Universität Siegen

Programmierpraktikum

Datenanalyse
und
Datenvorhersage

Gutachter

Dr. Viktor Bindewald

Name:	Corinne Pretz
Matrikelnummer:	1454415
Email:	corinne.pretz@student.uni-siegen.de
Studiengang:	Business Analytics
Abgabetermin:	31.08.2022

Inhaltsverzeichnis

1 Einleitung

Durch die immer weiter fortschreitende Digitalisierung gibt es auch immer mehr und mehr Daten, die erfasst werden. Ob es Daten von Bewertungen zu Produkten oder Filmen sind oder Einkaufsdaten der Kunden, in fast jedem Bereich werden heute digital Daten gespeichert. Diese erfassten Daten müssen jedoch weiterverarbeitet werden, sodass sich daraus Informationen gewinnen lassen oder eventuell sogar mittels Machine Learning Vorhersagen (Kaufverhalten, Umsätze, Unfallzahlen) getroffen werden können.

In dieser Arbeit soll zum einen das Gewinnen von Informationen aus einem oder mehreren Datensätzen erläutert werden. Zum anderen sollen mittels vorhandener Daten Zusammenhänge erkannt und darauf basierend Schlüsse auf die Zukunft gezogen werden.

Ersteres wird anhand zweier Filmdatensätze vertieft. Diese enthalten jegliche Filmdaten aus unterschiedlichen Genres zwischen 1902 und 2018 mit Bewertungen von Zuschauern. Dabei sollen unterschiedliche Informationen aus den Daten gewonnen werden, die für einen Nutzer interessant sein könnten. Mit einem weiteren Datensatz, welcher die Lernerfolge der Teilnehmer eines Portugiesischkurses sowie bestimmte Attribute (Jobs der Eltern, Stunden die gelernt wurde etc.) enthält, soll mit Hilfe von linearer Regression und der Support Vector Regression eine Vorhersage zu den Lernerfolgen von Schülern gemacht werden, die die Prüfung noch nicht abgelegt haben.

1.1 Aufbau der Arbeit

Diese Arbeit beginnt damit, in Kapitel 2 das Gewinnen von Informationen aus einem bzw. mehreren Datensätzen am Beispiel zweier Filmdatensätze zu beschreiben. Dabei werden die hier zur Verfügung gestellten Datensätze aufbereitet und visualisiert. In Kapitel 3 wird die Vorhersage mittels lineare Regression und die Support Vector Regression erläutert. Im letzten Kapitel werden diese Methoden zur Vorhersage von Lernerfolgen aufgegriffen.

2 Datenaufbereitung anhand zweier Filmdatensätze

In diesem Kapitel werden die in dieser Arbeit genutzten Datensätze kurz beschrieben. Hier werden zwei Datensätze zu Filmbewertungen genutzt. Anschließend wird darauf eingegangen, wie man aus diesen Daten Informationen ziehen kann, die interessant für einen Nutzer sein könnten. Um dies zu ermöglichen, müssen die Datensätze jedoch zunächst angepasst werden.

2.1 Filmdatensätze

Zur Verfügung stehen zwei Datensätze, die jegliche Informationen zu Filmbewertungen enthalten. Die Tabelle 1 zeigt eine Übersicht der Daten, die in den zwei Datensätzen „Movies“ und „Ratings“ enthalten sind. Hierbei ist zu sehen, dass nur ein Attribut (MovieId) in beiden Datensätzen zu finden ist und alle anderen nur in dem einen oder dem anderen zu finden sind. Zusätzlich ist zu erwähnen, dass ein Film mehreren Genres zugeordnet werden kann.

Datensatz	MovieId	UserId	Ratings (0.5-5)	Timestamp	Titel + Jahr	Genres
Ratings	✓	✓	✓	✓	×	×
Movies	✓	×	×	×	✓	✓

Tabelle 1: Übersicht der Daten in den Dataframes.

2.2 Datensatz Anpassungen

Um nützliche Informationen aus den Datensätzen gewinnen zu können, müssen diese meist verändert werden. So können in diesem Fall zum Beispiel ohne Veränderung der Datensätze die Durchschnittsbewertung einzelner Filme ermittelt werden, aber nicht die Durchschnittsbewertungen einzelner Genres. Auch eine zeitliche Eingrenzung von Bewertungen ist etwas aufwendiger, da nicht ein Datum, sondern ein Zeitstempel angegeben ist. Zeitliche Angaben zu den Erscheinungsjahren der Filme zu extrahieren gestaltet sich ebenfalls schwierig, da die Erscheinungsjahre mit im Titel gespeichert sind und nicht separat.

Die folgenden Änderungen wurden auf den Datensatz angewandt:

- Datensätze auf Dopplungen und Lücken prüfen
- Zusammenfügen der Dataframes anhand der MovieId

- Die Genres eines Films aufteilen (Jeder Film wird für jedes seiner Genres einmal gelistet)
- Erscheinungsjahre der Film aus den Titeln extrahieren
- Zeitstempel der Bewertungen in Daten (Jahr, Monat, Tag, Stunde) transformieren

2.3 Informationen aus den Daten

Im Folgenden wird kurz angeschnitten, welche Informationen aus den Daten gewonnen wurden. Für mehr Informationen steht das, im angegebenen GitHub Repository hinterlegte Juypter-Notebook „movie_ratings“ zur Verfügung.

Aus dem angepassten Datensatz lässt sich nun z.B. entnehmen, wie viele Filme es in einem Genre gibt und wie dessen Durchschnittsbewertung ausfällt.

So lässt sich auch die Information der Top bzw. Flop Filme ausgeben unter der Bedingung, dass die aufgegriffenen Filme eine mindest Anzahl an Bewertungen haben müssen (siehe Abbildung 1).

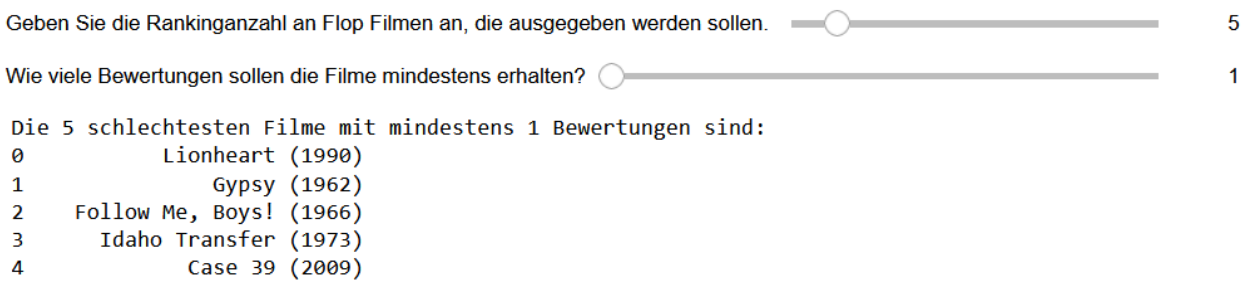


Abbildung 1: Ausgabe der fünf schlechtesten Filme mit mindestens einer Bewertung.

Da beim Vorbereiten des Datensatzes alle Genres aufgeteilt wurden, können jetzt auch die Bewertungen von Genres verglichen werden und nicht nur die der Filme. Dazu kann man in einem Histogramm die relative Häufigkeit der Bewertungen einer Bewertungsklasse miteinander vergleichen. Abbildung 2 zeigt den Vergleich der Genres „Animation“ und „Action“.

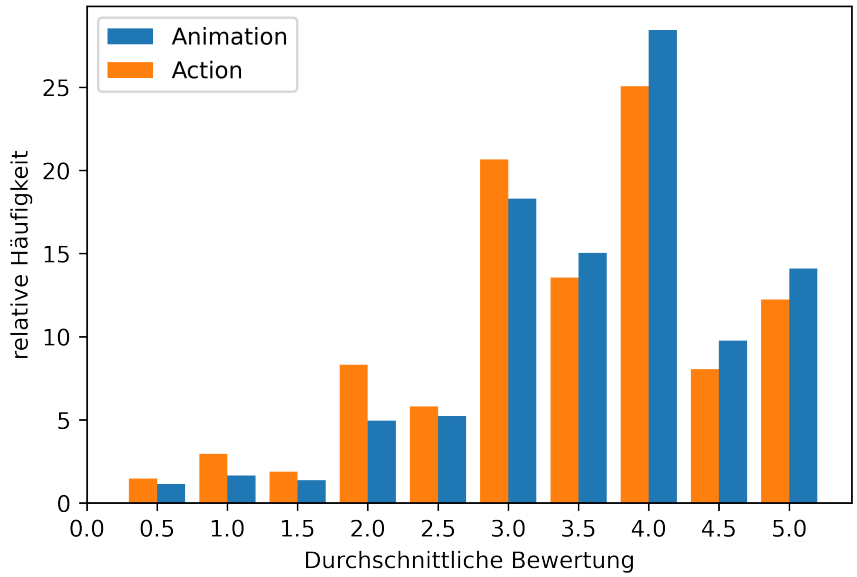


Abbildung 2: Vergleich zweier Filmgenres anhand der relativen Häufigkeit der Bewertungen.

Auch interessant für einen Nutzer kann es sein, sich die Anzahl an Bewertungen für einen Film und dessen Durchschnittswertung anhand einer MovieId ausgeben zu lassen.

Abbildung 3 zeigt den Verlauf der Menge der Filmproduktionen über die Jahre. Diese Darstellung ist nur durch das Extrahieren der Jahreszahlen aus den Filmtiteln möglich.

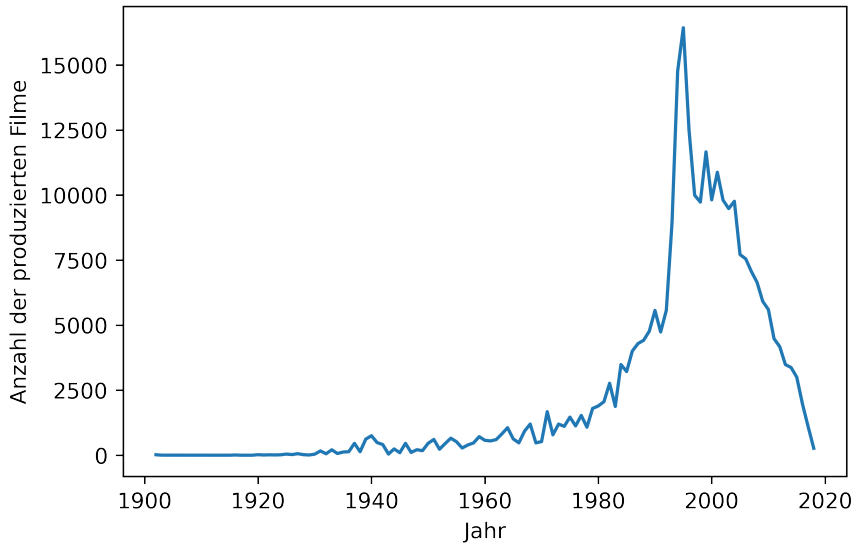


Abbildung 3: Verlauf der Anzahl an Filmproduktionen über die Jahre 1902 bis 2018

3 Datenvorhersage

In diesem Kapitel sollen die benötigten Grundlagen zur linearen Regression und zur Suport Vector Regression eingeführt und erläutert werden. Dazu werden beide Regressionsmodelle und ihre Funktionsweise kurz erläutert. Anschließend wird kurz auf den Begriff des Machine Learning eingegangen, da in dieser Arbeit sowohl die lineare Regression als auch die Suport Vector Regression zur Vorhersage von Daten genutzt werden.

3.1 Regressionsanalyse

Die Regressionsanalyse ist ein Verfahren zur Modellierung von Zusammenhängen zwischen Daten. Dabei wird ein Zusammenhang zwischen einer abhängigen und einer oder mehreren unabhängigen Variablen gesucht. Anhand solcher Zusammenhänge können dann Schlüsse für die Zukunft gezogen werden. Bekannte Verfahren dabei sind z.B. die lineare Regression, die multiple lineare Regression oder die polynomielle Regression.

Die Regressionsanalyse kann somit zum einen genutzt werden, um einzig und allein den Zusammenhang von Daten zu erkennen (verrauschte Daten glätten bzw. anpassen) und zum anderen zur Vorhersage von Daten (z.B. Verkaufszahlen vorhersagen) genutzt werden.

3.1.1 Lineare Regression

Bei der linearen Regression wird versucht, einen linearen Zusammenhang zwischen einer abhängigen und einer oder mehreren unabhängigen Variablen zu modellieren. Abbildung 4 stellt die lineare Regression visuell dar. Vorhanden ist eine Datenmenge, dessen Zusammenhang zur unabhängigen Variable unbekannt ist. Bei der linearen Regression wird versucht eine Gerade (grün) durch diese Daten zu legen, sodass z.B. die Summe der quadratischen Abweichung von der Geraden zu den Punkten minimal ist [3]. Somit lässt sich als Zielfunktion formulieren:

$$\min \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \,. \tag{1}$$

y ist dabei der tatsächliche Wert und $\hat{y}$ die Vorhersage zur gleichen unabhängigen Variable x.

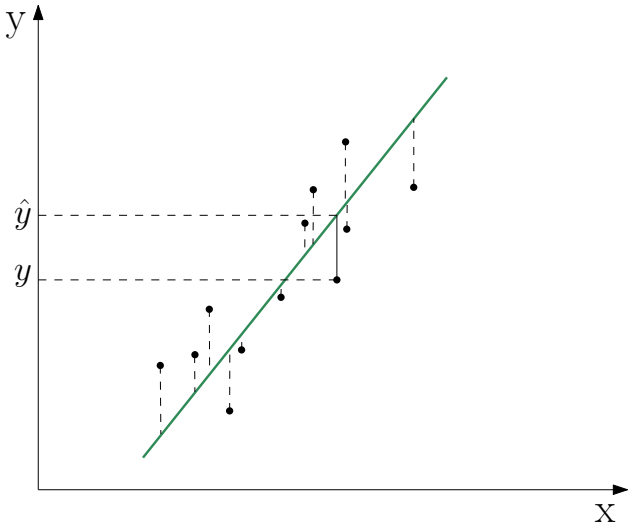


Abbildung 4: Beispielhafte lineare Regression. Die grüne Gerade ist die Regressionsgerade, die durch die lineare Regression ermittelt wurde. y ist der tatsächliche Wert und $\hat{y}$ ist die Vorhersage für eine identische Eingabe von x.

Im Folgenden werden zwei Spezialfälle der linearen Regression betrachtet. Zum einen die der linearen Einfachregression und zum anderen die der multiplen linearen Regression.

Lineare Einfachregression

Bei der linearen Einfachregression wird davon ausgegangen, dass es eine abhängige Variable und auch nur eine unabhängige Variable gibt. Somit ergibt sich für die Regressionsgerade der folgende Zusammenhang:

$$f(x) = w_0 + w_1 x\,. \tag{2}$$

x ist dabei die unabhängige Variable und $f(x)$ die abhängige Variable. w_0 und w_1 sind die durch die lineare Regression bestimmten Parameter, also der y-Achsenabschnitt (w_0) und die Steigung (w_1) der Regressionsgeraden.

Multiple lineare Regression

Bei der multiplen linearen Regression wird versucht, eine abhängige Variable durch mehrere unabhängige Variablen zu erklären. So ist auch hier die abhängige Variable eine Funktion der unabhängigen Variablen. Die multiple lineare Regression ist somit eine Verallgemeinerung

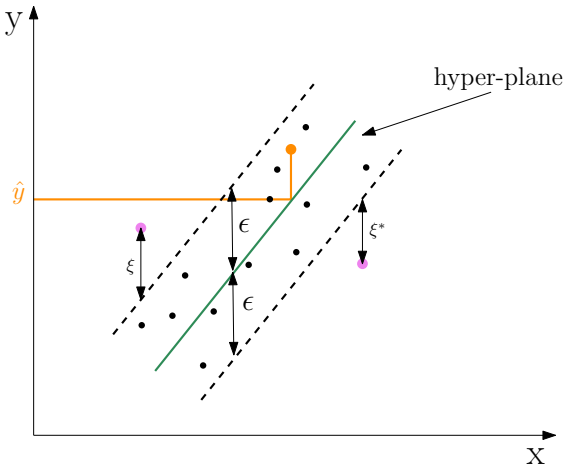
der einfachen linearen Regression auf höhere Dimensionen. Für die Regressionsgerade mittels multipler linearer Regression ergibt sich der folgende lineare Zusammenhang:

$$f(x) = w_0 + \sum_{i=1}^{m} w_i x_i \tag{3}$$

3.1.2 Support Vector Regression

Die Support Vector Regression (SVR) ähnelt der deutlich bekannteren Klassifizierungsmethode Support Vector Machine (SVM). Support Vector Machines werden häufig für Klassifizierungsprobleme beim maschinellen Lernen verwendet [6]. Im Gegensatz zu SVM sollen jedoch keine Daten klassifiziert werden, sondern Zusammenhänge zwischen Daten erkannt werden, um so Schlüsse auf unbekannte Daten ziehen zu können. In diesem Fall kann SVR hilfreich sein [1, 2, 4, 5].

In Abbildung 5 ist die SVR links einmal bildlich im zweidimensionalen Raum dargestellt und rechts das mathematische Modell.



$$\min \quad \frac{1}{2}||w||^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \tag{4}$$

$$\text{s.t.} \quad y_i - wx_i - b \leq \epsilon + \xi_i \tag{5}$$

$$wx_i + b - y_i \leq \epsilon + \xi_i^* \tag{6}$$

$$\xi_i, \xi_i^* \geq 0 \tag{7}$$

Abbildung 5: Die Abbildung links stellt die Support Vector Regression visuell dar. Die grüne Gerade ist dabei die Hyperebene, die gestrichelten Linien sind die „Decision Boundaries“ und ϵ der Abstand dieser zur Hyperebene. Punkte welche nicht innerhalb der „Decision Boundaries“ liegen, sind pink dargestellt und haben einen Abstand ξ zu den „Decision Boundaries“. Rechts ist das mathematische Modell der SVR aufgezeigt.

Die beiden gestrichelten Linien werden als „Decision Boundaries“ bezeichnet und die grüne Linie als die Hyperebene. Das Ziel bei der SVR ist es, die Hyperebene zu finden, bei der die meisten Punkte innerhalb der „Decision Boundaries“ liegen. Die zwei „Decision Boundaries“ befinden sich in einem beliebigen Abstand, von der Hyperebene. Dieser Abstand ϵ wird vom

Nutzer selbst festgelegt. So lässt es sich realisieren, dass nicht jeder Fehler betrachtet wird bzw. die Größe eines Fehlers in einem bestimmten Bereich egal ist. Dadurch wird ein Fehler innerhalb der „Decision Boundaries“ akzeptiert. Es ist ggf. nicht möglich, dass alle Punkte innerhalb der „Decision Boundaries“ liegen. In Abbildung 5 sind außerhalb der „Decision Boundaries“ liegende Datenpunkte sind pink gekennzeichnet und haben einen Abstand von ξ zu den „Decision Boundaries“. Die Anzahl der Punkte, die außerhalb liegen, soll minimiert werden. Somit wird nicht nur die Position der Hyperebene gesucht, die die maximalen Punkte im zugelassenen Bereich fasst, sondern auch die Hyperebene, die den Abstand und die Menge der Punkte außerhalb der „Decision Boundaries“ minimiert. Die Zielfunktion 4 im Modell stellt dies sicher. Die Abstände der Datenpunkte, die außerhalb der „Decision Boundaries“ liegen, werden mit einem vom Nutzer festgelegten Parameter C multipliziert. Je größer C ist, desto größer ist die Bestrafung des Abstands der Punkte außerhalb von ϵ. Wenn sich C dem Wert 0 nähert, werden diese Punkte vom Modell nicht zusätzlich bestraft. Für die beiden „Decision Boundaries“ ergeben sich dann die Funktionen 5 und 6. Diese gehen als Nebenbedingung in das Modell ein [1, 2, 4, 5, 6].

Eine Haupteigenschaft der SVR ist, dass sie sich einer sogenannten Kernelfunktion bedient, die dabei hilft eine Hyperebene im mehrdimensionalen Raum zu finden, wenn kein linearer Zusammenhang im Ausgangsraum zu erkennen ist. Abbildung 6 zeigt eine beispielhafte Anwendung des „Kernel-Tricks“ [5].

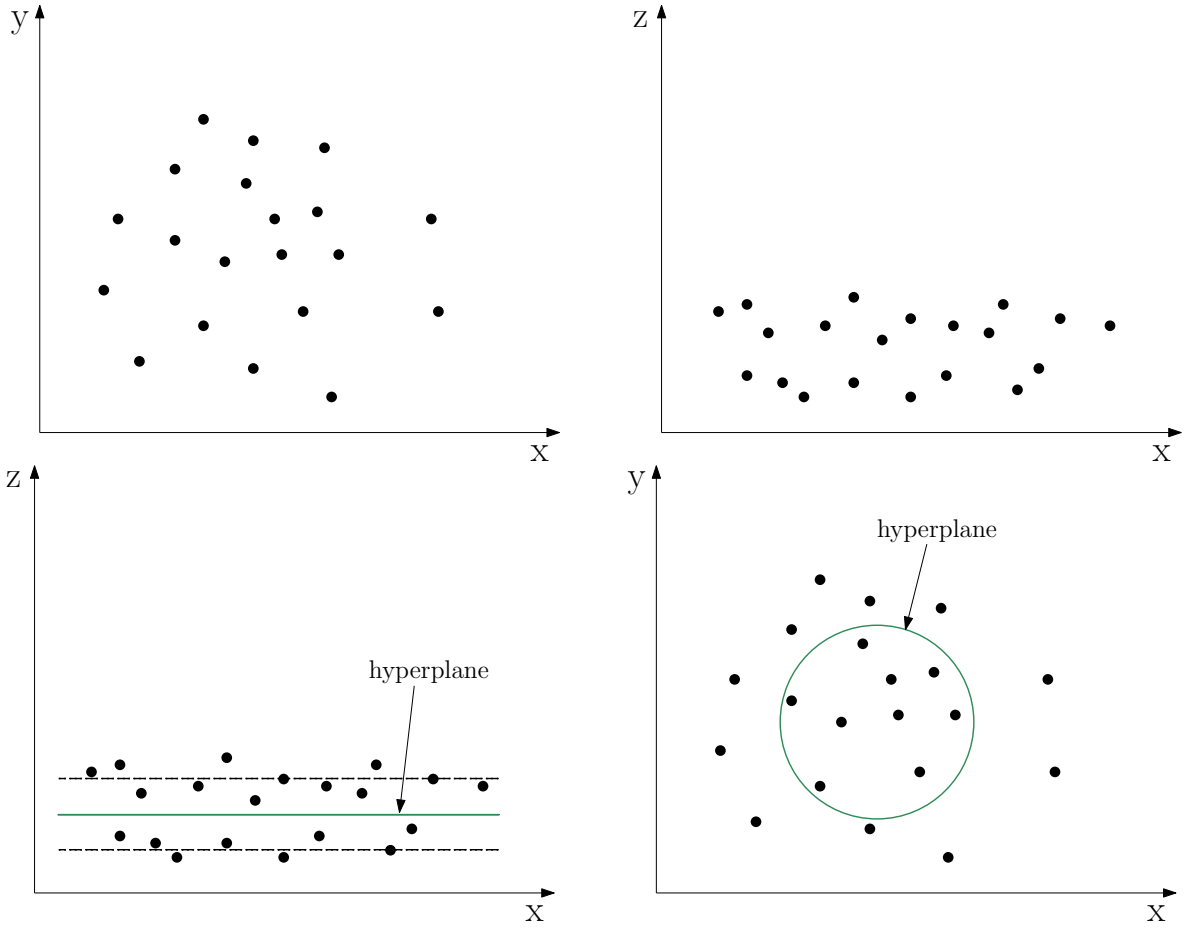


Abbildung 6: Beispielhafte Kernel Nutzung. Links oben zeigt die Datenmenge im Zweidimensionalen. Rechts oben zeigt die Datenmenge im Dreidimensionalen. Die Abbildung links unten zeigt die durch SVR gefundene Hyperebene. Rechts unten ist die Datenmenge wieder im Zweidimensionalen dargestellt, mit der vorher im Mehrdimensionalen gefundenen Hyperebene.

Die Abbildung 3.1.2 zeigt Ergebnisse einer SVR im zweidimensionalen Raum, bei unterschiedlichen Kernelfunktionen (linear, polynomiell und radial-basis-function).

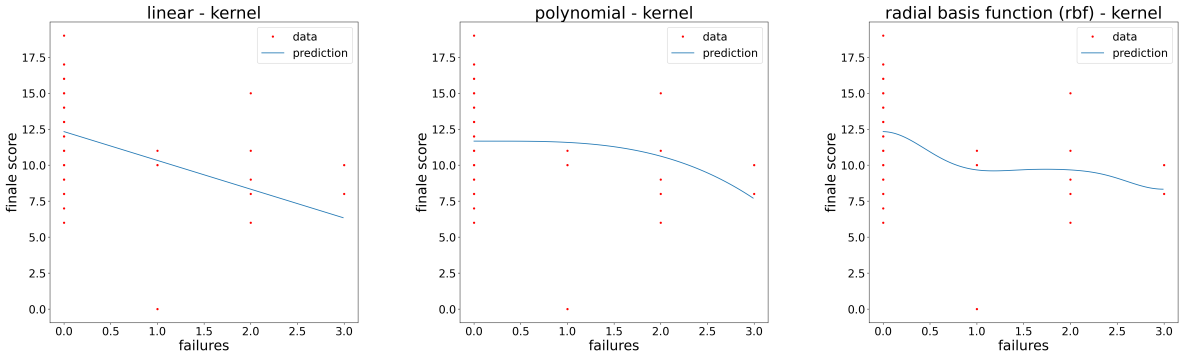


Abbildung 7: Support Vector Regression mit unterschiedlichen Kernelfunktionen. Links linear, mittig polynomiell und recht radial basis function.

3.1.3 Vergleich lineare Regression und SVR

Der wesentliche Unterschied von der linearen Regression zur Support Vector Regression ist der, dass bei der linearen Regression in der Zielfunktion alle Punkte betrachtet werden und so auch in die Fehlerrechnung mit eingehen. Bei der Support Vector Regression jedoch werden nur die Punkte außerhalb der „Decision Boundaries“ betrachtet.

Vorteil der Support Vector Regression ist der, dass Abweichungen einer gewissen Größenordnung zugelassen werden. Ein kurzes Beispiel bei dem dies hilfreich sein kann ist in folgender Referenz zu finden [6].

4 Ergebnisse der Regressionsmethoden

In diesem Kapitel wird sowohl die multiple lineare Regression als auch die Support Vector Regression zur Vorhersage von Lernerfolgen genutzt. Wie auch zu Kapitel 2 gibt es hier Jupyter-Notebooks zur Datenaufbereitung, zur lineare Regression und zur Support Vector Regression, welche im zugehörigen Github Repository hinterlegt sind.

4.1 Datensatz zum Lernerfolg von Schülern

Es wird ein Datensatz genutzt, der für jeden Schüler (649 Instanzen) eines Portugiesischkurses jegliche Informationen (30 Attribute) zur Person enthält und die zum Abschluss des Kurses erreichte Punktzahl. Einen Ausschnitt der Attribute findet sich in Tabelle 2. Einige der Daten sind nominal angegeben, so wie das Geschlecht oder auch die Schule. Andere wiederum in numerischen Werten, wie z.B. der Gesundheitszustand des Schülers.

Attribut	Skala	Ausprägungen
Geschlecht	binär	männlich, weiblich
Alter	numerisch	$[15; 22]$
Vormund	nominal	Mutter, Vater, Anderer
Fahrtzeit	numerisch	$< 15\text{min}$, $15 - 30\text{min}$, $30 - 60\text{min}$, $> 60\text{min}$
Internetzugang	binär	Ja, Nein
Gesundheitszustand	numerisch	$[1; 5]$

Tabelle 2: Auszug der Attribute, deren Skala und deren Ausprägung in dem hier genutzten Dataframe.

4.1.1 Änderung des Datensatzes zum Lernerfolg von Schülern

Vorerst muss geprüft werden, ob es fehlende Daten bzw. Lücken im Datensatz gibt. Der Datensatz ist jedoch vollständig und weist keine Lücken auf.

Da es sich mit nominalen Werten schlecht arbeiten lässt, müssen diese in numerische Werte umgewandelt werden. Binäre nominale Daten wie z.B. „Ja“ und „Nein“ Angaben werden in 1 bzw. 0 geändert. Sobald mehr als nur zwei Ausprägungen vorhanden sind, wird durchnummeriert.

Zusätzlich wird die Korrelation der Attribute zur Zielvariable betrachtet, um den Datensatz um die Attribute zu verringern, deren Korrelation zur Zielvariable nicht besonders hoch ist. Abbildung 8 zeigt die Korrelation der Attribute zur Zielvariable. Da so gut wie alle Attribute keine besonders hohe Korrelation zur Zielvariable aufzeigen, wird die Grenze der ausgewählten Attribute bei einer Korrelation von 0.2/-0.2 gesetzt, diese wird als rote Linie in Abbildung 8 dargestellt. Somit sind im neuen Datensatz nur Attribute, deren Korrelation zur Zielvariable mindestens 0.2/-0.2 beträgt.

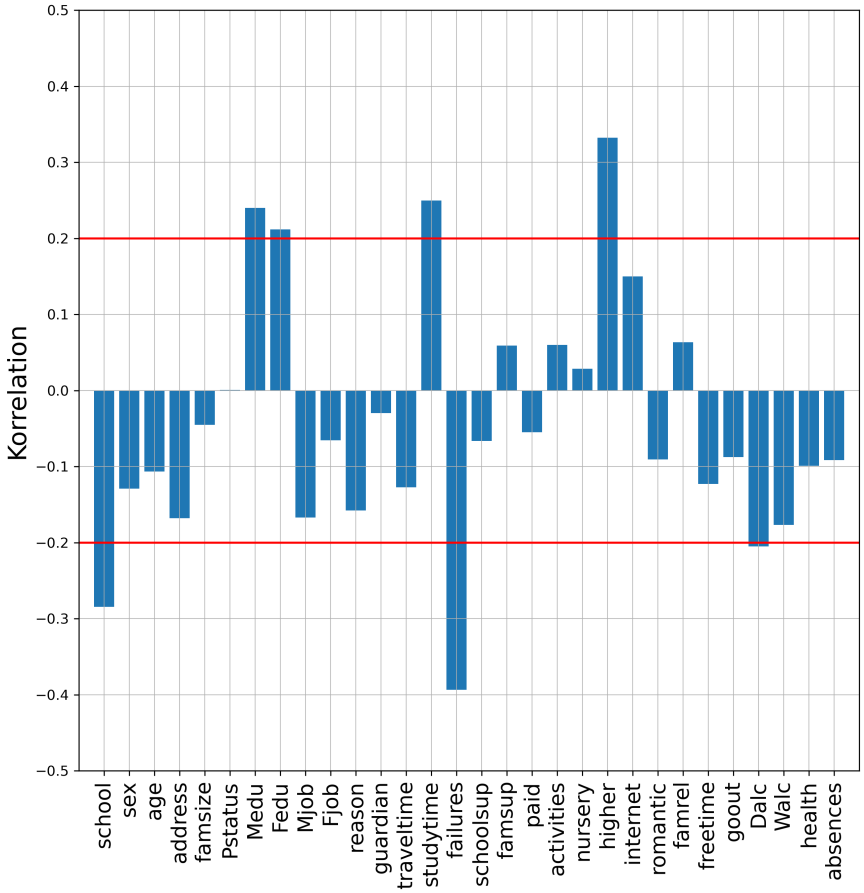


Abbildung 8: Korrelation zwischen den Attributen und der Zielvariable. Die rote Linie zeigt die Grenze auf, ab welchem Korrelationswert die Attribute zur Regression genutzt werden.

Da zum Testen der Modelle ebenfalls Daten benötigt werden, dessen Zielvariable man bereits kennt, müssen die vorhandenen Daten in Trainings und Testdaten aufgeteilt werden. Hierbei wurden 80 % der vorhandenen Daten als Trainingsdaten genutzt und 20 % als Testdaten.

4.2 Vorhersage mittels linearer Regression

In Abbildung 4.2 sind die Ergebnisse der linearen Regression mit allen vorhandenen Attributen (links) und mit den ausgewählte Attributen (rechts) dargestellt. Die roten Punkte in der Abbildung stellen die vorhergesagten Notenpunkte des Modells dar. Die blaue Diagonale in der Abbildung stellt die perfekte Vorhersage dar. Wenn alle Punkte, auf der Diagonalen liegen würden, wäre ein Fehler von 0 erreicht. Als Fehlermaß wird hier der mittlere quadratische Fehler berechnet. In dem Fall, dass alle Attribute berücksichtigt werden, liegt dieser bei ca. 6.49 und bei dem Fall, dass nur Attribute mit einbezogen werden, deren Korrelation mindesten 0.2/-0.2 ist, liegt der Fehler bei ca. 6.8.

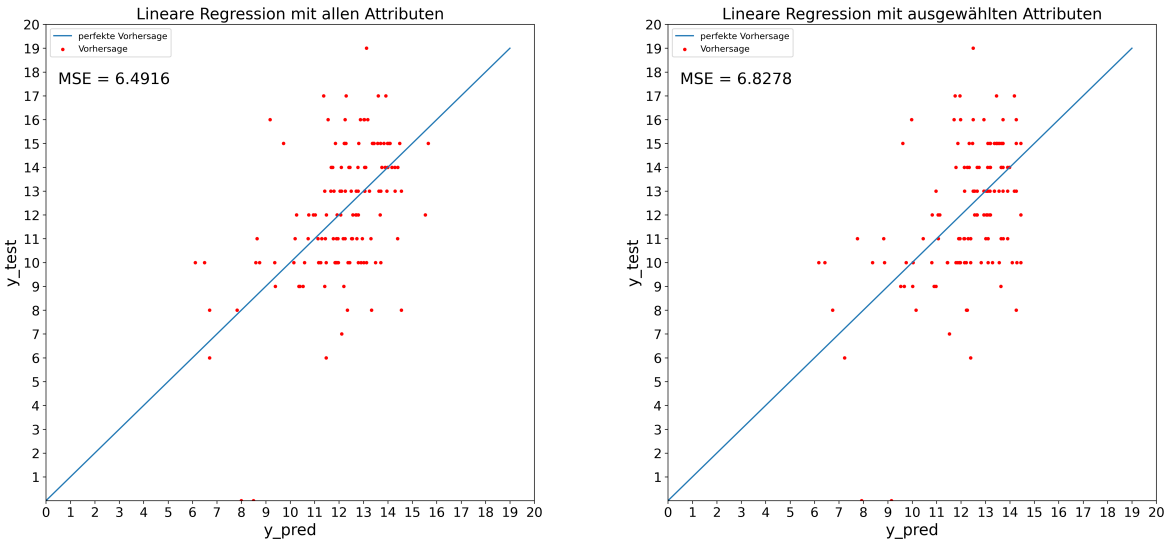


Abbildung 9: Regression mittels linearer Regression. Links mit allen verfügbaren Attributen und rechts mit allen Attributen, die eine Korrelation von min. 0.2 haben.

4.3 Vorhersage mittels Support Vector Regression

In Abbildung 10 sind die Ergebnisse der Support Vector Regression mit allen vorhandenen Attributen (links) und mit den ausgewählte Attributen (rechts) dargestellt. Zusätzlich wurden unterschiedliche Kernelfunktionen getestet. Für die erste Reihe der Abbildung wurde ein linearer Kernel genutzt, für die zweite ein polynomieller und für die dritte die radial-basis-function. Die roten Punkte in der Abbildung sind ebenfalls die Vorhersagen des Modells. Die blaue Diagonale stellt wie bei der linearen Regression die perfekte Vorhersage dar. Als Fehlermaß wird erneut der mittlere quadratische Fehler berechnet. Die Tabelle 3 gibt einen Überblick über die Fehler.

Kernel-Funktion	alle Attribute	ausgewählte Attribute
linear	6.6527	6.8342
polynomiell	7.0405	8.1423
radial-basis-function	6.1768	6.6298

Tabelle 3: Übersicht der mittleren quadratischen Abweichungen der Support Vector Regression mit unterschiedlichen Kernelfunktionen und unterschiedlichen Attributen.

Der Parameter C zur Gewichtung der Punkte außerhalb des erlaubten Bereiches ϵ kann ebenfalls variiert werden. Eine Änderung dieses Wertes verursacht jedoch keine große Änderung am mittleren quadratischen Fehler.

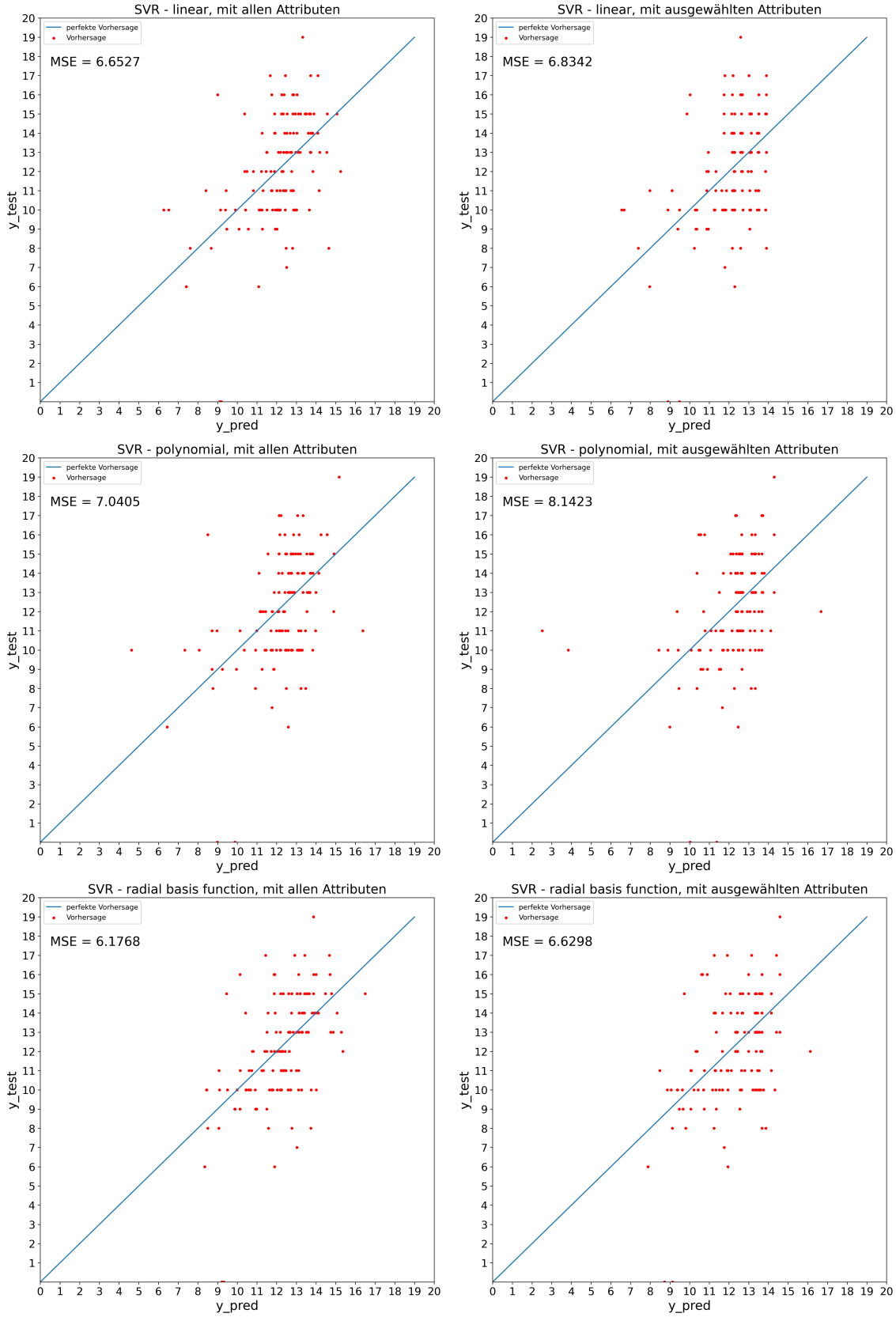


Abbildung 10: Vergleich der Lernerfolgsvorhersage mittels SVR mit unterschiedlichen Kernelfunktionen. Links sind die Vorhersagen bei denen alle Attribute miteinbezogen sind und rechts nur mit Attributen deren Korrelation min. 0.2/-0.2 ist.

Literatur

[1] Das, A. Support vector regression (svr) based prediction with r. https://www.youtube.com/watch?v=DBApaR2mTg0, 2022.

[2] Jahangiry, P. Part 23-support vector machines (svm) what are they? https://www.youtube.com/watch?v=V7mUNS3qXVY, 2022.

[3] Montgomery, D. C., Peck, E. A., and Vining, G. G. *Introduction to linear regression analysis*. John Wiley & Sons, 2021.

[4] scikitlearn. sklearn.svm.svr. https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html, 2022.

[5] Sethi, A. Support vector regression tutorial for machine learning. https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/, 2022.

[6] Sharp, T. An introduction to support vector regression (svr). https://towardsdatascience.com/an-introduction-to-support-vector-regression-svr-a3ebc1672c2, 2020.