# Theorie des maschinellen Lernens

Hans U. Simon

12. Juni 2017

## 11 Modellselektion und Validierung

Zur Lösung eines Lernproblems haben wir i.A. mehrere Optionen wie, zum Beispiel, die Auswahl zwischen verschiedenen Lernmethoden, zwischen verschiedenen Hypothesenklassen oder auch zwischen verschiedenen Wertzuweisungen an programmierbare Parameter eines Lernverfahrens (wie etwa die Festlegung des Parameters $T$ für die Anzahl der Iterationen bei AdaBoost). Das Problem, sich auf eine dieser Optionen festzulegen, wird als das Modellselektionsproblem bezeichnet. Tendenziell neigen zu primitive Modelle dazu, einen hohen Approximationsfehler zu produzieren (underfitting), wohingegen zu komplexe Modelle die Gefahr eines hohen Schätzfehlers (overfitting) bergen. Das resultierende Dilemma haben wir in einem früheren Kapitel als das „bias-complexity“ Dilemma bezeichnet.

Abbildung 1 veranschaulicht das Problem der Modellselektion:

- Die $L_S(h_S)$-Kurve fällt monoton mit wachsender Komplexität des Modells, da sich das Modell immer besser an die Daten in $S$ anpassen kann.
- Die (uns eigentlich interessierende) $L_{\mathcal{D}}(h_S)$-Kurve fällt zunächst bei wachsender Komplexität des Modells, da der Approximationsfehler abnimmt. Wird ein bestimmter Grad

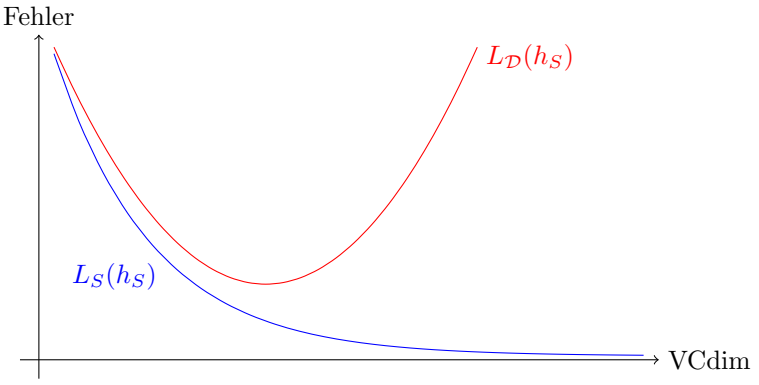


Abbildung 1: Schematische Darstellung des „bias-complexity“ Dilemmas.

der Komplexität überschritten, steigt der $L_{\mathcal{D}}(h_S)$-Wert jedoch wieder an, da $h_S$ sich zu stark an die Daten in $S$ anpasst.

Die Kunst besteht darin, die Komplexität des Modells so zu justieren, dass wir dem Minimalwert der $L_{\mathcal{D}}(h_S)$-Kurve nahekommen.

Im Abschnitt 11.1 besprechen wir den SRM-basierten Ansatz zur Lösung des Modellselektionsproblems. Er ist theoretisch gut fundiert, leidet aber an zu pessimistischen oberen Schranken für die Fehlerraten. Praktikabler ist die validierungsbasierte Methode, die wir in Abschnitt 11.2 besprechen. Was ist zu tun, wenn die von uns präferierte Methode zu schlechten Ergebnissen führt? Um klug zu reagieren, muss man zunächst herausfinden, ob „over-“ oder „underfitting“ (oder beides) vorliegt. In Abschnitt 11.3 machen wir entsprechende Diagnose- und Therapievorschläge.

## 11.1 SRM-basierte Modellselektion

Wir betrachten den Fall, dass $\mathcal{H}$ eine abzählbare Vereinigung von binären Hypothesenklassen ist, welche die uniforme Konvergenzbedingung erfüllen. Konkreter: es sei $\mathcal{H} = \cup_{d\geq 1}\mathcal{H}_d$ und, für alle $d \geq 1$, sei $\mathcal{H}_d$ eine binäre Hypothesenklasse der VC-Dimension $d$. Gemäß dem Fundamentaltheorem der PAC-Lerntheorie gibt es eine Konstante $c > 0$, so dass

$$m^{UC}_{\mathcal{H}_d}(\varepsilon, \delta) \leq \frac{c}{\varepsilon^2}\left(d + \ln\left(\frac{1}{\delta}\right)\right) \ . \tag{1}$$

Wie im Kapitel über nichtuniforme Lernbarkeit sei

$$\varepsilon^{UC}_d(m, \delta) = \min\{\varepsilon \in (0,1) |\ m^{UC}_{\mathcal{H}_d}(\varepsilon, \delta) \leq m\} \ .$$

Aus (1) lässt sich mit einer einfachen Rechnung ableiten:

$$\varepsilon^{UC}_d(m, \delta) = \sqrt{\frac{c}{m}\left(d + \ln\left(\frac{1}{\delta}\right)\right)} \ . \tag{2}$$

Die SRM-Lernregel wählt zu einer gegebenen Trainingsmenge $S \in Z^m$ eine Hypothese $h_S \in \mathcal{H}$ aus, welche die Kostenfunktion $L_S(h) + \varepsilon^{UC}_{d(h)}(m, w_{d(h)}\delta)$ minimiert. Hierbei sei $d(h)$ der kleinste Index $d$ mit $h \in \mathcal{H}_d$ und $w_1, w_2, \ldots > 0$ seien Gewichtsparameter, die sich zu 1 aufaddieren. Wir setzen $w_d = \frac{1}{d(d+1)}$. Folglich ist $h_S \in \mathcal{H}$ eine Hypothese, welche die Kostenfunktion

$$\begin{aligned} f_S(h) &:= L_S(h) + \varepsilon^{UC}_{d(h)}\left(m, \frac{\delta}{d(h)(d(h)+1)}\right) \\ &\overset{(2)}{=} L_S(h) + \sqrt{\frac{c}{m}\left(d + \ln(d(h)(d(h)+1)) + \ln\left(\frac{1}{\delta}\right)\right)} \end{aligned}$$

minimiert. Aus dem Hauptresultat in dem Kapitel über nichtuniforme Lernbarkeit wissen wir, dass mit einer Wahrscheinlichkeit von mindestens $1 - \delta$ eine zufällige Trainingsmenge

$S \sim \mathcal{D}^m$ generiert wird, so dass die Ungleichung $L_{\mathcal{D}}(h) \leq f_S(h)$ für alle $h \in \mathcal{H}$ erfüllt ist. Die SRM-Lernregel wählt also eine Hypothese $h_S \in \mathcal{H}$ (und damit ein „Modell“ $\mathcal{H}_{d(h_S)}$) aus, die diese (mit hoher Wahrscheinlichkeit geltende) obere Schranke für die Fehlerrate $L_{\mathcal{D}}(h)$ minimiert. Der Term

$$\sqrt{\frac{c}{m}\left(d + \ln(d(d+1)) + \ln\left(\frac{1}{\delta}\right)\right)}$$

kann als Strafterm für die Wahl des „Modells“ $\mathcal{H}_d$ (also für die Selektion einer Hypothese aus $\mathcal{H}_d$) angesehen werden. Modelle einer höheren VC-Dimension werden stärker „bestraft“ als Modelle einer vergleichsweise kleinen VC-Dimension. Damit soll die Wahl eines zu komplexen Modells, das sich zu stark an die Trainingsdaten anpasst (overfitting), verhindert werden. Andererseits sorgt der Term $L_S(h)$ dafür, dass auch zu primitive Modelle, welche die empirisch ermittelten Daten nicht gut beschreiben können (underfitting), diskreditiert werden. Das Problem bei dieser Hypothesen- und Modellselektion ist, dass die obere Schranke $f_S(h)$ für $L_{\mathcal{D}}(h)$ i.A. viel zu pessimistisch ist. Daher wird das eigentliche Ziel, eine Hypothese mit minimaler Fehlerrate auszuwählen, möglicherweise verfehlt. Im nächsten Abschnitt lernen wir ein praktikableres Verfahren für Modell- und Hypothesenselektion kennen.

## 11.2 Validierungsbasierte Modellselektion

In diesem Abschnitt sei $\mathcal{H}$ eine (nicht notwendig binäre) Hypothesenklasse und $\ell : \mathcal{H} \times Z \to [0,1]$ eine Verlustfunktion mit dem beschränkten Wertebereich $[0,1]$. Es sei $(S,V) \sim \mathcal{D}^m \times \mathcal{D}^{m'}$ eine Menge aus insgesamt $m + m'$ Trainingsbeispielen. Dabei wird $S \sim \mathcal{D}^m$ wie üblich als Trainingsmenge bezeichnet. $V \sim \mathcal{D}^{m'}$ nennen wir die *Validierungsmenge*. Bei der validierungsbasierten Modellselektion mit einer beschränkten Anzahl $r$ an Modellen (wie etwa $r$ verschiedenen Hypothesenklassen oder $r$ möglichen Werten für einen programmierbaren Parameter) gehen wir vor wie folgt:

1. Für jedes Modell $i \in [r]$ bestimmen wir in Abhängigkeit von der Trainingsmenge $S$ (aber ohne Ansehen von $V$) eine Hypothese $h_i$ und erhalten auf diese Weise $r$ miteinander konkurierende Hypothesen $h_1, \ldots, h_r$.

2. Wir bestimmen für jede Hypothese $h_i$ ihren mittleren Verlustwert $L_V(h_i)$ auf $V$ und entscheiden uns am Ende für die Hypothese $h_{i^*}$ mit dem kleinsten $L_V$-Wert.

Da die Hypothesen $h_i$ ohne Ansehen von $V$ ermittelt wurden, benötigen wir zur Kontrolle von $|L_{\mathcal{D}}(h_i) - L_V(h_i)|$ — also der Kontrolle der absoluten Differenz zwischen dem erwarteten Verlustwert und dem mittleren Verlustwert auf $V$ — kein uniformes Konvergenzkriterium, sondern können die Hoeffding-Ungleichung (in Verbindung mit der „Union Bound“) bemühen. Auf diese Weise erhält man leicht[1] das folgende Resultat:

$$\Pr_{V \sim \mathcal{D}^{m'}}\left[\forall i \in [r] : |L_{\mathcal{D}}(h_i) - L_V(h_i)| \leq \sqrt{\frac{\ln(2r/\delta)}{2m'}}\right] \geq 1 - \delta \ .$$

[1] Die mathematische Herleitung ist analog zur Analyse des PAC-Lernens endlicher Hypothesenklassen.

Den in dieser Schranke vorkommenden Parameter $r$ (Anzahl der in Betracht gezogenen Modelle) können wir als Strafterm dafür ansehen, dass wir uns nicht schon zu Anfang auf ein Modell festlegen. In Betracht ziehen von vielen Modellen birgt die Gefahr des „overfitting“.

Eine Variante der validierungsbasierten Modellselektion ist die sogenannte *Kreuzvalidierung*. Hierbei wird die Trainingsmenge $S \sim \mathcal{D}^m$ in $k$ ungefähr gleich große Blöcke $S_1, \ldots, S_k$ zerlegt. Bei insgesamt $k$ Runden spielt in Runde $i$ der Block $S_i$ die Rolle der Validierungsmenge und $S \setminus S_i$ die Rolle der Trainingsmenge. Der Mittelwert der Verlustwerte $L_{S_i}(\cdot)$ dient dann als Schätzwert für den erwarteten Verlustwert $L_{\mathcal{D}}(\cdot)$.

Stellen wir uns vor es sei ein Lernverfahren $A$ gegeben, das einen programmierbaren Parameter besitzt, welcher Werte $\theta$ aus einer endlichen Menge $\Theta$ annehmen kann. Modellselektion bedeutet dann Festlegung auf einen Wert $\theta^* \in \Theta$. Unter Verwendung von Kreuzvalidierung würden wir uns für den Wert $\theta^*$ entscheiden, der zu einer Hypothese mit dem kleinsten Mittelwert der $L_{S_i}$-Werte führt. Danach setzen wir den Algorithmus $A$, parametrisiert mit $\theta^*$, noch einmal auf die gesamte Trainingsmenge $S$ an, und erhalten die finale Hypothese $h_{\theta^*} = A(S; \theta^*)$. Der Pseudocode für diese Methode liest sich wie folgt:

**Eingabe:** $S \in Z^m$, $k \geq 1$, Algorithmus $A$, Wertemenge $\Theta$
**Methode:**

1. Zerlege $S$ in $k$ etwa gleich große Blöcke $S_1, \ldots, S_k$.
2. Für alle $\theta \in \Theta$ mache folgendes:
   (a) Für alle $i = 1, \ldots, k$ setze $h_{i,\theta} := A(S \setminus S_i; \theta)$.
   (b) Setze $\bar{L}(\theta) := \frac{1}{k} \sum_{i=1}^{k} L_{S_i}(h_{i,\theta})$.
3. Setze $\theta^* := \operatorname{argmin}_\theta[\bar{L}(\theta)]$ und $h_{\theta^*} := A(S; \theta^*)$.

**Ausgabe:** $h_{\theta^*}$

Die Methode der Kreuzvalidierung ist theoretisch noch nicht gut verstanden. Mathematische Analysen existieren nur für einige Spezialfälle (wie zum Beispiel „Nearest Neighbor“).

**Trainings-, Validierungs- und Testmenge.** Es ist eine naheliegende und häufig praktizierte Idee, die Trainings- und die Validierungsmenge um eine weitere Menge $T$ von *Testbeispielen* zu ergänzen, wobei $(S, V, T) \sim \mathcal{D}^m \times \mathcal{D}^{m'} \times \mathcal{D}^{m''}$. Mit Hilfe der Trainingsmenge $S$ werden, bei $r$ miteinander konkurrierenden Modellen, insgesamt $r$ verschiedene Hypothesen erzeugt. Mit Hilfe der Validierungsmenge ermittelt man das „Siegermodell“ $\theta^*$ und die „Siegerhypothese“ $h_{\theta^*}$. Schließlich bestimmt man den mittleren Verlustwert $L_T(h_{\theta^*})$ der Siegerhypothese auf der Testmenge. Da $h_{\theta^*}$ ohne Ansehen von $T$ ermittelt wurde, lässt sich der Fehlerterm $|L_{\mathcal{D}}(h_{\theta^*}) - L_T(h_{\theta^*})|$ über die Hoeffding-Ungleichung kontrollieren und es gilt

$$\Pr\left[|L_{\mathcal{D}}(h_{\theta^*}) - L_T(h_{\theta^*})| \leq \sqrt{\frac{\ln(2/\delta)}{2m''}}\right] \geq 1 - \delta \ .$$

## 11.3 Hilfsmaßnahmen bei hohem Testfehler

In diesem Abschnitt bezeichne $h_S$ eine ERM-Hypothese zu einer Trainingsmenge $S \sim \mathcal{D}^m$ (und zu einem per Modellselektion vorher bestimmten Modell $\theta^*$). Wenn der Testfehler $L_T(h_S)$ „groß“ ist, dann gilt dies vermutlich auch für den erwarteten Verlustwert $L_{\mathcal{D}}(h_S)$. Was ist dann zu tun? Es gibt eine ganze Reihe von möglichen Maßnahmen:

**Im Falle von „overfitting“:** Vergrößere die Trainingsmenge oder wähle ein weniger komplexes Modell.

**Im Falle von „underfitting“:** Wähle ein komplexeres Modell.

Perfiderweise können „over-“ und „underfitting“ sogar kombiniert auftreten. Zum Beispiel könnte das Modell $\theta^*$

**einerseits** so komplex sein, dass es eine Überanpassung an die Daten in $S$ ermöglicht,

**andererseits** so inadäquat für das vorliegende Lernproblem sein, dass es keine Hypothese mit kleinem erwarteten Verlustwert bereithält.

In diesem Fall sind radikalere Maßnahmen erforderlich. Beispielsweise:

- Ziehe andere Merkmalsvektoren zur Darstellung der Daten in Betracht.
- Ziehe andersartige Hypothesenklassen bzw. andersartige Lernalgorithmen in Betracht.

Maßnahmen dieser Art könnten auch bei reinem „overfitting“ in Betracht kommen, wenn sich einerseits keine weiteren Trainingsdaten beschaffen lassen, aber andererseits der Übergang zu einem weniger komplexen Modell schlagartig zu „underfitting“ führt. Eine entsprechende Bemerkung gilt für reines „underfitting“, falls der Übergang zu einem komplexeren Modell wegen eintretenden „overfittings“ versperrt ist.

Die genannten Ratschläge sind gut und schön, aber wie stelle ich fest, ob der hohe Testfehler auf „over-“ oder auf „underfitting“ beruht? Es wird sich im Verlaufe dieses Abschnittes zeigen: unser Hauptinstrument für Diagnose und Einleiten entsprechender „Therapiemaßnahmen“ ist die *Verlustwertzerlegung*.

Es bezeichne $h^* \in \mathcal{H}$ die Hypothese aus $\mathcal{H}$ mit dem kleinstmöglichen erwarteten Verlustwert $L_{\mathcal{D}}(h)$. Wir erinnern an die folgende Zerlegung aus einem früheren Kapitel:

$$L_{\mathcal{D}}(h_S) = \underbrace{L_{\mathcal{D}}(h^*)}_{=:\varepsilon_{app}} + \underbrace{(L_{\mathcal{D}}(h_S) - L_{\mathcal{D}}(h^*))}_{=:\varepsilon_{est}} \quad .$$

Wir hatten seinerzeit $\varepsilon_{app}$ als den *Approximationsfehler* und $\varepsilon_{est}$ als den *Schätzfehler* bezeichnet. Ein hoher Approximationsfehler zeigt „underfitting“ und ein hoher Schätzfehler zeigt „overfitting“ an. Das Problem ist nur: da wir $\mathcal{D}$ nicht kennen, kennen wir weder $\varepsilon_{app}$ noch $\varepsilon_{est}$. Wir betrachten daher besser die folgende Verlustwertzerlegung:[2]

$$L_{\mathcal{D}}(h_S) = \underbrace{(L_{\mathcal{D}}(h_S) - L_T(h_S))}_{\text{unbek. aber klein}} + \underbrace{(L_T(h_S) - L_S(h_S))}_{\text{bekannt}} + \underbrace{L_S(h_S)}_{\text{bekannt}} \quad .$$

[2]Wenn $h_S$ ohne Ansehen von $V$ ermittelt wurde, kann bei dieser Zerlegung $T$ durch $V$ ersetzt werden.

Wir verwenden hier und im Folgenden die Daumenregel, dass auf der Hoeffding-Ungleichung beruhende Schätzungen „gut“ sind. Die betreffenden Schätzfehler, wie zum Beispiel $L_{\mathcal{D}}(h_S) - L_T(h_S)$, bezeichnen wir salopp einfach als „klein“. Wenn $L_T(h_S)$ „klein“ ist, so ist auch $L_{\mathcal{D}}(h_S)$ „klein“ und wir können uns entspannt zurücklehnen. Alle folgenden Überlegungen setzen einen unbefriedigend hohen Wert von $L_T(h_S)$ voraus, so dass Handlungsbedarf besteht.

Da $L_T(h_S)$ „groß“ ist, muss auch mindestens einer der beiden Terme $L_T(h_S) - L_S(h_S)$ und $L_S(h_S)$ ebenfalls „groß“ sein. Wenn $L_T(h_S) - L_S(h_S)$ „groß“ ist, dann ist auch $L_{\mathcal{D}}(h_S) - L_S(h_S)$ „groß“, d.h., die Hypothese ist an die Trainingsmenge $S$ überangepasst und es liegt „overfitting“ vor. Um eine entsprechende Überlegung für den Term $L_S(h_S)$ anzustellen, betrachten wir die Zerlegung

$$L_S(h_S) = \overbrace{\underbrace{(L_S(h_S) - L_S(h^*))}_{\text{unbekannt}}}^{\leq 0} + \underbrace{(L_S(h^*) - L_{\mathcal{D}}(h^*))}_{\text{unbek. aber klein}} + L_{\mathcal{D}}(h^*) \ . \tag{3}$$

Beachte, dass die Differenz $L_S(h^*) - L_{\mathcal{D}}(h^*)$ darum (betragsmäßig) klein ist, weil die obige Definition von $h^*$ (als Hypothese aus $\mathcal{H}$ mit dem kleinstmöglichen erwarteten Verlustwert) nicht von $S$ abhängt und $L_S(h^*) - L_{\mathcal{D}}(h^*)$ daher mit der Hoeffding-Ungleichung kontrolliert werden kann. $L_S(h_S) - L_S(h^*) \leq 0$ gilt, da $h_S$ als ERM-Hypothese zu gegebener Trainingsmenge $S$ vorausgesetzt wurde. Aus (3) folgern wir: wenn $L_S(h_S)$ „groß“ ist, dann ist $\varepsilon_{app} = L_{\mathcal{D}}(h^*)$ ebenfalls groß und es liegt „underfitting“ vor. Wenn $L_S(h_S)$ „klein“ ist, so ist (wie weiter oben bereits schon einmal angemerkt) $L_T(h_S) - L_S(h_S)$ „groß“ und es liegt „overfitting“ vor. Allerdings ist es durchaus denkbar, dass zusätzlich auch noch $L_{\mathcal{D}}(h^*)$ „groß“ ist (was „underfitting“ impliziert): nämlich dann, wenn $L_S(h_S) - L_S(h^*)$ ein betragsmäßig großer negativer Wert ist. Um

$$L_S(h_S) \text{ „klein“ und } L_{\mathcal{D}}(h^*) \text{ „groß“ (Fall 1)}$$

zu unterscheiden von

$$L_S(h_S) \text{ „klein“ und } L_{\mathcal{D}}(h^*) \text{ „klein“ (Fall 2)} \quad ,$$

könnte man beobachten, wie sich der mittlere Verlustwert $L_S(h_S)$ von $h_S$ auf $S$ und der mittlere Verlustwert $L_T(h_S)$ von $h_S$ auf $T$ bei wachsender Anzahl von Beispielen (also wachsendem $m$ bzw. wachsendem $m''$) verhalten:

1. Im Fall 1 müsste $L_T(h_S)$ bei wachsendem $m''$ auf einem hohen Niveau verweilen (da $L_{\mathcal{D}}(h_S)$ nicht kleiner als $L_{\mathcal{D}}(h^*)$ werden kann).[3]

2. Im Fall 2 ist zu erwarten, dass $L_T(h_S)$ auf ein niedriges Niveau absinkt und $L_S(h_S)$ sich dem Niveau von $L_T(h_S)$ annähert.

[3] $L_S(h_S)$ wird bei wachsendem $m$ solange „klein“ bleiben, wie die Situation des „overfitting“ noch nicht beseitigt ist.

Wir fassen diese informellen Überlegungen zu folgenden Daumenregeln zusammen:

$$
\begin{aligned}
L_T(h_S) - L_S(h_S) \text{ „groß“} &\Rightarrow \text{„overfitting“} \\
L_S(h_S) \text{ „groß“} &\Rightarrow \text{„underfitting“} \\
L_S(h_S) \text{ „klein“ und Indizien für } L_{\mathcal{D}}(h^*) \text{ „groß“} &\Rightarrow \text{„over-“ und „underfitting“} \\
L_S(h_S) \text{ „klein“ und Indizien für } L_{\mathcal{D}}(h^*) \text{ „klein“} &\Rightarrow \text{„overfitting“}
\end{aligned}
$$

Es können dann, wie zu Anfang dieses Abschnittes besprochen, die entsprechenden Maßnahmen eingeleitet werden.