Technische Hochschule Wildau [FH]
Technical University of Applied Sciences
WILDAU

Maschinelles Lernen

Multivariate Lineare Regression mit Gradientenabstieg

Prof. Dr. Rainer Stollhoff

Übersicht

- Motivation
- Multivariater Gradientenabstieg

Supervised Learning

Supervised Learning

1. Aufgabe A

   Vorhersage $\hat{Y} = A(X)$

2. Qualität Q

   Verlustfunktion $\mathrm{L}(\hat{Y}, \mathrm{Y})$

3. Erfahrung E

Datensatz
(x_i, y_i) für $i = 1, \cdots, n$

Eine Maschine ***lernt*** aus Erfahrung E eine Aufgabe A mit der Qualität Q, wenn die Qualität Q beim erfüllen der Aufgabe A mit Erfahrung E steigt (T. Mitchell, MIT, 1988)

Einfache univariate Regression – Gradientenabstieg

Aufgabe: Regression, d.h. Vorhersage $\hat{y} = \hat{y}(x) = f(x)$

Erfahrung: Datensatz $(x_i, y_i)_{i=1}^n$

Qualität: Quadratische Verlustfunktion

$$L(y, \hat{y}) = \sum_{i=1}^{n} (y_i - \hat{y_i})^2 = \sum_{i=1}^{n} (y_i - f(x_i; \theta))^2 = L(\theta)$$

Maschine: vereinfachte lineare Regression mit

$$\mathrm{f}(x; \theta) = \theta \cdot x$$

Lernen: Finde einen Wert für θ , der die quadratische Verlustfunktion minimiert

Durch geeignete Wahl von θ in einem iterativen Prozess (Gradientenabstiegsverfahren):

1. Wähle Startwert z.B. $\theta^0 = 0$
2. Berechne Ableitung $\frac{d}{d\theta} L(\theta) = \frac{d}{d\theta} \sum_{i=1}^n (y_i - f(x_i; \theta))^2 = \frac{d}{d\theta} \sum_{i=1}^n (y_i - x_i \cdot \theta)^2$
$$= \sum_{i=1}^{n} 2 \cdot (y_i - x_i \cdot \theta) \cdot (-x_i)$$
3. Update $\theta^{t+1} = \theta^t + \alpha \cdot \frac{d}{d\theta} L(\theta^t) = \theta^k - \alpha \cdot 2 \sum_{i=1}^n (y_i - x_i \cdot \theta^t) \cdot x_i = \theta^k + 2 \sum_{i=1}^n (y_i - x_i \cdot \theta^t) \cdot x_i$

Multivariate Regression – Gradientenabstieg

Aufgabe: Regression, d.h. Vorhersage $\hat{y} = \hat{y}(x) = f(x)$

Erfahrung: Datensatz $(x_i, y_i)_{i=1}^{n}$

Qualität: Quadratische Verlustfunktion

$$L(y, \hat{y}) = \sum_{i=1}^{n} (y_i - \hat{y_i})^2 = \sum_{i=1}^{n} (y_i - f(x_i; \theta))^2 = L(\theta)$$

Maschine: vereinfachte lineare Regression mit

$$\mathrm{f}(x_1, x_2; \theta_0, \theta_1, \theta_2) = \theta_0 + \theta_1 \cdot x_1 + \theta_2 \cdot x_2$$

Lernen: Finde Werte für $\theta = (\theta_0, \theta_1, \theta_2)$, die die quadratische Verlustfunktion minimieren

Multivariate Regression – Gradientenabstieg

Technische Hochschule Wildau
Technical University of Applied Sciences
WILDAU

Aufgabe: Regression, d.h. Vorhersage $\hat{y} = \hat{y}(x) = f(x)$

Erfahrung: Datensatz $(x_i, y_i)_{i=1}^n$

Qualität: Quadratische Verlustfunktion

$$L(y,\hat{y}) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - f(x_i;\theta))^2 = L(\theta)$$

Maschine: vereinfachte lineare Regression mit

$$\mathrm{f}(x_1, x_2; \theta_0, \theta_1, \theta_2) = \theta_0 + \theta_1 \cdot x_1 + \theta_2 \cdot x_2$$

Lernen: Finde Werte für $\theta = (\theta_0, \theta_1, \theta_2)$, die die quadratische Verlustfunktion minimieren

Durch geeignete Wahl von θ in einem iterativen Prozess (Gradientenabstiegsverfahren):

1. Wähle Startwert z.B. $\theta^0 = 0$
2. Berechne Ableitung $\frac{d}{d\theta} L(\theta) = \frac{d}{d\theta} \sum_{i=1}^n (y_i - f(x_i;\theta))^2 = \frac{d}{d\theta} \sum_{i=1}^n (y_i - x_i \cdot \theta)^2$
   $= \sum_{i=1}^{n} 2 \cdot (y_i - x_i \cdot \theta) \cdot (-x_i)$
3. Update $\theta^{t+1} = \theta^t + \alpha \cdot \frac{d}{d\theta} L(\theta^t) = \theta^k - \alpha \cdot 2 \sum_{i=1}^n (y_i - x_i \cdot \theta^t) \cdot x_i = \theta^k + 2 \sum_{i=1}^n (y_i - x_i \cdot \theta^t) \cdot x_i$

Technische
Hochschule
Wildau [FH]
Technical University
of Applied Sciences

Maschinelles Lernen

Multivariate Analysis - Einführung

Prof. Dr. Rainer Stollhoff

Univariate Lineare Regression – Gradientenabstieg

Aufgabe: Regression, d.h. Vorhersage $\hat{y} = \hat{y}(x) = f(x)$

Erfahrung: Datensatz $(x_i, y_i)_{i=1}^{n}$

Qualität: Verlustfunktion: $L(y, \hat{y}) = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - f(x_i; \theta))^2 = L(\theta)$

Maschine: Regression mit $\mathrm{f}(x; \theta) = \theta \cdot x$

Lernen: Finde Werte für $\theta = (\theta)$, die die quadratische Verlustfunktion minimieren

Durch geeignete Wahl von θ in einem iterativen Prozess (Gradientenabstiegsverfahren):

1. Wähle Startwert z.B. $\theta^0 = (1)$
2. Berechne Gradienten

   $$\nabla L(\theta^0) = \left(\frac{\partial}{\partial \theta} L(\theta^0)\right) = \left(\sum_{i=1}^{n}(y_i - (\theta \cdot x_i)) \cdot (-2 \cdot x_i)\right)$$

3. Update $\theta^{t+1} = \theta^t + \alpha \cdot \nabla L(\theta^t)$

Bivariate Lineare Regression – Gradientenabstieg

Technische Hochschule Wildau
Technical University of Applied Sciences
WILDAU

Aufgabe: Regression, d.h. Vorhersage $\hat{y} = \hat{y}(x) = f(x)$

Erfahrung: Datensatz $(\boldsymbol{x}_i, y_i)_{i=1}^{n}$ **mit $\boldsymbol{x_i} = (\boldsymbol{x_{i,1}}, \boldsymbol{x_{i,2}})$**

Qualität: Verlustfunktion: $L(y, \hat{y}) = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - f(x_i;\theta))^2 = L(\theta)$

Maschine: Regression mit f$(x_1, \boldsymbol{x_2}; \boldsymbol{\theta_0}, \theta_1, \boldsymbol{\theta_2}) = \boldsymbol{\theta_0} + \theta_1 \cdot x_1 + \boldsymbol{\theta_2} \cdot \boldsymbol{x_2}$

Lernen: Finde Werte für $\theta = (\boldsymbol{\theta_0}, \theta_1, \boldsymbol{\theta_2})$, die die quadratische Verlustfunktion minimieren

Durch geeignete Wahl von θ in einem iterativen Prozess (Gradientenabstiegsverfahren):

1. Wähle Startwert z.B. $\theta^0 = \begin{pmatrix} \mathbf{0} \\ 1 \\ \mathbf{1} \end{pmatrix}$

2. Berechne Gradienten

$$\nabla L(\theta^0) = \begin{pmatrix} \boldsymbol{\frac{\partial}{\partial \theta_0} L(\theta^0)} \\ \frac{\partial}{\partial \theta_1} L(\theta^0) \\ \boldsymbol{\frac{\partial}{\partial \theta_2} L(\theta^0)} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\sum_{i=1}^{n}\left(y_i - \left(\theta_0^0 + \theta_1^0 \cdot x_{i,1} + \theta_2^0 \cdot x_{i,2}\right)\right) \cdot (-2 \cdot 1)} \\ \sum_{i=1}^{n}\left(y_i - \left(\boldsymbol{\theta_0^0} + \theta_1^0 \cdot x_{i,1} + \boldsymbol{\theta_2^0 \cdot x_{i,2}}\right)\right) \cdot (-2 \cdot x_{i,1}) \\ \boldsymbol{\sum_{i=1}^{n}\left(y_i - \left(\theta_0^0 + \theta_1^0 \cdot x_{i,1} + \theta_2^0 \cdot x_{i,2}\right)\right) \cdot (-2 \cdot x_{i,2})} \end{pmatrix}$$

3. Update $\theta^{t+1} = \theta^t + \alpha \cdot \nabla L(\theta^t)$

Multivariate Lineare Regression – Gradientenabstieg

Technische Hochschule Wildau
Technical University of Applied Sciences
WILDAU

Aufgabe: Regression, d.h. Vorhersage $\hat{y} = \hat{y}(x) = f(x)$

Erfahrung: Datensatz $(x_i, y_i)_{i=1}^{n}$ mit $x_i = (x_{i,1}, x_{i,2}, \cdots, \boldsymbol{x_{i,n}})$

Qualität: Verlustfunktion: $L(y, \hat{y}) = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - f(x_i; \theta))^2 = L(\theta)$

Maschine: Regression mit f$(\boldsymbol{x}; \boldsymbol{\theta}) = \theta_0 + \theta_1 \cdot x_1 + \theta_2 \cdot x_2 + \cdots + \boldsymbol{\theta_n} \cdot \boldsymbol{x_n}$

Lernen: Finde Werte für $\theta = (\theta_0, \theta_1, \theta_2, \cdots, \boldsymbol{\theta_n})$, die die quadratische Verlustfunktion minimieren

Durch geeignete Wahl von θ in einem iterativen Prozess (Gradientenabstiegsverfahren):

1. Wähle Startwert z.B. $\theta^0 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ \boldsymbol{1} \end{pmatrix}$

2. Berechne Gradienten

$$\nabla L(\theta^0) = \begin{pmatrix} \frac{\partial}{\partial \theta_0} L(\theta^0) \\ \frac{\partial}{\partial \theta_1} L(\theta^0) \\ \vdots \\ \boldsymbol{\frac{\partial}{\partial \theta_n} L(\theta^0)} \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1 \cdot x_1 + \theta_2 \cdot x_2 + \cdots + \boldsymbol{\theta_n} \cdot \boldsymbol{x_n})) \cdot (-2 \cdot 1) \\ \sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1 \cdot x_1 + \theta_2 \cdot x_2 + \cdots + \boldsymbol{\theta_n} \cdot \boldsymbol{x_n})) \cdot (-2 \cdot x_{i,1}) \\ \vdots \\ \boldsymbol{\sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1 \cdot x_1 + \theta_2 \cdot x_2 + \cdots + \theta_n \cdot x_n)) \cdot (-2 \cdot x_{i,n})} \end{pmatrix}$$

3. Update $\theta^{t+1} = \theta^t + \alpha \cdot \nabla L(\theta^t)$

Exkurs: Multivariate Lineare Regression – Analytisch / Lineare Algebra

Technische Hochschule Wildau
Technical University of Applied Sciences
WILDAU

- Datensatz $(\boldsymbol{x}_i, y_i)_{i=1}^{n}$ in Matrixschreibweise (X, Y)
- Verlustfunktion und Ableitung in Matrixschreibweise

$$\nabla \sum_{i=1}^{n} \left(y_i - \hat{f}(\boldsymbol{x}_i; \boldsymbol{\theta})\right)^2 = \nabla \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i \boldsymbol{\theta})^2$$
$$= \nabla\left((Y - X\boldsymbol{\theta})^T (Y - X\boldsymbol{\theta})\right) = 2X^T (Y - X\boldsymbol{\theta})$$

- Algebraische Lösung in Matrixschreibweise

$$\theta \quad = (X^T X)^{-1} X^T Y$$

Exkurs: Gradientenabstieg oder Lineare Algebra?

Gradientenabstieg

- Iterative Berechnung

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^{t} - \alpha \cdot \nabla L(\boldsymbol{\theta}^{t})$$

- Wahl der Lernrate α bzw. Matrix A
  - Fest
  - Adaptiv
- Benötigt u.U. viele Iterationen
- Funktioniert auch für großes n, d.h. viele Daten

Lineare Algebra

- Algebraische Lösung

$$\boldsymbol{\theta} \quad = (X'X)^{-1}X'Y$$

- keine Meta-Parameter
- Direkte Lösung, keine Iterationen
- Benötigt Berechnung von $(X'X)^{-1}$, d.h.
  - Inverse muss existieren – insbesondere keine linear abhängigen Variablen!
  - Rechenintensives Invertieren einer nxn Matrix ~ $O(n^3)$
- Langsam für große n

Ausblick: Multivariate Regression – Gradientenabstieg

Aufgabe: Regression, d.h. Vorhersage $\hat{y} = \hat{y}(x) = f(x)$

Erfahrung: Datensatz $(x_i, y_i)_{i=1}^{n}$ mit $x_i = (x_{i,1}, x_{i,2}, \cdots, x_{i,n})$

Qualität: Verlustfunktion: $\boldsymbol{L(y, \hat{y}) = L(\theta)}$

Maschine: Regression mit $\mathbf{f}(\boldsymbol{x}; \boldsymbol{\theta})$

Lernen: Finde Werte für $\theta = (\theta_0, \theta_1, \theta_2, \cdots, \theta_n)$, die die quadratische Verlustfunktion minimieren

Durch geeignete Wahl von θ in einem iterativen Prozess (Gradientenabstiegsverfahren):

1. Wähle Startwert z.B. $\theta^0 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$

2. Berechne Gradienten

$$\nabla L(\theta^0) = \begin{pmatrix} \frac{\partial}{\partial \theta_0} L(\theta^0) \\ \frac{\partial}{\partial \theta_1} L(\theta^0) \\ \vdots \\ \frac{\partial}{\partial \theta_n} L(\theta^0) \end{pmatrix} = \begin{pmatrix} \frac{\boldsymbol{d}}{\boldsymbol{df}} \boldsymbol{L}\left(\boldsymbol{f}(\boldsymbol{\theta^0})\right) \cdot \frac{\boldsymbol{\partial}}{\boldsymbol{\partial \theta_0}} \boldsymbol{f}\left(\boldsymbol{x}; \boldsymbol{\theta^0}\right) \\ \frac{\boldsymbol{d}}{\boldsymbol{df}} \boldsymbol{L}\left(\boldsymbol{f}(\boldsymbol{\theta^0})\right) \cdot \frac{\boldsymbol{\partial}}{\boldsymbol{\partial \theta_1}} \boldsymbol{f}\left(\boldsymbol{x}; \boldsymbol{\theta^0}\right)) \\ \vdots \\ \frac{\boldsymbol{d}}{\boldsymbol{df}} \boldsymbol{L}\left(\boldsymbol{f}(\boldsymbol{\theta^0})\right) \cdot \frac{\boldsymbol{\partial}}{\boldsymbol{\partial \theta_n}} \boldsymbol{f}\left(\boldsymbol{x}; \boldsymbol{\theta^0}\right)) \end{pmatrix}$$

3. Update $\theta^{t+1} = \theta^t + \boldsymbol{A} \cdot \nabla L(\theta^t)$