

統計的モデリング基礎⑤

～最尤推定（続き）～

鹿島久嗣
(情報学科 計算機科学コース)

マーケティング分野への応用を対象とした参考書



マーケティングの統計モデル

出版社：朝倉出版

発刊年月：2015.8

ISBN：4254128533

A5判；192ページ

マーケティングを題材としながら、基本的な統計的モデリングの方法が学べる

最尤推定：

データをもっともよく再現するパラメータを推定値とする

- n 個のデータ x_1, x_2, \dots, x_n から確率モデル $f(x \mid \theta)$ のパラメータ θ を推定したい
- n 個のデータが（互いに独立に）生成される確率（尤度）：

$$L(\theta) = \prod_{i=1}^n f(x_i \mid \theta)$$

実際には対数
尤度で扱うこと
が多い

- 尤度最大になるパラメータを推定値 $\hat{\theta}$ とする

$$\hat{\theta} = \operatorname{argmax}_{\theta} \prod_{i=1}^n f(x_i \mid \theta) = \operatorname{argmax}_{\theta} \sum_{i=1}^n \log f(x_i \mid \theta)$$

—もっともデータを生成する確率が高い（「最も尤もらしい」）

最尤推定の利点： モデリングの自動化

- 最尤推定の利点：確率モデルの形（データの生成プロセスの仮定）を決めればモデルパラメータが自動的に決まる
 - ーただし、最大化問題を解く必要がある
 - ー離散分布、ポアソン分布、正規分布などは解析的に解が求まる
 - ー線形回帰（正規分布でノイズが載る）は連立方程式（いちおう解析的な解）
 - ー多くのモデルでは、最適化問題を数値的に解く必要がある

判別問題：

ダミー変数を従属変数として説明（予測）する問題

- データ（ n 組の独立変数と従属変数）
 - 独立変数： $(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)})$
 - （ダミー）従属変数： $(y^{(1)}, y^{(2)}, \dots, y^{(n)}), y^{(i)} \in \{+1, -1\}$

以降、表記上の利便性からダミー従属変数を
 $\{0, 1\}$ でなく $\{+1, -1\}$ と表記する
(本質的な違いはナシ)

ロジスティック回帰： ダミー変数を従属変数とするモデル

- 以前、重回帰モデルでダミー変数を従属変数とすると、厳密には少しおかしいという話だった → もっとちゃんと扱いたい

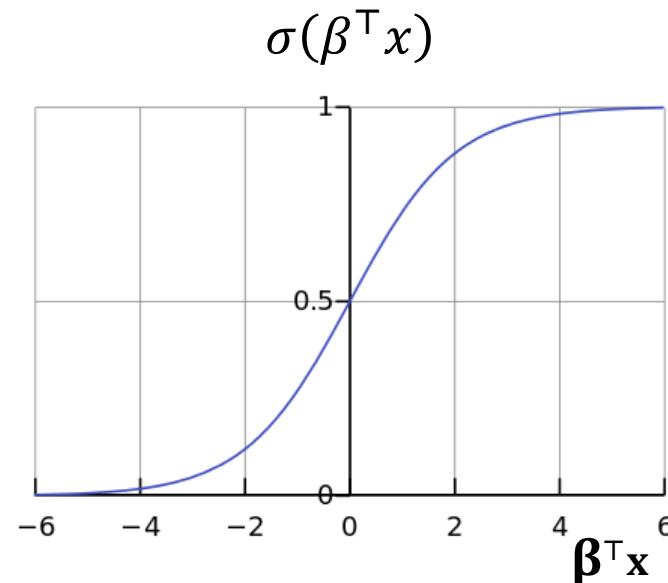
– 重回帰モデル $y = \boldsymbol{\beta}^\top \mathbf{x}$ の従属変数の値域は実数全体

- 従属変数の値域が $\{-1, +1\}$ もしくは、 $(0,1)$ ($Y = +1$ となる確率) となるようにしたい

- ロジスティック回帰モデル：

$$P(Y = 1 | \mathbf{x}, \boldsymbol{\beta}) = \frac{1}{1 + \exp(-\boldsymbol{\beta}^\top \mathbf{x})} = \sigma(\boldsymbol{\beta}^\top \mathbf{x})$$

– σ ：ロジスティック関数 ($\sigma: \mathbb{R} \rightarrow (0,1)$)



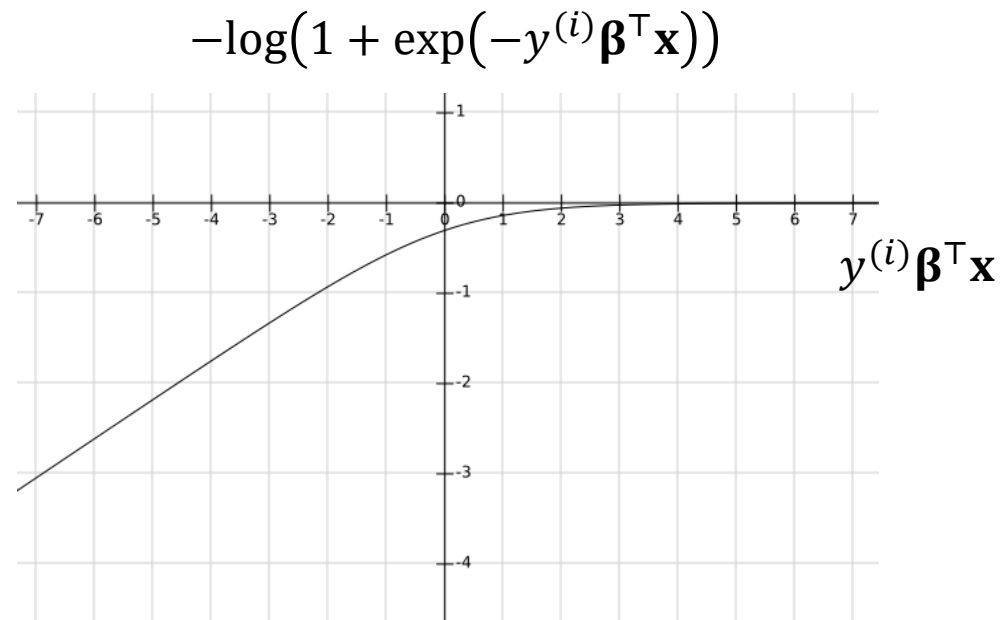
ロジスティック回帰モデルの対数尤度： 凸関数なので大局解が存在するが解析解はない

- 対数尤度：
$$L(\boldsymbol{\beta}) = -\sum_{i=1}^n \log(1 + \exp(-y^{(i)} \boldsymbol{\beta}^\top \mathbf{x}^{(i)}))$$
$$\left(= \sum_{i=1}^n \delta(y^{(i)} = 1) \log \frac{1}{1 + \exp(-\boldsymbol{\beta}^\top \mathbf{x}^{(i)})} + \delta(y^{(i)} = -1) \log \left(1 - \frac{1}{1 + \exp(-\boldsymbol{\beta}^\top \mathbf{x}^{(i)})} \right) \right)$$

- $L(\boldsymbol{\beta})$ は凸関数：

- 大局解がある

- 解析解はない



ロジスティック回帰のパラメータ推定： 非線形最適化

- 最尤推定の目的関数（最大化）：

$$L(\boldsymbol{\beta}) = - \sum_{i=1}^n \log(1 + \exp(-y^{(i)} \boldsymbol{\beta}^\top \mathbf{x}^{(i)}))$$

– 解析解は得られないが、凸関数（2階微分が ≤ 0 ）

- 数値的な最適化手法を使う

– パラメータの更新をくりかえす： $\boldsymbol{\beta}^{\text{NEW}} \leftarrow \boldsymbol{\beta} + \mathbf{d}$



パラメータ更新：

目的関数をもっとも改善する更新を行う

- 更新 $\boldsymbol{\beta}^{\text{NEW}} \leftarrow \boldsymbol{\beta} + \mathbf{d}$ によって目的関数の値が変化する：

$$L_{\mathbf{w}}(\mathbf{d}) = - \sum_{i=1}^n \ln(1 + \exp(-y^{(i)}(\boldsymbol{\beta} + \mathbf{d})^{\top} \mathbf{x}^{(i)}))$$

- $L_{\boldsymbol{\beta}}(\mathbf{d})$ を最大化する更新分 \mathbf{d}^* を見つけよ：

$$-\mathbf{d}^* = \operatorname{argmax}_{\mathbf{d}} L_{\boldsymbol{\beta}}(\mathbf{d})$$

最良のパラメータ更新： 目的関数をテイラー展開で近似

- 目的関数のテイラー展開：

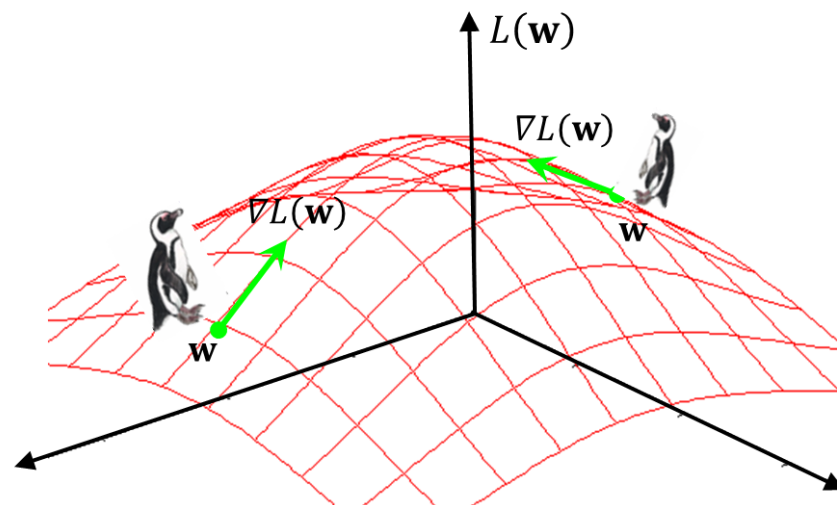
3次以上の項

$$L_{\boldsymbol{\beta}}(\mathbf{d}) = L(\boldsymbol{\beta}) + \mathbf{d}^{\top} \nabla L(\boldsymbol{\beta}) + \frac{1}{2} \mathbf{d}^{\top} \mathbf{H}(\boldsymbol{\beta}) \mathbf{d} + O(\mathbf{d}^3)$$

– 勾配： $\nabla L(\boldsymbol{\beta}) = \left(\frac{\partial L(\boldsymbol{\beta})}{\partial \beta_1}, \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_2}, \dots, \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_D} \right)^{\top}$

- 目的関数が最も急な方向

– ヘッセ行列： $[H(\boldsymbol{\beta})]_{i,j} = \frac{\partial^2 L(\boldsymbol{\beta})}{\partial \beta_i \partial \beta_j}$



ニュートン法：

2次近似した目的関数を最小化する解を求める

- テイラー展開で3次以降の項を無視する：

3次以上の項

$$L_{\beta}(\mathbf{d}) \approx L(\boldsymbol{\beta}) + \mathbf{d}^{\top} \nabla L(\boldsymbol{\beta}) + \frac{1}{2} \mathbf{d}^{\top} \mathbf{H}(\boldsymbol{\beta}) \mathbf{d} - \cancel{O(\mathbf{d}^3)}$$

- 最大化するために \mathbf{d} で微分： $\frac{\partial L_{\beta}(\mathbf{d})}{\partial \mathbf{d}} \approx \nabla L(\boldsymbol{\beta}) + \mathbf{H}(\boldsymbol{\beta}) \mathbf{d}$

- これを $= \mathbf{0}$ とおいて解くと： $\mathbf{d} = -\mathbf{H}(\boldsymbol{\beta})^{-1} \nabla L(\boldsymbol{\beta})$

実際には連立方程式を解く

- ニュートン法：

$$\boldsymbol{\beta}^{\text{NEW}} \leftarrow \boldsymbol{\beta} - \mathbf{H}(\boldsymbol{\beta})^{-1} \nabla L(\boldsymbol{\beta})$$



線形探索付きニュートン法：

近似は必ずしも正しくないので線形探索と組み合わせる

- ニュートン法の更新 $\boldsymbol{\beta}^{\text{NEW}} \leftarrow \boldsymbol{\beta} - \boldsymbol{H}(\boldsymbol{\beta})^{-1} \nabla L(\boldsymbol{\beta})$ は 2 次近似が正しいことを仮定している：

$$L_{\boldsymbol{\beta}}(\mathbf{d}) \approx L(\boldsymbol{\beta}) + \mathbf{d}^{\top} \nabla L(\boldsymbol{\beta}) + \frac{1}{2} \mathbf{d}^{\top} \boldsymbol{H}(\boldsymbol{\beta}) \mathbf{d}$$

– 実際には正しくない

- そこで更新の向き $\boldsymbol{H}(\boldsymbol{\beta})^{-1} \nabla L(\boldsymbol{\beta})$ のみを採用して：

$$\boldsymbol{\beta}^{\text{NEW}} \leftarrow \boldsymbol{\beta} - \eta \boldsymbol{H}(\boldsymbol{\beta})^{-1} \nabla L(\boldsymbol{\beta})$$

- 学習率 $\eta > 0$ の決定法：

– 適当にステップ数とともに適当に減衰

– 線形探索： $\eta^* = \operatorname{argmax}_{\eta} L(\boldsymbol{\beta} - \eta \boldsymbol{H}(\boldsymbol{\beta})^{-1} \nabla L(\boldsymbol{\beta}))$

適当な初期値から始めて、
目的関数が改善しない間
は η を半分にしていく

最急降下法：

ヘッセ行列を使わずシンプルで軽い更新

- ヘッセ行列の逆行列（もしくは連立方程式を解く）は高コスト：

- ニュートン法の更新： $\boldsymbol{\beta}^{\text{NEW}} \leftarrow \boldsymbol{\beta} - \eta \mathbf{H}(\boldsymbol{\beta})^{-1} \nabla L(\boldsymbol{\beta})$

- 最急降下法：

- ヘッセ行列の逆行列 $\mathbf{H}(\boldsymbol{\beta})^{-1}$ を単位行列 \mathbf{I} で置き換え：

$$\boldsymbol{\beta}^{\text{NEW}} \leftarrow \boldsymbol{\beta} - \eta \nabla L(\boldsymbol{\beta})$$

勾配

- $\nabla L(\boldsymbol{\beta})$ は最も急な（目的関数が最も変化する）向き

- 学習率 η は線形探索で求める：



確率的最適化とミニバッチ：

データの部分集合を用いた効率的な推定

- 対数尤度は各データの対数尤度の和： $L(\boldsymbol{\beta}) = \sum_{i=1}^n \ell^{(i)}$

- 勾配 $\frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_{i=1}^n \frac{\partial \ell^{(i)}}{\partial \boldsymbol{\beta}}$ の計算は $O(n)$ かかる

i 番目のデータの
対数尤度

- 勾配をデータ1個で近似： $\frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \approx n \frac{\partial \ell^{(i)}}{\partial \boldsymbol{\beta}}$

– 確率的最適化：毎回データをランダムに選ぶ

– オンライン推定も可能（時刻 t のデータの $\ell^{(i)}$ を使う）

- ミニバッチ： $1 < m < n$ 個のデータで勾配を近似：

$$\frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \approx \frac{n}{m} \sum_{j \in \text{MiniBatch}} \frac{\partial \ell^{(i)}}{\partial \boldsymbol{\beta}}$$

ロジスティック回帰の場合の勾配： 比較的簡単に計算可能

- 対数尤度： $L(\boldsymbol{\beta}) = -\sum_{i=1}^n \ln(1 + \exp(-y^{(i)} \boldsymbol{\beta}^\top \mathbf{x}^{(i)}))$

- $$\begin{aligned} \frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} &= -\sum_{i=1}^n \frac{1}{1 + \exp(-y^{(i)} \boldsymbol{\beta}^\top \mathbf{x}^{(i)})} \frac{\partial (1 + \exp(-y^{(i)} \boldsymbol{\beta}^\top \mathbf{x}^{(i)}))}{\partial \boldsymbol{\beta}} \\ &= \sum_{i=1}^n \frac{1}{1 + \exp(-y^{(i)} \boldsymbol{\beta}^\top \mathbf{x}^{(i)})} \exp(-y^{(i)} \boldsymbol{\beta}^\top \mathbf{x}^{(i)}) y^{(i)} \mathbf{x}^{(i)} \\ &= \sum_{i=1}^n (1 - f(y^{(i)} | \mathbf{x}^{(i)}, \boldsymbol{\beta})) y^{(i)} \mathbf{x}^{(i)} \end{aligned}$$

現在のパラメータでのモデルが与える確率

練習： ポアソン回帰の最尤推定

- （前回出てきた）ポアソン回帰の最尤推定

－対数尤度：

$$L(\boldsymbol{\beta}) = \sum_{i=1}^n y^{(i)} \boldsymbol{\beta}^\top \mathbf{x}^{(i)} - \sum_{i=1}^n \exp(\boldsymbol{\beta}^\top \mathbf{x}^{(i)}) + \text{const.}$$

－解析解は求まらない

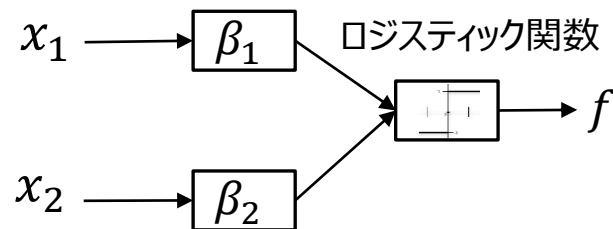
- 最急勾配法の更新式を求めてみる

ニューラルネットワーク：

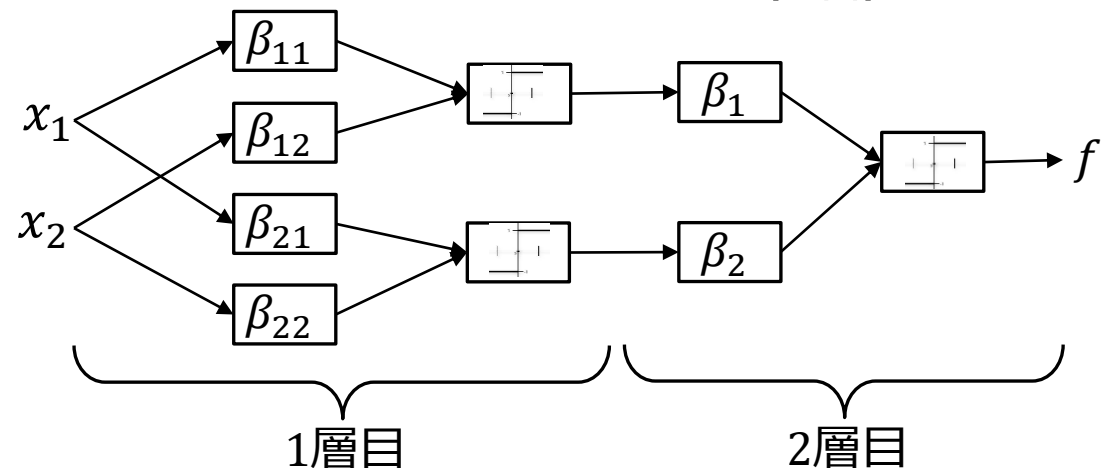
(ざっくりいえば) ロジスティック回帰モデルを連結したもの

- ニューラルネットワークはロジスティック回帰モデルを連結したもの
 - 複数のロジスティック回帰モデルの出力が、別のロジスティック回帰モデルの入力になる
 - ロジスティック関数（非線形）によりモデルに非線形性を導入
 - 両者とも $Y = 1$ である確率 $f(\mathbf{x}; \boldsymbol{\beta})$ を出力するモデル

ロジスティック回帰モデル



ニューラルネットワーク（2層）



ニューラルネットワークのパラメータ推定： 最急降下法を適用するために勾配の計算が必要

- パラメータ推定を最尤推定で行うとすると、目的関数は：

$$L(\boldsymbol{\beta}) \\ = - \sum_{i=1}^n \left(\delta(y^{(i)} = 1) \log f(x^{(i)}) + \delta(y^{(i)} = -1) \log (1 - f(x^{(i)})) \right)$$

- $L(\boldsymbol{\beta})$ の勾配 $\frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}$ が計算できれば最急降下法を適用できる
 - 実際は確率的最適化やミニバッチを用いることが多い

ニューラルネットワークの勾配計算： 誤差逆伝播法によって計算する

- 誤差逆伝播法： $L(\boldsymbol{\beta})$ の勾配 $\frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}$ を計算する方法
 - f から「後ろ向きに」遡っていく計算（微分の連鎖率）
- 1次元の場合で考える（多次元でもほぼそのまま）：

$$- \frac{\partial L}{\partial \beta_2} = \frac{\partial L}{\partial f} \cdot \frac{\partial f}{\partial g_2} \cdot \frac{\partial g_2}{\partial \beta_2}$$

$$- \frac{\partial L}{\partial \beta_1} = \frac{\partial L}{\partial f} \cdot \frac{\partial f}{\partial g_2} \cdot \frac{\partial g_2}{\partial f_1} \cdot \frac{\partial f_1}{\partial g_1} \cdot \frac{\partial g_1}{\partial \beta_1}$$

共通

