Logistische Regression

M. Gruber

KW 41

Beispiel 1 (Kreditanalyse) *[1], Example 3.4. Kreditanalyse führt je nach Art der Fragestellung zu verschiedenen Grundaufgaben des Maschinenlernens.*

Fragestellung	*Lernmethode*	*Lernalgorithmus*	*Fehlerfunktion*
Vergabe Ja/Nein?	*Perzeptron*	*PLA, Pocket*	$\frac{1}{N}\sum_{1\le n\le N}[\operatorname{sgn} w^T x^{(n)} \neq y^{(n)}]$
Kredithöhe?	*Lineare Regression*	*Pseudoinverse*	$\frac{1}{N}\sum_{1\le n\le N}\lvert w^T x^{(n)} - y^{(n)}\rvert^2$
Ausfallwahrscheinlichkeit?	*Logistische Regression*	*Gradientenabstieg*	$\frac{1}{N}\sum_{1\le n\le N}\ln(1+e^{-y^{(n)}w^T x^{(n)}})$

Bei der logistischen Regression wird die bedingte Wahrscheinlichkeit $\mathbf{P}(y \mid x)$, $y = \pm 1$, gelernt, die für die Entstehung der Lernmenge $D = \{[x^{(1)}, y^{(1)}], \ldots, [x^{(N)}, y^{(N)}]\}$ verantwortlich ist. Der Ansatz für die Hypothese w ist

$$\mathbf{P}(y \mid x) = \begin{cases} \dfrac{e^{w^T x}}{1+e^{w^T x}} & \text{für } y = 1, \\ 1 - \dfrac{e^{w^T x}}{1+e^{w^T x}} & \text{für } y = -1. \end{cases}$$

Die verwendete logistische Funktion $\theta(s) = \dfrac{e^s}{1+e^s}$ hat zwei schöne Eigenschaften. Erstens bildet sei die reellen Zahlen auf das Intervall $[0, 1]$ ab. Ihre Werte können als Wahrscheinlichkeiten interpretiert werden. Zweitens ist $1 - \theta(s) = \theta(-s)$. Das erlaubt eine einfachere Darstellung von $\mathbf{P}(y \mid x)$, nämlich

$$\mathbf{P}(y \mid x) = \theta(y w^T x) = \frac{e^{y w^T x}}{1+e^{y w^T x}} \text{ für } y = \pm 1.$$

Welche Hypothese w ist die beste? Diejenige, die die Lermenge am wahrscheinlichsten erscheinen lässt (Likelihood-Argument). Gesucht ist also die Hypothese w, die das Likelihood-Maß

$$\prod_{1\le n\le N} \mathbf{P}(y^{(n)} \mid x^{(n)}) = \prod_{1\le n\le N} \theta(y^{(n)} w^T x^{(n)})$$

maximiert. Es ist

$$\prod_{1\le n\le N} \theta(y^{(n)} w^T x^{(n)}) = \prod_{1\le n\le N} \frac{e^{y^{(n)} w^T x^{(n)}}}{1+e^{y^{(n)} w^T x^{(n)}}} = \prod_{1\le n\le N} \frac{1}{1+e^{-y^{(n)} w^T x^{(n)}}}.$$

Statt $\prod_{1 \leq n \leq N} \frac{1}{1 + e^{-y^{(n)} w^T x^{(n)}}}$ kann man $\sum_{1 \leq n \leq N} - \ln(1 + e^{-y^{(n)} w^T x^{(n)}})$ maximieren oder den Kreuzentropiefehler

$$\frac{1}{N} \sum_{1 \leq n \leq N} \ln(1 + e^{-y^{(n)} w^T x^{(n)}})$$

minimieren.

Zur Minimierung des Kreuzentropiefehlers kann man auf die Gradientenabstiegsmethode zurückgreifen. Sei $e_n(w) = \ln(1 + e^{-y^{(n)} w^T x^{(n)}})$. Der Gradientenabstieg iteriert den Schritt

$$w_{\text{neu}} \leftarrow w_{\text{alt}} - \eta \frac{1}{N} \sum_{1 \leq n \leq N} \nabla e_n(w_{\text{alt}})$$

bis zur Erreichung eines (lokalen) Minimums. Dabei ist η ein geeigneter Lernkoeffizient.

Es ist $\nabla e_n(w) = -\theta(-y^{(n)} w^T x^{(n)}) y^{(n)} x^{(n)}$.

Der Aufwand für den Gradientenabstieg ist hoch, denn bei jedem Schritt muss die gesamte Lernmenge zur Berechnung von w_{neu} herangezogen werden. Günstiger ist der *stochastische Gradientenabstieg*. Bei diesem wird für jeden Schritt ein $n \in \{1, \ldots, N\}$ zufällig bestimmt und

$$w_{\text{neu}} \leftarrow w_{\text{alt}} - \eta \nabla e_n(w_{\text{alt}})$$

ausgeführt. In unserem Fall bedeutet dies

$$w_{\text{neu}} \leftarrow w_{\text{alt}} + \eta \theta(-y^{(n)} w_{\text{alt}}^T x^{(n)}) y^{(n)} x^{(n)}.$$

Für den stochastischen Gradientenabstieg sprechen drei Argumente. Erstens ist der Erwartungswert seiner Einzelschritte der Einzelschritt aus dem klassischen Gradientenabstieg. Zweitens bleibt der stochastische Gradientenabstieg nicht so leicht in einem lokalen Minimum stecken wie der klassische. Und drittens ist der Aufwand geringer.

Für den Lernkoeffizient η ist 0.1 oft die passende Wahl (Erfahrungswert).

Literatur

[1] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin. *Learning From Data*. AMLBook, 2012.