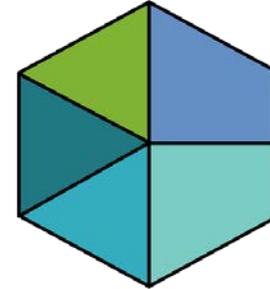




**Hochschule
Kaiserslautern**
University of
Applied Sciences

Angewandte
Ingenieurwissenschaften
Kaiserslautern



elab2go
Mobile
Engineering
Lab

Machine Learning in der Industrie: Predictive Maintenance am Beispiel eines Automotive-Datensatzes

Prof. Dr. Eva Maria Kiss, M. Sc. Anke Welz

Offene Digitalisierungsallianz Pfalz



WissensBar

Transfer@HSKL, 04.05.2021

elab2go – Mobile Engineering Lab: ein Projekt der Offenen Digitalisierungsallianz Pfalz

elab2go.de ist eine Plattform, die die Funktionsweise technologischer Trends an praxisrelevanten didaktisch aufbereiteten Demonstratoren veranschaulicht.

Aktuell im Fokus:

Maschinelles Lernen

und

Predictive Maintenance

mit

Anwendungsfällen im Automotive- und Energiesektor



Wer sind wir? Das elab2go-Team stellt sich vor

Fachbereich Angewandte Ingenieurwissenschaften Hochschule Kaiserslautern



**Prof. Dr.
Eva Maria Kiss**

Arbeitsgebiete:

- Machine Learning-Anwendungen
- Datenmodellierung und Datenbanken
- Web-Technologien und Cloud Computing



M. Sc. Anke Welz

Arbeitsgebiete:

- Klassifikations- und Vorhersagemodelle
- Machine Learning-Anwendungen
- Web-Technologien und –Sprachen, App-Erstellung

Was ist Maschinelles Lernen?

Eine Maschine **lernt** aus der **Erfahrung E** hinsichtlich einer Klasse von **Aufgaben T** und dem **Performance-Maß P**, falls die Performance P hinsichtlich T mit E sich verbessert.

[Tom Mitchell, Machine Learning, 1997]

Informell:

Maschinelles Lernen ist ein Teilgebiet der Künstlichen Intelligenz, das es Systemen ermöglicht, auf Basis von Trainingsdaten automatisch zu lernen.

Maschinelles Lernen befasst sich mit der Entwicklung lernfähiger Systeme und Algorithmen.

Historische Entwicklung

Maschinelles Lernen hat seit seinen Anfängen um 1950 mehrere Auf-und-ab-Phasen erlebt. Datenzentrisches Maschinelles Lernen ist seit 1990 eine Disziplin der Datenanalysten und Statistiker. Seit ca. 2005 hat **Deep Learning** die Forschung beflügelt und zu einem neuen Hype geführt.

2005 **Nature-Artikel „Deep Learning“**

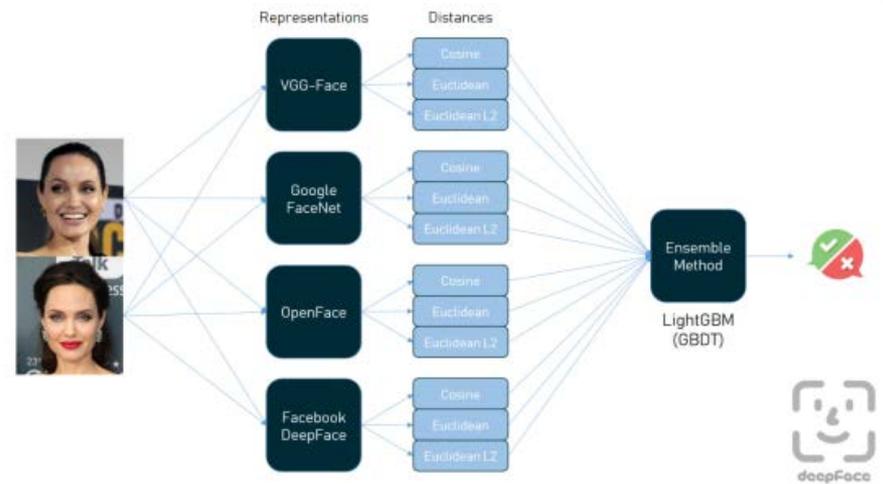
2010 **Kaggle** geht online

2012 **Google Brain-Forschung:**

Spracherkennung, Fotosuche,
YouTube-Videoempfehlungen, Robotik

2014 **Facebook DeepFace:**
Gesichtserkennung in Fotos

...



[>> DeepFace-Projekt auf pypi.org](https://pypi.org/project/deepFace/)

Aktueller Hype bedingt durch Big Data

Durch die verstärkte Speicherung anonymisierter Nutzerdaten im Internet und die Digitalisierung in der Industrie (Industrie 4.0) sind **riesige Datenspeicher** entstanden, **die als Basis für lernende Systeme verwendet werden können.**

Der Bedarf, aus den schnelllebigem und (semi)-strukturierten Daten sinnvolle Informationen zu extrahieren (Spracherkennung, Bilderkennung) hat zur Weiterentwicklung der Sprachen und Bibliotheken für Maschinelles Lernen geführt.

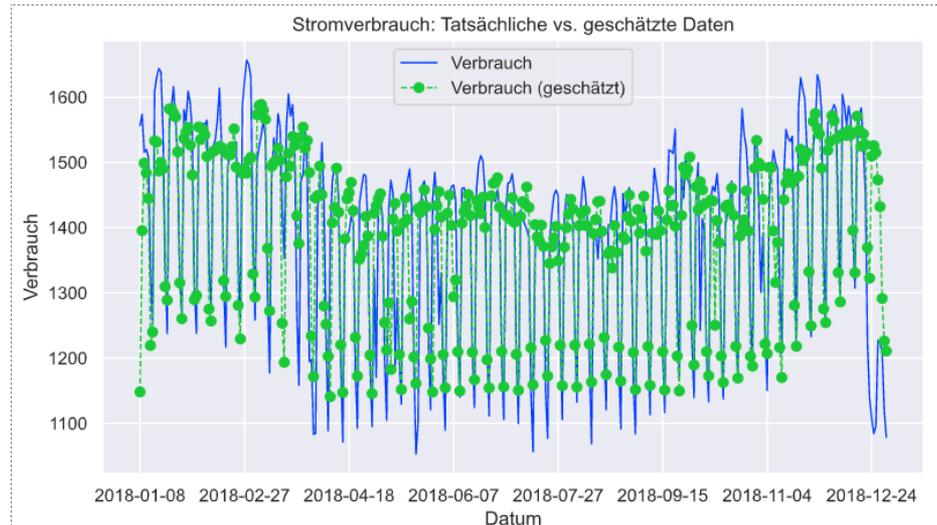
Heute wird Angewandtes Maschinelles Lernen verstärkt auch von Informatikern und Ingenieuren eingesetzt.

Machine Learning: Ziele und Anwendungen

Machine Learning-Algorithmen haben im weitesten Sinne das Ziel, aus Input-Daten **sinnvolle Zusammenhänge zu erkennen** und daraus Regeln abzuleiten. Anwendungen sind z.B.

Muster erkennen, Prognosen erstellen, Trends vorhersagen, Daten nach bestimmten Kriterien gruppieren.

	Verbrauch	Wind	Solar	Wind+Solar
Datum				
2014-01-01	1080.08	220.08	30.49	250.58
2014-01-02	1343.10	304.17	13.68	317.85
2014-01-03	1379.78	340.64	23.51	364.15
...
2017-12-29	1295.09	584.28	29.85	614.13
2017-12-30	1215.45	721.25	7.47	728.71
2017-12-31	1107.11	721.18	19.98	741.16



Machine Learning-Anwendungen in der Industrie

Anwendungsgebiete in der Industrie sind:

▪ Predictive Maintenance

- Klassifikation von Ausfällen
- Prognose der Entwicklung von Sensordaten

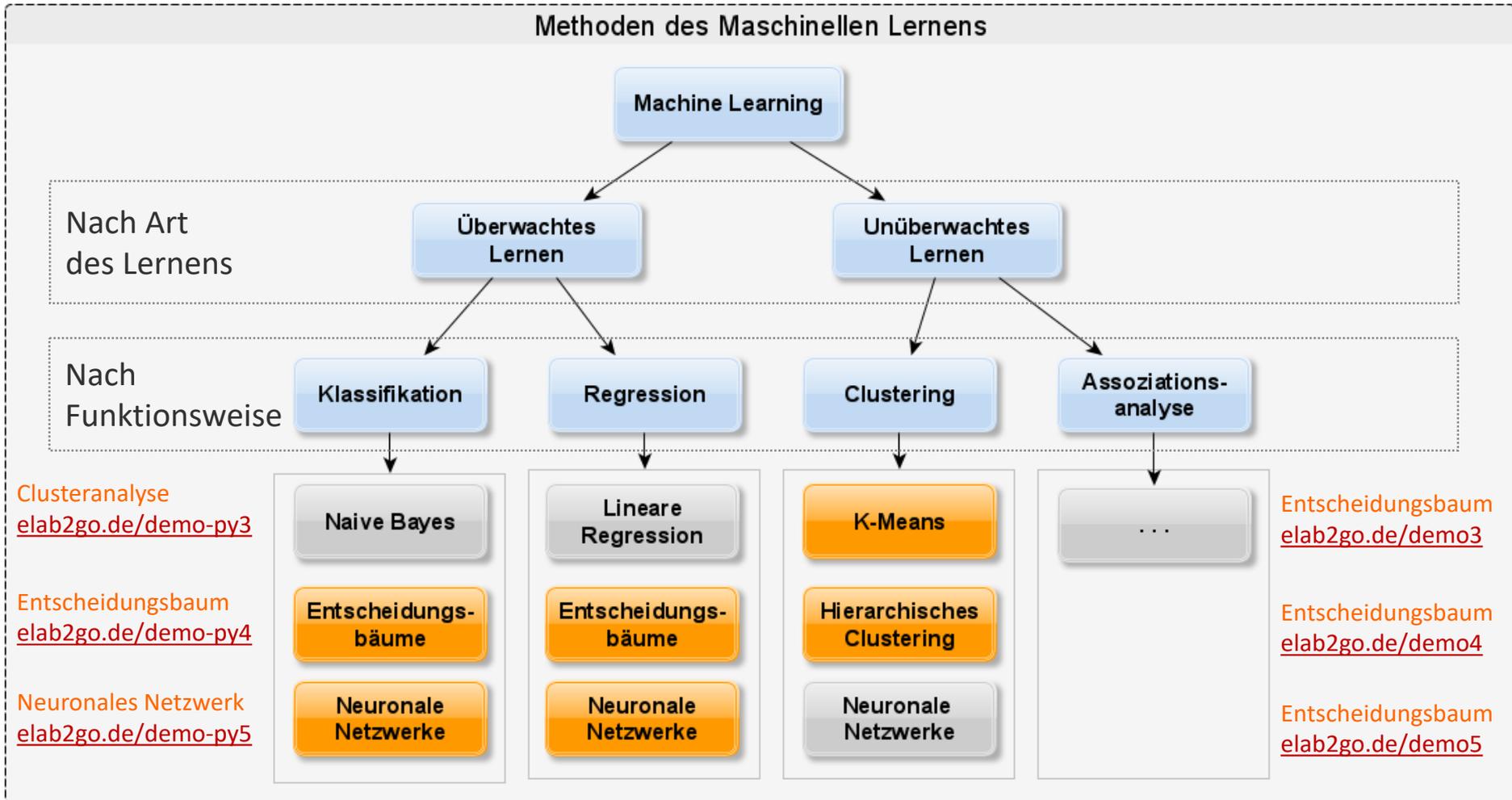
▪ Predictive Analytics

- Analytisches Customer Relationship Management: Produktempfehlungen, Optimierung von Marketing-Kampagnen
- Betrugsaufdeckung: Ist eine E-Mail Spam oder nicht?

▪ Für Internet, Soziale Medien, Marketing:

- Bild- und Texterkennung, Text-To-Speech; Videoüberwachung
- Virtuelle Assistenten, Stimmungsanalyse

Machine Learning: Methoden und Algorithmen



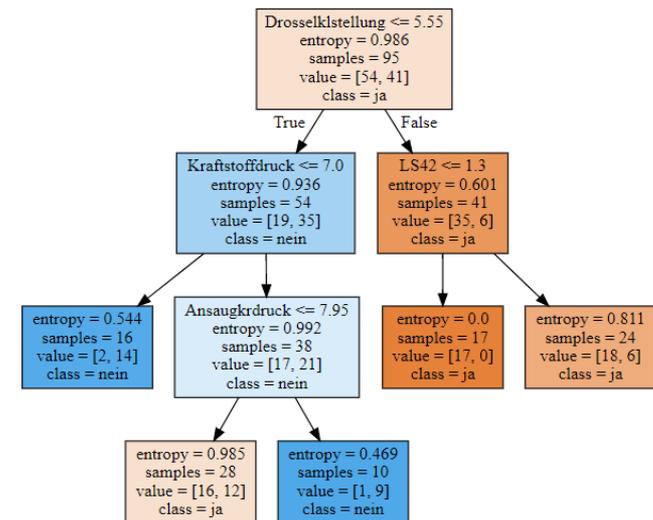
Machine Learning #1: Überwachtes Lernen

Beim **überwachten Lernen** liegt für jeden Datensatz der Input-Daten eine Bewertung vor, d.h. die Daten sind schon in Kategorien unterteilt.

Beispiel: **Entscheidungsbaum-Klassifikationsmodell** für die Vorhersage von Ausfällen an einem Automotive-Datensatz

Automotive Datensatz:
oben Trainings-Daten, unten: Testdaten

Messungsnummer	Ausfall	Kuehlmitteltemperatur	EinspritzmengeKurzzeit	Luftdruck	KatalysatorTempKategorie
M_001	nein	2,6	,9	2,8	normal
M_002	ja	9,2	22,7	2,8	hoch
M_003	nein	3,2	28,5	,1	normal
M_004	ja	2,8	2,6	,2	normal
M_005	ja	4,9	14,4	3,0	hoch
M_006	nein	,8	5,5	2,9	hoch
M_007	nein	6,2	26,4	,1	normal
⋮					
Messungsnummer	Ausfall	Kuehlmitteltemperatur	EinspritzmengeKurzzeit	EinspritzmengeLangzeit	Kraftstoffdruck
M_137	?	2,6	,9	4,2	13,8
M_138	?	9,2	22,7	15,3	8,6
M_139	?	3,2	28,5	8,7	4,9
M_140	?	2,8	2,6	5,9	7,4



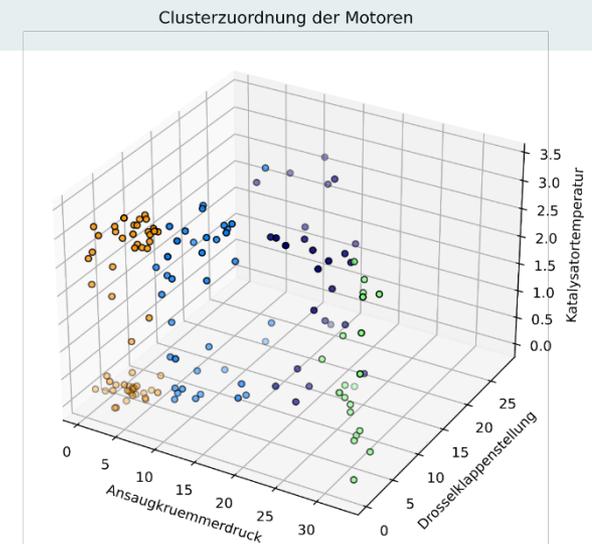
Machine Learning #2: Unüberwachtes Lernen

Beim **unüberwachten Lernen** liegen für die Input-Daten **keine Bewertungen** vor, d.h. das Lernverfahren muss ein Modell ohne Zielvorgaben erstellen, indem es Muster in den Daten erkennt.

Beispiel: **Clusteranalyse** für die Gruppierung der Ausfälle an einem Automotive-Datensatz

Die Frage, die hier mit einer Clusteranalyse beantwortet werden soll, lautet:

"Welche Clusterzuordnung der Motoren erfolgt bzgl. der drei Merkmale, die den stärksten Einfluss auf den Ausfall hatten?"



Tools und Sprachen für Maschinelles Lernen

Für die Entwicklung einer ML-Anwendung werden drei Sprachen mit den dazu passenden Entwicklungsumgebungen verwendet:

- R und RStudio → **elab2go Demo2, Demo3, Demo4**
 - Die klassische Lösung (open source)
 - Wird bevorzugt von Datenanalysten in Offline-Szenarien eingesetzt
- Python → **elab2go Demo-PY3, Demo-PY4, Demo-PY5**
 - Die innovative Lösung (open source)
 - Wird im Internet-Umfeld eingesetzt, in Online- oder Echtzeit-Szenarien, von Einsteigern und Programmierern
- MATLAB → **elab2go Demo5**
 - Die enterprise Lösung (kommerziell)
 - Wird bevorzugt von Ingenieuren eingesetzt

Anwendungen im Automotive-Bereich

Zwei Apps aus dem Werkzeugkasten für die **Vorausschauende Wartung** im Automotive-Bereich:

Die **Interaktive PredMaintApp** beantwortet die Frage:

„Welche Kombination der Motor-Sensorwerte führt zu einem Ausfall?“

[>> Demo 4: PredMaintApp \(R-Shiny\)](#)

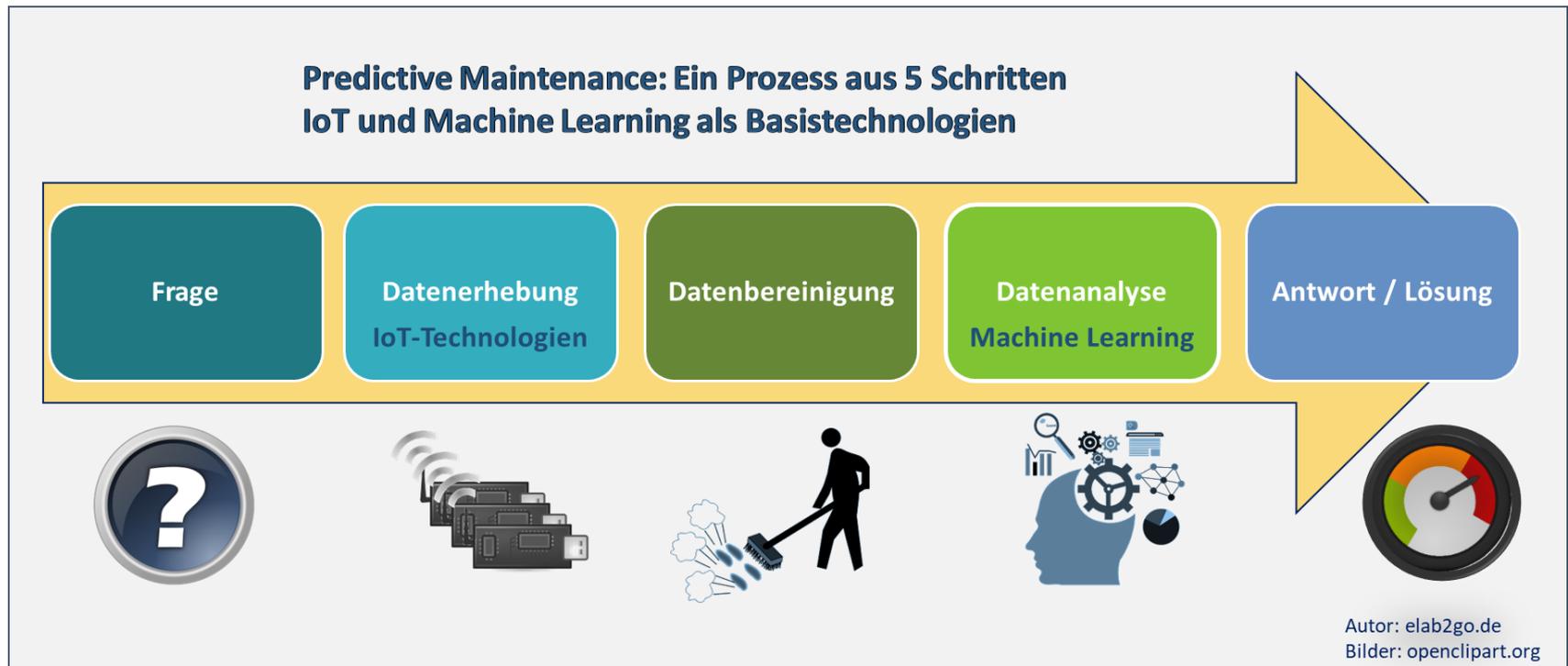
[>> Demo-PY4 PredMaintApp \(Python\)](#)

Die **Clusteranalyse "Automotive in 3D"** beantwortet die Frage

„Welche Clusterzuordnung der Motoren erfolgt bzgl. der drei Merkmale, die den stärksten Einfluss auf den Ausfall hatten?“

[>> Demo-PY3: Clusteranalyse "Automotive in 3D"](#)

Predictive Maintenance: ein Prozess aus 5 Schritten



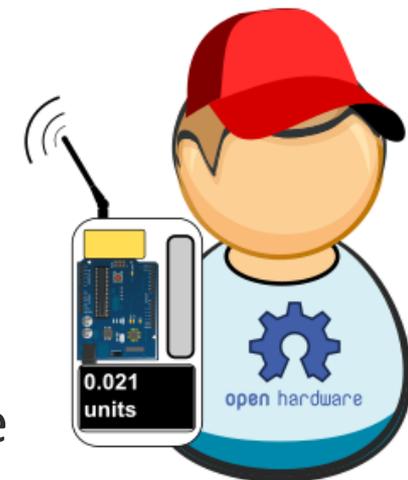
Vorteile der Predictive Maintenance

Durch die Vorausschauende Wartung

- lassen sich ungeplante Maschinenausfälle vermeiden und
- Ressourcen für Instandhaltungsarbeiten besser planen, dies **verringert Kosten** und **verbessert die Produktivität** der Maschinen.

Durch die andauernde Überwachung mittels der bereits installierten Sensoren

- kann das Wartungssystem verbessert und
- immer individueller an die zu überwachende Anlage angepasst werden.



Interaktive PredMaintApp (R-Shiny): <https://elab2go.shinyapps.io/predmaintapp/>

elab2go.shinyapps.io/predm

PredMaintApp

- Info
- Datenanalyse
- Performance
- Vorhersage

 **Welcher Motor wird ausfallen und eine Wartung benötigen?**

Die Antwort liefert ein Vorhersage-Modell, dessen Erstellung ein Schritt des Predictive Maintenance-Prozesses ist:

1. Auswahl der Merkmale
2. Einstellen der Parameter

elab2go.shinyapps.io/predm

PredMaintApp

- Info
- Datenanalyse
- Performance
- Vorhersage

 **Wie sicher ist die Vorhersage eines Motor-Ausfalls mittels Modell?**

Die Antwort liefern die Kennzahlen **Accuracy, Precision, Recall und AUC**, die mittels des Testverfahrens der Kreuzvalidierung erhoben werden.

Performance des Vorhersage-

elab2go.shinyapps.io/predm

PredMaintApp

- Info
- Datenanalyse
- Performance
- Vorhersage

 **Welcher Motor wird ausfallen und eine Wartung benötigen?**

Die Antwort liefert das in den vorherigen Schritten erstellte und validierte Vorhersage-Modell. Durch Anwendung des Modells auf neu erfasste Daten lässt sich eine Vorhersage für den Motorausfall machen.

Interaktive PredMaintApp: Die Vergangenheitsdaten enthalten eine Bewertung

Für den Automotive-Datensatz liegt eine Bewertung der Beobachtungen vor. Die 140 Motoren sind in die beiden Kategorien **AUSFALL JA** oder **NEIN** unterteilt.

Die gewählten Daten

Zeige Beobachtungen Suche:

	Ausfall	Kuehlmitteltemperatur	EinspritzmengeKurzzeit	EinspritzmengeLangzeit	Kraftstoffdr
1	nein	2.6	0.9	4.2	
2	ja	9.2	22.7	15.3	
3	nein	3.2	28.5	8.7	
4	ja	2.8	2.6	5.9	
5	ja	4.9	14.4	23.9	

PredMaintApp 1: Merkmale auswählen

Die Merkmale, die zur Erstellung des Baums herangezogen werden, können hier gewählt werden:

Merkmale

Zur Auswahl anklicken:

- Kühlmitteltemperatur
- Einspritzmenge Kurzzeit
- Einspritzmenge Langzeit
- Kraftstoffdruck
- Ansaugkrümmerdruck
- Drosselklappenstellung
- Einlasslufttemperatur
- Lambdasonde11
- Lambdasonde21
- Lambdasonde31
- Lambdasonde41
- Lambdasonde12
- Lambdasonde22
- Lambdasonde32
- Lambdasonde42
- Kraftstoffleitungsdruck
- Abgasrückführung
- Kraftstoffdampfsäuberung
- Kraftstoffeinsatz
- EVAPDruck
- Luftdruck
- Katalysatortemperatur Kategorie
- alle

PredMaintApp 2: Parameter einstellen

Der Entscheidungsbaum kann über das Einstellen von Parametern optimiert werden:

Parameter

Min. Anz. Beobachtungen pro Knoten:

5 15 50



Mit diesem Parameter wird festgelegt, dass nur Knoten erlaubt sind, zu denen mindestens 15 Merkmalskombinationen führen.

Maximale Tiefe des Baumes:

1 15

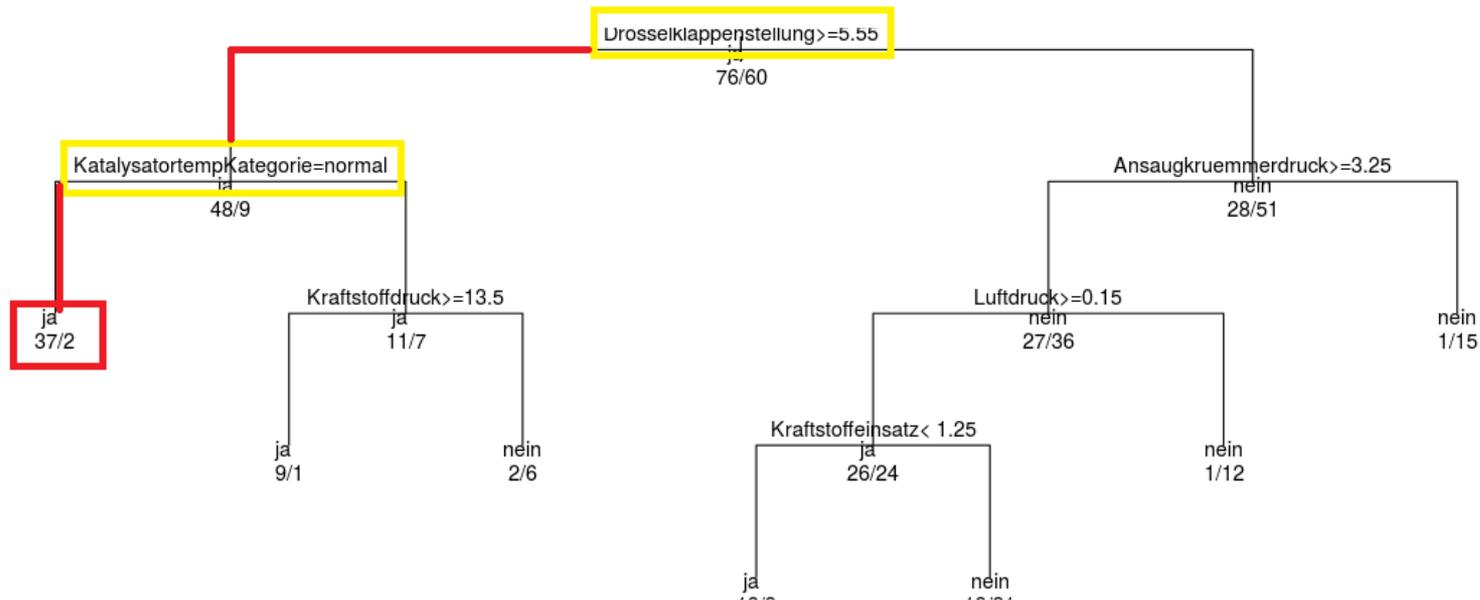


Mit diesem Parameter wird die maximale Tiefe des Baums festgelegt, also wie viele Verzweigungen der Baum maximal enthalten soll.

Diese Parameter ändern nicht nur das Aussehen des Baums, sondern auch seine Performance (→ Live-Demo).

PredMaintApp 3: Entscheidungsbaum interpretieren

Je Knoten: Eine Bedingung für das Merkmal prüfen, gehe nach links (Bedingung erfüllt) oder rechts (Bedingung NICHT erfüllt) zum weiteren Knoten bis zu einem Ja/Nein- Blatt des Baumes.

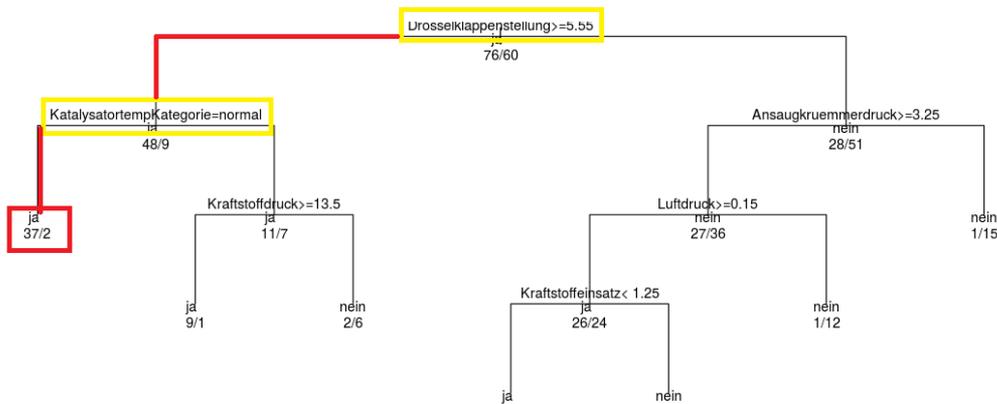


PredMaintApp 4: Performance und Vorhersage

Nach einer Bewertung des Modells anhand von verschiedenen Kennzahlen, erfolgt bei Vorliegen eines „guten“ Modells, die Vorhersage für neue Daten.

Z.B. wird für die neue Motor-Sensorkombination beim Durchlaufen des Baums ein Ausfall vorhergesagt.

Drosselklappenstellung	KatalysatortempKategorie	Kraftstoffdruck	Ansaugkruemmerdruck	...
13.80	normal	13.80	23.40	...



Kennzahlen ermitteln

recall	precision	accuracy	auc
0.77	0.73	0.74	0.76

recall	precision	accuracy	auc
gut	gut	gut	gut

Interaktive PredMaintApp: Fazit

Die PredMaintApp ist eine praktische Anwendung im Kontext Industrie 4.0 und ein greifbares Beispiel für das „Internet der Dinge“ (IoT).

- Vorgehen: Erhebung und Speicherung von Mess- und Produktionsdaten → Information über den aktuellen Zustand
- Analyse der Daten: Anwendung von Methoden und Algorithmen des maschinellen Lernens, hier: Entscheidungsbaum
- Vorhersage: Zustand in Ordnung oder Empfehlung der Wartung.



Clusteranalyse „Automotive in 3D“

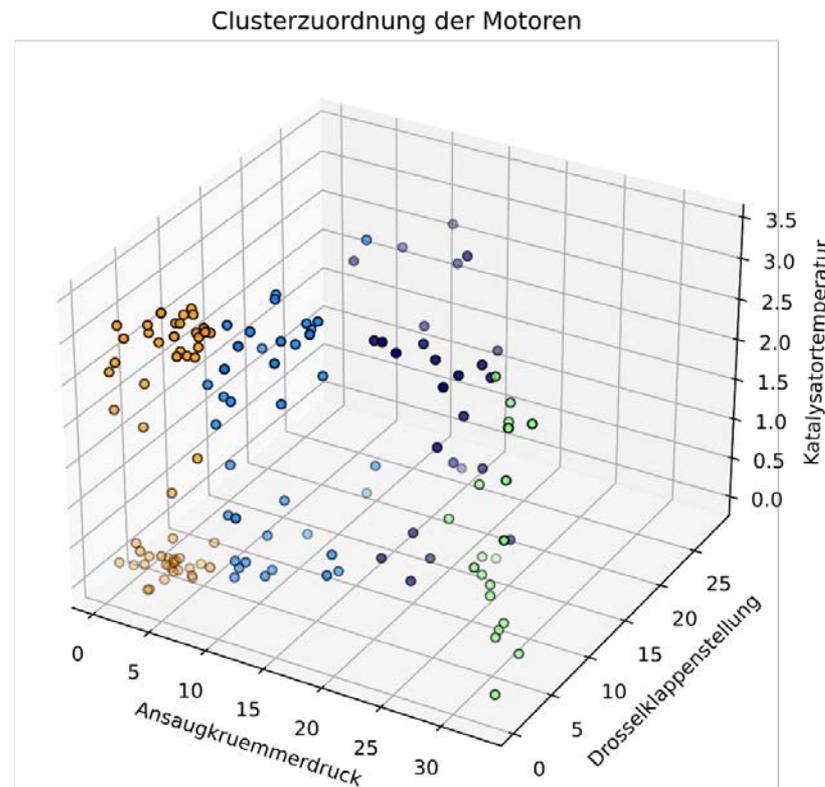
Die Vergangenheitsdaten enthalten keine Bewertung

Betrachte nur die drei Merkmale, die am stärksten zum Ausfall des Motors beitragen, ausgewählt: Ansaugkrümmerdruck, Drosselklappenstellung und Katalysatortemperatur.

	Ansaugkruemmerdruck	Drosselklappenstellung	Katalysatortemperatur
0	23.4	14.8	2.3
1	5.7	3.0	2.8
2	3.7	3.0	0.1
3	23.8	17.1	0.1
4	5.5	4.1	3.2

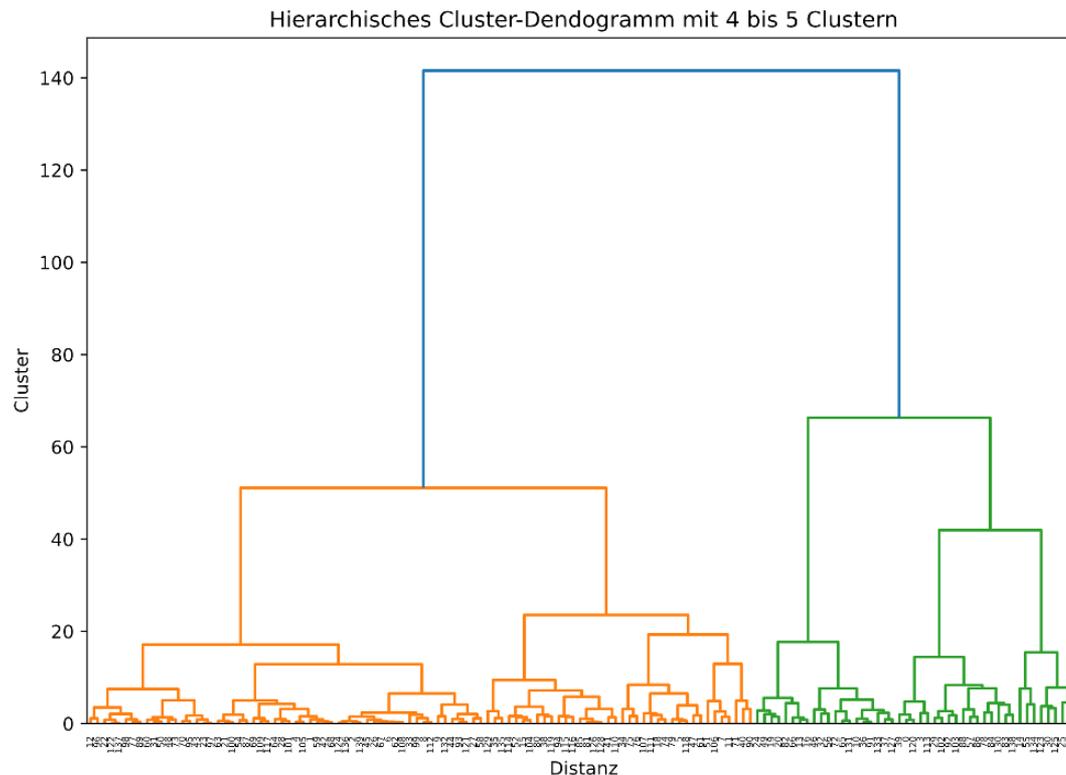
Clusteranalyse „Automotive in 3D“ (Python)

Für neue Motoren ist anhand der drei Merkmale eine Zuordnung zu den vorher gebildeten Clustern möglich, also findet eine **Vorhersage in die Cluster** statt.



Schritt 1: Anzahl der Cluster bestimmen

Zunächst wird mit Hilfe eines Linkage-Algorithmus ein Dendrogramm erstellt, um die Anzahl der Cluster herauszufinden.
Hier: 4 oder 5 Cluster



Schritt 2: Zuordnung zu den Clustern

Nach Festlegen der Clusteranzahl liefert die detaillierte hierarchische Clusteranalyse eine Zuordnung, die folgendermaßen interpretiert wird: der erste Motor ist im Cluster 0, der zweite im Cluster 3, ... der letzte im Cluster 3.

```
Erstelle Modell für Hierarchisches Clustering ...
```

```
Anzahl Beobachtungen: 140
```

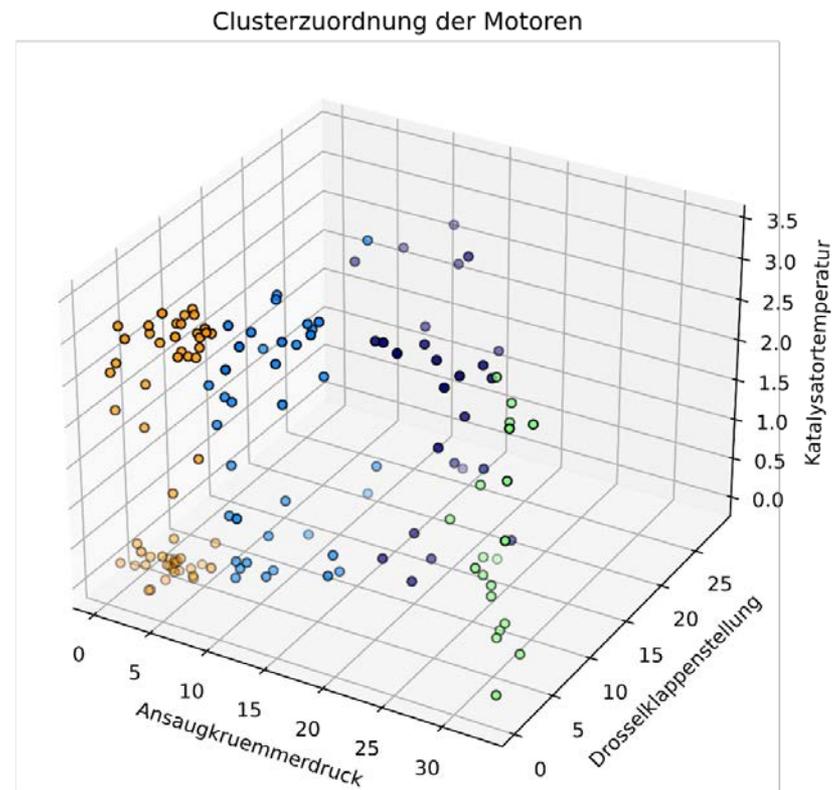
```
Zuordnung der Beobachtungen zu Cluster-Zentren 0, 1, 2, 3:
```

```
[0 3 3 0 3 1 3 1 3 3 2 1 3 2 0 1 2 0 1 2 2 3 3 3 2 0 3 3 3 0 0 3 2 3 1 1 2  
2 1 0 1 1 3 3 3 3 2 1 3 2 3 1 1 0 3 0 2 0 3 3 3 1 3 3 3 2 2 3 3 3 3 1 2 3  
1 1 1 3 0 1 1 1 2 0 0 1 0 3 0 3 1 2 0 3 1 3 3 3 3 3 3 3 0 0 1 3 1 1 3 3 1  
1 3 0 1 1 1 3 1 1 0 3 3 0 3 0 1 2 1 1 0 2 3 2 0 1 3 3 0 3]
```

Schritt 3: Visualisierung der Cluster

Bei der Bildung und Visualisierung der Cluster fällt auf, dass

- 1) die Cluster unabhängig von der Katalysatortemperatur gebildet wurden
- 2) Cluster: Ansaugkrümmerdruck niedrig und Drosselklappenstellung niedrig (Farbe: orange)
- 3) Cluster: Ansaugkrümmerdruck hoch und Drosselklappenstellung niedrig (Farbe: grün)



Schritt 4: Interpretation der Cluster

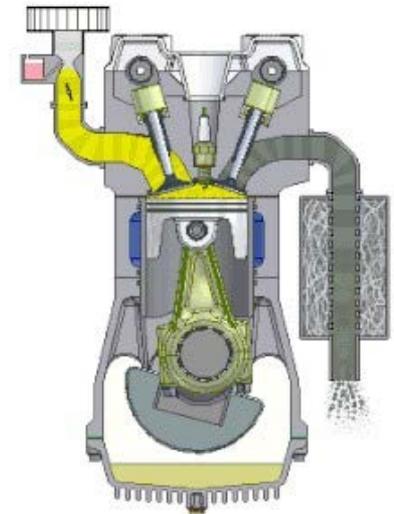
Die Bildung der Cluster (orange und grün) spiegeln den funktionalen Zusammenhang zwischen den Bauteilen Ansaugkrümmer und Drosselklappe wieder.

- Die beiden Zustände Leerlauf und Anlassen des Motors werden abgebildet:
Leerlauf = Drosselklappe geschlossen
Anlassen = leicht geöffnete Drosselklappe
- Ansaugkrümmerdruck während des Leerlaufs: geschlossene Drosselklappe drosselt den Luftdurchfluss aus Umgebung, wenn der Motor Luftmasse von Ansaugkrümmer weg pumpt.
→ Verringerung der Ansaugkrümmerluftdichte
(orangenes Cluster)

Clusteranalyse „Automotive in 3D“: Fazit

Die mit drei vorab gewählten Merkmale gebildeten Cluster geben ein passendes Abbild hinsichtlich der Wirkungsweise der zugeordneten Bauteile.

- Eine erweiterte Clusteranalyse mit zusätzlichen Merkmalen stellt ein geeignetes Mittel dar, um die Wirkungen der Bauteile untereinander und ihrer Zustände in nicht kategorisierten Daten zu erkennen.
- **Der Gesamtzustand des Motors, seine Leistungsfähigkeit, eine mögliche Wartung oder ein Ausfall** kann mit Hilfe einer Clusteranalyse abgebildet werden.



Zusammenfassung und Ausblick

Maschinelles Lernen ist die Entwicklung lernfähiger Systeme:

- durch **überwachtes / unüberwachtes Lernen**
- mit Modellen: **Entscheidungsbaum, Clusteranalyse, Künstliche Neuronale Netze, Deep Learning**
- Programmiersprachen und Tools: **Python, R, MATLAB (...)**

Auf elab2go decken wir aktuell die wichtigsten Tools mit zwei Anwendungsfällen aus dem Automotive- und Energiesektor ab. **Weitere Erläuterungen und Demos sind auf [elab2go](#) zu finden.**

Wir freuen uns über Ihr Feedback!