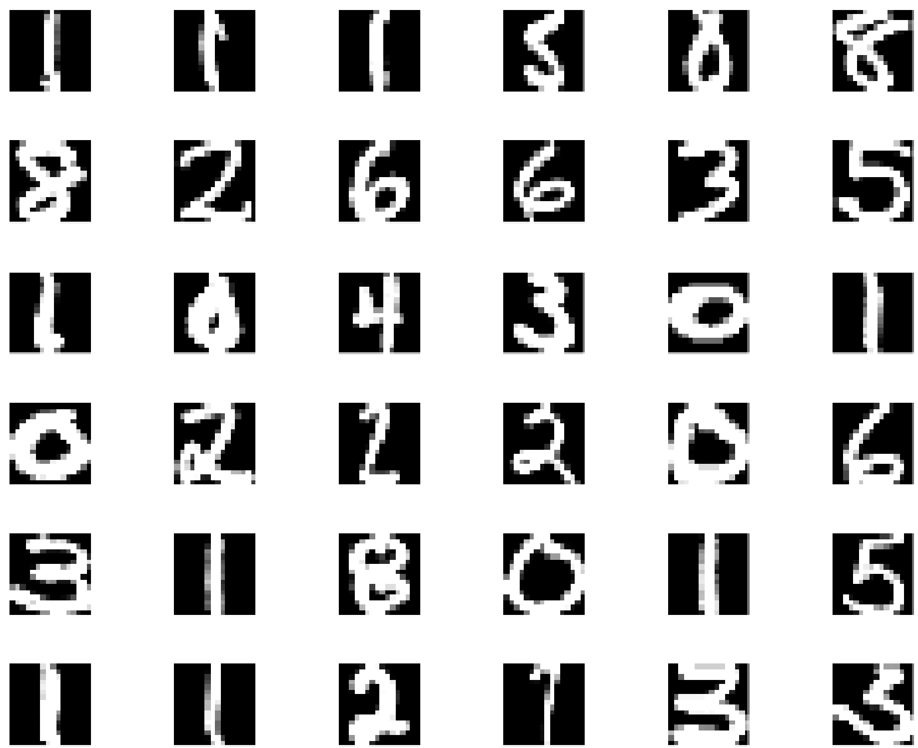
- $(\boldsymbol{X}^\top \boldsymbol{X})\widehat{\boldsymbol{\theta}} = \boldsymbol{X}^\top \boldsymbol{y}$ $\quad \boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^\top$

  $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$

- 以下の手順を考える
  1. 関係 $\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i = \boldsymbol{0}$ を用いて$\boldsymbol{\mu}_-$ と$\boldsymbol{X}^\top \boldsymbol{y}$ を
     それぞれ $\boldsymbol{\mu}_+, n_-, n_+$ を用いて表す
  2. $\boldsymbol{X}^\top \boldsymbol{X}$ を $\widehat{\boldsymbol{\Sigma}}, \boldsymbol{\mu}_+, n_-, n_+$ を用いて表す
  3. 以下の事実を用いる:
     任意のベクトル$\boldsymbol{v}, \boldsymbol{\theta}$に関して, ある定数$c = c_{\boldsymbol{v},\boldsymbol{\theta}}$
     に関して $\boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{\theta} = c \cdot \boldsymbol{v}$と表される

宿題2

- **訓練標本**:0～9まで各500文字ずつ計5000文字
- **テスト標本**:各200文字ずつ計2000文字
- ガウスカーネルに対する最小二乗回帰により，パターン認識を行え



宿題2（続き）

- OctaveやMATLAB以外の言語を使う学生は，

  http://www.ms.k.u-tokyo.ac.jp/software/digit.zip

  から入手できるcsv形式の手書き数字データを用いる
- このzipファイルには，digit_train0.csv, digit_test0.csvなど，訓練用とテスト用の各数字データが格納された20個のファイルが含まれている
- 各ファイルの各行には，数字画像に対応する256個の数値がコンマで区切られて並んでいる
  - -1,-1,-1,-0.99995,-0.99593… ＜ー 1文字目
  - -1,-0.99998,-0.99923,-0.98674,… ＜ー 2文字目

宿題2（続き）

- 実行例：1908／2000＝95．4%の正解率（一対他法）

予測したカテゴリ

正解のカテゴリ	1	2	3	4	5	6	7	8	9	0
1	199	1	0	0	0	0	0	0	0	0
2	0	191	0	6	0	0	2	1	0	0
3	0	0	189	0	5	0	2	4	0	0
4	1	0	0	185	0	4	0	1	9	0
5	0	1	4	2	187	0	0	0	4	2
6	0	2	0	1	1	195	0	0	0	1
7	1	1	0	4	0	0	188	0	6	0
8	1	1	6	1	3	0	0	185	2	1
9	1	0	0	1	0	0	3	2	193	0
0	0	1	0	0	0	3	0	0	0	196