5 Vorspiel 1: Lineare Modelle

5.1 Die einfache lineare Regression

Ein sehr wichtiges und einfaches Verfahren des maschinellen Lernens ist die lineare Regression. Hier wird versucht, eine Funktion mittels einer linearen Funktion zu approximieren. Etwas allgemeiner verwendet man lineare Regression für die Approximation mittels Polynomialfunktionen beliebigen Grades. Nehmen wir erstmal eine einfache lineare Funktion $f : \mathbb{R} \to \mathbb{R}$, die die Form haben soll:

$$f(x) = ax + b \tag{24}$$

wobei $a, b \in \mathbb{R}$ die Parameter sind. In diesem Fall muss unser Datensatz $D \subseteq \mathbb{R} \times \mathbb{R}$ einfach eine Abhängigkeit zweier reellwertiger Parameter darstellen. Wir setzen als Konvention

$$D = \{(a_1, b_1), ..., (a_n, b_n)\}$$

und für $x = a_i$ schreiben wir y_x für b_i, also der Wert den D x zuweist. Wir suchen nun diejenige lineare Funktion, die die Differenz minimiert, also

$$argmin_{a,b\in\mathbb{R}} \sum_{(x,y)\in D} (ax + b - y)^2 \tag{25}$$

Das Quadrat ist dafür da, dass Werte positiv werden – sonst würden sich negative und positive Abweichungen ausgleichen. Damit gewichten wir natürlich weitere Abweichungen stärker, was nicht unbedingt erwünscht ist; allerdings gibt es kaum andere Möglichkeiten: die Betragsfunktion $|-|$ ist nicht differenzierbar, wir brauchen allerdings die erste Ableitung der Funktion, wie wir unten sehen werden.

Wir machen nun einen Trick: eigentlich sind die Parameter a, b festgelegt, während x das variable Argument der Funktion ist. Weil wir aber nur an denjenigen x interessiert sind, die in unserem Datensatz auftauchen (d.h. endlich viele), während wir alle reellen Parameter berücksichten müssen. Daher ist die Funktion, die wir minimieren müssen, eigentlich folgende:

$$\sum_{(a,b)\in D} (ax + y - b)^2 = (a_1x + y - b_1)^2 + ... + (a_nx + y - b_n)^2 \tag{26}$$

Hier haben wir einfach die Konstanten und Variablen vertauscht, und daraufhin eine arithmetische Umformung vorgenommen. Am Ende bekommen wir die einfache Form (denn $a_1, ..., a_n, b_1, ...b_n$ sind einfache gegebene Konstanten), wir haben natürlich

$$(a_1x + y - b_1)^2 = a_1^1x^2 + y^2 + a_1xy - a_1b_1x - b_1y + b_1^2 \tag{27}$$

Diese Umformung machen wir für alle Summanden, und da alle dieselben Variablenformen aufweisen, können wir aufaddieren; bekommen also:

$$\sum_{(a,b)\in D} (ax + y - b)^2 = ax^2 + by^2 + cxy + dx + ey + f \tag{28}$$

Wobei $a = a_1^2 + ... + a_n^2$, $b = n$, ..., $f = b_1^2 + ... + b_n^2$. Wir suchen nun einfach

$$argmin_{x,y\in\mathbb{R}} ax^2 + by^2 + cxy + dx + ey + f \tag{29}$$

Das berechnet man mit der gewohnten Methode: wir bilden (in diesem Fall partielle) Ableitungen und konstruieren damit den Gradienten. Das ist natürlich besonders einfach:

$$\nabla f(x, y) = ((2ax + cy + d), (2by + cx + e)) \tag{30}$$

Wir haben also 2 Gleichungen, die wir auf 0 setzen müssen:

$$2ax + cy + d = 0 \tag{31}$$

$$2by + cx + e = 0 \tag{32}$$

Wir haben 2 Gleichungen und 2 Variablen, also eine Lösung:

$$x = -\frac{cy + d}{2a} \tag{33}$$

also

$$2by - \frac{c^2y + cd}{2a} + e = 0 \leftrightarrow \tag{34}$$

$$y = \frac{cd - 2ae}{4ab - c^2} \tag{35}$$

Wir haben also die Nullstelle für x, y berechnet, und wissen somit, wie wir die Funktion minimieren können.

5.2 Der komplexe lineare Fall

Im komplexeren lineare Fall nehmen wir an, dass

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + ... + a_0 \quad (36)$$

f ist also nun ein Polynom, keine lineare Funktion mehr. Warum ist das immer noch lineare Regression? Weil wir, bei der eigentlichen Regression, also der Suche nach den Parametern die die beste Funktion ausmachen, x als eine Konstante behandeln (wir setzen nämlich Datenpunkte ein, bekommen also einfach konstante Werte in $\mathbb{R}$), während die eigentlichen Variablen die Werte $a_0, ..., a_n$ sind. In diesen Werten ist die resultierende Funktion nach wie vor linear – wir haben also einen Fall der etwas komplexer ist als der vorhergehende, aber nach wie vor durch die Lösung linearer Gleichungssysteme lösbar ist.

$$f(x_0, ..., x_n) = \sum_{(a,b) \in D} (a^n x_n + a^{n-1} x_{n-1} + ... + x_0 - b)^2 \quad (37)$$

Hier sind a^n etc. und b fixe reelle Zahlen, während die Variablen nur linear auftreten. Wir müssen nun

$$\nabla f : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1} \quad (38)$$

konstruieren, kriegen also ein lineares Gleichungssystem mit $n+1$ Variablen und $n+1$ Gleichungen, das wir entsprechend lösen können. Lineare Regression lässt sich also mit elementaren mathematischen Methoden lösen, und das ist der große Vorteil dabei.

Eine weitere Erweiterung, die ohne große Probleme funktioniert, ist folgende: anstatt einer Funktion $f : \mathbb{R} \to \mathbb{R}$ suchen wir eine Funktion $f : \mathbb{R}^n \to \mathbb{R}$, die die Form hat

$$f(x_1, ..., x_n) = a_1 x_1 + a_2 x_2 + ... + a_n x_n + b \quad (39)$$

Auch das geht ohne große Probleme mit den obigen elementaren Methoden, und auch die Erweiterung auf Polynomialfunktionen (auch wenn natürlich alles etwas schwieriger wird).

5.3 Lineare Regression in R

In R gibt es ein einfaches Kommando zur (einfachen) lineaeren Regression, nämlich `lm`. Erstmal brauchen wir Daten; dazu nehmen wir eine Menge $D \subseteq \mathbb{R}^2$. In R geht das einfacher, wenn man zwei Vektoren nimmt: