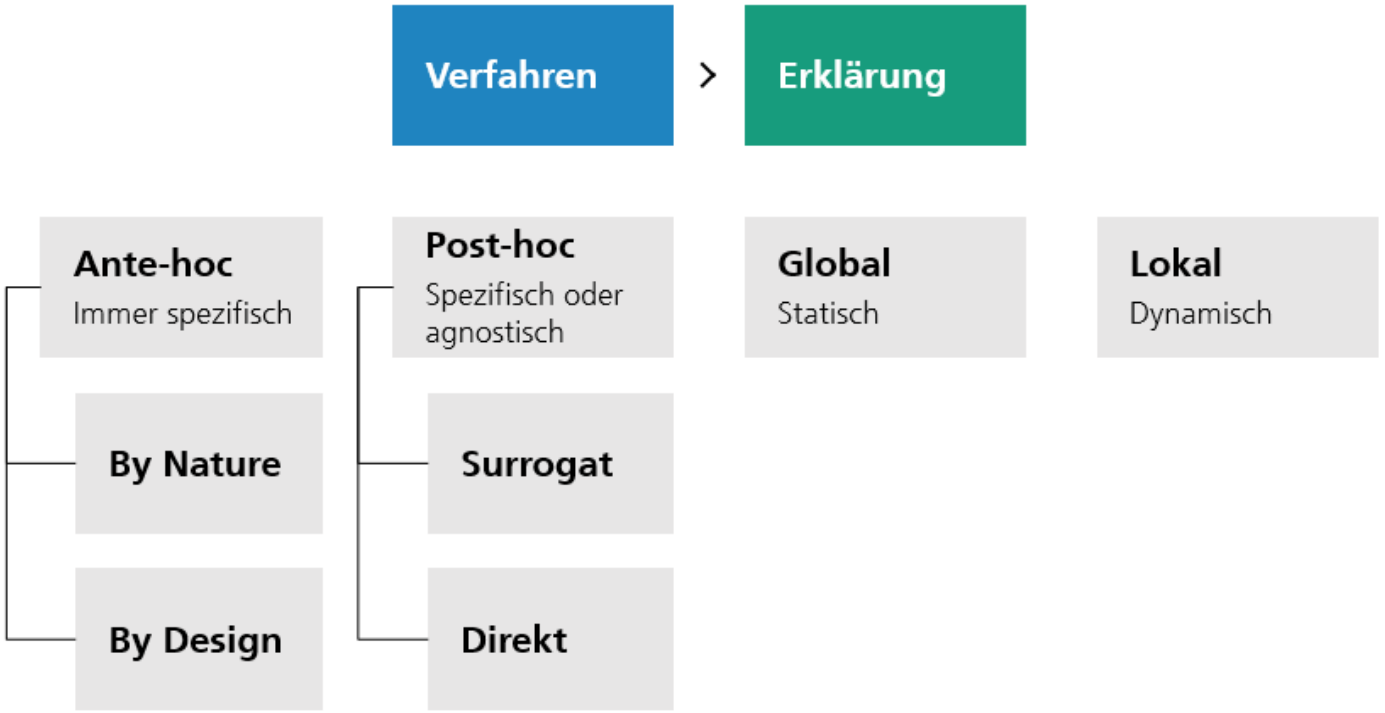
ERKLÄRBARE KÜNSTLICHE INTELLIGENZ

Wiederholung wichtigster Inhalte der letzten Vorlesung

5 Typen von Erklärungen im Bereich überwachtes maschinelles Lernen

Erklärbare KI
XAI Taxonomie



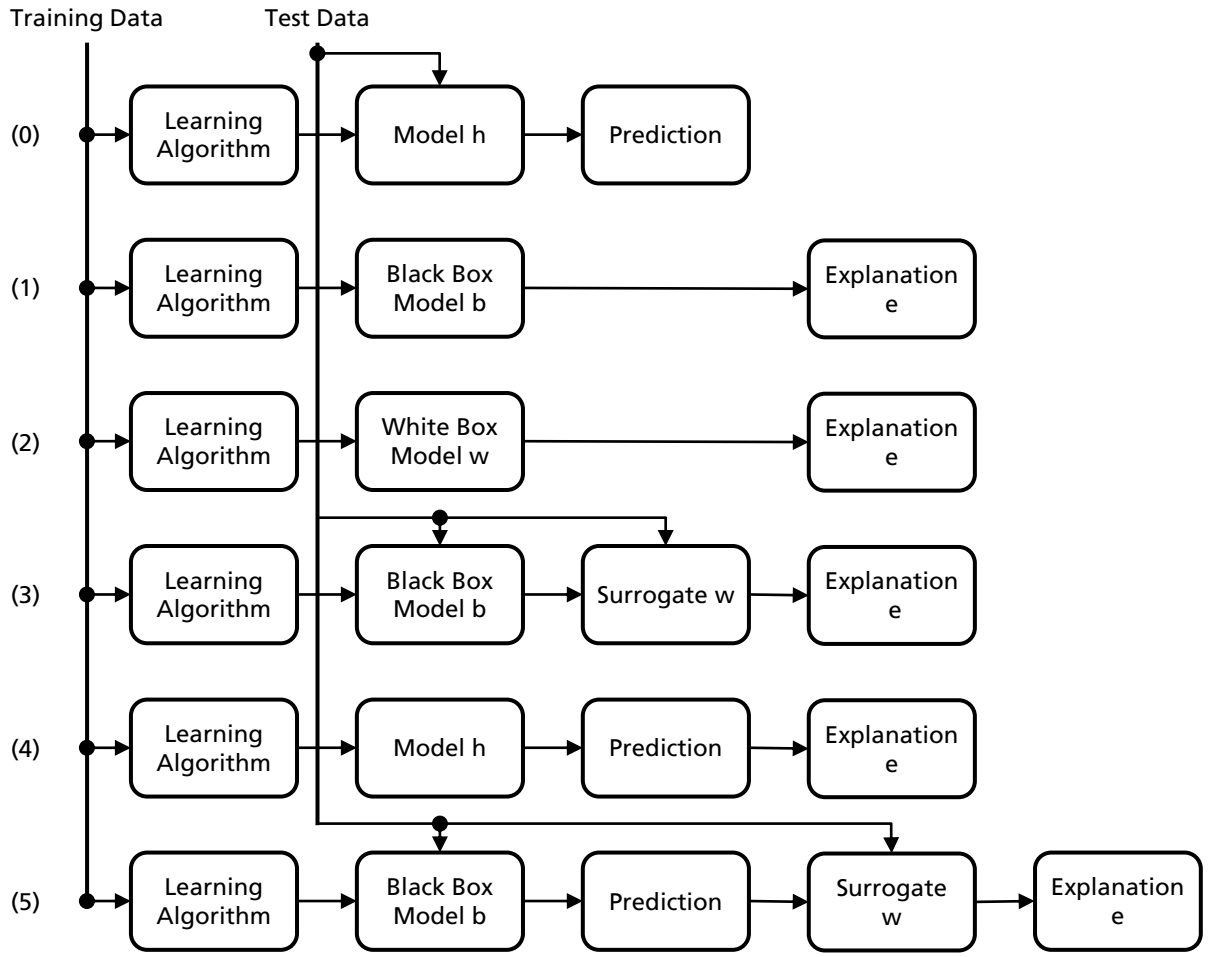
Erklärbare KI

XAI Taxonomie

- **Ante-hoc:** Die Modelle sind für den Anwender vollständig interpretierbar und ohne ein weiteres Zusatzmodell nachvollziehbar
- **Post-hoc:** Im Anschluss an das Training der Black Box werden Erklär-Modelle erzeugt um diese nachzuvollziehen
- **Agnostisch:** Die Erklär-Modelle sind generisch auf unterschiedliche Black-Box-Modelle anwendbar
- **Spezifisch:** Die Modelle sind für spezielle Black-Box-Modelle zugeschnitten z. B. Entscheidungsbäume oder Random Forests
- **Lokal:** Die Erklärungen sind nur für eine spezielle Region oder eine einzelne Instanz nachvollziehbar
- **Global:** Die Erklärungen sind gültig für das gesamte Modell und ein globales Modellverständnis wird erzeugt

Erklärbare KI

Vorgehensmodell - Fünf Typen von Erklärungen[1,2]



→ Überwachtes maschinelles Lernen (Ausgangsproblem)

→ Post-hoc Modell- Erklärungen

→ Interpretierbare Modelle

→ Globale Surrogat-Modelle

→ Instanz-Erklärungen

→ Lokale Surrogat-Modelle

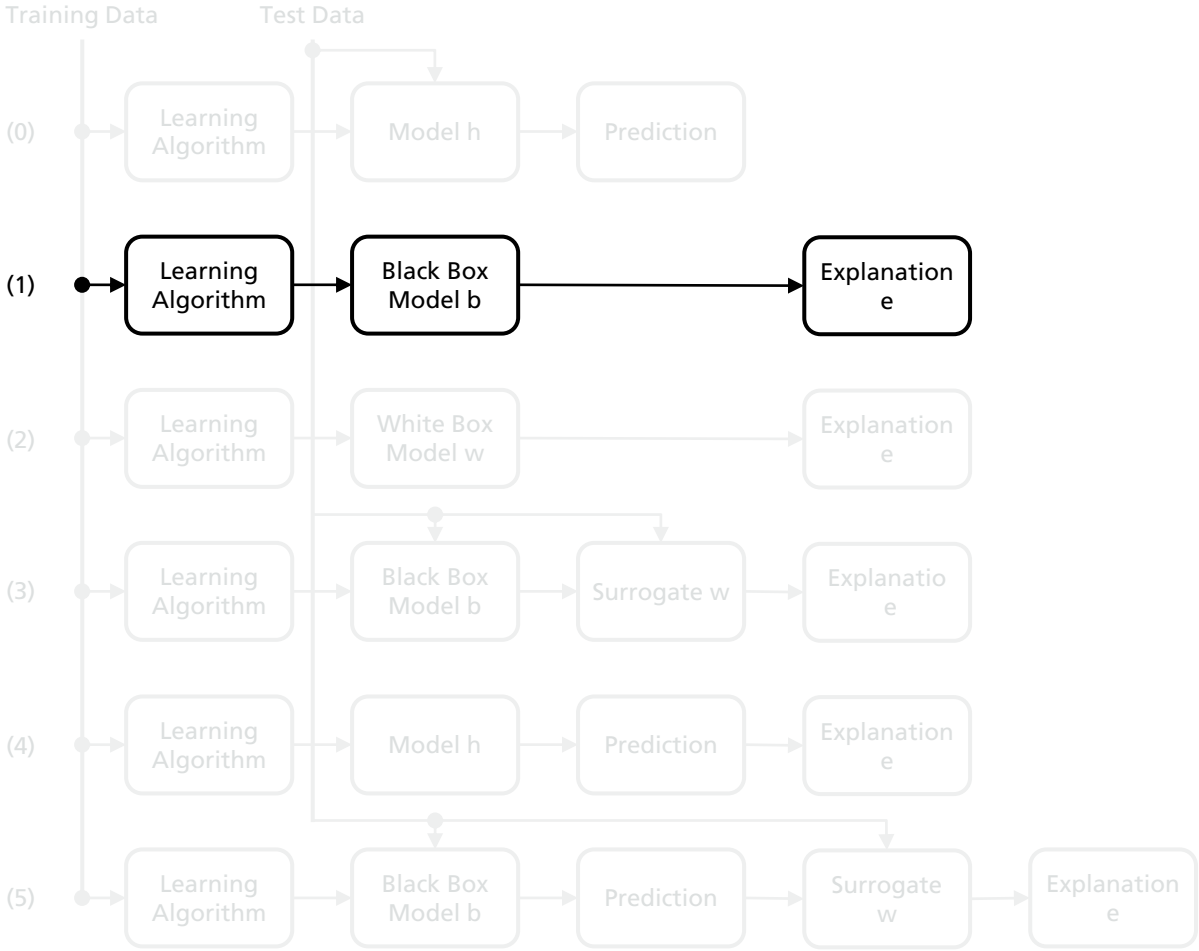
[1] Burkart, Nadia, and Marco F. Huber. "A survey on the explainability of supervised machine learning." *Journal of Artificial Intelligence Research* 70 (2021): 245-317.
[2] Burkart, Nadia, and Marco F. Huber "5 Types of XAI" *Talk at IJCAI 2021*.

Erklärbare KI
Taxonomie der 5 Typen

	VERFAHREN	ERKLÄRUNG
Direkte Modell-Erklärungen	Post-hoc > Direkt	Global
Interpretierbare Modell-Erklärungen	Ante hoc > by design / by nature	Global
Globale Surrogat-Erklärungen	Post-hoc > Surrogat	Global
Direkte Instanz-Erklärungen	Post-hoc > Direkt	Lokal
Lokale Surrogat-Erklärungen	Post-hoc > Surrogat	Lokal

Erklärbare KI

Post-hoc Modell-Erklärungen



Post-hoc Modell-Erklärungen

Erklärbare KI

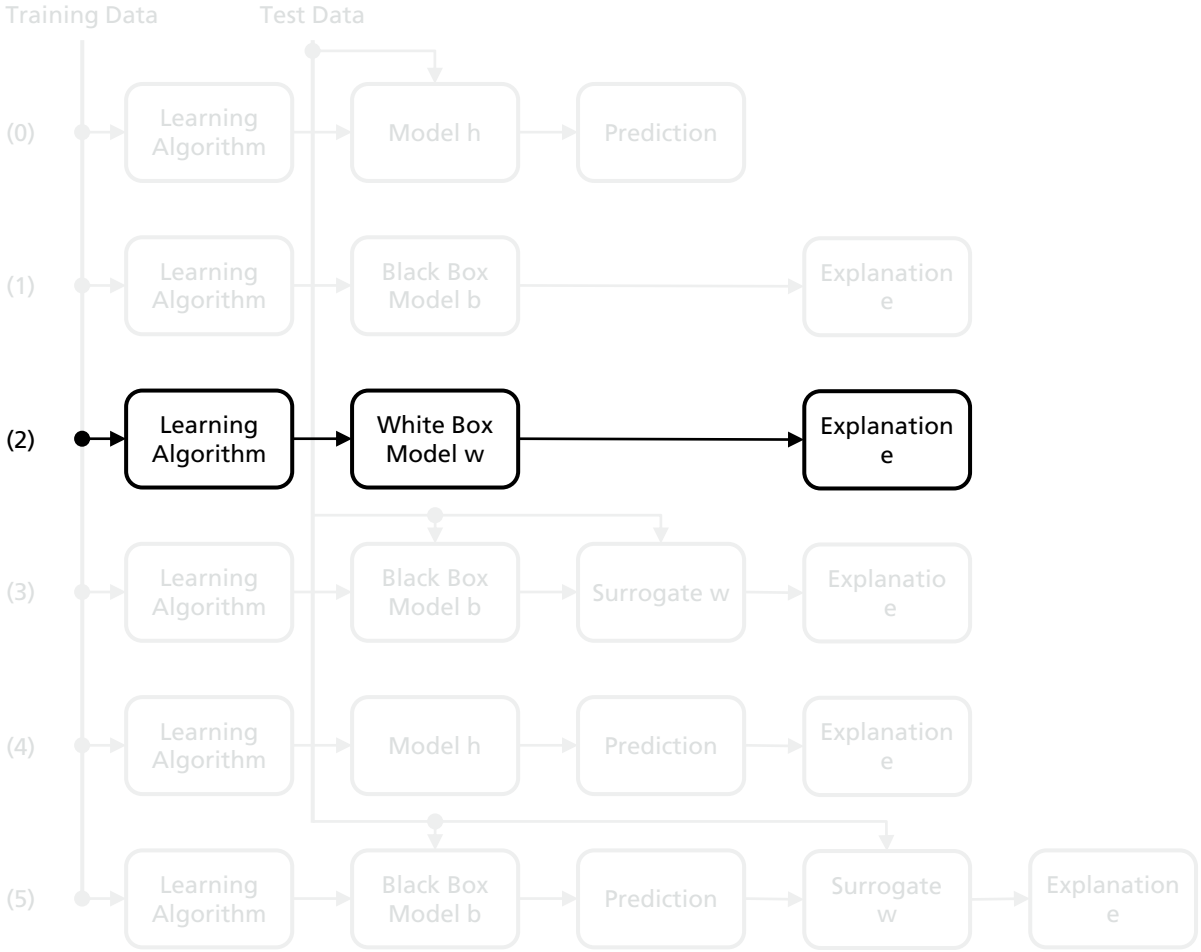
Beispiele für post-hoc Modell-Erklärungen

Feature	Importance
sepal length (cm)	0.47174951
sepal width (cm)	0.40272703
petal length (cm)	0.10099771
petal width (cm)	0.02452574

Tabelle: Merkmalswichtigkeit eines Random Forest

Erklärbare KI

Interpretierbare Modelle



Interpretierbare Modelle

Erklärbare KI
Beispiel für interpretierbare Modelle

Interpretable by Nature

Setosa
Versicolor
Virginica

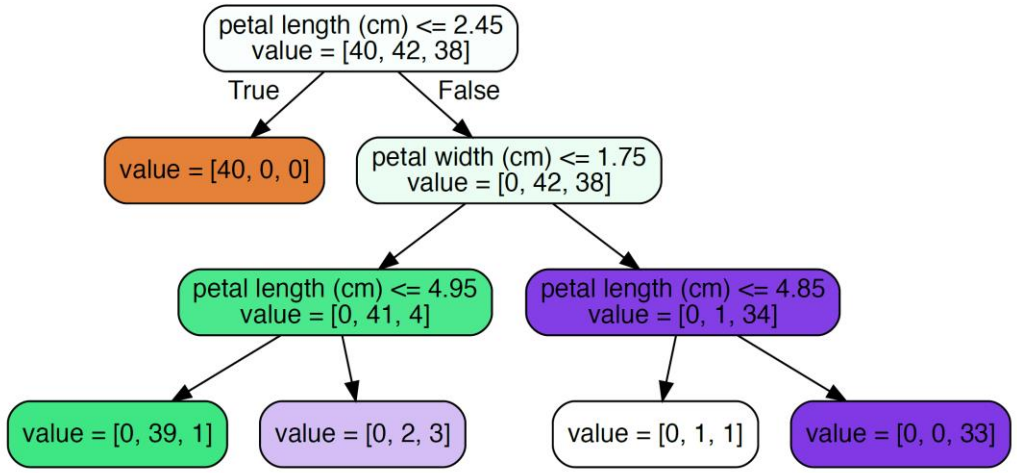


Abbildung: Einfacher Entscheidungsbaum der Tiefe 3

Interpretable By Design

Virginica Classification

```
IF petal length (cm) : 5.15 to inf THEN probability of virginica: 96.6%
ELSE IF petal length (cm) : -inf to 4.75 THEN probability of virginica: 2.6%
ELSE IF petal width (cm) : -inf to 1.75 THEN probability of virginica: 25.0%
```

Setosa Classification

```
IF petal width (cm) : 0.8 to inf THEN probability of setosa: 1.2%
ELSE IF petal length (cm) : -inf to 2.45 THEN probability of setosa: 97.4%
ELSE probability of setosa: 50.0%
```

Versicolor Classification

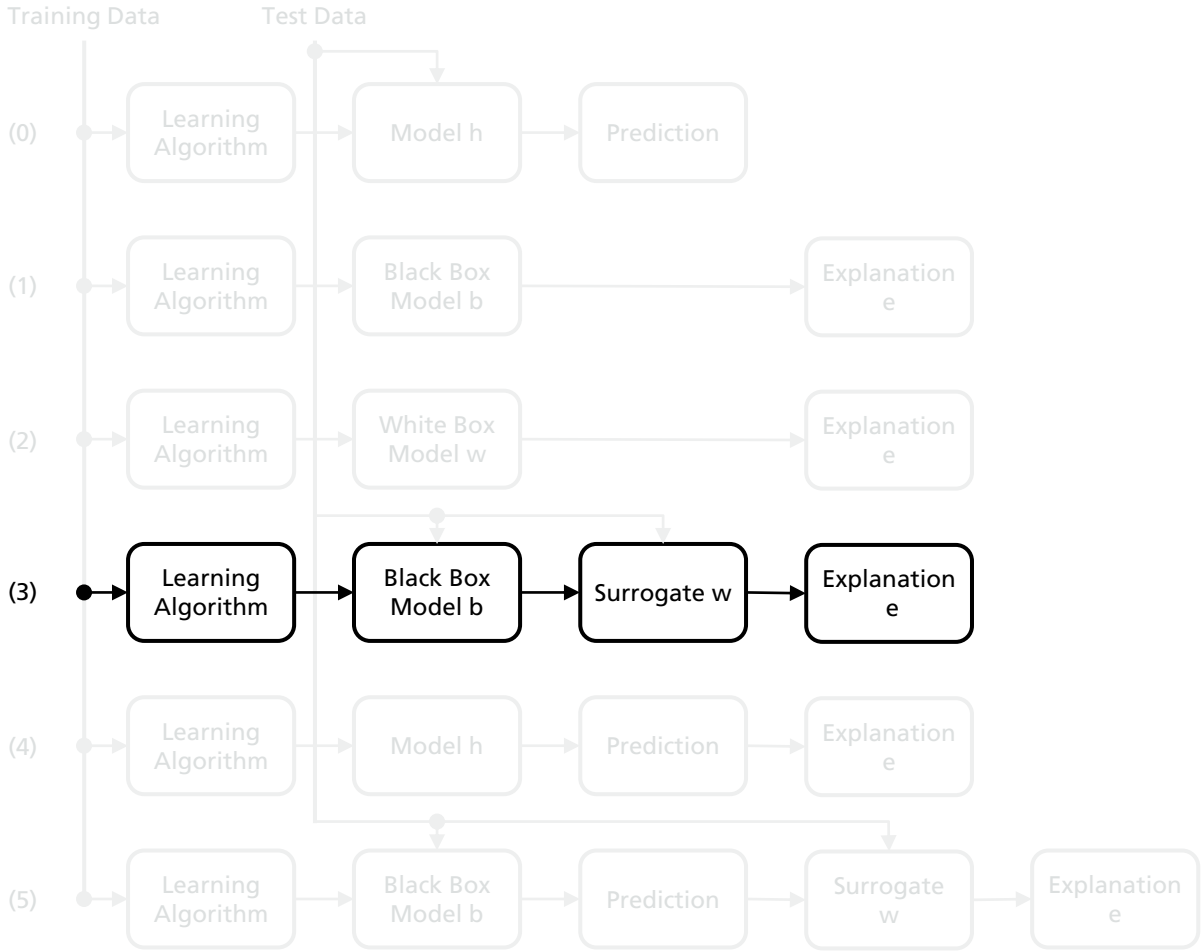
```
IF petal length (cm) : 2.45 to 4.75 THEN probability of versicolor: 97.3%
ELSE IF petal width (cm) : 0.8 to 1.7 THEN probability of versicolor: 42.9%
ELSE probability of versicolor: 2.4%
```

Listing: Bayesian Rule List[1]

[1]Wang, Tong, et al. "A bayesian framework for learning rule sets for interpretable classification." *The Journal of Machine Learning Research* 18.1 (2017): 2357-2393.

Erklärbare KI

Globale Surrogat-Modelle



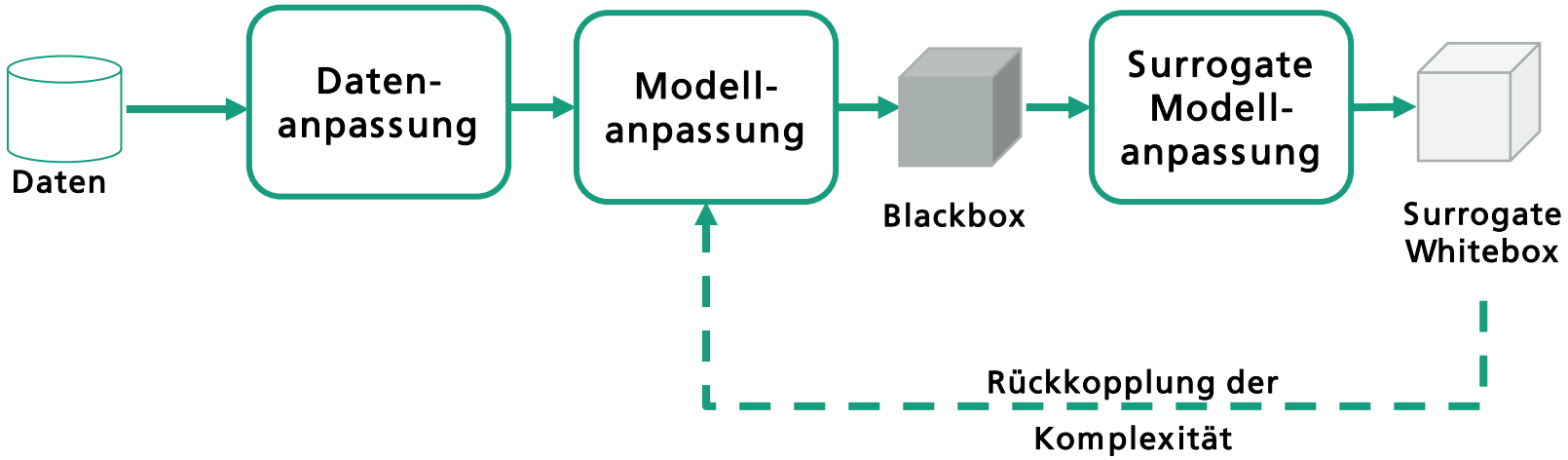
Globale Surrogat-Modelle

Erklärbare KI

Globale Surrogat-Modelle[1,2]

Vorgehensweise

- Aufbau regelbasierter Modelle als Surrogate für das neurale Netz (NN) durch Einsatz der Regularisierung
- Rückführung der Komplexität dieses Surrogats als Strafterm in das Training des NN-Modell



[1] Burkart, Nadia, Marco Huber, and Phillip Faller. "Forcing interpretability for deep neural networks through rule-based regularization." *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. IEEE, 2019.
[2] Burkart, Nadia, et al. "Batch-wise Regularization of Deep Neural Networks for Interpretability." *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE, 2020.

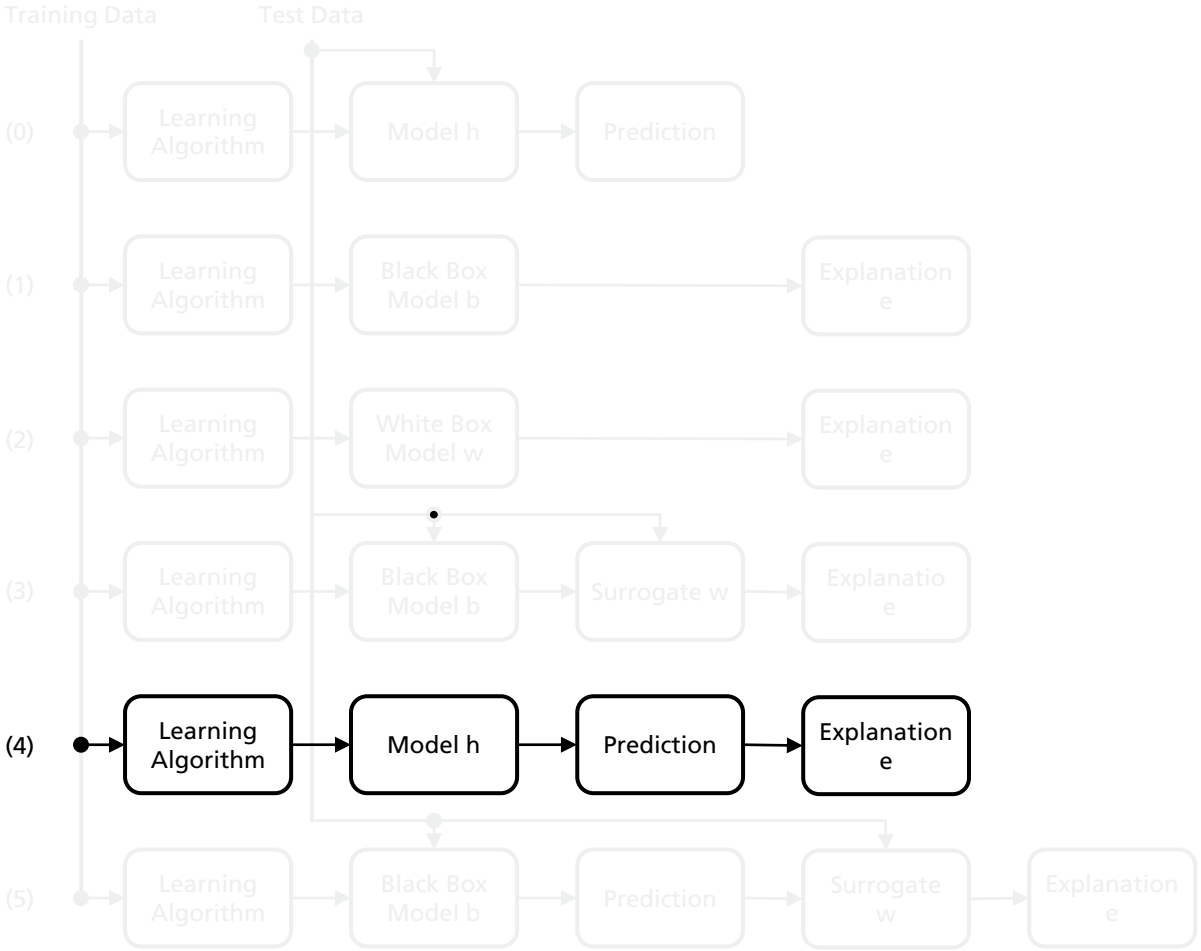
Erklärbare KI

Globale Surrogat-Modelle - Ergebnisse

IF (ExternalRiskEstimate in (57.4, 69.6]) **AND** (NetFractionRevolvingBurden in (46.4, 92.8]) **THEN** Target = Bad
IF (ExternalRiskEstimate in (81.8, 94.0]) **AND** (MaxDelqEver in 6.0) **THEN** Target = Good
IF (NetFractionRevolvingBurden in (46.4, 92.8]) **AND** (PercentTradesWBalance in (80.0, 100.0]) **THEN** Target = Bad
IF (MSinceOldestTradeOpen in (162.2, 322.4]) **AND** (PercentTradesWBalance in (40.0, 60.0]) **THEN** Target = Good
IF (NetFractionRevolvingBurden in (-0.232, 46.4]) **AND** (PercentTradesWBalance in (20.0, 40.0]) **THEN** Target = Good
IF (MSinceOldestTradeOpen in (322.4, 482.6]) **AND** (PercentTradesNeverDelq in (80.0, 100.0]) **THEN** Target = Good
IF (PercentTradesWBalance in (80.0, 100.0]) **AND** (AverageMInFile in (3.621, 79.8]) **THEN** Target = Bad
IF (ExternalRiskEstimate in (81.8, 94.0]) **AND** (NumSatisfactoryTrades in (-0.079, 15.8]) **THEN** Target = Good
IF (NumTradesOpeninLast12M in (-0.019, 3.8]) **AND** (NetFractionRevolvingBurden in (46.4, 92.8]) **THEN** Target = Bad
IF (ExternalRiskEstimate in (69.6, 81.8]) **AND** (NumTrades90Ever2DerogPubRec in (-0.162, 3.686]) **THEN** Target = Bad
IF (PercentInstallTrades in (-0.1, 20.0]) **AND** (PercentTradesWBalance in (40.0, 60.0]) **THEN** Target = Bad
IF (MSinceMostRecentInqexcl7days in (-0.35, 4.54]) **AND** (AverageMInFile in (3.621, 79.8]) **THEN** Target = Bad
IF (MSinceOldestTradeOpen in (322.4, 482.6]) **THEN** Target = Good
IF (ExternalRiskEstimate in (81.8, 94.0]) **AND** (NetFractionRevolvingBurden in (-0.232, 46.4]) **THEN** Target = Good
IF (ExternalRiskEstimate in (69.6, 81.8]) **AND** (PercentTradesWBalance in (20.0, 40.0]) **THEN** Target = Bad
IF TRUE **THEN** Target = Bad

Erklärbare KI

Instanz-Erklärungen



Instanz-Erklärungen

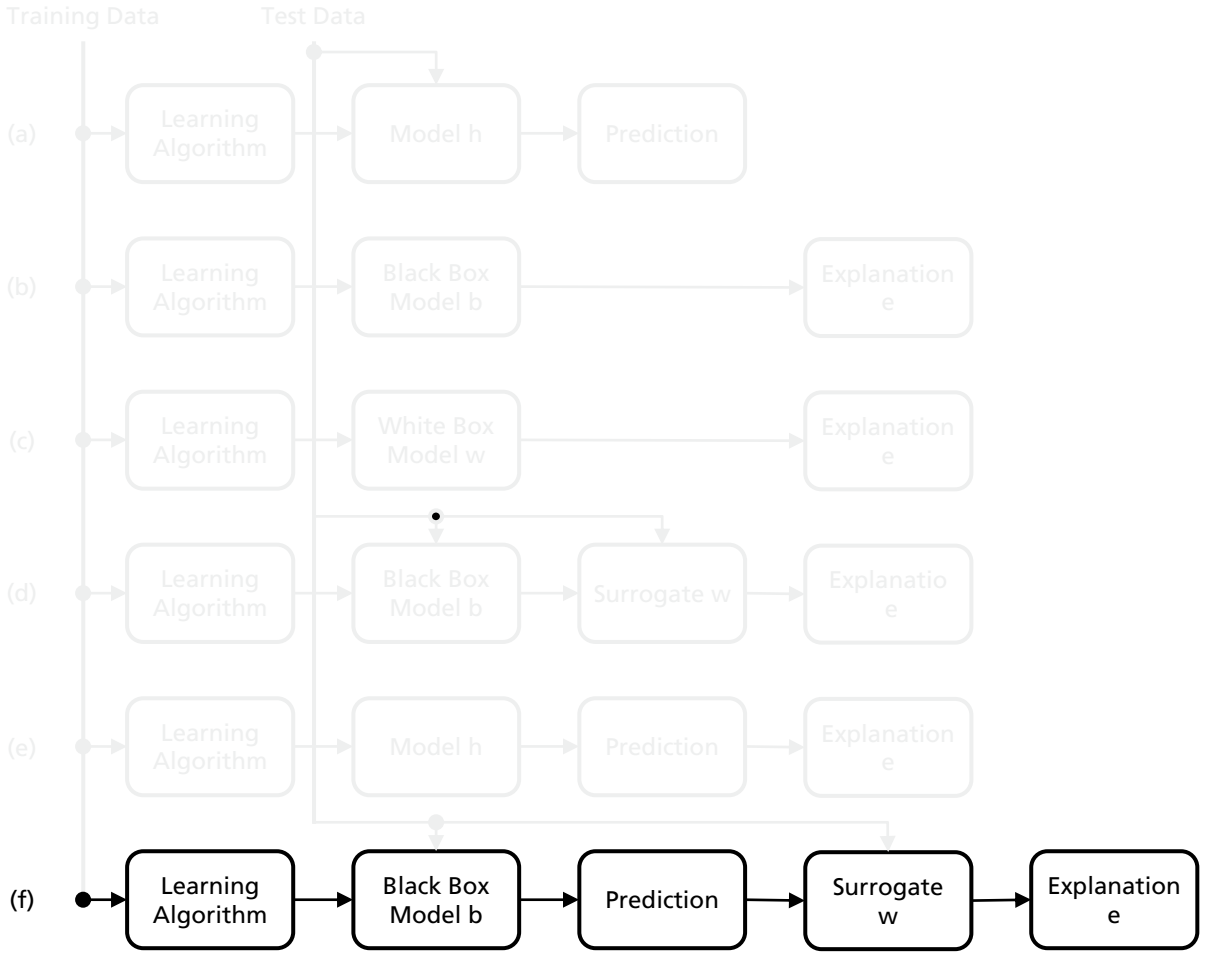
Erklärbare KI
Instanz-Erklärung

- Grad-CAM-Verfahren[1] zur Erzeugung von Aktivierungskarten (Class Activation Maps (CAM)) zur Klassifizierung von Fahrzeugmarken
- Eingabe: Klassifizierungsergebnis und neuronales Netz
- Ausgabe: Bildregionen mit signifikantem Beitrag zur Klassifizierung
- CAMs werden durch Gradientenabstieg ermittelt
- Mehrwert für Endanwender, sowie Entwickler durch Analyse des Netzfokuses

[1] Selvaraju, Ramprasaath R., et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization." Proceedings of the IEEE international conference on computer vision. 2017.

Erklärbare KI

Lokale Surrogat-Modelle



Lokales Surrogat-Modelle

Erklärbare KI
Lokales Surrogat-Modell

- LIME[1] erzeugt Merkmalswichtigkeit für bestimmte Instanz
- Erklärungen werden lokal oder für jede Instanz unabhängig gefunden
- Ein einfaches Modell wird lokal an die Vorhersagen des komplexen Modells angepasst
- Erklärungen werden auf der Grundlage der ursprünglichen Instanz gegeben

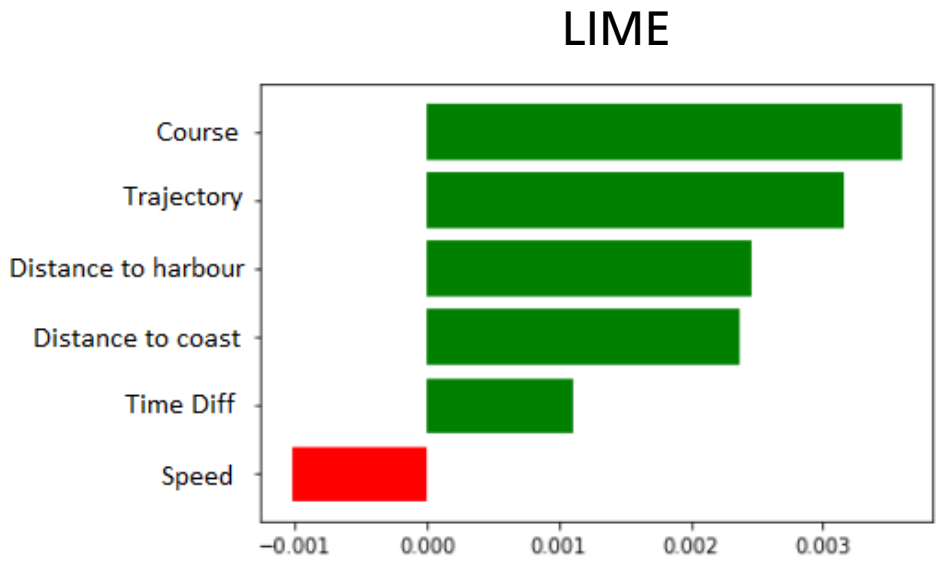
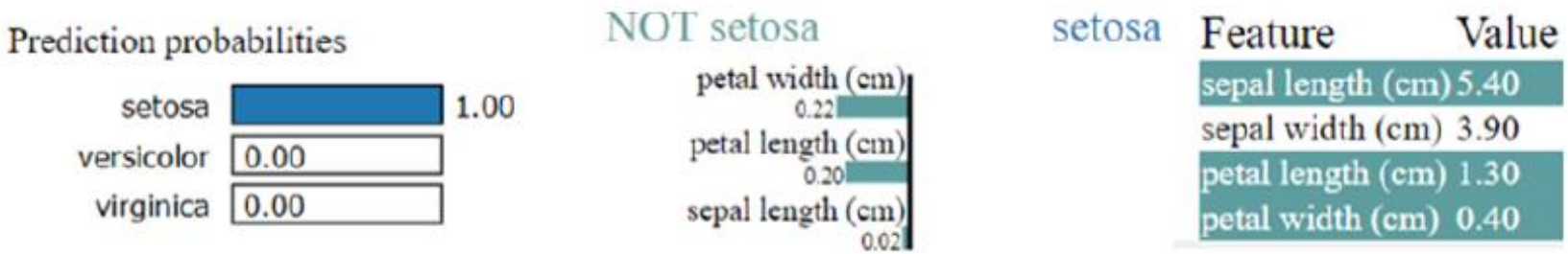
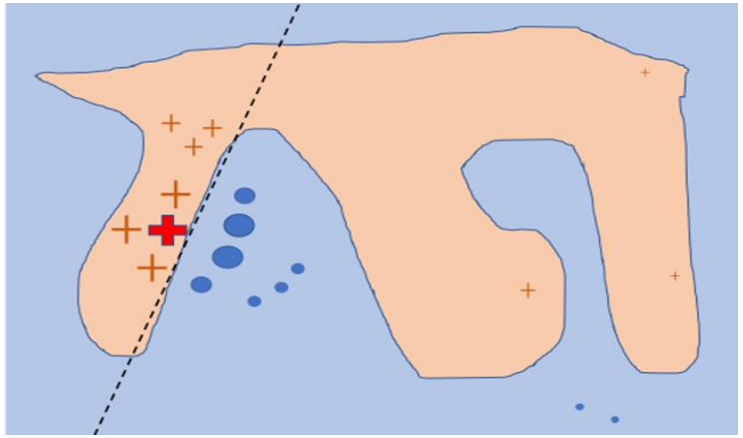


Abbildung: Ergebnis von LIME für eine Instanz mit der Vorhersage Cargo-Tanker

[1]Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why should i trust you?" Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. 2016.


Prediction probabilities
setosa
versicolor
virginica
1.00
0.00
0.00
NOT setosa
petal width (cm)
0.22
petal length (cm)
0.20
sepal length (cm)
0.02
setosa
Feature
Value
sepal length (cm) 5.40
sepal width (cm) 3.90
petal length (cm) 1.30
petal width (cm) 0.40

Vorgehen zu LIME

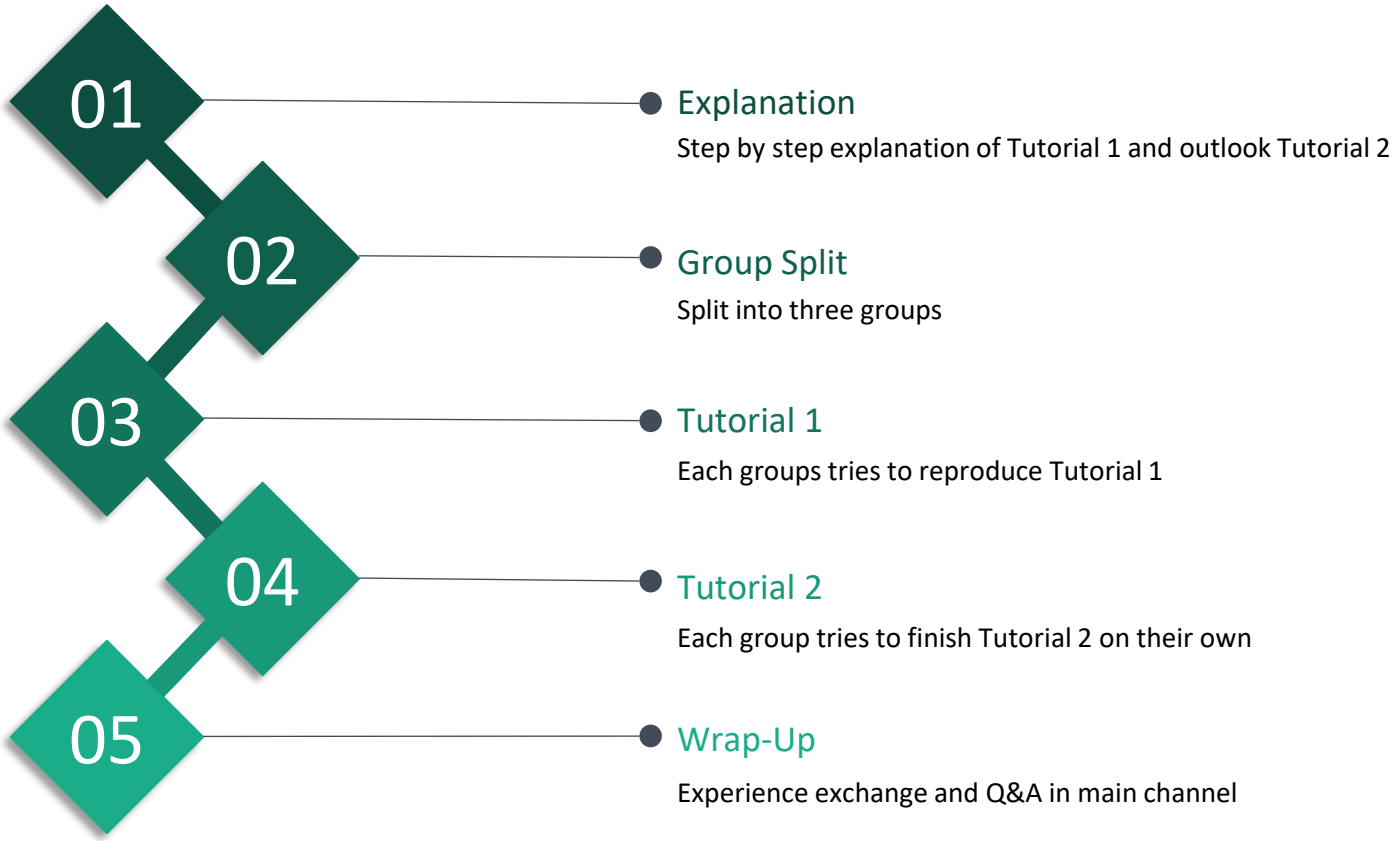


•**Vorhersage des Modells:**
•Du hast ein Modell, das eine Vorhersage für einen bestimmten Datenpunkt macht (z.B. „Hund“ oder „Katze“ für ein Bild oder „Kredit bewilligen“ für eine Anfrage).
•**Erstellung leicht veränderter Versionen:**
•LIME erzeugt viele leicht veränderte Versionen des ursprünglichen Datenpunkts. Diese Versionen sind ähnlich, aber mit kleinen Abweichungen (z.B. werden bei einem Bild bestimmte Pixel zufällig verändert oder bei Text einige Wörter entfernt).
•**Modellvorhersagen auf den veränderten Datenpunkten:**
•Das Modell trifft Vorhersagen für all diese veränderten Datenpunkte.
•**Erstellung eines einfachen Modells (z.B. lineares Modell):**
•LIME baut ein **einfaches, erklärbares Modell** (meist ein lineares Modell) nur für die Umgebung des ursprünglichen Datenpunkts. Es berücksichtigt, wie das ursprüngliche Modell auf die veränderten Datenpunkte reagiert hat.
•**Identifizierung wichtiger Merkmale:**
•Das einfache Modell zeigt, welche Merkmale (z.B. bestimmte Pixel oder Wörter) für die Vorhersage am wichtigsten sind und welche das Ergebnis beeinflusst haben.
•**Erklärung der Vorhersage:**
•LIME präsentiert eine leicht verständliche Erklärung, welche Merkmale für die Entscheidung des Modells bei diesem speziellen Datenpunkt eine wichtige Rolle gespielt haben.

- Exkurs Video zu Lime: https://www.youtube.com/watch?v=hUnRCxnydCc

Tutorials

Erklärbare KI
Tutorials - Schedule

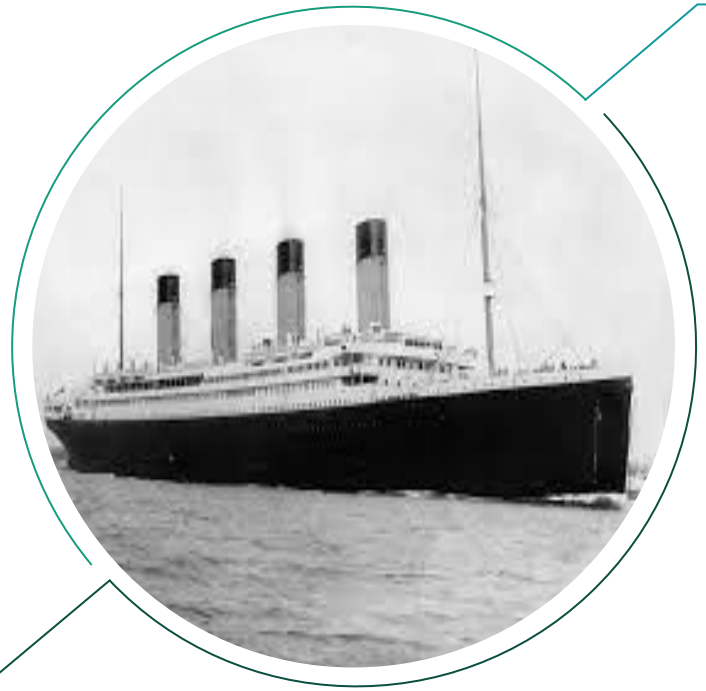


Erklärbare KI

Tutorial 1: Predict the survival of Titanic passengers

Data

Use the given passenger data (e.g. name, age, gender, socio-economic class, etc).



Build a predictive model that answers the question: “which passenger would survive?” and WHY with LIME

The Challenge

Erklärbare KI

Tutorial 2: Predict the quality of wine

Data

Use the given wine data (e.g. fixed acidity, volatile acidity, citric acid, residual sugar, chlorides).

Build a predictive model that answers the question: “how good is the wine”? And Why with LIME

The Challenge

DeepDive SHAP

SHAP

- *SHAP (SHapley Additive ExPlanations) ist ein spieltheoretischer Ansatz, um die Ergebnisse jedes maschinellen Lernmodells zu erklären*
- *Es verbindet eine Vorhersage mit einer lokalen Erläuterungen mithilfe der klassischen Shapley - Werten aus der Spieltheorie und ihren zugehörigen Erweiterungen können*
- Shapley-Werte
  - Wie oben erwähnt, basieren Shapley-Werte auf der klassischen Spieltheorie.
  - Es gibt viele Spieltypen wie kooperativ /nicht kooperativ, symmetrisch / nicht symmetrisch, Nullsumme / Nicht-Nullsumme usw.
  - Die Shapley-Werte basieren jedoch auf der kooperativen (Koalitions-) Spieltheorie.
- In der Koalitionsspieltheorie kommt eine Gruppe von Spielern zusammen, um einen Wert zu schaffen.
- Sie können sich eine Gruppe von Menschen vorstellen, die zusammenkommen, um ein Unternehmen zu gründen und Gewinn zu erzielen.
- Der Shapley-Wert ist eine Methode, um diesen Gewinn unter den Spielern zu verteilen.
  - Ziel: Gewinn fair auf die Spieler verteilen, basierend auf ihrem Beitrag.

SHAP

- Annahme: Modell trainiert, um Immobilienpreise vorherzusagen.
- Modell prognostiziert Immobilienpreis von 100.000 USD.
  - Die Größe des Hauses beträgt 2400 Quadratmeter mit 3 Schlafzimmern.
- Ziel ist es nun, diese Vorhersage zu erklären.
- Annahme: 85.000 USD durchschnittliche Hauspreis für den gegebenen Datensatz
  - Shapley-Werte erklären, inwieweit jeder Merkmalswert im Vergleich zur durchschnittlichen Vorhersage zur Vorhersage beigetragen hat.
- In dem obigen Beispiel gibt es die Merkmale Grösse, Umgebung, Stadt, Anzahl Schlafzimmer Schlafzimmer um zusammen die Prognose von 100.000 $ zu erreichen.
- Ziel ist es, den Unterschied zwischen der tatsächlichen Vorhersage (100.000 USD) und der durchschnittlichen Vorhersage (85.000 USD) zu erklären: einen Unterschied von 15.000 USD.
- Eine mögliche Erklärung könnte die Größe, Area, Stadt 25.000 US-Dollar, 25.000 US-Dollar und Schlafzimmer 15.000 US-Dollar sein.
- Die Beiträge summieren sich auf 15.000 USD - dies ist nichts anderes als die endgültige Prognose abzüglich des durchschnittlichen prognostizierten Immobilienpreises.

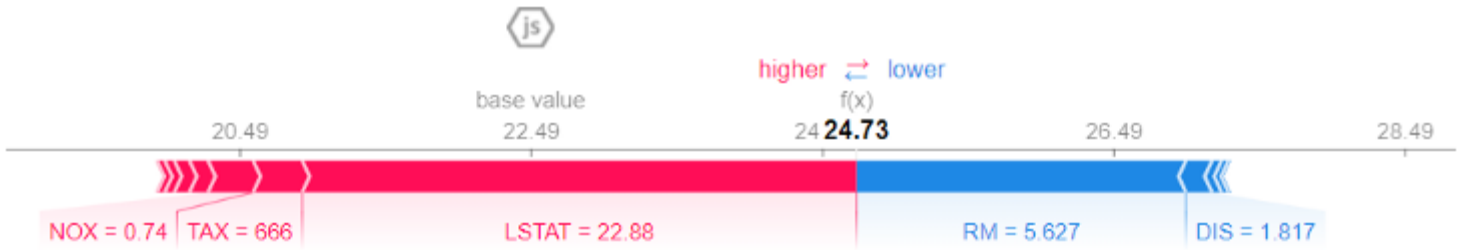
SHAP

- Schritt 1: Im ersten Schritt ziehen wir zufällig eine Instanz aus den Daten und berechnen den Hauspreis ohne Verwendung des Merkmals Schlafzimmer.
  - Wir können diese Aktivität mehrmals durchführen und den Durchschnitt ermitteln.
- Schritt 2: Im zweiten Schritt nehmen wir den Durchschnitt der Hauspreise , wenn alle Merkmale verwendet werden.
  - Auch hier können wir Abtastschritte mehrmals durchführen und den Durchschnitt ermitteln.
- Der SHAP-Wert basiert auf Shapley-Werten.
  - Inspiriert von verschiedenen Methoden schlugen die Autoren von SHAP einen einheitlichen Ansatz zur Interpretation von Modellvorhersagen vor.
  - Das Kernkonzept der Shapley-Werte bei der Berechnung des SHAP-Werts ist immer dasselbe!

Arten SHAP

- **TreeExplainer** - ein Algorithmus zum Berechnen von SHAP-Werten für Bäume und Baumensembles.
- **DeepExplainer** - ein Algorithmus zur Berechnung von SHAP-Werten für Deep-Learning-Modelle, die auf Verbindungen zwischen SHAP und dem DeepLIFT-Algorithmus basieren.
- **GradientExplainer** - eine Implementierung erwarteter Gradienten zur Annäherung an SHAP-Werte für Deep-Learning-Modelle.
- **LinearExplainer** - ein Algorithmus für ein lineares Modell mit unabhängigen Funktionen zur Berechnung der genauen SHAP-Werte.
- **KernelExplainer** - eine modellunabhängige Methode zum Schätzen von SHAP-Werten für jedes Modell.

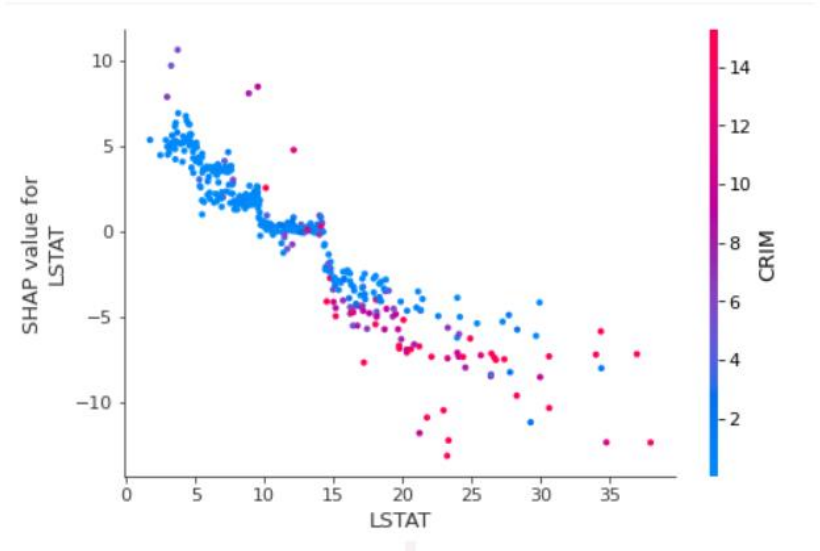
Visualisierung von SHAP



- base value
  - ist die mittlere Vorhersage über den gesamten Testdatensatz. Dies ist der Wert, der vorhergesagt werden würde, wenn wir keine Funktionen für die aktuelle Ausgabe kennen würden.
- Merkmale, die die Vorhersage nach oben drücken, werden rot angezeigt, und diejenigen, die die Vorhersage nach unten drücken, werden blau angezeigt.
- LSTAT Feature hat einen hohen positiven Einfluss auf den Immobilienpreis und drückt die Vorhersage nach rechts.
- Die anderen wichtigen Merkmale, die den Immobilienpreis auf einen höheren Wert treiben, sind TAX& NOX.
- RM hat einen hohen negativen Einfluss auf die Immobilienpreise, gefolgt von DISFeature.

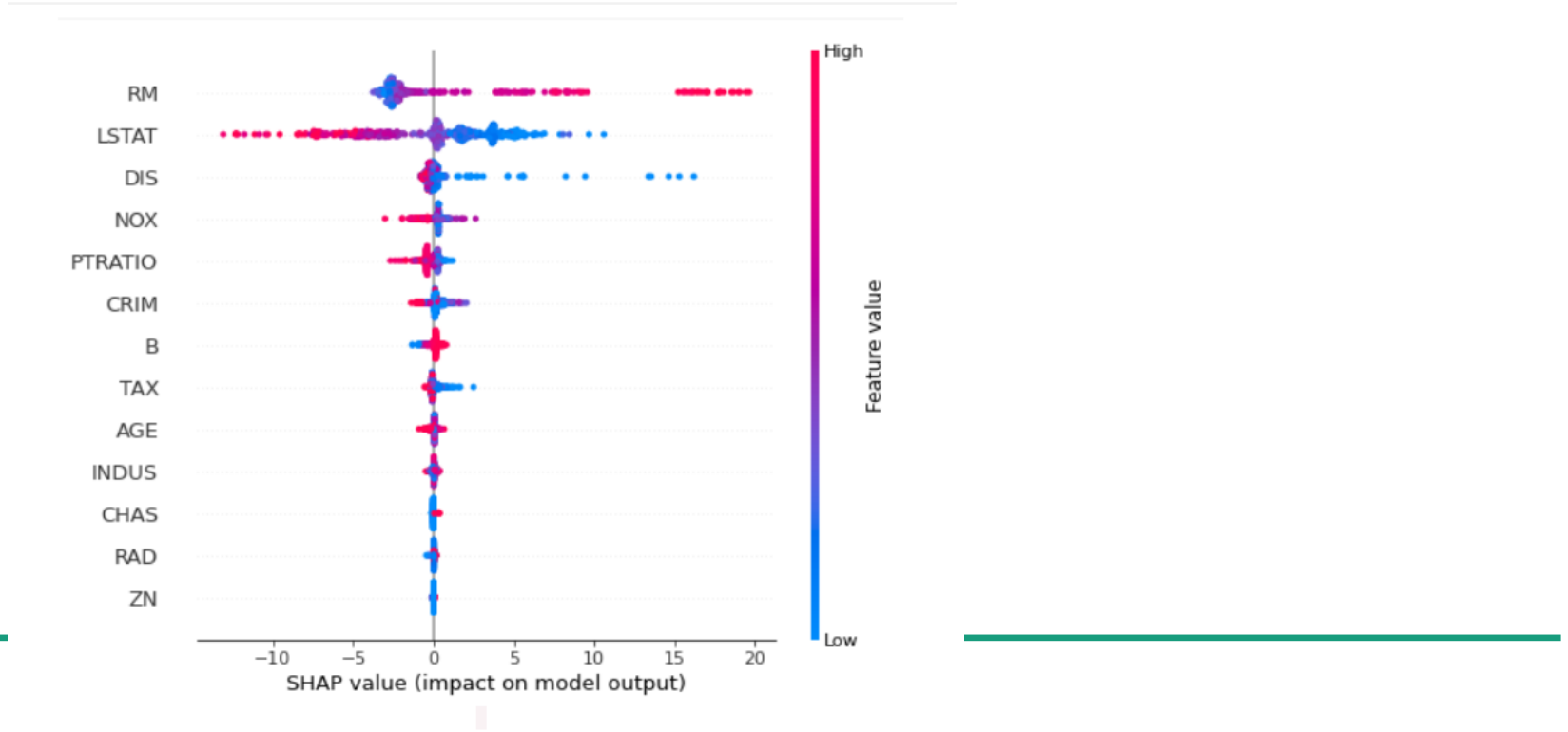
Abhängigkeitsdiagramm SHAP

- Um zu verstehen, wie sich ein einzelnes Feature auf die Ausgabe des Modells auswirkt, kann ein Abhängigkeitsdiagramm verwendet werden.
- Das partielle Abhängigkeitsdiagramm zeigt die marginale Auswirkung, die ein oder zwei Merkmale auf das vorhergesagte Ergebnis eines maschinellen Lernmodells haben.
- Es zeigt an, ob die Beziehung zwischen dem Ziel und einem Merkmal linear, monoton oder komplexer ist.
- Es enthält automatisch eine andere Variable, mit der Ihre ausgewählte Variable am meisten interagiert.
- In der folgenden Darstellung LSTATinteragiert das Feature am meisten mit dem Feature CRIM und wir können den ungefähren linearen Trend zwischen LSTAT und der Zielvariablen sehen.



Zusammenfassendes Diagramm SHAP

- Zusammenfassende Diagramme werden verwendet, um herauszufinden, welche Funktionen für ein Modell am wichtigsten sind.
- Beispiel zeigt SHAP-Werte jedes Features für jede Stichprobe
- Diagramm wird dann nach der Summe der SHAP-Werte über alle Stichproben sortiert.



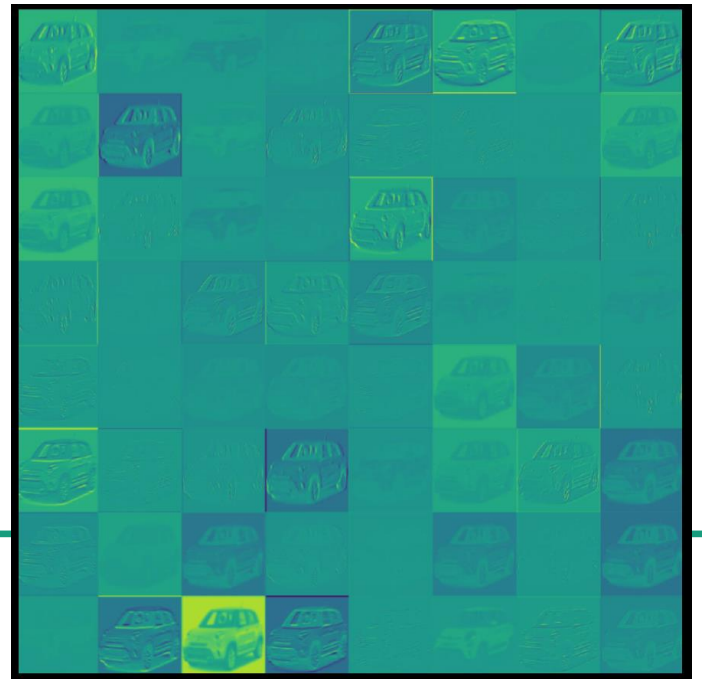
DeepDive GradCam

Grad Cam

- Inzwischen gibt es verschiedene Methoden, um CNN-Outputs zu erklären
- Bibliothek tf-explain eine breite Palette nützlicher Methoden für TensorFlow 2.x.
- Wir werden nun kurz auf die Ideen der verschiedenen Ansätze eingehen, bevor wir uns Grad-CAM zuwenden
  - Activation Visulalization
  - Vanilla Gradient
  - Occlusion Sensitivitx
  - CNN Fixations

Activations Visualization

- Activations Visualization ist die einfachste Visualisierungstechnik
- Hierbei wird die Ausgabe einer bestimmten Layer innerhalb des Netzwerks während des Vorwärtsdurchlaufs ausgegeben
- Diese kann hilfreich sein, um ein Gefühl für die extrahierten Features zu bekommen, da die meisten Activations während des Trainings gegen Null tendieren (bei Verwendung der ReLu-Activation)
- Ein Beispiel für die Ausgabe der ersten Faltungsschicht des Auto-Modell-Classifiers ist unten dargestellt

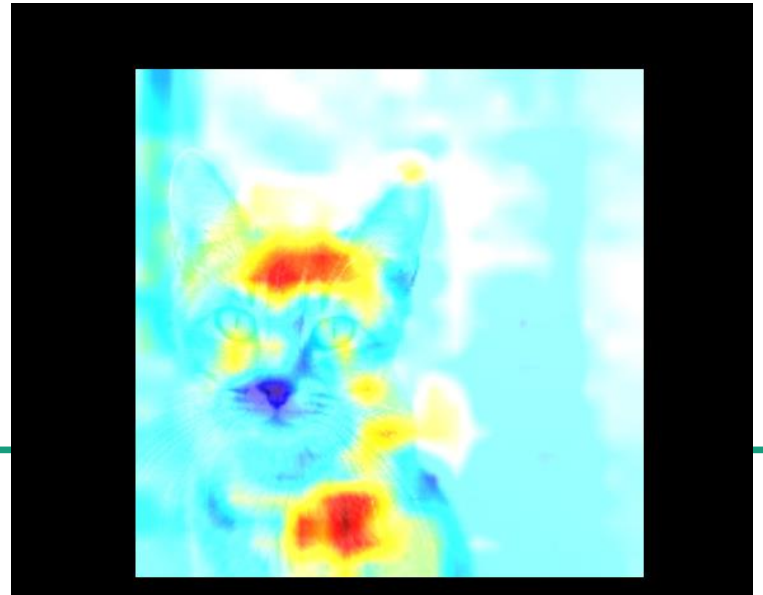


Vanilla Gradient

- Man kann die Vanilla-Gradients der Ausgabe der vorhergesagten Klassen für das Eingangsbild verwenden, um die Bedeutung der Eingangspixel abzuleiten
- Hervorgehobene Bereich fokussiert hauptsächlich das Auto
- Im Vergleich zu den unten besprochenen Methoden ist der diskriminierende Bereich viel weniger eingegrenzt.

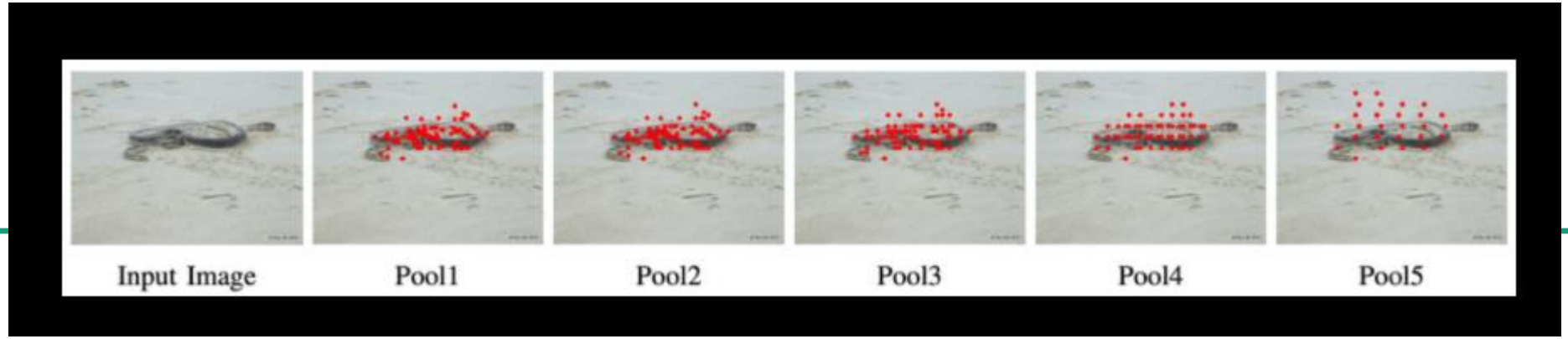
Occlusion Sensitivity

- Bei diesem Ansatz wird die Signifikanz bestimmter Teile des Eingangsbildes berechnet, indem die Vorhersage des Modells für verschiedene ausgeblendete Teile des Eingangsbildes bewertet wird
- Teile des Bildes werden iterativ ausgeblendet, indem sie durch graue Pixel ersetzt werden.
- Je schwächer die Vorhersage wird, wenn ein Teil des Bildes ausgeblendet ist, desto wichtiger ist dieser Teil für die endgültige Vorhersage.
- Basierend auf der Unterscheidungskraft der Bildregionen kann eine Heatmap erstellt und dargestellt werden.
- Beispielbild von tf-explain, welches das Ergebnis der Anwendung des Verfahrens der Occlusion Sensitivity für ein Katzenbild zeigt.



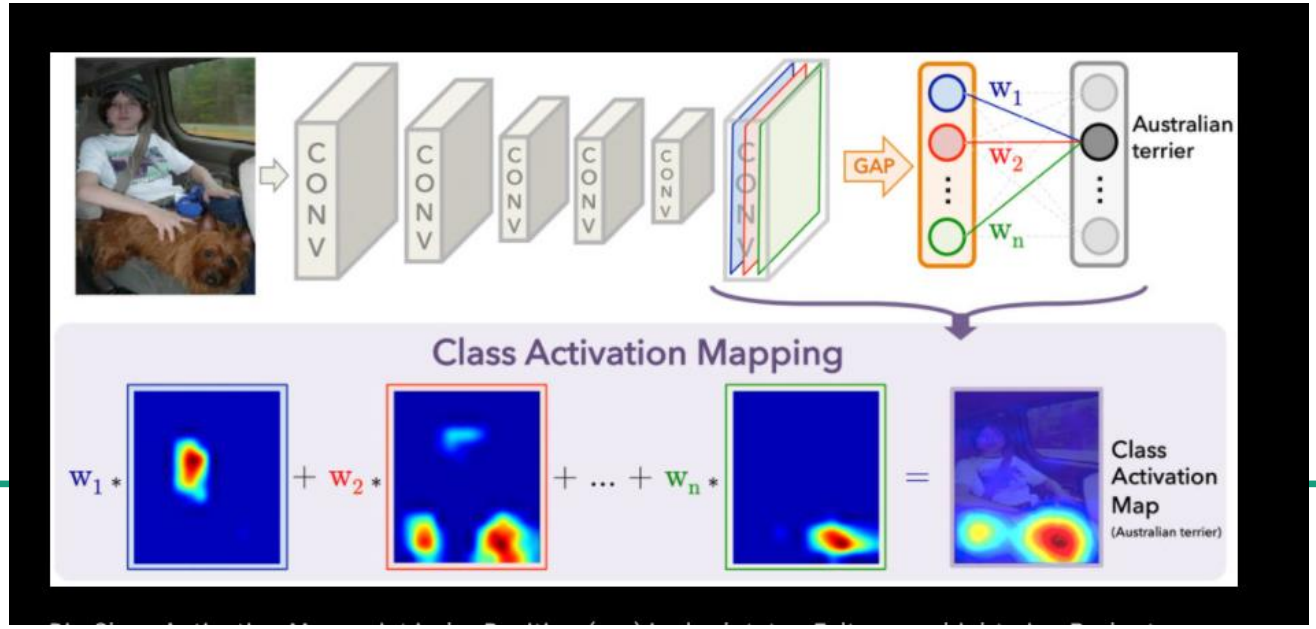
CNN Fixations

- **CNN Fixations**
- Die Idee dabei ist, zurück zu verfolgen, welche Neuronen in jeder Schicht signifikant waren, indem man die Activations aus der Vorwärtsrechnung und die Netzwerkgewichte betrachtet.
- Die Neuronen mit großem Einfluss werden als Fixations bezeichnet.
- Dieser Ansatz erlaubt es also, die wesentlichen Regionen für das Ergebnis zu finden, ohne wiederholte Modellvorhersagen berechnen zu müssen (wie dies z.B. für die oben erklärte Occlusion Sensitivity der Fall ist).
- Das Verfahren kann wie folgt beschrieben werden:
  - Der Knoten, der der Klasse entspricht, wird als Fixation in der Ausgabeschicht gewählt.
  - Dann werden die Fixations für die vorherige Schicht bestimmt, indem berechnet wird, welche der Knoten den größten Einfluss auf die Fixations der nächsthöheren Ebene haben, die im letzten Schritt bestimmt wurden.
  - Die Knotengewichtung wird durch Multiplikation von Activations und Netzwerk-Gewichten errechnet (Backtracking wird so lange durchgeführt, bis das Eingabebild erreicht ist



Class Activation Maps

- Die Class Activation Map weist jeder Position (x, y) in der letzten Faltungsschicht eine Bedeutung zu, indem sie die Linearkombination der Activations – gewichtet mit den entsprechenden Ausgangsgewichten für die beobachtete Klasse berechnet.
- Die resultierende Class Activation Mapping wird dann auf die Größe des Eingabebildes hochgerechnet.
- Aufgrund der Architektur von CNNs ist die Aktivierung, z. B. oben links für eine beliebige Schicht, direkt mit der oberen linken Seite des Eingabebildes verbunden.
- Deshalb können wir nur aus der Betrachtung der letzten CNN-Schicht schließen, welche Eingabebereiche wichtig sind.
- Bei dem Grad-CAM-Verfahren, handelt es sich um eine Verallgemeinerung von CAM.
- Grad-CAM kann auf Netzwerke mit allgemeinen CNN-Architekturen angewendet werden, die mehrere fully connected Layers am Ausgang enthalten.



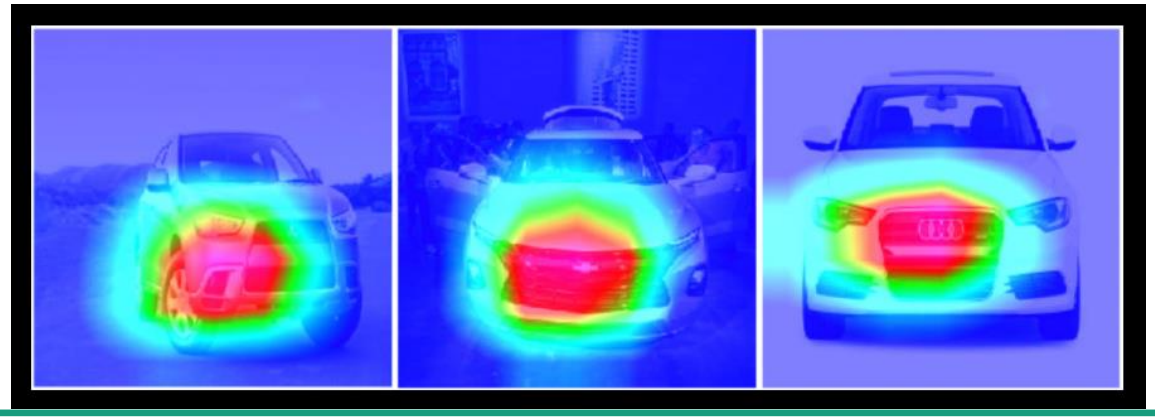
GradCAM

- **Schrittweises Vorgehen von Grad-CAM:**

1. **Vorhersage des Modells:**
    1. Das Bild wird durch das neuronale Netz verarbeitet, und das Modell trifft eine Vorhersage (z.B. „Hund“ oder „Katze“).
2. **Gradientenberechnung:**
    1. Grad-CAM nutzt die Gradienten der Vorhersage, d.h. die Ableitungen der Klassifikationspunkte, bezogen auf die Feature Maps der letzten **Convolutional Layer** (die tiefen Schichten, die visuelle Merkmale lernen).
    2. Diese Gradienten zeigen, wie stark eine kleine Änderung in einem Bereich der Feature Map die Vorhersage beeinflusst.
3. **Gewichtung der Feature Maps:**
    1. Die Gradienten werden als **Gewichte** verwendet, um die Bedeutung jeder Feature Map zu bestimmen. Dadurch wird klar, welche Merkmale das Modell am wichtigsten für die Entscheidung findet.
4. .

GradCAM

1. **Erzeugung der Heatmap:**
    1. Die gewichteten Feature Maps werden zu einer **Heatmap** kombiniert, die zeigt, welche Bildbereiche für die Vorhersage am relevantesten sind.
    2. Diese Heatmap wird auf das Originalbild gelegt, wobei warme Farben (rot, gelb) die relevanten Bereiche hervorheben und kalte Farben (blau) weniger relevante Bereiche zeigen.
2. **Interpretation:**
    1. Durch die Heatmap sieht man, welche Bereiche des Bildes das Modell für die Entscheidung genutzt hat. Das hilft dabei, besser zu verstehen, **„was das Modell sieht“** und zu überprüfen, ob es sinnvolle Teile des Bildes fokussiert.



Gesetze der Erklärbarkeit

- Angelehnt an die drei von Isaac Asimov definierten Robotergesetze, die das Zusammenleben zwischen Menschen und Robotern definieren sollen, haben Biecek und Burzykowski [12] drei Gesetze der Erklärbarkeit für KI-Modelle abgeleitet. Diese legen Anforderungen fest, die jedes KI-Modell erfüllen sollte [12]:
  - **1. Vorhersagevalidierung:** für jede Vorhersage eines KI-Modells sollte überprüft werden können, wie stark die Evidenz ist, die die Vorhersage stützt.
  - **2. Vorhersagerechtfertigung:** für jede Vorhersage sollte man verstehen können, welche Variablen die Vorhersage in welchem Umfang beeinflussen.
  - **3. Vorhersageerwartung**: für jede Vorhersage sollte man in der Lage sein zu verstehen, wie sich die Vorhersage bei einer Veränderung der Werte der im Modell enthaltenen Variablen ändern würde.
- Es gibt zwei Möglichkeiten, diese Anforderungen zu erfüllen. Die erste ist, White-Box-Modelle zu verwenden. Alternativ können xAI-Verfahren zur Erklärung von Black-Box-Modellen genutzt werden.

Eigenschaften von interpretierbaren KI-Modellen

- Eigenschaften von KI-Modellen grundsätzlich aufweisen um als interpretierbar zu gelten.
  - **1. Simulierbarkeit:** Nutzende sollten in der Lage sein, mithilfe der Eingangsdaten und der Modellparameter in angemessener Zeit jeden für eine Vorhersage benötigten Berechnungsschritt nachvollziehen zu können.
  - **2. Zerlegbarkeit:** Jeder Bestandteil des KI-Modells (d. h. Eingabedaten, Parameter und Berechnung) ist intuitiv erklärbar. Dies bedeutet jedoch auch, dass ein besonders umfangreiches und komplexes Aufbereiten der Rohdaten für das KI-Modell (Feature-Engineering und Feature-Extraction) die Interpretierbarkeit negativ beeinflusst.
  - **3. Algorithmische Transparenz:** Der eingesetzte Lernalgorithmus selbst ist nachvollziehbar. Erfüllt ein KI-Modell also diese Eigenschaften, kann es als interpretierbar gelten. Diese Art von Modelle werden auch als »White-Box«-Modelle bezeichnet.

Nutzergruppenbezogene Erklärungen

- Über die Entwicklung und den Einsatz von KI-Modellen hinweg kommen unterschiedliche Nutzergruppen mit den Modellen in Berührung.
- Tomsett et al. [10] plädieren daher dafür, auch die Erklärbarkeit von KI-Modellen nutzergruppenbezogen zu betrachten.
- Unterschiedliche Anwenderinnen haben unterschiedliche Ansprüche und Erwartungen an KI-Modelle, weshalb auch die Interpretierbarkeit der Modelle in diesen Kontext gesetzt werden muss.
- Dabei definieren die Autoren sechs verschiedene Nutzergruppen:
  - *KI-Entwickler*innen:* Eigentümer*innen oder Entwickler*innen des KI-Systems
  - *Anwender*innen:* Personen, die direkt mit dem KI-System interagieren
  - *Ausführende*: Personen, die basierend auf KI-Systemen Entscheidungen treffen
  - *Entscheidungsbetroffene:* Personen, die unmittelbar von einer KI-Entscheidung betroffen sind, z. B. Endkund*innen
  - *Daten-Subjekte:* Personen, deren Daten genutzt wurden, um das KI-System zu trainieren
  - *Prüfende:* Personen, die das KI-System begutachten bzw. auditieren

Erklärungen

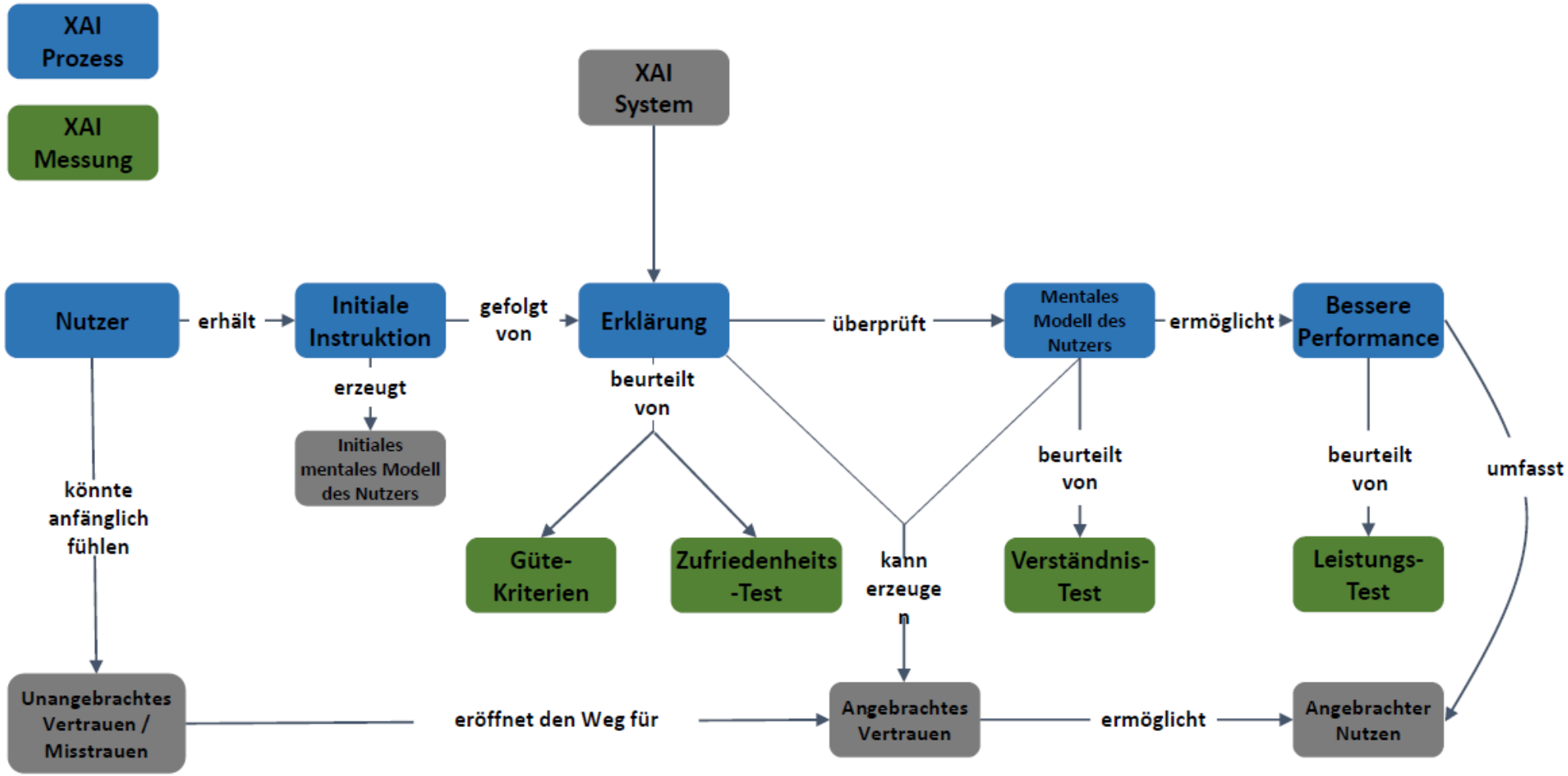
- Erklärungen geben die Antwort auf eine Warum-Frage [Mil19]
- Weiter zeigt er auf, dass Erklärungen anhand der folgenden Kriterien evaluiert werden
  - Wahrscheinlichkeit
    - Zum einen die Wahrscheinlichkeit zu der eine Erklärung wahr ist,
    - und zum anderen die Angabe einer Wahrscheinlichkeit innerhalb einer Erklärung.
  - Einfachheit
  - Generalisierbarkeit
  - Übereinstimmung mit früheren Überzeugungen
- Menschen bevorzugen simple und generalisierte Erklärungen

Anforderungen an Erklärungen

- Kass et al. [Kas88] stellen heraus, dass die Qualität einer Erklärung von drei unterschiedlichen Kriterien abhängig ist:
  - der *Relevanz*
  - der *Überzeugungskraft*
  - und der *Verständlichkeit* einer Erklärung
- Eine Erklärung ist nur dann relevant, wenn sie den aktuellen Zielen und Bedürfnissen des Anwenders entspricht.
- Die Erklärung sollte möglichst viele Informationen enthalten, um diese Ziele zu erreichen.
- Zusätzlich sollte die Erklärung so kurz wie möglich sein, um zu vermeiden, dass Informationen gegeben werden, die nicht notwendig sind.
- Der Anwender ist von einer Erklärung überzeugt, wenn sie auf Fakten beruht, die dieser glaubt. Die Verständlichkeit einer Erklärung wird durch verschiedene Aspekte erreicht
- So sollte die Erklärung einen bestimmten Darstellungstyp verwenden, die der Anwender leicht verstehen kann, prägnant sein und dem Anwender interessante Aspekte aufzeigen.

Anforderungen an Erklärungen

- Flexibilität und Reaktionsfähigkeit
- Versteht der Benutzer eine Erklärung nicht, sollte das System weiteres Wissen zur Verfügung stellen, um auf die speziellen Bedürfnisse des Benutzers zu reagieren [Swa91].
- Fischer et al. [Fis90] bestätigen die These von Kass et al. [Kas88], dass eine Erklärung so kurz wie möglich sein sollte.
- Diese führen ein Erklärsystem ein, das ein Minimum an Erklärungen generiert.
- Erhält das System vom Anwender die Rückmeldung, dass die Erklärung nicht ausreichend war, fügt dieses der gegebenen Erklärung weitere Details hinzu [Fis90]. Dieser Ansatz ist in der Lage, die Bedürfnisse des Anwenders optimal zu befriedigen, ohne diesen dabei zu überfordern.
- Darüber hinaus versucht dieser Ansatz, komplexe Erklärungen zu vermeiden.
- Essenziell ist es zu wissen, welches Hintergrundwissen der Anwender bereits hat und welche Ziele er verfolgt.
- Dabei kann ein mentales Modell unterstützen, um nachvollziehen zu können, wie ein Anwender ein bestimmtes Ereignis, einen Prozess oder ein System versteht


XAI Prozess
XAI Messung
XAI System
Nutzer
erhält
Initiale Instruktion
gefolgt von
Erklärung
überprüft
Mentales Modell des Nutzers
ermöglicht
Bessere Performance
erzeugt
Initiales mentales Modell des Nutzers
beurteilt von
könnte anfänglich fühlen
Güte-Kriterien
Zufriedenheits-Test
kann erzeugen
beurteilt von
Verständnis-Test
beurteilt von
Leistungs-Test
umfasst
Unangebrachtes Vertrauen / Misstrauen
eröffnet den Weg für
Angebrachtes Vertrauen
ermöglicht
Angebrachter Nutzen

Anforderungen an Erklärungen

- Lipton [Lip18] postuliert, dass sich Erklärungen auf anormale Gründe stützen sollten
  - Diese zeichnen sich dadurch aus, dass sie dem Anwender zunächst ungewöhnlich erscheinen.
- Zusätzlich betont Lipton [Lip18], dass ein nachvollziehbares Modell *menschlich-simulierbar* sein sollte.
- Menschlich simulierbar:
  - ein Anwender in der Lage sein sollte, die Ergebnisse eines Modells nachzuvollziehen.
  - Der Anwender soll mit den Eingabedaten und den Parametern des Modells imstande sein, in angemessener Zeit jede Berechnung, die zur Erstellung einer Vorhersage erforderlich ist, auszuführen [Lip18].

Eigenschaften von Erklärungen

- **Eigenschaft der Erklärung:** Zuletzt weisen natürlich auch die Erklärungen selbst unterschiedliche Eigenschaften auf. Diese sind meist relativ abstrakt definiert – bis heute fehlt eine formale Definition von Erklärungseigenschaften. Einige Beispiele für Erklärungseigenschaften, entnommen aus sind:
  - ***Genauigkeit:*** Vorhersagegenauigkeit für unbekannte Daten.
  - ***Wiedergabetreue*** gibt an, inwieweit eine post-hoc Erklärung das Verhalten des Black-Box-Modells widerspiegelt. Die Wiedergabetreue ist eine besonders erstrebenswerte Eigenschaft, da nur bei korrekt wiedergegebenem Modellverhalten durch Analyse der Erklärungen sinnvolle Rückschlüsse gezogen werden können.
  - ***Konsistenz*** gibt an, ob für ein definiertes Modell ähnliche Datenpunkte ähnliche Erklärungen erhalten.
  - ***Verständlichkeit und Praxistauglichkeit*** gibt an, wie gut Nutzende die Erklärung verstehen und für ihre individuelle Aufgabe nutzen können. Die Verständlichkeit ist eine der wichtigsten und zugleich am schwersten zu formalisierenden Erklärungseigenschaften.

Bewertung von xAI Methoden

- *Stabilität*
  - Eine allgemein gültige Metrik für die Erklärbarkeit ist die Stabilität [38].
  - Erfüllt eine Erklärungsmethode diese Eigenschaft, bedeutet dies, dass man für den gleichen Datenpunkt stets die gleiche Erklärung erhält.
  - Wird die Eigenschaft hingegen nicht erfüllt, kann dies aufgrund inkonsistenter Ergebnisse Verwirrungen und schlimmstenfalls einen Vertrauensverlust in die Erklärungen zur Folge haben.
  - Zudem erschweren instabile Erklärungen das Ableiten von Handlungsoptionen.

Bewertung von xAI Methoden

- *Trennbarkeit* (engl. separability)
  - stellt sicher, dass unterschiedliche Datenpunkte unterschiedliche Erklärungen aufweisen.
  - So wird ausgeschlossen, dass beispielsweise zwei komplett gegensätzliche Datenpunkte die gleiche Erklärung enthalten.
  - Da Entscheidungsbäume und Wenn-Dann-Regelsätze mehrere Datenpunkte über einen Baumpfad bzw. eine Regel einordnen, ist Trennbarkeit für diese Modelle nicht erstrebenswert.

Bewertung von xAI Methoden

- *Konsistenz*
  - ist dann für eine Erklärungsmethode gegeben, wenn sie ähnliche Erklärungen für nur geringfügig unterschiedliche Instanzen generiert.
  - Die Konsistenz bezieht sich dabei immer auf ein festgelegtes Modell.
  - Um die Konsistenz eines Datenpunktes zu berechnen, werden die Eingangsmerkmale geringfügig verändert und im Anschluss geprüft, ob sich die Erklärung dadurch signifikant verändert.

Bewertung von xAI Methoden

- *Erklärungsähnlichkeit:*
  - Im besten Fall sollten sich die Erklärungen, die mithilfe unterschiedlicher xAI-Verfahren für denselben Datenpunkt generiert wurden, ähnlich sein.
  - Das Phänomen,wenn dasselbe Ereignis durch verschiedene, widersprüchliche Erklärungen durch unterschiedliche Akteure erklärt wird, ist auch als »Rashomon Effekt« bekannt [40].
  - Tritt dieses Phänomen auf, d. h. wird ein Datenpunkt durch verschiedene Erklärungen unterschiedlich erklärt, stehen Nutzende vor der Herausforderung, zu entscheiden, welche Erklärung besser ist.
  - Bestenfalls sollte daher eine starke Diskrepanz zwischen verschiedenen Erklärungen vermieden werden.

Bewertung von xAI Methoden

- **Wiedergabetreue**
  - Die ***Wiedergabetreue*** (engl. fidelity) gibt an, inwieweit eine Erklärung der Vorhersage des Modells ähnlich ist.
  - Diese Metrik kann in direkter Form lediglich für ML-Modelle berechnet werden(z. B. Entscheidungsbäume).
  - Für andere Erklärungsergebnisse werden daher Approximationen benötigt, was einen Vergleich zwischen unterschiedlichen Erklärungsergebnissen schwierig bis unmöglich macht.
  - Um die Wiedergabetreue zu berechnen, wird untersucht, in wie vielen Fällen das Erklärungsmodell und das Black-Box- Modell die gleiche Entscheidung getroffen haben.
  - Je besser die Erklärung die Vorhersage imitiert, desto höher ist der Wiedergabetreue-Wert.

Bewertung von xAI Methoden

- ***Regel-Wiedergabetreue***
  - Die Wiedergabetreue von Wenn-Dann-Regeln hängt eng zusammen mit der Regelabdeckung (engl.: coverage).
  - Diese gibt an, für wie viele Datenpunkte des Datensatzes die Regel gilt.
  - Die Wiedergabetreue informiert dann darüber, für wie viele von der Regel abgedeckten Datenpunkte das ML-Modell die gleiche Entscheidung trifft wie die betreffende Regel.
  - Die Regel-Wiedergabetreue wird also für jede einzelne Regel berechnet.

Metrik	Lernaufgabe		Ergebnis				
	Klass.	Regr.	Heat map	Merkmals-relevanz	Daten-punkt	Regeln	Surrogat (EB)
Stabilität	●	●	●	●	●	●	●
Trennbarkeit	●	●	●	●	●	●	●
Konsistenz	●	●	●	●	●	●	●
Erklärungs-ähnlichkeit	●	●	●	●	●		
AOPC	●	O	●	O			
Wiedergabe-treue	●	●				●	●
Regel-Wieder-gabetreue	●					●	●*
Laufzeit	●	●	●	●	●	●	●

Tabelle 2: Anwendbarkeit der betrachteten Evaluationsmetriken auf unterschiedliche Erklärungsergebnisse und Lernaufgaben.

* *wenn der EB in Wenn-Dann-Regeln übersetzt wird*

Legende: ● anwendbar O mit Modifikationen anwendbar

Evaluation von xAI Methodem

- Für einen erfolgreichen praktischen Einsatz von xAI-Methoden ist es von größter Bedeutung, die Verfahren bzw. die Erklärungen hinsichtlich diverser Kriterien (z. B. der Verständlichkeit) evaluieren zu können.
- Je nach verfolgter Zielsetzung sind hierbei unterschiedliche Ansätze zu präferieren.
- Doshi-Velez und Kim [19] definieren drei Herangehensweisen, um Erklärbarkeit zu evaluieren:
  - **1.** Anwendungsbezogene Evaluation
  - **2.** Nutzerbezogene Evaluation
  - **3.** Funktional-basierte Evaluation

Projektaufgabe

- Pro Team
  - Implementierung und Anwendung von LIME, SHAP (tabellarischer Datensatz) oder GradCam (Bilddatensatz)
  - Jede Person aus dem Team übernimmt eine Fragestellung (s.u.)
  - Datensatz ist frei wählbar (Klassifikation, UCI oder Kaggle)
- Bearbeitung Projektaufgabe
  - Verfahren erklären
  - Notebook vorstellen mit Datensatz und Ergebnissen
  - Was überzeugt an dem Verfahren?
  - Was fällt negativ auf?
  - Sonstiges?
- Präsentation der Ergebnisse in Vorlesung (Pro Team 20 Min)