KUZ
Kunststoff-Zentrum Leipzig

Technische Hochschule
Brandenburg
University of
Applied Sciences

Bachelorarbeit zum Thema:

Entwicklung einer Methodik zur Qualitätsvorhersage im Spritzgießprozess mittels maschinellen Lernens am Beispiel des Formteilgewichtes

von
Tim Garbe

Betreuung
Dipl.-Inform. Ingo Boersch
Prof. Dr.-Ing. Jochen Heinsohn
Technische Hochschule Brandenburg
Dipl.-Ing. Stefan Lehmann
Kunststoff-Zentrum in Leipzig gGmbH

Technische Hochschule Brandenburg

12. Oktober 2021

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Die Bachelorarbeit findet im Anschluss an ein Praktikum im Kunststoff-Zentrum Leipzig (KUZ) statt. Hauptaufgabe des Praktikums war es, eine unterstützende Toolchain[1] als Python Package für die Datenanalyse zu entwickeln, welche in dieser Arbeit Anwendung findet.

Die Bachelorarbeit erfolgt im Rahmen des Forschungsprojekts „KIQuality“ des KUZ, welches sich das Ziel gesetzt hat, die Zusammenhänge von Prozessparametern und Qualitätskenngrößen zu untersuchen und diese mittels maschinellen Lernens abzubilden. Hierbei soll u. a. eine gesamtheitliche Betrachtung des Spritzgießsystems und der Prozessparameter durchgeführt werden. Ein Spritzgießsystem, wie in Abbildung 1.1 zu sehen, kann zur Verarbeitung von Kunststoffen bzw. zur Herstellung von Kunststoff-Formteilen verwendet werden.[4]

Abbildung 1.1: Spritzgießmaschine Sumitomo Demag [7]

Die Herstellung eines Formteiles umfasst einen zyklischen Prozess, bei dem mehrere Prozessparameter auftreten. Jene Prozessparameter sollen im Zuge des KIQuality Projektes zur Vorhersage der Qualität der Formteile genutzt werden. Zu den Qualitätsmerkmalen zählen unter anderem die Bemaßung, Oberfläche, Farbe und das Gewicht. Die Feststellung über die Höhe der Qualität gestaltet sich für die zuvor genannten Merkmale aufwendig. Um diesen Aufwand zu reduzieren, soll künstliche Intelligenz bzw. maschinelles Lernen zur Qualitätsvorhersage angewendet werden. In der Bachelorarbeit wird hierzu das Formteilgewicht als Qualitätsmerkmal zur KI-gestützten Vorhersage betrachtet. Dabei wird eine Methodik von der Datenerfassung bis hin zur Modellerstellung und -bewertung entwickelt, welche sich auf weitere Qualitätsmerkmale anwenden lassen soll.

[1] Ansammlung von Methoden und Funktionen für komplexere Aufgabenstellungen

Zudem wird in der nachstehenden Arbeit ermittelt, welche Vorverarbeitungen der Daten für Analysen und Vorhersagen notwendig sind.

Mit Hilfe einer KI-gesteuerten Qualitätsprognose könnten beispielsweise fehlerhafte Formteile erkannt und automatisiert aussortiert werden. Weiterführend könnte dem Spritzgießer ein automatisierter Handlungsvorschlag übergeben oder eine automatische Justierung der Maschine anhand der Prozessparameter durchgeführt werden. Vorteile wären unter anderem weniger Ausschuss und somit die Einsparung von Ressourcen, wie beispielsweise Material und Energie.

Das Ergebnis der Arbeit ist die Methode der Vorverarbeitung und der damit verbundene Datensatz, sowie Zusammenhänge der Prozessparameter durch eine explorative Analyse und Konzept des Evaluationsprozesses der KI-Modelle.Im ersten Kapitel der Arbeit werden unter anderem die Grundlagen zum Spritzgießprozess, zur Künstlichen Intelligenz und zur vorhandenen Infrastruktur erläutert. Im darauf folgenden Gliederungspunkt wird der erste Datensatz zur explorativen Analyse verwendet. Die sich daraus ergebenden Schritte zur Vorverarbeitung stellen ein Ergebnis der Arbeit dar und werden im Anschluss zusammengefasst. Nach der Vorverarbeitung der Daten werden diese im darauf folgenden Kapitel zum Anlernen der Modelle genutzt. Des Weiteren findet die Bestimmung relevanter Merkmale sowie die Erläuterung des Konzeptes zum Evaluationsprozess der Modelle statt. Abschließend erfolgt eine zusammenfassende Betrachtung der Methodik und ein Ausblick auf Anwendungsmöglichkeiten.

2 Grundlagen

2.1 Spritzgießprozess

Im Spritzgießprozess wird meist thermoplastischer Kunststoff verarbeitet, um ein Produkt (sog. Formteil) herzustellen. Dabei durchläuft die Herstellung eines Formteiles mehrere Phasen. Zunächst wird das Kunststoffgranulat in einen Trichter gegeben, durch den es auf eine Förderschnecke gelangt. Das Granulat wird durch Rotation der Schnecke vorangetrieben. Dabei entsteht Reibungshitze, die das Kunststoffgranulat neben den zusätzlichen Heizzonen erhitzt. Nachdem der Kunststoff plastifiziert und dosiert ist, wird dieser durch eine Vorwärtsbewegung der Schnecke in das geschlossene Werkzeug gegeben. Der Einspritzsprozess erfolgt zunächst geschwindigkeitsgeregelt entsprechend eines eingestellten Geschwindigkeitsprofils. Im Umschaltpunkt erfolgt der Wechsel in den Nachdruck. Der Nachdruck ist zum Ausgleich der Materialschwindung während des Auskühlens. [6, S. 1-2] Ist das Formteil erkaltet öffnet sich das Werkzeug, sodass das Formteil entnommen werden kann. Dabei kann das Formteil mittels Auswerferstifte vom Werkzeug entfernt und in einer Kiste gesammelt werden. Im Falle des KUZ wurde für das „KIQuality“ Forschungsprojekt ein sogenanntes „Handlingsystem“ eingerichtet, welches das Formteil automatisiert entnimmt. Das Handlingsystem ist ein 3-Achsen-Roboter mit Saugnoppen, der das Formteil mittels Unterdruck entnimmt und an sich haften lässt. Dabei hat das Handlingsystem einen vordefinierten Ablauf an Bewegungen, bei dem u.a. das Formteil kurzzeitig zum Wiegen auf eine Waage gelegt wird. Ein Zyklus (oder auch Schuss) beschreibt dabei das Durchlaufen aller Phasen, vom Aufdosieren des Kunststoffes bis hin zur Entnahme des Formteiles.

2.2 Formteilgewicht

Das Formteilgewicht stellt ein Qualitätsmerkmal dar. Qualitätsmerkmale beim Spritzgießen zu erkennen ist wichtig um Fehler oder Abweichungen in der Produktion zu erkennen. [1, S. 1] Das Formteil wurde speziell für das Forschungsprojekt „KIQuality“ zur wissenschaftlichen Untersuchungen und zur Erfassung von Prozessdaten entwickelt.

2.3 Maschinelles Lernen

Maschinelles Lernen umfasst Algorithmen, die Zusammenhänge aus Daten identifizieren und abbilden können. Hervorgehend aus diesen ermittelten Zusammenhängen kann wiederum Wissen generiert werden. [14, S. 29]

Maschinelles Lernen stellt einen Teilbereich der Künstlichen Intelligenz dar, wobei in der Art des Feedbacks, zwischen dem überwachten Lernen (supervised), dem unüberwachten Lernen (unsupervised) und dem verstärkenden Lernen (reinforcement) unterschieden wird. [2, S. 274]

Das überwachte Lernen lässt sich in Klassifikation und Regression unterteilen. Bei der Klassifikation wird durch das Lernen, – basierend auf Beobachtung –, die Zuordnung von neuen Instanzen zu kategorialen Klassen vorgenommen. Die Regression beinhaltet hingegen die Ermittlung von Zusammenhängen zwischen erklärenden Variablen und Zielvariablen und damit die Vorhersage von numerischen Ergebnissen. Dies wird auch als Regressionsanalyse bezeichnet. [14]

2.4 Programmierung

Python stellt eine Programmiersprache dar, welche leicht erlernbar ist und dennoch die Entwicklung von Programmen für komplexe Sachverhalte erlaubt. Python bietet hierbei diverse Vorteile und zählt deshalb zu einer der bevorzugten Programmiersprachen im Bereich Data Science . [16, S. 13-14][14, S. 42] Ein besonderer Vorteil ergibt sich für den wissenschaftlichen Bereich vor allem durch die Verfügbarkeit von Drittanbieterressourcen, wie unter anderem Pandas zur Manipulation von heterogenen und gekennzeichneten sowie NumPy zur Manipulation homogener arraybasierter Daten. [17, S. xii bis xiii] Eine Zusammenführung der essentiellen Python-Packages - wie Pandas oder NumPy - findet sich in der Distribution Anaconda wieder. [14, S. 43]

2.5 Detact

„Detact“ ist eine Software der Firma Symate GmbH, welche auf die automatisierte Datensammlung, -verarbeitung und -analyse von Spritzgießsystemen spezialisiert ist. Im KUZ hat die Software zum Zeitpunkt der Bachelorarbeit folgende Anwendungsfälle:
- Zentrales Ablegen aller Prozessparameter
- Graphische Darstellung gewünschter Prozessparameter
- Manuelles Eintragen von Prozessdaten

Für die Bachelorarbeit wurde die Software „Detact“ als Datenbank mit graphischer Oberfläche benutzt. Dabei wurde der Zugang zu den Daten über eine REST API realisiert.

3 Portrait der Kunststoff-Zentrum in Leipzig gGmbH

Die heutige Kunststoff-Zentrum in Leipzig gGmbH (KUZ) wurde 1960 als Zentrallaboratorium für Plastverarbeitung gegründet. Das KUZ ist ein industrienahes und gemeinnütziges Forschungsinstitut, das über 60 Mitarbeiter beschäftigt. [10]

Kunststoffteile finden sich in zahlreichen Produkten wieder, weshalb der Branche der kunstoffverarbeitenden Industrie eine besondere Bedeutung als Zuliefererindustrie zukommt. Dabei sind die Kunststoffteile häufig entscheidend für die Funktionalität der Baugruppen bzw. Bauteile, in denen sie verarbeitet worden sind. Der Fokus der Forschungs- und Entwicklungsarbeiten des KUZ liegt auf der Betrachtung des gesamten Herstellungsprozesses von Formteilen bzw. Baugruppen mit verarbeiteten Kunststoff, wobei die Hauptkompetenzen im Spritzgießen, der Polyurethan-Reaktionstechnik und dem Schweißen, Nieten und Bördeln liegen.[8, S. 7-8] Die Forschungsfelder des KUZ gliedern sich in vier Schwerpunkte, welche in der nachfolgenden Tabelle wiedergegeben werden:

Forschungsschwerpunkte	**Forschungsprojekte**
Technologie- und Innovationsforschung	Untersuchungen zur Fügenahtgestaltung Verbesserte Antistatik in Sandwich-Bauteilen 3D-HRD
Digitalisierung / Künstliche Intelligenz	KIQuality Smarter US-Nietassistent Digitalisierung der PUR-Verarbeitung KIScha
Leichtbau	Funktionsträger RecySchaum Maßhaltigkeit geschäumter Spritzgießteile
Mikrokunststofftechnik	Bierstick Zwanzig 20 – RESPONSE

Tabelle 3.1: Forschungsschwerpunkte 2020 im Kunststoff-Zentrum Leipzig

Wie eingangs erwähnt ist die Bachelorarbeit dem Projekt KIQuality zuzuordnen, welches das primäre Zier verfolgt, die Qualität von Spritzgießformteilen mittels KI-unterstützter Fehleranalyse vorherzusagen. [9, S. 7]

4 Datengenerierung

4.1 Herkunft der Daten

Das im KUZ verwendete System umfasst eine speziell ausgerüstete Versuchsanlage, die für wissenschaftliche Untersuchungen und zur Erfassung von Prozessdaten eingesetzt wird. Der Vorteil gegenüber einer ausschließlich produzierenden Anlage ist, dass die Maschineneinstellungen beliebig variiert werden können, da nicht nur Gutteile[2] produziert werden müssen.

Um die Vergleichbarkeit der einzelnen Versuche zu gewährleisten, wurden ausschließlich Daten aus einem stabil laufenden Spritzgießprozess verwendet. Wird beispielsweise eine Soll-Temperatur verändert, benötigt die Maschine eine gewisse Zeit, bis die Temperatur erreicht ist. Der Spritzgießer an der Maschine erkennt einen stabilen Prozesszustand anhand konstanter Prozessparameter und gleichmäßig schwankendem Formteilgewicht, und setzt den Stückzähler der Maschine zurück. Der Stückzähler ist hierbei ein Prozessparameter, der von der Maschine nach jedem Zyklus inkrementiert wird, beginnend bei Null. Indem der Spritzgießer den Stückzähler zurücksetzt, gibt dieser den Start eines Versuches an. Hat der Stückzähler den Wert 59 angenommen, ist der Versuch beendet. Ein Versuch kann eindeutig durch Versuchsreihe und Versuchsnummer identifiziert werden. Dabei ist ein Versuch durch eine für mindestens 60 Schuss gleichbleibende Parametereinstellung definiert. Anhand dieser 60 Schuss können Schwankungen innerhalb der Versuche berücksichtigt und ein repräsentativer Wert für alle Prozessparameter gefunden werden. Anhand der Versuchsreihe lässt sich erkennen, welche Soll-Parameter[3] zueinander verändert wurden, um die spätere Analyse zu vereinfachen. Welche Soll-Parameter aktiv vom Spritzgießer verändert wurden und in welchem Maße, kann den vom Experten erstellten Versuchsplänen entnommen werden. Bei den Versuchsplänen handelt es sich um teilfaktorielle Versuchspläne, da ein Vollfaktorplan durch die hohe Anzahl an Faktoren nicht mehr durchführbar ist. Dabei definiert der Experte für jeden Soll-Parameter einen Bereich, in dem dieser variiert wird. Das ist aufgrund physikalischer, thermischer und technologischer Grenzen erforderlich. [15, S. 28]

Für jede Versuchsreihe werden Soll-Parameter ausgewählt, bei denen ein Zusammenhang zum Formteilgewicht vermutet wird. Angenommen für jeden Soll-Parameter wird zwischen seinem Minimum und Maximum gleichmäßig in zehn Schritten iteriert, so müssten bereits bei fünf Soll-Parametern 1.000.000 (10^5) Versuche stattfinden. Bei einer Versuchsdauer[4] von ungefähr 30 Minuten, würden die Versuche mehrere Jahrzehnte in Anspruch

[2] Formteile ohne Formteilfehler
[3] Wert den die Maschine annehmen soll / Teilmenge der Prozessparameter
[4] Die Zeit in der 60 gültige Zyklen durchlaufen werden

nehmen. Aufgrund dieser exponentiell wachsenden Wissensbasis ist eine Begrenzung der Versuche durch den Experten für eine zeitnahe Analyse notwendig. Die Wissensbasis besteht aus deklarativem Wissen aus den Versuchen und ist dabei ein Abbild des gesamten Wissensraumes.

4.2 Datenabfrage

Nachdem die Daten anhand der Versuchspläne generiert wurden, können diese mit wenigen Ausnahmen aus der Datenbank (Detact) entnommen werden. Die Ausnahme beschränkt sich auf eine Excel-Datei mit Informationen über die Zeiträume, in denen Versuche stattfanden und Gültigkeit der Versuche. Die Datensätze aus der Datenbank kann über eine REST-API abgefragt werden. Dafür ist im KUZ bereits das hausinterne Python package namens pydetact vorhanden, welches die verschiedenen Anfragen an die REST-API und die Authentifizierung übernimmt. ur Verarbeitung der Prozessdaten wird programmatisch ein spezielles Datenformat (pandas DataFrame) verwendet. Ein Anwendungsbeispiel zur Datenabfrage vom Detact sieht wie folgt aus.

Quelltext 4.1: Datenabfrage in Python mit pydetact

```
import pandas as pd
import numpy as np
from pydetact import detact as de
from pydetact import kiquality as kq

t1 = "2021-06-18T05:00:00" # Startzeitpunkt
t2 = "2021-08-27T18:00:00" # Ende Datensatz BA
params_df = de.get_parameters(filter_dict={'tag':'KIQuality'})
df = de.get_data(params_df, t1, t2)
```

Die Funktion `get_parameters()` liefert ein DataFrame, welches eine Auflistung aller Prozessparameter enthält. Hierbei kann bereits über die Parameterliste mittels Dictionary gefiltert werden. Der DataFrame wird wiederum als Eingabe für die `get_data()` Funktion benutzt, welche die Inhalte aus dem DataFrame zur Datenabfrage verwendet. Dabei kann der Zeitraum über `t1` und `t2` definiert werden. Der Rückgabewert ist der unformatierte Datensatz als DataFrame.

Mit `df.shape` kann die Form bzw. der Umfang des DataFrames ausgegeben werden. Die Variable `df` steht in dieser Arbeit immer repräsentativ für ein DataFrame. Der verwendete Datensatz bezieht sich auf die Versuche im Zeitraum vom 18.06.2021 bis 27.08.2021 und hat nach der ersten Abfrage 1.971.868 Zeilen und 30 Spalten.

4.3 Datentransformation

Da nun die Daten in Python als DataFrame vorliegen, kann im nächsten Schritt der Datensatz mit Hilfe von Python untersucht werden. Aus einer Ausgabe der Daten lässt sich

erkennen, dass der Index aus Zeitpunkten und die Spaltenbezeichnungen aus den bereits im Vorfeld gefilterten Prozessparametern bestehen. Um Zusammenhänge der Prozessparameter zum Formteilgewicht erkennen zu können, werden diese zunächst je Zyklus zusammengefasst. Somit soll final jede Datenzeile alle Daten der Prozessparameter (inklusive Formteilgewicht) enthalten, die in dem jeweiligen Zyklus auftreten. Eine solche Zusammenfassung kann unter Umständen einen Informationsverlust bedeuten, da Prozessparameter, die mehrere Daten in einem Zyklus übermitteln, durch die Aggregation auf einen Wert reduziert werden. Für alle in dieser Arbeit verwendeten Prozessparameter ist dieser Wert representativ für den gesamten Zyklus. Übermittelt beispielsweise eine Heizzone mehrere folgende relativ konstante Werte in einem Zyklus von 30 Sekunden (199.98°C, 199.99°C, 200.00°C, 200.01°C, 200.02°C), so kann das arithmetische Mittel (von 200°C) als repräsentativer Wert für diesen Zyklus verwendet werden.

Bei großen Abweichungen der Werte innerhalb eines Zyklus ist die Abstrahierung auf das arithmetische Mittel unter Umständen nicht ausreichend, da wichtige Informationen verloren gehen können. Dieser Fall tritt bei Verlaufsparametern auf. Ein beispielhafter Verlaufsparameter ist der Werkzeuginnendruck, dessen Kurvenform zusätzliche Informationen über den Prozess enthält. Für solche Verlaufsparameter ist es unter Umständen sinnvoll, kurvenbeschreibende Merkmale zu extrahieren. Welche Merkmale aus den jeweiligen Verlaufsparametern für ein Prognosemodell extrahiert werden sollen, erfordert Prozesswissen, welches im Expertengespräch ermittelt werden kann. Beispiele für Merkmale, welche aus einem Verlaufsparameter extrahiert werden können, sind: Extremwert, Mittelwert, Steigung an bestimmter Stelle, polynomiale Beschreibung oder Ableitung. Jedes extrahierte Merkmal wird dann als zusätzliche Spalte dem Datensatz hinzugefügt. Da diese Arbeit lediglich einen ersten Einblick in eine mögliche Methodik zur Qualitätsprognose liefern soll, wurde an dieser Stelle aufgrund des zeitlichen Aufwandes, die Extrahierung der Merkmale aus den Verlaufsdaten nicht behandelt. Die zyklische Zusammenfassung der Daten wird in Abbildung 4.1 allgemein dargestellt. Dabei sind alle Werte, die sich auf einen bestimmten Zyklus beziehen, in der Grafik mit derselben Farbe hinterlegt. Da der Stückzähler von der Maschine zeitlich nach jedem Zyklus inkrementiert wird und die meisten Werte bereits vor dem jeweiligen Stückzähler übermittelt wurden, bietet sich eine Zusammenfassung über den rückwärts aufgefüllten Stückzähler an. Wird der Stückzähler rückwärts aufgefüllt, so deckt er einen Großteil der auftretenden Werte ab (s. farbliche Rahmen in der Darstellung). Im Bezug auf das Auftreten der einzelnen Werte, existieren vier Arten von Prozessparametern. Jede Art von Prozessparameter ist beispielhaft in dem Diagramm mit einem Buchstaben (A bis D) dargestellt.

Prozessparameter A tritt einmalig je Zyklus auf. Beispiele sind Stückzähler und Zykluszeit, bei denen der Wert auf das neue Format direkt übertragen werden kann. Prozessparameter B tritt mehrfach je Zyklus auf, ein Beispiel ist die Vorlauftemperatur. Da an dieser Stelle keine Verlaufsparameter mehr behandelt werden müssen, kann angenommen werden, dass es sich bei den Prozessparametern vom Typ B um konstante Prozessparameter handelt. Für konstante Prozessparameter mit wenig Schwankung kann das arithmetische Mittel aus den im Zyklus auftretenden Werten als repräsentativer Wert verwendet werden. Prozessparameter C tritt nur einmalig zu Beginn auf, dies ist vor allem bei den Soll-Parametern der Fall, da diese nur bei einer Änderung und nach jedem Neustart der Maschine einen Wert liefern. Nach der Zuordnung zum passenden Zyklus ist ein

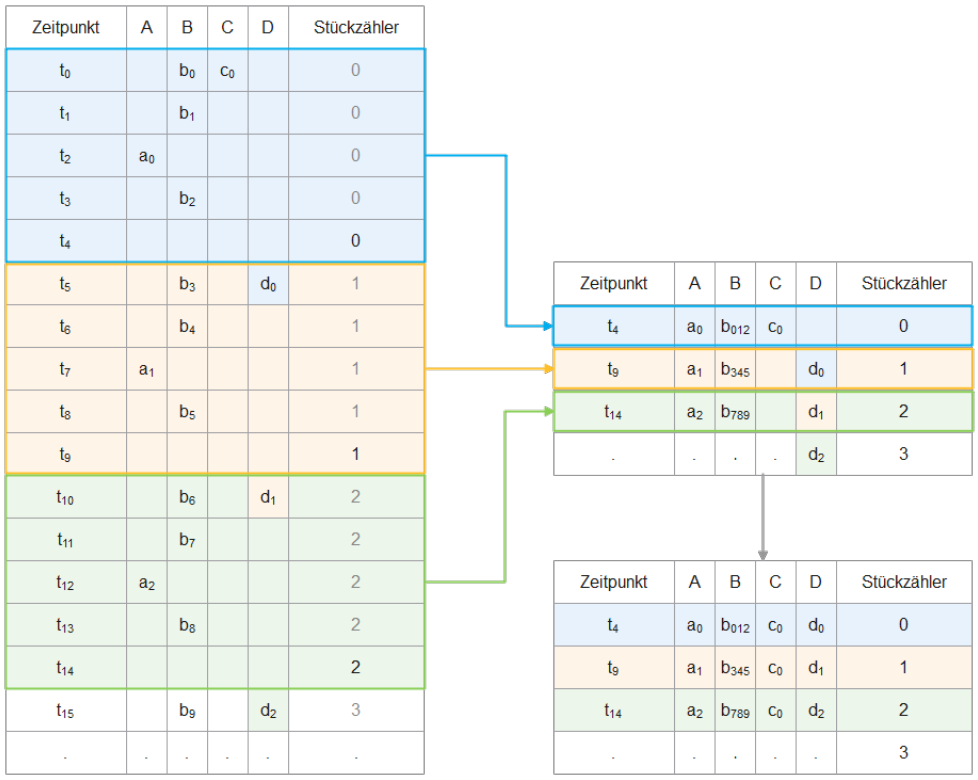


Abbildung 4.1: Vorgehen der zyklischen Zusammenfassung der Daten

fortlaufendes Auffüllen der Soll-Parameter sinnvoll, da auch im Falle dessen, dass keine neuen Werte geliefert werden, der letzte Wert im Prozess gültig ist. Prozessparameter D ist sehr ähnlich zu Prozessparameter A, mit dem Unterschied, dass dieser zeitlich nach dem Stückzähler übermittelt wird, jedoch noch zu dem vorherigen Zyklus gehört. Ein Beispiel ist das Formteilgewicht, welches durch den Roboterarm erst nach jedem Zyklus ermittelt wird. Ein Verschieben aller Werte nach der Zuordnung korrigiert diesen Versatz (letzter Schritt in der Darstellung).

Die Aggregation mit dem arithmetischen Mittel ist nur bei numerischen Werten sinnvoll, sodass eine Außnahmebehandlung bei nominalen Werten notwendig ist. Für alle Prozessparameter, deren arithmetisches Mittel nicht von der pandas Funktion berechnet werden kann, wird der häufigste auftretende Wert eines Zyklus verwendet. Nachdem alle Prozessparameter aus der Datenbank in dem neuen Format sind, kann die Zuordnung der zusätzlichen Informationen (Versuchsreihe, Versuchsnummer, Charge) aus der bereits erwähnten Excel-Datei stattfinden. Das Ergebnis ist ein unbereinigter Datensatz mit 29798 Zeilen und 34 Spalten, bei dem jede Datenzeile Informationen über einen bestimmten Zyklus enthält. Dieser Datensatz ist ein Ergebnis der Arbeit und wird für weitere Analysen als „001.csv“ gespeichert.

5 Explorative Analyse

Nachdem der Datensatz durch die Datentransformation im Abschnitt 4.3 in der Form von Eingang- zu Ausgangsgrößen gebracht wurde, wird dieser anschließend mittels explorativer Analyse untersucht.Die explorative Analyse hilft dem Data Scientist[5] Zusammenhänge und Muster zu erkennen und bereits vorhandene Vermutungen zu bestätigen oder zu widerlegen. Außerdem können Fehler im Datensatz durch eine explorative Analyse gefunden werden. Dabei existieren einige Methoden und Techniken, die bei einer explorativen Analyse verwendet werden können. [5]

5.1 Fehlende Daten

Eine einfache Ausgabe des Datensatzes zeigt, dass der Datensatz nach der zyklischen Zusammenfassung und dem Auffüllen der Soll-Parameter weiterhin leere Zellen enthält. Dabei wird nach der zyklischen Zusammenfassung angenommen, dass es sich hierbei um fehlende Daten im Datensatz handelt. Fehlende Daten sind in der Realität nicht unüblich und im DataFrame als „NaN“ (engl. „Not a Number“) gekennzeichnet. Das Ziel ist es einen Datensatz ohne fehlende Daten zu erzeugen bzw. die fehlenden Daten bestmöglich aufzufüllen, um einen möglichst aussagekräftigen Datensatz zu erhalten.

Um die Anzahl der fehlenden Daten anzeigen zu lassen, wird zunächst die Funktion `df.isnull()` verwendet. Hierbei wird der Datensatz in eine binäre Form mit Wahrheitswerten für fehlende Daten umgewandelt. Der Wahrheitswert `True` signalisiert dabei fehlende Daten, die in der Tabelle 5.1 mittels `df.isnull().sum()` gezählt wurden. Zur Übersichtlichkeit werden die Prozessparameter ohne fehlende Daten weggelassen.

Eine Strategie einen Datensatz ohne unvollständige Daten zu erhalten ist, alle Zeilen mit teilweise fehlende Daten (NaN) zu entfernen. Da in diesem Beispiel fast alle Zeilen teilweise fehlende Daten enthalten und nach der Löschung nur ein Bruchteil der Daten übrig bleiben würde, muss eine andere Strategie verfolgt werden. Beim Betrachten der Prozessparameter fällt zunächst auf, dass auch zwei Soll-Parameter aufgelistet sind, obwohl diese bereits in der Vorverarbeitung aufgefüllt wurden. Im Expertengespräch wurde erläutert, dass die Temperiergeräte mit den Vorlauftemperaturen (TG1, VL Temp Soll und TG2, VL Temp Soll) zu Beginn der Versuche ihre Daten nicht übermittelt haben, da die nötige Infrastruktur zu diesem Zeitpunkt noch nicht abschließend eingerichtet war. Ein Nachtragen der Soll-Werte für die Vorlauftemperaturen ist an dieser Stelle sinnvoll, da das Wissen vorhanden ist und den Datensatz ergänzt.

[5] „Die Aufgabe des Data Scientist umfasst in erster Linie die Lösung analytischer Fragestellungen in Bezug auf zumeist große und polystrukturierte Datenmengen (ver)mittels geeigneter Techniken und Technologien.“ [3, S. 7]

Tabelle 5.1: Anzahl fehlender Daten je Prozessparameter in 001.csv

SGM, Temp HZ 1 Ist	27413
TG1, VL Temp Soll	5776
TG1, VL Temp Ist	5776
SGM, Einspritzen Druck max.	301
TG2, VL Temp Ist	5218
SGM, Massepolster	448
TG2, RL Temp Ist	5218
TG1, RL Temp Ist	5776
SGM, Dosiervolumen	22005
SGM, Fs Ist	1340
TG2, VL Temp Soll	5218
SGM, Temp Einzug Ist	27047
SGM, Temp HZ 2 Ist	29035
SGM, Staudruck	1
SGM, Temp HZ 3 Ist	28472
Waage, Formteilgewicht	547
SGM, Zykluszeit	671
Versuchsreihe	10072
Versuchsnummer	10072

Die in der Auflistung auftretenden Ist-Parameter liefern ihre Daten nicht regelmäßig, sondern erst bei Änderungen des Prozesswertes. Bei Konsistenz der Prozessparameter werden somit keine neuen Daten übermittelt,[6] sodass für einen gesamten Zyklus kein Wert geliefert wird (NaN). Die fehlenden Daten können hierbei durch den vorherigen Wert realitätsnah aufgefüllt werden. Da dies nicht bei allen Prozessparametern der Fall ist, wurde im Rahmen eines Expertengespräches identifiziert, welche Parameter zur Auffüllung, unter Berücksichtigung der Rahmenbedingungen, geeignet sind. Prozessparameter, dessen fehlende Daten beispielsweise nicht von dem vorherigen Wert ableitbar sind, sind Formteilgewicht, Massepolster und Zykluszeit, da der vorherige Wert eine andere Ausgangssituation der Maschine haben könnte. Zusätzlich können alle Zeilen mit fehlenden Daten der Ausgangsparameter entfernt werden, da nur Zeilen mit Ausgangsgrößen für Analysen und Modelle verwendet werden sollen. Außerdem werden alle Zeilen ohne Versuchsnummer oder Versuchsreihe entfernt, da wie bereits im Abschnitt 4.1 beschrieben nur der stabile Prozess aufgrund der Vergleichbarkeit betrachtet wird.

Bei der Untersuchung der Zykluszeiten ist aufgefallen, dass es wenige Fälle gibt, bei denen eine Abweichung zwischen Zyklen innerhalb eines Versuches verzeichnet wurde. Dies kann durch eine Störung im Prozessverlauf auftreten, wenn beispielsweise Spritzgießer aufgrund von Verbrennungen am Formteil gezwungen ist, die Maschine anzuhalten. Nach einer Standzeit[7] liegt eine veränderte Ausgangssituation für die folgenden Zyklen vor, sodass die Zeilen der nachfolgenden Zyklen entfernt werden müssen. Die Anzahl

[6] Der letzte Wert ist noch aktuell.
[7] Zeit in der die Maschine nicht kontinuierlich produziert.

von 5 zu entfernenden Zyklen nach der Standzeit wurde vom Experten anhand des Schneckenvolumens[8] berechnet und festgelegt.

5.2 Zusammenhänge der Prozessparameter

Im nächsten Schritt der explorativen Analyse werden die Zusammenhänge zwischen den verschiedenen Prozessparametern ermittelt. Dies wird erreicht, indem zunächst eine lineare Korrelationsanalyse durchgeführt wird. Konkret wird hierbei der Korrelationskoeffizent nach Pearson[9] berechnet, welche die Standardmethode der angewandten Funktion `df.corr()` aus der Pandasbibliothek darstellt. Der Korrelationskoeffizent gibt den linearen Zusammenhang zwischen zwei Merkmalen, in diesem Fall zwischen den Prozessparametern, wieder. Der Korrelationskoeffizent bewegt sich hierbei im Intervall von -1 bis 1, wobei 0 keinen und 1 (und -1) einen absoluten (negativen) linearen Zusammenhang darstellt. [14, S. 346] Die Korrelationskoeffizienten der einzelnen Prozessparameterpaare lassen sich zur besseren Veranschaulichung als heatmap [10] (im Folgenden als Korrelationsmatrix bezeichnet) ausgeben, welche der Abbildung 5.1 zu entnehmen ist. In der Korrelationsmatrix sind zwischen bestimmten Gruppierungen von Prozesssparametern eindeutige absolute lineare Zusammenhänge zu erkennen (siehe gelbe Quadrate entlang der Hauptdiagonalen). Die sich dabei ergebenden drei Gruppierungen anhand dessen Korrelation lassen sich wie folgt auflisten:

- erste Gruppierung: TG1, VL Temp; TG2, VL Temp
- zweite Gruppierung: SGM, Nachdruck Druck Start; SGM, Nachdruck Druck Z1
- dritte Gruppierung: SGM, Temp HZ 1 Soll; SGM, Temp HZ 2 Soll; SGM, Temp HZ 3 Soll

Dabei wurden lediglich Soll-Parameter behandelt. Es kann durch die zweistellige Rundung in der Korrelationsmatix vorkommen, dass zwischen Soll- und Ist-Parametern eine Korrellation von 1 angezeigt wird. Diese ist in der Realität aber kleiner als 1. Die jeweiligen Prozessparameter wurden voneinander abhängig angepasst, wodurch sich eine Korrelation von 1 ergibt. Beispielsweise wurden die beiden Vorlauftemperaturen parallel angepasst, um eine möglichst gleichmäßige Werkzeugtemperatur zu erhalten. Da die Anpassung der Prozessparameter in gleicher Weise erfolgte, können die gruppierten Prozessparameter zu jeweils einem Prozessparameter zusammengefasst werden. Dieser Schritt ist auch unter dem Namen „Dimensionsreduktion“ bekannt. Die KI-Modelle würden von den zusätzlichen Prozessparametern mit zeilenweise gleichen Daten nicht profitieren, da bis auf eine mögliche veränderte Gewichtung kein Informationsgehalt aus den zusätzlichen Spalten hervorgeht. Aus diesem Grund können die überflüssigen „Dimensionen“ entfernt werden. Mehr zu den Modellen in Kapitel 7.
Neben den Gruppierungen ließen sich aus der Korrelationsmatrix weitere Zusammenhänge erkennen. In der ersten Analyse wurden unter anderem fehlende Zusammenhänge zum Formteilgewicht erkannt. Konkret wurde ein fehlender Zusammenhang durch die

[8] Volumen des sich um der Schnecke befindlichen Kunststoffes.
[9] Ausmaß der linearen Abhängigkeit. [14, S. 346]
[10] Eine farbliche Matrixdarstellung.

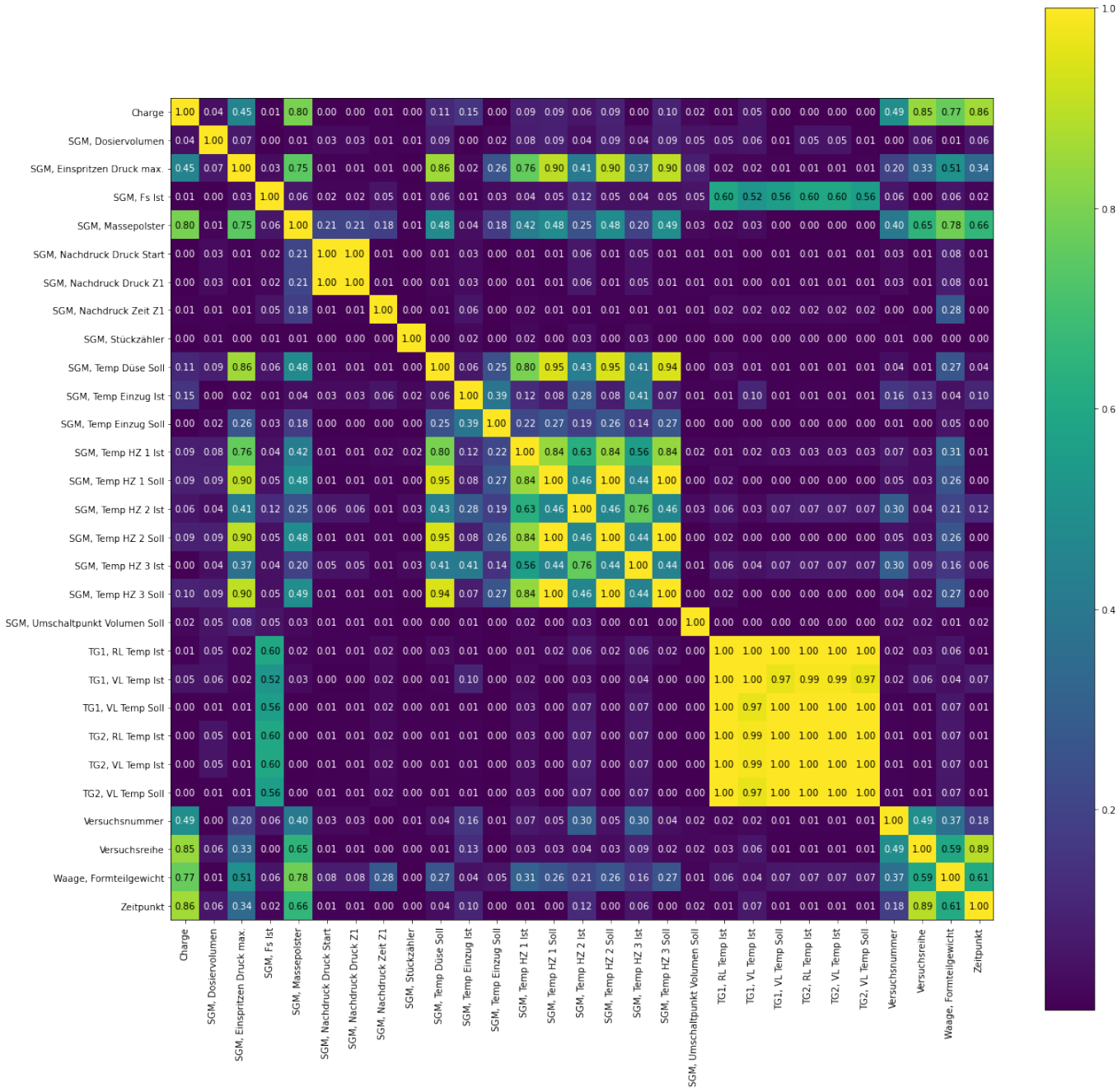


Abbildung 5.1: Korrelationsmatrix [14, S. 347]

hohe Korrelation zwischen Formteilgewicht und Zeitpunkt festgestellt. Der Zeitpunkt sollte jedoch keinen realen Einfluss auf das Formteilgewicht haben. Mit Hilfe von Expertenbefragungen und gegenüberstellenden Diagrammen (siehe Abbildung 5.2) konnte die

Charge als fehlende Einflussgröße identifiziert werden. Diese wurde in der Konsequenz zur weiteren Analyse ergänzt.

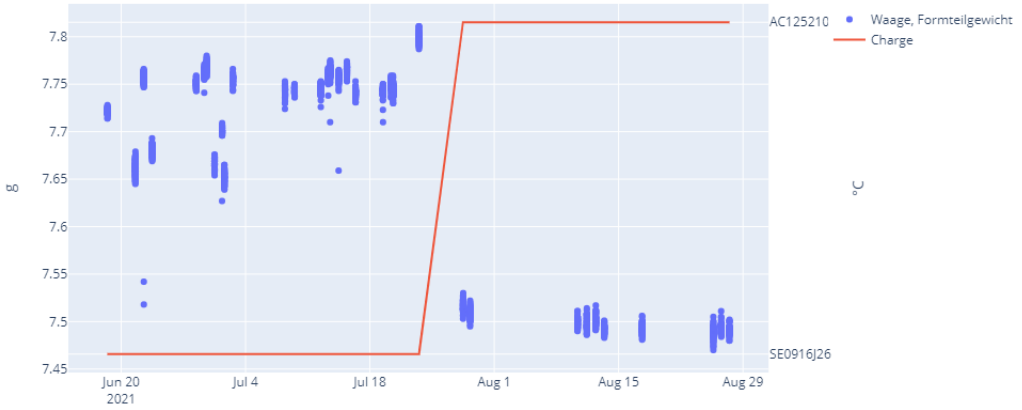


Abbildung 5.2: Streudiagramm Formteilgewicht Charge

Eine weitere Strategie, Zusammenhänge zwischen den Prozessparametern aufzudecken, ist die paarweise Darstellung der Prozessparameter mittels Streudiagrammen. Quelltext 5.1 zeigt beispielhaft eine Möglichkeit zur Darstellung mehrerer Streudiagramme mit `plotly.express`.

Quelltext 5.1: Darstellung einer Streudiagrammatrix in Python

```
import plotly.express as px
cols = ['Waage, Formteilgewicht', 'SGM, Nachdruck Zeit Z1',
        'SGM, Temp Düse Soll', 'TG1, VL Temp Soll']
fig = px.scatter_matrix(df,
                        dimensions=cols, color='Charge',
                        labels={col:col.split(' ', 1)[-1]
                                for col in df.columns}, # remove maschine
                        height = 1000, width = 1000)
fig.update_traces(diagonal_visible=True)
fig.show()
```

In der Abbildung 5.3 wurden beispielhaft die Prozessparameter verwendet, welche in Versuchsreihe 5 verändert wurden. Anhand der gleichmäßig verteilten Punkte zwischen den Soll-Parametern ist zu erkennen, dass die Anpassung der Parameter wie erwartet schematisch durchgeführt wurde. Durch die unterschiedliche Einfärbung nach den zwei verwendeten Chargen ist außerdem zu erkennen, dass die Charge einen vergleichsweise großen Einfluss auf das Formteilgewicht aufweist. Der Einfluss der Charge war vom Experten nicht in diesem Ausmaß erwartet, wodurch die Versuche der ersten Charge mit der neuen Charge zukünftig wiederholt werden müssen. Zusätzlich wird für die Charge

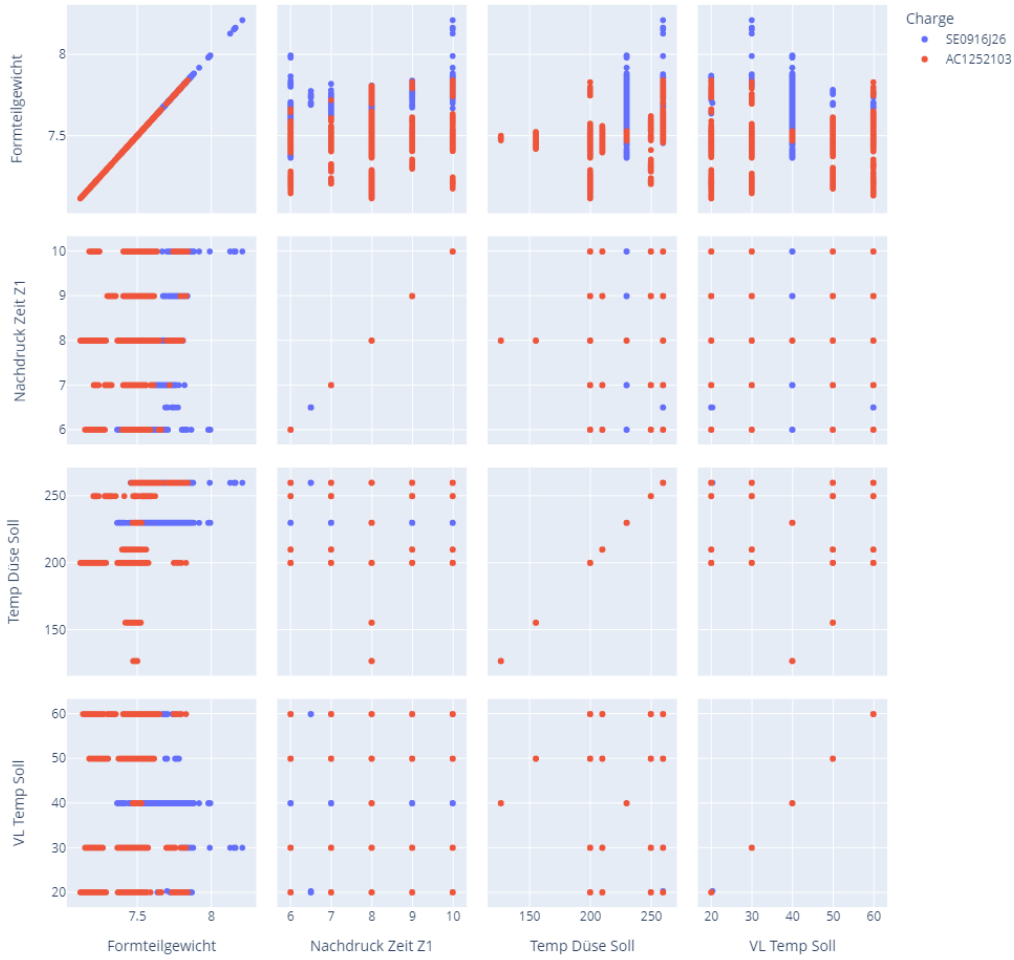


Abbildung 5.3: Streudiagrammatrix

künftig der Schmelzindex[11] verwendet, um aus den nominalen Werten numerische Werte zu erhalten, aus denen mehr Informationen gewonnen werden können.

Das Anzeigen von Prozessparametern in Bezug zur Zeit kann auch zu neuen Erkenntnissen führen, wie folgendes Beispiel zeigt. In Abbildung 5.4 wurden die Gewichte der Formteile aus Versuchsreihe 7, Versuchsnummer 48 dargestellt. Neben der horizontalen Punktwolke, sind vier Ausreißer von über 7,7g zu erkennen. Die Versuche mit solch einer Anomalie beim Formteilgewicht wurden bei der Versuchsdurchführung in der Excel-Tabelle für spätere Analysen markiert. Es ergaben sich bis zu dem Zeitpunkt dieser Arbeit eine Anzahl von 13 Fällen, bei denen ein Zusammenhang zur Vorlauftemperatur zu erkennen ist. Diese Ausreißer beeinflussen die Art Zusammenfassung der Formteilgewichte je Versuch und „lenken [..] vom wesentlichen (zu lernenden) Zusammenhang ab“[2,

[11] Im KUZ ermittelt. Numerischer Wert, der die Fließfähigkeit des Materials angibt.

S. 303]. Da unter Umständen, durch die große Lücke der Gewichte, das arithmetische Mittel einen unrealistischen Wert annehmen würde, ist der Median[12] für eine realistische Zusammenfassung die bessere Wahl.

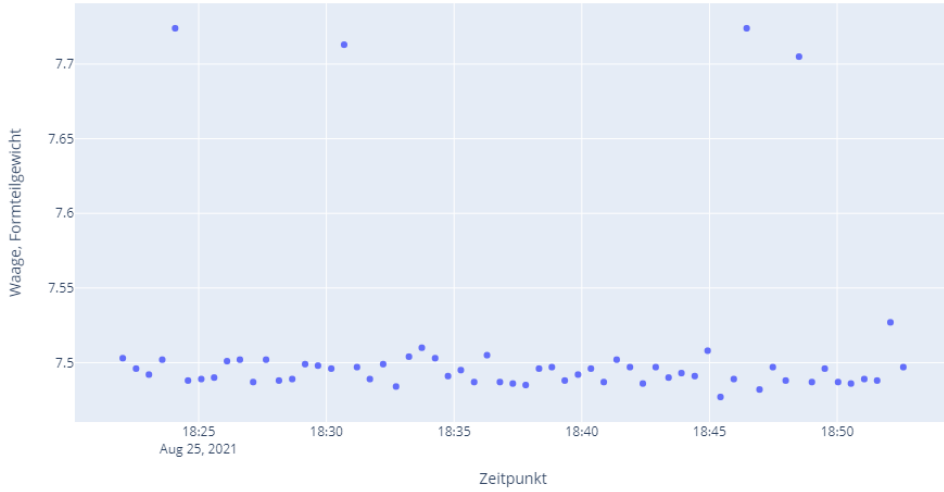


Abbildung 5.4: Gewichte der Formteile in Versuchsreihe 7, Versuchsnummer 48

5.3 Wiederholte Versuchsdurchführungen

Es existiert eine Ausgangseinstellung, die mehrfach an der Maschine an unterschiedlichen Zeitpunkten durchlaufen wurde, um längerfristige Schwankungen erkennen zu können. Mit Quelltext 5.2 wurde die Ausgangseinstellung durch Gruppierung über die Soll-Parameter identifiziert. Zunächst wurden die vorhandenen Soll-Parameter mittels Schnittmenge über alle im Forschungsprojekt verwendeten Soll-Parametern gebildet, damit der Algorithmus auch mit Veränderungen der Prozessparameter umgehen kann. Bei der Gruppierung wurden die dabei auftretenden Versuchsnummern pro Gruppierung gezählt. Zur Gruppierung wurden die Soll-Parameter in der zusätzlichen Spalte „Einstellung“ kombiniert. Alternativ kann eine Gruppierung über einen Multiindex oder mehrere Spalten erfolgen. Da bei dieser Methode alle Gruppen, die leere Zellen enthalten, ignoriert werden würden, wurde sich für die Variante mit der zusätzlichen Spalte „Einstellung“ entschieden. [13]

Quelltext 5.2: Python-Code zur Auflistung von Duplikaten der Soll-Parameter

```
g = set(df.columns) & set(kq.ki_set_param)
df = df.set_index(list(g))
df = df.groupby(level=[x for x in range(len(g))]).count()
```

[12] Jener Wert, der nach Sortierung zentral liegt.

```
df = df.sort_values(by=['Versuchsnummer'], ascending=False)
df = df.rename(columns={'Versuchsnummer':'Anzahl Zyklen'})
df['Anzahl'].to_frame().head(10)
```

Tabelle 5.2 beinhaltet eine Kombination der Soll-Parameter (Maschineneinstellung) und die Anzahl an gültigen Zyklen für die jeweilige Einstellung. Dabei wurden nur Zeilen mit mehr als 61 Zyklen angezeigt, was im Kontext gleichbedeutend mit mehreren durchgeführten Versuchen ist. Entgegen der Erwartung, dass nur eine Einstellung mehrfach

Tabelle 5.2: Anzahl Duplikate der Maschineneinstellungen

Maschineneinstellung	Anzahl Zyklen
210.0 | 230.0 | 172 | 25.0 | 220.0 | 400.0 | 8.0 | 25.0 | 400.0 | 40.0 | 70.0 | 40.0 | 230.0 | 80.0 | 2.5	2165
230.0 | 250.0 | 172 | 25.0 | 240.0 | 400.0 | 7 | 25.0 | 400.0 | 20.0 | 70.0 | 20.0 | 250.0 | 80.0 | 2.5	122
230.0 | 250.0 | 172 | 25.0 | 240.0 | 400.0 | 7 | 25.0 | 400.0 | 50.0 | 70.0 | 50.0 | 250.0 | 80.0 | 2.5	121
240.0 | 260.0 | 172 | 25.0 | 250.0 | 400.0 | 10.0 | 25.0 | 400.0 | 60.0 | 70.0 | 60.0 | 260.0 | 80.0 | 2.5	119
190.0 | 210.0 | 172 | 25.0 | 200.0 | 400.0 | 6 | 25.0 | 400.0 | 20.0 | 70.0 | 20.0 | 210.0 | 80.0 | 2.5	115
210.0 | 230.0 | 172 | 25.0 | 220.0 | 390.0 | 9.0 | 25.0 | 390.0 | 40.0 | 70.0 | 40.0 | 230.0 | 80.0 | 2.3	114

durchlaufen wurde, sind in der Tabelle mehrere Einstellungen zu erkennen. Es wird angenommen, dass die Einstellung mit der höchsten Anzahl die Ausgangsparametereinstellung darstellt. Neben der Ausgangsparametereinstellung existieren fünf weitere Maschineneinstellungen, die über mehrere Versuche durchgeführt worden sind. Hierfür gibt es zwei mögliche Erklärungen. Entweder die Daten wurden nicht übermittelt oder gespeichert, oder dem Spritzgießer ist bei der Anpassung der Soll-Parameter ein Fehler unterlaufen. Im zweiten Fall wären die Versuche rein vom Informationsgehalt gültig und könnten weiter verwendet werden. Da es im ersten Fall durch das Auffüllen der Soll-Parameter zu falschen Informationen im Datensatz kommt, ist eine Löschung aller doppelt durchgeführten Versuche die sichere Variante.

6 Datenvorverarbeitung und -struktur

Bevor im nächsten Kapitel die Methode zur Prognose des Formteilgewichts erläutert wird, werden zunächst die sich aus Kapitel 4 und Kapitel 5 ergebenden notwendigen Schritte zur Vorverarbeitung der Daten zusammengetragen und dessen Strukturen erläutert.

Da die verwendeten Prozessparameter unterschiedlich behandelt werden müssen, wurde zusätzliches Prozesswissen in Form von Listen in einer Python Datei namens „`kiquality.py`“ abgelegt. Dabei enthalten die Listen die Namen bestimmter Prozessparameter einer Kategorie. Ein Beispiel ist die Liste, die alle Ausgangsparameter (Formteilgewicht, Zykluszeit) enthält. Diese Listen sollen die Anpassbarkeit der Vorverarbeitung an zentraler Stelle ermöglichen. Die Beziehungen der Dateien und der Informationsfluss kann der nachfolgenden Abbildung 6.1 entnommen werden.

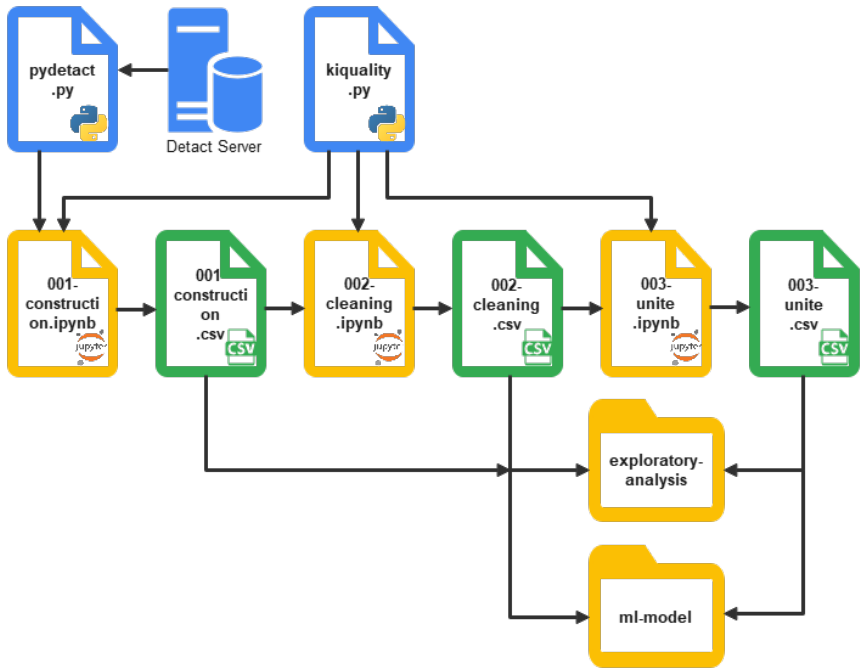


Abbildung 6.1: Dateibeziehungen

Die CSV-Dateien sind dabei Ergebnisse der Vorverarbeitungen aus den Jupyter-Notebooks. Die Vorverabeitung des Datensatzes findet wie in der Abbildung zu erkennen ist sequentiell statt. In Tabelle 6.1 sind die Verarbeitungsschritte und die sich daraus ergebene Datenreduzierung aufgelistet.

Tabelle 6.1: Datenvorverabeitung

Schritt	Beschreibung	Datenreduzierung	Implementierung
zyklische Zusammenfassung	Zuordnung aller Daten auf den jeweiligen Zyklus und Auffüllen der Soll-Parameter	1.971.868 → 29798	001-construction.ipynb
Datenbereinigung	Individuelles Auffüllen aller Prozessparameter, löschen ungültiger Zyklen und Zyklen ohne Ausgangsgrößen	29798 → 19091	002-cleaning.ipynb
versuchsweise Zusammenfassung	individuelle Zusammenfassung der Prozessparameter je Versuch und löschen doppelter Versuche	19091 → 237	003-unite.ipynb

6.1 Versuchsweise Zusammenfassung

Der letzte in der explorativen Analyse verwendete Datensatz (`002.csv`) enthält ausschließlich gültige Versuche mit jeweils vorhandenen Zielgrößen. Für jeden Versuch existieren in dem Datensatz bis zu[13] 60 Zeilen, die im nächsten Schritt zusammengefasst werden, um eine bestmögliche Repräsentation dieser Maschineneinstellung mit den dazugehörigen Ausgangsgrößen zu liefern. In den meisten Fällen ist das arithmetische Mittel für eine Aggregierung sinnvoll, dennoch ergab die explorative Analyse, dass für das Formteilgewicht der Median eine bessere Aggregierung darstellt.Zur Aggregierung wurden die Funktionen mean, median, first, mode und std aus der pandas Bibliothek für die Prozessparameter verwendet. Bis auf die Funktion mean existiert für jeden Aggregationstyp eine Liste mit den Namen der passenden Prozessparameter. Die Standardabweichung ist an dieser Stelle eine Art Merkmalsextraktion und wird als zusätzliche Spalte dem Datensatz übergeben. Beispielsweise wird die Standardabweichung für das Formteilgewicht innerhalb eines jeden Versuchs ermittelt. Nach der Aggregation über die Versuche existieren in Hinsicht auf die Soll-Parameter zeilenweise Duplikate. Diese Duplikate kommen durch die Ausgangseinstellung der Maschine zustande. Diese Einstellung wurde im gesamten Versuchsplan mehrfach an regelmäßigen Zeitpunkten durchlaufen, um mögliche Veränderungen im Prozess bzw. der Maschine aufzudecken.

Da die einzelnen Versuche gleich behandelt werden und keine Gewichtung mit Hinsicht der KI-Modelle stattfinden soll, wurden alle Duplikate aus dem Datensatz entfernt.
Eine Zeile in dem neuen Datensatz besteht nun aus einer eindeutigen Kombination von

[13] Durch das Bereinigen können Zeilen entfernt worden sein.

Soll-Parametern und dessen repräsentativen Ist- sowie Ausgangswerte. Die Ergebnistabelle mit 237 Zeilen wird als „`003.csv`“ gespeichert und wird im nächsten Kapitel zum erstellen der KI-Modelle verwendet.

7 Modellerstellung und -evaluation

In diesem Kapitel wird eine Methode vorgestellt, um die besten Modelle für die Vorhersage des Formteilgewichtes zu identifizieren. Als Modell ist hierbei eine Sequenz von Datentransformationen bis hin zum Schätzer, der das Formteilgewicht prognostiziert, definiert.

7.1 Konzept des Evaluationsprozesses

Um die Modelle untereinander vergleichen zu können, werden deren Performances auf unbekannte Daten ermittelt. Die Basis der Evaluation ist das Bestimmtheitsmaß (R^2). Zur besseren Einordnung und Bewertung des Bestimmtheitsmaßes, lassen sich die folgenden Aussagen treffen. Bei einem Modell, welches immer einen konstanten Mittelwert liefert, ist ein Bestimmtheitsmaß von 0.0 zu erwarten. Ein Bestimmtheitsmaß von 1 ist wiederum die bestmögliche Wertung. Ein negatives Bewertungsmaß kann bei willkürlichen (schlechten) Modellen auftreten.[12]

Durch die Abbildung des Bestimmtheitsmaßes ist der Zusammenhang zur Ausgangsgröße, in diesem Fall das Formteilgewicht, nicht mehr gegeben. Um einen Zusammenhang der Bewertung auf die Ausgangsgröße zu liefern, wurde die Wurzel der quadratischen Abweichung für die finalen Modelle zusätzlich berechnet.
Die für die Modelle unbekannten Daten, die zur Evaluierung benötigt werden, werden zunächst vom Datensatz getrennt. Es ergeben sich dadurch zwei Datensätze aus 85% Trainingsdaten und 15% Testdaten. Das Aufteilen und somit Zurückhalten der Daten für die Evaluierung wird auch als „Holdout-Methode“ bezeichnet.[14, S. 218] Da der ursprüngliche Datensatz zeitlich sortiert ist, sollte der Datensatz vor der Aufteilung für eine bestmögliche Bewertung gemischt werden. Ohne ein Mischen des Datensatzes würde ggf. nur anhand der Daten aus der letzten (oder ersten) Versuchsreihe getestet werden, und somit würde auch die Evaluierung lediglich eine Aussage über diesen Wissensbereich tätigen. Das Mischen und Aufteilen der Daten wird von der Funktion `train_test_split()` übernommen.

Mit den Trainingsdaten aus dem Datensatz wird für jedes Modell eine zehnfache Kreuzvalidierung durchgeführt. Bei der Kreuzvalidierung werden die Daten in gleichgroße Teile aufgeteilt, wobei ein Teil zum Testen und die restlichen zum Anlernen verwendet werden. Dabei wird jeder Teil iterativ zum Testen verwendet und der Mittelwert der Validierungsgrößen berechnet.[14, S. 219-220]

In der Abbildung 7.1 wird zur Übersichtlichkeit die analoge fünffache Kreuzvalidierung innerhalb der Holdout-Methode dargestellt. Dabei ist ein „Fold“ jeweils ein Teil der Trai-

ningsdaten. Für jeden „Split“ wird die Validierungsgröße berechnet, die im Anschluss einen Mittelwert bilden. Die Kreuzvalidierung wird hierbei für jedes Modell durchgeführt

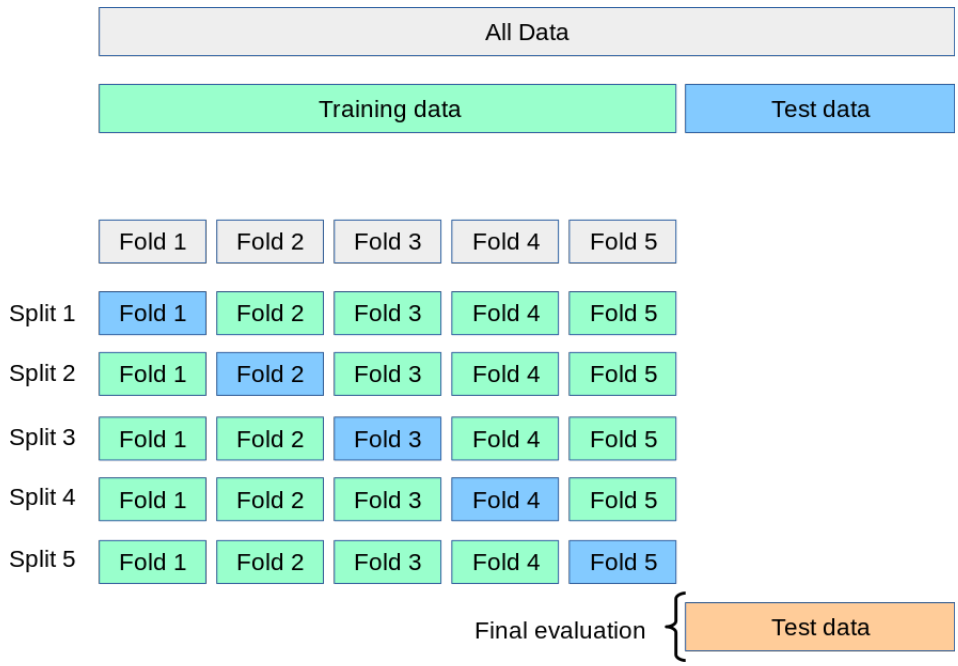


Abbildung 7.1: fünffache Kreuzvalidierung [11]

und die besten Modelle anhand des Bestimmunsgsmaßes R^2 ausgewählt.
Für das Bearbeiten des Datensatzes und das Anlernen des Schätzers sowie der Hyperparameteroptimierung wird die Pipeline Klasse aus ski-kit learn verwendet.
Eine Pipeline hat folgende Schritte:

1. Skalierung („slc“)
   Eine Skalierung ist für einige Schätzer sinnvoll, um keine ungewollten Gewichtungen durch unterschiedliche Größen der Merkmale zu erhalten. In diesem Fall werden die Skalierungs-Funktionen `MinMaxScaler()` oder `StandardScaler()` verwendet. Neben den Skalierungs-Funktionen wird zusätzlich die Performance der Modelle auf unskalierte Datensätze betrachtet.

2. Dimensionsreduktion („reduce_dim“)
   In der Vorverarbeitung hat bereits eine Dimensionsreduktion stattgefunden. Hier wird mit der Funktion `PCA()` geprüft, ob möglicherweise Modelle existieren, die mit einer weiteren automatischen Dimensionsreduktion bessere Ergebnisse liefern. Die Dimensionsreduktion reduziert die Anzahl der Merkmale (Prozessparameter). Die Modelle werden auch ohne Dimensionsreduktion behandelt.

3. Schätzer („reg“)
   Der Schätzer ist der Kern der Pipeline und enthält die eigentlichen Algorithmen zur Prognose. Jeder Schätzer wird hierbei als eine „Blackbox“[14] mit ggf. Hyperparame-

[14] Ab Eingabe der Daten bis hin zur Ausgabe ist die Vorgehensweise unbekannt.

tern betrachtet. Die „Blackbox“ bekommt zum Anlernen einen Datensatz und kann im Anschluss auf beliebige Eingabedaten eine Prognose für das Formteilgewicht liefern.

Die bereits erwähnten Hyperparameter eines Schätzers können einen großen Einfluss auf die Performance des Modells haben, da sie die Funktionsweise der Algorithmen beeinflussen. Um eine Optimierung der Hyperparameter durchzuführen, kann beispielsweise die `GridSearchCV` Klasse aus sci-kit learn verwendet werden. Dabei wird der Klasse eine Pipeline, ein Dicitionary der zu durchsuchenden Hyperparameter, eine Kreuzvalidierungsfunkton und eine Bewertungsfunktion übergeben. Aus der Klasse kann nach dem Anlernen (mittels `fit()`) über den Klassenparameter `best_params_` die Hyperparameter mit der besten Bewertung abgefragt werden. In diesem Fall wurde, um die Laufzeit der Suche zu verringern, für die Hyperparameteroptimierung mit `GridSearchCV` eine separate dreifache anstatt zehnfache Kreuzvalidierung durchgeführt.

Alle Modelle werden über sogenannte „List Comprehensions“[15] in Python zusammengesetzt. Jedes Modell wird anschließend angelernt und auf dem oben beschriebenen Weg validiert. Die sich daraus ergebene Tabelle gibt einen Überblick über die Modelle und deren Performances. Tabelle 7.1 ist ein Ausschitt der Performance-Tabelle. Die vollständige Tabelle kann der Datei `performance_tabelle.csv` aus dem Anhang entnommen werden.

Tabelle 7.1: 10 beste Modelle der Performance-Tabelle der Kreuzvalidierung `2021-10-09_performancetabelle.csv`

cv r2	cv r2 std	estimator	reduce dim	scaler
0.773	0.113	RandomForestRegressor		StandardScaler()
0.772	0.113	RandomForestRegressor		
0.772	0.113	RandomForestRegressor		MinMaxScaler()
0.76	0.098	Ridge	PCA(18)	
0.76	0.098	Ridge		
0.74	0.125	MLPRegressor	PCA(18)	MinMaxScaler()
0.732	0.136	GradientBoostingRegressor		StandardScaler()
0.732	0.145	MLPRegressor	PCA(10)	MinMaxScaler()
0.73	0.14	GradientBoostingRegressor		MinMaxScaler()
0.727	0.138	MLPRegressor		MinMaxScaler()

7.2 Diskussion der Modelle

Die Performance-Tabelle (Ausschitt: Tabelle 7.1) enthält neben Modellbeschreibung und Modellbewertung, auch die Standardabweichung („cv r2 std“) der in der Kreuzvalidierung berechneten Bewertungen. Anhand einer hohen Standardabweichung ist anzunehmen, dass das Modell nicht die richtigen Zusammenhänge erkennt, da es nicht auf allen Teilen der Daten ein gleichmäßiges Ergebnis liefert. Ein weiteres Indiz für eine Überanpassung

[15] Kurzer, übersichtlicher Syntax für die Erstellung von Listen.

der Modelle ist, eine schlechtere Performance auf größerer Trainingsmenge. Bei einer größeren Trainingsmenge sind für das Modell mehr Informationen enthalten, somit kann eine bessere Performance erwartet werden.

Anhand der Performance-Tabelle lässt sich erkennen welche Vorverarbeitungen auf dem Datensatz eine vermutlich bessere Wahl ist. Beispielsweise verwenden fünf aus den 10 besten Modellen den MinMaxScaler zur Skalierung der Daten und lediglich zwei Modelle den StandardScaler. Somit lässt sich vermuten, dass der MinMaxScaler generell bessere Ergebnisse bei den Modellen liefert. Um zu entscheiden, welche Schätzer bei den Modellen vielversprechend sind, kann ein Violinendiagramm wie in Abbildung 7.2 helfen.

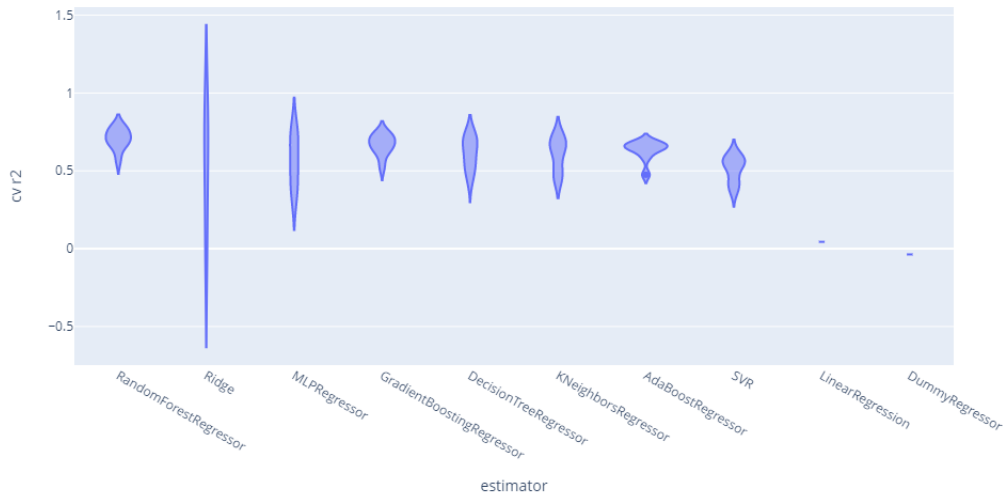


Abbildung 7.2: Schätzer und dessen Performance als Violinendiagramm

Im Violinendiagramm kann sich eine Häufung der Modelle erkennen lassen, was auf die Grenzen des jeweiligen Schätzers hinweisen könnte. Die drei besten Modelle mit unterschiedlichen Schätzern werden zur finalen Bewertung mit den gesamten Trainingsdaten (anstelle von 9/10 der Trainingsdaten bei der Kreuzvalidierung) angelernt und anhand der Testdaten validiert. Durch die größere Trainingsdatenmenge kann eine Verbesserung der Performance erwartet werden. Für die besten Modelle wurde, wie bereits erwähnt, neben R^2 zusätzlich die Wurzel der quadratischen Abweichung ausgegeben. Die laut Performance-Tabelle besten Schätzer sind RandomForestRegressor, Ridge und MLPRegressor.

Tabelle 7.2: Ausschnitt der `2021-10-09_finale_modelle.csv`

cv r2	r2 neu	rmse	estimator	reduce dim	scalar
0.773	0.907	0.046	RandomForestRegressor		StandardScaler()
0.76	0.928	0.04	Ridge	PCA(18)	
0.74	0.888	0.05	MLPRegressor	PCA(18)	MinMaxScaler()

In Tabelle 7.2 ist eine Performanceverbesserung der drei Modelle zu erkennen, was auf die größeren Trainingsdaten zurückzufühen ist. Außerdem lässt sich erkennen, dass eine Funktion zur Dimensionsreduktion bei zwei Modellen verwendet wurde. Die Dimensionsreduktion bei den besten modellen lässt darauf schließen, dass nicht die besten Merkmale zur Bestimmung des Formteilgewichtes aus dem Datensatz erhoben wurde. Da im Forschungsprojekt „KIQuality “ zukünftig die Benutzung weiterer Prozessparameter (Verlaufsparameter) geplant ist und somit die Merkmale für die Modelle neu betrachtet werden, wird die Dimensionsreduktion in dieser Arbeit nicht weiter verfolgt.

7.3 Ausblick

Im Folgenden wird das fertige und mit 100% der Daten angelernte Modell als „KI“ bezeichnet. Eine Integration der KI an einer produzierenden Maschine könnte dem Spritzgießer zum Beispiel in Form eines Ampelsystems. Hinweise auf ein ausreichendes Formteilgewicht geben. Das Ampelsystem könnte folgende Aussagen treffen:

Grün: Das Modell prognostiziert ein ausreichendes Formteilgewicht und die aktuellen Eingabedaten liegen im bzw. nah am Wissensbereich, der zum Anlernen der KI verwendet wurde.

Gelb: Das Modell prognostiziert ein ausreichendes Formteilgewicht jedoch sind die aktuellen Eingabedaten fern von dem Wissensbereich, der zum Anlernen der KI verwendet wurde.

Rot: Das Modell prognostiziert ein nicht ausreichendes Formteilgewicht.

Anhand der von der KI prognostizierten Formteilgewichte könnte eine automatisierte Aussortierung der schlechten Formteile geschehen. Wenn sich die KI in der Praxis als sehr zuverlässig herausstellt, könnte eine automatische Anpassung der Maschineneinstellung anhand der von der KI getroffenen Prognosen vorgenommen werden. Zusätzliche Expertenregeln für die Anpassung der Maschineneinstellungen könnten bei dieser Anwendung hilfreich sein. Im Forschungsprojekt „KIQuality“ soll im nächsten Schritt die Methodik weiterentwickelt werden. Dabei sollen beispielsweise die Verlaufsdaten aus den Prozessparametern extrahiert werden, um eine noch bessere Genauigkeit der Prognose zu erhalten. Des Weiteren sollen weitere Zielgrößen behandelt werden, um auch Formteilfehler prognostizieren zu können, die nicht ausschließlich am Gewicht zu erkennen sind.

8 Abschließende Betrachtung

Im Rahmen dieser Arbeit wurde eine Methodik entwickelt das Formteilgewicht anhand von gesammelten Messdaten einer Spritzgießmaschine zu prognostizieren.
Ein wesentlicher Bestandteil ist dabei die Vorverarbeitung der Daten. Bei der Analyse wurden neue Erkenntnisse und Zusammenhänge der Prozessparameter dargelegt und der Datensatz hinreichend ergänzt. Dabei wurden unter anderem mehrere fehlende Daten unterschiedlich behandelt um einen Datensatz zu erhalten der die Realität bestmöglich widerspiegelt.

Alle Vorverarbeitungen wurden in dokumentierten JupyterLab-Notebooks programmiert, welche den jeweiligen verarbeiteten Datensatz als CSV-Datei ablegt. Anhand des vorverarbeiteten Datensatzes wurde das Konzept zum Evaluationsprozesses der KI-Modelle erläutert und anschließend durchgeführt. Hierbei ergaben sich Hinweise auf den für den Datensatz geeigneten Schätzer, sowie einer geeigneten Skalierungsfunktion.

Literaturverzeichnis

[1] F. Beitl. *1000 Tipps zum Spritzgießen Band 11 Qualitätskontrolle*. Beuth Verlag, 2017. ISBN: 9783410266358.

[2] I. Boersch, J. Heinsohn und R. Socher. *Wissensverarbeitung*. Spektrum, 2007. ISBN: 9873827418449.

[3] Uwe Haneke, Stephan Trahasch, Michael Zimmer und C Felden. „Data Science". In: (2018). URL: http://sigs.de/tdwi/Wissen/eBooks/TDWI_E_Book_Data_Science.pdf.

[4] C. Hopmann und W. Michaeli. *Einführung in die Kunststoffverarbeitung*. Carl Hanser Verlag GmbH & Company KG, 2017. ISBN: 9783446453562. URL: https://books.google.de/books?id=kr7ADgAAQBAJ.

[5] IBM. *What is Exploratory Data Analysis?* 2021. URL: https://www.ibm.com/cloud/learn/exploratory-data-analysis. zuletzt besucht: 01.10.2021.

[6] Christoph Jaroschek. *Spritzgießen für Praktiker*. Carl Hanser Verlag GmbH Co KG, 2013.

[7] S. Koll. *Deutliche Umsatzsteigerung für Sumitomo (SHI) Demag*. URL: https://www.k-zeitung.de/deutliche-umsatzsteigerung-fuer-sumitomo-shi-demag/. zuletzt besucht: 11.10.2021.

[8] Kunststoffzentrum in Leipzig gGmbH. „Jahresbericht 2018". 2018.

[9] Kunststoffzentrum in Leipzig gGmbH. „Jahresbericht 2020". 2020.

[10] M. Mayer, E. Gandert und S. Fischer. *60 Jahre Kunststoff-Zentrum in Leipzig*. 2020. URL: https://www.plastverarbeiter.de/100787/60-jahre-kunststoff-zentrum-in-leipzig/. zuletzt besucht: 30.09.2021.

[11] PyData Development Team. *3.1. Cross-validation: evaluating estimator performance*. URL: https://scikit-learn.org/stable/modules/cross_validation.html. zuletzt besucht: 11.10.2021.

[12] PyData Development Team. *sklearn.metrics.r2_score*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html. zuletzt besucht: 09.10.2021.

[13] PyData Development Team. *Working with missing data — pandas 1.3.3 documentation*. URL: https://pandas.pydata.org/pandas-docs/stable/user_guide/missing_data.html#na-values-in-groupby. zuletzt besucht: 09.10.2021.

[14] Sebastian Raschka und Vahid Mirjalili. *Machine Learning mit Python und Keras, TensorFlow 2 und Scikit-learn*. mitp, 2021. ISBN: 978-3-95845-700-3.

[15] Karl Siebertz, David van Bebber und Thomas Hochkirchen. *Statistische Versuchsplanung*. Springer, 2017.

[16] Thomas Theis. *Einstieg in Python*. Galileo Press Bonn, 2011.

[17] Jake VanderPlas. *Python data science handbook: Essential tools for working with data*. O'Reilly Media, Inc., 2016.

A Anhang

Jahresbericht 2018 KUZ, S.7-8

Mit den Mitgliedern unserer Fördergemeinschaft verbinden uns in besonderer Weise wirtschaftliche und wissenschaftliche Kooperationen und Kontakte. Diese zum gegenseitigen Vorteil auszubauen und zu entwickeln, ist ein wichtiges Anliegen unserer Arbeit.

In der Forschungsarbeit sind Mitglieder der Fördergemeinschaft unmittelbar in Projekten oder projektbegleitenden Ausschüssen beteiligt. Im Bereich Weiterbildung nehmen sie aktiven Anteil, indem sie versierte Fachleute als Vortragende entsenden oder eigene Veranstaltungen im KUZ durchführen. Ebenso vielfältig sind die technischen, Prüf- und Beratungsleistungen, die wir für Mitglieder der Fördergemeinschaft erbringen. Umgekehrt nutzen wir Leistungen der Unternehmen für die Bearbeitung unserer Aufgaben.

Region Mitteldeutschland ist Kunststoffregion

Zur weiteren Stärkung unserer Region Mitteldeutschland als Kunststoffregion tragen wir durch unser breites Arbeitsspektrum und durch unsere intensive Netzwerkarbeit aktiv bei. So realisieren wir auch einen großen Teil unserer Aktivitäten in FuE, Dienstleistungen und der Weiterbildung mit regionalen Unternehmen. Dabei spielen die Mitglieder des AMZK eine besondere Rolle.

Das KUZ in Deutschland und Europa

Stetig zugenommen haben unsere überregionalen wirtschaftlichen und wissenschaftlichen Aktivitäten mit Unternehmen und Forschungseinrichtungen. Auch hier haben wir uns einen guten Namen erarbeitet und konnten neue Kunden und Partner gewinnen.

Zu den renommierten Seminaren für Polyurethantechnik kommen Teilnehmer von Unternehmen europaweit, so z. B. aus Österreich, der Schweiz, aus den Beneluxstaaten und aus Osteuropa (hier insbesondere aus Polen und Tschechien). Internationale Beteiligung können wir auch bei den Fachkonferenzen verzeichnen.

Das KUZ ist Mitglied im weltweit auf dem Gebiet der Mikrotechnik tätigen Netzwerk IVAM. Auch in der Mikrokunststofftechnik kommen Kunden aus dem Ausland, insbesondere aus dem europäischen.

Das KUZ unterstützt die Wirtschaftsförderung Sachsen GmbH und die Stadt Leipzig mit der Invest Region Leipzig GmbH bei Ansiedlungsprojekten in der Region sowie bei wirtschaftlichen Angeboten an große ausländische Investoren. Auf diese Weise konnten Kontakte und Aufträge mit ausländischen Unternehmen realisiert werden.

Unsere zielgerichtete Strategie

Bauteile und Baugruppen aus und mit Kunststoff erfüllen konkrete Aufgaben bei ihrem Einsatz und erlangen dadurch ihren eigentlichen funktionalen Zweck. Das Produkt - das Kunststoffteil - ist entscheidend, denn damit verdient unsere Branche Geld.

Kunststoffteile sind in fast allen Produkten enthalten und häufig bestimmend für deren Funktion. Die Kunststoff verarbeitende Industrie ist eine Zulieferindustrie für alle Branchen. Sie ermöglicht die Herstellung vielfältiger Produkte für die unterschiedlichsten Bedürfnisse und verkörpert somit technisch/technologischen Fortschritt. Ohne die innovative Kunststoffindustrie mit ihren substanziellen Zulieferungen kann Deutschland nicht Export(vize)weltmeister sein.

Folgerichtig beschäftigen wir uns im Rahmen unserer FuE-Arbeit mit dem „Prozess zur Herstellung von Formteilen und Baugruppen aus/mit Kunststoff". Unser Knowhow in unseren technologischen Kernkompetenzen

- Spritzgießen (speziell auch das Mikrospritzgießen mit allen Sonderverfahren),

- Polyurethan-Reaktionstechnik (auch in Kombination mit dem Thermoplastspritzgießen) und
- das Schweißen/Nieten/Bördeln

bauen wir kontinuierlich aus.

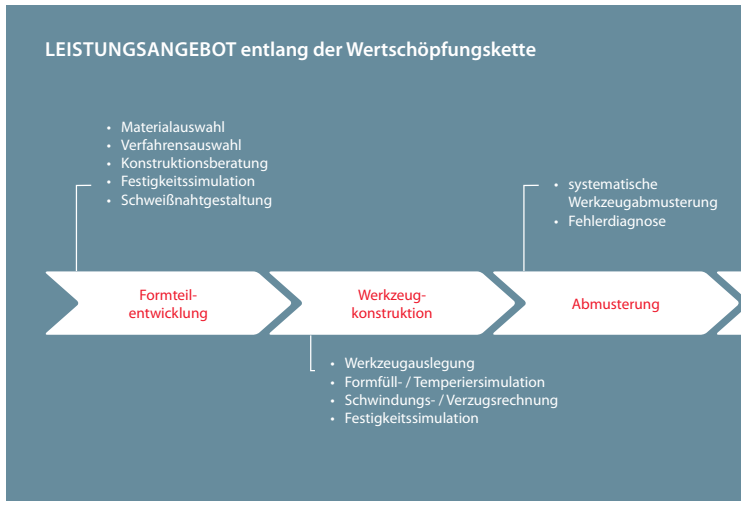
Mit diesem Technologieportfolio haben wir bei außeruniversitären FuE-Einrichtungen auf dem Gebiet der Kunststofftechnik ein Alleinstellungsmerkmal und eine hohe Zukunftsfähigkeit. Technologien sind besonders wichtig, sonst erhält man kein Formteil.

Kunststofftechnische Fragestellungen werden zunehmend komplexer. Dementsprechend haben wir eine Systemkompetenz entlang der Wertschöpfungskette aufgebaut, um unseren Industriepartnern kompetente Unterstützung geben zu können. Diese reicht vom Kunststoff selbst und seinen Eigenschaften sowie dem Compoundieren zum Maßschneidern von Kunststoffen, über Werkzeug- und Formteilkonstruktion, die Verarbeitungs- und Verbindungstechnik bis hin zur Kunststoffprüfung. Kontinuierlich arbeiten wir daran, bestehende Lücken zu schließen.

Kunststoffteile betrachten wir mit den Augen des Herstellers. So sind wir gefragter Ansprechpartner, wenn es um die Erfüllung hoher Qualitätsansprüche im Hinblick auf, Oberflächenqualität sowie Langlebigkeit, Präzision, Leichtbau, Funktionsintegration, intelligente Konstruktionen, effiziente Herstellprozesse bezüglich Energie, Material und Kosten sowie Ausschussreduktion bzw. -vermeidung geht.

Innovative Industrie-Dienstleistungen

Der Wettbewerbsvorteil der Unternehmen ist unser Anliegen. Gemeinsam mit und für die Unternehmen arbeiten wir an optimalen Lösungen für ihre Aufgabenstellungen. Das beinhaltet Beratungen zu Technologie- und Werkstoffeinsatz sowie zur Technologieoptimierung bis hin zu



Jahresbericht 2020 KUZ, S.7

Technologie- und **Innovationsforschung**

Innovationen als Treiber von Wachstum

Durch die exponentielle Entwicklungsgeschwindigkeit neuer Technologien erleben wir auch in den Forschungs- und Entwicklungsfeldern der Kunststoffbranche eine schnelllebige Transformation. Mithilfe von interdisziplinären Forschungsprojekten sowie einem hohen Maß an Flexibilität und Kreativität gelingt es dem KUZ, neue F&E-Trends innerhalb der Prozess-/Produktentwicklung und Technologieoptimierung zu formulieren und somit technologische Innovationen zu entwickeln. Verarbeitungsprozesse und das Wissen um komplexe Wirkungszusammenhängen werden entlang der Wertschöpfungskette – vom Werkstoff über die Produktentwicklung, die Produktion bis hin zur Wiederverwertung – erforscht und weiterentwickelt. Die Aktivitäten des KUZ konzentrieren sich auf Rohstoff und Compoundierung, Formteil- und Werkzeugkonzeption, Thermoplast- und PUR-Verarbeitung, Verbindungstechnik sowie die Kunststoffprüfung. Dabei spielt die prozessbasierte Integration von speziellen Funktionen in Formteilen eine wichtige Rolle in Richtung Effizienz. Darüber hinaus wird interdisziplinär die nachhaltige Verwertung des Kunststoffes ebenso wie das werkstoffliche Recycling betrachtet.

Ausgewählte Forschungsprojekte 2020

- Untersuchungen zur Fügenahtgestaltung für geschäumte Thermoplastformteile unter Anwendung des Infrarotschweißens
- Verbesserte Antistatik in Sandwich-Bauteilen aus Kunststoff
- 3D-HRD: Druckfähige Kompositmaterialien und hohlglaskugelgefüllte Kunststoffe

Digitalisierung und **Künstliche Intelligenz**

Einen zunehmenden Anteil nimmt die im KUZ junge Forschungsdisziplin Digitalisierung und Vernetzung von Prozessen ein. Diese bietet die Möglichkeit entlang der gesamten Prozesskette Informationen aufzuzeichnen, zu verarbeiten und zu analysieren. Die Digitalisierung beeinflusst Transformationsprozesse exponentiell. Mittelfristig können Technologien mit Hilfe von KI bzw. Machine Learning gesteuert werden. Die automatisierte Analyse großer Datenmengen und deren Vergleich mit mathematischen Modellen ermöglicht eine neuartige Form der wissenschaftsgeleiteten Erkenntnis. Auf diesem Wissensfundus aufbauend will das KUZ Chancen und Herausforderungen für die Branche ableiten. Vor diesem Hintergrund wurden unter Zuhilfenahme technischer Lösungen, Ansätze im Bereich der Produkt- und Prozessentwicklung vorangetrieben und Forschungsprojekte gestartet.

Ausgewählte Forschungsprojekte 2020

- KIQuality: Qualitätsprognose an Spritzgießformteilen mittels KI-unterstützter Fehleranalyse
- Smarter US-Nietassistent – KI verbindet: Intelligente Qualitätsprognose beim Ultraschallnieten durch ein selbstlernendes System
- Digitalisierung der PUR-Verarbeitung
- KIScha: Schaumcharakterisierung an TSG-Bauteilen mittels KI-unterstützter Bruchflächenanalyse

7

Eidesstattliche Erklärung

Hiermit versichere ich, dass meine Bachelorarbeit zum Thema: „Entwicklung einer Methodik zur Qualitätsvorhersage im Spritzgießprozess mittels maschinellen Lernens am Beispiel des Formteilgewichtes“ selbständig verfasst wurde und dass keine anderen Quellen und Hilfsmittel als die angegebenen benutzt wurden. Diese Aussage trifft auch für alle Implementierungen und Dokumentationen im Rahmen dieses Projektes zu.

Brandenburg an der Havel, den 12. Oktober 2021,

_____

(Tim Garbe)