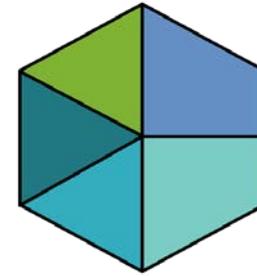




**Hochschule
Kaiserslautern**
University of
Applied Sciences

Angewandte
Ingenieurwissenschaften
Kaiserslautern



elab2go
Mobile
Engineering
Lab

Predictive Maintenance mit R: Eine interaktive R-Shiny-App für die Ausfallprognose von Motoren

Prof. Dr. Eva Maria Kiss, M. Sc. Anke Welz
Offene Digitalisierungsallianz Pfalz

Digitaltag
2021

Was ist Predictive Maintenance?

<https://www.elab2go.de/demo2/>

Der Begriff Predictive Maintenance kann mit „Vorausschauender Wartung“ übersetzt werden und 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
- Vorhersage: Zustand in Ordnung oder Empfehlung der Wartung.

→ Predictive Maintenance = **Vorausschauende Wartung**

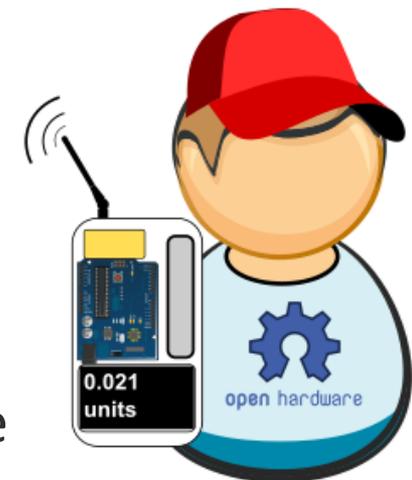
Vorteile der Predictive Maintenance

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

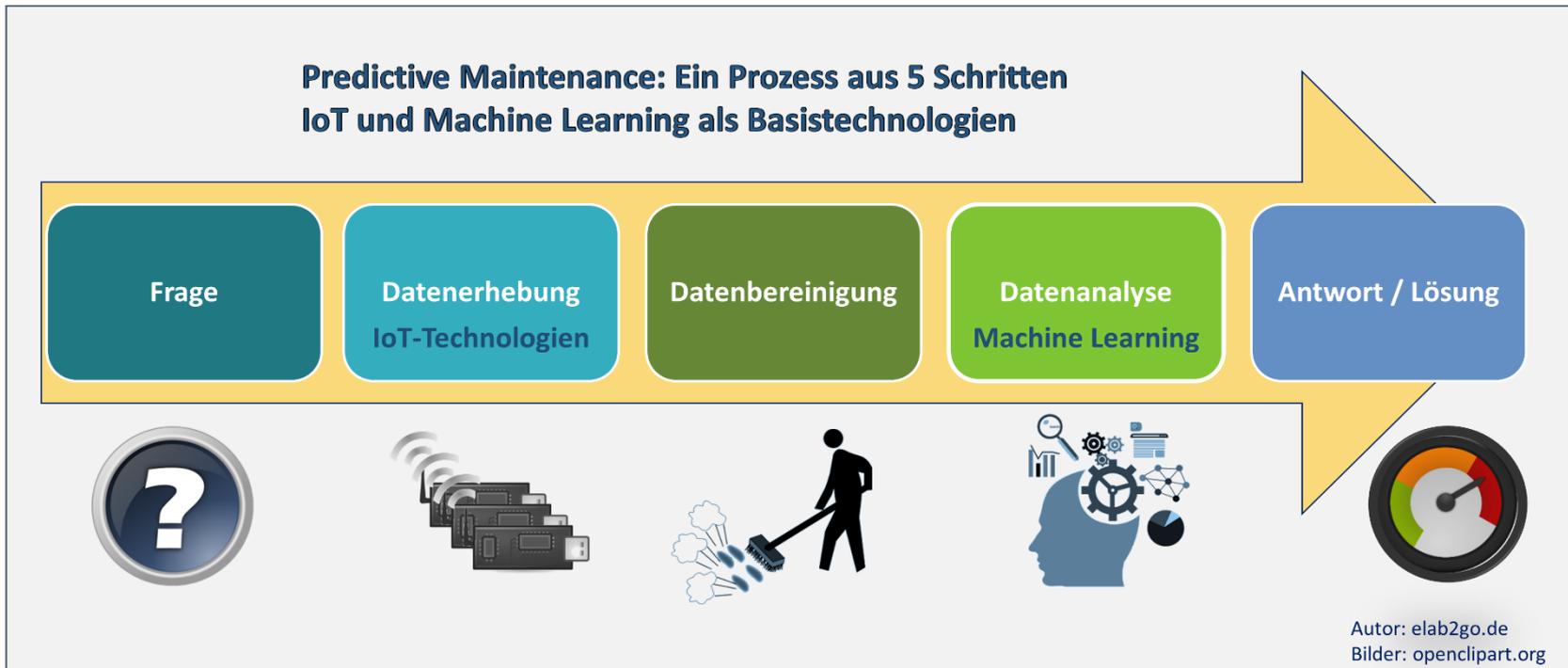
- kann das Wartungssystem verbessert und
- immer individueller an die zu überwachende Anlage angepasst werden.



Predictive Maintenance in fünf Schritten

<https://www.elab2go.de/demo2/>

Predictive Maintenance wird in vielen Unternehmen als ein Prozess in fünf Schritten abgebildet.



Predictive Maintenance in fünf Schritten

Schritt 1: Die Fragestellung

z.B. Wann (bei welchen Messwerten) sollte ein Bauteil ausgewechselt oder gewartet werden?

Schritt 2: Datenerhebung

Die Anlage wird mit Sensoren ausgestattet, die die nötigen Werte (Temperatur, Laufbewegung, Druck, usw.) an den entsprechenden Stellen erfassen und speichern.

Schritt 3: Datenbereinigung

Fehlmessungen oder nicht vollständige Messwerte entfernen oder bearbeiten.

Predictive Maintenance in fünf Schritten

Schritt 4: Datenanalyse

Anwendung von Methoden und Algorithmen des maschinellen Lernens, je nach Art und Funktionsweise des Lernens.

Schritt 5: Die Antwort/Lösung

Die Ausgabe der Analyse-Software wird nun von dem Analysten in eine verständliche Antwort übersetzt und ein **anwendbares Wartungssystem** entwickelt.

Datenanalyse im Rahmen der Pred. Maint.

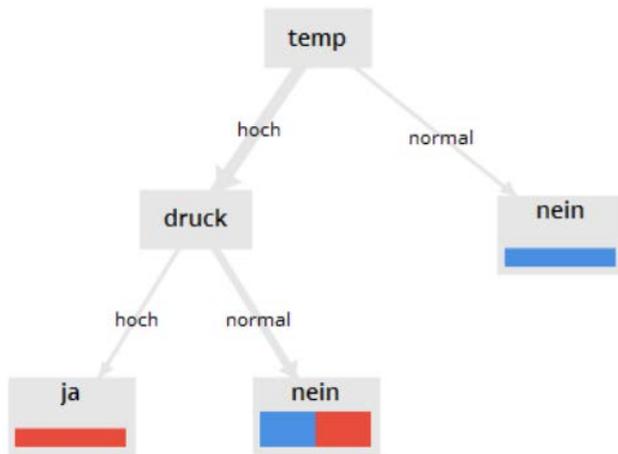
<https://www.elab2go.de/demo2/datenanalyse.php>

- Anwendung von **Methoden und Algorithmen des maschinellen Lernens**, je nach Art und Funktionsweise des Lernens.
- Die Input-Daten werden in **Trainingsdaten** und **Testdaten** unterteilt.
- Auf Basis der Trainingsdaten wird mit Hilfe eines passenden Algorithmus ein **Vorhersagemodell** erstellt, dessen Güte mit Hilfe von **Performance-Kennzahlen** ermittelt wird.

Datenanalyse im Rahmen der Pred. Maint.

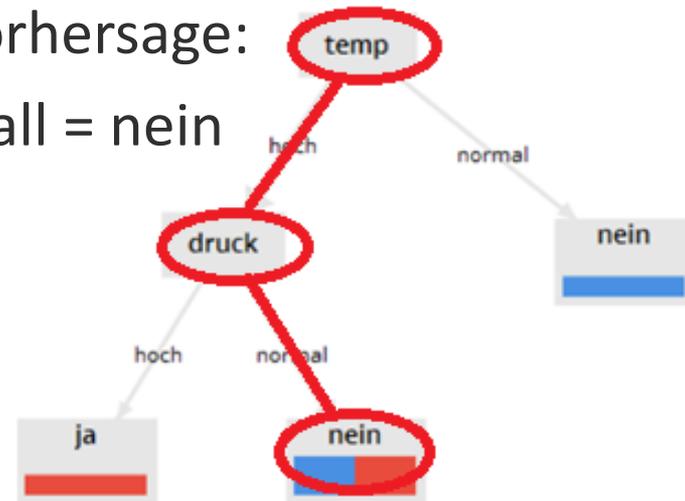
Der **Entscheidungsbaum** als Anwendung des Überwachten Lernens ist ein Klassifikationsmodell und stellt einen möglichen Ansatz dar.

1. Vorhersagemodell:

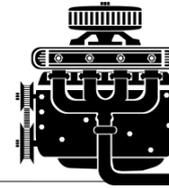


2. Neue Daten: {temp=hoch, druck=normal}

3. Vorhersage:
Ausfall = nein



Der Automobildatensatz



An Motoren wurden mittels Sensoren

- Temperatur- und Druckmessungen
- sowie Mengenangaben zum Kraftstoff und zu Abgasdämpfen erfasst und zusätzlich auch, ob der **Motor ausgefallen ist oder nicht**. Insgesamt: 22 Merkmale in jeder der 140 Beobachtungen

| Messungsnr | Ausfall | Kuehlmitteltemp | EinspritzmKurz | EinspritzmLang | Einlasslufttemp | LS11 |
|------------|---------|-----------------|----------------|----------------|-----------------|------|
| M_001 | nein | 2.8 | 0.9 | 3.8 | 1.6 | 1.5 |
| M_002 | ja | 9.9 | 24.6 | 15 | 1.2 | 0.8 |
| M_003 | nein | 3.4 | 28.5 | 8.7 | 1.8 | 1.5 |
| M_004 | ja | 2.8 | 2.8 | 6.3 | 2.4 | 2.2 |
| M_005 | nein | 4.9 | 15.4 | 25.9 | 1.5 | 1.1 |
| M_006 | nein | 0.8 | 5.2 | 15.7 | 2.1 | 2.4 |
| M_007 | ja | 5.9 | 27.2 | 13.2 | 2 | 1.5 |
| M_008 | ja | 2.6 | 16.3 | 18.3 | 8.7 | 2.7 |
| ... | | | | | | |
| M_136 | nein | 3.1 | 5.4 | 8.8 | 1.6 | 0.3 |
| M_137 | ja | 5.3 | 10.5 | 28.7 | 2.6 | 1.4 |
| M_138 | nein | 1 | 7.9 | 29.7 | 0.8 | 0.2 |
| M_139 | ja | 4.3 | 10.7 | 13.7 | 3.3 | 1.9 |
| M_140 | nein | 3.7 | 4.9 | 30.7 | 1.7 | 1.5 |

Datenanalyse – Die Fragestellung

Welche Kombination von Merkmalen wird zu einem Ausfall führen?

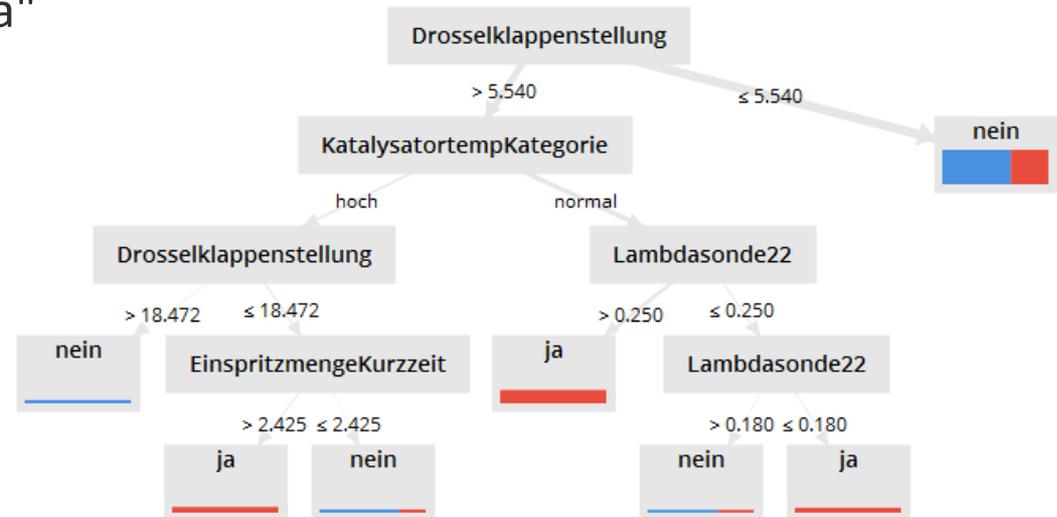
| Messungsnummer | Ausfall | Kuehlmitteltemperatur | Einspritzmenge | Kurzzeit | Luftdruck | Katalysatortemp | Kategorie |
|----------------|---------|-----------------------|----------------|----------|-----------|-----------------|-----------|
| M_005 | ja | 4,9 | | 14,4 | 3,0 | | hoch |
| M_006 | nein | ,8 | | 5,5 | 2,9 | | hoch |
| M_007 | nein | 6,2 | | 26,4 | ,1 | | normal |
| | | ⋮ | | | | | |
| M_133 | nein | 2,0 | | 3,6 | ,2 | | normal |
| M_134 | nein | 1,5 | | 1,2 | 1,4 | | normal |

**Ein Entscheidungsbaum
als Vorhersagemodell
liefert eine übersichtliche
und schnell interpretierbare
Antwort ...**

Datenanalyse – Der Entscheidungsbaum

Welche Kombination von Merkmalen wird zu einem Ausfall führen?

Wenn am Ende eines Pfades "Ja" steht, dann führt der Pfad/ die Merkmalskombination zu einem Ausfall des Motors andernfalls, d.h. bei "Nein", kommt es zu keinem Ausfall.



Tools: R: <https://www.elab2go.de/demo3/> , <https://www.elab2go.de/demo4/>

Python: <https://www.elab2go.de/demo-py4/>

MATLAB: <https://www.elab2go.de/demo5/>

Datenanalyse – Die Vorhersage

Welche Kombination von Merkmalen wird zu einem Ausfall führen?

Eine Vorhersage für neue Merkmals-kombinationen ist nun möglich: **Durchlaufen des passenden Pfad.**



Vorhersage für die neuen Daten: **kein** Ausfall

Neue Daten:

| | Drosselklappenstellung | KatalysatorTempKategorie | Lambdasonde22 |
|--|------------------------|--------------------------|---------------|
| | 13,8 | normal | 0,2 |

RStudio – Die Entwicklungsumgebung

<https://www.elab2go.de/demo3/>

- RStudio ist eine Entwicklungsumgebung und grafische Benutzeroberfläche für Datenanalyse mit Hilfe der statistischen Programmiersprache R, die in Open-Source- und kommerziellen Editionen erhältlich ist.
- Die zur Datenanalyse benötigten Algorithmen sind in Pakete zusammengefasst, die nach Bedarf geladen werden können.
- Mit Hilfe des Packages R Shiny aus R können z.B. Webseiten für Datenanalysen gebaut werden. → Demo auf elab2go



```
library(shiny)
library(shinydashboard) #for dashboard design app v1.0
library(shinyBS) #for popovers and tooltips
```

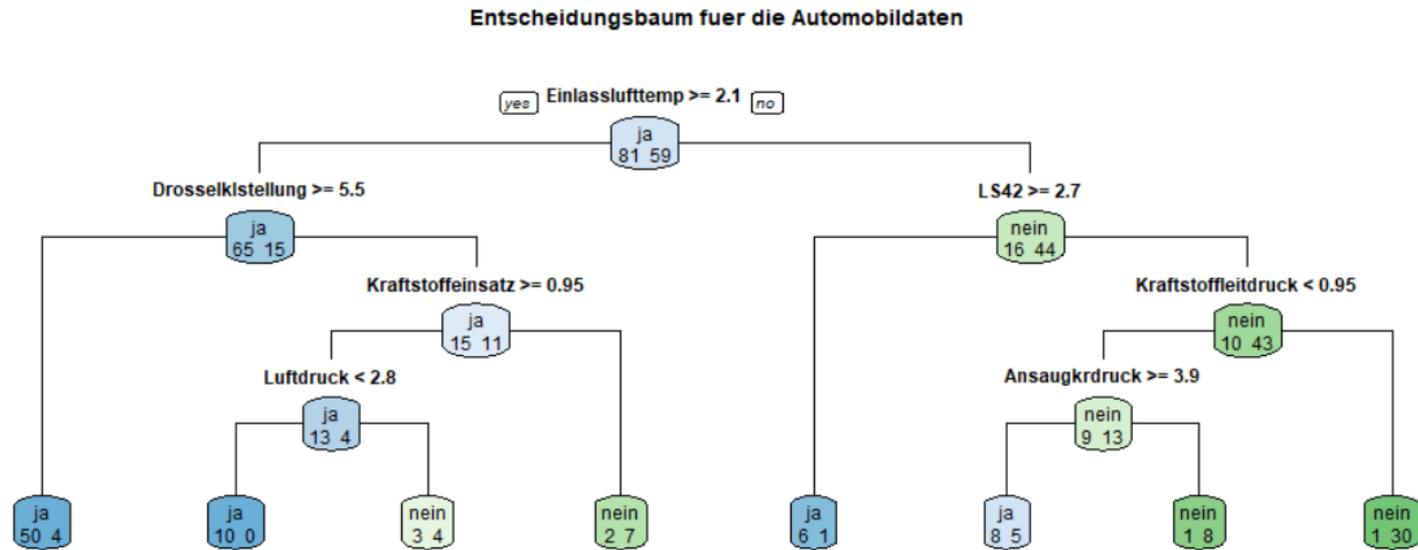
Datenanalyse mit RStudio – Der Quellcode

Im Analyse-Code werden die benötigten Pakete geladen, z.B. das "rpart"-Paket (Zeile 5), dieses enthält Funktionen, die dann zur Erstellung eines Klassifikationsmodells verwendet werden (Zeile 11-15).

```
1. # Demo 3: Predictive
   Maintenance mit R
2. #       Analyse
3.
4. # Lade benötigte Pakete
5.   library(rpart)
6. # Lese CSV-Datei und speichere
7. # sie unter dem Namen "demo3"
8.   demo3=read.csv2(
       "demo2_data.csv")
9. # Erstelle Entscheidungsbaum
10. # mit dem CART-Algorithmus
11.   fit = rpart(Ausfall ~ .,
12.              data=demo3training,
13.              method="class",
14.              parms=list( split="information"),
15.              control=list( minsplit=15))
16.
17. # Gebe den Baum in Textform aus
18.   print(fit)
```

Datenanalyse mit RStudio – Der Entscheidungsbaum

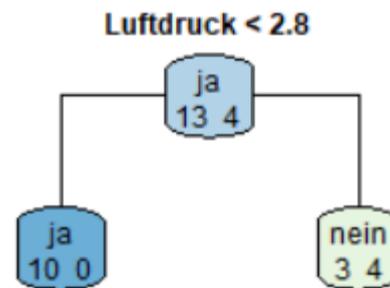
Der Entscheidungsbaum wählt nur einen Teil der 22 Merkmale zur Vorhersage eines Ausfalls aus.



An jedem Knoten wird für das jeweilige Merkmal eine Bedingung abgefragt und je nach Beobachtungswert wird nach links (Bedingung erfüllt) oder rechts (Bedingung NICHT erfüllt) zum weiteren Knoten verwiesen.

Datenanalyse mit RStudio – Der Entscheidungsbaum

- Je höher das Merkmal im Baum steht, d.h. je weiter oben dessen Position ist, desto mehr Einfluss hat es auf die Vorhersage eines Ausfalls, in unserem Baum hat die **Einlasslufttemperatur den höchsten Einfluss**.
- An der Aufteilungen in Ja/Nein in den Blättern erkennt man, wie viele Beobachtungen des Trainingsdatensatzes mit beobachteten Ausfällen Ja/Nein diesem Blatt zugeordnet werden.



- Z.B. insgesamt 7 Beobachtungen im ganzen rechten Ast, bei denen 3 mal kein Ausfall vorliegt und 4 mal eine Ausfall. Daraus lassen sich die **Wahrscheinlichkeiten für Ausfall Ja oder Nein** je Blatt bestimmen, die auch bei der Ausgabe des Baums **in Textform** angegeben sind.

Datenanalyse mit RStudio – Der Entscheidungsbaum

n= 140

```
node), split, n, loss, yval, (yprob)
```

```
* denotes terminal node
```

```
1) root 140 59 ja (0.57857143 0.42142857)
  2) Einlasslufttemp>=2.05 80 15 ja (0.81250000 0.18750000)
    4) Drosselklstellung>=5.45 54 4 ja (0.92592593 0.07407407) *
    5) Drosselklstellung< 5.45 26 11 ja (0.57692308 0.42307692)
      10) Kraftstoffeinsatz>=0.95 17 4 ja (0.76470588 0.23529412)
        20) Luftdruck< 2.8 10 0 ja (1.00000000 0.00000000) *
        21) Luftdruck>=2.8 7 3 nein (0.42857143 0.57142857) *
      11) Kraftstoffeinsatz< 0.95 9 2 nein (0.22222222 0.77777778) *
  3) Einlasslufttemp< 2.05 60 16 nein (0.26666667 0.73333333)
    6) LS42>=2.65 7 1 ja (0.85714286 0.14285714) *
    7) LS42< 2.65 53 10 nein (0.18867925 0.81132075)
      14) Kraftstoffleitdruck< 0.95 22 9 nein (0.40909091 0.59090909)
        28) Ansaugkrdruck>=3.9 13 5 ja (0.61538462 0.38461538) *
        29) Ansaugkrdruck< 3.9 9 1 nein (0.11111111 0.88888889) *
      15) Kraftstoffleitdruck>=0.95 31 1 nein (0.03225806 0.96774194) *
```

**Zuordnungen:
Ausfall JA zu
100% richtig,
Ausfall NEIN zu
57% richtig**

Datenanalyse mit RStudio – Die Vorhersage

Die Vorhersage für neue Merkmalskombinationen erfolgt mittels der „predict“-Funktion:

```
# Vorhersage für die Testdaten/neue Beobachtungen
pred = predict(fit, newdata=testdata, type="class")
```

Die Vorhersage wird ausgegeben:

Die neuen Beobachtungen

| | Kuehlmitteltemp | EinspritzmKurz | EinspritzmLang | Kraftstoffdruck | Ansaugkrdruck | Drosselkstellung |
|---|-----------------|----------------|----------------|-----------------|---------------|------------------|
| 1 | 2.6 | 0.9 | 4.2 | 13.8 | 23.4 | 13.8 |
| 2 | 9.2 | 22.7 | 15.3 | 8.6 | 5.7 | 3 |
| 3 | 3.2 | 28.5 | 8.7 | 4.9 | 3.7 | 3 |
| 4 | 2.8 | 2.6 | 5.9 | 7.4 | 24.3 | 16.2 |

... Die Vorhersage des Ausfalls (JA/NEIN) für die neuen Beobachtungen lautet:



```
Motor 1    nein
Motor 2     ja
Motor 3    nein
Motor 4     ja
```

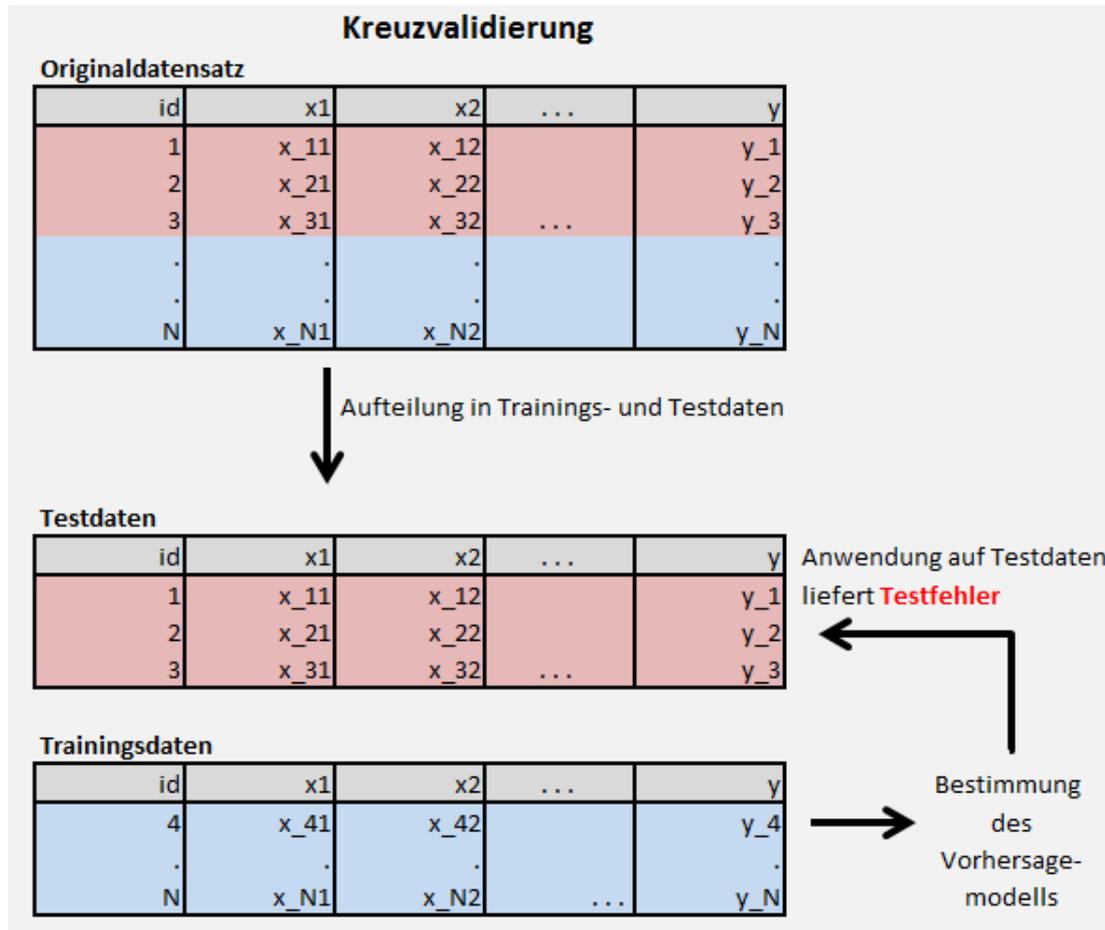
Datenanalyse – Zwischenergebnis

- Wir haben wir die Frage nach den Kombinationen von Merkmalen, die zu einem Ausfall eines Motors führen, anhand des Vorhersagemodells des Entscheidungsbaumes beantwortet.
 - Dieser Baum wurde danach interpretiert und kann zur Vorhersage eines Ausfalls bei neu erfassten Daten verwendet.
- **Vor der Vorhersage stellt sich aber die Frage wie gut das Vorhersagemodell überhaupt ist.**

Datenanalyse im Rahmen der Pred. Maint. – Fortsetzung

- Uns interessiert, wie sicher das Vorhersagemodell den Ausfall eines Motors vorhersagt.
- Anhand der Kennzahlen zum Testfehler lässt sich dies beantworten: Liegen die Kennzahlen in einem vorher festgelegten, akzeptablen Bereich? Wenn ja, dann ist das Modell gut.
- Bei einem guten Modell können wir uns auf die Aussagen und Vorhersagen durch das Modell verlassen.

Datenanalyse – Die Kennzahlen mit RStudio

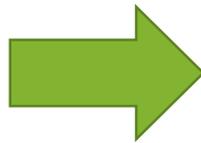


Datenanalyse – Die Beurteilung des Modells

<https://www.elab2go.de/demo3/index2.php>

Ein Modell/eine Vorhersage kann als gut/sicher bewertet werden, wenn die Kennzahlen einen hohen Wert annehmen. Was genau ein hoher Wert ist, hängt vom Verfahren und von der Einschätzung des Experten ab, d.h. welche Abweichungen er noch als in Ordnung gelten lässt.

```
recall precision accuracy auc
0.77      0.77      0.72      0.78
```



In dieser App gilt beispielhaft die Einteilung in die Wertebereiche:

- 0% bis ca. 55%: Das Modell ist verbesserungsfähig.
- 56% bis ca.65%: Das Modell ist moderat.
- ab 66%: Das Modell ist gut bis sehr gut.

Damit ergeben sich folgende Beurteilungen des Modells je nach Kennzahl:

```
recall precision accuracy auc
gut      gut      gut      gut
```

Anwendung im Automotive-Bereich

Zwei Apps aus dem Werkzeugkasten des elab2go für die **Vorausschauende Wartung** im Automotive-Bereich:

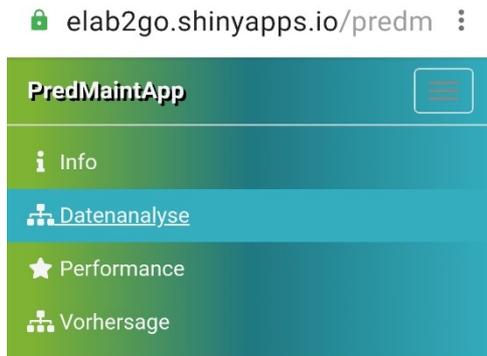
Die **Interaktive PredMaintApp** ist eine interaktive R-Shiny-App für die Ausfallprognose von Motoren und beantwortet die Frage:

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

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

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

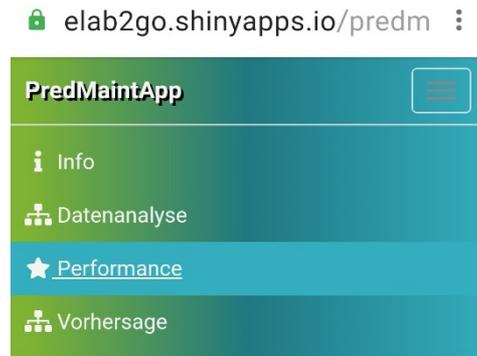
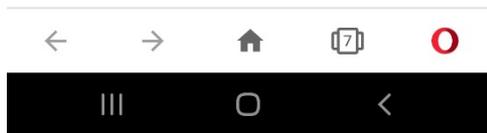
Interaktive PredMaintApp (R-Shiny): <https://elab2go.shinyapps.io/predmaintapp/>



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

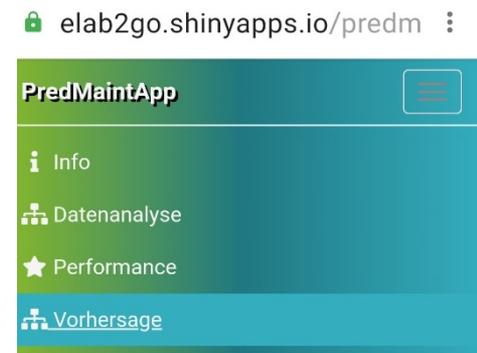
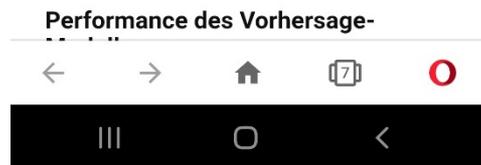
Die Antwort liefert ein Vorhersage-Modell, dessen Erstellung ein Schritt des Predictive Maintenance-Prozesses ist:

1. [Auswahl der Merkmale](#)
2. [Einstellen der Parameter](#)



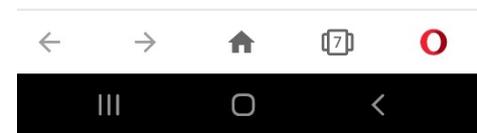
 **Wie sicher ist die Vorhersage eines Motor-Ausfalls mittels Modell?**

Die Antwort liefern die Kennzahlen [Accuracy](#), [Precision](#), [Recall](#) und [AUC](#), die mittels des Testverfahrens der Kreuzvalidierung erhoben werden.



 **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

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:

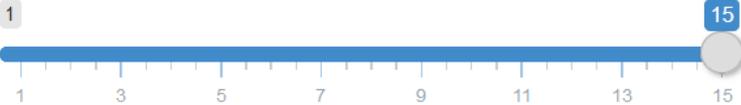
5 15 50



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

Maximale Tiefe des Baumes:

1 15



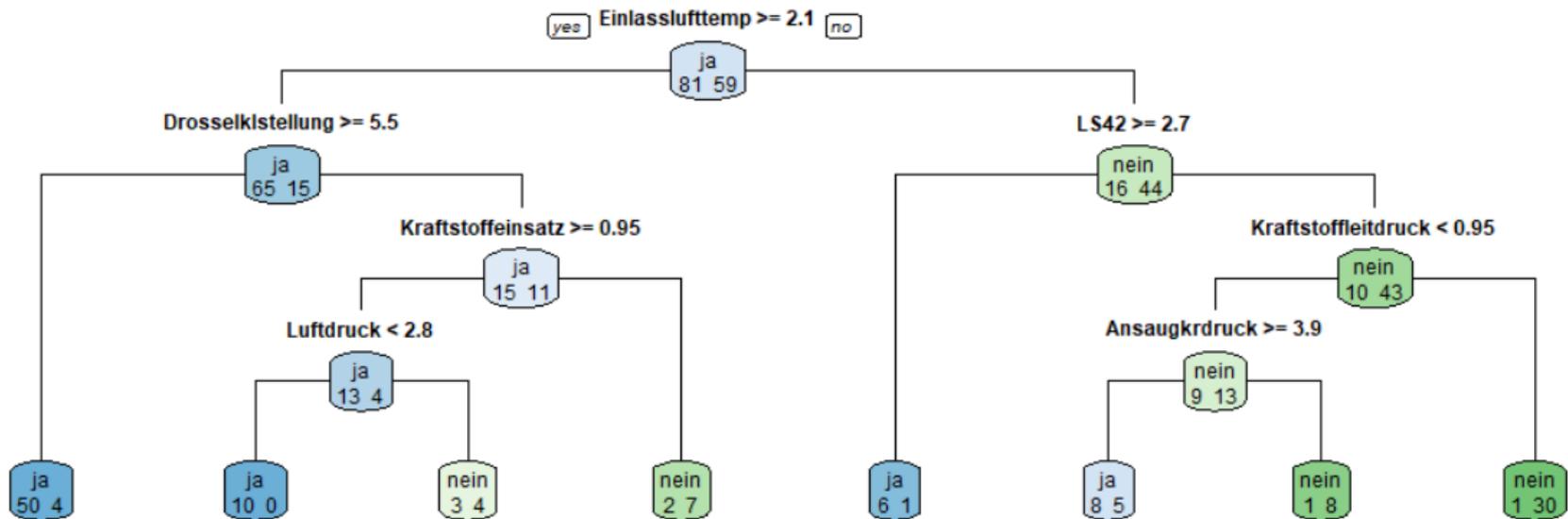
Mit diesem Parameter wird die maximale Tiefe des Baums festgelegt, also wie viele Verzweigungen der Baum maximal enthalten soll.

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

PredMaintApp 3: Entscheidungsbaum interpretieren

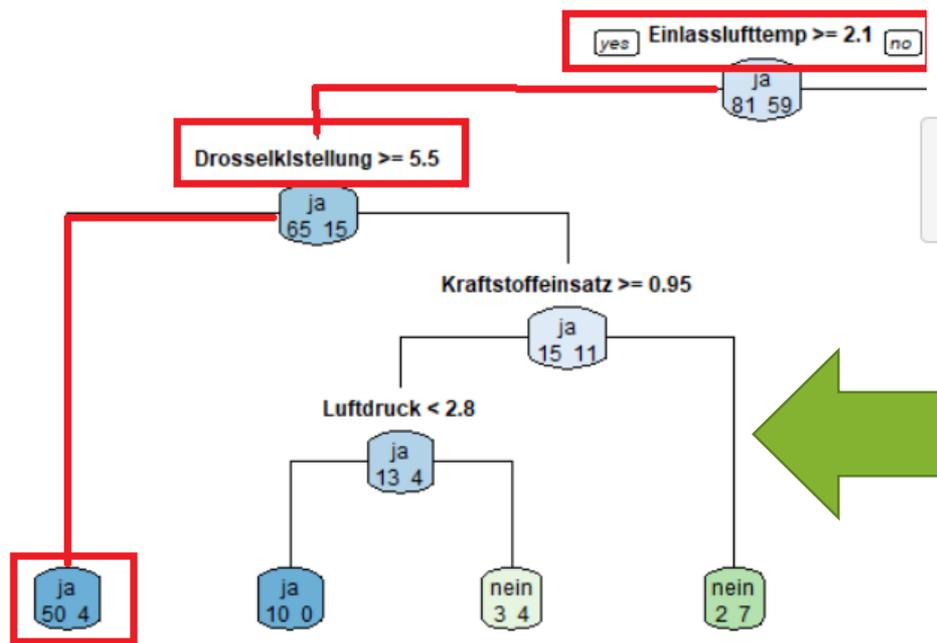
Je Knoten: Eine Bedingung für das Merkmal prüfen, gehe nach links (Bedingung erfüllt) oder rechts (Bedingung NICHT erfüllt) zum weiteren Knoten bis zu einem Ja/Nein-Blatt des Baumes.

Entscheidungsbaum fuer die Automobildaten



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.



Kennzahlen ermitteln

| recall | precision | accuracy | auc |
|--------|-----------|----------|------|
| 0.77 | 0.73 | 0.74 | 0.76 |

| recall | precision | accuracy | auc |
|--------|-----------|----------|-----|
| gut | gut | gut | gut |

| Drosselklstellung | Einlasslufttemp |
|-------------------|-----------------|
| 16.20 | 2.40 |



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.



Zusammenfassung

- Die Vorausschauende Wartung ist ein Prozess aus 5 Schritten, einer davon ist die **Datenanalyse**.
- In der Datenanalyse werden durch Anwendung von Algorithmen und Verfahren des maschinellen Lernens Modelle zur **Vorhersage eines Ausfalls** einer Maschine oder eines Bauteils erstellt.
- Die Datenanalyse beinhaltet auch die **Validierung dieser Modelle** um die Sicherheit eines Ausfalls zu bestimmen.
- Durch eine langfristige Datenerhebung können die Vorhersagemodelle verbessert und gezielt auf ein Bauteil ausgerichtet werden. (→ **Updates**)

Zusammenfassung

- Beispielhaft wird im elab2go die Datenanalyse anhand eines Automobildatensatzes durchgeführt.
 - **Verschiedene Tools (RStudio, Jupyter Notebook (Python), MATLAB)** stellen die zur Datenanalyse benötigten Algorithmen und Verfahren zur Verfügung und je nach Kontext kann eine Auswahl eines Tools erfolgen.
- Durch die Predictive Maintenance wird **frühzeitig** vor dem Ausfall einer Maschine/Bauteils gewarnt und somit **Ausfallzeiten und Kosten niedrig** gehalten.

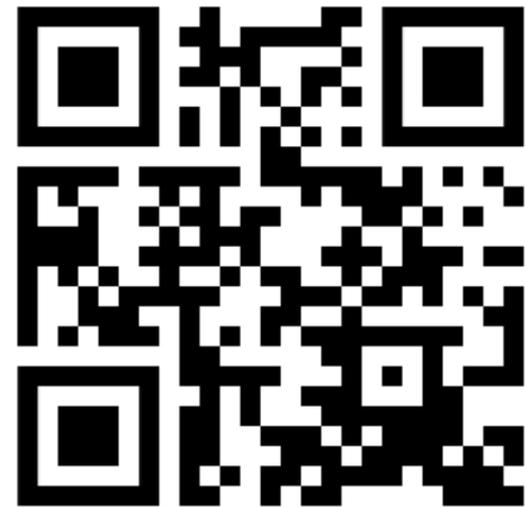
Weitere Informationen

Kontakt elab2go Mobile Engineering Lab

Prof. Dr. Eva Maria Kiss, Studiengangsleiterin Elektrotechnik
M. Sc. Anke Welz, Offene Digitalisierungsallianz Pfalz

Hochschule Kaiserslautern
Fachbereich Angewandte Ingenieurwissenschaften
Schoenstrasse 11, 67659 Kaiserslautern

E-Mail:
evamaria.kiss@hs-kl.de
anke.welz@hs-kl.de



www.elab2go.de

elab2go - Links

Demonstratoren und Erklärungen:

- <https://www.elab2go.de/demo2/>
- <https://www.elab2go.de/demo2/datenanalyse.php>
- <https://www.elab2go.de/demo3/>
- <https://www.elab2go.de/demo4/>
- <https://elab2go.shinyapps.io/predmaintapp/>
- <https://www.elab2go.de/demo5/>
- <https://www.elab2go.de/demo5/matlabPredMaint.php>
- <https://www.elab2go.de/demo-py4/>

Tutorials:

- <https://www.elab2go.de/demo-r1/>
- <https://www.elab2go.de/demo-py1/>
- <https://www.elab2go.de/demo-mat1/>