Informatik/Mathematik/Komplexe Systeme

Statistische Lerntheorie und Empirische Inferenz

Schölkopf, Bernhard

Max-Planck-Institut für biologische Kybernetik, Tübingen

Abteilung - Empirische Inferenz für maschinelles Lernen und Wahrnehmung
Korrespondierender Autor: Schölkopf, Bernhard
E-Mail: bernhard.schoelkopf@tuebingen.mpg.de

Zusammenfassung
Die Lerntheorie befasst sich mit der Extraktion von Gesetzmäßigkeiten aus Beobachtungen. Das Grundproblem hierbei ist die ‚Generalisierung:' die extrahierten Gesetzmäßigkeiten sollen nicht nur die bereits vorliegenden Beobachtungen (die ‚Trainingsmenge') korrekt erklären, sondern auch für neue Beobachtungen zutreffend sein. Dieses Problem der Induktion berührt Grundsatzfragen nicht nur der Statistik, sondern der empirischen Wissenschaften im Allgemeinen. Dazu gehören die Repräsentation von Daten und von Vorwissen, sowie die Komplexität oder Kapazität von Erklärungen bzw. Modellen. Wenn anhand eines Modells geringer Komplexität (in einer geeigneten Formalisierung dieses Begriffs) eine gegebene Menge von empirischen Beobachtungen hinreichend genau erklärt werden kann, dann garantiert die statistische Lerntheorie, dass mit hoher Wahrscheinlichkeit auch zukünftige Beobachtungen mit dem Modell konsistent sein werden.

Abstract
Statistical learning theory studies the process of inferring regularities from empirical data. The fundamental problem is what is called generalization: how it is possible to infer a law which will be valid for an infinite number of future observations, given only a finite amount of data? This problem hinges upon fundamental issues of statistics and science in general, such as the problems of complexity of explanations, a priori knowledge, and representation of data.

Lernen aus Beispielen

Das Problem des Lernens aus Beispielen [7] lässt sich wie folgt formulieren: sei $y = f(x)$ ein unbekannter funktionaler Zusammenhang der Größen x und y. Ziel ist, aus einer begrenzten Anzahl von Beispielen (der Trainingsmenge) $(x_1,y_1),\ldots, (x_l,y_l)$ die Funktion f zu lernen.
Welche Aspekte gilt es hierbei zu berücksichtigen, um zu einer verlässlichen Schätzung f* zu kommen? Zunächst einmal sollte die gelernte Funktion die Beispiele erklären, d.h. $f^*(x_1) = y_1,\ldots,f^*(x_l) = y_l$, wenigstens näherungsweise. **Abbildung 1** zeigt, dass dies noch nicht alles ist: Die abgebildeten Beispiele werden durch die gestrichelte Funktion im Gegensatz zu der durchgezogenen Geraden vollständig erklärt; nichtsdestotrotz ist man geneigt, der Geraden mehr Vertrauen zu schenken. Die Komplexität der (Klasse von) Funktionen, die man für die Lösung des Problems zulässt, beeinflusst also unser Vertrauen in die gefundene Lösung. Die Formalisierung dieser Erkenntnis bildet den Kern der statistischen Lerntheorie.

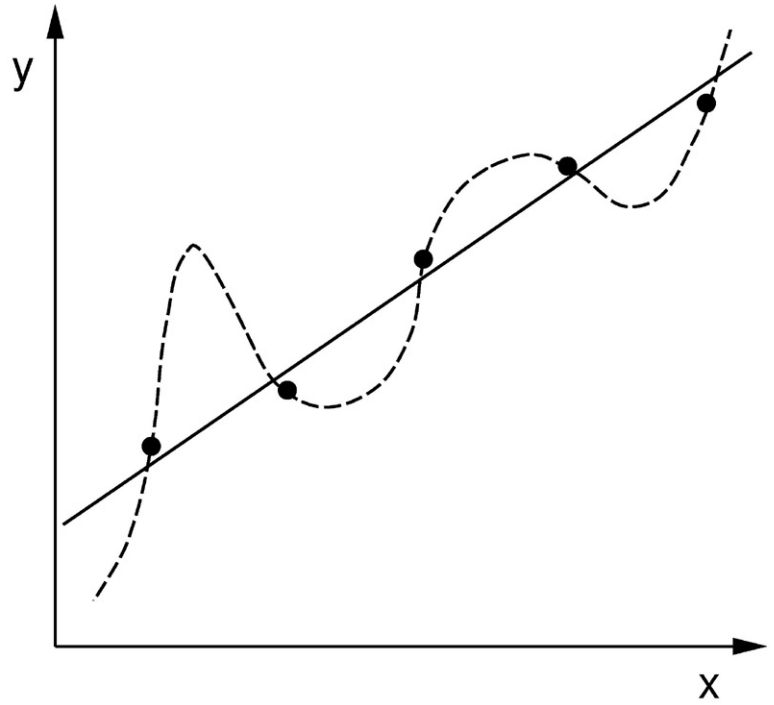

Abb. 1 : *Aus einer gegebenen Menge von Beispielen (im Bild: schwarze Punkte) soll ein unbekannter funktionaler Zusammenhang geschätzt werden. Welche der beiden abgebildeten Hypothesen scheint plausibler?*
Bild : Max-Planck-Institute für biologische Kybernetik/Schölkopf

Der Einfachheit halber werden wir uns im folgenden auf (binäre) Mustererkennung beschränken, d.h. auf Funktionen, die nur zwei verschiedene Werte annehmen können (so genannte Entscheidungsfunktionen). Obige Erkenntnis von der Wichtigkeit der zugelassenen Funktionen berücksichtigen wir, indem wir das Lernproblem wie folgt fassen:
Gegeben sei eine Menge von Entscheidungsfunktionen, und eine Menge von Trainingsbeispielen (x_1, y_1),…, (x_l,y_l), zufällig erzeugt gemäß einer unbekannten Wahrscheinlichkeitsverteilung. Ziel ist es, eine Funktion auszuwählen, die den Zusammenhang zwischen x und y für Testbeispiele, gezogen aus derselben Wahrscheinlichkeitsverteilung, am besten wiedergibt.
Die Muster x könnten z.B. Repräsentationen von Krankheitssymptomen sein, aus denen auf das Vorhandensein y einer bestimmten Krankheit geschlossen werden soll. Der Zusammenhang zwischen x und y wird durch eine Wahrscheinlichkeitsverteilung P(x,y) modelliert. Diese enthält als Spezialfall die Möglichkeit eines deterministischen Zusammenhanges y = f(x).
Die gesuchte Funktion soll den Zusammenhang zwischen x und y im Mittel am besten reproduzieren (die Wahrscheinlichkeit für Klassifikationsfehler bei Testbeispielen, das (erwartete) Risiko, soll minimal sein). Die Mittelung wird hierbei gemäß der Wahrscheinlichkeitsverteilung P(x,y) vollzogen. Da man diese nicht kennt, benötigt man ein Induktionsprinzip, um die Funktion wenigstens näherungsweise zu finden, und zwar auf Basis dessen, was wir über P(x,y) beobachtet haben, also der Trainingsmenge.

Induktionsprinzipien

Eine nahe liegende und vielfach verwendete Methode zur Risikominimierung besteht in der Minimierung des empirischen Risikos, also des Anteils der Fehlklassifikationen auf der Trainingsmenge. Das Prinzip der empirischen Risikominimierung ist jedoch kein Garant für hohe Generalisierungsfähigkeit: Ein nicht in der Trainingsmenge enthaltenes Beispiel, erzeugt gemäß derselben Wahrscheinlichkeitsverteilung, liegt nicht notwendig auf der gestrichelten Linie in Abbildung 1. In anderen Worten: Von einem niedrigen empirischen Risiko alleine kann nicht auf ein niedriges erwartetes Risiko geschlossen werden. Hierzu muss gleichzeitig die Kapazität der Funktionsklasse kontrolliert werden, die sich beispielsweise durch die Vapnik-Chervonenkis-(VC-)Dimension messen lässt. Der trade-off zwischen Minimierung des empirischen Risikos und der Kapazität wird durch probabilistische Schranken beschrieben, deren Studium ein wesentliches Thema der Lerntheorie ist.
Die anschauliche Bedeutung dieser Schranken lässt sich wie folgt fassen: Schafft man es, die Trainingsdaten mit einem einfachen Modell (d.h. einer Funktionsklasse, deren VC-Dimension im Vergleich zur Anzahl der Trainingsbeispiele niedrig ist) zu erklären (d.h. das empirische Risiko gering zu halten), so besteht Grund zu der Annahme, dass der wirkliche funktionale Zusammenhang gefunden wurde. Kann man die Daten nur mit einer Lernmaschine (bzw. Funktionsklasse) von vergleichsweise hoher VC-Dimension erklären, so ist dies nicht der Fall: Die Maschine kann ihre Kapazität (VC-Dimension) dazu verwendet haben, die Beispiele einzeln zu memorisieren (overfitting), anstatt eine kompaktere zugrunde liegende Regularität zu lernen — dementsprechend ist nicht zu erwarten, dass neue Beispiele zuverlässig klassifiziert werden können. Das Prinzip der strukturellen Risikominimierung verwendet oben genannte probabilistische Schranken, um das erwartete Risiko durch Kontrolle von empirischem Risiko und VC-Dimension zu minimieren, d.h. um eine Funktion zu finden, die möglichst gut auf neue Beispiele generalisiert. Strukturelle Risikominimierung passt also in diesem Sinne die Komplexität der Lernmaschine dem zu lösenden Problem an, und stellt somit eine Basis für moderne Lernalgorithmen dar.
Die VC-Dimension stimmt in manchen Fällen mit der Anzahl der Parameter einer Funktionsklasse überein, die wiederum oft mit der Dimensionalität der Beobachtungsgrößen xi zusammenhängt. So lässt sich verstehen, dass die Anzahl der freien Parameter manchmal als ein Maß für die Komplexität oder Kapazität einer Funktionsklasse betrachtet wird. Dies ist allerdings nicht immer richtig, mit der nachteiligen Konsequenz, dass die Abschätzung der Komplexität einer Funktionsklasse sich nicht auf das Zählen von Parametern reduziert, sondern ein schwieriges mathematisches Problem sein kann. Der Vorteil ist jedoch, dass die Hoffnung besteht, auch dann noch generalisieren zu können, wenn die Daten (bzw. die Parametervektoren) sehr hochdimensional sind. Eine Klasse von Lernalgorithmen, die diese Hoffnung bestätigen, wurde in den letzten Jahren als Support-Vektor-Maschinen (SVM) oder allgemeiner Kernalgorithmen bekannt [5, 6].

Kernalgorithmen

Die Schwierigkeit beim Abschätzen der Kapazität von Funktionsklassen besteht darin, dass Größen wie die VC-Dimension kombinatorischer Natur sind. Dies führt zu einem Dilemma: Gute Abschätzungen existieren vor allem für einfache (z.B. lineare) Funktionsklassen; Auf der anderen Seite ist es aber unser Ziel, Lernalgorithmen zu konstruieren, die für nichtlineare Funktionsklassen funktionieren. Nur so können wir hoffen, damit komplexe Phänomene der Naturwissenschaften analysieren zu können. Dieses Dilemma wird von Kernalgorithmen elegant aufgelöst. Die von Kernalgorithmen verwendeten Funktionen entsprechen linearen Funktionen in einem zugeordneten hochdimensionalen Merkmalsraum H; als Funktionen auf den Eingabedaten sind sie jedoch nichtlinear. Die theoretische Analyse kann in dem Merkmalsraum vorgenommen werden, in der Praxis arbeiten wir aber mit nichtlinearen Funktionen.

Im Fall der Mustererkennung tun Support-Vektor-Maschinen dies, indem sie die Muster effektiv in einen hochdimensionalen Merkmalsraum abbilden und dort eine Hyperebene zur Separation der beiden Klassen konstruieren. Für diese Funktionsklasse lässt sich die VC-Dimension abschätzen und algorithmisch kontrollieren. Durch Lösung eines quadratischen Optimierungsproblems findet man die Funktion, die die beste Generalisierungsleistung verspricht. Es stellt sich heraus, dass diese Funktion nur von wenigen Trainingsbeispielen, den Support-Vektoren, abhängt. Klassifikation eines neuen Musters geschieht durch gewichteten Vergleich mit den Support-Vektoren. Unterschiedliche Typen von Support-Vektor-Maschinen lassen sich durch die Wahl des so genannten Kernes k implementieren. Allen Kernalgorithmen ist gemein, dass der Kern ein Skalarprodukt im Merkmalsraum H berechnet. Formell bedeutet dies, dass eine Abbildung $\Phi : \chi \rightarrow H$ existiert mit der Eigenschaft, dass für alle Eingabedaten x,x' welche Elemente aus χ sind die Gleichung $k(x,x') = \langle \Phi(x), \Phi(x') \rangle$ gilt. Beispiele von solchen positiv definiten Kernen sind, unter geeigneten Bedingungen an den Inputraum χ, der Gauß-Kern $k(x,x') = \exp(-||x - x'||^2)$ sowie der Polynom-Kern $k(x,x') = \langle x,x' \rangle^d$ (wobei d Elemente aus der Menge der natürlichen Zahlen sind). Man kann zeigen, dass letzterer eine Abbildung Φ induziert, die alle Produkte der Ordnung d in den Inputkoordinaten berechnet.

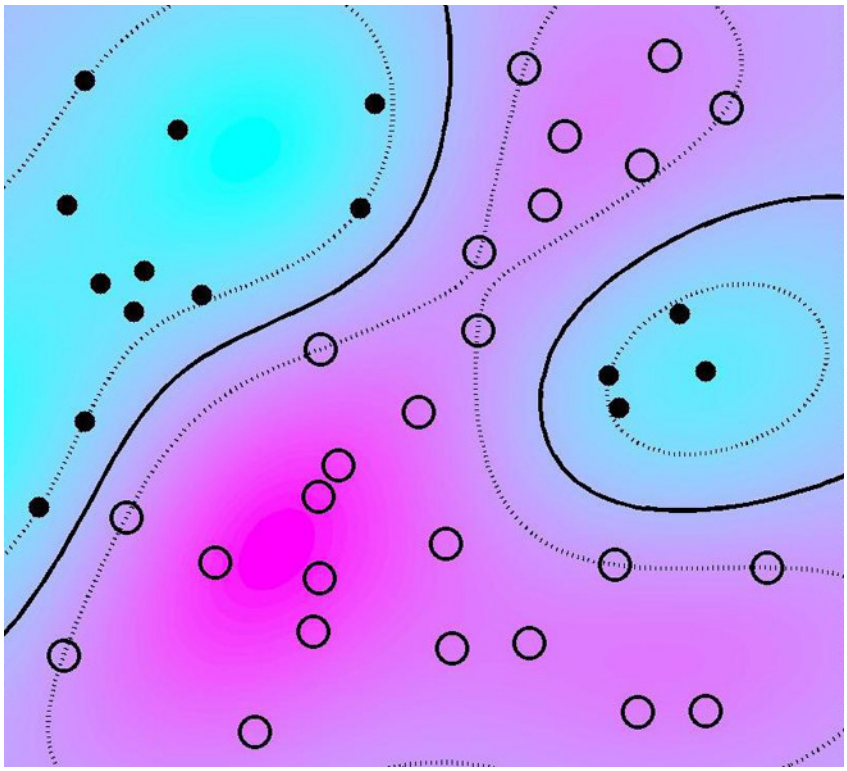

Abb. 2 : *Eine Entscheidungsfunktion, die zwei Klassen von zweidimensionalen Mustern (dargestellt als Scheiben bzw. Kreise) separiert. Verwendet wurde eine Support-Vektor-Maschine mit Gauß-Kern. Die Support-Vektoren liegen auf den gestrichelten Linien. Die durchgezogenen Linien stellt die Entscheidungsgrenze dar; im hochdimensionalen Merkmalsraum entsprechen sie einer Hyperebene.*

Bild : Max-Planck-Institute für biologische Kybernetik/Schölkopf

Positiv definite Kerne lassen sich auch für Datentypen definieren, die nicht vektorieller Natur sind, wie z.B. Sequenzdaten. Die Abbildung Φ liefert dann eine Repräsentation der Daten in dem Skalarproduktraum H — so können geometrische Methoden für allgemeine Datentypen verwendet werden. Dies hat zu einer Reihe von interessanten Anwendungen insbesondere im Bereich der

computational biology geführt [6, 3]. Aber auch in anderen Feldern sind mit Kernmethoden sehr gute Ergebnisse erzielt worden, so zum Beispiel in der Objekterkennung und -Detektion [4], und im Training von Brain-Computer-Interfaces [2]. Besonders attraktiv an Kernmethoden ist aber nicht nur die Tatsache, dass die Anwendungsergebnisse hervorragend sind und Weltrekorde auf bekannten Benchmarks einschließen [1, 8], sondern, dass sie drei Grundprobleme der empirischen Inferenz thematisieren:
• Datenrepräsentation — der Kern k induziert eine Einbettung der Daten in den Vektorraum H
• Ähnlichkeit — der Kern k kann als (nichtlineares) Ähnlichkeitsmaß aufgefasst werden, mit dem Datenpunkte verglichen werden
• A-priori-Wissen — die Lösungen von Kern-Lernalgorithmen lassen sich im Allgemeinen als Kern-Entwicklungen in den Trainingsdaten ausdrücken. Daher parametrisiert der Kern die Klasse der Funktionen, aus denen die Lösung entnommen wird, d.h., dasjenige Wissen, das zusätzlich zu den Trainingsdaten in den Lernprozess eingeht.

Schlussbemerkung

Methoden des maschinellen Lernens wurden entwickelt zur Lösung von Induktionsproblemen, wie sie in den Naturwissenschaften überall vorkommen. Der traditionelle Ansatz zur Lösung solcher Probleme besteht darin, ein vorliegendes System so detailliert zu studieren, bis die zugrunde liegenden Vorgänge aufgedeckt sind und ein mechanistisches Modell aufgestellt werden kann. Das maschinelle Lernen kann solche Modelle nicht liefern, ist aber dafür in vielen Fällen einsetzbar, wo mechanistische Modelle nicht möglich sind, weil die zugrunde liegenden Prozesse zu komplex sind. Falls Trainingsdaten vorliegen, kann es uns in solchen Fällen oft zu sehr guten Nachbildungen von interessanten Aspekten komplexer Systeme liefern, die deren Verständnis entscheidend voranbringen können. Moderne Methoden des maschinellen Lernens und der empirischen Inferenz werden eine wesentliche Rolle beim Umgang mit einer komplexen Welt spielen.

Referenzen und weiterführende Links

Dieser Bericht mit zusätzlichen Anmerkungen als pdf-Datei

[1] D. DeCoste and B. Schölkopf. Training invariant support vector machines. *Machine Learning*, 46:161-190, 2002. Also: Technical Report JPL-MLTR-00-1, Jet Propulsion Laboratory, Pasadena, CA, 2000.

[2] T. N. Lal, M. Schröder, T. Hinterberger, J. Weston, M. Bogdan, N. Birbaumer, and B. Schölkopf. Support vector channel selection in BCI. *IEEE Transactions on Biomedical Engineering*, 51:1003-1010, 2004.

[3] G. Rätsch, S. Sonnenburg, J. Srinivasan, H.Witte, K.-R. Müller, R.J. Sommer, and B. Schölkopf. Improving the *Caenorhabditis elegans* genome annotation using machine learning techniques. *Submitted*, 2004.

[4] S. Romdhani, P. H. S. Torr, B. Schölkopf, and A. Blake. Efficient face detection by a cascaded support vector machine expansion. *Proceedings of the Royal Society, Series A, in press*, 2004.

[5] B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

[6] B. Schölkopf, K. Tsuda, and J.-P. Vert. *Kernel Methods in Computational Biology*. MIT Press, Cambridge, MA, 2004.

[7] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, New York, 1995.

[8] J.Weston, F. Perez-Cruz, O. Bousquet, O. Chapelle, A. Elisseeff, and B. Schölkopf. Feature selection and transduction for prediction of molecular bioactivity for drug design. *Bioinformatics*, 19:764.

