



Data-Science-Showcase

ANOMALIE-ERKENNUNG

INHALTE

- ✓ Worum geht es bei der Erkennung von Anomalien?
- ✓ So helfen wir Ihnen bei der Umsetzung
- ✓ Die wichtigsten Erkenntnisse
- ✓ Unser Fazit für Sie



Worum geht es?

WAS SIND ANOMALIEN UND WARUM KÖNNTE DEREN
ERKENNUNG FÜR SIE VON BEDEUTUNG SEIN



„Anomaly detection refers to the problem of finding patterns in data that do not conform to **ex**pected behavior.“

[Chandola et. al. (2009): “Anomaly detection: A survey”]

Worum geht es?



WAS SIND ANOMALIEN UND WARUM KÖNNTE DEREN ERKENNUNG FÜR SIE VON BEDEUTUNG SEIN?

Egal ob es um das Verhalten von Menschen oder die Messwerte von IoT Geräten geht – immer kann es zu **Abweichungen von den zu erwartenden Ergebnissen** kommen.

Diese unerwarteten Abweichungen werden als Anomalien bezeichnet und können auf **potenzielle Probleme** hindeuten, deren Identifizierung in **zahlreichen Geschäftsfeldern** von großer Bedeutung sein kann:

Erkennen von Betrugsversuchen bei Bestellungen

Überwachen von IoT-Sensordaten und Logfiles

Sicherstellen der Einhaltung von Einkaufsprozessen

Worum geht es?

ANOMALIEN ERFORDERN SCHNELLES HANDELN

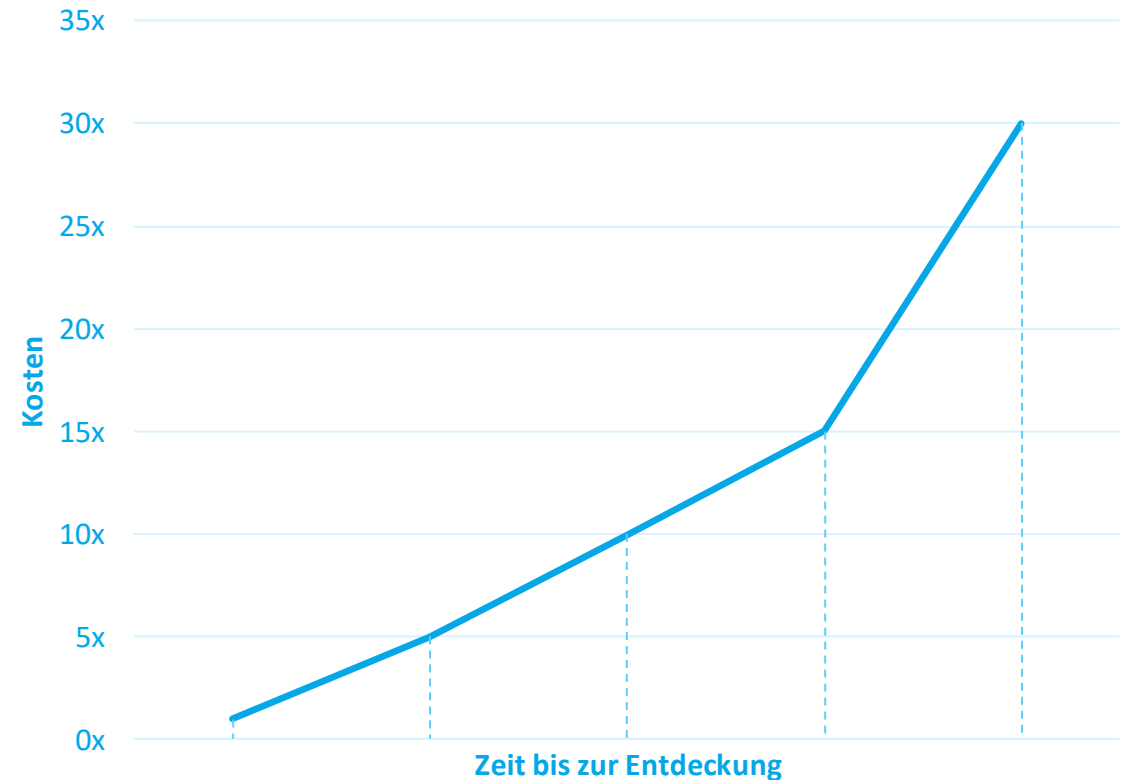
Die **verursachten Kosten einer Anomalie** sind selbstverständlich abhängig vom Geschäftsfeld in dem sie auftritt.

Ein weiterer **signifikanter Faktor** ist jedoch die **Zeit bis zu ihrer Entdeckung**:

- Stellen Sie sich einen Bug in einer Software als eine Anomalie vor.
- Die Grafik verdeutlicht den Zusammenhang zwischen der Entdeckung eines Bugs in einem Softwareprojekt und den Kosten zur Behebung.

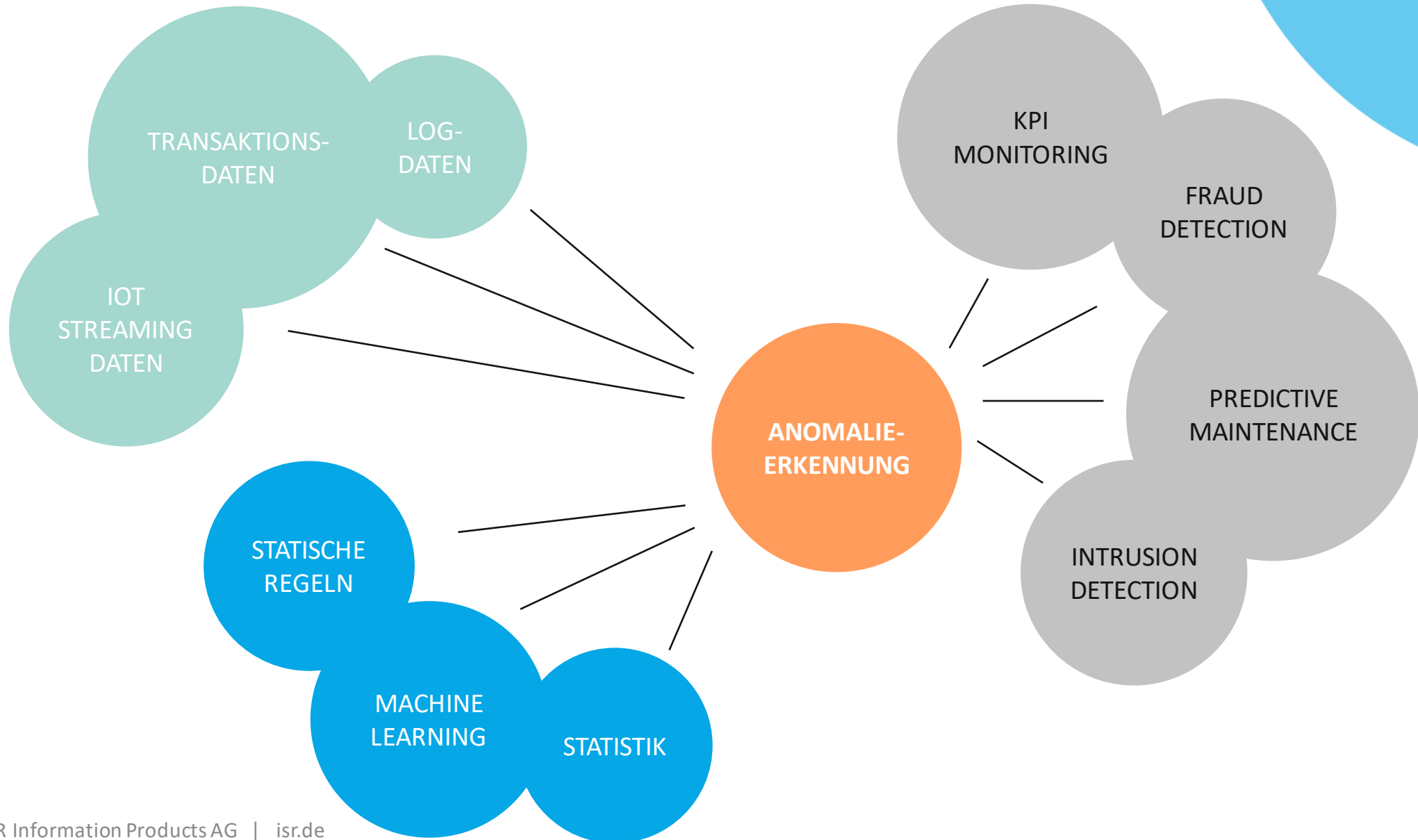
Mit unserem Showcase wollen wir Ihnen einen Lösungsansatz präsentieren, um die **Zeitspanne zwischen Auftreten und Entdecken** einer Anomalie zu **minimieren**.

Entwicklung der Kosten bis zur Entdeckung eines Fehlers



National Institute of Standards and Technology (NIST)

Anomalie-Erkennung ist vielschichtig



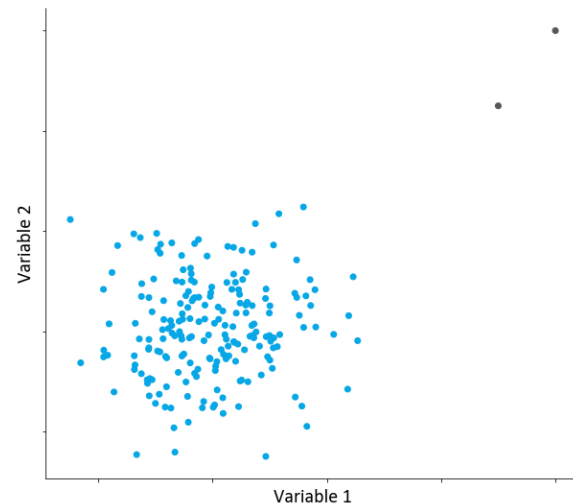
Anomalien sind vielschichtig



WELCHE ARTEN VON ANOMALIEN EXISTIEREN?

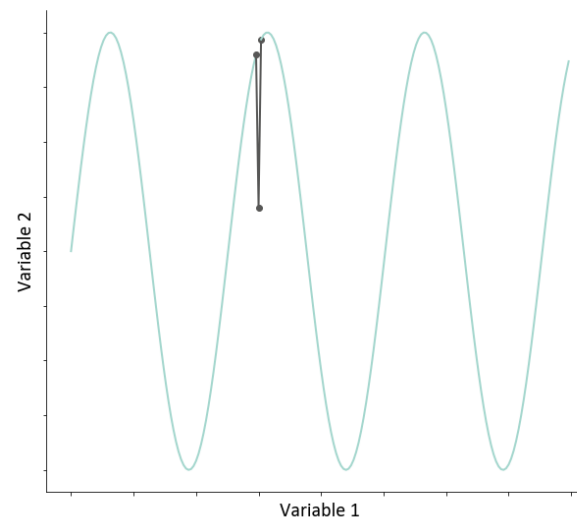
Punktuelle Anomalie

Ein Datensatz wird im Vergleich mit den restlichen Daten als anormal betrachtet.



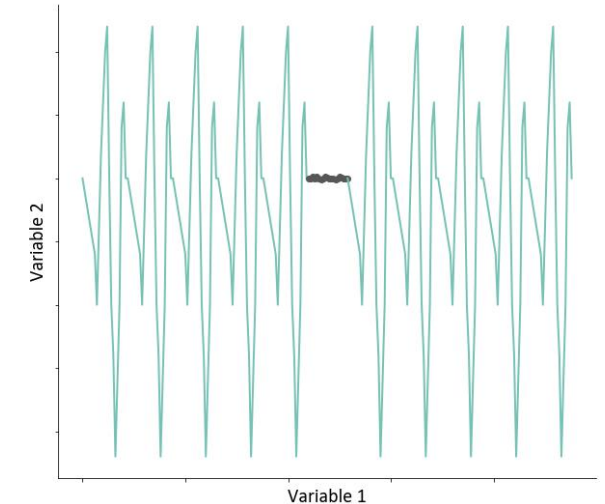
Kontextuelle Anomalie

Ein Datensatz wird in einem spezifischen Kontext als anormal betrachtet.



Kollektive Anomalie

Mehrere Datensätze werden im Vergleich mit den restlichen Daten als anormal betrachtet.



Unser Showcase

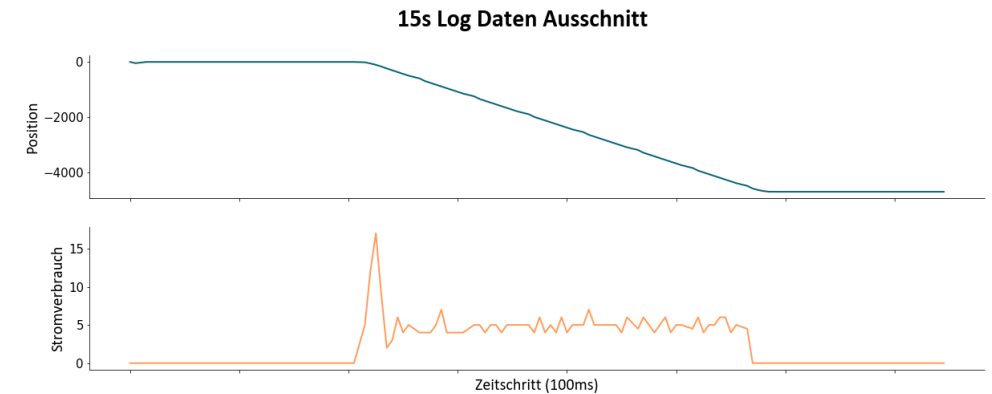
ANOMALIEN IN LOG-DATEN VON MASCHINEN

In unserem Showcase demonstrieren wir anhand eines konkreten Beispiels, wie sich Anomalien **mithilfe von Künstlicher Intelligenz** aufdecken lassen:

- Der genutzte Datensatz beinhaltet **Log-Daten**, welche die **Positionsänderungen eines Bauteils** einer Maschine sowie den damit verbundenen **Stromverbrauch** protokollieren.

Ziel ist es, **Anomalien im Stromverbrauch** zu identifizieren, welche nicht zu den Bewegungsänderungen der Bauteile passen.

- Die Erkennung kann dabei vollkommen **automatisiert** und ohne menschliches Zutun erfolgen.



<https://www.getinge.com/de/uber-uns/presse-und-medien/bildergalerie/>

Unser Showcase

ANOMALIEN IN LOG-DATEN VON MASCHINEN

Das von uns trainierte Machine-Learning-Modell hilft, **Fehlerquellen im Produktionseinsatz** oder der Entwicklung rechtzeitig zu identifizieren:

- Mechanische Probleme, die durch Reibung und Blockaden den Stromverbrauch erhöhen.
- Programmierfehler in Softwareupdates, die zu einem unerwartet hohen Stromverbrauch führen.
- Messfehler der Sensoren, welche durch Verschleiß o.ä. ungenaue Werte liefern.





Unser Lösungsansatz

SO KÖNNEN WIR IHNEN HELFEN, ANOMALIEN ZU
IDENTIFIZIEREN

Unser Data-Science-Toolkit

AUF DIESE TOOLS SETZEN WIR

Für die Umsetzung unserer Data-Science-Projekte setzen wir auf eine Mischung aus **populären Open Source Frameworks** wie



sowie „Watson Studio“ – die **neuste Data-Science-Plattform** von



Weitere Einzelheiten zu Projekten, die wir mithilfe von IBM Watson Studio bei Kunden erfolgreich umgesetzt haben, finden Sie auf [unserer Website](#).



Nicht alle Wege führen nach Rom

VON DEN DATEN ZUR MODELL-ARCHITEKTUR

Jeder Use Case erfordert ein individuelles Vorgehen bei der Konzeption des Machine-Learning-Modells – je nach Randbedingungen sowie der **Art der zu erwartenden Anomalien**.

Zentrale Fragen, die wir in Abstimmung mit Ihnen klären, sind:

- I. Sind die kausalen Zusammenhänge in den Daten bekannt?
- II. Existieren Trainingsdaten mit bereits erkannten Anomalien?
- III. Welche Arten von Anomalien sind zu erwarten?

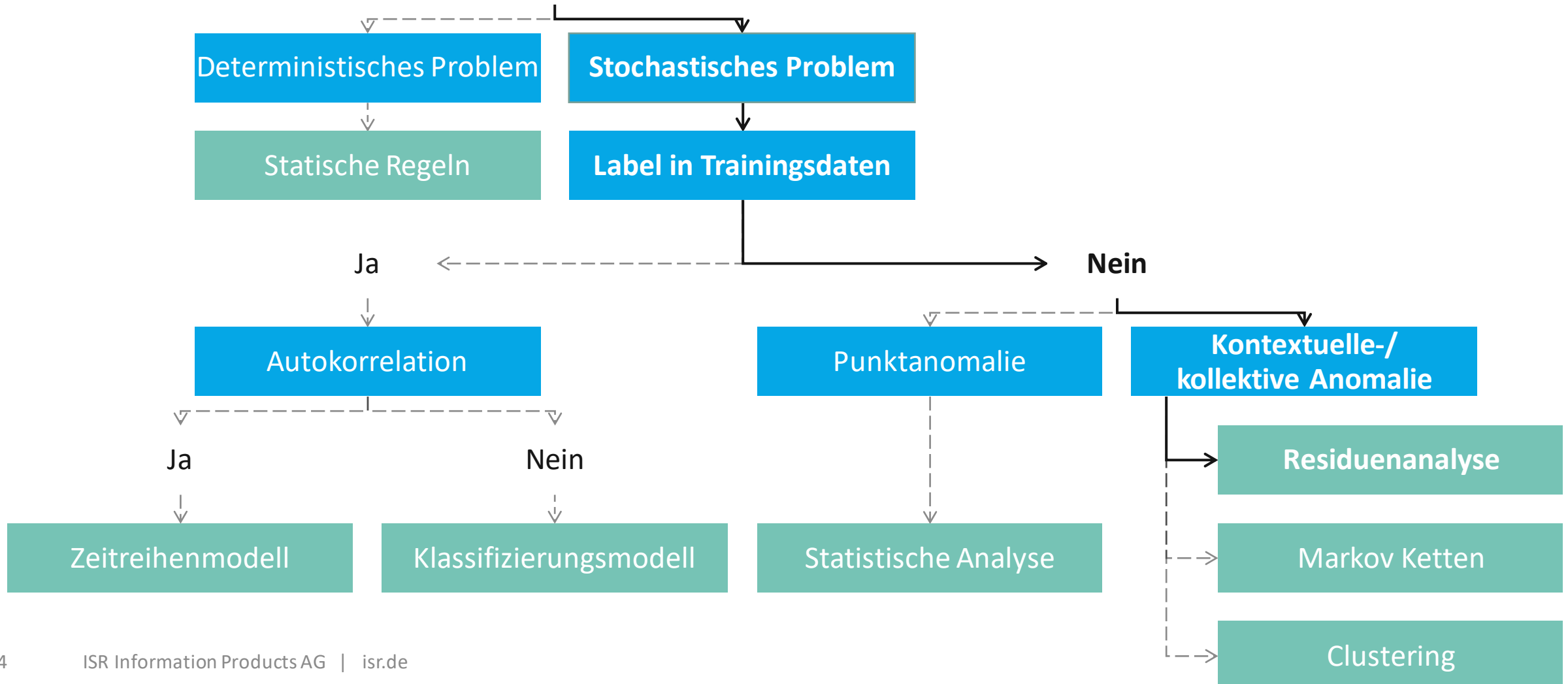


Nicht alle Wege führen nach Rom

ARCHITEKTUR-FRAMEWORK



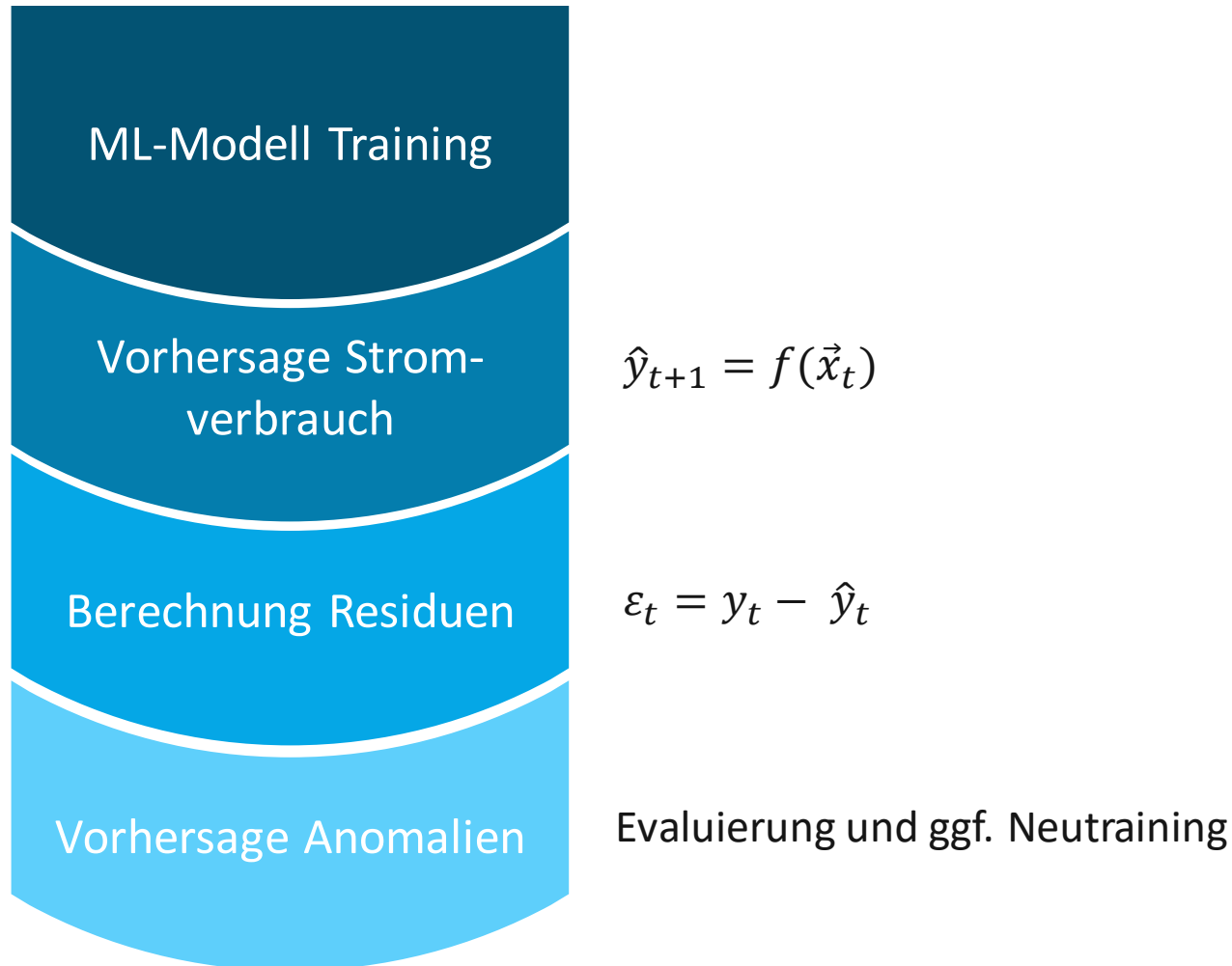
Trainingsdaten



Anomalie-Erkennung mittels Residuenanalyse



STEP-BY-STEP APPROACH



Das Vorgehen lässt sich grob in vier Phase einteilen:

- I. Zunächst wird, basierend auf den **Daten ohne Anomalien**, ein Machine-Learning-Modell (LSTM Netzwerk) trainiert.
- II. Mithilfe des Modells werden in einem Zwischenschritt **Vorhersagen** für den zukünftigen Stromverbrauch **generiert**.
- III. Die **Abweichung** zwischen tatsächlichem Stromverbrauch zum Zeitpunkt t und der Vorhersage wird berechnet.
- IV. Sofern die Abweichung einen gewissen **Schwellwert überschreitet**, liegt eine Anomalie vor.

Anomalie-Erkennung mittels Residuenanalyse



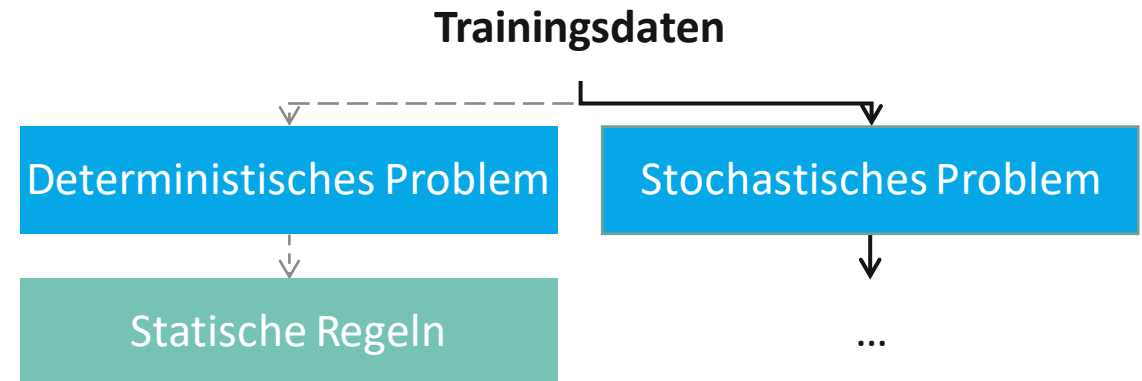
WARUM EINFACH NICHT IMMER BESSER IST

Im konkreten Beispiel existieren eine **Vielzahl von Einflussfaktoren**, von denen der Stromverbrauch letztendlich abhängt:

- Aktuelle Position des Bauteils
- Geschwindigkeit der Bewegungsänderung
- Restschwung
- Trägheitskraft
- ...

Für all diese Einflussfaktoren **manuell ein Regelset** zu definieren, ist kaum möglich.

Machine-Learning-Algorithmen können jedoch **trotz der hohen Komplexität** Muster erkennen und aus diesen lernen.



Anomalie-Erkennung mittels Residuenanalyse



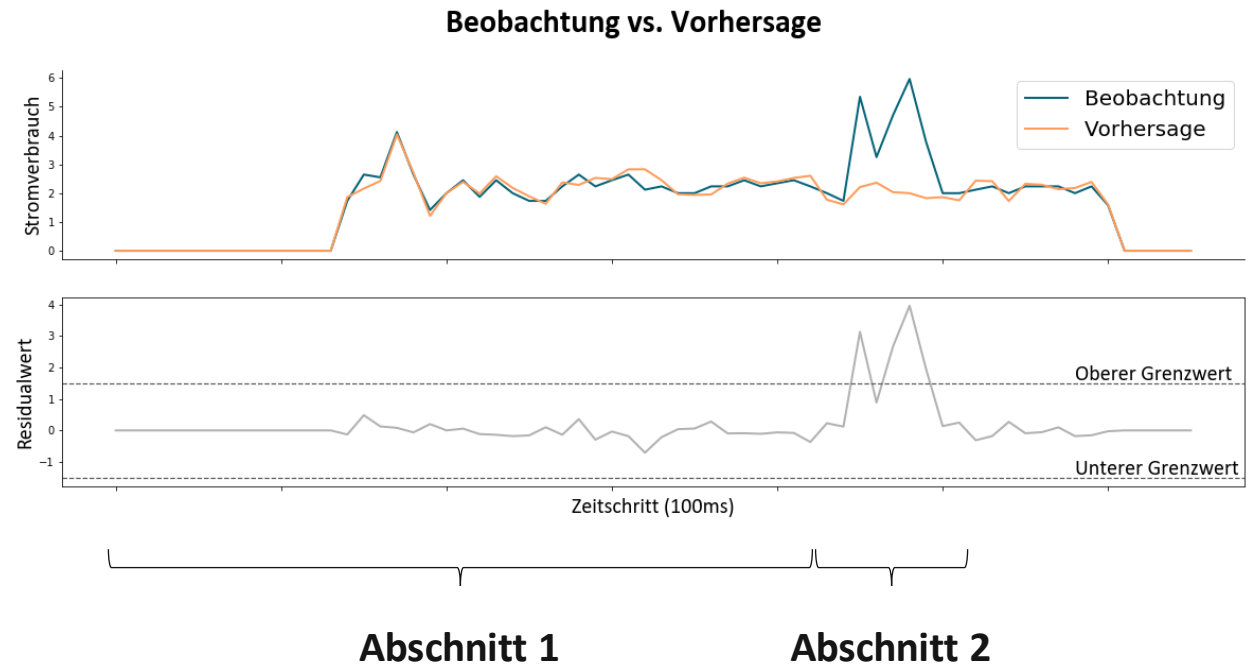
ANSCHAUUNGSBEISPIEL

In **Abschnitt 1** stimmen die beobachteten Messwerte für den Stromverbrauch mit den Vorhersagen unseres Modells sehr gut überein.

- Die Residuen liegen alle **zwischen den Grenzwerten**, weswegen von einem **normalen Verhalten** auszugehen ist.

Abschnitt 2 weist jedoch größere Unterschiede zwischen Beobachtung und Vorhersage auf.

- Die Residuen überschreiten den oberen Grenzwert deutlich, was auf eine Anomalie hindeutet.
- Stromverbrauch und Bewegungsänderung passen nicht in das bekannte Muster.



Für mehr Transparenz – Ergebnisse sichtbar machen



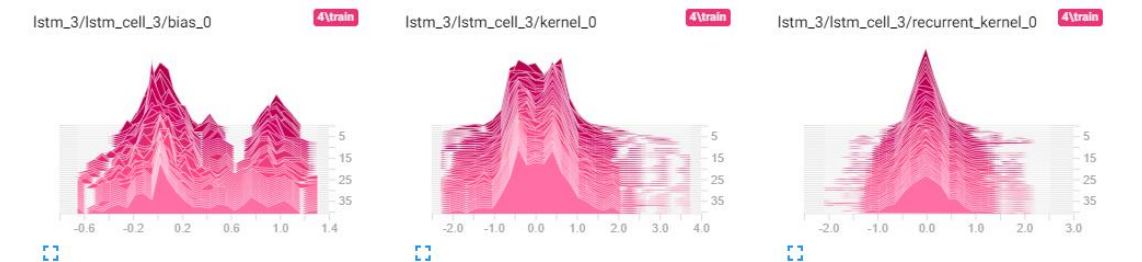
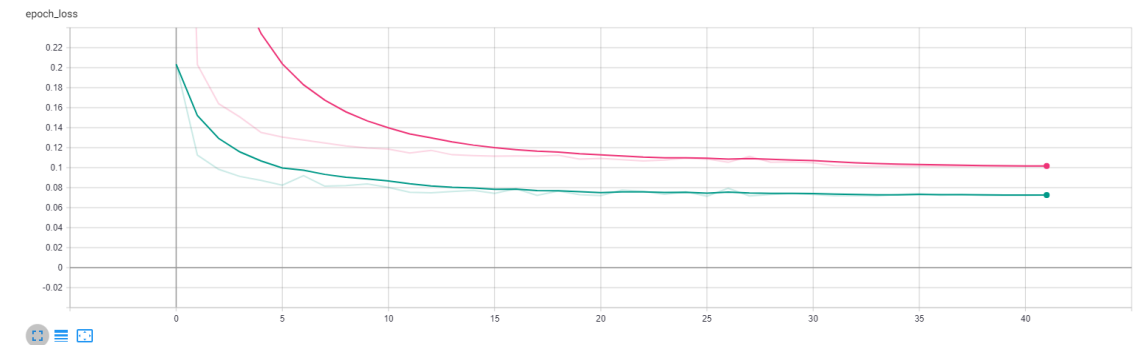
EXPERIMENTE MITTELS TENSORBOARD VISUALISIEREN

Im Rahmen von Experimenten werden hunderte von Modellen trainiert, um die **bestmögliche Kombination** aus Trainingsdaten, Modellarchitektur und Hyperparametern zu finden.

Zur **Visualisierung der Experimente** setzen wir auf **Tensorboard**. Dessen grafische Benutzeroberfläche ermöglicht es u.a.

- die Modellperformance während der einzelnen Trainingsrunden nachzuverfolgen,
- Modellgraphen zu visualisieren sowie
- Hyperparameterkombinationen zu evaluieren und zu protokollieren.

Trial ID	Show Metrics	lstm_num_units	lstm_dropout_rate	shuffle	learning_rate	Validation MSE
0d1e7316c9d188...	☐	5.0000	0.0000	False	0.010000	1.2407
0ff564d267dbd4d...	☐	12.0000	0.0000	False	0.010000	1.1322
148c331593e080...	☐	12.0000	0.0000	False	0.010000	1.1960



Für mehr Transparenz – Ergebnisse messbar machen

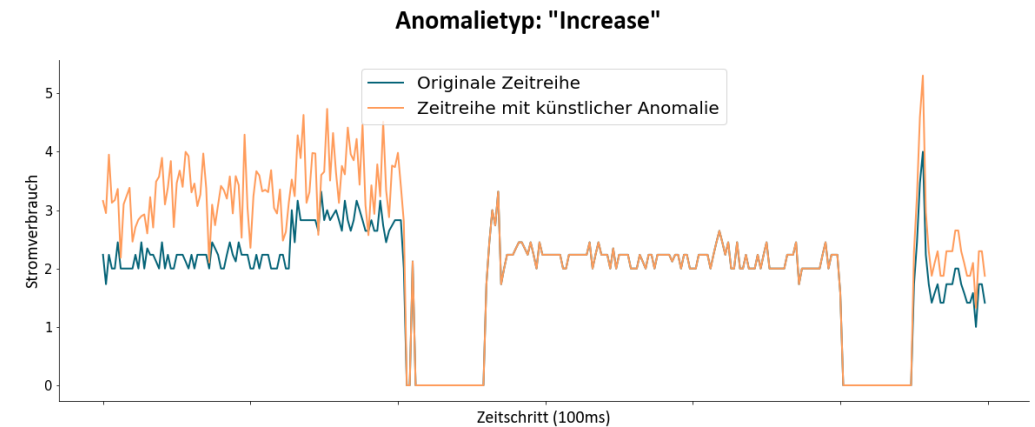


VALIDIERUNG MITTELS KÜNSTLICHER ANOMALIEN

Echte Anomalien sind sehr selten, was in Data-Science-Projekten häufig zu Problemen führt.

Um die Genauigkeit des Algorithmus trotzdem validieren zu können, werden sechs Verlaufsmuster „künstlicher Anomalien“ simuliert.

- Diese werden an zufälligen Stellen in der originalen Zeitreihe eingefügt.
- Die Ausprägungen sind ebenfalls zufällig und können je nach Bedarf individuell konfiguriert werden.
- Jedes Verlaufsmuster stellt **unterschiedliche Herausforderungen** in der Identifizierung dar.



```
def peak(seq, start, end):  
    """Erhöht einen zufälligen Wert um einen zufälligen Faktor"""  
    value = rd.choice(range(start, end), size=1)  
    seq[value] = seq[value] * rd.uniform(1.2, 1.4, 1)  
    return seq  
  
def drop(seq, start, end):  
    """Senkt einen zufälligen Wert um einen zufälligen Faktor"""  
    value = rd.choice(range(start, end), size=1)  
    seq[value] = seq[value] * rd.uniform(0.5, 0.6, 1)  
    return seq  
  
def increase(seq, start, end):
```

> 100.000

Messwerte zur Analyse

5

getestete Deep-Learning-
Algorithmen

6

Verschiedene Anomalie-Typen

> 300

getestete Hyperparameter
Kombinationen



Die wichtigsten Erkenntnisse

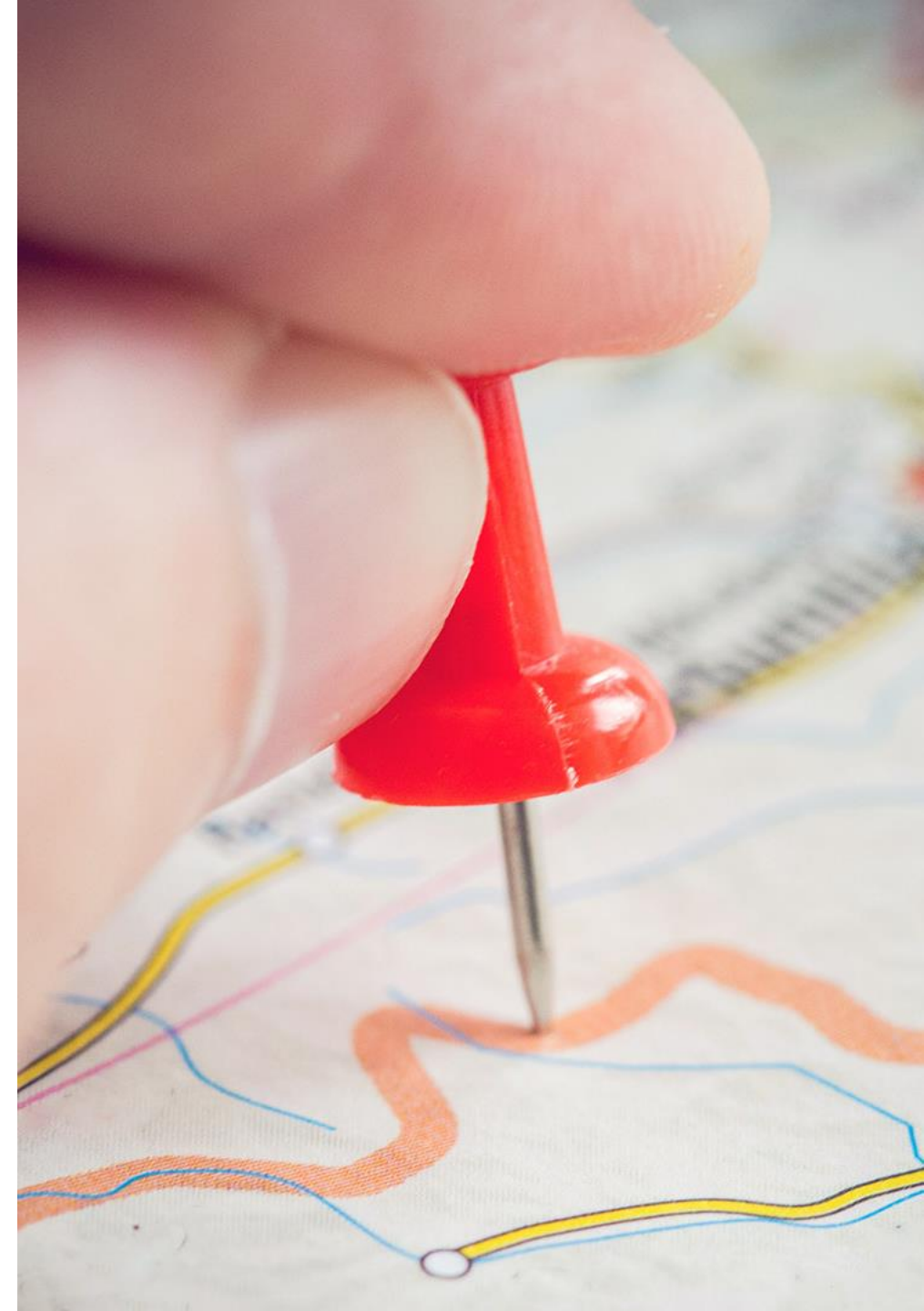
MIT DIESEM WISSEN KÖNNEN WIR SIE IN ZUKUNFT
UNTERSTÜTZEN

Die wichtigsten Erkenntnisse

FÜR SIE AUF DEN PUNKT GEBRACHT

Der Lösungsansatz unseres Showcases ist **direkt nutzbar** und kann mit wenig Aufwand auch **an Ihre Anforderungen angepasst** werden.

- Der Algorithmus erreicht eine **hohe Erkennungsquote** von 90% bei der Identifizierung künstlicher Anomalien im Testdatensatz.
- Auch **real auftretende Anomalien** konnten erkannt werden, deren Seltenheit jedoch eine zuverlässige Evaluierung ausschließt.
- Die **Kombination verschiedener Machine-Learning-Verfahren** (Ensemble Modell) – je nach Beschaffenheit der Anomalie – kann die Treffergenauigkeit noch einmal steigern.





174

Anomalien im Datensatz

90 %

Erkennungsquote

2 %

False-Positive-Rate

0,84

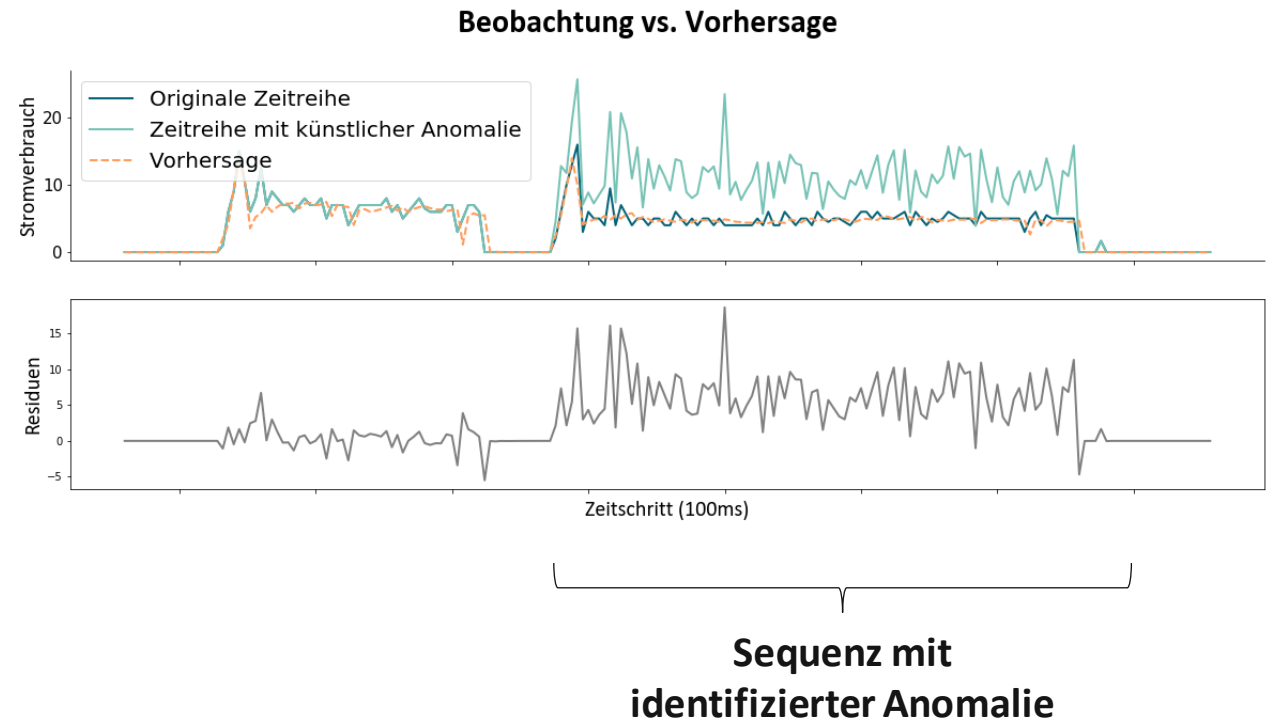
F1 Score

Die wichtigsten Erkenntnisse

ANSCHAUUNGSBEISPIEL #1

Das Anschauungsbeispiel zeigt eine Anomalie im Testdatensatz, die **erfolgreich identifiziert** wurde.

- Bei Betrachtung des vorhergesagten Stromverbrauchs sowie der originalen Zeitreihe, wird die **hohe Genauigkeit des ML-Modells** deutlich.
- Dahingegen entsprechen die Messwerte der Zeitreihe mit der Anomalie nicht dem zu erwartenden Verlauf.
- Dies resultiert in **den großen Residualwerten**, aufgrund derer die Anomalie erkannt wurde.



Die wichtigsten Erkenntnisse

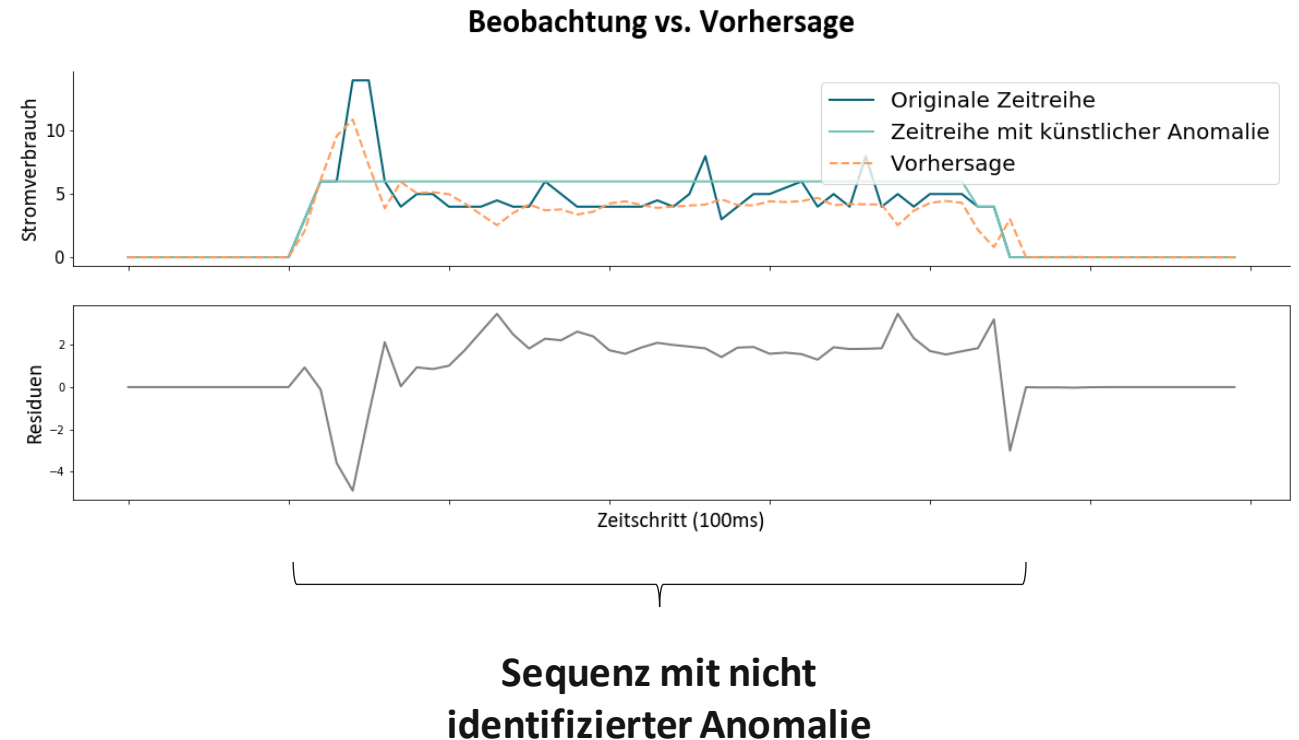
ANSCHAUUNGSBEISPIEL #2

Das zweite Beispiel zeigt eine Anomalie im Testdatensatz, die **nicht identifiziert** werden konnte.

- In diesem Fall sind die **Abweichungen** zwischen der anomalen Zeitreihe und dem vorhergesagtem Verlauf **zu gering**, als dass die Anomalie zuverlässig hätte erkannt werden können.

Künftige Ansatzpunkte zur verbesserten Erkennung von Anomalien könnten darin bestehen,

- ... mehrere ML-Modelle zur Erhöhung der Vorhersagegenauigkeit zu kombinieren.
- ... eine komplexere Logik bei der Auswertung der Residualwerte zu implementieren.



Die Herausforderungen im Showcase

UND WIE WIR SIE GELÖST HABEN

Das Problem:

- Die Aufbereitung des Datensatzes, ist in jedem Data-Science-Projekt essenziell: Schwerpunkte für uns waren die Behandlung unregelmäßiger Messwerte sowie nicht vorhandene Trainingslabels.

Der in der Praxis verbreitete Ansatz, **Anomalien zu simulieren**, hat uns geholfen, das Machine-Learning-Modell unter möglichst realistischen Bedingungen zu evaluieren und zu verbessern.



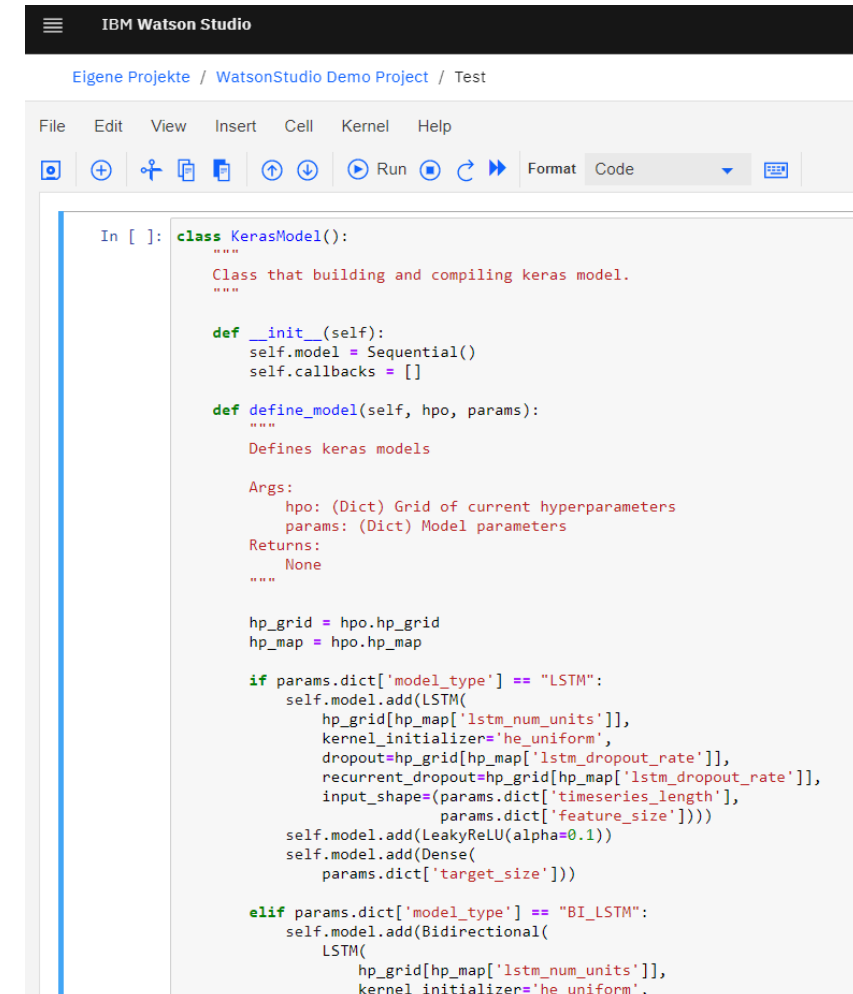
Die Herausforderungen im Showcase

UND WIE WIR SIE GELÖST HABEN

Das Problem:

- Experimente mit hunderten Modellen im Rahmen der Hyperparameteroptimierung können, je nach zur Verfügung stehender Hardware, Stunden oder Tage dauern.

Die **frei skalierbare** Hardwarekonfiguration von [IBM Watson Studio Cloud](#) hat uns geholfen, das Modelltraining drastisch zu beschleunigen.



```
IBM Watson Studio

Eigene Projekte / WatsonStudio Demo Project / Test

File Edit View Insert Cell Kernel Help

[Icons] Run [Icons] Format Code [Dropdown] [Icon]

In [ ]: class KerasModel():
    """
    Class that building and compiling keras model.
    """

    def __init__(self):
        self.model = Sequential()
        self.callbacks = []

    def define_model(self, hpo, params):
        """
        Defines keras models

        Args:
            hpo: (Dict) Grid of current hyperparameters
            params: (Dict) Model parameters
        Returns:
            None
        """

        hp_grid = hpo.hp_grid
        hp_map = hpo.hp_map

        if params.dict['model_type'] == "LSTM":
            self.model.add(LSTM(
                hp_grid[hp_map['lstm_num_units']],
                kernel_initializer='he_uniform',
                dropout=hp_grid[hp_map['lstm_dropout_rate']],
                recurrent_dropout=hp_grid[hp_map['lstm_dropout_rate']],
                input_shape=(params.dict['timeseries_length'],
                             params.dict['feature_size']))
            self.model.add(LeakyReLU(alpha=0.1))
            self.model.add(Dense(
                params.dict['target_size']))

        elif params.dict['model_type'] == "BI_LSTM":
            self.model.add(Bidirectional(
                LSTM(
                    hp_grid[hp_map['lstm_num_units']],
                    kernel_initializer='he_uniform',
```



Fazit

UNSER FAZIT AUS DEM SHOWCASE



„In der automatisierten Anomalie-
Erkennung sehen wir ein großes Potenzial,
das bisher zu selten ausgeschöpft wird.“

[Silvio Bergmann, Senior Manager ISR]

Unser Fazit aus dem Showcase

FÜR SIE GREIFBAR GEMACHT

In unserem Showcase haben wir demonstriert, wie sich Anomalien **mittels moderner Deep-Learning-Modelle** identifizieren lassen:

- Unsere präsentierte Lösung hat sich dabei sehr bewährt und konnte auch schwer zu entdeckende Anomalien erkennen.
- Den **größten Vorteil** sehen wir darin, dass für die zuverlässige Erkennung **im Vorfeld keine gekennzeichneten Anomalien** benötigt werden. So spart man im Vergleich zu anderen Vorgehensweisen einen Großteil des Aufwands.





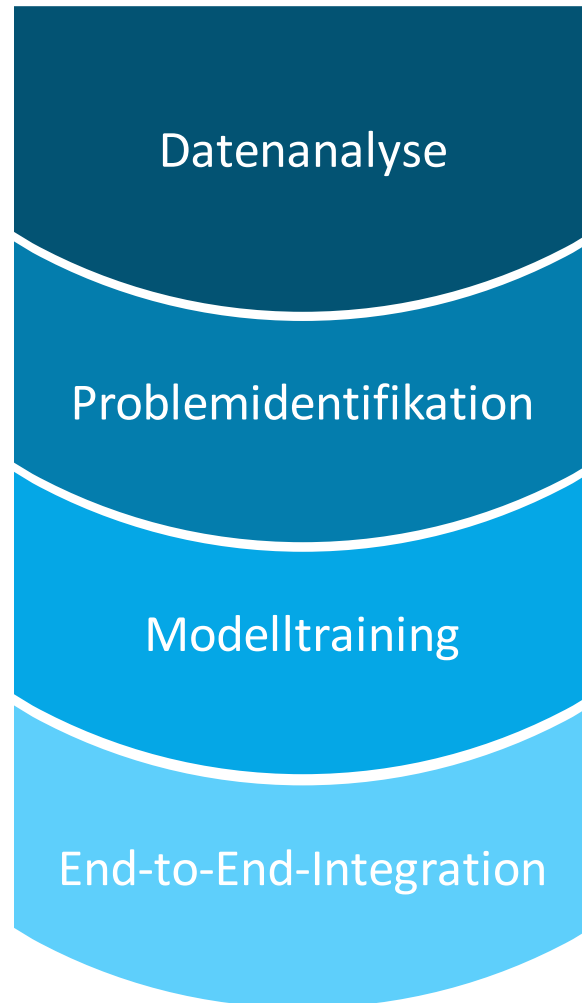
„Die steigenden Datenmengen in Unternehmen sowie die **Analyse dieser Daten in Echtzeit** machen den Einsatz von Machine-Learning-Algorithmen zunehmend attraktiv.

Gerade im **Bereich der Anomalie-Erkennung** können diese Algorithmen sowohl Erkenntnisse zu interessanten Geschäftsvorfällen liefern als auch Sicherheitslücken für Getinge aufdecken.“

Anomalie-Erkennung als Bestandteil Ihres Data-Science-Portfolios



MACHEN SIE UNSERE EXPERTISE ZU IHREM VORTEIL



Decken Sie problematische Anomalien auf, bevor sie Schaden für Ihr Unternehmen anrichten können.

Profitieren Sie von unserer Erfahrung – vom theoretischen Konzept bis hin zur fertigen Implementierung.

Starten Sie nicht bei Null – mit unserem Showcase haben wir ein Framework, das sich auf Ihren Use Case anpassen lässt.

Dieses Team wird Sie begeistern.



Silvio Bergmann

Senior Manager

Neue Burg 2
20457 Hamburg

+49 (0) 151 422 05 383
Silvio.Bergmann@isr.de

Pascal Meißner

Consultant/ Data Scientist

Hintern Brüdern 23
38100 Braunschweig

+49 (0) 151 422 05 412
Pascal.Meissner@isr.de

Wenn Sie mehr zum Thema „Anomalie-Erkennung“ und unserem Showcase erfahren wollen, sprechen Sie uns gerne an!



for your digital smile