ロジスティック回帰

氏 名： 西本　洋紀

2019 年 6 月 11 日

1　導入

ロジスティック回帰による 2 クラス分類を考える。データ集合

$$\{\phi_n, t_n\}, t_n \in \{0, 1\} \tag{1}$$

$$\text{但し} \phi_n = \phi(\mathbf{x_n}), n = 1, ..., N \tag{2}$$

が与えられたとする。

$\phi(\mathbf{x})$ は基底関数と呼ばれる。基底関数が非線形だったら入力空間の決定局面も非線形になるが、パラメータに関しては依然線形なので計算が簡単というメリットが有る。本稿では、とりあえず基底関数は恒等関数 $\phi(\mathbf{x}) = x$ であると思って差し支えない。

クラスラベルを C_1, C_2 とすると、クラスの事後確率は下のように書ける。

$$p(C_1|\phi) = y(\phi) = \sigma(\mathbf{w}^T\phi) \tag{3}$$

$$p(C_2|\phi) = 1 - p(C_1|\phi) \tag{4}$$

ここで、事後確率 (訓練データが与えられたとき、新しいデータがクラス C に属する確率) を陽に定式化するモデルを識別モデルという。ロジスティック回帰は識別モデルの 1 つである。

$y_n = \sigma(a_n), a_n = \mathbf{w}^T\phi_n$ として、尤度関数は

$$p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^{N} y_n^{t_n}\{1 - y_n\}^{1-t_n} \tag{5}$$

と書ける。これは、パラメータ $\mathbf{w}$ がある値をとったときに、手元のデータが生起する確率を表す。「尤度」は事象の尤もらしさ、すなわち起こりやすさであり、これをパラメータ $\mathbf{w}$ の関数と見ているので尤度関数という。

ここで、

$$\sigma(x) = \frac{1}{1 + \exp(-x)} \tag{6}$$

はロジスティックシグモイド関数と呼ばれ、$(-\infty, \infty)$ の入力を $(0, 1)$ に押し込める、連続な単調増加関数である

負の対数尤度 (尤度関数の対数をとってマイナスをかけたもの) は

$$-\ln p(\mathbf{t}|\mathbf{w}) = -\sum_{n=1}^{N}\{t_n \ln y_n + (1-t_n)\ln(1-y_n)\} \tag{7}$$

と書ける。

対数をとるのは、尤度関数は確率の積であり、サンプル数 N が多いと尤度はすごく小さくなってしまうからである。もっというと、コンピュータで小さすぎる小数を扱うのはすごく大変である。対数をとると、積がわになって問題を解決できる。

cf.) float 型の有効数字は小数点以下 7 桁。

マイナスを掛けるのは、最大化問題よりも最小化問題として扱いたいからである（慣習）。

2　ロジスティック回帰のお気持ち

ロジスティック回帰のお気持ちは、手元のデータが生起する確率（尤度) を最大化するようなパラメータ $\mathbf{w}$ を選ぶことで、これは、負の対数尤度の最小化と等価である。

すなわち、ロジスティック回帰では、負の対数尤度を誤差関数とみて最小化する。

$$E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{w}) = -\sum_{n=1}^{N}\{t_n y_n + (1-t_n)\ln(1-y_n)\} \tag{8}$$

2.1　最適化問題の考察

一般に、「f のヘッセ行列が凸関数 $\Leftrightarrow$ f が任意の点で半正定値」が成り立つ。$E(\mathbf{w})$ のヘッセ行列は半正定値なので、$E(\mathbf{w})$ は凸関数であり、局所解は大局解に一致する。

ここで、$n \times n$ の行列Aが半正定値であるとは、全てのn次元実ベクトル $\mathbf{x}$ に対して二次形式 $\mathbf{x}^T A\mathbf{x}$ 非負 であることをいう。

また、関数 f のヘッセ行列とは、ざっくり言うと f の 2 回微分を要素とする行列である。

よって $E(\mathbf{w})$ が最小になるのは、

$$\nabla E(\mathbf{w}) = 0 \tag{9}$$

のときに限ることがわかる。

実際の計算は、ニュートンラフソン法で求める。

ニュートンラフソン法は、関数 $f(x) = 0$ を数値計算で解く方法の 1 つで、曲線 $y = f(x)$ を局所的に直線近似し、直線が $y = 0$ と交わる点で次の更新を行うことを繰り返す。

$$\mathbf{w}^{new} = \mathbf{w}^{old} - \mathbf{H}^{-1}\nabla E(\mathbf{w}) \tag{10}$$

∇ はナブラ演算子と呼ばれ、ベクトル関数に掛かって勾配を与えるものである。

3　正則化

特に、線形分離可能なデータ集合に対し、上で定式化したロジスティック回帰は過学習しやすいことが知られている。また、基底関数を高次多項式関数にするなどして、柔軟なモデルをデータに当てはめた場合や、データ数が少ない場合にも、過学習が起こりやすい。そのような過学習を抑えるアイデアの 1 つに、ただ誤差関数 $E(\mathbf{w})$ を最小化するのではなく、パラメータ $\mathbf{w}$ の大きさも一緒に最小化する**正則化**がある。

L1 正則化は、$E(\mathbf{w})$ の代わりに下の誤差関数を最小化する。

$$E'(\mathbf{w}) = E(\mathbf{w}) + C\|\mathbf{w}\| \tag{11}$$

L2 正則化では、$E(\mathbf{w})$ の代わりに下の誤差関数を最小化する。

$$E''(\mathbf{w}) = E(\mathbf{w}) + C\|\mathbf{w}\|_2^2 \tag{12}$$

L1 正則化、L2 正則化ともに、$E(\mathbf{w})$ を最小化するのと同じような手順で最適解 $\mathbf{w}^*$ が求められる。

L1 正則化の最適解 $\mathbf{w}^*$ は、要素の多くが 0 に潰れており、それにかかる $\mathbf{x}$ の要素は予測にほとんど寄与しないことがわかる。この性質から、L1 正則化は特徴量選択の方法の 1 つとして使われる。

入力データ $\mathbf{x}$ の要素間の相関が強いときには、多重共線性の問題が生じるが、L2 正則化項を誤差関数に付与することで、この問題を回避できることが知られている。