Anwendung der Regressions-SVM zur Vorhersage studentischer Leistungen

Alexander Askinadze
Institut für Informatik
Heinrich-Heine-Universität Düsseldorf
Universitätsstr. 1
40225 Düsseldorf, Deutschland
alexander.askinadze@hhu.de

ABSTRACT

Um als Bildungsanbieter bei gefährdeten Studenten rechtszeitig intervenierend eingreifen zu können, sind Verfahren zur Vorhersage studentischer Leistungen notwendig. Viele Arbeiten haben den Einsatz des SVM-Klassifikators vorgeschlagen. Allerdings wurden unzureichende Angaben zur Wahl eines geeigneten Kernel gegeben. Außerdem kann der SVM-Klassifikator bei fehlenden Trainingsdaten zu allen möglichen Noten nicht erfolgreich trainiert werden. Zur Lösung dieser Probleme untersuchen wir die Regressions-SVM mit verschiedenen geeigneten Kernel. Dabei erreichen wir mit dem RBF-Kernel und einer σ-Parameter Heuristik auf einem öffentlichen Datensatz eines Mathematikkurses bessere Ergebnisse als in [3] mit einer SVM erreicht wurden. Für den Fall, dass zusätzlich zu den privaten Daten der Studenten auch vorherige Noten bekannt waren, konnte die Vorhersage von „bestanden oder nicht bestanden" mit einer Genauigkeit von 90.57% erreicht werden. Das ermöglicht eine praktische Anwendbarkeit der Regressions-SVM zur Erkennung gefährdeter Studenten.

Categories and Subject Descriptors

K.3.1 [**Computers and Education**]: Computer Uses in Education; H.2.8 [**Database Applications**]: Data Mining

Keywords

Education data mining, learning analytics, student performance prediction, support vector machines, regression, kernel

1. EINFÜHRUNG

Die Leistungen von Studenten sind ein wichtiger Faktor für Bildungseinrichtungen. Mit Hilfe der erreichten Noten der Schüler und Studenten wird entschieden, ob ein Fach oder gar eine Abschlussarbeit bestanden wurde. Da die Noten einen Einfluss auf das erfolgreiche Absolvieren der Schule oder des Studiums haben, haben die Bildungseinrichtungen unter anderem aus Vergleichszwecken und finanziellen Gründen Interesse daran, möglichst viele Lernende zu einem erfolgreichen Abschluss zu führen. Eine automatische Vorhersage über die Leistungen der einzelnen Studenten kann daher helfen, rechtzeitig bei gefährdeten Studenten einzugreifen.

Die Vorhersage der studentischen Leistungen ist eines der beliebtesten Themen innerhalb des junges Forschungsgebietes „Education Data Mining" (EDM). Bei dem Begriff „Data Mining" handelt es sich grob um eine Disziplin, die sich mit der Extraktion von impliziten Mustern aus großen Datenbeständen beschäftigt. Hierbei werden statistische Verfahren und Algorithmen aus dem maschinellen Lernen verwendet. EDM kann laut der internationalen EDM-Gesellschaft[1] so definiert werden:

> Aufkommende Disziplin, die sich mit der Entwicklung von Methoden zur Erforschung der Daten aus Bildungsumgebungen befasst und diese Methoden einsetzt, um Studenten und ihre Lernumgebungen besser zu verstehen.

Einer der beliebtesten Algorithmen aus dem Bereich Data Mining ist die Support Vector Machine (SVM). Dieser Klassifikator lässt sich auch für Multiklassen-Probleme, wie sie bei der Noten-Vorhersage gegeben sind, verwenden. Wird jede mögliche Note als Klasse betrachtet, so benötigt die SVM für jede Klasse Trainingsdaten. Bei kleinen Trainingsmengen mit einer großen Anzahl an möglichen Noten (beispielsweise bei einer Notenskala von 1-20) kann es vorkommen, dass es zu einigen Noten keine Trainingsdaten gibt, sodass die SVM mit den üblichen Multiklassen-Ansätzen one-vs-one oder one-vs-all nicht trainiert werden kann. Hierfür eignet sich der Einsatz einer Regressions-SVM, welche auch ohne Existenz aller nötigen Trainingsdaten in der Lage ist alle Notenstufen zu approximieren.

Im Kapitel 2 werden verschiedene Arbeiten vorgestellt, welche die SVM zur Vorhersage von studentischen Leistungen eingesetzt haben. Die Ergebnisse zeigen, dass die SVM sich gut für diese Aufgabe eignet. Im Kapitel 3 werden die theoretischen Hintergründe von SVM und SVM-Kernel, sowie die für weitere Versuche notwendige Regressions-SVM vorgestellt. Das Kapitel 4 stellt den in der Evaluation verwendeten Datensatz mit allen Attributen vor. Im Kapitel 5 werden die untersuchten Klassifikations- und Regressionsprobleme erläutert und die verwendeten Evaluationsmaße dargestellt. Im Kapitel 6 wird auf einer öffentlichen Da-

[1] `http://www.educationaldatamining.org/`

28^{th} GI-Workshop on Foundations of Databases (Grundlagen von Datenbanken), 24.05.2016 - 27.05.2016, Nörten-Hardenberg, Germany.

tenbank [3] mit privaten Attributen, Zwischennoten und Abschlussnoten von portugiesischen Schülern untersucht, wie mit geeigneten SVM-Kernel und geeigneter Kernel-Parameter-Auswahl bessere SVM-Ergebnisse als in [3] erreicht werden können. Schließlich wird im Kapitel 7 ein Fazit gezogen.

2. RELATED WORK

Es gibt bereits viele Arbeiten, die verschiedene Klassifikatoren wie Decision Tree (DT), Random Forest (RF), kNN, Neuronale Netze (NN) und SVM zur Vorhersage studentischer Leistungen verwendet haben. In diesem Kapitel untersuchen wir eine Auswahl der Arbeiten zur Vorhersage von studentischen Leistungen, welche die SVM verwendet oder diese mit anderen Klassifikatoren verglichen haben [1, 3, 6, 8, 9, 10, 13, 14, 15, 16, 19]. Eine Zusammenfassung der Arbeiten ist in Tabelle 1 dargestellt. Die Klassifikationsergebnisse der untersuchten Arbeiten sind schwer miteinander zu vergleichen, da diese auf unterschiedlichen und nicht standardisierten Datensätzen mit verschiedenen Eigenschaften durchgeführt wurden.

Die Arbeiten [6, 9] nutzen die SVM, um aus Daten zur studentischen Forum-nutzung die Abschlussnoten vorhersagen zu können. In beiden Arbeiten lieferten die SVM-Ergebnisse im Vergleich zu anderen Klassifikatoren gute Ergebnisse, wobei in beiden Arbeiten nicht erwähnt wird, welcher SVM-Kernel und welche entsprechenden Parameter verwendet wurden.

In der Arbeit [10] wurde eine spezielle SVM-Variante, die SSVM [5] zur Vorhersage der studentischen Leistungen eingesetzt. Die SVM hat hier gute Ergebnisse zur Vorhersage von Studenten mit sehr guten und Studenten mit sehr schlechten Leistungen geliefert (über 90% Genauigkeit). Als SVM-Kernel wurde der RBF-Kernel mit zwei festen Parametern trainiert, wobei nicht erläutert wird, wie die Parameter des RBF-Kernels ausgewählt wurden.

Die Arbeit [1] untersucht ebenfalls die SVM zur Vorhersage studentischer Leistungen. Zwar hat die SVM im Vergleich die besten Ergebnisse, jedoch wird von den Autoren der Einsatz von Entscheidungsbäumen empfohlen. Dies wird damit begründet, dass die Entscheidungsbäume nicht signifikant schlechtere Ergebnisse liefern und gleichzeitig besser verständlich sind. Hier wird keine Angabe zur Wahl des SVM Kernels gegeben.

In der Arbeit [16] hat die SVM im Vergleich zu einem Entscheidungsbaum- und einem NN-Klassifikator ebenfalls die besten Ergebnisse zur Vorhersage des Notendurchschnitts geliefert. Auch hier wurden keine Angaben zur Wahl des Kernels und der entsprechender Parameter gemacht.

In [19] wird der Einsatz der SVM zur Vorhersage der Abschlussarbeitsnote verwendet. Die SVM wird hier mit neuronalen Netzen, Entscheidungsbäumen und Naive Bayes verglichen und liefert die besten Ergebnisse. Auch in dieser Arbeit wird nichts zur Wahl des SVM-Kernels angegeben.

Die Arbeit [13] hat verschiedene Klassifikatoren wie kNN, Entscheidungsbäume und SVM zur Vorhersage von studentischen Leistungen verglichen. Hierbei hat die SVM sowohl das Problem „bestehen oder durchfallen“, als auch für die Regression der Noten vergleichsweise die besten Ergebnisse geliefert. In dieser Arbeit wird ebenfalls nicht näher auf den eingesetzten SVM-Kernel eingegangen.

In [15] wurde untersucht, wie sich bei der Vorhersage der Leistungen, im Fall eines Klassenungleichgewichts (z. B. gibt es mehr Studenten, die durchfallen, als Studenten, die bestehen) verbessern lässt. Die SVM hat hierbei in Kombination mit den vorgestellten Methoden zur Lösung des Klassenungleichgewichts vergleichsweise gute Ergebnisse geliefert. Als SVM-Kernel wurde der Polynom- und der RBF-Kernel verwendet. Zur Parameterwahl der Kernel wurde die grid search verwendet.

In [14] werden die Klassifikatoren SVM, NN, ELM (extreme learning machine) zur Vorhersage der Durchschnittsnote verglichen. Der Autor zeigt, dass mit der SVM die besten Werte erreicht werden. Es wurde der RBF-Kernel mit einer grid search zur Auswahl der Kernel-Parameter verwendet.

Anders als bei den oben genannten Arbeiten kommen die Autoren in [8] zu dem Ergebnis, dass der MLP-Klassifikator (Mult-Layer-Perceptron) im Vergleich zur SVM bessere Ergebnisse zur Vorhersage der studentischen Leistungen liefert. Auch hier wird nicht erläutert, welcher SVM-Kernel verwendet wurde.

Die Arbeit [3] untersucht den Einsatz von DT, NN, RF und SVM zur Vorhersage von gefährdeten Studenten und der Vorhersage von Notenstufen. Die Autoren kommen dabei zum Ergebnis, dass die Entscheidungsbaum-Klassifikatoren bessere Ergebnisse liefern als die SVM. Als SVM-Kernel wird der RBF-Kernel mit einer grid search zur Auswahl der Parameter verwendet.

Der Tabelle 1 können wir entnehmen, dass die SVM in vielen Arbeiten erfolgreich eingesetzt wurde und die Wahl der SVM-Kernel oder die Wahl der zugehörigen Kernel-Parameter oftmals nicht angegeben wird. Die Unklarheit über die Wahl geeigneter Kernel und das Problem, dass es keine Trainingsdaten zu einigen Noten geben könnte, motiviert die Untersuchung der Regressions-SVM und zugehöriger geeigneter Kernel zur Vorhersage der studentischen Leistungen.

3. SVM

In diesem Kapitel werden wir die theoretischen Hintergründe von Regressions-SVM und Kernel beleuchten, da diese die zentralen Aspekte dieser Publikation darstellen.

Die SVM [2] ist ein binärer Klassifikator $f : \mathbb{R}^n \rightarrow \{-1, +1\}$, der für zwei linear trennbare Punktmengen eine Trenn-Hyperebene findet.

Die Klassifikation eines neues Punktes x^* auf eine der beiden Klassen $\{+1, -1\}$ kann mit Hilfe einer gefundenen Hyperebene in $\mathbb{R}^n$ mit dem Vektor $w \in \mathbb{R}^n$ und $b \in \mathbb{R}$ folgenderweise ausgedrückt werden:

$$f(x^*) = \text{signum}(w^T x^* - b) = \begin{cases} +1, & \text{wenn } w^T x^* > b \\ -1, & \text{wenn } w^T x^* < b \end{cases} \quad (1)$$

Die Trainingsmenge der SVM sei mit $\{(x_1, y_1), ..., (x_L, y_L) | x_i \in \mathbb{R}^n, y_i \in \{-1, +1\}\}$ angegeben, wobei x_i die Trainingspunkte und y_i die zugehörigen Klassen sind. Um die Hyperebene zu finden, muss für die L Trainingstupel das folgende Optimierungsproblem gelöst werden [17]:

$$\begin{aligned} &\min \frac{||w||^2}{2} \\ &\text{u.d.N. } y_i(w^T x_i - b) - 1 \geq 0 \ \forall i \in \{1, .., L\} \end{aligned} \quad (2)$$

Dieses „Quadratic Programming” (QP)-Problem lässt sich in ein äquivalentes Problem mit linearen Nebenbedingungen

Tabelle 1: Übersicht über bisherige Publikationen zur Vorhersage von studentischen Leistungen mit SVM

Publikation	Jahr	Vorhersage von	Daten	SVM-Ergebnisse unter den besten Ergebnissen?	Kernel	Kernel-parameter
[19]	2015	Abschlussarbeitsnote	Durchschnittsnoten keine genauen Angaben	ja	k.A.	k.A.
[13]	2015	Bestehen, nicht bestehen Abschlussnote	pers. Daten und bish. Studienlaufbahn	ja	k.A.	k.A.
[14]	2014	Abschluss-Durchschnittsnote	Durchschnittsnoten der ersten 3 Jahre	ja	RBF	grid search
[1]	2014	Abschlussnote	pers. Daten und vorherige Noten	ja	k.A.	k.A.
[16]	2014	Durchschnittsnote	pers. Daten und vorherige Noten	ja	k.A.	k.A.
[8]	2013	Examensnote	pers. Daten und bish. Noten	nein	k.A.	k.A.
[6, 9]	2012, 2013	Abschlussnote	Forum-Nutzungsdaten	ja	k.A.	k.A.
[10]	2011	Abschlussnote	„psychometric factors“	ja	RBF	k.A.
[15]	2009	Bestehen, nicht bestehen Absolventenquote	versch. Datensätze	ja	Polynom RBF	grid search
[3]	2008	Bestehen, nicht bestehen Notenstufen Abschlussnote	pers. Daten und bish. Noten	nein	RBF	grid search

(NB) umschreiben, dass einfacher gelöst werden kann:

$$\begin{aligned} &\max \sum_{i=1}^{L} \alpha_i - \frac{1}{2} \sum_{i=1}^{L} \sum_{j=1}^{L} \alpha_i \alpha_j y_i y_j x_i^T x_j \\ &\text{u.d.N. } \alpha_i \geq 0, \sum_{i=0}^{L} \alpha_i y_i = 0 \end{aligned} \tag{3}$$

Dieses QP-Problem lässt sich mit dem SMO-Verfahren (Sequential Minimal Optimization) [7] auch für große Trainingsdaten effizient lösen, indem ein großes QP-Problem in eine Reihe von kleinen QP-Problemen zerlegt wird.

3.1 Nicht separierbarer Fall

Ist die Trainingsmenge nicht linear separierbar, so ist eine Abbildung $\phi(x) : \mathbb{R}^n \to \mathbb{R}^d$ notwendig, welche einen n-dimensionalen Vektor $x \in \mathbb{R}^n$ in einen d-dimensionalen Vektor $\phi(x) \in \mathbb{R}^d$ mit $\phi(x) = (\phi_1(x), ..., \phi_d(x))$ transformiert. Mit einer geeigneten Funktion $\phi(x)$ können die Daten in dem höherdimensionalen Raum mit größerer Wahrscheinlichkeit linear getrennt werden.

Die Klassifikationsregel von Gleichung 1 lässt sich entsprechend umschreiben zu:

$$\begin{aligned} f(x^*) &= \text{signum}(w^T \phi(x^*) - b) \\ &= \begin{cases} +1, & \text{wenn } w^T \phi(x^*) > b \\ -1, & \text{wenn } w^T \phi(x^*) < b \end{cases} \end{aligned} \tag{4}$$

Das Produkt $w^T \phi(x)$ aus Gleichung 4 kann mit der Gleichung 5 angegeben werden:

$$\begin{aligned} w^T \phi(x^*) &= \phi(x^*)^T w = \phi(x^*)^T \sum_{i=1}^{L} \alpha_i y_i \phi(x_i) \\ &= \sum_{i=1}^{L} \alpha_i y_i \phi(x^*)^T \phi(x_i) \end{aligned} \tag{5}$$

3.2 SVM-Kernel

Das Skalarprodukt $\phi(x^*)^T \phi(x_i)$ in Gleichung 5 kann von einer Kernelfunktion ersetzt werden:

$$K(x, y) = \phi(x)^T \phi(y) \tag{6}$$

Auf diese Weise muss $\phi(x)$ nicht explizit ausgerechnet werden. Dies wird Kerneltrick genannt. Die Klassifikationsregel ergibt sich damit zu:

$$\begin{aligned} f(x^*) &= \text{signum} \left(\sum_{i=1}^{L} \alpha_i y_i K(x^*, x_i) - b \right) \\ &= \begin{cases} +1, & \text{wenn } \sum_{i=1}^{L} \alpha_i y_i K(x^*, x_i) > b \\ -1, & \text{wenn } \sum_{i=1}^{L} \alpha_i y_i K(x^*, x_i) < b \end{cases} \end{aligned} \tag{7}$$

Es gibt sehr viele verschiedene Kernel. Einige bekannte Exemplare, die wir in Kapitel 6 untersuchen werden, sind:

- RBF-Kernel:

$$K(x, y) = \exp(-\frac{1}{2\sigma} ||x - y||^2) \tag{8}$$

- χ^2-Kernel:

$$K(x, y) = 1 - \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\frac{1}{2}(x_i + y_i)} \tag{9}$$

- Histogrammschnitt (HS)-Kernel:

$$K(x, y) = \sum_{i=1}^{n} \min(x_i, y_i) \tag{10}$$

3.3 Regressions-SVM

Die SVM kann nicht nur zur Lösung von Klassifikationsproblemen, sondern auch zur Regression eingesetzt werden. Mit Hilfe der Regression-SVM erhalten wir also keine Trennebene, sondern eine Hyperebene, die unsere Daten möglichst gut beschreibt. Der Algorithmus bleibt dabei seiner

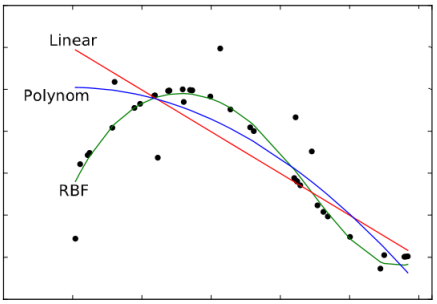

Abbildung 1: Kernel-Vergleich zur SVM-Regression

Ursprungsform ähnlich. In [11] wird die Idee für diese Vorgehensweise beschrieben.

Sei $\{(x_1, y_1), ..., (x_l, y_l)\} \subset X \times \mathbb{R}$ die Trainingsmenge, wobei ein $x_i \in X \subset \mathbb{R}^d$ beispielsweise ein d-dimensionaler numerischer Vektor ist, der die studentischen Attribute beschreibt und y_i die zugehörige Abschlussnote.

Wir suchen eine Funktion f, die möglichst gut unsere Trainingsmenge approximiert, d.h. bis auf kleine Fehler soll $f(x_i) = y_i$ gelten. Um die Regressionsgerade $f(x) = w^T x + b$ mit $w \in \mathbb{R}^d$ und $b \in \mathbb{R}$ für unsere Trainingsmenge zu erhalten, muss das folgende Optimierungsproblem gelöst werden:

$$\begin{aligned} &\min \frac{||w||^2}{2} \\ &\text{u.d.N} \begin{cases} y_i - f(x_i) \leq \epsilon \\ f(x_i) - y_i \leq \epsilon \end{cases} \end{aligned} \tag{11}$$

Für das Optimierungsproblem in (11) wird angenommen, dass eine solche Funktion f existiert, welche alle Punktepaare unserer Trainingsmenge mit einer Genauigkeit ϵ approximiert. In der Regel ist es nicht der Fall, sodass auch hier mit Kernel und sogenannten Schlupfvariablen ξ gearbeitet wird.

In der Abbildung 1 ist die Regression mit der Auswahl einiger Kernel auf einer Datenmenge mit zweidimensionalen Punkten visualisiert. Die Punkte sind zufällig aus einer Sinus-ähnlichen Funktion mit hinzugefügter Streuung entnommen. Wir können bereits hier sehen, wie sich die Wahl eines geeigneten Kernel auf die Güte der Regression auswirkt. Mit dem RBF-Kernel können die Daten hier viel besser approximiert werden, als mit dem Polynom-Kernel.

4. DATENSATZ

In dieser Arbeit wird der Datensatz aus [3] verwendet. Es handelt sich um die Daten einer portugiesischen Schule mit 395 Schülerdaten einer Mathematik-Klasse. Die gesammelten Daten umfassen folgende Attribute:

- student's age (numeric: from 15 to 22), student's school (binary: Gabriel Pereira or Mousinho da Silveira), student's home address type (binary: urban or rural), parent's cohabitation status (binary: living together or apart), mother's education (numeric: from 0 to 4), mother's job (nominal), father's education (numeric: from 0 to 4), father's job (nominal), student's guardian (nominal: mother, father or other), family size (binary: ≤ 3 or > 3), quality of family relationships (numeric: from 1 – very bad to 5 – excellent), reason to choose this school (nominal: close to home, school reputation, course preference or other), home to school travel time (numeric: 1 – $<$ 15 min., 2 – 15 to 30 min., 3 – 30 min. to 1 hour or 4 – $>$ 1 hour), weekly study time (numeric: 1 – $<$ 2 hours, 2 – 2 to 5 hours, 3 – 5 to 10 hours or 4 – $>$ 10 hours), number of past class failures (numeric: n if $1 \leq$ n < 3, else 4), extra educational school support (binary: yes or no), family educational support (binary: yes or no), extra-curricular activities (binary: yes or no), extra paid classes (binary: yes or no), Internet access at home (binary: yes or no), attended nursery school (binary: yes or no), wants to take higher education (binary: yes or no), with a romantic relationship (binary: yes or no), free time after school (numeric: from 1 – very low to 5 – very high), going out with friends (numeric: from 1 – very low to 5 – very high), weekend alcohol consumption (numeric: from 1 – very low to 5 – very high), workday alcohol consumption (numeric: from 1 – very low to 5 – very high), current health status (numeric: from 1 – very bad to 5 – very good), number of school absences (numeric: from 0 to 93)
- G1: first period grade (numeric: from 0 to 20)
- G2: second period grade (numeric: from 0 to 20)
- G3: final grade (numeric: from 0 to 20)

Die Notenskala hat insgesamt 21 Stufen, 0 ist die schlechteste und 20 die beste Note.

5. EVALUATIONS-FRAMEWORK

Als Evaluationsmaße werden wie in [3] die Genauigkeit PCC und die Wurzel der mittleren quadratischen Abweichung RMSE (Root Mean Square Error) verwendet. Sei $\hat{y_i}$ die Vorhersage für das i-te Test-Exemplar und y_i die tatsächliche Note, dann sind die Maße folgenderweise definiert:

$$\begin{aligned} \Phi(i) &= \begin{cases} 1, & \text{wenn } y_i = \hat{y_i} \\ 0, & \text{wenn} \end{cases} \\ PCC &= \sum_{i=1}^{N} \frac{\Phi(i)}{N} \times 100(\%) \\ RMSE &= \sqrt{\sum_{i=1}^{N} \frac{(y_i - \hat{y_i})^2}{N}} \end{aligned} \tag{12}$$

Um aussagekräftigere Ergebnisse zu erhalten und um die Ergebnisse mit [3] vergleichen zu können, wird eine 10-fache Kreuzvalidierung angewandt. Das bedeutet, der Datensatz wird zufällig auf 10 möglichst gleich große Teile aufgeteilt. Jede der 10 Teilmengen wird einmal als Testmenge und die restlichen 9 Teilmengen als Trainingsmenge verwendet. Dieser Prozess wird insgesamt 20 Mal wiederholt, sodass das Endergebnis für jeden einzelnen Versuch ein Mittelwert aus insgesamt 200 Durchläufen ist.

Für die Implementierung der Regressions-SVM wird das Accord.NET Framework [12] verwendet.

Die Studentenvektoren werden normiert, wie in [18] empfohlen. Für den Vektor $x_i = (x_{i1}, .., x_{in}, y_i)$ mit $y_i = \text{G3}$ wird folgende Normierung durchgeführt: $\hat{x_{i1}} = \frac{x_{i1}}{\sum_{j=1}^{n} x_{ij}}$. Für den normierten Vektor $\hat{x_i}$ gilt dann $\sum_j^n \hat{x_{ij}} = 1$.

6. EVALUATION

Um die Ergebnisse möglichst gut mit [3] zu vergleichen, werden gleiche Evaluationsbedingungen verwendet. Es werden drei verschiedene Probleme betrachtet:

1. Binäre Klassifikation - bestanden, falls G3 ≥ 10, sonst durchgefallen
2. 5-Level Klassifikation - (basierend auf dem Erasmus Noten-Umwandlungs-System wird die Notenskala auf 5 Level aufgeteilt)
3. Genaue Vorhersage - numerische Ausgabe des G3-Wertes

Zusätzlich werden wir die folgenden 3 Fälle unterscheiden:

1. A: Alle Features werden verwendet
2. B: Alle Features außer G2 werden verwendet
3. C: Alle Feautures außer G1 und G2 werden verwendet

Als Erstes untersuchen wir den Einfluss der σ-Wahl für den RBF-Kernel und der Regressions-SVM. Wir verwenden die σ-Heuristik aus [4] $\sigma = \text{median}(\{\text{dist}(u, v)|u \neq v\})$ und vergleichen diese mit einer naiven Wahl $\sigma = 1$ und den SVM-RBF-Ergebnissen aus [3]. Diese Evaluation führen wir zunächst für den Fall A (alle Features) aus.

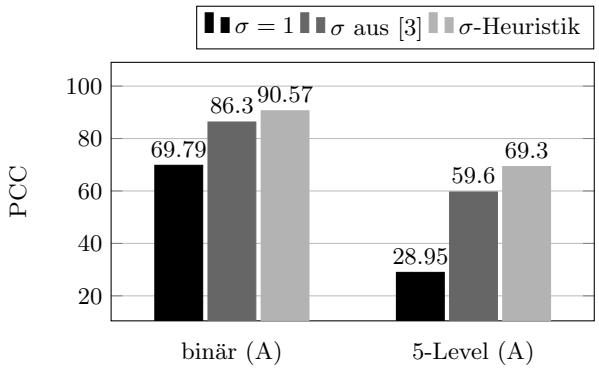

Abbildung 2: Vergleich der σ-Auswahl für das binäre und das 5-Level Problem (A: Alle Features verwendet)

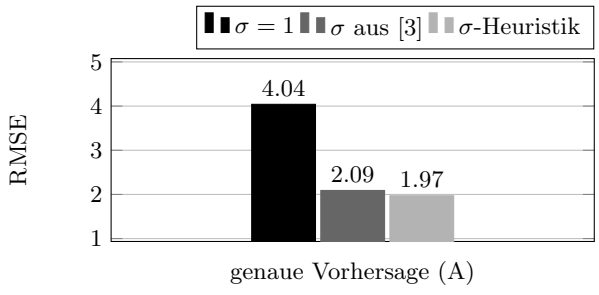

Abbildung 3: Vergleich der σ-Auswahl für die genaue Vorhersage (A: Alle Features verwendet)

Aus den Abbildungen 2 und 3 können wir für den Fall A die Wichtigkeit einer geeigneten Wahl für den σ-Parameter des RBF-Kernels erkennen. Insbesondere bringt die verwendete σ-Heuristik für das binäre Klassifikationsproblem mit einer Genauigkeit von 90.57% eine Verbesserung gegenüber der σ-Wahl aus [3] mit 86.3%. Auch für das 5-Level Problem ist eine Verbesserung um etwa 10% gegeben. Beim Problem der genauen Vorhersage kann der RMSE Wert gegenüber [3] leicht verbessert werden. Die naive Wahl $\sigma = 1$ wie sie als Standard-Parameter im Accord.NET Framework für den RBF-Kernel gegeben ist, liefert beim binären Problem eine deutlich schlechtere Genauigkeit von 69.79% und ist damit um mehr als 20% schlechter. Beim 5-Level Problem liefert die naive Wahl gegenüber der σ-Heuristik sogar eine um etwa 40% schlechtere Genauigkeit. Beim Problem der genauen Vorhersage liefert die naive σ-Wahl einen mehr als doppelt so großen RMSE-Wert und ist damit deutlich schlechter als die σ-Heuristik.

Da die naive Wahl $\sigma = 1$ offensichtlich keine guten Ergebnisse liefert, vergleichen wir in den nächsten Versuchen nur die σ-Heuristik und die σ-Wahl von [3] auf den Fällen B (alle Features außer G2) und C (alle Features außer G1 und G2) für das binäre und das 5-Level Problem miteinander. Die Ergebnisse sind in Abbildung 4 visualisiert. In Fall B (alle Features außer G3) liefert die σ-Heuristik für das binäre Klassifkations-Problem mit 84.64% eine um etwa 4% bessere Genauigkeit. Für das 5-Level Problem liefert die σ-Heuristik mit 58.23% eine um mehr als 10% bessere Genauigkeit. Im Fall C (alle Features außer G1 und G2) werden die Werte bereits schlechter. Die σ-Heuristik eignet sich somit bei dem binären und dem 5-Level Problem ausschließlich für die Fälle A und B.

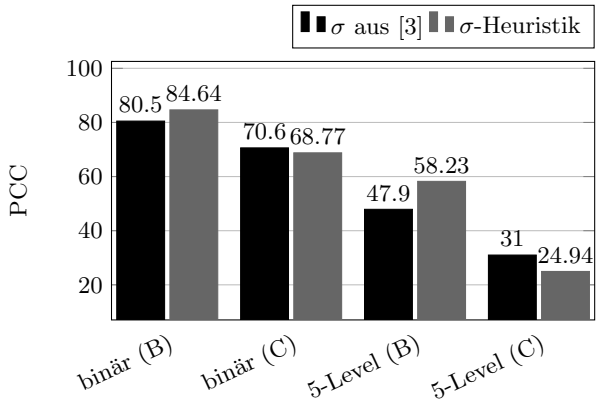

Abbildung 4: Vergleich der σ-Auswahl für das binäre und das 5-Level Problem (B: alle Features außer G2, C: alle Features außer G1 und G2)

Die Fälle B und C vergleichen wir auch für die genaue Vorhersage der Note G3. Die Ergebnisse sind in Abbildung 5 visualisiert. Im Fall B ist die σ-Heuristik noch etwas besser und im Fall C sind die Ergebnisse identisch.

Zusätzlich wollen wir den RBF-Kernel mit dem χ^2-Kernel und dem Histogrammschnitt-Kernel vergleichen. Die Ergebnisse dieser beiden Kernel sind in Tabelle 6 abgebildet. Wir sehen, dass sich die Werte nicht signifikant von den Ergebnissen des RBF-Kernels mit der σ-Heuristik unterscheiden. Diese beiden Kernel sind also ebenso einsetzbar, wenn ein Kernel verwendet werden soll, der keine zusätzlichen Parameterangaben braucht.

binäre Vorhersage						5-Level Vorhersage						genaue Vorhersage					
Fall A		Fall B		Fall C		Fall A		Fall B		Fall C		Fall A		Fall B		Fall C	
χ^2	HS	χ^2	HS	χ^2	HS	χ^2	HS	χ^2	HS	χ^2	HS	χ^2	HS	χ^2	HS	χ^2	HS
90%	90%	85%	85%	67%	70%	68%	69%	56%	57%	26%	28%	2.34	2.14	3.06	2.9	4.4	4.34

Tabelle 2: Ergebnisse des χ^2-Kernel und des Histogrammschnitt-Kernel (HS)

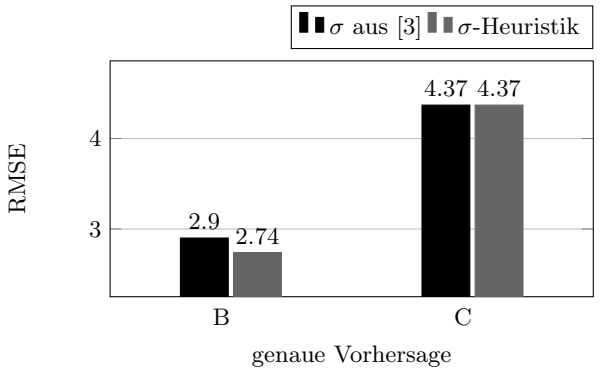

Abbildung 5: Vergleich der σ-Auswahl für die genaue Vorhersage von G3 (B: alle Features außer G2, C: alle Features außer G1 und G2)

7. FAZIT

Wir haben in dieser Arbeit die Anwendung der Regressions-SVM zur Vorhersage studentischer Leistungen mit unterschiedlichen Kernel untersucht. Eine Regressions-SVM ermöglicht das Training auch, wenn aufgrund weniger Trainingsdaten z. B nicht alle Noten abgedeckt sind. Die Evaluation wurde auf den Schülerdaten eines Mathematikkurses durchgeführt. Mit dem Einsatz einer σ-Heuristik für den RBF-Kernel konnten die Ergebnisse einer früheren Arbeit auf diesen Daten verbessert werden. Für den Fall, dass zusätzlich zu den privaten Daten auch vorherige Noten bekannt waren, konnte die Vorhersage von „bestanden oder nicht bestanden“ mit einer Genauigkeit von 90.57% erreicht werden, was eine praktische Anwendbarkeit ermöglicht. Bei einer Unterteilung der 21 möglichen Noten in 5 Notenstufen konnte die richtige Notestufe mit einer Genauigkeit von 69.3% bestimmt werden. Waren weniger vorherige Noten bekannt, so war die erreichte Genauigkeit kleiner. Wir haben zusätzlich den χ^2- und HS-Kernel untersucht. Mit ähnlich guten Ergebnissen eignen sich diese ebenfalls.

8. REFERENCES

[1] A. Acharya and D. Sinha. Early prediction of students performance using machine learning techniques. *International Journal of Computer Applications*, 107(1), 2014.

[2] C. Cortes and V. Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.

[3] P. Cortez and A. M. G. Silva. Using data mining to predict secondary school student performance. 2008.

[4] T. S. Jaakkola, M. Diekhans, and D. Haussler. Using the fisher kernel method to detect remote protein homologies. In *ISMB*, volume 99, pages 149–158, 1999.

[5] Y.-J. Lee and O. L. Mangasarian. Ssvm: A smooth support vector machine for classification. *Computational optimization and Applications*, 20(1):5–22, 2001.

[6] M. I. Lopez, J. Luna, C. Romero, and S. Ventura. Classification via clustering for predicting final marks based on student participation in forums. *International Educational Data Mining Society*, 2012.

[7] J. C. Platt. fast training of support vector machines using sequential minimal optimization. *Advances in kernel methods*, pages 185–208, 1999.

[8] V. Ramesh, P. Parkavi, and K. Ramar. Predicting student performance: a statistical and data mining approach. *International journal of computer applications*, 63(8), 2013.

[9] C. Romero, M.-I. López, J.-M. Luna, and S. Ventura. Predicting students' final performance from participation in on-line discussion forums. *Computers & Education*, 68:458–472, 2013.

[10] S. Sembiring, M. Zarlis, D. Hartama, and E. Wani. Prediction of student academic performance by an application of data mining techniques. *International Proceedings of Economics Development & Research*, 6:110–114, 2011.

[11] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222.

[12] C. R. Souza. *The Accord.NET Framework*. http://accord-framework.net/, 2014.

[13] P. Strecht, L. Cruz, C. Soares, J. Mendes-Moreira, and R. Abreu. A comparative study of classification and regression algorithms for modelling students' academic performance. In *8th Conference on Educational Data Mining (EDM2015)*, 2015.

[14] A. Tekin. Early prediction of students' grade point averages at graduation: a data mining approach. *Eurasian Journal of Educational Research*, (54):207–226, 2014.

[15] N. Thai-Nghe, A. Busche, and L. Schmidt-Thieme. Improving academic performance prediction by dealing with class imbalance. In *Intelligent Systems Design and Applications, 2009. ISDA'09. Ninth International Conference on*, pages 878–883. IEEE, 2009.

[16] K. Watkins. An improved recommendation models on grade point average prediction and postgraduate identification using data mining. In *Advances in Neural Networks, Fuzzy Systems and Artificial Intelligence*, pages 186–194. WSEAS Press, May 2014.

[17] A. Webb and K. Copsey. *Statistical Pattern Recognition*. Wiley, 2011.

[18] C. wei Hsu, C. chung Chang, and C. jen Lin. A practical guide to support vector classification. `https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf`, 2010.

[19] W. Zhang, S. Zhang, and S. Zhang. Predicting the graduation thesis grade using svm. *International Journal of Intelligent Information Processing*, 5(3):60, 2015.