

**Universität Paderborn**

**Fakultät für Wirtschaftswissenschaften**

**Department Wirtschaftsinformatik**

**Lehrstuhl für Wirtschaftsinformatik, insb. Data Analytics**

Studienarbeit

Predictive Analytics

von

Fabian Voß

7155877

Königstraße 35

59909 Bestwig

vfabian@mail.uni-paderborn.de

vorgelegt bei

**Erstkorrektor:**<Titel und Name des Erstkorrektors>

**Zweitkorrektor:**<Titel und Name des Zweitkorrektors>

Eingereicht am 28.02.2021

**Eidesstattliche Erklärung:**

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit selbstständig, ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus fremden Quellen (einschließlich elektronischer Quellen) direkt oder indirekt übernommenen Gedanken, Tabellen, Skizzen, Zeichnungen, bildliche Darstellungen usw. sind ausnahmslos als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden.

Paderborn, 28.02.2021

|  |
| --- |
| (Vorname Nachname) |

Inhaltsverzeichnis

[Abbildungsverzeichnis IV](#_Toc64983774)

[1 Business Understanding 1](#_Toc64983775)

[1.1 Problematik 1](#_Toc64983776)

[1.2 Zielsetzung 1](#_Toc64983777)

[1.3 Vorgehensweise 2](#_Toc64983778)

[2 Data Understanding 3](#_Toc64983779)

[2.1 Collect Initial Data 3](#_Toc64983780)

[2.2 Explore Data 3](#_Toc64983781)

[2.3 Verify Data Quality 3](#_Toc64983782)

[2.4 Data Interpretation 5](#_Toc64983783)

[3 Data Preparation 7](#_Toc64983784)

[3.1 Select Data 7](#_Toc64983785)

[3.2 Clean Data 7](#_Toc64983786)

[4 Modeling 8](#_Toc64983787)

[4.1 Select Model Technique 8](#_Toc64983788)

[4.2 Build Model 8](#_Toc64983789)

[4.3 Assess Model 9](#_Toc64983790)

[5 Evaluation 11](#_Toc64983791)

[5.1 Evaluate Results 11](#_Toc64983792)

[5.2 Review Process 11](#_Toc64983793)

[5.3 Determine Next Steps 11](#_Toc64983794)

[6 Deployment 12](#_Toc64983795)

[6.1 Plan Deployment 12](#_Toc64983796)

[6.2 Plan Monitoring and Maintanance 12](#_Toc64983797)

[6.3 Produce Final Report 12](#_Toc64983798)

[6.4 Review Project 12](#_Toc64983799)

[Literaturverzeichnis 13](#_Toc64983800)

Abbildungsverzeichnis

Abbildung 1: Ausschnitt Korrelationsmatrix mit Farbskala 4

Abbildung 2: Anteil Good/Bad Buy im Fahrzeugalter 5

Abbildung 3: Verhältnis Good/Bad Buy im Fahrzeugalter 6

*Abbildung 4: Verhältnis Good/Bad Buy bei unterschiedlicher Laufleistung* 6

Abbildung 5: Entscheidungsbaum ROC Curve 8

Abbildung 6: ROC Curve Random Forest 9

Abbildung 7: Tune Ranger Ergebnisse 10

Abbildung 8: ROC Curve 10

# Business Understanding

## Problematik

Der Gebrauchtwagenmarkt ist ein gutes Beispiel für asymmetrische Informationsketten, da der Käufer vor Vertragsabschluss die Qualität des Produktes nicht hundertprozentig kennt. Hinzu kommt, dass bei Auktionen zwischen zwei Bewertungskriterien unterschieden wird: defensive beziehungsweise offensive Bewertung. Die defensive Bewertung geschieht durch Mechaniker und Tester, welche alle Fehler oder Schwachstellen finden möchten. Die offensive Bewertung erfolgt durch das Marketing- und Verkaufspersonal (Anders Parment, 2016)

Trotz der defensiven Bewertungen besteht für den Käufer bei Auktionen von Gebrauchtwagen immer das Risiko eines Fehlkaufes. Daher besteht einer der größten Herausforderungen eines Käufers (z.B. Autohaus) das Risiko, dass das Fahrzeug schwerwiegende Probleme hat, zu verhindern. Der Fehlkauf kann dazu führen, dass die Reparaturkosten den Wert des Fahrzeuges überschreiten und somit ein Weiterverkauf an einen Kunden ausgeschlossen ist.

Ein Fehlkauf wird in Amerika als „lemon“ bezeichnet und insbesondere im Zusammenhang mit dem Gebrauchtwagenmarkt häufig erwähnt und analysiert. Genauer handelt es sich um ein Fahrzeug, das nach dem Kauf als Fehlkauf bewertet wird. Ein Fehlkauf ist genau dann vorhanden, wenn der tatsächliche Zustand des Fahrzeuges wegen zu reparierender Mängel nicht dem Kaufpreis gerecht wird. Genauer bewertet Eric W. Bond (1982) ein Fahrzeug als Fehlkauf, wenn innerhalb von zwölf Monaten ab Kauf eine große Reparatur in den Kategorien Motor, Getriebe, Bremsen oder Achsen erforderlich ist.

Durch die oben genannten Fehlkäufe fallen für den Autohändler Kosten an, die das Geschäft des Händlers negativ beeinflussen und zu einem unattraktiven Bestandsangebot für Kunden des Autohändlers führen.

## Zielsetzung

Das Ziel besteht darin, das Risiko eines Fehlkaufes zu verringern. Käufer\*innen sollen somit die „Kicks“ vermeiden und den Bestand an qualitativen Fahrzeugen erhöhen können. Weiter wird mit dem Aussortieren von potenziell schlechten Fahrzeugen ein zu hohes Verhältnis von Lemons auf dem Markt vermieden. Denn nach Akerlof (1970) drängen die Lemons die guten Fahrzeuge vom Markt, da diese zum gleichen Preis verkauft werden.

## Vorgehensweise

Um das oben beschriebene Ziel erreichen zu können, werden die im folgenden Absatz erklärten Kennzahlen des Datensatzes benutzt, um anhand dieser eine Vorhersage über den Kauf zu machen. Dabei steht „0“ für keinen Fehlkauf und die „1“ repräsentiert ein hohes Risiko eines fehlerhaften Fahrzeuges.

# Data Understanding

In diesem Kapitel wird der Datensatz analysiert, auf Zusammenhänge untersucht und mögliche Probleme werden herausgestellt. Der Datensatz wurde in Zusammenhang mit der Competition auf Kaggle durch die Firma „Carvana“ veröffentlich.

## Collect Initial Data

Der Datensatz wurde auf der Plattform Kaggle von dem Unternehmen „Carvana“ zur Verfügung gestellt und umfasst sechs Dateien. Nach dem Download aller Dateien in Form eines .zip Archives wurden die Daten entpackt.

Da die Daten nicht selbst erhoben wurden, umfasst der Datensatz eine Beschreibung aller Felder.

## Explore Data

Wie aus der Datei "Carvana\_Data\_Dictionary.txt" zu erkennen ist, bestehen die Datensätze aus 36 Spalten. Davon ist ein Attribut die ID (RefID) und eins die Bewertung, ob es ein schlechter Kauf war, oder nicht. Die weiteren Attribute beschreiben den Kauf des Fahrzeuges, also wann, wo und für wie viel es gekauft wurde.

Zwölf der 36 Attribute geben Werte über das Fahrzeug an sich an. Darunter fallen allgemeinere Daten wie das Baujahr, die Marke, das Modell und die Farbe. Darüber hinaus umfasst der Datensatz aber auch detailliertere Informationen über z.B. den Kilometerstand, das Getriebe und Ausstattungslevel.

Weitere acht Attribute beziehen sich auf die aktuelle Preislage des Fahrzeuges. Diese Preise wurden von dem Manheim Market Report erhoben und sind einige der genauesten Richtwerte für einen Gebrauchtwagenpreis (Autoauctionmall, 2018).

Die letzten Attribute geben Angaben darüber, wo das Fahrzeug gekauft wurde, wie hoch die Gewährleistung war und wann es zurück zur Auktion „gekickt“ wurde.

Diese Spalten finden sich in den Dateien „training.csv“, „test.csv“ und „example\_entry.csv“ wieder. Die mit Abstand größte Datei ist „training.csv“ mit 72983 Zeilen. Danach folgt die Datei „test.csv“ mit 48707 Zeilen und danach „example\_entry.csv“ mit ebenfalls 48707 Zeilen.

## Verify Data Quality

Betrachtet wird zunächst der Trainingsdatensatz mit 72983 Zeilen und 34 Spalten. Im Vergleich zu allen beschriebenen Spalten fehlen hier die Attribute „AcquisitionType“ und „KickDate“.

Im ersten Schritt, um die Daten richtig interpretieren zu können, wurden die Einträge, die keine Preisangabe hatten, ausgefiltert. Dies betrag 315 Datensätze und ist aufgrund der erheblichen Menge an Datensätzen zu vernachlässigen.

In dem Summary des Datensatzes fällt auf, dass das vorherzusagende Feld „IsBadBuy“ einen Mittelwert von 0.123 hat. Da es sich hierbei um ein Binärattribut handelt, besteht die Vermutung, dass der Trainingsdatensatz im Hinblick auf dieses Attribut nicht balanciert ist. Ein Balkendiagramm der gruppierten Daten bestätigt die Vermutung und zeigt, dass lediglich 8944 der insgesamt 72668 Einträge einen schlechten Kauf beinhalten. Lediglich ca. zwölf Prozent sind demnach schlechte Käufe.

Das Ungleichgewicht kann bei späteren Vorhersagen zu einer sehr schlechten Performance führen und muss im folgenden Kapitel zur Data Preparation beachtet werden.

Durch die im Notebook erstellte Summary des gesamten Datensatzes konnte festgestellt werden, dass die Attribute VehYear, VehicleAge, VehOdo gleichverteilt sind. Die Attribute, die den Preis beschreiben sind teilweise gleichverteilt und teilweise verschoben.

Zusammenhänge innerhalb des Datensatzes lassen sich mit Hilfe einer Korrelationsmatrix feststellen. Dazu wurden die relevanten, numerischen Werte ausgewählt und mit einer Farbskala versehen.

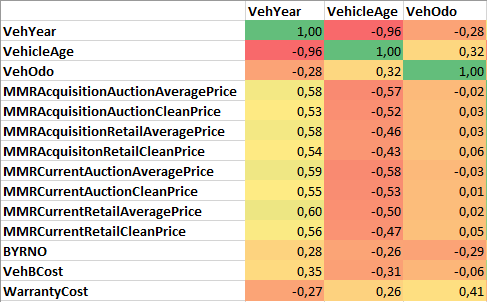


Abbildung 1: Ausschnitt Korrelationsmatrix mit Farbskala

Zwischen den Attributen VehYear und VehOdo besteht eine erhöhte Korrelation, was in Anbetracht des Zusammenhangs plausibel erscheint. Was auffällt ist, dass die Kosten weniger durch die gebrachte Laufleistung, sondern durch das Fahrzeugalter beeinflusst werden. Die Kosten für die Garantie hängen außerdem eher von dem Produktionsjahr des Fahrzeuges ab.

## Data Interpretation

Bezüglich der Interpretation der Daten besteht die Vermutung, dass im zunehmenden Alter der Fahrzeuge der Anteil der Fehlkäufe ansteigt. Um dies zu prüfen wurde das Fahrzeugalter auf der x – Achse und die Anzahl der Fahrzeuge, aufgeteilt nach Fehlkauf oder nicht, auf der y – Achse dargestellt.

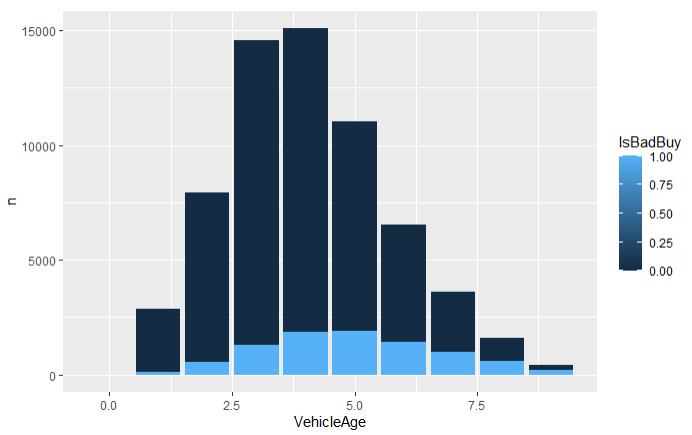


Abbildung 2: Anteil Good/Bad Buy im Fahrzeugalter

Die Grafik stützt die Vermutung, da der Anteil der Fehlkäufe immer weiter zu nimmt. Weiter verdeutlicht wird dieser Zusammenhang in folgender Grafik, in der auf der x – Achse wieder das Fahrzeugalter und auf der y – Achse der prozentuale Anteil der Fehlkäufe dargestellt ist.

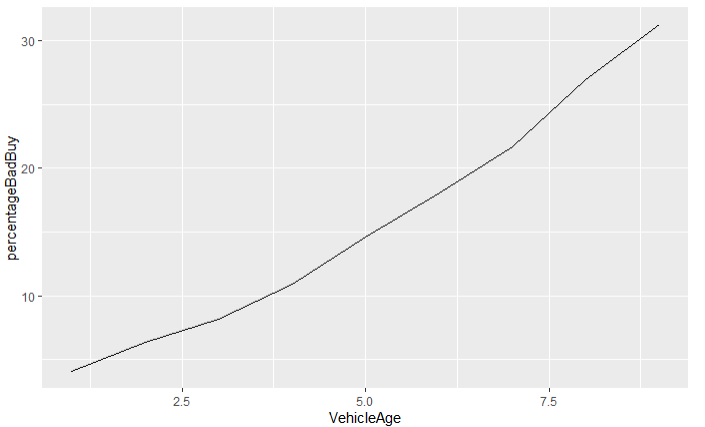
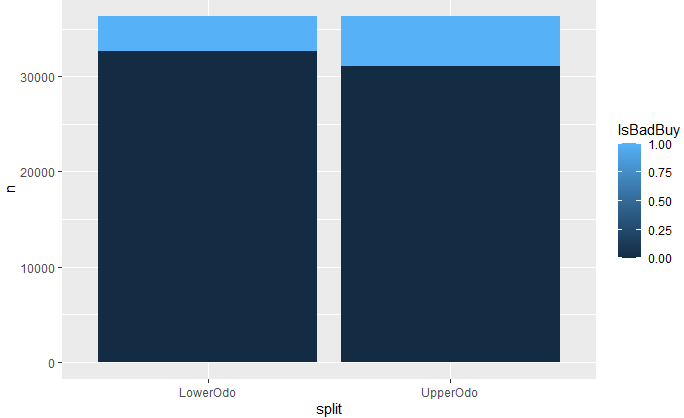


Abbildung 3: Verhältnis Good/Bad Buy im Fahrzeugalter

Weiter wurde die erbrachte Laufleistung im Zusammenhang mit den Fehlkäufen untersucht. Hierbei wurde zwar ein etwas erhöhter Anteil bei höherer Laufleistung festgestellt, im Hinblick auf den geringen Unterschied ist dies aber eher weniger aussagekräftig.

*Abbildung 4: Verhältnis Good/Bad Buy bei unterschiedlicher Laufleistung*

# Data Preparation

In diesem Abschnitt werden die Daten so vorbereitet, dass sie in der Modeling Phase benutzt werden können. Der erste Schritt ist die Wahl der relevanten Attribute für das Modell.

## Select Data

Die ausgewählten Daten sollen den größten Nutzen im Hinblick auf unser Modell geben. Daher können zunächst die ID’s entfernt werden, also die RefID und die BYRNO. Ebenso fällt das Attribut PRIMEUNIT heraus, da nur weniger als 4000 Daten vorhanden sind. Auch ist die Anzahl bei dem Attribut AUCGUART zu gering, damit es aussagekräftig sein kann.

## Clean Data

Im Datensatz sind, wie oben genannt, numerische und kategorische Variablen vorhanden. Die fehlenden kategorischen Werte können entweder gelöscht oder durch einen gemeinsamen Eintrag ersetzt werden. In diesem Fall werden die mit „NULL“ importierten Einträge durch „UNKNOWN“ ersetzt, sodass diese bei der Umwandlung in Faktoren berücksichtig werden können.

Weiter werden die numerischen Attribute betrachtet. Die im Datensatz fehlenden Werte haben den Wert 0. Diese Einträge werden mit dem Durchschnitt des jeweiligen Attributes ersetzt.

Die Attribute „Model“ und „SubModel“ enthalten noch weitere Informationen, die über die eigentliche Modellbezeichnung hinaus gehen. Deswegen wurden die Informationen aus den Feldern in neue Attribute unterteilt und dem Data Frame hinzugefügt. Die so entstandenen Attribute sind „Doors“, „Cylinder“, „Liter“ und „WheelDrive“. Da die Informationen nicht bei allen Datenreihen gepflegt sind, werden fehlende Werte mit „UNKNWON“ ergänzt.

Im nächsten Schritt wurde der Datensatz mit Hilfe von Over- und Undersampling balanciert, sodass die Zielvariable „IsBadBuy“ gleich verteilt ist. Der daraus entstandene Datensatz beinhaltet insgesamt ca. 18000 Einträge, die in etwa zu gleicher Anzahl aus je Fehlkäufen und nicht Fehlkäufen bestehen. Damit wurden mehr Datensätze von erfolgreichen Käufen entfernt als Fehlkäufe hinzugefügt, um den Datensatz so weit es geht mit originalen Daten zu belassen.

Zum Over- und Undersampling wurde das Paket Rose verwendet.

# Modeling

In diesem Abschnitt wird das Model zur Vorhersage der Zielvariable erarbeitet. Berücksichtig werden hier frühere Arbeiten, die bereits den besten Algorithmus für das hier vorliegende Problem herausgearbeitet haben.

## Select Model Technique

Als Model Technik werden Entscheidungsbäume genutzt, da diese für Klassifizierungsaufgaben mit vielen Merkmalen sehr gute Performance bringen (Ruiz-Gazen & Villa, 2007). Zunächst wird also ein einzelner Entscheidungsbaum erstellt und dieser dann weiter ausgewertet und evaluiert.

## Build Model

Da nach der Data Preparation der balancierte und bereinigte Datensatz vorliegt, kann mit dem Erstellen eines Modells begonnen werden. Dafür wird der Trainingsdatensatz in 80% Trainingsdaten und 20% Validierungsdaten aufgeteilt. Das Model wird auf den Trainingsdaten trainiert und hinterher mit den Validierungsdaten bewertet.

Um den Entscheidungsbaum aufzubauen wird das Paket und die gleichnamige Funktion „rpart“ benutzt. Als Methode wird „class“ angegeben und der Baum wird auf den zuvor erstellten Trainingsdaten erstellt. Weiter wird der so erstellte Entscheidungsbaum zur Vorhersage der Daten des Validierungssets benutzt. Zur ersten Evaluierung wird eine ROC Curve erstellt und die Area Under The Curve berechnet.

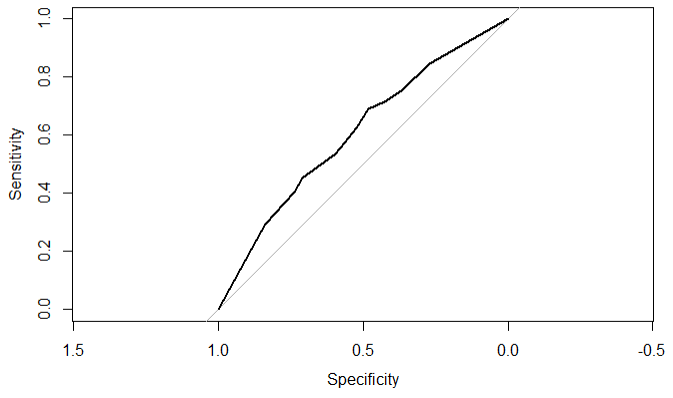


Abbildung 5: Entscheidungsbaum ROC Curve

Der so entstandene Entscheidungsbaum hat eine Area Under The Curve von 0,6057.

Als weiterführende Model Technik wird Random Forest aus dem Bereich des Supervised Learning gewählt. Das Klassifizierungs- bzw. Regressionsverfahren bietet die Vorteile wenig Rechenleistung auch bei größeren Datensätzen in Anspruch zu nehmen. Weiter kann das Verfahren mit einem Datensatz umgehen, der viele Merkmale besitzt (Hackl, 2015).

Um das Model zu entwickeln, wird das Paket „ranger“ installiert und importiert. Mit diesem Paket kann der Random Forest Algorithmus nach Breimann implementiert werden. Als Splitrule wird „Gini“ verwendet.

Die Anzahl der zu berechnenden Bäume wird auf 1280 festgelegt, was ca. 10% der Anzahl der vorhandenen Trainingsdatensätzen entspricht. Aus dem entstandenen Random Forest Modell ergibt sich eine Area Under The Curve von 0,6845. Dies entspricht einer Verbesserung zum vorherigen Modell um 0,0788.

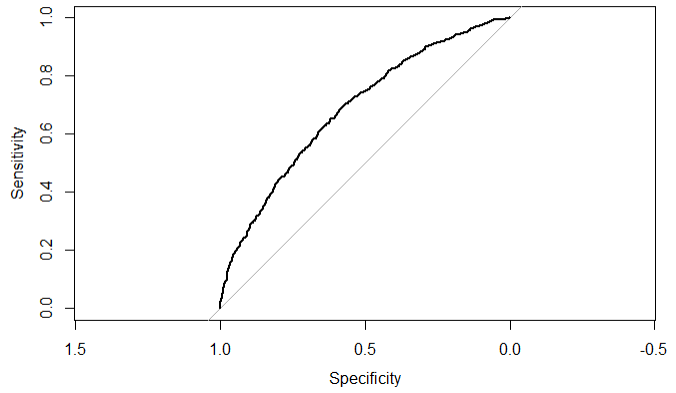


Abbildung 6: ROC Curve Random Forest

## Assess Model

Um eine bestmögliche Vorhersage ermöglichen zu können, wird mit Hilfe von dem Paket „tuneRanger“ zunächst ein Klassifizierungstask angelegt. Mit Hilfe von diesem Task werden die Hyperparameter „mtry“, „min.node.size“ und „sample.fraction“ getuned. Dabei werden 1500 Bäume pro Durchgang erstellt und mit der Bewertungsmethode „Area Under The Curve“ bewertet. Es werden 30 „warm-up“ und 80 normale Durchgänge durchgeführt, um die bestmöglichen Parameter für das Modell zu berechnen. Die berechneten Parameter sehen wie folgt aus:

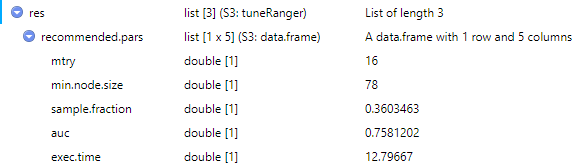


Abbildung 7: Tune Ranger Ergebnisse

Die berechneten Parameter sind in dem Model „res$model$learner.model“ bereits hinterlegt und können direkt zur Vorhersage von Werten benutzt werden. Die Vorhersage erfolgt vom Typ „response“, sodass ein Wahrscheinlichkeitswert für die Vorhersage erstellt wird. Je größer dieser Wert ist, umso größer ist die Wahrscheinlichkeit, dass es sich um einen Fehlkauf handelt.

Nach der Vorhersage der Validierungsdaten ergibt sich eine Area Under The Curve von 0,6926. Die ROC Curve sieht wie folgt aus:

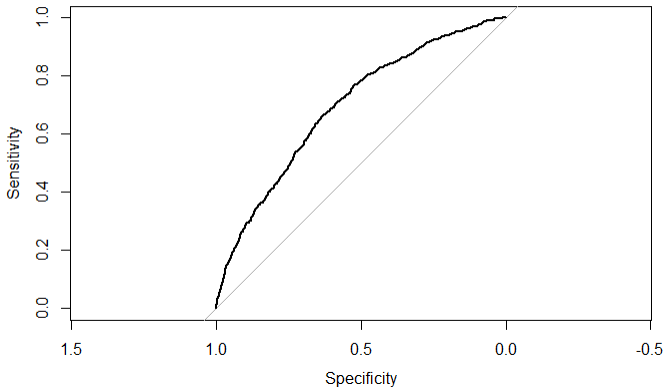


Abbildung 8: ROC Curve

# Evaluation

## Evaluate Results

Das erstellte Model, welches auf den Ergebnissen des optimierten Random Forest basiert, liefert eine Area Under The Curve von ca. 0,69. Außerdem wurde durch die Funktion „ineq“ ein Gini Score der Validierungsdaten von 0,495 berechnet. Bei der Teilnahme an der Kaggle Challenge wurde mit diesem Modell ein Gini Score von ca. 0,10390 erzielt, welcher im Vergleich zu den veröffentlichten Notebooks unter dem Reiter „Code“ in der Challenge gut abschneidet, im Hinblick auf den Gewinner Score aber noch Verbesserungspotential hat (vgl. https://www.kaggle.com/c/DontGetKicked).

Das im Abschnitt Business Understanding festgelegte Ziel war, eine Entscheidungshilfe für den Kauf eines Fahrzeuges bieten und damit das Risiko eines Fehlkaufes besser abschätzen zu können. Das Modell kann als weiteres Entscheidungskriterium in Betracht gezogen werden und erfüllt somit die im Business Problem definierten Anforderungen.

## Review Process

Die im Kapitel Data Understanding festgestellten Probleme konnten durch Undersampling und Modifizierung der einzelnen Werte im Kapitel Data Preparation gelöst werden. Informationen, die nicht atomar in den Merkmalen Model und Submodel beinhaltet waren, konnten in zusätzliche Merkmale extrahiert werden und numerische Merkmale ohne Wert konnten ebenfalls durch die jeweiligen Durchschnittswerte angepasst werden.

Der Prozess zur Entwicklung des Modells zeigte ständige Verbesserung in der Bewertungsmethode Area Under The Curve und konnte gleichzeitig immer höhere Werte bei der Abgabe auf Kaggle erzielen. Die stetige Verbesserung vom einzelnen Entscheidungsbaum über den Random Forest bis hin zum Random Forest mit optimierten Hyperparametern spricht für eine positive Entwicklung innerhalb des Modellierungsprozesses.

## Determine Next Steps

Die nächsten Schritte beinhalten die Erstellung eines Deployment Plans, sowie wichtige Aspekte im Monitoring des gesamten Modells. Ebenso wird das Modell in Software umgesetzt, die im Hinblick auf das Business Problem von einem End User bedient werden kann. So kann das hier entwickelte Modell einen Weg in die Praxis finden.

# Deployment

## Plan Deployment

Zum Ausliefern des Modells und zur praktischen Anwendung wird eine Shiny App entwickelt, die eine einfache UI beinhaltet, die die Möglichkeit zur Eingabe von Fahrzeugdaten bietet. Die dort eingegebenen Daten werden einem Data Frame hinzugefügt. Mit Hilfe eines vorher trainierten Modells erfolgt dann die Vorhersage der Eingabe.

Der Output der Vorhersage wird in Prozent ausgegeben, sodass eine Einschätzung über die Wahrscheinlichkeit eines Fehlkaufes erfolgen kann. Die erstellte Shiny App kann im lokalen Netzwerk oder auf einem öffentlichen Server gehostet werden.

## Produce Final Report

Das im Business Understanding analysierte Problem auf einem asymmetrischen Markt konnte mit Hilfe des von Carvana bereitgestellten Datensatzes insoweit gelöst werden, dass die bereitgestellte Software eine Entscheidungshilfe bieten kann. Das Modell der Software basiert auf einem Random Forest mit optimierten Hyperparametern, welcher auf dem zuvor durch Undersampling balancierten Datensatz trainiert wurde. Das finale Produkt bietet dem User die Möglichkeit die gängigen Automodelle mit individuellen Daten einzugeben und einen Risikowert zu berechnen.

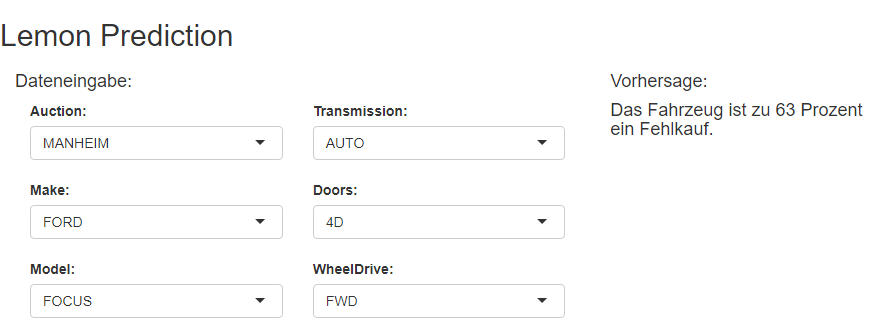


Abbildung 9: Ausschnitt der finalen Software

## Review Project

Das Projekt behandelte alle Elemente des CRISP-DM Modells und stellte anfangs ein Ziel auf, welches während des Projektvorgangs stets im Fokus stand. Das Ziel, eine Risikoermittlung / Kaufempfehlung für ein Gebrauchtwagen zu erstellen, konnte somit erfolgreich umgesetzt werden.

# Literaturverzeichnis

Akerlof, G. (1970). The Market for "Lemons": Quality Uncertainty and the Market Mechanism. *The Quarterly Journal of Economics,* *84*(3), 488-500. Retrieved February 04, 2021, from http://www.jstor.org/stable/1879431

Bond, E. (1982). A Direct Test of the "Lemons" Model: The Market for Used Pickup Trucks. *The American Economic Review,* *72*(4), 836-840. Retrieved February 04, 2021, from <http://www.jstor.org/stable/1810022>

Dwivedi, Y. K., Wade, M. R. & Schneberger, S. L. (2011). *Information Systems Theory: Explaining and Predicting Our Digital Society, Vol. 1 (Integrated Series in Information Systems Book 28) (English Edition)* (2012. Aufl.). Springer. https://doi.org/10.1007/978-1-4419-6108-2\_11

Ho, A., Romano, R. & Wu, X. A. (2012, 14. Dezember). *Don’t Get Kicked -Machine Learning Predictions for Car Buying*. stanford.edu. http://cs229.stanford.edu/proj2012/HoRomanoWu-KickedCarPrediction.pdf

Lichtendahl, K. C. (2020, 13. September). *Carvana: IsBadBuy? by Kenneth C. Lichtendahl :: SSRN* [Vorlesungsfolien]. papers.ssrn.com. https://papers.ssrn.com/sol3/papers.cfm?abstract\_id=3614450

Miltic, N. (2016). Predicting risk of buying low quality car with rapid miner. In University of Belgrade, A. Jankovic & M. Vukicevic (Hrsg.), *Data science and business intelligence* (S. 201–206). University of Belgrade.

*MMR Under the Hood*. (o. D.). publish.manheim.com. Abgerufen am 6. Januar 2021, von https://publish.manheim.com/en/help/mmr/mmr-under-the-hood.html

Oswaldo F. Domejean, O. D. (2014, Mai). *Data Science with Kaggle’s Competition „Don’t Get kicked!“* Researchgate. https://www.researchgate.net/profile/Oswaldo-Figueroa-Domejean/publication/262523736\_Data\_Science\_with\_Kaggles\_Competition\_Dont\_Get\_kicked/links/02e7e537e4b8e6c97b000000/Data-Science-with-Kaggle-s-Competition-Don-t-Get-kicked.pdf

Parment, A. (2016). *Die Zukunft des Autohandels: Vertrieb und Konsumentenverhalten im Wandel - Wie das Auto benutzt, betrachtet und gekauft wird* (1. Aufl. 2016 Aufl.). Springer Gabler.

Ruiz, A. (2008, 4. April). *Storms prediction : Logistic regression vs random forest for unbalanced data*. arXiv.org. https://arxiv.org/abs/0804.0650

Team, A. A. M. (2019, 6. Juni). *What does the Manheim Market Report (MMR) mean in Car Auction?* Auto Auction Mall. https://www.autoauctionmall.com/learning-center/what-does-mmr-mean/#:%7E:text=MMR%20in%20the%20car%20business,an%20indicator%20of%20wholesale%20prices.

Wirth, R. & Hipp, J. (2000). *CRISP-DM: Towards a Standard Process Model for Data Mining*. cs.unibo.it. http://www.cs.unibo.it/~danilo.montesi/CBD/Beatriz/10.1.1.198.5133.pdf

Yavas, B. F., Freed, R. & Vardiabasis, D. (2008). Uncertainty, the Lemon Problem, Asymmetric Information and Countertrade. *Taylor & Francis*. https://www.tandfonline.com/doi/abs/10.1300/J130v05n01\_02

Anhang