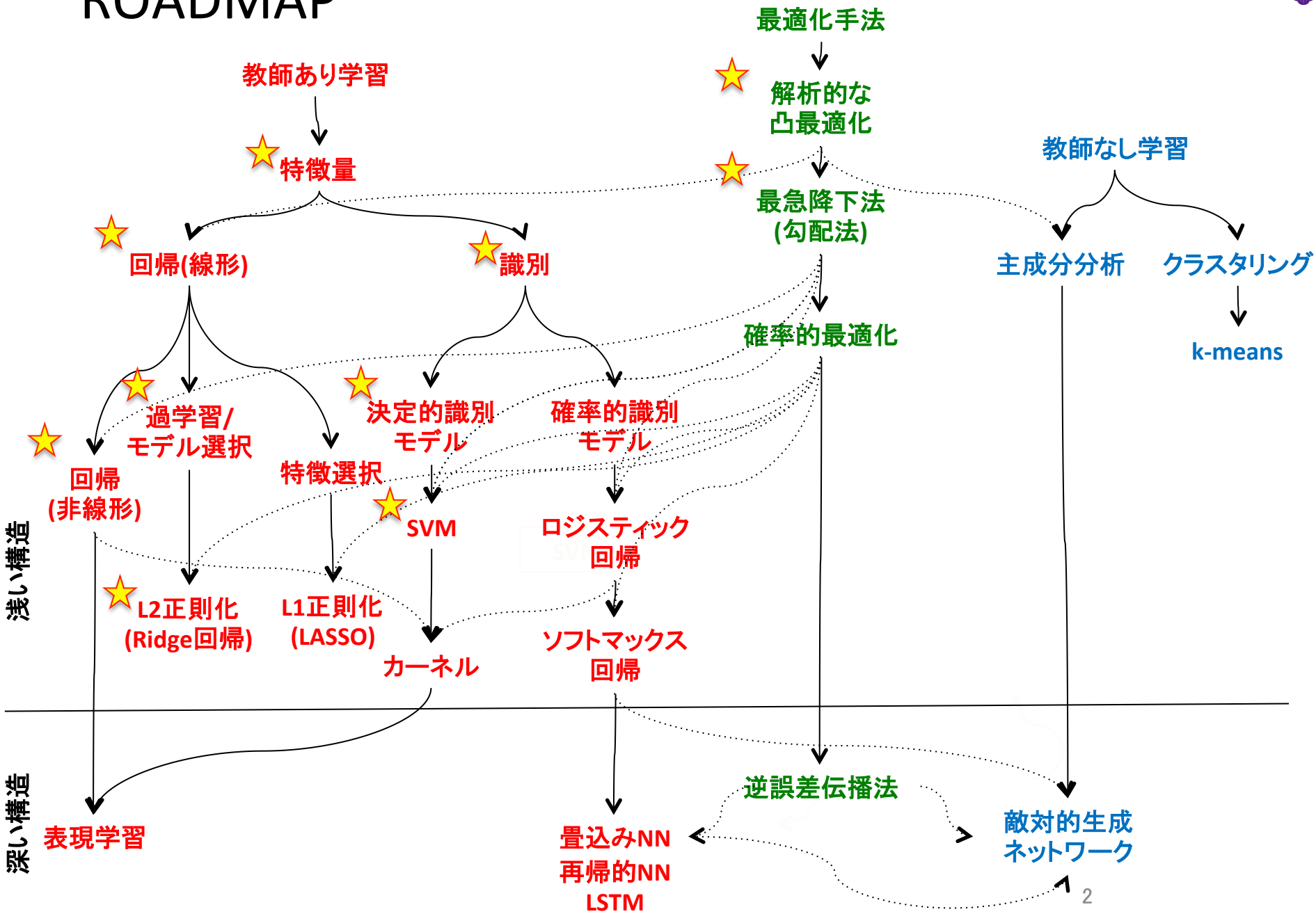
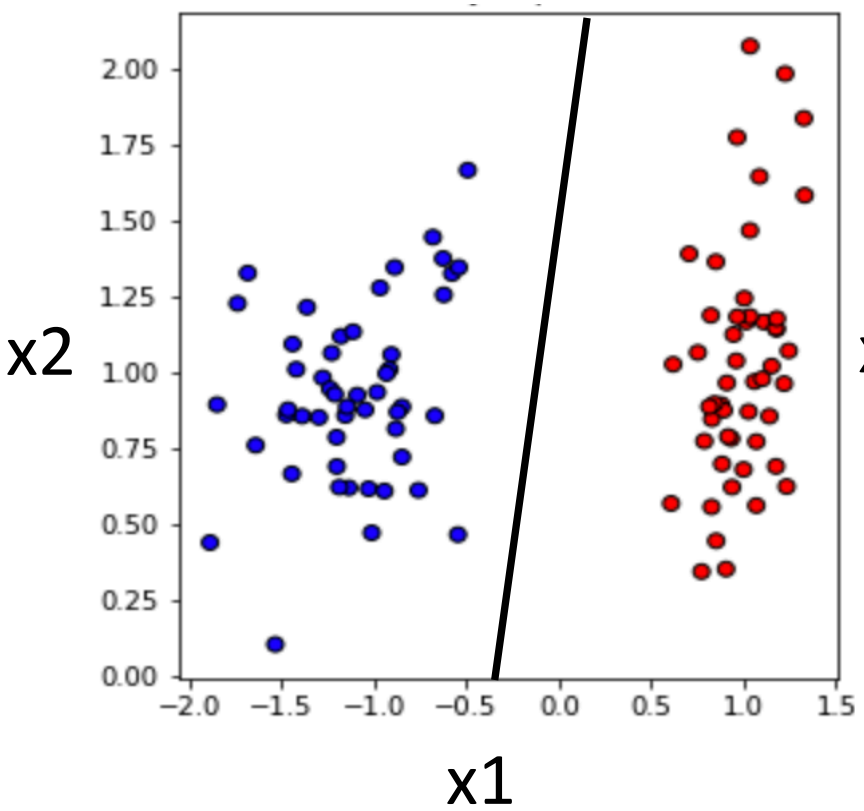
機械学習(6)
カーネル/確率的識別モデル1

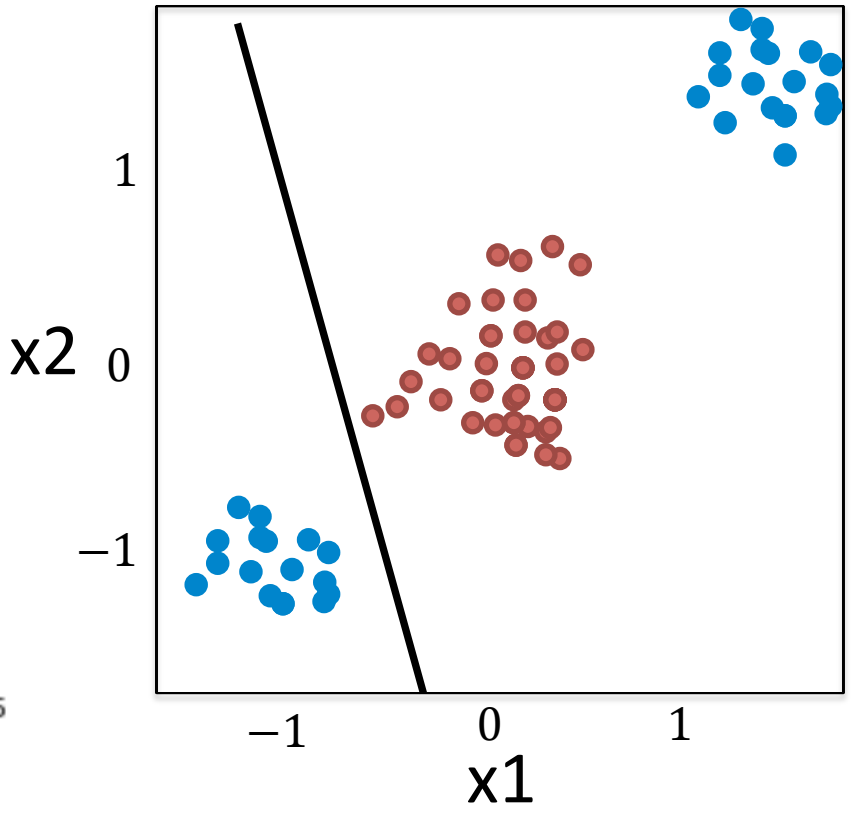
情報科学類　佐久間　淳

ROADMAP

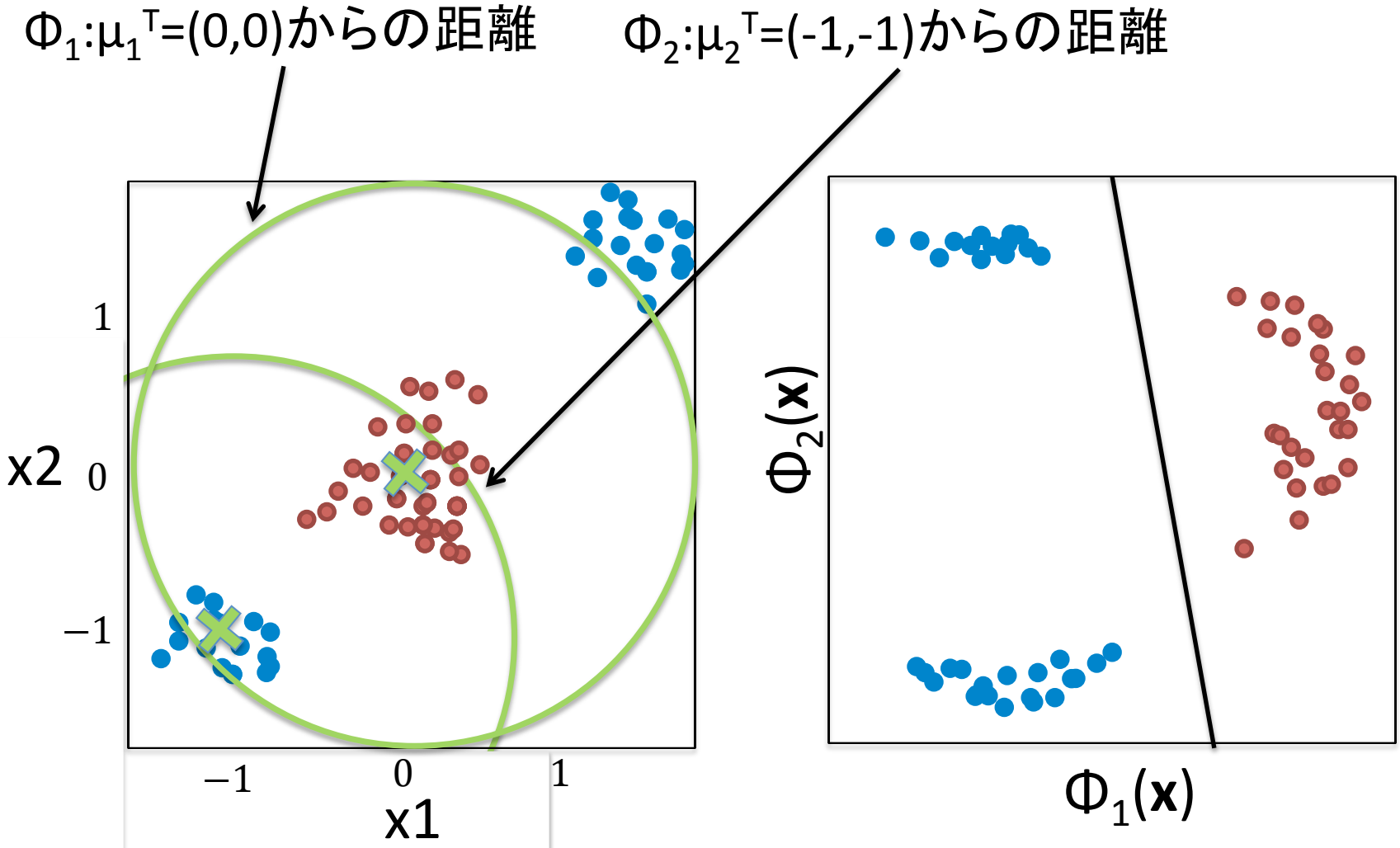

最適化手法
教師あり学習
解析的な
凸最適化
特徴量
教師なし学習
最急降下法
(勾配法)
回帰(線形)
識別
主成分分析
クラスタリング
確率的最適化
k-means
過学習/
モデル選択
決定的識別
モデル
確率的識別
モデル
回帰
(非線形)
特徴選択
SVM
ロジスティック
回帰
浅い構造
L2正則化
(Ridge回帰)
L1正則化
(LASSO)
カーネル
ソフトマックス
回帰
逆誤差伝播法
表現学習
畳込みNN
再帰的NN
LSTM
敵対的生成
ネットワーク
深い構造

線形分類器からカーネルへ

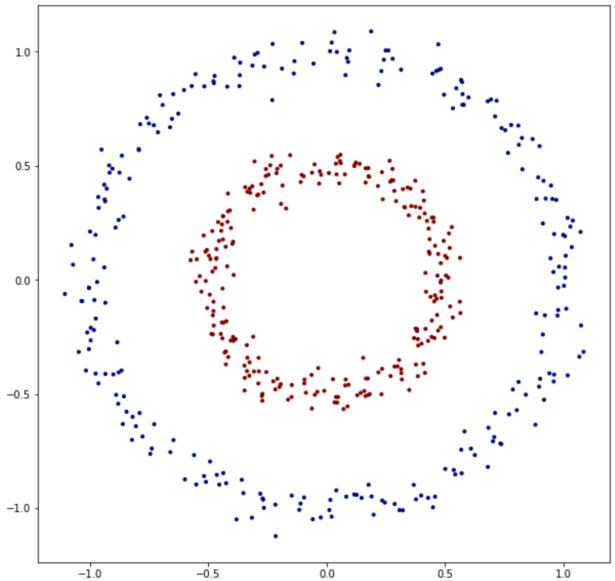




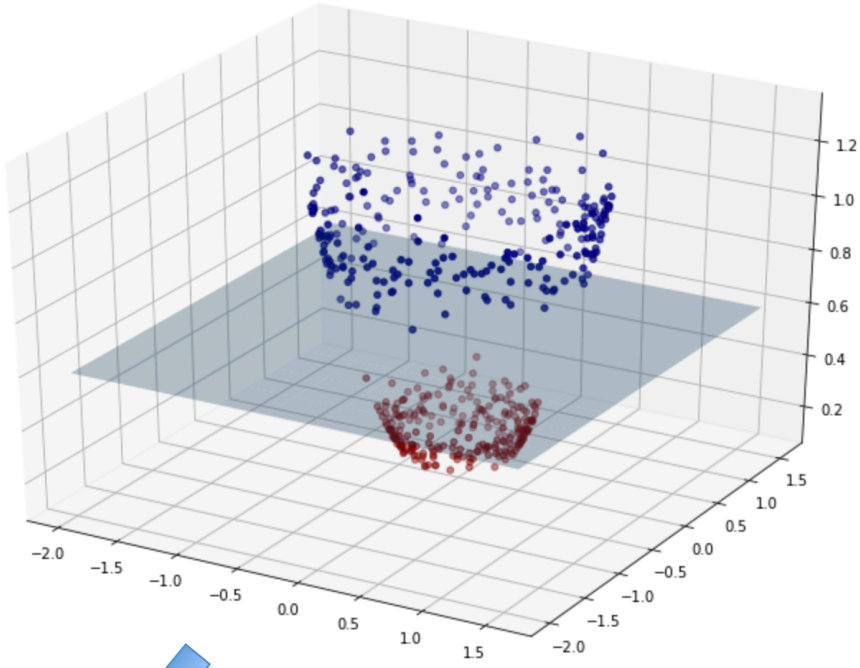
- 青と赤を線形モデル(超平面)で分離したい
- 線形モデルでは限界がある

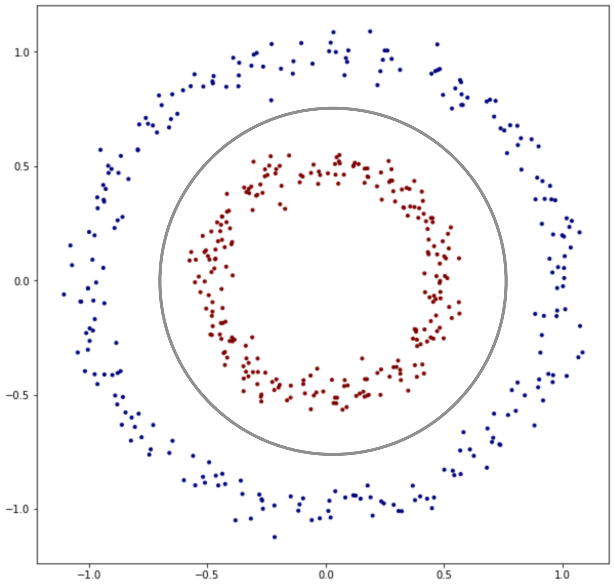


- 青と赤を超平面(線形モデル)で分離したい
  - 左はどんな直線でも線形分離できない
  - 非線形な変換を導入すると線形分離可能



ϕ





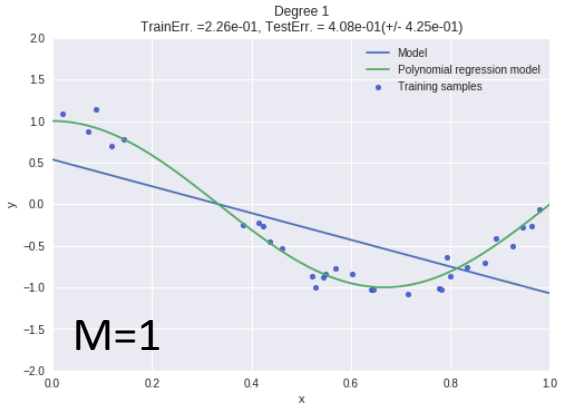
ϕ^{-1}

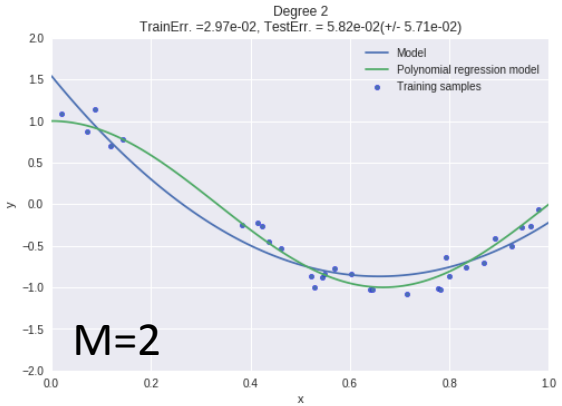
再掲

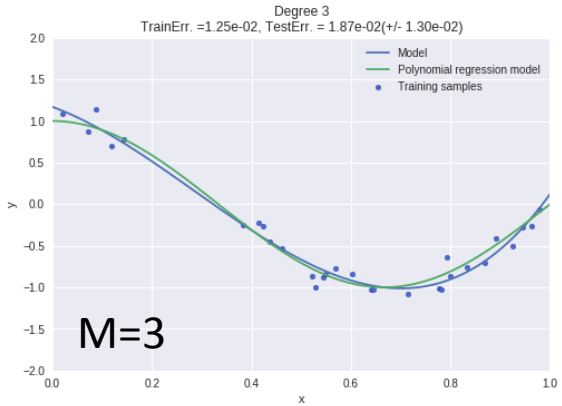
1次元多項式特徴量による回帰

- 特徴量関数　$\boldsymbol{\phi} : \mathbb{R} \to \mathbb{R}^{M+1}$
  - 多項式特徴量 $\boldsymbol{\phi}(x) = (x^0, x^1, x^2, \ldots, x^M)$
- 多項式特徴量による線形回帰モデル

$$t = \boldsymbol{w}^T \boldsymbol{\phi}(x)$$







交互作用項を持つ多項式特徴量

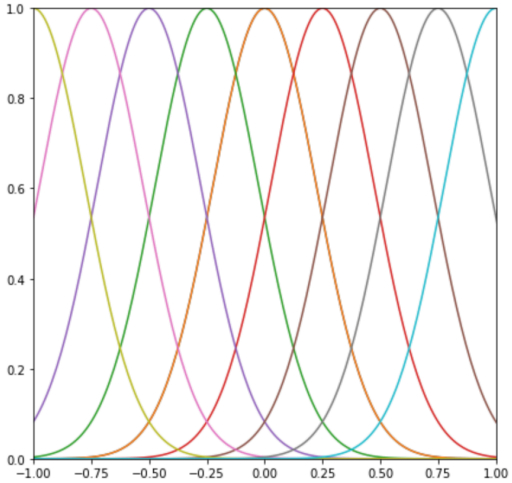
- 2次元2次多項式特徴量
  $\boldsymbol{\phi}(\boldsymbol{x})^T = (1, x_1, x_2, x_1^2, x_2^2, x_1 x_2)$
- 2次元3次多項式特徴量
  $\boldsymbol{\phi}(\boldsymbol{x})^T = (1, x_1, x_2, x_1^2, x_2^2, x_1 x_2, x_1^3, x_2^3, x_1^2 x_2, x_1 x_2^2)$
- 3次元2次多項式特徴量
  $\boldsymbol{\phi}(\boldsymbol{x})^T = (1, x_1, x_2, x_3, x_1^2, x_2^2, x_3^2, x_1 x_2, x_2 x_3, x_1 x_3)$

- 特徴量同士の積(交互作用項)が含まれることに注意

その他の特徴量関数

- ガウスカーネル特徴量

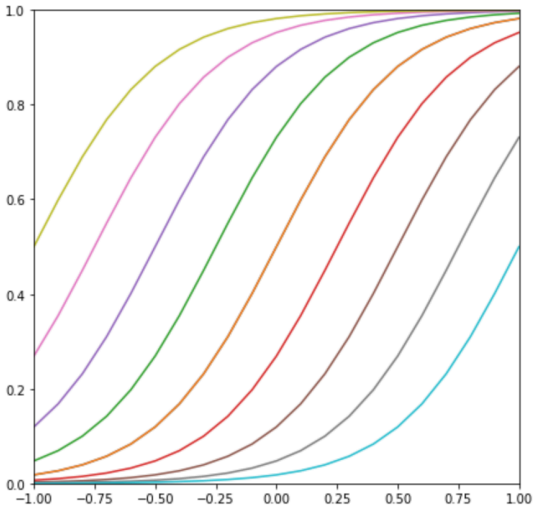
$$\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$$



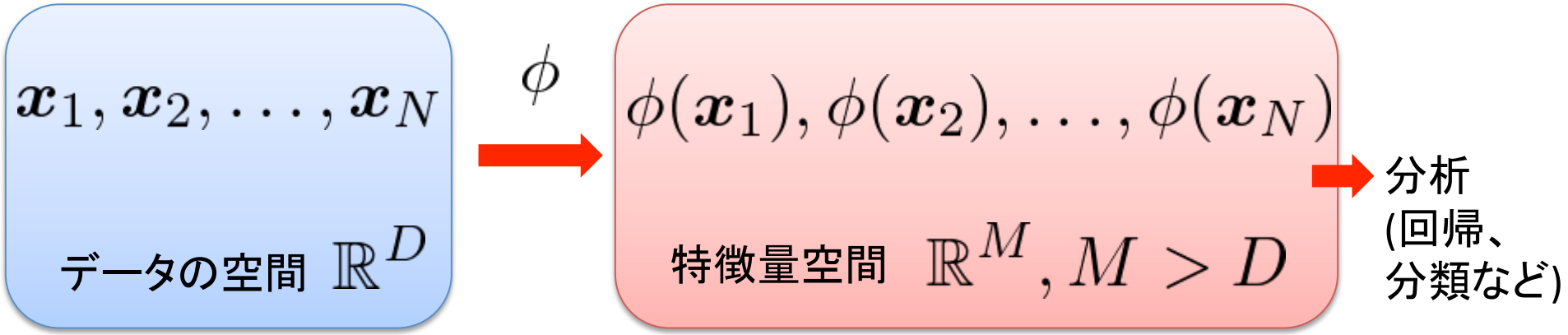
- シグモイド関数

$$\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right),$$

$$\sigma(a) = \frac{1}{1+\exp(-a)}$$



特徴量の考え方



次元は低いが非線形性があって扱いにくい

次元は高いが線形性がある
(と期待され) 扱いやすい

特徴量次元の高さを実感する

- 交互作用項を持つ多項式特徴量
- 2次元ベクトルの2次までの組み合わせ
  $\boldsymbol{\phi}(\boldsymbol{x})^T = (1, x_1, x_2, x_1^2, x_2^2, x_1 x_2)$
- 2次元ベクトルの3次までの組み合わせ
  $\boldsymbol{\phi}(\boldsymbol{x})^T = (1, x_1, x_2, x_1^2, x_2^2, x_1 x_2, x_1^3, x_2^3, x_1^2 x_2, x_1 x_2^2)$
- D次元ベクトルM次多項式だと…?
  - 特徴量次元数が組合せ爆発
- そのような膨大な次元数は扱えない？

多項式カーネル

- 特徴量ベクトル (2次元3次多項式)

$$\phi(\boldsymbol{x})^T = (x_1{}^3, x_2{}^3, 1, \sqrt{3}x_1{}^2x_2, \sqrt{3}x_1x_2{}^2, \sqrt{3}x_2{}^2, \sqrt{3}x_2, \sqrt{3}x_1{}^2, \sqrt{3}x_1, \sqrt{6}x_1x_2),$$
$$\phi(\boldsymbol{x}')^T = (x'_1{}^3, x'_2{}^3, 1, \sqrt{3}x'_1{}^2x'_2, \sqrt{3}x'_1x'_2{}^2, \sqrt{3}x'_2{}^2, \sqrt{3}x'_2, \sqrt{3}x'_1{}^2, \sqrt{3}x'_1, \sqrt{6}x'_1x'_2).$$

- 多項式カーネル
  - 3次多項式カーネル　$k(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}^T\boldsymbol{x}' + 1)^3$
  - M次多項式カーネル　$k(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}^T\boldsymbol{x}' + 1)^M$
- カーネルトリック

$$k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x})^T\boldsymbol{\phi}(\boldsymbol{x}')$$

6.1　多項式カーネル

交互作用項を含む多項式特徴量と多項式カーネルについて以下の問題に答えなさい。

1. 2 次元ベクトルの 3 次多項式特徴量を考える。また 3 次多項式カーネルを $k(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}^T \boldsymbol{x}' + 1)^3$ と定義する。二次多項式特徴量同士の内積は、カーネルを通じて $k(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^T \phi(\boldsymbol{x}')$ と計算できることを示せ。

2. 多項式次元数の増加につれて交互作用項の数は指数的に増加するため, 計算量的な意味で扱いが厄介であるが、予測においては重要な役割を果たすことがある。例えば、spam メール判定ルールとして、x_1:実行可能ファイルが添付ファイルに含まれる (false=0/true=1), x_2:同一メールが K 以上の異なるドメインのメールアドレスに送信されている (false=0/true=1)、という二つの特徴量を用いて、両者が真である場合のみスパムメールと判定する $(x_1 \wedge x_2 = 1)$ によってスパム判定することを考えよう。このような分類は、交互作用項 $x_1 x_2$ を使って自然に表現できる $x_1 x_2 \geq 1$。この例にならって、交互作用項が分類の予測に役立つ特徴量と分類器の例をあげなさい。

カーネルトリック (ガウシアンカーネル) と無限次元特徴量

- ガウシアンカーネル $k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{\sigma^2}\right)$

$$k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\boldsymbol{x}^T\boldsymbol{x}}{\sigma^2}\right)\exp\left(\frac{2\boldsymbol{x}^T\boldsymbol{x}'}{\sigma^2}\right)\exp\left(-\frac{\boldsymbol{x}'^T\boldsymbol{x}'}{\sigma^2}\right)$$

テイラー展開 $\exp(x) = \sum_{n=0}^{\infty}\frac{1}{n!}x^n$ より

$$\exp\left(\frac{2\boldsymbol{x}^T\boldsymbol{x}'}{\sigma^2}\right) = \sum_{k=0}^{\infty}\frac{1}{k!}\left(\frac{2\boldsymbol{x}^T\boldsymbol{x}'}{\sigma^2}\right)^k = \sum_{k=0}^{\infty}\frac{1}{\sqrt{k!}}\left(\frac{\sqrt{2}\boldsymbol{x}^T}{\sigma}\right)^k \cdot \frac{1}{\sqrt{k!}}\left(\frac{\sqrt{2}\boldsymbol{x}'}{\sigma}\right)^k$$

$$= \sum_{k=0}^{\infty}\boldsymbol{\varphi}_k(\boldsymbol{x})^T\boldsymbol{\varphi}_k(\boldsymbol{x}') = \boldsymbol{\psi}(\boldsymbol{x})^T\boldsymbol{\psi}(\boldsymbol{x}')$$

ここで、$\boldsymbol{\varphi}_k(\boldsymbol{x}) = \frac{1}{\sqrt{k!}}\left(\frac{\sqrt{2}\boldsymbol{x}}{\sigma}\right)^k, \boldsymbol{\psi}(\boldsymbol{x})^T = (\boldsymbol{\varphi}_0(\boldsymbol{x})^T, \boldsymbol{\varphi}_1(\boldsymbol{x})^T, \ldots)$
としている。

$$\therefore k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\boldsymbol{x}^T\boldsymbol{x}}{\sigma^2}\right)\boldsymbol{\psi}(\boldsymbol{x})^T\boldsymbol{\psi}(\boldsymbol{x}')\exp\left(-\frac{\boldsymbol{x}'^T\boldsymbol{x}'}{\sigma^2}\right) = \boldsymbol{\phi}(\boldsymbol{x})^T\boldsymbol{\phi}(\boldsymbol{x}')$$

無限次元の特徴量も、内積に限って言えば、有限の時間で計算できる

カーネルをどう活用するか？

- カーネルを使えば、内積だけなら、非常に高い次元(あるいは無限次元)特徴量も(有限の)効率的な時間で扱える
  - 過学習が心配？→正則化すればよい
- もし望む計算が特徴量$\phi(\boldsymbol{x})$ を経由せず、カーネル$k(\boldsymbol{x}, \boldsymbol{x}')$ だけで達成できるなら、特徴量次元数がどんなに高くても困らない!
- そんなことができるか…?

SVMカーネル版

- 特徴量ベクトルによるSVM

等価

目的関数 $$\min_{\boldsymbol{w}} \|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{N} \max\{0, 1 - t_i \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}_i)\}$$

分類器 $$t_i = \mathrm{sgn}[\boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}_i)]$$

- カーネルで表現したSVM

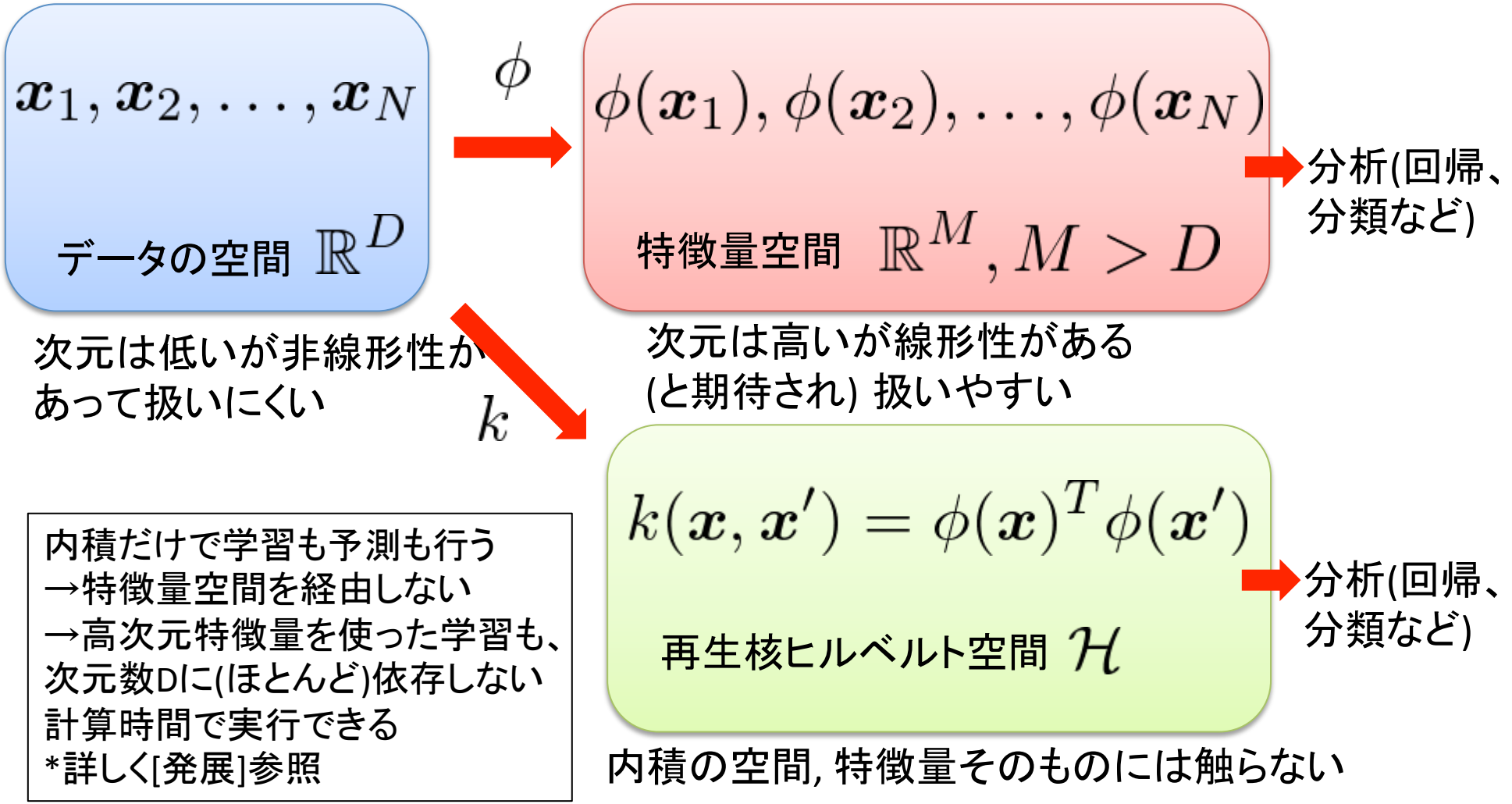
特徴量にアクセスせず、内積(カーネル)だけで目的関数、分類器を表現

目的関数

$$\max_{\alpha_1,\ldots,\alpha_N} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j\in[N]} \alpha_i \alpha_j t_i t_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \text{ sub. to } 0 \leq \alpha_i \leq C, \sum_{i=1}^{N} y_i \alpha_i = 0$$

分類器 $$t_i = \mathrm{sgn}\left[\sum_{i:\alpha_i \neq 0} \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i)\right]$$

カーネルの考え方



内積だけで学習も予測も行う
→特徴量空間を経由しない
→高次元特徴量を使った学習も、次元数Dに(ほとんど)依存しない計算時間で実行できる
*詳しく[発展]参照

演習　様々なカーネル

6.2　カーネルと SVM

svm_with_kernels.ipynb を実行して以下の問いに答えなさい。

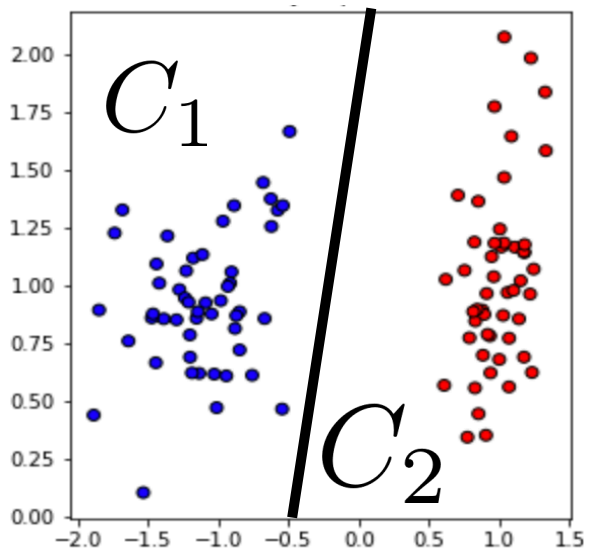
1. ”We first create the data” では、プログラムは 3 種類のデータを生成します。線形モデルでの分類性能が、最も良いと考えられるデータセットはどれか。最も悪いと考えられるデータセットはどれか。

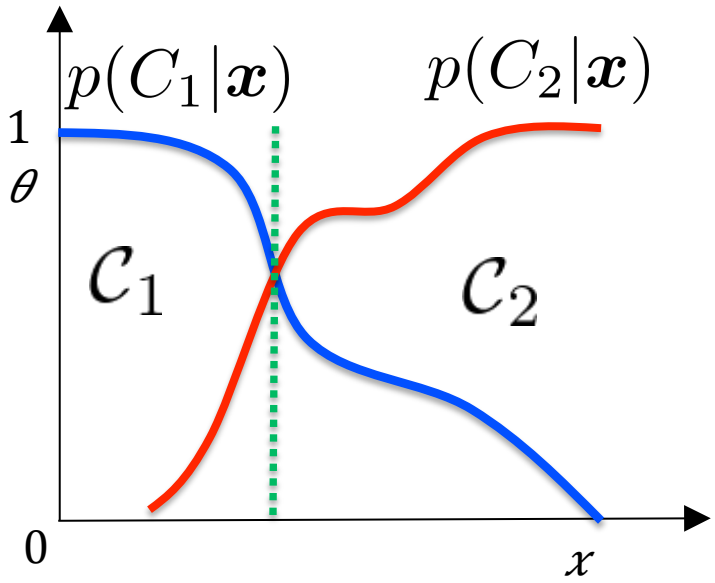
2. ”Linear kernel” では、特徴量やカーネルを用いない線形モデルの SVM を学習する。ここで、パラメータ C は、SVM の定義で現れたパラメータ C である。この C を大きくすると (あるいは小さくすると) どのような挙動が期待できるか？実際にパラメータを変化させたとき、何が起こったか考察しなさい。

3. ”Gaussian kernel” ではガウシアンカーネル (円形基底カーネル, RBF カーネル) における 3 つのデータの分類性能と決定境界を表示する。パラメータ s は、スライドにおけるガウシアンカーネルのパラメータ σ^2 に相当する。s を変化させた時に、分類器の挙動はどのように変化するか 考察しなさい。なお、このプログラムではパラメータ C は [1,30000] の区間で, 1000 刻みで最適な値（最もテスト誤差を小さくする値）が選ばれている。

復習

2つのアプローチ





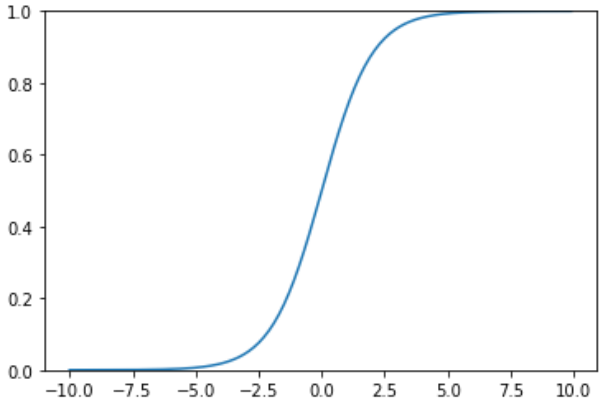
- 決定的識別モデル
  - 識別面 $f(\boldsymbol{x})$ を学習 $f(\boldsymbol{x}) > 0$ ならば $\boldsymbol{x}$ のラベルは C_1
- 確率的識別モデル
  - 条件付き確率 $p(C_1|\boldsymbol{x})$ を学習
  - $p(C_1|\boldsymbol{x}) > 0.5$ ならば $\boldsymbol{x}$ のラベルは C_1

確率的識別モデルの考え方

- 決定的識別モデル 注意：ラベルは(-1,1)
  - 最小マージンを最大にする識別平面(超平面)
  - SVM=ヒンジ損失+L2正則化
- 確率的識別モデル 注意：ラベルは(0,1)
  - ラベル{0,1}を直接予測する代わりに，分類結果=1である確率 $P(t=1|\boldsymbol{x}) = f(\boldsymbol{x})$ を予測
- 懸念：連続値の予測なので回帰が使えるが…
  - 回帰 $f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x}$ は $[-\infty, \infty]$ の値を返す
  - $f(\boldsymbol{x})$ は確率なので $0 \leq f(\boldsymbol{x}) \leq 1$ に入って

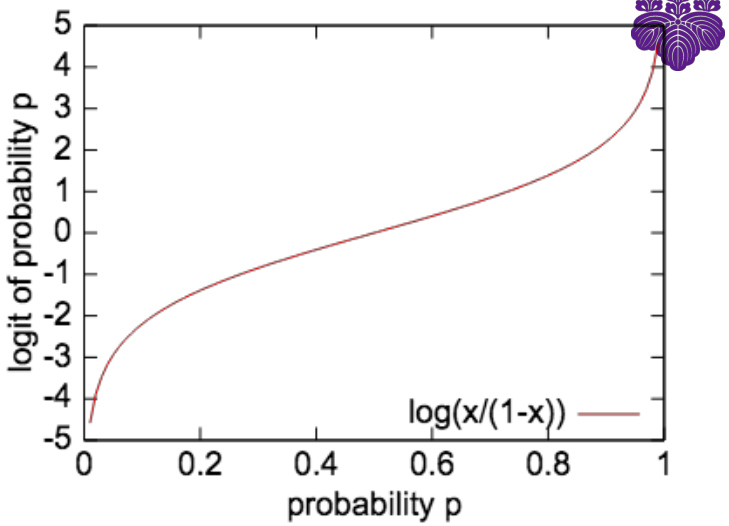
ロジスティックシグモイド関数

- 関数形 $\sigma(a) = \dfrac{1}{1+\exp(-a)}$



  - 全実数を[0,1]に押し込む
  - [-3,3]付近は感度が高い(線形関数に近い)
  - 絶対値が大きな値に対しては感度が低い
  - 神経細胞の挙動を模倣している側面がある
- 性質
  1. $\sigma(-a) = 1 - \sigma(a)$ y=1/2に対して対称
  2. $\dfrac{d}{da}\sigma(a) = \sigma(a)(1-\sigma(a))$ 符号はどうなっている？

ロジスティック回帰



- xのラベルがt=1である確率をxの式でモデル化したい
- 確率の線形回帰によるモデル化
  - $P(t=1|\boldsymbol{x}) = f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x}$
  - 確率の値が無限大に発散しうる
- 確率pのロジット: $\mathrm{logit}(p) = \log \dfrac{p}{1-p}$

確率を実数値Rに写像

$$\mathrm{logit} : [0,1] \mapsto [-\infty, \infty]$$

- ロジスティック回帰:
  - $p_{\boldsymbol{x}} = \Pr(t=1|\boldsymbol{x})$ 書く(簡単のため)
  - 確率の代わりにロジットを線形回帰でモデリング

$$\mathrm{logit}(p_{\boldsymbol{x}}) = \boldsymbol{w}^T\boldsymbol{x} \iff p_{\boldsymbol{x}} = \sigma(\boldsymbol{w}^T\boldsymbol{x})$$

演習　ロジスティックシグモイド関数

6.3　ロジスティックシグモイド関数

ロジスティックシグモイド関数 $\sigma(a) = \frac{1}{1+\exp(-a)}$ について以下の問いに答えよ。

1. $\sigma(-a) = 1 - \sigma(a)$ を示せ.
2. $\frac{d}{da}\sigma(a) = \sigma(a)(1 - \sigma(a))$ を示せ.
3. $a \in \mathbb{R}$ のとき, $\sigma(a) \in [0, 1]$ を示せ.
4. $\log\left(\frac{P(t=1|\boldsymbol{x})}{1-P(t=1|\boldsymbol{x})}\right) = \boldsymbol{w}^T\boldsymbol{x}$ は $P(t = 1|\boldsymbol{x}) = \sigma(\boldsymbol{w}^T\boldsymbol{x})$ と等価であることを示せ.

6.4　ロジスティック回帰による予測

LR_prediction.ipynb を実行して以下の問いに答えなさい。プログラムは 10 点の二次元のラベル付きサンプル (ラベルは 0 or 1) を生成します。生成された特徴とラベルは以下の通り

	1	2	3	4	5	6	7	8	9	10
特徴 x_1	1.5	-0.5	1.0	1.5	0.5	1.5	-0.5	1.0	0.0	0.0
特徴 x_2	-0.5	-1.0	-2.5	-1.0	0.0	-2.0	-0.5	-1.0	-1.0	0.5
ラベル t	1	0	0	1	1	1	0	1	0	0

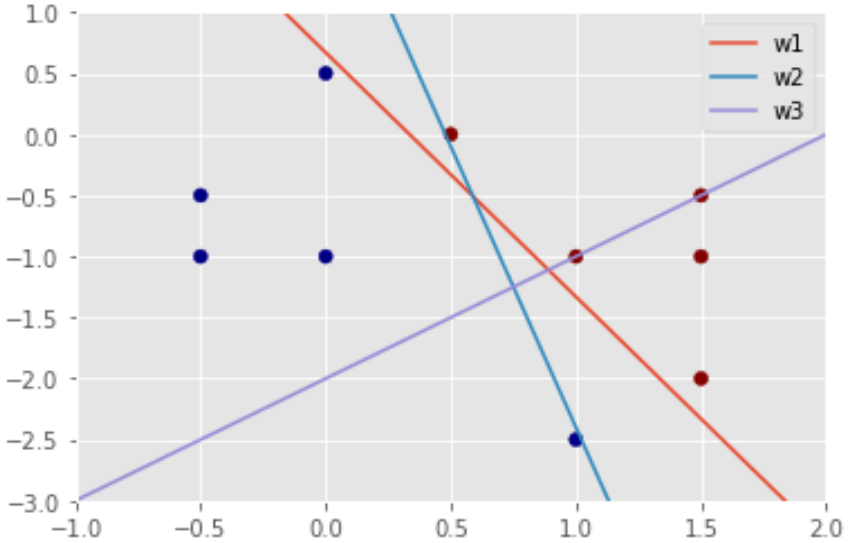
このデータを超平面 $\boldsymbol{w}^T\boldsymbol{x} = 0$ に基づくロジスティック回帰 $y = \sigma(\boldsymbol{w}^T\boldsymbol{x})$ で分類することを考えます。ここでは三つの超平面 $\boldsymbol{w}_1^T = (6, 3, -2)$, $\boldsymbol{w}_2^T = (4.6, 1, -2.2)$, $\boldsymbol{w}_3^T = (1, -1, -2)$ を考えます。エクセルやプログラムなどを用いて、以下の問いに答えなさい。この課題については、結果は manaba からダウンロードしたエクセルシートに結果を記入して提出すること。

(参考) プログラムは三つのモデルに基づくロジスティック回帰で予測された $P(t = 1|\boldsymbol{x})$ の等高線を表示しています。回答の参考にしてください。

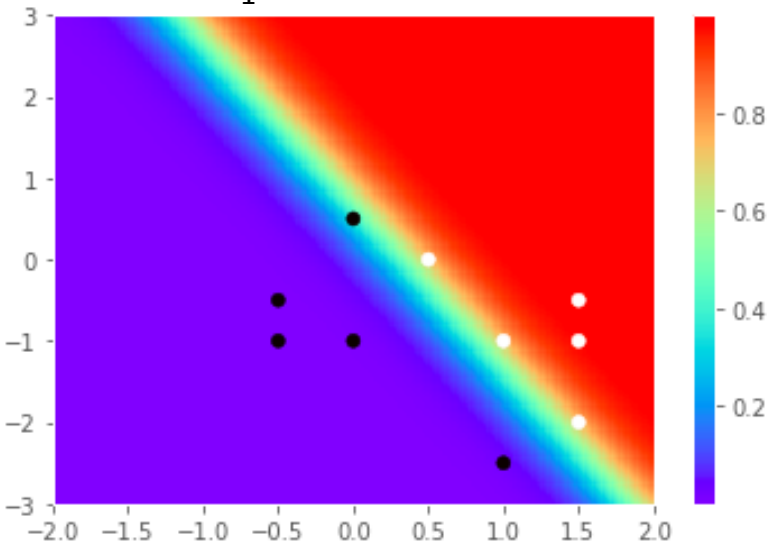
1. $\boldsymbol{w}_i^T\boldsymbol{x}_j$　　$(i = 1, 2, 3, j = 1, \ldots, 10)$ を求めよ
2. $\sigma(\boldsymbol{w}_i^T\boldsymbol{x}_j)$　　$(i = 1, 2, 3, j = 1, \ldots, 10)$ を求めよ
3. モデル $\sigma(\boldsymbol{w}_i^T\boldsymbol{x})$ による $\boldsymbol{x}_j$ の分類結果 (確率ではなく分類結果) を求めよ
4. $\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3$ で実現されるロジスティック回帰モデルの、観測値 $(x_1, \ldots, x_{10})$ に対する正解率を求めよ

演習6.4

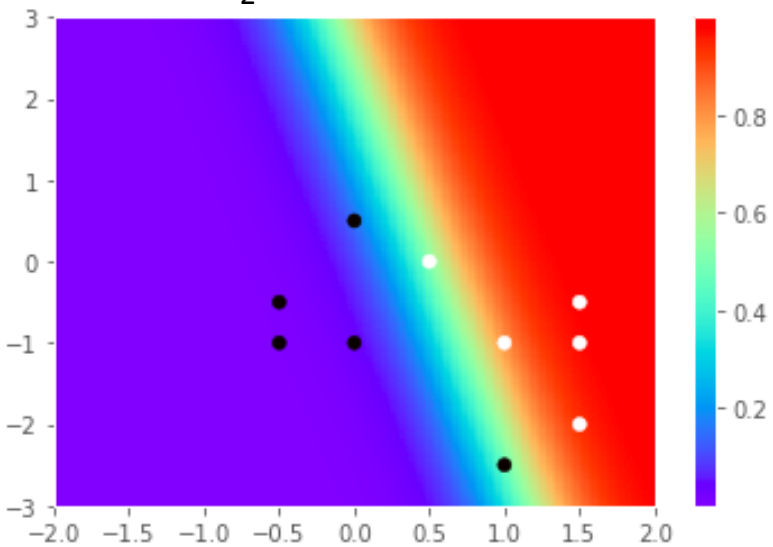
サンプルの分布



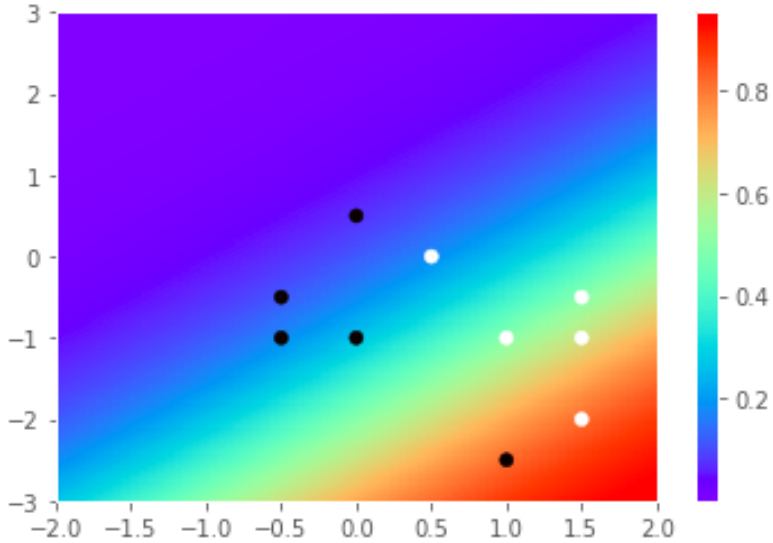
モデルw_1におけるロジット等高線



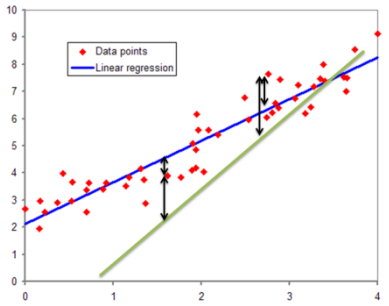
モデルw_2におけるロジット等高線



モデルw_3におけるロジット等高線



ロジスティック回帰の損失



- 回帰の場合は損失として二乗誤差を考えた

正解ラベル

$t_1 = 2.1, t_2 = 4.6, \ldots, t_N = 32.7$

予測

$\boldsymbol{w}^T \boldsymbol{x}_1 = 2.33, \boldsymbol{w}^T \boldsymbol{x}_2 = 4.51, \ldots,$

$\boldsymbol{w}^T \boldsymbol{x}_N = 33.3$

回帰 $t = f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x}$

損失 $E(\boldsymbol{w}) = \sum_{i=1}^{N} (t_i - \boldsymbol{w}^T \boldsymbol{x}_i)^2$

- ロジスティック回帰は損失として何を考えるか?

正解ラベル

$t_1 = 0, t_1 = 1, t_2 = 1, \ldots, t_N = 0$

予測

$\sigma(\boldsymbol{w}^T \boldsymbol{x}_1) = 0.12, \sigma(\boldsymbol{w}^T \boldsymbol{x}_2) = 0.86, \ldots,$

$\sigma(\boldsymbol{w}^T \boldsymbol{x}_N) = 0.56$

分類 $P(\mathcal{C}_1 | \boldsymbol{x}) = \sigma(\boldsymbol{w}^T \boldsymbol{x})$

損失 ???

確率分布(ベルヌーイ分布)

確率μで表(x=1)が出て，確率1-μで裏(x=0)が
出るコインを考える

確率変数　パラメータ

$$\mathrm{B}(x;\mu) = \mu^x(1-\mu)^{1-x} \quad x \in \{0,1\}$$

- 分布Bはパラメータμでパラメトライズされている
- 真のパラメータは神のみぞ知る(コインに固有の値)
- 人間が観測できるのは、有限回のコインの出目
- コインの出目から、真のパラメータμを知るには？

尤度と最尤推定

- 尤度(likelihood)
  - なんらかの前提条件に従って観測値が出現する場合に、逆に観測値から前提条件が「何々であった」と推測する**尤もらしさ**(もっともらしさ)を表す数値
  - E.g. 前提：確率0.3で表が出るコイン、観測値：その出目
- 尤度関数 $\mathrm{L}(\theta) = \prod_{i=1}^{N} p(x_i;\theta)$ ベルヌーイ分布の場合 $L(\mu) = \prod_{i=1}^{N} \mu^{x_i}(1-\mu)^{1-x_i}$
  - 観測データ $(x_1, \ldots, x_N)$ の前提条件(パラメータ)θでパラメトライズされた確率分布 に対する尤度
- 最尤推定 $\max_\theta L(\theta)$
  - 観測データに対して、最大の尤度関数値
- 最尤推定量 $\theta^* = \arg\max_\theta L(\theta)$
  - 尤度を最大にするパラメータ

最尤推定

- 尤度関数を最大にするパラメータ

→最尤推定量

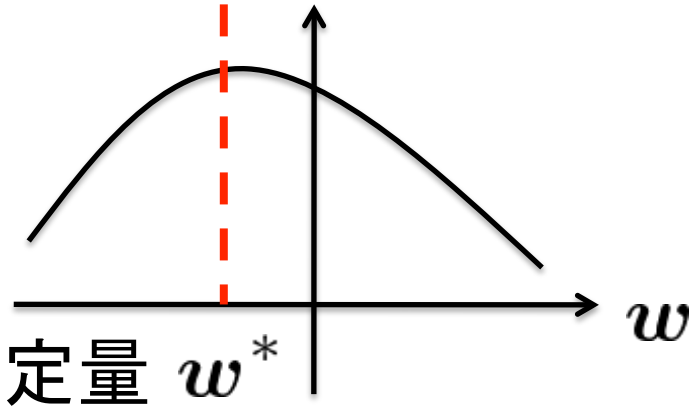
最も尤もらしい推定量

- $f(\boldsymbol{x})$ が凸関数のとき

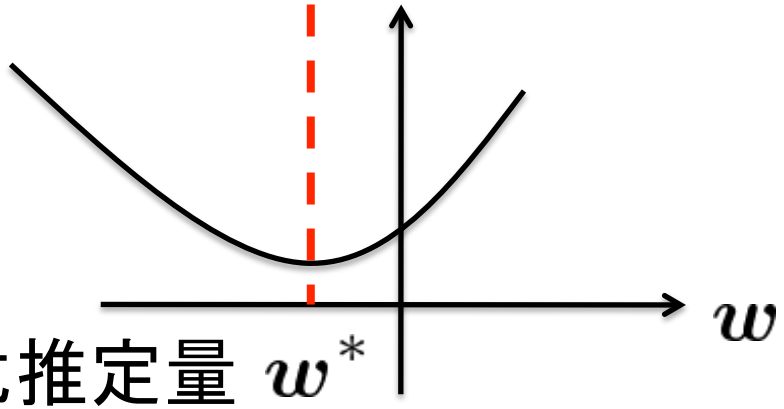
$\log f(\boldsymbol{x})$

も凸関数

- 尤度最大化よりも負の対数尤度最小化のほうが計算に便利なことが多い



最尤推定量 $\boldsymbol{w}^*$



最尤推定量 $\boldsymbol{w}^*$

ベルヌーイ分布の尤度 $L(\mu) = \prod_{i=1}^{N} \mu^{x_i}(1-\mu)^{1-x_i}$

ベルヌーイ分布の対数尤度 $-\log L(\mu) = -\sum_{i=1}^{N} x_i \log \mu + (1-x_i)\log(1-\mu)$

演習 ベルヌーイ分布の尤度

6.5　ベルヌーイ分布の尤度

A 君がコインを 7 回投げたところ, その観測値として (表, 裏, 表, 裏, 裏, 裏, 裏) を得た。以下の問いに答えなさい。

1. このコインの出目を $\mu = 0.25$ および $\mu = 0.1$ のベルヌーイ分布としたときの尤度をそれぞれ求めよ. ただしベルヌーイ分布では表に対応する確率変数の値を 1, 裏に対応する確率変数の値を 0 とする。

2. $\mu = 0.25$ のベルヌーイ分布に従うコインと $\mu = 0.1$ のベルヌーイ分布に従うコイン、この観測値においてはどちらがより尤もらしいか？

3. この観測値に対するパラメータ μ におけるベルヌーイ分布の対数尤度を μ の式で表し、最尤推定量を求めよ

ロジスティック回帰の損失の考え方

- 例えば以下のような状況を考える:

	$(x_1, t_1=1)$	$(x_2, t_2=0)$	$(x_3, t_3=0)$	$(x_4, t_4=1)$
モデル1による予測確率 $\sigma(\boldsymbol{w}_1^T \boldsymbol{x}_i)$	0.9	0.1	0.1	0.9
モデル2による予測確率 $\sigma(\boldsymbol{w}_2^T \boldsymbol{x}_i)$	0.8	0.2	0.2	0.8
実現値(訓練データのラベル) t_i	1	0	0	1

- w_1とw_2, どちらがよりよいモデルと言えるか？
- モデルをwでパラメトライズされたベルヌーイ分布、訓練データ中のラベルを観測値とする
- 尤度を最大にするwを尤も「良い」予測を与えるモデルと考える

ロジスティック回帰の損失

- 損失関数(尤度最大化の枠組みで)
  - ロジスティック回帰が出力する予測 $\hat{t}_n = \sigma(\boldsymbol{w}^T \boldsymbol{x}_n)$
  - 実現値 t_n
  - ロジスティック回帰の予測をベルヌーイ分布とみなした時の実現値(訓練データ)の尤度

$$L(\boldsymbol{w}) = \prod_{n=1}^{N} \hat{t}_n^{t_n} (1 - \hat{t}_n)^{(1-t_n)}$$

L(w)はロジスティック回帰 $\hat{t}_n = \sigma(\boldsymbol{w}^T \boldsymbol{x}_n)$ が訓練データに対してどの程度尤もらしいかを評価

ロジスティック回帰の尤度関数

$$L(\boldsymbol{w}) = \prod_{n=1}^{N} \hat{t}_n^{t_n}(1-\hat{t}_n)^{(1-t_n)}$$

	$\hat{t}_n^{t_n}$	$(1-\hat{t}_n)^{(1-t_n)}$	$\hat{t}_n^{t_n}(1-\hat{t}_n)^{(1-t_n)}$
$t_n = 1, \hat{t}_n \simeq 1$	(1に近い値)^1	(0に近い値)^0	1に近い値
$t_n = 1, \hat{t}_n \simeq 0$	(0に近い値)^1	(1に近い値)^0	0に近い値
$t_n = 0, \hat{t}_n \simeq 0$	(0に近い値)^0	(1に近い値)^1	1に近い値
$t_n = 0, \hat{t}_n \simeq 1$	(1に近い値)^0	(0に近い値)^1	0に近い値

予測が当たってるほうが尤度は確かに大きくなる

交差エントロピー誤差関数の最小化

- 尤度関数最大化の代わりに負の対数尤度関数(交差エントロピー損失と呼ばれる)を最小化

$$E(\boldsymbol{w}) = -\ln L(\boldsymbol{w}) = -\sum_{n=1}^{N} \{t_n \ln \hat{t}_n + (1-t_n)\ln(1-\hat{t}_n)\}$$

$$\hat{t}_n = \sigma(\boldsymbol{w}^T \boldsymbol{x}_n)$$

- 凸関数(証明は略)
  - 微分できれば最小化できる
  - でもシグモイド関数が挟まっているので解析的な解法(微分をゼロにするwを求める)ことはできない
  - どうやって最小化する?

演習　ロジスティック回帰と対数尤度

6.6　ロジスティック回帰と対数尤度

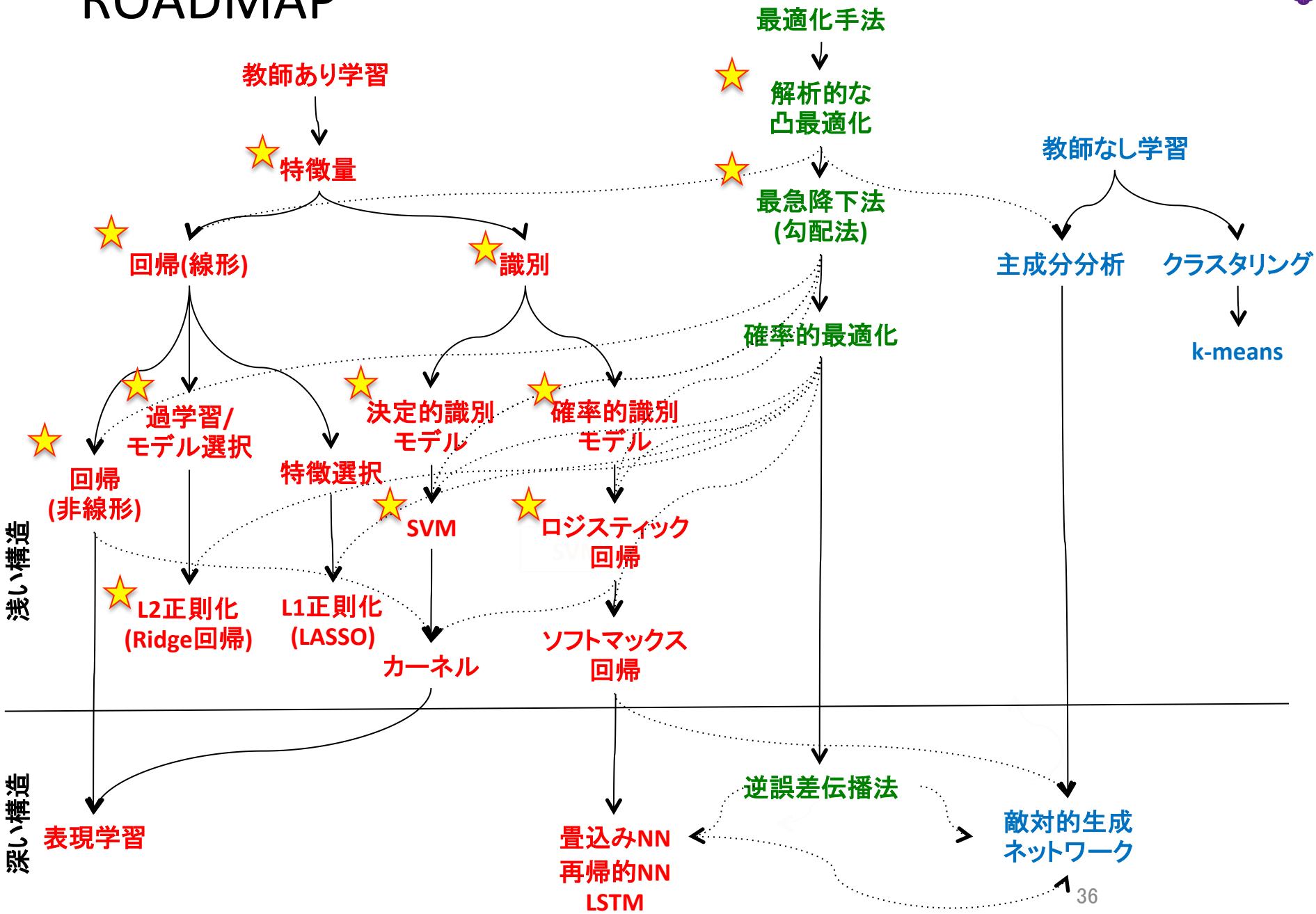
LR_prediction.ipynb を実行して以下の問いに答えなさい。用いるデータやモデルは 6.4 と同じである。

1. 訓練サンプル $(x_1, \ldots, x_{10})$ について交差エントロピー損失 $E(\boldsymbol{w}_1), E(\boldsymbol{w}_2), E(\boldsymbol{w}_3)$ を求めなさい
2. 最も尤度の高いモデルはどれか
3. $\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3$ どれが分類モデルとして適切か, 理由とともに答えよ

まとめ

- 確率的識別モデルでは、データがクラス1に属する確率をシグモイドロジスティック関数 $P(C_1|x)=\sigma(w^Tx)$を用いてモデル化する
- 誤差最小化の代わりに対数尤度最大化を行う

ROADMAP


最適化手法
教師あり学習
解析的な
凸最適化
特徴量
教師なし学習
最急降下法
(勾配法)
回帰(線形)
識別
主成分分析
クラスタリング
確率的最適化
k-means
過学習/
モデル選択
決定的識別
モデル
確率的識別
モデル
回帰
(非線形)
特徴選択
SVM
ロジスティック
回帰
浅い構造
L2正則化
(Ridge回帰)
L1正則化
(LASSO)
カーネル
ソフトマックス
回帰
逆誤差伝播法
表現学習
畳込みNN
再帰的NN
LSTM
敵対的生成
ネットワーク
深い構造