Theorie des maschinellen Lernens

Hans U. Simon

25. Mai 2016

9 Lineare Voraussagefunktionen

Lineare Voraussagefunktionen sind wichtig, und zwar aus folgenden Gründen:

- Sie werden in Anwendungen oft benutzt.
- Sie führen zu effizient lösbaren Lernproblemen (mit gewissen Einschränkungen im agnostischen Fall).
- Sie sind der Intuition zugänglich und einfach zu interpretieren.
- Wenn sie geeignet eingesetzt werden, beschreiben sie oftmals die in Anwendungen vorliegenden Daten auf eine befriedigende Weise.

Im Folgenden sei

$$\mathrm{LIN}_d := \{x \mapsto h_{w,b}(x) : w \in \mathbb{R}^d, b \in \mathbb{R}\} \tag{1}$$

mit

$$h_{w,b} : \mathbb{R}^d \to \mathbb{R}\ ,\ h_{w,b}(x) = \langle w, x\rangle + b = \sum_{i=1}^{d} w_i x_i + b\ .$$

die Klasse der *affinen Funktionen*. Wir bezeichnen w als den *Gewichtsvektor* und b als das *Bias* der Abbildung $h_{w,b}$. Eine *lineare Voraussagefunktion* mit Werten in $\mathcal{Y}$ ergibt sich aus der Komposition einer Funktion $\phi : \mathbb{R} \to \mathcal{Y}$ mit einer affinen Funktion aus LIN_d:

$$\phi \circ h_{w,b} : \mathbb{R}^d \to \mathcal{Y}\ ,\ \phi(h_{w,b}(x)) = \phi(\langle w, x\rangle + b)\ .$$

Mögliche Wahlen von ϕ:

1. $\phi = \mathrm{sign} : \mathbb{R} \to \{-1, 1\}$ mit

   $$sign(a) = \begin{cases} +1 & \text{if } a \ge 0 \\ -1 & \text{if } a < 0 \end{cases}$$

   für alle $a \in \mathbb{R}$. Diese Wahl erfolgt im Zusammenhang mit binären Klassifikationsproblemen.

2. $\phi = \mathrm{id} : \mathbb{R} \to \mathbb{R}$ mit $\mathrm{id}(a) = a$ für alle $a \in \mathbb{R}$. Diese Wahl erfolgt im Zusammenhang mit Regressionsproblemen.

Es bezeichne $\mathrm{LIN}_d^0 = \{h_{w,0} : w \in \mathbb{R}^d\}$ die Klasse der *linearen Funktionen* (affine Funktionen mit Bias 0). Es gilt einerseits $\mathrm{LIN}_d^0 \subset \mathrm{LIN}_d$; andererseits können wir eine Funktion $h_{w,b} \in \mathrm{LIN}_d$ auch als Funktion in LIN_{d+1}^0 auffassen, indem wir folgende Einbettungen von $x, w \in \mathbb{R}^d$ in den $(d+1)$-dimensionalen Euklidischen Raum vornehmen:

$$x' = (1, x) = (1, x_1, \ldots, x_d) \neq \vec{0} \ \text{ und } \ w' = (b, w) = (b, w_1, \ldots, w_d) \ . \tag{2}$$

Offensichtlich gilt dann:

$$h_{w,b}(x) = \langle w, x \rangle + b = \langle w', x' \rangle = h_{w',0}(x') \ .$$

Wann immer wir das bequem finden, werden wir daher lineare Funktionen (angewendet auf vom Nullvektor verschiedene Vektoren) anstelle von affinen Funktionen betrachten, d.h., wir werden mitunter oBdA $b = 0$ setzen und $x \neq \vec{0}$ voraussetzen. Man spricht dann auch vom "homogenen Fall" (im Unterschied zum sogenannten"inhomogenen Fall" bei Betrachtung von allgemeinen affinen Funktionen).

9.1 Halbräume

Es bezeichne

$$H_{w,b} = \{x \in \mathbb{R}^d : h_{w,b}(x) = 0\} = \{x \in \mathbb{R}^d : \langle w, x \rangle + b = 0\}$$

die durch w, b gegebene affine Hyperebene im $\mathbb{R}^d$. Diese zerteilt den $\mathbb{R}^d$ in zwei Halbräume:

$$H_{w,b}^+ = \{x \in \mathbb{R}^d : \langle w, x \rangle + b \geq 0\} \ \ \text{(positiver Halbraum)} \ .$$
$$H_{w,b}^- = \{x \in \mathbb{R}^d : \langle w, x \rangle + b < 0\} \ \ \text{(negativer Halbraum)} \ .$$

Wir bezeichnen dann

$$\mathrm{HS}_d = \{H_{w,b}^+ : w \in \mathbb{R}^d, b \in \mathbb{R}\}$$

als die Klasse der *(affinen) Halbräume*. Alternativ können wir einen positiven Halbraum $H_{w,b}^+$ auch identifizieren mit der Abbildung $x \mapsto \mathrm{sign}(h_{w,b}(x))$, welche die Punkte im positiven Halbraum mit 1 und die im negativen Halbraum mit -1 markiert. Die Klasse der affinen Halbräume, aufgefasst als Klasse von ± 1-wertigen Funktionen, wäre dann gegeben durch

$$\mathrm{HS}_d = \mathrm{sign} \circ \mathrm{LIN}_d = \{x \mapsto \mathrm{sign}(\langle w, x \rangle + b) : w \in \mathbb{R}^d, b \in \mathbb{R}\} \ .$$

Wir werden Halbräume meist mit $H_{w,b}^+$ identifizieren (gelegentlich vielleicht aber auch mit $x \mapsto \mathrm{sign}(h_{w,b}(x))$).

Eine affine Hyperebene $H_{w,b}$ heißt *homogen*, falls $b = 0$ und *inhomogen* falls $b \neq 0$. Der Gewichtsvektor $w \in \mathbb{R}^d$ wird auch als der *Normalenvektor* von $H_{w,b}$ bezeichnet. Aus der linearen Algebra ist bekannt, dass folgendes gilt:

Homogener Fall: $x \in H_{w,0}$ (bzw. $x \in H^+_{w,0} \setminus H_{w,0}$ oder $x \in H^-_{w,0}$) genau dann, wenn x und w einen rechten Winkel (bzw. einen spitzen oder einen stumpfen) Winkel miteinander bilden.

Inhomogener Fall: Es sei $a \in H_{w,0}$ ein beliebiger Punkt auf der affinen Hyperebene $H_{w,b}$. Dann gilt

$$H_{w,b} = a + H_{w,0} \ , \ H^+_{w,b} = a + H^+_{w,0} \ \text{ und } H^-_{w,b} = a + H^-_{w,0} \ .$$

Somit gilt $a + x \in H_{w,b}$ (bzw. $a + x \in H^+_{w,b} \setminus H_{w,b}$ oder $a + x \in H^-_{w,b}$) genau dann, wenn x und w einen rechten Winkel (bzw. einen spitzen oder einen stumpfen) Winkel miteinander bilden.

9.1.1 Informationskomplexität der Klasse der Halbräume

Wir wissen, dass die Anzahl zum (evtl. agnostischen) PAC-Lernen einer Hypothesenklasse $\mathcal{H}$ benötigten Trainigsbeispiele proportiomal zur VC-Dimension von $\mathcal{H}$ ist. In diesem Abschnitt werden wir zeigen, dass die VC-Dimension von HS_d exakt $d + 1$ beträgt und somit eine lediglich lineare Abhängigkeit in dem „Komplexitätsparameter“ d aufweist.

Es bezeichne $\mathrm{HS}^0_d = \{H^+_{w,0} : w \in \mathbb{R}^d\}$ die Klasse der homogenen Halbräume. Wir notieren mit $\vec{e}_i$ den Vektor mit einer 1 in Position i und Nullen in allen anderen Positionen. Im Zusammenhang mit Halbräumen benutzen wir im Folgenden die Binärlabels $-1, 1$ (anstelle der Labels $0, 1$).

Lemma 9.1 *Eine Punktmenge $S \subseteq \mathbb{R}^d$ ist aufspaltbar durch HS^0_d genau dann, wenn S aus linear unabhängigen Vektoren besteht.*

Beweis Wir beginnen mit der Beweisrichtung „$\Leftarrow$“. Wir dürfen annehmen, dass S aus d linear unabhängigen Vektoren $x_1, \ldots, x_d \in \mathbb{R}^d$ besteht (da wir kleinere Mengen S zu einer Basis ergänzen können). Es sei A die $(d \times d)$-Matrix mit $x_1^\top, \ldots, x_d^\top$ als Zeilenvektoren. Da A den Rang d hat, existiert die inverse Matrix A^{-1}. Dann ist das Gleichungssystem $Aw = y$ für jede Wahl von $y \in \mathbb{R}^d$ lösbar, nämlich mit $w = A^{-1}y$. Dies gilt insbesondere für alle $y \in \{-1, 1\}^d$. Daher ist S durch HS^0_d aufspaltbar.
Die Beweisrichtung „$\Rightarrow$“ zeigen wir indirekt. Wir nehmen an, dass S aus linear abhängigen Vektoren $x_1, \ldots, x_m$ besteht und zeigen, dass dann nicht alle Binärmuster durch HS^0_d realisierbar sind. Wegen der linearen Abhängigkeit existiert ein Vektor $a \in \mathbb{R}^m \setminus \{\vec{0}\}$ mit $\sum_{i=1}^m a_i x_i = 0$. Betrachte die zwei Indexmengen $I = \{i \in [m] : a_i > 0\}$ und $J = \{j \in [m] : a_j < 0\}$, von denen mindestens eine nichtleer ist.

Fall 1: $J = \emptyset$.

Dann folgt $I \neq \emptyset$ und $\sum_{i \in I} a_i x_i = 0$. Nimm an, dass das Bitmuster $y_i = -1$ für alle $i \in I$ realisierbar ist, sagen wir mit dem Gewichtsvektor w. Wir erhalten einen Widerspruch wie folgt:

$$0 > \sum_{i \in I} a_i \langle w, x_i \rangle = \left\langle w, \sum_{i \in I} a_i x_i \right\rangle = \left\langle w, \vec{0} \right\rangle = 0 \ .$$

Fall 2: $J \neq \emptyset$.

Die Gleichung $\sum_{i=1}^{m} a_i x_i = 0$ lässt sich umschreiben zu $\sum_{i \in I} a_i x_i = \sum_{j \in J} |a_j| x_j$. Nimm an, dass das Binärmuster $y_i = 1$ für alle $i \in I$ und $y_j = -1$ für alle $j \in J$ realisierbar ist, sagen wir durch den Gewichtsvektor w. Wir erhalten einen Widerspruch wie folgt:

$$0 \leq \sum_{i \in I} a_i \langle w, x_i \rangle = \left\langle w, \sum_{i \in I} a_i x_i \right\rangle = \left\langle w, \sum_{j \in J} |a_j| x_j \right\rangle = \sum_{j \in J} |a_j| \langle w, x_j \rangle < 0 \ . \qquad (3)$$

Damit ist der Beweis des Lemmas abgeschlossen. **qed.**

Lemma 9.2 *Eine Punktmenge $S \subseteq \mathbb{R}^d$ ergänzt um den Nullvektor $\vec{0}$ ist durch HS_d aufspaltbar genau dann, wenn S aus linear unabhängigen Vektoren besteht.*

Beweis Aus der in (2) beschriebenen Darstellung affiner Funktionen aus LIN_d als lineare Funktionen aus LIN_{d+1}^0 folgt leicht: $S \cup \{\vec{0}\}$ ist aufspaltbar durch HS_d genau dann, wenn $S' := \{(1, x) : x \in S\} \cup \{(1, \vec{0})\}$ aufspaltbar durch HS_{d+1}^0 ist. Gemäß Lemma 9.1 ist dies der Fall genau dann, wenn S' aus linear unabhängigen Vektoren besteht. Dies wiederum gilt genau dann, wenn die ursprüngliche Menge S aus linear unabhängigen Vektoren besteht. **qed.**

Theorem 9.3 *VCdim*$(\mathrm{HS}_d^0) = d$ *und* *VCdim*$(\mathrm{HS}_d) = d + 1$.

Beweis $\mathrm{VCdim}(\mathrm{HS}_d^0) = d$ ergibt sich unmittelbar aus Lemma 9.1. Wir wissen bereits, dass HS_d sich als eine Unterklasse von HS_{d+1}^0 auffassen lässt. Daraus folgt $\mathrm{VCdim}(\mathrm{HS}_d) \leq \mathrm{VCdim}(\mathrm{HS}_{d+1}^0) = d + 1$. Aus Lemma 9.2 folgt, dass $\mathrm{VCdim}(\mathrm{HS}_d) \geq d + 1$, da jede Basis von $\mathbb{R}^d$ ergänzt um den Nullvektor durch HS_d aufspaltbar ist. **qed.**

Punkte x mit $\langle w, x \rangle + b = 0$ erhalten vom Halbraum $H_{w,b}$ das Label 1, da $\mathrm{sign}(0) = 1$. Es wäre befriedigender, wenn wir garantieren könnten, dass positive Labels nur aus positiven Werten von $\langle w, x \rangle + b$ resultieren. Dies kann man in der Tat auf endlichen Punktmengen immer erreichen, wie die folgenden Ausführungen zeigen werden.

Definition 9.4 *Es seien $w, x \in \mathbb{R}^d$ und $b \in \mathbb{R}$. Die Größe $|\langle w, x \rangle + b|$ heißt der* (vorzeichenlose) funktionale Randabstand *der affinen Hyperebene $H_{w,b}$ vom Punkt x.*

Beachte, dass jeder von Null verschiedene funktionale Randabstand beliebig vergrößert werden kann, indem w und b um den gleichen positiven Faktor $\lambda > 0$ hochskaliert werden. Dabei bleibt das Vorzeichen unverändert: $\mathrm{sign}(\langle w, x \rangle + b) = \mathrm{sign}(\langle \lambda w, x \rangle + \lambda b)$.

Lemma 9.5 *Für eine endliche Punktmenge S in $\mathbb{R}^d$ gilt folgendes. Wenn ein Binärmuster auf den Punkten aus S mit einem affinen Halbraum realisierbar ist, dann ist dies bei geeigneter Wahl von (w, b) auch so möglich, dass $H_{w,b}$ einen funktionalen Randabstand von mindestens* 1 *zu jedem der Punkte aus S aufweist.*

Beweis Betrachte eine affine Hyperebene H, die ein vorgegebenes Binärmuster realisiert. Sie hat zu allen nicht auf ihr liegenden Punkten aus S einen echt positiven Randabstand. Durch hinreichend kleine Parallelverschiebung können wir erreichen, dass einerseits zu allen Punkten aus S ein positiver funktionaler Randabstand entsteht, aber andererseits das realisierte Binärmuster nicht verändert wird. Sind alle funktionalen Randabstände verschieden von 0, dann können wir sie durch Skalierung beliebig vergrößern. **qed.**

9.1.2 Die Berechnungskomplexität der Klasse der Halbräume

Beim *uniformen Kostenmaß* wird eine Zahl als eine Einheit gesehen und eine arithmetisch-logische Grundoperation auf zwei Zahlen zählt als 1 Rechenschritt. Beim *logarithmischen Kostenmaß* wird die Länge einer Zahl mit der Anzahl der Bits in ihrer Binärkodierung gemessen (was auch bei der Bestimmung der Eingabelänge eine Rolle spielt) und die Kosten einer arithmetisch-logischen Operation auf zwei Zahlen werden gleich gesetzt mit der Summe ihrer Bitlängen (d.h. im Wesentlichen: es wird die Anzahl der benötigten Bitoperationen gezählt).

Wir werden in diesem Abschnitt zeigen, dass das Konsistenzproblem für die Klasse $\mathcal{H}_d$ effizient (in Anhängigkeit vom Komplexitätsparameter d) lösbar ist, wenn wir $\mathbb{Q}^d$ (statt $\mathbb{R}^d$) als den Grundbereich ansehen und das logarithmische Kostenmaß zugrunde legen. Wir nutzen dabei aus, dass das Problem der linearen Programmierung — Optimierung einer linearen Zielfunktion unter linearen Randbedingungen vom Typ „$\geq$“ oder „$\leq$“ — effizient lösbar ist (zumindest bei Zugrundelegung des logarithmischen Kostenmaßes).[1]

Ein Spezialfall von linearer Programmierung ist das Problem zu entscheiden, ob vorgegebene lineare Randbedingungen (in Variablen mit Werten in $\mathbb{Q}$) simultan erfüllt werden können. Wir werden im Beweis zu folgendem Satz zeigen, dass sich das Konsistenzproblem für affine Halbräume als ein solches Erfüllbarkeitsproblem darzustellen lässt.

Theorem 9.6 *Das Konsistenzproblem zur Klasse der affinen Halbräume (über dem Grundbereich $(\mathbb{Q}^d)_{d\geq 1}$) ist ein Teilproblem von linearer Programmierung und daher (bezüglich des logarithmischen Kostenmaßes) effizient lösbar.*

Beweis Das Konsistenzproblem zu $(\mathrm{HS}_d)_{d\geq 1}$ ist, wie wir wissen, folgendes Problem:

Eingabeinstanz: zwei endliche Punktmengen $S^+, S^- \subset \mathbb{Q}^d$ (wobei der obere Index „+“ bzw. „$-$“ die binäre Markierung anzeigt)

Frage: Existieren $w \in \mathbb{Q}^d$ und $b \in \mathbb{Q}$, so dass $S^+ \subseteq \mathrm{HS}^+_{w,b}$ und $S^- \subseteq \mathrm{HS}^-_{w,b}$?

[1] Die Frage, ob lineare Programmierung auch bezüglich uniformem Kostenmaß in Polynomialzeit lösbar ist, ist völlig offen. Bei logarithmischem Kostenmaß gibt es Polynomialzeitalgorithmen (zum Beispiel der Ellipsoid-Algorithmus oder „Interior-Point“-Methoden).

Mit Lemma 9.5 ergibt sich, dass folgende Aussagen äquivalent sind:

$$\begin{array}{rl} \exists w \in \mathbb{Q}^d, b \in \mathbb{Q}: & (S^+ \subseteq \mathrm{HS}^+_{w,b}) \wedge (S^- \subseteq \mathrm{HS}^-_{w,b}) \\ \exists w \in \mathbb{Q}^d, b \in \mathbb{Q}: & (\forall x \in S^+ : \langle w, x\rangle + b \geq 0) \wedge (\forall x \in S^- : \langle w, x\rangle + b < 0) \\ \exists w' \in \mathbb{Q}^d, b' \in \mathbb{Q}: & (\forall x \in S^+ : \langle w', x\rangle + b' \geq 1) \wedge (\forall x \in S^- : \langle w', x\rangle + b' \leq -1) \end{array}$$

Wir landen somit bei der Frage, ob wir Werte an die Variablen $w = (w_1, \ldots, w_d)$ und b so zuweisen können, dass die linearen Randbedingungen

$$(\forall x \in S^+ : \langle w, x\rangle + b \geq 1) \wedge (\forall x \in S^- : \langle w, x\rangle + b \leq -1)$$

simultan erfüllt sind (Spezialfall von linearer Programmierung). **qed.**

Wir merken kurz an, dass die Punktmengen S^+, S^- *linear separierbar* heißen, falls eine sie trennende Hyperebene $H_{w,b}$ im Sinne von

$$(\forall x \in S^+ : \langle w, x\rangle + b > 0) \wedge (\forall x \in S^- : \langle w, x\rangle + b < 0) \qquad (4)$$

existiert. Die „Realizability Assumption“ des PAC-Lernmodells entspricht also der Voraussetzung, dass die gegebenen Mengen S^+, S^- linear separierbar sind. Beim agnostischen PAC-Lernmodell befinden wir uns dagegen dann im nicht-separierbaren Fall. Es lässt sich zeigen (hier ohne Beweis), dass das Problem MinDis-E(HS) NP-hart ist.

Effiziente Lösbarkeit des Konsistenzproblems und NP-Härte von „Minimum Disagreement“ haben Konsequenzen:

Theorem 9.7 *1. Die Klasse der affinen Halbräume ist effizient PAC-lernbar.*

2. Die Klasse der affinen Halbräume ist nicht effizient agnostisch PAC-lernbar (außer wenn RP = NP).

9.2 Das Rosenblatt’sche Perzeptron

Es sei $(x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^d \times \{-1, 1\}$ eine Folge von markierten Trainigsbeispielen. Die Menge $\{x_1, \ldots, x_m\}$ zerfällt dann in $S^+ = \{x_i : y_i = 1\}$ und $S^- = \{x_i : y_i = -1\}$. Die Bedingung (4) für eine S^+ und S^- linear separierende Hyperebene lässt sich mit Hilfe der Labels y_i auch schreiben wie folgt:

$$\forall i = 1, \ldots, m : y_i(\langle w, x_i\rangle + b) > 0 \ . \qquad (5)$$

Im Jahre 1958 publizierte Rosenblatt einen Algorithmus, genannt Perzeptron, welcher w, b iterativ modifiziert, solange die Bedingung (5) verletzt ist. Wir beschreiben diesen Algorithmus der Einfachheit halber für den homogenen Fall (mit $b = 0$), wobei wir voraussetzen dürfen, dass alle x_i vom Nullvektor verschieden sind.[2]

[2] Vergleiche mit der Reduktion des d-dimensionalen inhomogenen Falles auf den $(d + 1)$-dimensionalen homogenen Fall.

Initialisierung: Setze $w^{(1)} = \vec{0}$.

Hauptschleife: Für $t = 1, 2, 3, \ldots$ mache folgendes:
Falls ein $i \in [m]$ mit $y_i\langle w, x_i\rangle \leq 0$ existiert, dann setze

$$w^{(t+1)} = w^{(t)} + y_i x_i \quad . \tag{6}$$

Falls nicht, dann gibt $w^{(t)}$ aus und stoppe.

Die Aktualisierung (6) arbeitet (im Sinne der Bedingung (5)) in die richtige Richtung, da

$$y_i\left\langle w^{(t+1)}, x_i\right\rangle = y_i\left\langle w^{(t)} + y_i x_i, x_i\right\rangle = y_i\left\langle w^{(t)}, x_i\right\rangle + \langle x_i, x_i\rangle$$

und für $x_i \neq \vec{0}$ gilt $\langle x_i, x_i\rangle > 0$.

Das folgende Resultat zeigt, dass das Perzeptron nach endlich vielen Iterationen eine separierende Hyperebene findet (sofern sie existiert):

Theorem 9.8 *Wir setzen den separablen Fall voraus. Es sei*

$$\begin{aligned} B &= \min\{\|w\| : y_i\langle w, x_i\rangle \geq 1 \text{ für alle } i \in [m]\} \ , \\ R &= \max\{\|x_i\| : i \in [m]\} \ . \end{aligned}$$

Sei T die Anzahl der Iterationen, die (6) durchlaufen. Mit diesen Notationen gilt: $T \leq R^2B^2$ und der ausgegebene Gewichtsvektor $w^{(T+1)}$ erfüllt die Separabilitätsbedingung (5).

Beweis Das Perzeptron steigt aus der Hauptschleife nur dann aus, wenn die Separabilitätsbedingung erfüllt ist. Es genügt also zu zeigen, dass die Anzahl T der Aktualisierungen des Gewichtsvektors durch R^2B^2 nach oben beschränkt ist. Der Parameter B (s. oben) gibt die kleinstmögliche Euklidische Länge eines Gewichtsvektors an, der die Bedingung (5) mit einem funktionalen Randabstand von mindestens 1 erfüllt. Es sei w^* ein Gewichtsvektor mit $\|w^*\| = B$, der das leistet. Unser Ziel ist der Nachweis von folgender Bedingung:

$$1 \geq \underbrace{\frac{\left\langle w^*, w^{(T+1)}\right\rangle}{\|w^*\| \cdot \|w^{(T+1)}\|}}_{=\cos(w^*, w^{(T+1)})} \geq \frac{\sqrt{T}}{RB} \quad . \tag{7}$$

Wenn wir die Ungleichungen in (7) nach T auflösen, ergibt sich sofort $T \leq R^2B^2$. Die erste Ungleichung in (7) ergibt sich aus „Cauchy-Schwartz". Der Nachweis der zweiten Ungleichung in (7) ergibt sich aus folgenden Rechnungen:

1. $\left\langle w^*, w^{(t+1)}\right\rangle - \left\langle w^*, w^{(t)}\right\rangle = \langle w^*, y_i x_i\rangle = y_i\langle w^*, x_i\rangle \geq 1$.
   Mit $\left\langle w^*, w^{(1)}\right\rangle = \left\langle w^*, \vec{0}\right\rangle = 0$ ergibt sich induktiv $\left\langle w^*, w^{(T+1)}\right\rangle \geq T$.

2. $\|w^{(t+1)}\|^2 = \|w^{(t)} + y_i x_i\|^2 = \|w^{(t)}\|^2 + \underbrace{2y_i\left\langle w^{(t)}, x_i\right\rangle}_{\le 0} + \|x_i\|^2 \le \|w^{(t)}\|^2 + R^2$.

   Mit $\|w^{(1)}\|^2 = \|\vec{0}\|^2 = 0$ ergibt sich hieraus induktiv $\|w^{(T+1)}\|^2 \le TR^2$ und somit $\|w^{(T+1)}\| \le \sqrt{T}R$.

Aus $\left\langle w^*, w^{(T+1)}\right\rangle \ge T$, $\|w^*\| = B$ und $\|w^{(T+1)}\| \le \sqrt{T}R$ folgt unmittelbar

$$\frac{\left\langle w^*, w^{(T+1)}\right\rangle}{\|w^*\| \cdot \|w^{(T+1)}\|} \ge \frac{T}{B\sqrt{T}R} = \frac{\sqrt{T}}{RB} \ ,$$

was den Beweis abschließt. **qed.**

Im „worstcase“ können B und R exponentiell von der Eingabelänge (in binärer Kodierung) abhängen. I.A. ist also das Perzeptron kein effizienter Algorihmus. Wenn jedoch die Parameter R und B nur moderat groß sind (was bei vielen Anwendungen der Fall sein mag), dann ist die obere Schranke R^2B^2 gar nicht so übel.

9.3 Lineare Regression

In diesem Abschnitt operieren wir mit dem Grundbereich $\mathcal{X} = \mathbb{R}^d$ und der Labelmenge $\mathcal{Y} = \mathbb{R}$. Eine Trainingssequenz hat daher die Form

$$S = [(x_1, y_1), \ldots, (x_m, y_m)] \in (\mathbb{R}^d \times \mathbb{R})^m \ . \tag{8}$$

Wir betrachten die Hypothesenklasse LIN_d der affinen Funktionen gemäß (1) und die quadratische Verlustfunktion

$$\ell(h, (x, y)) = (h(x) - y)^2 \ .$$

Es sei D eine Verteilung auf $\mathbb{R}^d \times \mathbb{R}$. Wie üblich setzen wir uns das Ziel, eine Hypothese $h \in \mathrm{LIN}_d$ mit möglichst kleinem Wert von $L_D(h)$ (dem erwarteten quadratischen Fehler von h bezüglich D) zu finden.

Warnung: Die auf VC-Dimension aufbauende Theorie ist nicht unmittelbar auf nicht-binäre Voraussageprobleme anwendbar. Evtl. werden wir zu einem späteren Zeitpunkt der Vorlesung klären, wie groß Trainingsmengen bei Regressionsproblemen (oder allgemeiner bei nicht-binären Voraussageproblemen) zu sein haben.

Beispiel 9.9 *Das Erraten des Gewichtes eines Babys anhand seines Geburtsgewichtes und seines Alters (auf Basis einer gegebenen Trainingsmenge) ist ein* 2*-dimensionales Regressionsproblem.*

Wir betrachten im Folgenden die ERM-Lernregel: gegeben eine Trainingsmenge S, finde eine Hypothese $h \in \mathrm{LIN}_d$ mit möglichst kleinem Wert von $\hat{L}_S(h)$ (dem mittleren quadratischen Fehler von h auf der Trainingsmenge S). Im Falle $d = 1$ wird eine der ERM-Lernregel entsprechende Hypothese als „Ausgleichsgrade“ bezeichnet. Die Methode zum Berechnen einer Hypothese h gemäß der ERM-Lernregel bei beliebig hoher Dimension d ist unter dem Namen „Least Squares“ bekannt.

9.3.1 Die „Least Squares“-Methode

Wir beschränken uns der Einfachheit halber wieder auf den homogenen Fall (mit $b = 0$) und betrachten das Problem

$$\mathrm{argmin}_{w \in \mathbb{R}^d} \hat{L}_S(h_{w,0}) = \mathrm{argmin}_{w \in \mathbb{R}^d} \left\{ \frac{1}{m} \cdot \sum_{i=1}^{m} \left(\langle w, x_i \rangle - y_i \right)^2 \right\} \ .$$

Wenn wir den Gradienten der Zielfunktion auf Null setzen, erhalten wir die Gleichung

$$\frac{2}{m} \sum_{i=1}^{m} (\langle w, x_i \rangle - y_i) x_i = 0 \ .$$

Dies ist äquivalent zu

$$0 = \sum_{i=1}^{m} x_i (x_i^\top w - y_i) = \left(\sum_{i=1}^{m} x_i x_i^\top \right) w - \sum_{i=1}^{m} y_i x_i \ .$$

Wenn wir

$$A = \sum_{i=1}^{m} x_i x_i^\top \ \text{ und } \ b = \sum_{i=1}^{m} y_i x_i \tag{9}$$

setzen, haben wir also das lineare Gleichungssystem $Aw = b$ zu lösen.

Beachte: $x_i \in \mathbb{R}^d$ ist ein Spaltenvektor und, als Matrix aufgefasst, eine $(d \times 1)$-Matrix. Somit ist $x_i^\top$ eine $(1 \times d)$-Matrix und $x_i x_i^\top$ (sowie die Matrix A) eine $(d \times d)$-Matrix. Der Eintrag in Zeile j und Spalte k von $x_i x_i^\top$ ist gleich $x_i(j)x_i(k)$, also die j-te Komponente des Vektors x_i multipliziert mit seiner k-ten Komponente.

Setze $r = \mathrm{rank}(A)$. Ein sympathisch einfacher Fall wäre $r = d$. Dann besitzt A eine Inverse A^{-1} und $w = A^{-1}b$ wäre eine optimale Wahl von w. Wir wollen uns im Folgenden aber mit dem interessanteren Fall $r < d$ beschäftigen.

Mit $\mathrm{span}(A)$ bezeichnen wir den Vektorraum, der von den Spaltenvektoren von A aufgespannt wird. Das Gleichungssystem $Aw = b$ ist genau dann lösbar, wenn $b \in \mathrm{span}(A)$. Dies ist bei unserer Anwendung glücklicherweise der Fall:

Lemma 9.10 *Wenn A und b gemäß (9) gewählt werden, dann gilt $b \in \mathrm{span}(A)$.*

Den Beweis des Lemmas holen wir weiter unten nach.

Das folgende Resultat klärt, wie wir im Falle der Lösbarkeit an eine Lösung w des Gleichungssystems $Aw = b$ gelangen.

Lemma 9.11 *Falls $b \in \mathrm{span}(A)$, dann existiert eine Matrix $A^\dagger$ (genannt das* Pseudo-Inverse *zu A) mit*

$$A(A^\dagger b) = (AA^\dagger)b = b \ .$$

Weiterhin läßt sich $A^\dagger$ aus A und b effizient berechnen.

Auch den Beweis dieses Lemmas holen wir weiter unten nach.

Im Falle $r = d$ stimmt das Pseudo-Inverse $A^\dagger$ von A mit der Inversen A^{-1} überein. Wir erhalten somit das Hauptresultat dieses Abschnittes:

Theorem 9.12 *S sei gemäß (8) und A, b seien gemäß (9) gegeben. Dann ist $w = A^\dagger b$ ein der ERM-Lernregel (bezüglich quadratischem Fehler) entsprechender Gewichtsvektor. Er lässt sich effizient aus S ermitteln.*

Am Ende dieses Abschnittes holen wir die zwei fehlenden Beweise nach.

Beweis[Lemma 9.10] Der Beweis benutzt die aus der linearen Algebra bekannte Tatsache, dass jede (reelle) Matrix M die Gleichung $\mathrm{rank}(MM^\top) = \mathrm{rank}(M)$ erfüllt. Es ist leicht nachzurechnen, dass für $M = [x_1 \dots x_m]$ (also die Matrix mit den Spaltenvektoren $x_1, \dots, x_m$) die Gleichung

$$\sum_{i=1}^{m} x_i x_i^\top = MM^\top$$

gilt. Da $A = \sum_{i=1}^m x_i x_i^\top = MM^\top$ und da die Spaltenvektoren der Matrix $MM^\top$ Linearkombinationen von M's Spaltenvektoren sind, können wir $\mathrm{span}(A) \subseteq \mathrm{span}(M)$ folgern. Wegen $A = MM^\top$ haben A und M den gleichen Rang. Daher gilt sogar $\mathrm{span}(A) = \mathrm{span}(M)$. Wegen $b = \sum_{i=1}^m y_i x_i$ und $M = [x_1 \dots x_m]$ gilt $b \in \mathrm{span}(M)$ und damit auch $b \in \mathrm{span}(A)$. **qed.**

Beweis[Lemma 9.11] Der Beweis benutzt das aus der linearen Algebra bekannte Spektraltheorem, welches folgendes besagt. Jede symmetrische $(d \times d)$-Matrix A vom Range r (wie zum Beispiel $A = \sum_{i=1}^m x_i x_i^\top$) besitzt eine (effizient berechenbare) Zerlegung der Form

$$A = VDV^\top \quad \text{mit} \quad D = \mathrm{diag}(\lambda_1, \dots, \lambda_r, 0, \dots, 0) \ .$$

Hierbei gilt:

1. $V = [v_1, \dots v_d]$ bezeichnet eine orthogonale Matrix (d.h., $VV^\top = V^\top V = I$ mit $I =$ Einheitsmatrix) und es gilt $\mathrm{span}(A) = \mathrm{span}(v_1, \dots, v_r)$.
2. $\mathrm{diag}(\dots)$ bezeichnet eine Diagonalmatrix mit den in „…“ aufgelisteten Werten auf der Hauptdiagonalen und $\lambda_1, \dots, \lambda_r$ sind die r nicht-trivialen (also von 0 verschiedenen) reellen Eigenwerte von A.

Wir setzen nun $D^\dagger = \mathrm{diag}(\lambda_1^{-1}, \dots, \lambda_r^{-1}, 0, \dots, 0)$, $A^\dagger = VD^\dagger V^\top$ und rechnen nach:

$$AA^\dagger b = VDV^\top VD^\dagger V^\top b = VDD^\dagger V^\top b = V\mathrm{diag}(1, \dots, 1, 0, \dots, 0)V^\top b = \sum_{i=1}^{r} v_i v_i^\top b = \sum_{i=1}^{r} \langle v_i, b\rangle v_i \ .$$

Der letzte Ausdruck beschreibt die Projektion von b auf $\mathrm{span}(v_1, \dots, v_r) = \mathrm{span}(A)$. Da wir $b \in \mathrm{span}(A)$ vorausgesetzt haben, muss diese Projektion mit b übereinstimmen. **qed.**

9.3.2 Reduktion von polynomieller auf lineare Regression

Definition 9.13 *Es seien $\mathcal{H} = (\mathcal{H}_n)_{n\geq 1}$ und $\mathcal{H}' = (\mathcal{H}'_n)_{n\geq 1}$ zwei parameterisierte Hypothesenklassen reellwertiger Funktionen; $\mathcal{X} = (\mathcal{X}_n)_{n\geq 1}$ und $\mathcal{X}' = (\mathcal{X}'_n)_{n\geq 1}$ seien die zugehörigen Grundbereiche. Wir sagen, dass das Regressionsproblem für $\mathcal{H}$* reduzierbar *ist auf das Regressionsproblem für $\mathcal{H}'$, notiert als $\mathcal{H} \longrightarrow \mathcal{H}'$, wenn Abbildungen*

$$n \mapsto n' \ , \ \mathcal{X}_n \ni x \mapsto x' \in \mathcal{X}'_{n'} \ , \ \mathcal{H}_n \ni h \mapsto h' \in \mathcal{H}'_{n'} \ \text{ und } \ \mathcal{H}'_{n'} \ni h' \mapsto h \in \mathcal{H}_n$$

mit folgenden Eigenschaften existieren:

1. *n' ist polynomiell in n beschränkt.*
2. *Die Abbildungen $\mathcal{X}_n \ni x \mapsto x' \in \mathcal{X}'_{n'}$ und $\mathcal{H}'_{n'} \ni h' \mapsto h \in \mathcal{H}_n$ sind effizient berechenbar.*
3. *Für alle $n \in \mathbb{N}$, alle $x \in \mathcal{X}_n$, und alle $h \in \mathcal{H}_n$ gilt $h(x) = h'(x')$. Dabei bezeichnen x' bzw. h' die Objekte, die sich aus der Instanzentransformation $x \mapsto x'$ bzw. der Hypothesentransformation $h \mapsto h'$ ergeben.*
4. *Für alle $n \in \mathbb{N}$, alle $x \in \mathcal{X}_n$, und alle $h' \in \mathcal{H}'_{n'}$ gilt $h(x) = h'(x')$. Dabei bezeichnen x' bzw. h die Objekte, die sich aus der Instanzentransformation $x \mapsto x'$ bzw. der Hypothesentransformation $h' \mapsto h$ ergeben.*

Theorem 9.14 *Unter der Voraussetzung $\mathcal{H} \longrightarrow \mathcal{H}'$ gilt folgendes:*

1. *Wenn das Regressionsproblem zur Klasse $\mathcal{H}'$ effizient lösbar ist (im Sinne des üblichen (ε, δ)-Kriteriums), so gilt dies auch für das Regressionsproblem zur Klasse $\mathcal{H}$.*
2. *Wenn die ERM-Lernregel für $\mathcal{H}'$ effizient implementiert werden kann (Auffinden einer Hypothese aus $\mathcal{H}'$ mit kleinstmöglichem quadratischen Fehler auf den Trainingsdaten), so gilt dies entsprechend auch für die Klasse $\mathcal{H}$.*

Beweis Wir beschränken uns auf den Beweis der zweiten Aussage. Es sei

$$S = (x_1, y_1), \ldots, (x_m, y_m) \in (\mathcal{X}_n \times \mathbb{R})^m$$

eine gegebene Sequenz von Trainingsdaten. Zum Auffinden einer ERM-Hypothese aus $\mathcal{H}$ gehe vor wie folgt:

1. Berechne $S' = (x'_1, y_1), \ldots, (x'_m, y_m) \in (\mathcal{X}'_{n'} \times \mathbb{R})^m$. Dies ist eine m-fache Anwendung der Instanzentransformation $x \mapsto x'$.
2. Berechne eine ERM-Hypothese $h' \in \mathcal{H}'_{n'}$ zu der Trainingsmenge S'.
3. Berechne die Hypothese h, die sich aus h' gemäß der Hypothesentransformation $h' \mapsto h$ ergibt.

4. Gib h als ERM-Hypothese zur Trainingsmenge S aus.

Fixiere eine beliebige Hypothese $h_* \in \mathcal{H}_n$. Wir haben $\hat{L}_S(h) \le \hat{L}_S(h_*)$ zu zeigen. Dies geht mit einem „Schach Matt“ in drei Zügen:

- Es sei h'_* die Hypothese, die sich über $h_* \mapsto h'_*$ aus h_* ergibt. Da $h_*(x_i) = h'_*(x'_i)$ und da S' die binären Labels von S übernommen hat, folgt $\hat{L}_{S'}(h'_*) = \hat{L}_S(h_*)$.
- Da h' eine ERM-Hypothese zu S' ist, folgt $\hat{L}_{S'}(h') \le \hat{L}_{S'}(h'_*)$.
- Da h sich aus h' über $h' \mapsto h$ ergibt, gilt wieder $h(x_i) = h'(x'_i)$, was $\hat{L}_S(h) = \hat{L}_{S'}(h')$ zur Folge hat.

Wenn wir die drei Beobachtungen zusammensetzen, erhalten wir $\hat{L}_S(h) = \hat{L}_{S'}(h') \le \hat{L}_{S'}(h'_*) = \hat{L}_S(h_*)$, was den Beweis abschließt. **qed.**

Es bezeichne POL_d die Klasse der Polynome in d Variablen über $\mathbb{R}$, d.h.,

$$\mathrm{POL}_d = \{x \mapsto p(x) : p \in \mathbb{R}[X_1, \ldots, X_d]\} \ .$$

Weiter bezeichne POL_d^n die Teilklasse der Polynome vom Maximalgrad n. Der Grad eines Monoms der Form $X_1^{n_1} \ldots X_d^{n_d}$ ist per Definition gleich $n_1 + \ldots + n_d$ (und der Grad eines Polynoms ist das Maximum der Grade seiner Monome).

Lemma 9.15 *Es gibt $\binom{n+d}{d}$ Monome vom Maximalgrad n in d Variablen.*

Beweis Es gibt genau $\binom{n+d}{d}$ geordnete Partitionen der Zahl n in $d+1$ Summanden (eine aus der Vorlesung über „Diskrete Mathematik“ bekannte Tatsache). Da diese Zahlpartitionen und die Monome vom Maximalgrad n in d Variablen sich 1-zu-1 entsprechen, ergibt sich die Aussage des Lemmas. **qed.**

Beispiel 9.16 *Es sei $n = 6$ und $d = 3$. Das Monom $X_1 X_3^2 = 1^3 X_1^1 X_2^0 X_3^2$ entspricht der geordneten Partition von $n = 6$ in die Summanden $(3, 1, 0, 2)$, und umgekehrt.*

Theorem 9.17 *Es sei d eine beliebige aber fest gewählte Konstante,* $\mathrm{LIN} = (\mathrm{LIN}_n)_{n \ge 1}$ *und* $\mathrm{POL} = (\mathrm{POL}_d^n)_{n \ge 1}$. *Mit diesen Bezeichnungen gilt:* $\mathrm{POL} \longrightarrow \mathrm{LIN}$.

Beweis Wir setzen $n' = \binom{n+d}{d} = O(n^d)$. Da d eine Konstante ist, ist n' polynomiell in n beschränkt. Es sei $M_1, \ldots, M_{n'}$ eine beliebige aber feste Auflistung aller Monome vom Maximalgrad n in d Variablen. Es sei $X'_1, \ldots, X'_{n'}$ eine entsprechende Auflistung von neuen Variablen. Wenn wir in einem Polynom $h \in \mathrm{POL}_d^n$ jedes Monom M_i durch die Variable X'_i ersetzen, erhalten wir eine lineare Funktion h'. Dies liefert die Hypothesentransformation $h \mapsto h'$. Die Rücksubstitution von M_i für X'_i ergibt die Hypothesentransformation $h' \mapsto h$. Es sei $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$. Es bezeichne $M_i(x)$ die Auswertung von M_i an der Stelle x. Wir setzen $x' = (M_1(x), \ldots, M_{n'}(x))$. Dies liefert die Instanzentransformation $x \mapsto x'$. Es ist leicht zu verifizieren, dass diese Transformationen alle in Definition 9.13 gestellten Anforderungen erfüllen. **qed.**

Indem wir die Theoreme 9.12, 9.14 und 9.17 kombinieren, erhalten wir die

Folgerung 9.18 *Die ERM-Lernregel (bezüglich quadratischem Fehler) ist für die Klasse* POL *effizient implementierbar.*

9.4 Logistische Regression

Es bezeichne

$$S(z) = \frac{1}{1 + \exp(-z)}$$

die sogenannte *Sigmoid-Funktion*. Sie hat folgende (leicht zu verifizierende) Eigenschaften:

1. S is streng monoton wachsend.
2. $\lim_{z\to-\infty} S(z) = 0$, $\lim_{z\to\infty} S(z) = 1$ und $S(0) = 1/2$.

Wir können S als eine glatte Version einer von 0 auf 1 wechselnden Schwellenfunktion ansehen. Wir definieren die Klasse SIG_d^0 wie folgt:

$$\mathrm{SIG}_d^0 = S \circ \mathrm{LIN}_d^0 = \{x \mapsto S(\langle w, x\rangle) : w \in \mathbb{R}^d\} \ .$$

Wir setzen der Kürze halber

$$h_w(x) = S(\langle w, x\rangle) = \frac{1}{1 + \exp(-\langle w, x\rangle)} \ .$$

Lemma 9.19 *Wenn wir $h_w(x)$ (bzw. $1 - h_w(x)$) interpretieren als die Wahrscheinlichkeit, eine Instanz x mit dem Label 1 (bzw. -1) zu versehen, dann ist die Wahrscheinlichkeit x mit dem Label $y \in \{-1, 1\}$ zu versehen gegeben durch* $(1 + \exp(-y\langle w, x\rangle))^{-1}$.

Beweis Die Aussage ist für $y = 1$ offensichtlich richtig. Eine einfache Rechnung zeigt, dass

$$1 - h_w(x) = \frac{1}{1 + \exp(\langle w, x\rangle)} \ .$$

Somit stimmt die Aussage auch für $y = -1$. **qed.**

Die *logistische Verlustfunktion* ist definiert wie folgt:

$$\ell(h_w, (x, y)) = \log(1 + \exp(-y\langle w, x\rangle)) \ .$$

Wir sagen eine Verlustfunktion ℓ *entspricht dem „Maximum Likelihood (ML)“-Prinzip*, wenn die ERM-Lernregel bezüglich ℓ zu einer Hypothese führt, welche die Wahrscheinlichkeit einer gegebenen Trainingsmenge maximiert.

Theorem 9.20 *Wir interpretieren $h_w(x)$ (bzw. $1 - h_w(x)$) als die Wahrscheinlichkeit, eine Instanz x mit dem Label 1 (bzw. -1) zu versehen. Weiterhin unterstellen wir, dass die zufälligen Markierungen verschiedener Instanzen unabhängig von einander erfolgen. Unter diesen Voraussetzungen entspricht die logistische Verlustfunktion dem ML-Prinzip.*

Beweis Es sei $S = (x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^d \times \{-1, 1\}$ eine gegebene Sequenz von Trainingsdaten. Die S durch h_w zugewiesene Wahrscheinlichkeit ist dann gegeben durch

$$p_S(w) = \prod_{i=1}^{m} \frac{1}{1 + \exp(-y\langle w, x\rangle)} \ .$$

Offensichtlich sind die folgenden Optimierungsprobleme äquivalent:

- Maximieren von $p_S(w)$.
- Minimieren von $\prod_{i=1}^{m}(1 + \exp(-y\langle w, x\rangle))$
- Minimieren von $\frac{1}{m}\sum_{i=1}^{m} \log(1 + \exp(-y\langle w, x\rangle))$.

Die letzte Zielfunktion ist aber identisch zu $\hat{L}_S(h_w)$, wenn wir die logistische Verlustfunktion zugrunde legen. **qed.**

Wir merken kurz an, dass die Zielfunktion

$$\hat{L}_S(h_w) = \frac{1}{m}\sum_{i=1}^{m} \log(1 + \exp(-y\langle w, x\rangle))$$

eine in w konvexe Funktion ist. Zur Minimierung konvexer Funktionen existieren effiziente Standardwerkzeuge (zum Beispiel „Interior Point"-Methoden). Die ERM-Lernregel bei Zugrundelegung der logistischen Verlustfunktion ist daher effizient implementierbar.