Kern Fisher Diskriminanten

Sebastian Mika
idalab GmbH & Fraunhofer FIRST
Mohrenstraße 63
10117 Berlin

email: mika@{first.fraunhofer.de, idalab.com}

1 Einleitung

Diese Zusammenfassung meiner Doktorarbeit (Mik02) beschäftigt sich mit statistischer Lerntheorie und statistischen Lernmaschinen. Eine statistische Lernmaschine versucht aus gegebenen Beispielen eine Regel zu inferieren mit der man in der Lage ist auf ungesehenen Beispielen korrekte Vorhersagen zu machen. Diese Vorhersagen sind reellwertig (Regression) oder diskret (Klassifikation). Wir betrachten insbesondere so genannte kernbasierte Lernverfahren. Die Hauptbeiträge dieser Arbeit lassen sich wie folgt zusammenfassen:

- Aufbauend auf der Theorie der reproduzierenden Kerne wird eine neue Lernmaschine vorgeschlagen, die auf der Maximierung eines Rayleigh Koeffizienten in einem Kernmerkmalsraum basieren. Dies wird beispielhaft für orientierte (Kern) Hauptkomponentenanalyse und insbesondere für Fishers Diskriminanten gemacht, was in Kern Fisher Diskriminanten (KFD) resultiert.
- Dann wird gezeigt, dass KFD eng mit quadratischer und linearer Optimierung verbunden ist. Darauf aufbauend werden verschiedene Möglichkeiten diskutiert mit den Optimierungsproblemen umzugehen die bei kernbasierten Methoden und insbesondere KFD entstehen.
- Die Formulierung als mathematisches Optimierungsproblem ist der Ausgangspunkt um verschiedene wichtige und interessante Varianten von KFD herzuleiten: Robuste KFD, sparse KFD und lineare KFD. Die mathematische Optimierung ermöglicht es darüber hinaus KFD mit anderen Techniken wie Support Vektor Maschinen, der Relevanzvektormethode und Boosting in Verbindung zu setzen. Durch einen strukturellen Vergleich der zugrundeliegenden Optimierungsprobleme wird illustriert, dass viele moderne Lernmethoden, auch KFD, sich sehr ähneln.
- Außerdem werden erste Ergebnisse über Lerngarantien für Eigenwerte und Eigenvektoren die aus Kovarianzmatrizen geschätzt werden präsentieren. Es wird gezeigen, dass unter schwachen Annahmen, die empirischen Eigenwerte mit hoher Wahrscheinlichkeit nahe an den zu erwartenden Eigenwerten liegen. Für Eigenvektoren

zeigen wir, dass mit hoher Wahrscheinlichkeit ein empirischer Eigenvektor nahe zu einem Eigenvektor der zugrundeliegenden Verteilung sein wird.

- In einer großen Sammlung von Experimenten wird demonstriert, dass KFD und seine Varianten sehr gut in der Lage sind mit dem technischen Standard zu konkurrieren. Es werden KFD mit Boosting und Support Vektor Maschinen verglichen und sorgfältig die Vor- und Nachteile der vorgeschlagenen Methoden diskutiert.

1.1 Maschinelles Lernen

Maschinelles Lernen, wie schon der Name suggeriert, versucht Algorithmen zu entwickeln um komplexe Probleme durch das Lernen einer Lösungen zu finden, nicht durch einen ingenieurmäßigen Entwurf. Letzteres ist für bestimmte Probleme oft schwierig. Ein aktuelles und sehr typisches Beispiel ist die Analyse des menschlichen Genoms. Im Prinzip, zumindest wird dies häufig vermutet, existiert ein festes Modell welches beschreibt, wie Information im Genom kodiert ist und wie diese Information in etwas nützliches umgesetzt wird. Es ist auch bekannt, dass nur Teile der DNS benutzt werden, die Gene. Aber es ist noch nicht zur Gänze verstanden, wo die Gene sind und wie man sie genau findet. Es existieren einige biologische Modelle aber diese sind sehr komplex und erklären nur zum Teil die Zusammenhänge. Maschinelles Lernen versucht dieses Problem auf eine andere Art zu lösen. Anstatt ein Modell zu suchen welches den zugrundeliegende Prozess erklärt und das dann benutzt werden kann um Antworten daraus abzuleiten versucht man eine Regel zu finden die in der Lage ist für jedes Stückchen DNS die Frage zu beantworten: ist hier ein Gen oder nicht? Allgemeiner gesprochen versucht maschinelles Lernen die Zusammenhänge zwischen Objekten zu modellieren. Die Kunst im maschinellen Lernen ist dies zu tun ohne dabei große Mengen an Expertenwissen zu brauchen.

Maschinelles Lernen ist ein Rahmenwerk mit dem sich eine große Zahl verschiedener Probleme lösen lassen, nicht nur ein spezifisches Problem. Die Komplexität des Ingenieur-Problems wird also verlagert von der Lösung eines schwierigen aber speziellen Problems zu der Lösung eines immer noch schwierigen aber allgemeineren Problems: Wie können Maschinen lernen? Der Vorteil ist klar: Wenn das Lernproblem einmal gelöst ist kann man eine große Menge andere Probleme mit den entwickelten Techniken ebenfalls lösen.

1.2 Lernen aus Beispielen

Es gibt viele verschiedene Gebiete in der künstlichen Intelligenz und maschinelles Lernen ist eines von ihnen. Das Paradigma dem diese Arbeit folgt ist das des “Lernen aus Beispielen”. Die Grundidee ist, dass oft wenige Beispiele von Relationen zwischen Objekten genügen um eine allgemeine Regel für diesen Zusammenhang herzuleiten. Das Lernproblem ist genau diese induktive Inferenz von der unvollständigen Information die durch die Beispiele gegeben wird zu einer Vorhersageregel die bestimmte Aspekte der Beobachtung beschreibt. Diese Regel werden meist durch Funktionen modelliert welche jedes Einga-

beobjekt auf ein Ausgabeobjekt abbilden. Je nach der Natur insbesondere der Ausgaben spricht man von Klassifikation oder Regression. Die Klassifikationaufgabe besteht darin jede Eingabe einer von endlich vielen Klassen zuzuordnen. Beim Regressionproblem will man jeder Eingabe eine oder mehrere kontinuierliche Ausgaben zuordnen. In beiden Fällen ist die einzige Information die man zur Verfügung hat eine endlich Menge von Eingabe-Ausgabe Paaren.

Während es etwas enttäuschen ist, dass sich im maschinellen Lernen fast alles um Funktionsschätzung dreht zeigen sich schnell die evidenten Vorteile dieser Herangehensweise. Eine rigorose mathematische Formulierung erlaubt es mittels Statistik und anderen Hilfsmitteln zu untersuchen unter welchen Bedingungen Lernen Erfolg haben kann. Das Gebiet der statistischen Lerntheorie (z.B. (Vap98)) untersucht welches die wichtigsten Eigenschaften von Lernalgorithmen und Funktionen sind und wie diese sich zu deren Fähigkeit verhalten bestimmte Schätzprobleme erfolgreich zu lösen. Damit wird es möglich Techniken herzuleiten die diese Größen optimieren und somit bessere Resultate erzielen. Darüberhinaus ist es der Theorie möglich Garantien über die maximale Anzahl von Fehlern abzugeben die eine bestimmte Methode in der Zukunft machen wird. Ein wichtiger Unterschied zwischen modernen Erkenntnissen im maschinellen Lernen und den frühen Bemühungen in der künstlichen Intelligenz ist, dass diese Garantien bereits für praktisch relevante Fälle gelten, nicht erst in irgendeinem asymptotischen Sinne.

Obwohl wir das Lernproblem auf das der Schätzung von Funktionen reduziert haben, muss immer noch ein komplexe Aufgabe gelöst werden. Abgesehen von theoretischen Limitierungen gibt es viele praktische Details, die Lernen schwierig machen, wobei das wichtigste die Tatsache ist, dass wir nur über eine limitierte Anzahl von Trainingsbeispielen verfügen. Das Problem ist die wahre Abhängigkeit zwischen Ein- und Ausgaben basierend auf dieser limitierten Information zu schätzen. Das Gegenteil kann aber auch ein Problem sein: Bei sehr großen Mengen von Trainingsdaten ist eine praktische Anwendung vieler Methoden nicht mehr möglich. Andere Probleme sind, dass die Trainingsdaten die das Problem beschreiben nicht vollständig sind, d.h. nicht alle Informationen sind vorhanden, oder dass die gegebene Information inhomogen ist, z.B. sich von Beispiel zu Beispiel etwas unterscheidet. Ein allgemeines Problem bei der Schätzung von Funktionen ist, dass man annehmen muss, dass die gegebene Information nicht genau und z.B. durch Messfehler verunreinigt ist.

Die aktuelle Forschung im Bereich des maschinellen Lernens beschäftigt sich sehr intensive mit genau diesen Problemen, d.h. wie macht man Lernen praktikabel mit sehr kleinen Datenmengen, wie geht man mit Rauschen und Ausreißern um oder wie behandelt man fehlende Werte. Insbesondere innerhalb der letzten zehn Jahre wurden hier große Fortschritte gemacht.

In dieser Arbeit betrachten wir Lernalgorithmen die auf so genannten Rayleigh Koeffizienten basieren, insbesondere Fishers Diskriminante, und solche die Kernfunktionen benutzen. Fishers Diskriminante (Fis36) ist eine Technik um lineare Funktionen zu finden die gut zwischen zwei oder mehr Klassen diskriminieren. Als eine Technik die es schon fast siebzig Jahre gibt ist sie sehr bekannt und wird viel benutzt um Klassifikatoren zu konstruieren. Allerdings sind viele aktuelle Lernprobleme nicht hinreichend lösbar mit linearen Techniken. Deshalb schlagen wir eine nichtlineare Variante von Fishers Diskrimi-

nanten vor. Die "Nichtlinearisierung" wird mögliche durch den Einsatz von so genannten Kernfunktionen, eine Technik die von den Support-Vektor-Maschinen entlehnt ist (Vap98). Kernfunktionen stellen einen allgemeinen und eleganten Weg dar um nichtlineare Algorithmen zu formulieren. Die hergeleiteten Methoden haben eine klare und intuitive Interpretation. Darüberhinaus, obwohl die Techniken in der Lage sind hoch komplexe, nichtlineare Beziehungen zu modellieren ist es möglich Algorithmen anzugeben die eine global optimale Lösung in polynomieller Zeit finden. Es wird gezeigt, dass solche Lernmaschinen eine dem Stand der Technik entsprechende Leitung haben.

2 Kern Fisher Diskriminanten

Das Ziel von Diskriminanten Analyse kann man so zusammenfassen, dass man eine Funktion sucht, die einen Wert zurückgibt der eine gute Differenzierung zwischen verschiedenen Klassen ermöglicht. Formell sucht man eine Funktion $f : \mathcal{X} \to \mathbb{R}^D$, so dass $f(\mathbf{x})$ und $f(\mathbf{z})$ ähnlich sind, wenn $\mathbf{x}$ und $\mathbf{z}$ ähnlich sind, d.h. aus derselben Klasse, und unterschiedlich sonst. In dem speziellen Fall von lineare Diskriminanten sucht man eine linear Funktion, d.h. eine Menge von Projektionen

$$f(\mathbf{x}) = W^\top \mathbf{x}, \qquad W \in \mathbb{R}^{N \times D},$$

wobei die Matrix W so gewählt ist, dass ein Kontrast-Kriterium G optimiert wird, ggf. unter ein paar Nebenbedingungen $\mathcal{S}$:

$$\max G(W) \text{ wobei } W \in \mathcal{S}. \tag{1}$$

Die verschiedenen Techniken um lineare Diskriminanten zu bestimmen unterscheiden sich nur in der spezifischen Wahl von G und $\mathcal{S}$. Um die Präsentation zu vereinfachen werden im folgenden nur eindimensionale Diskriminanten untersuchen, d.h. f ist von der Form $f = (\mathbf{w} \cdot \mathbf{x})$. Allerdings lassen sich die meisten Ergebnisse einfach auf den mehrdimensionalen Fall erweitern.

Eine der bekanntesten linearen Diskriminante ist Fishers Diskriminante (Fis36). Fishers Idee war, dass man nach einer Richtung $\mathbf{w}$ sucht, die die Klassenmittel, wenn man sie auf $\mathbf{w}$ projiziert gut separiert und dabei gleichzeitig eine kleine Varianz um diese Mittelwerte hat. Dies ist in Abbildung 1 dargestellt. Die Annahme ist, dass es dann einfach ist sich für eine der beiden Klassen zu entscheiden ohne viele Fehler zu machen. Die Größe die den Abstand der Mittelwerte misst wird interklassen Varianz genannt, die Größe, die die Varianz um diese Mittelwerte misst intraklassen Varianz. Dann ist das Ziel eine Funktion zu finden die die interklassen Varianz maximiert und gleichzeitig die intraklassen Varianz minimiert.

Mathematisch lässt sich dies wie folgt beschreiben. Sei $\mathcal{X}$ der Raum unserer Beobachtungen (z.B. $\mathcal{X} \subseteq \mathbb{R}^N$) und $\mathcal{Y}$ eine Menge von möglichen Ausgaben (hier $\mathcal{Y} = \{+1, -1\}$). Desweiteren sei $\mathcal{Z} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_M, y_M)\} \subseteq \mathcal{X} \times \mathcal{Y}$ die Menge der Trainingsbeispiele und $\mathcal{Z}_1 = \{(\mathbf{x}, y) \in \mathcal{Z} | y = 1\}$ und $\mathcal{Z}_2 = \{(\mathbf{x}, y) \in \mathcal{Z} | y = -1\}$ deren Einteilung in zwei Klassen der Größe $M_i = |\mathcal{Z}_i|$. Dann definieren wir noch $\mathbf{m}_1$ und $\mathbf{m}_2$ als die

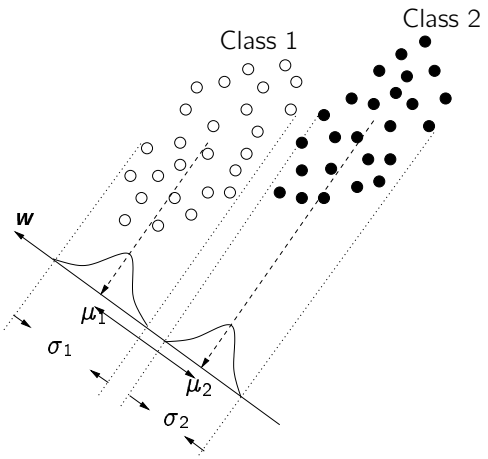

Abbildung 1: Illustration von Fishers Diskriminante. Man sucht eine Richtung $\mathbf{w}$, so dass die Differenz zwischen den auf diese Richtung projizierten Klassenmitteln (μ_1 und μ_2) groß ist und gleichzeitig die Varianz um diese Mittel (σ_1 und σ_2) klein ist.

empirischen Klassen-Mittelwerte, d.h.[1]

$$\mathbf{m}_i = \frac{1}{M_i} \sum_{\mathbf{x} \in \mathcal{Z}_i} \mathbf{x}.$$

Auf ähnliche Weise kann man die Mittelwerte der Daten wenn man sie auf eine Richtung $\mathbf{w}$ projiziert berechnen:

$$\begin{aligned} \mu_i &= \frac{1}{M_i} \sum_{\mathbf{x} \in \mathcal{Z}_i} \mathbf{w}^\top \mathbf{x} \\ &= \mathbf{w}^\top \mathbf{m}_i, \end{aligned} \tag{2}$$

d.h. die Mittelwerte μ_i der Projektionen sind die projizierten Mittelwerte. Die Varianz [2] σ_1, σ_2 der projizierten Daten kann geschrieben werden als

$$\sigma_i = \sum_{x \in \mathcal{Z}_i} (\mathbf{w}^\top \mathbf{x} - \mu_i)^2. \tag{3}$$

Dann kann man die interklassen Varianz maximieren und die intraklassen Varianz minimieren indem man

$$G(\mathbf{w}) = \frac{(\mu_1 - \mu_2)^2}{\sigma_1 + \sigma_2}, \tag{4}$$

maximiert. Dies ergibt eine Richtung $\mathbf{w}$, so dass das Verhältnis von inter- zu intraklassen Varianz maximal ist. Wenn man nun die Ausdrücke (2) für die Mittelwerte und (3) für die

[1] Unter Missbrauch der Notation schreiben wir das $\mathbf{x}$ in $(\mathbf{x}, y) \in \mathcal{Z}$ als $\mathbf{x} \in \mathcal{Z}$.

[2] Genaugenommen ist σ_i die unnormalisierte Varianz welche manchmal auch Scatter genannt wird.

Varianzen in (4) einsetzt, ergibt dies

$$G(\mathbf{w}) = \frac{\mathbf{w}^\top S_B \mathbf{w}}{\mathbf{w}^\top S_W \mathbf{w}}, \tag{5}$$

wobei wir die inter- und intraklassen Scatter Matrizen S_B und S_W als

$$S_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^\top \quad S_W = \sum_{i=1,2} \sum_{\mathbf{x} \in \mathcal{Z}_i} (\mathbf{x} - \mathbf{m}_i)^2, \tag{6}$$

definieren. Es ist einfach zu prüfen, dass (4) equivalent zu (5) ist. Die Größe in Gleichung (5) wird als Rayleigh Koeffizient bezeichnet.

2.1 Berechnung von $\mathbf{w}$

Eine besonders gute Eigenschaft von Fishers Diskriminante ist, (i) dass (5) ein globales Optimum hat, und (ii) dass ein global optimales $\mathbf{w}$ welches (5) maximiert analytisch durch die Lösung eines Eigenwertproblems bestimmt werden kann. Es ist allgemein bekannt, dass das $\mathbf{w}$ welches (5) maximiert der erste Eigenvektor des generalisierten Eigenwertproblems

$$S_B \mathbf{w} = \lambda S_W \mathbf{w}, \tag{7}$$

ist. Betrachtet man dieses Eigenproblem näher, stellt man fest, dass es einen noch einfacheren Weg gibt, dass optimale $\mathbf{w}$ zu finden. Ohne hier ins Detail zu gehen, lässt sich zeigen, dass folgender Ausdruck zu einer äquivalenten Lösung führt:

$$\mathbf{w} = S_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1),$$

d.h. man kann die optimale Richtung $\mathbf{w}$ finden, indem man die intraklassen Scatter Matrix S_W invertiert.

2.2 Einführen von Kernen

Der Kern Trick, welcher hier nur heuristisch beschrieben werden soll, besteht darin einen linearen Algorithmus anstatt im Eingaberaum in einem Merkmalsraum $\mathcal{E}$ anzuwenden. Aber anstatt dies dadurch zu erreichen, dass man die Daten explizit in den Raum $\mathcal{E}$ abbildet (durch eine Abbildung $\Phi : \mathcal{X} \rightarrow \mathcal{E}$) tut man dies implizit indem man in $\mathcal{E}$ lediglich Skalarprodukte berechnet, d.h. $(\Phi(\mathbf{x}) \cdot \Phi(\mathbf{z}))$. Es existieren nämlich bestimmte Abbildung Φ die in sehr große und mächtige Merkmalsräume abbilden für die sich solch ein Skalarprodukt trotzdem einfach berechnen lässt: durch die so genannte Kernfunktion k. Man hat, dass $\mathrm{k}(\mathbf{x}, \mathbf{z}) = (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{z}))$.

Wir suchen jetzt also nach einer Diskriminante der Form

$$f(\mathbf{x}) = \mathbf{w}^\top \Phi(\mathbf{x}),$$

wobei $\mathbf{w} \in \mathcal{E}$ jetzt ein Vektor im Merkmalsraum $\mathcal{E}$ ist. Der Kern Trick besteht darin den zuvor linearen Algorithmus so zu formulieren, dass $\Phi(\mathbf{x})$ nur noch in Skalarprodukten vorkommt. Diese werden dann durch die Kernfunktion ersetzt.

Für Fishers Diskriminanten ist dies besonders einfach. Um die lineare Diskriminante im Merkmalsraum $\mathcal{E}$ zu finden (welche dann nicht linear im Eingaberaum ist) muss

$$J(\mathbf{w}) = \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}},$$

maximiert werden, wobei jetzt $\mathbf{w} \in \mathcal{E}$ und S_B und S_W die Matrizen (6) sind aber mit $\Phi(x)$ anstatt $\mathbf{x}$. Man kann nun zeigen, dass sich dieser Ausdruck so umformulieren lässt, dass die $\Phi(\mathbf{x})$ nur noch in Skalarprodukten vorkommen und dementsprechen durch Kernfunktionen ersetzt werden können. Zunächst kann man zeigen, dass sich $\mathbf{w}$ schreiben lässt als

$$\mathbf{w} = \sum_{i=1}^{M} \alpha_i \Phi(\mathbf{x}_i), \tag{8}$$

d.h. als Linearkombination der in den Merkmalsraum abgebildeten Trainingsdaten. Ohne hier ins Detail zu gehen, kann man weiter zwei Matrizen N und M der Größe $M \times M$ definieren, so dass sich die optimalen Koeffizienten α_i finden lassen durch

$$J(\alpha) = \frac{\alpha^T M \alpha}{\alpha^T N \alpha}.$$

Man muss also immer noch einen Rayleigh Koeffizienten optimieren, bekommt aber eine nichtlineare Lösung.

Der Vorteil dieser impliziten Rechnung in einem Merkmalsraum liegt darin, dass man auch in extrem hochdimensionalen oder gar unendlich dimensionalen Räumen arbeiten kann. Außerdem ist es möglich durch die Wahl des Kernes Expertenwissen mit einzubauen.

2.3 Weiterführende Themen

Das soeben dargelegte ist allerdings nur der Ausgangspunkt für viele weitere Untersuchung in der zugrundeliegenden Arbeit (Mik02). Um KFD wirklich praktisch und leistungsfähig zu machen ist es wichtig die Komplexität der Lösungen genau zu kontrollieren. Dazu werden verschiedene Regularisierungsstrategien vorgeschlagen und untersucht. Weiter lassen sich für KFD interessante Zusammenhänge zu der Methode der kleinsten Quadrate herstellen. Diese wiederrum führt zu einer Reformulierung von KFD als ein mathematischen Optimierungsproblem. Diese Formulierung von Lernalgorithmen als mathematische Optimierungsprobleme wird näher untersucht, mit dem Ergebnis, dass viele moderne Methoden strukturell sehr ähnlich sind (z.B. Support Vektor Maschinen, Relevance Vektor Maschinen (Tip00), Arc-GV und natürlich KFD). Dies mag auch eine Erklärung dafür sein, warum diese Methoden in der Praxis alle eine ähnlich gute Performance liefern.

Aufbauend auf der Formulierung als mathematisches Optimierungsproblem werden verschiedene, interessante Varianten von KFD entwickelt. Es wird die sparse KFD (SKFD)

vorgeschlagen, die er erlaubt den Lösungsvektor $\mathbf{w}$ mit einem Bruchteil der Trainingsdaten zu beschreiben, d.h. in der Expansion (8) sind viele α_i Null. Dies lässt sich z.B. im Entwurf von Optimierungsstrategien ausnutzen, hat aber auch den Vorteil, dass die Auswertung schneller wird. Um die Methoden robuster gegen Fehler in der Daten zu machen wird die robuste bzw. lineare KFD vorgeschlagen. Schließlich lassen sich die Vorteile der sparsen Lösungen und der Robustheit auch kombinieren, was zur linearen, sparsen KFD führt (LSKFD).

Im folgenden werden dann verschiedenste Strategien entwickelt und untersucht um die resultierenden Optimierungsproblem effizient zu lösen. Diese reichen von einfachen Heuristiken über iterative Verfahren bis hin zu ausgeklügelten Modifikationen von mathematischen Optimierungsproblemen.

Schließlich wird in einem separaten Abschnitt eine Theorie entwickelt, die es erlaubt Fehlerschranken für KFD und ähnliche Methoden herzuleiten. Diese Theorie basiert auf neuesten Erkenntnissen aus der Statistik, wie z.B. Stabilitätsschranken und so genannten Luckiness Funktionen. Deren Darstellung würde hier jedoch bei weitem den Rahmen sprengen.

2.4 Empirische Evaluation

In diesem Abschnitt sind einige empirische Ergebnisse für Kern Fisher Diskriminanten zusammengefasst. Um die Leistungsfähigkeit der verschiedenen KFD Varianten zu evaluieren wurden sie auf mehreren Benchmark Datensätzen mit anderen Techniken verglichen. Es wurden KFD, Support-Vektor-Maschinen, RBF-Netzwerke (z.B. MD89), AdaBoost (FS97), und regularisiertes AdaBoost (Rät01) verglichen (siehe Tabelle 1 und 2).

Tabelle 1: Vergleich zwischen KFD, RBF Netzen, AdaBoost (AB), regularisiertem AdaBoost (AB_R) und SVMs. Beste Methode fett, zweit beste kursiv. Gezeigt wird der Generalisierungsfehler der zehn Datensätze gemittelt über 100 Wiederholungen und die Standardabweichung des Mittels.

	RBF	AB	AB_R	SVM	KFD
Banana	**10.8±0.06**	12.3±0.07	*10.9±0.04*	11.5±0.07	**10.8±0.05**
B.Cancer	27.6±0.47	30.4±0.47	26.5±0.45	*26.0±0.47*	**25.8±0.46**
Diabetes	24.3±0.19	26.5±0.23	23.8±0.18	*23.5±0.17*	**23.2±0.16**
German	24.7±0.24	27.5±0.25	24.3±0.21	**23.6±0.21**	*23.7±0.22*
Heart	17.6±0.33	20.3±0.34	16.5±0.35	**16.0±0.33**	*16.1±0.34*
Ringnorm	1.7±0.02	1.9±0.03	*1.6±0.01*	1.7±0.01	**1.5±0.01**
F.Sonar	34.4±0.20	35.7±0.18	34.2±0.22	**32.4±0.18**	*33.2±0.17*
Thyroid	4.5±0.21	*4.4±0.22*	4.6±0.22	4.8±0.22	**4.2±0.21**
Titanic	23.3±0.13	*22.6±0.12*	*22.6±0.12*	**22.4±0.10**	23.2±0.20
Waveform	10.7±0.11	10.8±0.06	**9.8±0.08**	*9.9±0.04*	*9.9±0.04*
Durchschnitt	18.0%	20.2%	17.5%	17.2%	17.2%

Tabelle 2: Vergleich zwischen SVM, KFD, sparser KFD (SKFD) und linearer, sparser KFD (LSKFD). Bestes Ergbnis in fett, zweitbestes kursiv.

	SVM	KFD	SKFD	LSKFD
Banana	11.5±0.07	*10.8±0.05*	11.2±0.48	**10.6±0.04**
B.Cancer	26.0±0.47	*25.8±0.46*	**25.2±0.44**	*25.8±0.47*
Diabetes	23.5±0.17	*23.2±0.16*	**23.1±0.18**	23.6±0.18
German	**23.6±0.21**	*23.7±0.22*	**23.6±0.23**	24.1±0.23
Heart	**16.0±0.33**	*16.1±0.34*	16.4±0.31	**16.0±0.36**
Ringnorm	1.7±0.01	**1.5±0.01**	*1.6±0.01*	**1.5±0.01**
F.Sonar	**32.4±0.18**	*33.2±0.17*	33.4±0.17	34.4±0.23
Thyroid	4.8±0.22	**4.2±0.21**	*4.3±0.18*	4.7±0.22
Titanic	**22.4±0.10**	23.2±0.20	22.6±0.17	*22.5±0.20*
Waveform	**9.9±0.04**	**9.9±0.04**	*10.1±0.04*	10.2±0.04
Durchschnitt	17.2%	17.2%	17.2%	17.3%

Das Ergebnis dieser Evaluation ist, dass KFD und seine Varianten SKFD und LSKFD mit allen modernen Methoden mithalten können. Insbesondere die sparse KFD und lineare, sparse KFD zeigen eine beeindruckende Performance.

Die empirische Evaluation der vorgeschlagenen Algorithmen zeigt, dass es mit diesen möglich ist die KFD Probleme schnell und effizient zu lösen.

3 Zusammenfassung

In der dieser Kurzfassung zugrundeliegenden Doktorarbeit wurden Lernmethoden die auf der Maximierung eines Rayleigh Koeffizienten beruhen untersucht. Es wurden nichtlineare Verallgemeinerungen von verschiedenen Methoden vorgeschlagen, unter anderem orientierter Hauptkomponentenanalyse und insbesondere Fishers Diskriminanten.

Zentraler Aspekt der Arbeit ist die Anwendung des “Kerntricks” auf Rayleigh Koeffizienten bei gleichzeitiger Berücksichtigung der Komplexitätskontrolle im Rahmen der strukturellen Risikominimierung. Es wurde gezeigt, wie auf diesem Wege neue, machtvolle Algorithmen hergeleitet werden können deren Leistung dem heutigen Stand der Technik entspricht.

In einem weiteren Teil wurde gezeigt, dass KFD als ein mathematisches (quadratisches) Optimierungsproblem formuliert werden kann. Aufbauend auf dieser Einsicht wird diskutiert und aufgezeigt, wie mathematische Optimierung als ein allgemeines Rahmenwerk für die Analyse von Lernverfahren dienen kann. Außerdem erlaubt diese Betrachtung die Herleitung mehrerer interessanter und nützlicher Varianten von KFD: robuste KFD, sparse KFD und lineare, sparse KFD. Schließlich wird diskutiert wie die den Lernproblemen zu Grunde liegenden Optimierungsprobleme effizient gelöst werden können.

Um die Leistungsfähigkeit der vorgeschlagenen Algorithmen zu illustrieren und sie mit anderen Techniken zu vergleichen wird eine große Anzahl von experimentellen Resultaten

präsentiert. Dabei werden sowohl künstliche als auch reale Daten verwandt.

Zusammenfassend lässt sich sagen, das gezeigt wurde, dass Fishers Diskriminanten durch Nutzung von Kernen zu den besten heute verfügbaren Lernmethoden zählen. Ihre intuitive Interpretation, die Eigenschaft, dass Resultate erzeugt werden welche sich als Wahrscheinlichkeiten interpretieren lassen und ihre einfach Umsetzung machen sie für viele Anwendungen interessant. Andererseits wurde auch gezeigt, dass die meisten modernen Lernmethoden, abgesehen davon, dass sie sehr ähnliche Optimierungsprobleme lösen, kaum Unterscheide in ihrer Leitung zeigen. Es wäre sicher falsch aus dieser Arbeit den Schluss zu ziehen, dass KFD besser ist als andere Techniken. Aber KFD ist sicher genauso gut wie andere existierende Methoden. Und wie mit jeder Technik gibt es bestimmte Anwendungen wo KFD besonders geeignet ist.

Literatur

[Fis36] R.A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[FS97] Y. Freund and R.E. Schapire. A Decision-theoretic Generalization of On-line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[MD89] J. Moody and C. Darken. Fast learning in networks of locally-tuned processing units. *Neural Computation*, 1(2):281–294, 1989.

[Mik02] S. Mika. *Kernel Fisher Discriminants*. PhD thesis, University of Technology, Berlin, Germany, December 2002.

[Rät01] G. Rätsch. *Robust Boosting via Convex Optimization*. PhD thesis, University of Potsdam, Neues Palais 10, 14469 Potsdam, Germany, October 2001.

[Tip00] M.E. Tipping. The Relevance Vector Machine. In S.A. Solla, T.K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems*, volume 12, pages 652–658. MIT Press, 2000.

[Vap98] V.N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

Sebastian Mika, geboren 1973, studierte an der Technischen Universität Berlin Informatik und Mathematik. Er hat 1998 sein Diplom in Informatik mit Auszeichnung erhalten. Von 1998 an hat Herr Mika an der Technischen Universität und dem Fraunhofer Institut FIRST gearbeitet. Neben einer Vielzahl wissenschaftlicher Publikationen und mehrere Auslandsaufenthalten, unter anderem bei AT&T Research, Microsoft Research und der Australian National University, entstand seine Doktorarbeit. Im Dezember 2002 hat Herr Mika seine Promotion mit Auszeichnung an der Technischen Universität abgelegt.