機械学習 第６回 回帰

立命館大学 情報理工学部

福森 隆寛

Beyond Borders

講義スケジュール

（第1～4回、第14回）　（第5～13回、第15回）

- 担当教員：村上 陽平先生・福森 隆寛

1	機械学習とは、機械学習の分類		9	深層学習
2	機械学習の基本的な手順		10	アンサンブル学習
3	識別（1）		11	モデル推定
4	識別（2）		12	パターンマイニング
5	識別（3）		13	系列データの識別
6	**回帰**		14	強化学習
7	サポートベクトルマシン		15	半教師あり学習
8	ニューラルネットワーク			

- 担当教員：叶 昕辰先生（第16回の講義を担当）

今回の講義内容

- 取り扱う問題の定義
- 線形回帰
- 回帰モデルの評価
- 正則化（せいそくか）
- バイアスと分散のトレードオフ
- 回帰木（かいきぎ）
- 演習問題

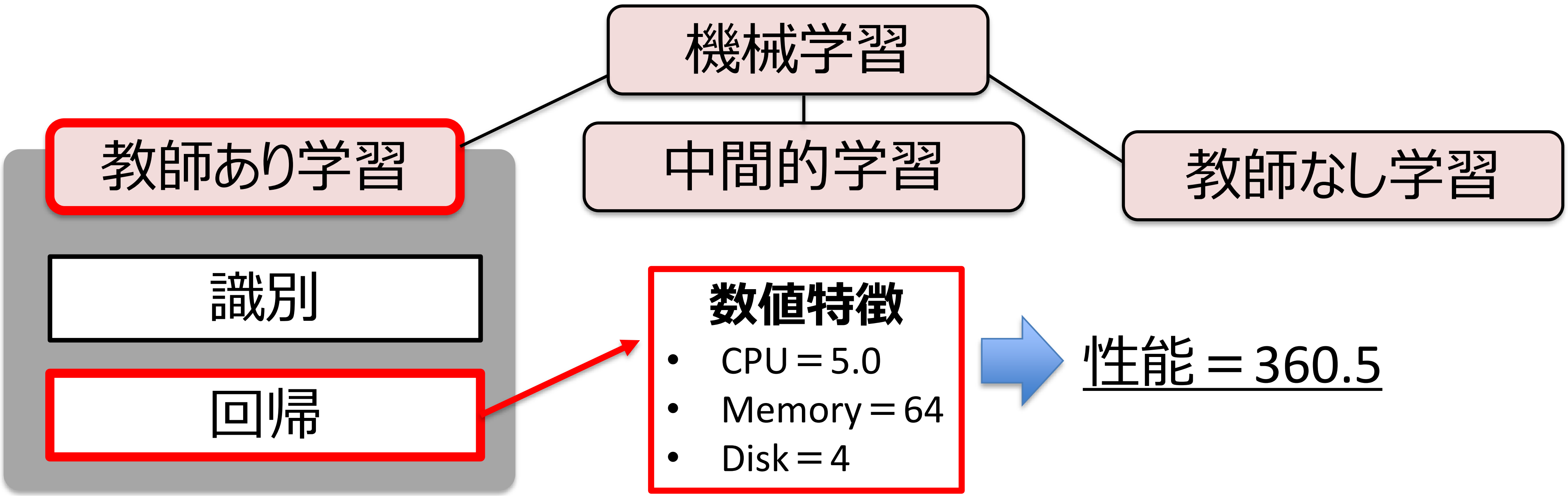
取り扱う問題の定義：教師あり・回帰

- **数値**データからなる特徴ベクトルを入力して、**数値**を出力する関数を作る

※ 教師あり学習の回帰問題での学習データは、以下のペアで構成される

入力データの特徴ベクトル（数値データ） ← $\{\boldsymbol{x}_i, y_i\}, \quad i = 1,2,\dots,N$ → 学習データの総数

数値形式の正解情報→「ターゲット」と呼ぶ



取り扱う問題の定義：教師あり・回帰

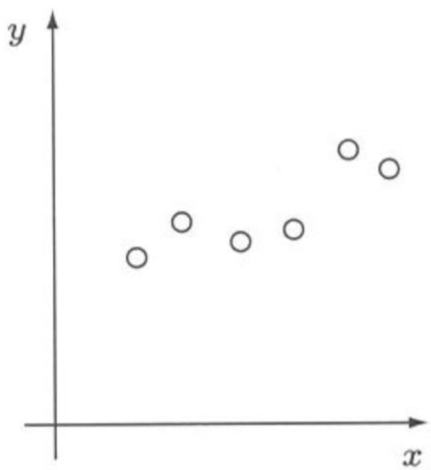
- 識別と回帰の境界は、それほど明確(めいかく)ではない
  - 識別：数値特徴を入力として**クラス**を出力
  - 回帰：数値特徴を入力として**数値**を出力

- クラスによって異なる値をとるクラス変数を導入(どうにゅう)し
  入力からクラス変数の値を予測する問題を考えると
  識別問題を回帰問題として考えることもできる

線形回帰

- 最も単純な入力も出力もスカラーである場合の回帰問題を考える
  - 学習データから入力xを出力yに写像する関数$\hat{c}(x)$を推定

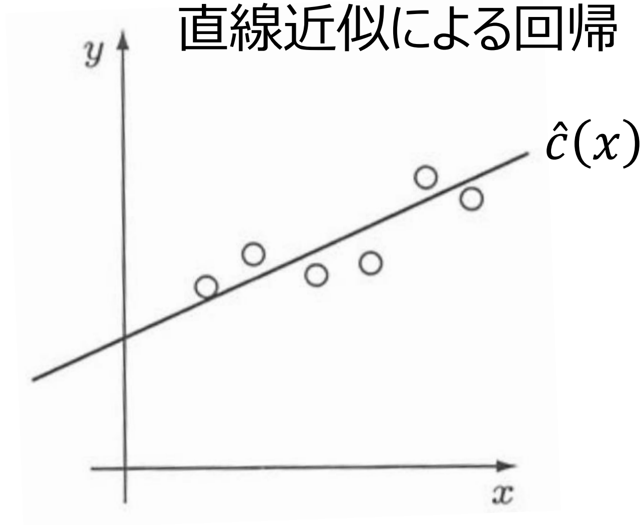
学習データ



入力xが大きくなると
出力yも大きい値になる
傾向がみられる

この傾向を直線で表して
入力xと出力yを関係づける

直線近似による回帰



最小二乗法と同様の方法で
なるべく誤差の少ない直線を求める

線形回帰

- 最小二乗法から回帰式を求める
  - 回帰式を $\hat{c}(x) = w_1 x + w_0$ とする
  - 誤差の二乗和は

$$E(\boldsymbol{w}) = \sum_{i=1}^{N} \{y_i - \hat{c}(x_i)\}^2 = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})$$

    - $\boldsymbol{X}$：1列目の全要素が1、2列目i行の要素がx_iのパターン行列
    - $\boldsymbol{w}$：重みベクトル ※ $\boldsymbol{w} = (w_0, w_1)^T$
  - $\boldsymbol{w}$で微分したものを0とすると、線形回帰式の重みは下式で計算できる

$$\boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{y}$$

  - 入力$\boldsymbol{x}$がd次元の場合も、同様の方法で計算できる

回帰モデルの評価

- 回帰式が未知データに対して正しく出力値を予測するかを評価

- 評価指標
  - 相関係数 R：正解と予測が、どの程度似ているのか
  - 決定係数 R^2
    - 「正解との離れ具合」と「平均との離れ具合」の比を1から引く
    - $\tilde{y}$： y_iの平均値

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\{y_i - \hat{c}(x_i)\}^2}{\sum_{i=1}^{N}(y_i - \tilde{y})^2}$$

式変形により相関係数の二乗と一致するのでR^2と表記する

演習問題6-1（10分間）

- 右表のような身長と体重のデータが与えられた
- 身長xから体重$\hat{c}(x)$を予測する線形回帰式が
  $$\hat{c}(x) = 0.625x - 48.604$$
  であるときの決定係数と相関係数を計算せよ

身長と体重データ

番号	身長 [cm]	体重 [kg]
1	147.9	41.7
2	163.5	60.2
3	159.8	47.0
4	155.1	53.2
5	163.3	48.3
6	158.7	55.2
7	172.0	58.5
8	161.2	49.0
9	153.9	46.7
10	161.6	52.5

正則化

望ましい線形回帰式

- 汎化能力という点では、入力が**少し**変化したときに、出力も**少し**変化する回帰式が良い
  - 重みが大きいと、入力が**少し**変化するだけで出力が**大きく**変化
  - そのような回帰式は、たまたま学習データの近くを通っても、未知データに対する出力は信用(しんよう)できない
- 線形回帰の重みは、値が0となる次元を多くすれば良い
  - 回帰式の係数wに関して、「大きな値の重みが、なるべく少なくなる」あるいは「0となる重みが多くなる」ような方法が必要
- このような工夫(くふう)が**正則化**
  - 誤差関数の式に正則化項を追加(ついか)する

正則化：Ridge回帰

□ Ridge回帰

- ■ パラメータ$\boldsymbol{w}$の二乗を正則化項とする
- ■ パラメータの値が小さくなるように正則化させる

$$E(\boldsymbol{w}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}) + \lambda \underbrace{\boldsymbol{w}^T\boldsymbol{w}}_{\text{正則化項}}$$

λ：正則化項の重み（重みが大きければ、性能よりも正則化の結果を重視）

- ■ 最小二乗法でパラメータを求めたときと同様に、
$\boldsymbol{w}$で微分した値が0となる$\boldsymbol{w}$の値を求めると...

$$\boldsymbol{w} = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

※$\boldsymbol{I}$：単位（たんい）行列

正則化：Lasso回帰

□ Lasso回帰

- ■ パラメータ$\boldsymbol{w}$の絶対値を正則化項とする
- ■ 値を0とするパラメータが多くなるように正則化される

$$E(\boldsymbol{w}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}) + \lambda \sum_{j=1}^{d} |w_j|$$

λ：正則化項の重み（大きければ、値を0とする重みが多くなる）
w_0：回帰式の切片（せっぺん）は汎化能力に影響（えいきょう）なし（通常は正則化の対象としない）

- ■ Lasso回帰の解は、解析的に求められない
  - 原点で微分不可能な絶対値を含むため
  - 正則化項の上限（じょうげん）を微分可能な2次関数で押（お）さえ、その関数のパラメータを誤差が小さくなるように逐次（ちくじ）更新する方法が提案（ていあん）

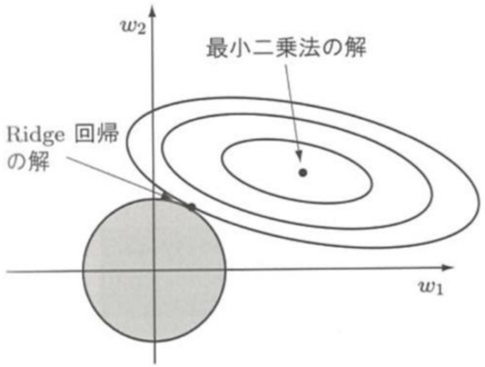
正則化：正則化の振る舞い

Ridge回帰

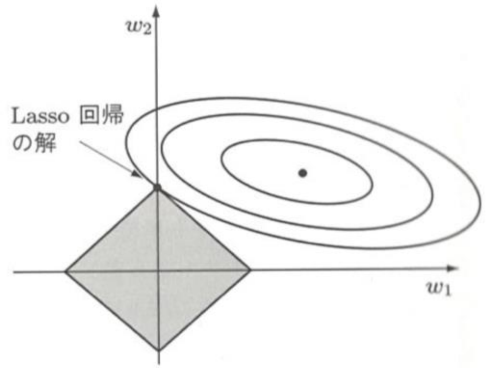
- パラメータの存在する範囲を円(えん)（d次元では超球(ちょうきゅう)）の中に限定して、それぞれの重みが大きな値をとれないようにする
  - 重み：誤差関数の等位線(とういせん)との接点(せってん)（＝円周(えんしゅう)上の点）

Lasso回帰

- パラメータの和が一定という条件なので、それぞれの軸で角をもつ領域に値が制限
  - 角(かど)で誤差関数の等位線と接(せっ)する（多くのパラメータが0になる）



Ridge回帰における
正則化



Lasso回帰における
正則化

バイアスと分散のトレードオフ

- 回帰式を高次方程式に置き換えて適用（てきよう）できる
  - 特徴ベクトル$\boldsymbol{x}$に対して、基底（きてい）関数ベクトル$\boldsymbol{\phi}(\boldsymbol{x})$を考える

$$\boldsymbol{\phi}(\boldsymbol{x}) = \left(\phi_1(\boldsymbol{x}), \dots, \phi_b(\boldsymbol{x})\right)^T$$

    - 例：1次元ベクトルxに対して $\boldsymbol{\phi}(x) = \left(1, x, x^2, \dots, x^b\right)^T$となる

  - 以下のように回帰式を定義すれば、
    係数が線形という条件のもとで最小二乗法が適用可能

$$\hat{c}(\boldsymbol{x}) = \sum_{j=0}^{b} w_j \, \phi_j(\boldsymbol{x})$$

- 複雑な関数を用いることで、
  真のモデルに近い形を表現できるのか？

バイアスと分散のトレードオフ

- バイアスと分散はトレードオフの関係
  - バイアス：真のモデルとの距離
  - 分散：学習結果の散らばり具合

- **単純なモデル** → バイアス：大、分散：小

  こ べ つ
  - 個別のデータに対する誤差が大きくなりやすいが、学習データが少し変動しても結果として得られるパラメータは大きく変動しない

  へんどう

- **複雑なモデル** → バイアス：小、分散：大
  - 個別のデータに対する誤差を小さくしやすいが、学習データの値が少し変動すると、結果が大きく異なることがある

バイアスと分散のトレードオフ

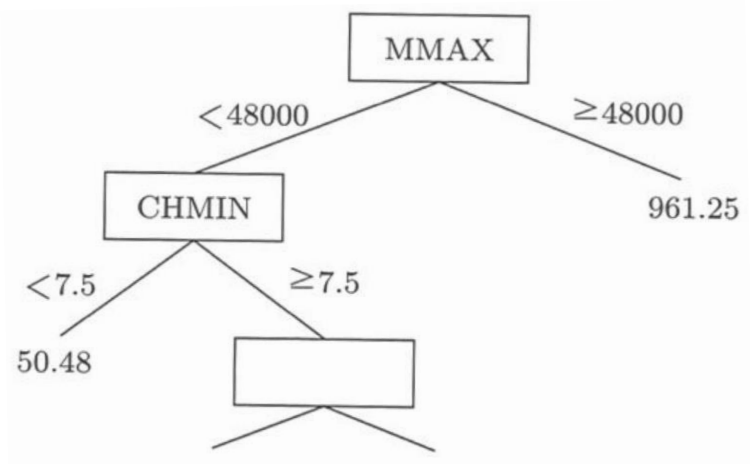
- 回帰問題におけるバイアスと分散
  - 線形回帰式の場合
    - 求まった超平面は、学習データ内の点をほとんど通らないので、バイアスが大きい
  - 「学習データの個数－１」次の高次回帰式の場合
    - 求まった回帰式は、全学習データを通る（学習データと一致する関数が求まる）ので、バイアスが小さい
    - データが少し動いただけで、この高次式は大きく変動するので、結果の分散は大きい
- 機械学習では、バイアスー分散のトレードオフを常に意識しなければならない
  - 正則化：緩いバイアスで分散を減らすのに有効

回帰木

- **回帰木**
  - 識別における決定木の考え方を回帰問題に適用する方法

- 決定木による識別問題の学習
  - 特徴の値によって学習データを同じクラスの集合になるように分割する

- 回帰木による回帰問題の学習
  - 出力値の近いデータが集まるように、特徴の値によって学習データを分割
  - 特徴をノードとし、出力値をリーフとする回帰木が得られる



回帰木の例

回帰木：CART

- **CART (classification and regression tree)**
  - にぶんぎ
    木の構造を二分木に限定した決定木
  - ふじゅんど
    分類基準：ジニ不純度（Gini impurity）

- CARTによる識別問題
  - 分類前後の集合のジニ不純度Gを求めて、改善度ΔGが最大のものをノードに選ぶことを再帰的に繰り返す
    さいきてき

$$G = 1 - \sum_{j=1}^{c} N(j)^2 \qquad \Delta G(D) = G(D) - P_L \cdot G(D_L) - P_R \cdot G(D_R)$$

D：あるノードに属するデータの全体
$N(j)$：データ中のクラスjの割合
D_L：左の部分木（D_Rは右の部分木）
P_L：D_Lに属するデータの割合（P_RはD_Rに属する）

回帰木：CART

CARTによる回帰問題

- 分類基準として、データの散らばりSSの減り方ΔSSが最大になるものを選択

$$SS(D) = \sum_{y_i \in D} (y_i - \tilde{y})^2$$

$$\Delta SS(D) = SS(D) - P_L \cdot SS(D_L) - P_R \cdot SS(D_R)$$

$\tilde{y}$： Dに属するデータの平均値
D：あるノードに属するデータの全体

D_L：左の部分木（D_Rは右の部分木）
P_L：D_Lに属するデータの割合（P_RはD_Rに属する）

$SS(D)$では、データDの分散を求めているので、
この基準は分割後の分散が最小となるような分割を求めている

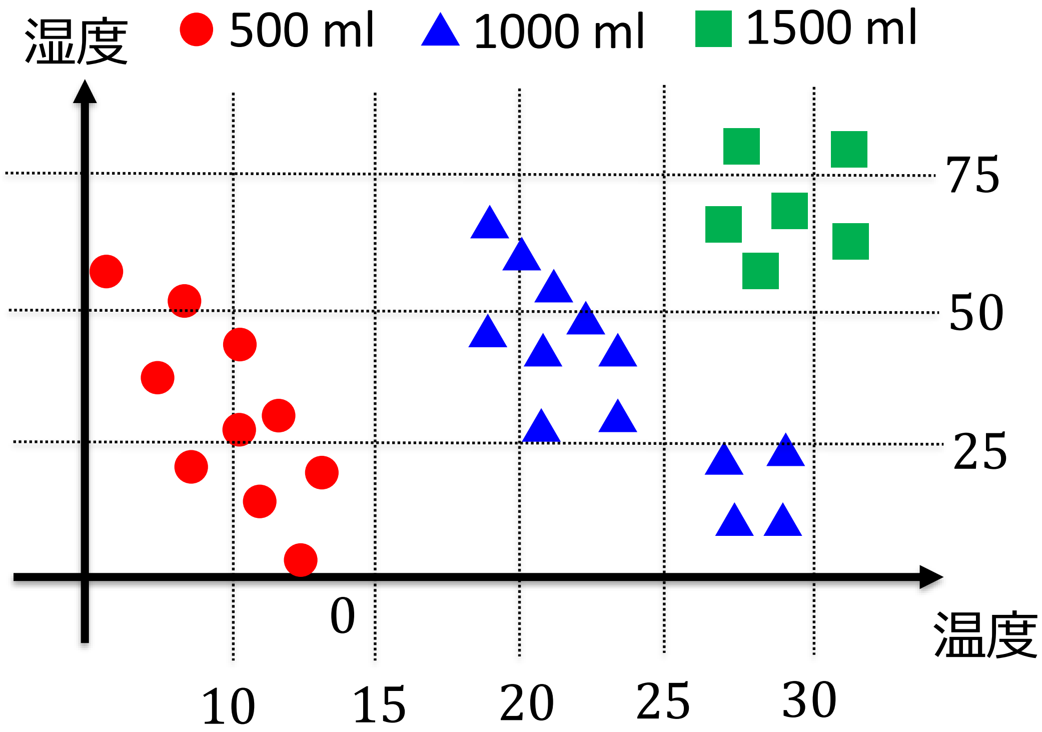
回帰木：モデル木

モデル木

- 回帰木の**リーフの値を線形回帰式**とした木
  - 回帰木と線形回帰の双方の利点を活かした方法
- モデル木の生成手順
  1. 出力値が近い区間を切り出せる特徴を選んでデータ分割
  2. 分割後のデータに対して線形回帰式を計算
- 特定の要因によって振る舞いが異なるデータを分割し、それぞれに対応する規則性を見つけ、かつ、その分割の要因を木構造によって説明できる
  - 例えば、季節によって出力に影響を及ぼす要因が異なるデータなどに有効

演習問題6-2（10分間）

- 以下の学習データが与えられたとき、温度と湿度から1日あたりのビールの消費量を予測する回帰木を作成せよ



温度・湿度・1日あたりのビールの消費量の関係