DAV/DGVFM
Herbsttagung
2022

Interpretable Machine Learning: Verfahren und Anwendungen

Thomas Hofmann (msg life)

DAV
DEUTSCHE AKTUARVEREINIGUNG e.V.

DGVFM
DEUTSCHE GESELLSCHAFT FÜR VERSICHERUNGS- UND FINANZMATHEMATIK e.V.

Herbsttagung von DAV und DGVFM, 14./15.11.2022

Agenda

1. Motivation
2. Interpretable Machine Learning (IML) Taxonomie
3. Verfahren und Anwendungen
4. Zusammenfassung

Agenda

1. **Motivation**
2. Interpretable Machine Learning (IML) Taxonomie
3. Verfahren und Anwendungen
4. Zusammenfassung

Motivation

- Maschinelles Lernen findet in vielen Branchen immer mehr Anwendung, auch im Versicherungsumfeld.
- Unternehmen sehen darin zum Beispiel die Chance, Risiken besser einzuschätzen und Kosten zu senken.
- Selbst wenn ein ML-Modell Vorhersagen mit hoher Genauigkeit erbringt, kann dem Modell nicht einfach blind vertraut werden.
- Transparenz über die Modellvorhersage ist im stark regulierten Versicherungsumfeld ebenso wichtig wie Genauigkeit.
- Das Modell ist mittels einer Loss-Funktion trainiert, die auf einer einzigen Metrik basiert.
- Ideal wäre ein Modell, das nur kausale Merkmale enthält.

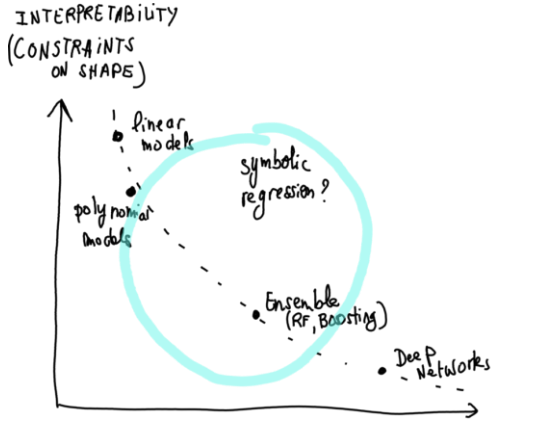
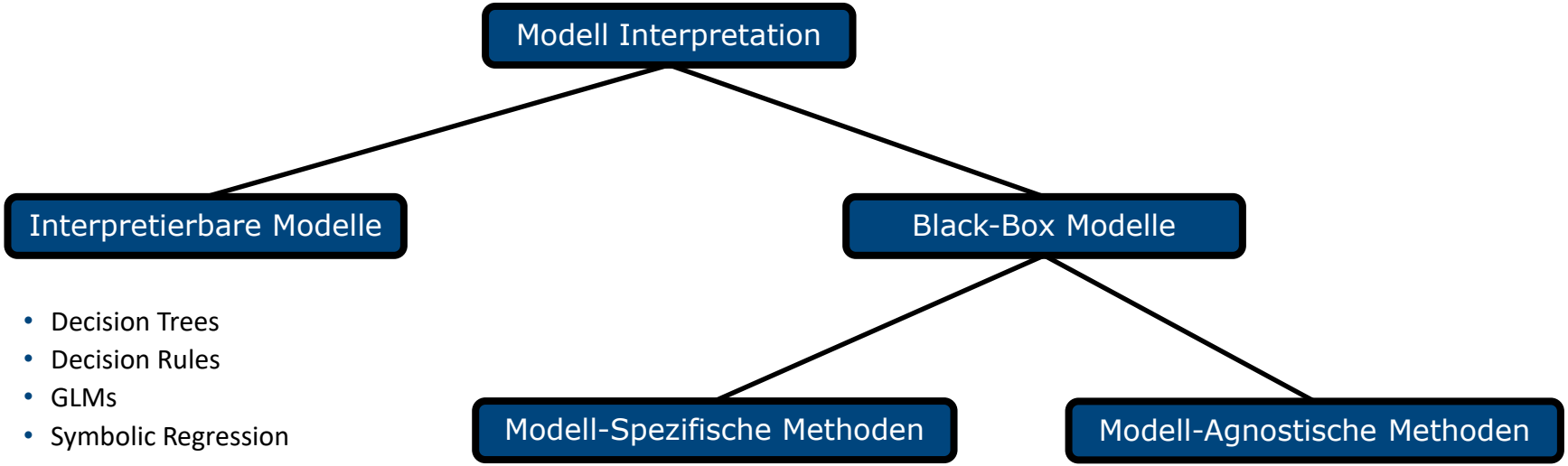


Image source: https://towardsdatascience.com/what-if-we-didnt-have-to-compromise-between-interpretability-and-performance-da00d4e30a44

Agenda

1. Motivation
2. **Interpretable Machine Learning (IML) Taxonomie**
3. Verfahren und Anwendungen
4. Zusammenfassung

IML Taxonomie



- Decision Trees
- Decision Rules
- GLMs
- Symbolic Regression
- …

- Random Forest Explainer
- Visual-Activation für Neural Networks
- …

- Für alle Modelle anwendbar
- Feature Effect Methoden
- Feature Importance Methoden
- Feature Interaction Methoden
- Surrogate Modelle
- …

Funktionsweise vieler Modell-Agnostischer IML Methoden

Modifizierte Daten

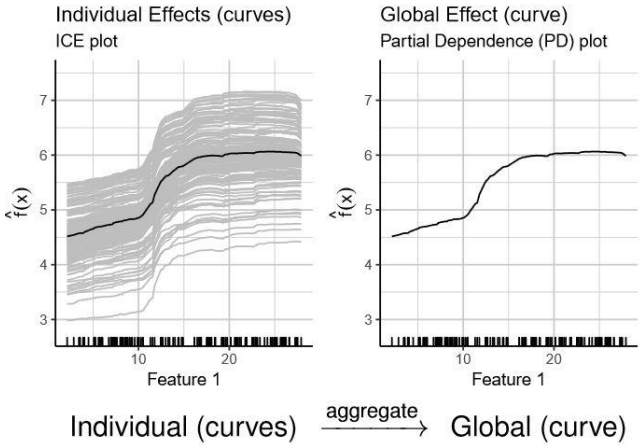
x_1	x_2	y
A	21	100
B	4	165

x_1	x_2	y
A	4	100
B	21	165

Modellvorhersage

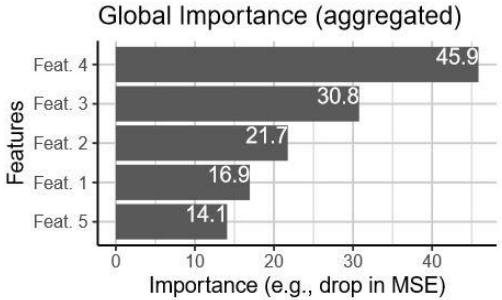
BLACK BOX

Zusammenfassung

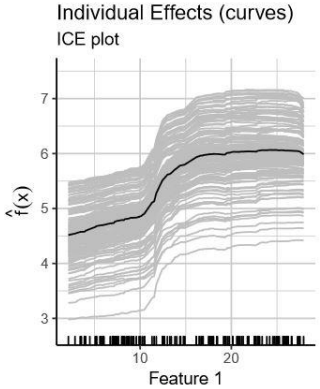


Feature Importance und Feature Effects

- **Feature Importance** Methoden ordnen die Merkmale danach ein, wie stark sie zur Vorhersageleistung oder Vorhersagevarianz des Modells beitragen.
  - Ergebnis ist komprimiert auf eine Zahl pro Merkmal.
  - Gibt Einblick in den Zusammenhang zwischen Merkmal und Zielvariable.
  - Modell-Agnostische Methoden: PFI, CFI, LOCO, Shapley Values,…
  - Pendant in LM: t-statistic, p-value



- **Feature Effects** Methoden zeigen die Veränderung der Modellvorhersage aufgrund von Änderungen der Feature-Werte an.
  - Ein Plot pro Merkmal erforderlich.
  - Die Ausprägung der Zielvariablen wird nicht berücksichtigt, nur die Modellvorhersage.
  - Modell-Agnostische Methoden: PDP, ICE curves, ALE plots,…
  - Pendant in LM: Regression coefficient β_j



Globale vs. Lokale Interpretierbarkeit

- **Globale Interpretationsmethoden** beschreiben das erwartete Verhalten des gesamten Modells bzgl. der Verteilung der Daten.
  - Ziel ist es, das gesamte Modell zu verstehen. D.h. wie entstehen Vorhersagen mit Hilfe der Merkmale und der Modellkomponenten.
  - Es kann nicht unterschieden werden, ob sich verschiedene Auswirkungen global überschneiden (Interaktionen).
  - Erklärung kann ggf. irreführend sein, falls die einzelnen Merkmale miteinander korrelieren (unrealistische Datenpunkte).
  - Permutation Feature Importance (PFI), Partial Dependence Plots (PDP), Accumulated Local Effect (ALE) plots,…

- **Lokale Interpretationsmethoden** erklären einzelne bzw. regionale Modellvorhersagen.
  - Ziel ist es zu erklären, wie eine einzelne Modellvorhersage zustande kommt.
  - Interaktionen zwischen den einzelnen Merkmalen können ermittelt werden.
  - Es können lokale, systematische Modellfehler erkannt werden.
  - Individual Conditional Expectation (ICE) curves, Local Interpretable Model-Agnostic Explanations (LIME), Shapley values,...

Agenda

1. Motivation
2. Interpretable Machine Learning (IML) Taxonomie
3. **Verfahren und Anwendungen**
4. Zusammenfassung

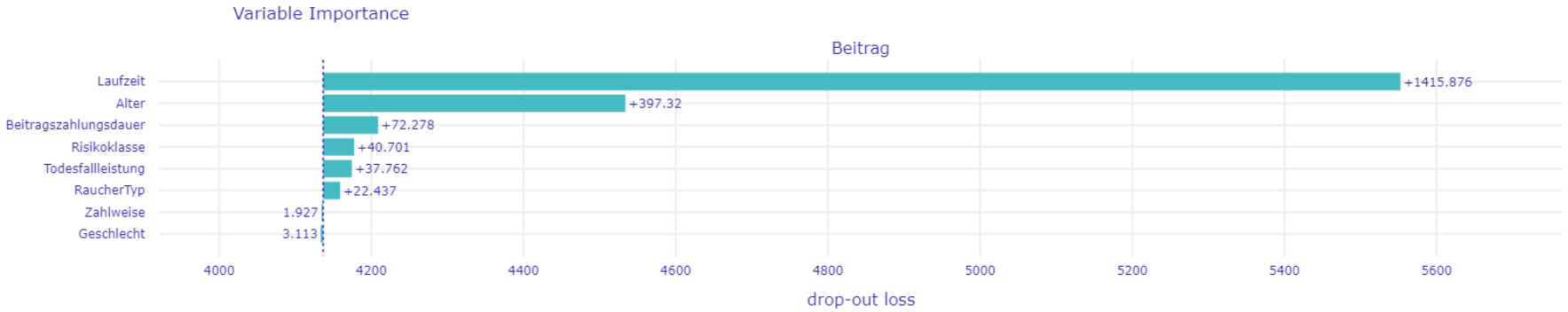
Verfahren und Anwendungen - PFI

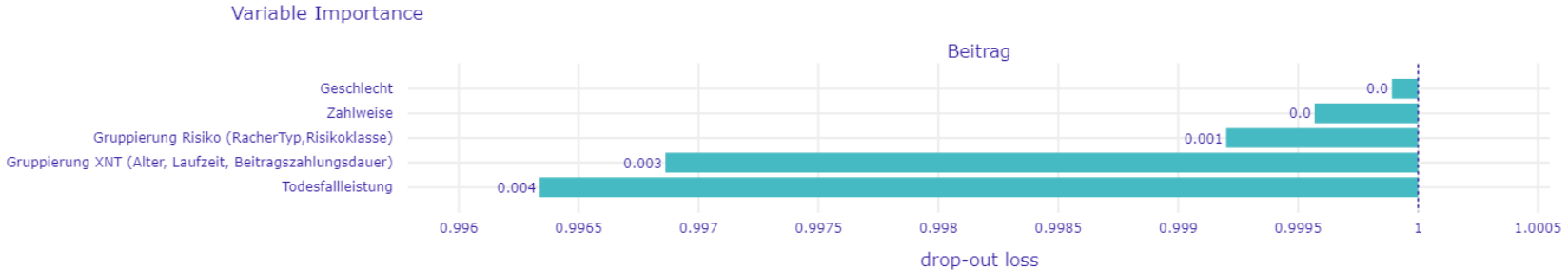
PFI misst den Anstieg des Modellfehlers, nachdem die Werte des Merkmals permutiert wurden.

$$\boldsymbol{FI}_j = \boldsymbol{L}\big(y, \hat{f}(X)\big) - \boldsymbol{L}\big(y, \hat{f}(X_{perm})\big)$$

- Ein Merkmal ist "wichtig", wenn das Permutieren seiner Werte den Vorhersagefehler erhöht.
- Ein Merkmal ist "unwichtig", wenn die Permutation seiner Werte den Modellfehler unverändert lässt.
- Erneutes Modelltraining ist nicht notwendig.
- Liefert stark komprimierten, globalen Einblick in das Verhalten des Modells.
- Erhält die Randverteilung der Variable, zerstört jedoch jegliche Korrelation mit der Zielvariablen.
- Ergebnis ist abhängig von der Anzahl der verwendeten Daten und Permutationen.
- Verfälschte Bedeutung der Ergebnisse, falls Merkmale korrelieren.

Verfahren und Anwendungen - PFI





Verfahren und Anwendungen – ICE-Curves / PDP

ICE-Curves zeigen pro Instanz $x^i = (x_S^i, x_C^i)$, welche Auswirkung eine Änderung des Merkmals x_S^i auf die Modellvorhersage hat.

- Visualisieren die Abhängigkeit der Modellvorhersage von einem Merkmal x_S in jeweils einzelnen Kurven für jede Instanz.
- Für jede Instanz in $\{(x_S^i, x_C^i)\}_{i=1}^N$ wird die Kurve $\hat{f}_S^i$ in x_S^i geplottet, wobei x_C^i unverändert bleibt.
- Lokale Interpretationsmethode, da nur eine Instanz berücksichtigt wird.
- Es können heterogene Effekte bzw. Interaktionen ermittelt werden.
- ICE-Curves repräsentieren nur ein Merkmal.
- Interpretation problematisch, falls die einzelnen Merkmale miteinander korrelieren.

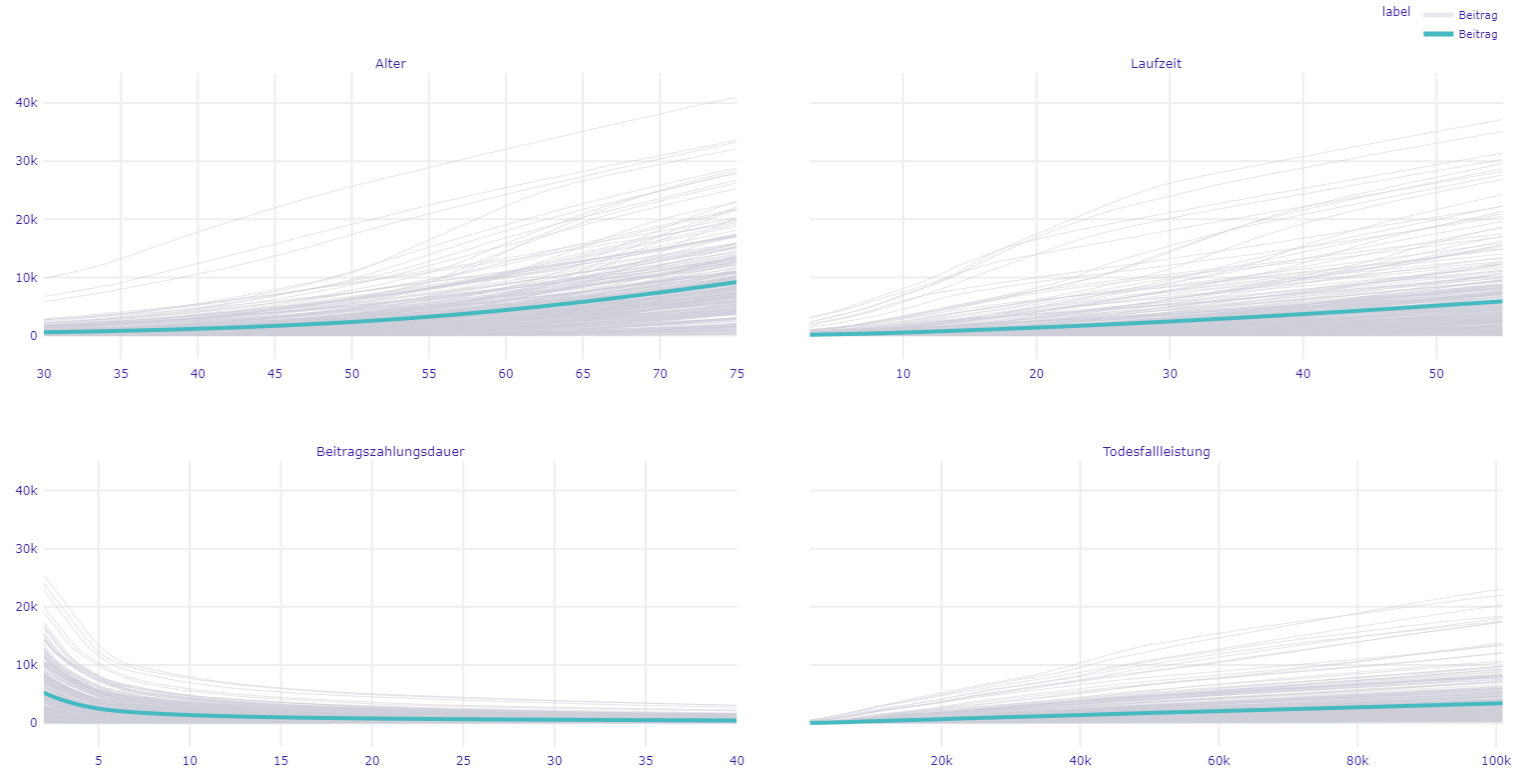
Verfahren und Anwendungen – ICE-Curves / PDP

PDPs zeigen die gemittelte Auswirkung eines Merkmals auf die Modellvorhersage. D.h. die erwartete Vorhersage $\hat{f}(x_S, X_C)$ bzgl. der Randverteilung der Merkmale X_C.

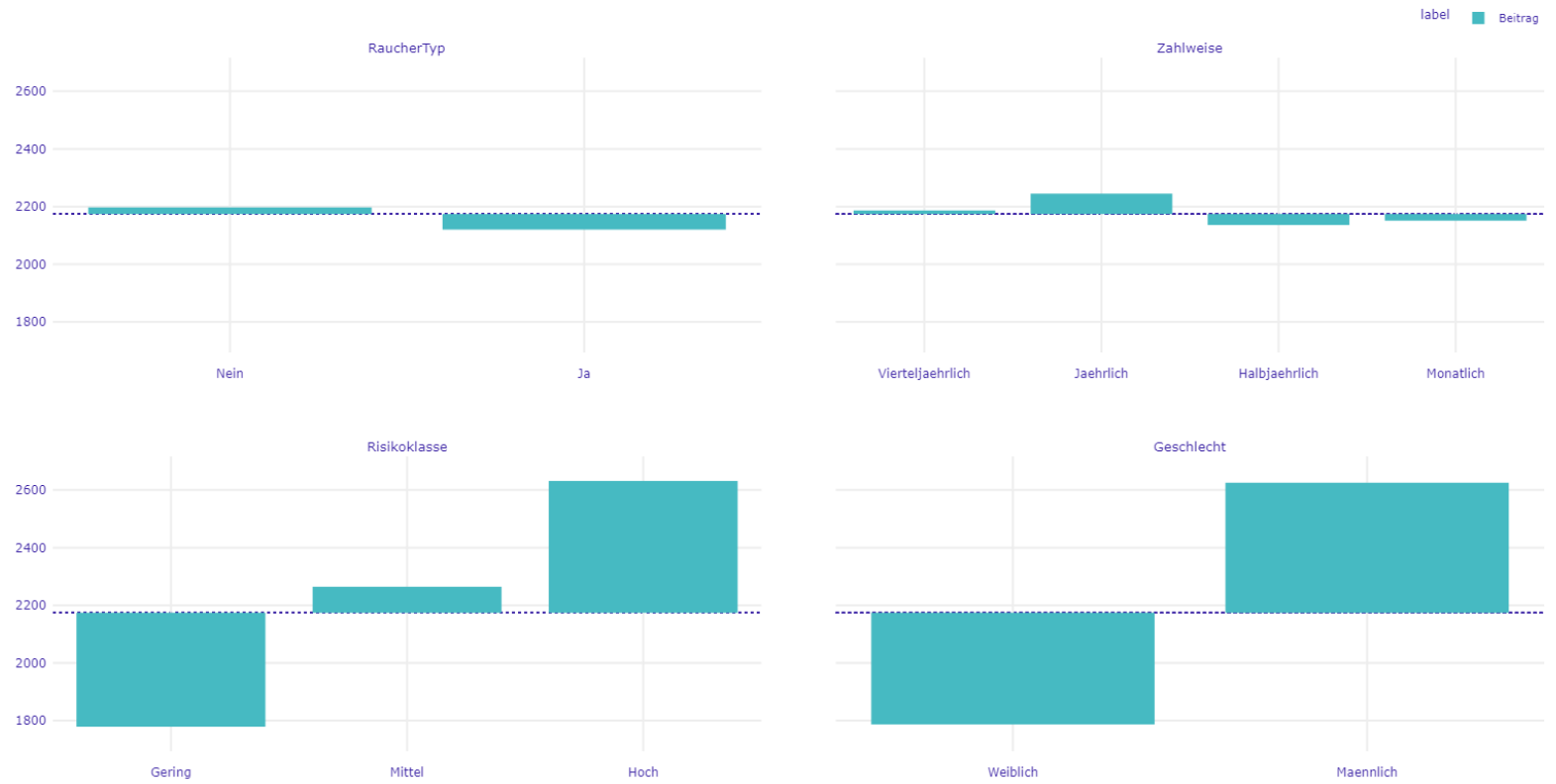
$$\hat{f}_S(x_S) = \boldsymbol{E}_{X_C}\left[\hat{f}(x_S, X_C)\right] = \frac{1}{n}\sum_{i=1}^{n} \hat{f}(x_S, x_C^i)$$

- Zeigen, ob die Beziehung zwischen der Zielvariablen und einem Merkmal linear, monoton oder komplexer ist.
- Globale Interpretationsmethode, da alle Instanzen des Modells berücksichtigt werden.
- Basiert auf der Annahme, dass die einzelnen Merkmale unabhängig voneinander sind.
- Falls x_S mit den Merkmalen X_C unkorreliert ist, dann stellt der PDP genau dar, wie x_S die Vorhersage im Durchschnitt beeinflusst.
- Es können maximal zwei Merkmale auf einmal betrachtet werden.
- Heterogene Effekte könnten verborgen bleiben.

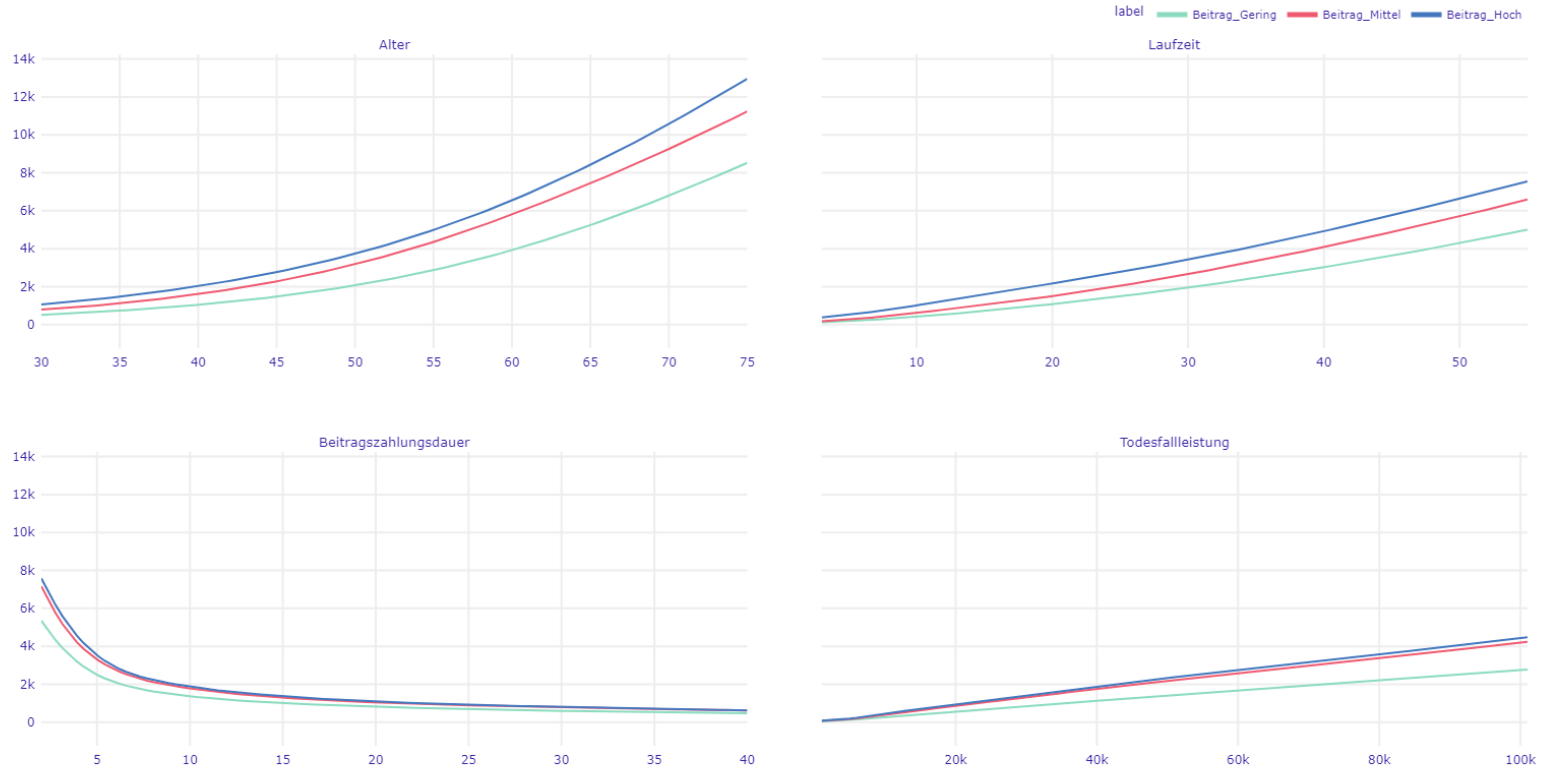
Verfahren und Anwendungen – ICE-Curves / PDP



Verfahren und Anwendungen – ICE-Curves / PDP



Verfahren und Anwendungen – ICE-Curves / PDP



Verfahren und Anwendungen – LIME

LIME ist ein lokales Surrogate Modell zur Erklärung einzelner Modellvorhersagen anhand eines (einfacheren) interpretierbaren Modells.

$$explanation(x) = arg\min_{g \in G} \boldsymbol{L}(f, g, \pi_x) + \Omega(g)$$

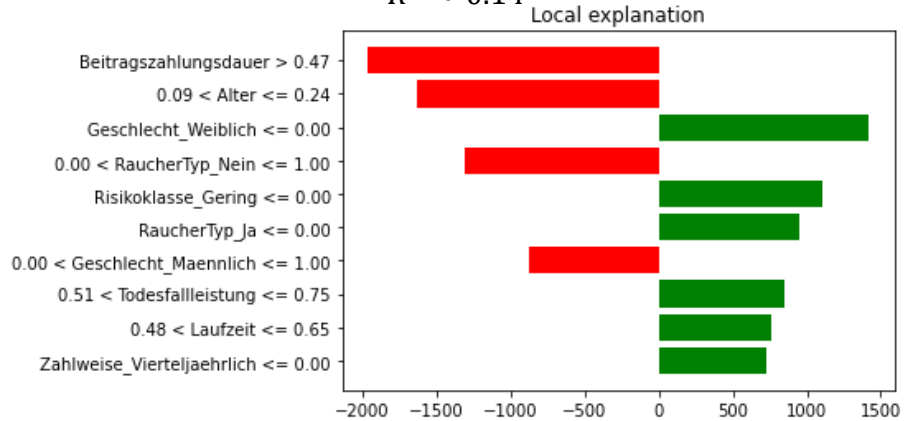
- Liefert einfache lokale Erklärungen, die auch für Nicht-Experten leicht verständlich sind.
- Es werden oft interpretierbare Modelle wie LM, GLMs, Decision trees, … verwendet.
- Ziel ist es zu erklären, wie die Modellvorhersage $\hat{y}$ anhand der Eingabe x zustande kommt.
- Das Surrogate Modell g soll bei minimaler Komplexität eine maximale lokale Wiedergabegenauigkeit erreichen.
- Das Training des Surrogate Modells g basiert auf künstlichen Datenpunkten um den Eingabewert x.

Verfahren und Anwendungen – LIME

Junge Person

- Geschlecht = männlich
- Alter = 35
- Laufzeit = 30
- Beitragszahlungsdauer = 25
- Todesfallleistung = 70.000
- Zahlweise = monatlich
- Risikoklasse = hoch
- RaucherTyp = nein

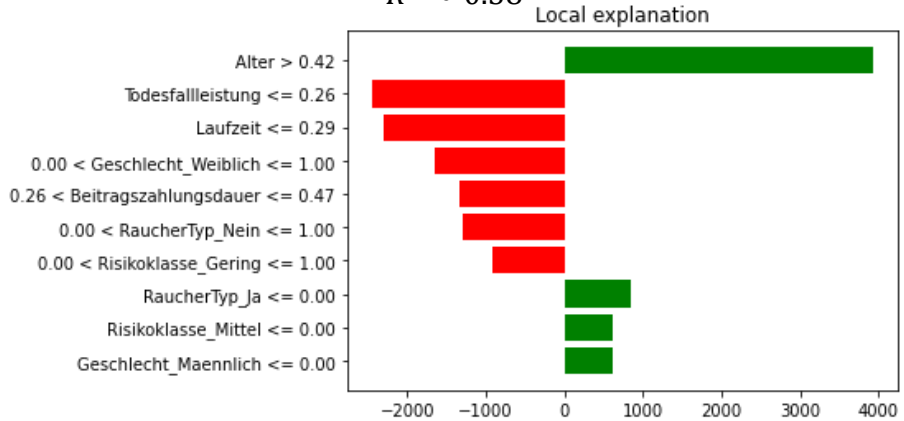
$R^2 \approx 0.14$



Alte Person

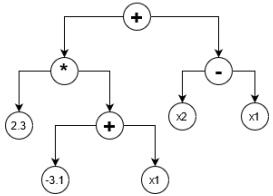
- Geschlecht = weiblich
- Alter = 55
- Laufzeit = 15
- Beitragszahlungsdauer = 15
- Todesfallleistung = 20.000
- Zahlweise = jährlich
- Risikoklasse = gering
- RaucherTyp = nein

$R^2 \approx 0.38$



Verfahren und Anwendungen– Symbolic Regression

- Finde **symbolische Annäherung** für Eingabe-Ausgabe-Paare (X, Y). $SR(X,Y) = g$, so dass $g(X) \approx f(X)$
- Wird in erster Linie zum Finden von Funktionen eingesetzt. $f(x_1, x_2) = \left(2.3 * (-3.1 + x_1)\right) + (x_2 - x_1)$
- Wird in der Regel durch genetische Programmierung gelöst.
- Die gebräuchlichste Kodierung erfolgt dabei mittels symbolischer Regressionsbäume.



- Bei der genetischen Programmierung werden iterativ neue Lösungen erstellt, bis ein bestimmtes Abbruchkriterium erreicht ist.
- **Selektion -** Es muss einen Mechanismus geben, der die Mitglieder einer Population auswählt, um Nachkommen zu erzeugen und ihre genetische Information weiterzugeben.
- **Vererbung -** Es muss eine Möglichkeit geben, dass Kinder die Eigenschaften ihrer Eltern erhalten.
- **Variation -** Es muss eine Vielzahl von Merkmalen in der Population vorhanden sein, oder es muss eine Möglichkeit geben, neue Merkmale einzuführen.

Verfahren und Anwendungen– Symbolic Regression

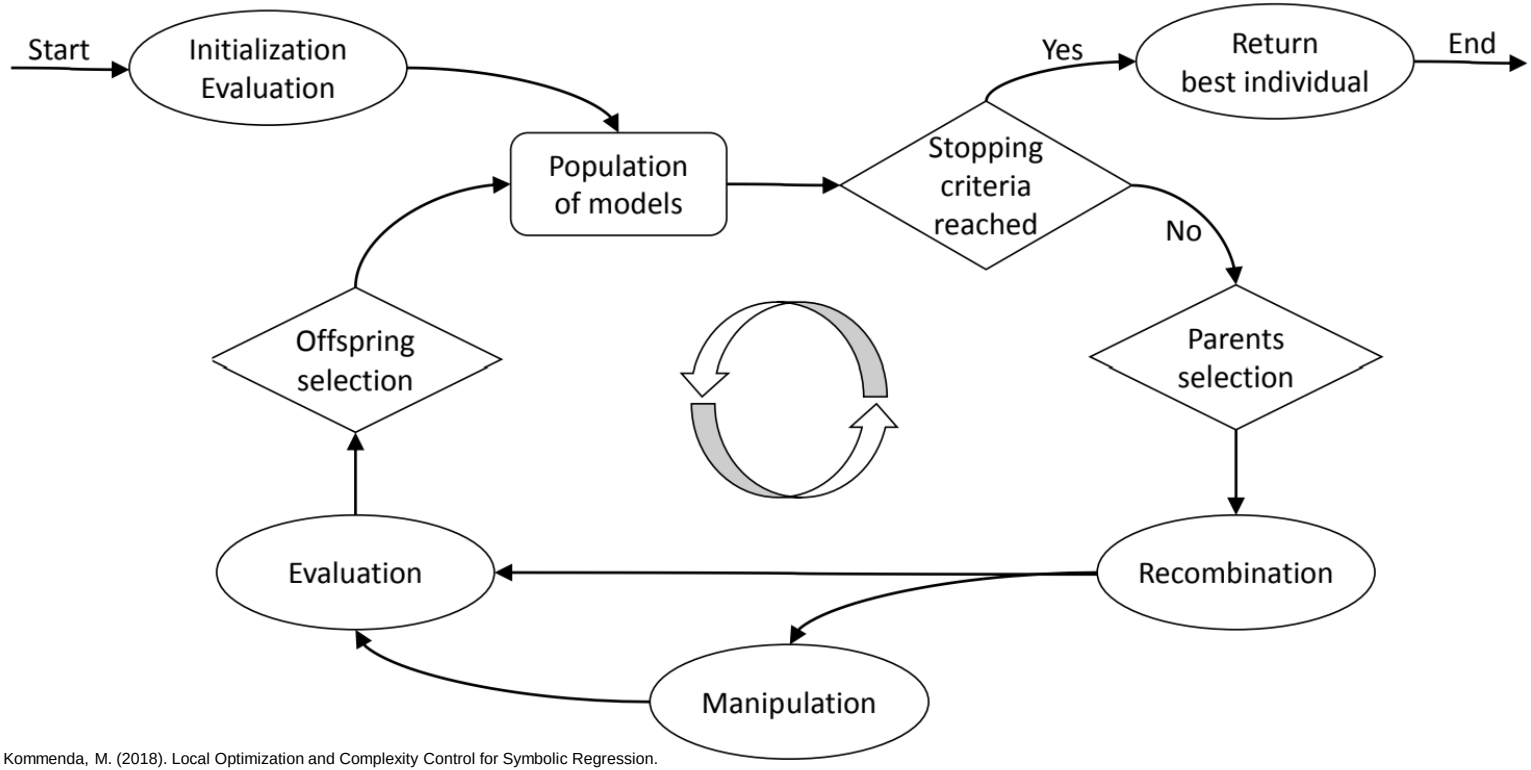


Image source: Kommenda, M. (2018). Local Optimization and Complexity Control for Symbolic Regression.

Verfahren und Anwendungen– Symbolic Regression

$$Beitrag = \frac{Lbw \cdot Todesfallleistung - Reserve}{Bbw}$$

$$RKW = \max(0.0, Reserve - 0.015 \cdot Todesfallleistung)$$

```
X_premium_train = df[[TODESFALLLEISTUNG,BBW,RESERVE,LBW,AEXN,AEXT,ALTER,LAUFZEIT]]
Y_premium_train = df[[BEITRAG]]
```
✓ 0.4s

```
model_premium = createPySRRegressor()
model_premium.fit(X_premium_train,Y_premium_train)
```
✓ 1m 10.2s

```
c:\Users\hof09889\Envs\pysr_test_1\lib\site-packages\pysr\sr.py:1067: UserWarning: Note: i
turned off.
  warnings.warn(
```

PySRRegressor

```
PySRRegressor.equations_ = [
      pick      score                                           equation  \
0        0   0.000000                                        5889.646
1        1   0.139151                    (Todesfallleistung / 8.899107)
2        2   1.378069            ((Todesfallleistung - Reserve) / aext)
3   >>>>  3  11.240925      (((Todesfallleistung * Lbw) - Reserve) / Bbw)
4        4   0.000118  (((Todesfallleistung * Lbw) - (Reserve - -0.00...

             loss  complexity
0  4.022280e+07           1
```

```
simplify(model_premium.get_best().sympy_format)
```
✓ 0.3s

$$\frac{LbwTodesfallleistung - Reserve}{Bbw}$$

```
X_surrender_train = df[[TODESFALLLEISTUNG,RESERVE,BEITRAG,ALTER,LAUFZEIT,
                        BEITRAGSZAHLUNGSDAUER]]
Y_surrender_train = df[[RKW]]
```
✓ 0.3s

```
model_surrender =createPySRRegressor(minmax=True)
model_surrender.fit(X_surrender_train,Y_surrender_train)
```
✓ 34.1s

```
c:\Users\hof09889\Envs\pysr_test_1\lib\site-packages\pysr\sr.py:1067: UserWarning: Note:
turned off.
  warnings.warn(
```

PySRRegressor

```
PySRRegressor.equations_ = [
      pick     score                                           equation  \
0        0  0.000000                                            Reserve
1        1  0.747657                        max(Reserve, -0.11324182)
2        2  0.802105                   max(Reserve - 813.38995, Alter)
3   >>>>  3  3.654451  max(Reserve + (-0.015176028 * Todesfallleistun...
4        4  0.156091  max(Reserve + ((-0.015176028 * Todesfallleistu...

             loss  complexity
0  2.521927e+06           1
```

```
simplify(model_surrender.get_best().sympy_format)
```
✓ 0.8s

$$\max(0.05713699, Reserve - 0.015176028Todesfallleistung)$$

Agenda

1. Motivation
2. Interpretable Machine Learning (IML) Taxonomie
3. Verfahren und Anwendungen
4. **Zusammenfassung**

Zusammenfassung

- Je nach Anwendungsfall ist zu entscheiden, ob die Transparenz oder die Genauigkeit des ML-Modells im Fokus stehen soll.
- Es existieren unterschiedliche IML-Methoden, die je nach Fragestellung ausgewählt werden müssen.
- Die Kombination von globalen und lokalen IML-Methoden geben einen tieferen Einblick in die Funktionsweise des Modells.
- Man sollte sich den Stärken und Schwächen der jeweiligen IML-Methode bewusst sein.
- Die Ergebnisse der IML-Methoden müssen fachlich überprüft werden.
- Wahl der IML-Methode je nach Stakeholder.
- IML-Methoden geben Aufschluss über die Funktionsweise des Modells, nicht über die kausalen Zusammenhänge in den Daten.

Referenzen

- Biecek P. & Burzykowski T. (2020). Explanatory Model Analysis. Explore, Explain, and Examine Predictive Models. With examples in R and Python.
  https://ema.drwhy.ai/

- Cramer M. (2020). PySR: Fast & Parallelized Symbolic Regression in Python/Julia.
  https://astroautomata.com/PySR/

- Kommenda, M. (2018). Local Optimization and Complexity Control for Symbolic Regression.

- Molnar, C. (2022). Interpretable Machine Learning. A Guide for Making Black Box Models Explainable.
  https://christophm.github.io/interpretable-ml-book/

Kontakt

INSUR:IT

Thomas Hofmann

Thomas.Hofmann@msg-life.com

msg insur:it – a brand of msg life and nexinsure

msg life ag
Humboldtstraße 35
70771 Leinfelden-Echterdingen

msg nexinsure ag
Robert-Bürkle-Straße 1
85737 Ismaning / München

www.msg-insurit.com