ERKLÄRBARE KÜNSTLICHE INTELLIGENZ

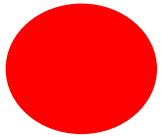
Vorlesung 4

Inhalte der heutigen Vorlesung

- XAI-Verfahren
  - SHAP
  - Counterfactual Explanations
  - GradCam
- Programmierübungen zu den XAI-Verfahren

SHAP

SHAP



- SHAP funktioniert nach dem Prinzip der Spieltheorie und hilft uns zu verstehen, **wie viel jedes Merkmal** (z. B. Einkommen, Alter, Schulden) zu einer bestimmten **Modellvorhersage** beiträgt.
- Das Besondere an SHAP ist, dass es faire Beiträge berechnet, basierend auf den **Shapley-Werten** aus der Spieltheorie.
- Funktionsweise:
  - Jedes Feature im Modell wird wie ein Spieler in einem Team betrachtet. Das Modell trifft eine Entscheidung (wie die Vergabe eines Kredits), und wir möchten wissen, wie viel jedes einzelne Feature zu dieser Entscheidung beigetragen hat.
  - Um den Beitrag eines Features zu berechnen, schaut SHAP, wie sich das Ergebnis ändert, wenn dieses Feature weggelassen wird oder hinzugefügt wird. Es testet alle möglichen Kombinationen der Features, um herauszufinden, wie wichtig jedes einzelne ist.
  - Shapley-Werte berechnen den durchschnittlichen Beitrag eines Features, indem sie alle möglichen Reihenfolgen betrachten, in denen die Features dem Modell "vorgestellt" werden. Auf diese Weise wird für jedes Feature eine faire Einschätzung erstellt, wie viel es zur Vorhersage beigetragen hat.

SHAP

- **Beispiel:**
  - Wenn ein Modell vorhersagt, ob jemand einen Kredit bekommt, könnte SHAP zeigen:
  - Das Einkommen hat die Chancen, den Kredit zu bekommen, um 20 % erhöht.
  - Ein schlechter SCHUFA-Score hat die Chancen um 30 % verringert.
- Diese Werte zeigen klar, **wie stark** jedes Feature die Entscheidung beeinflusst hat, und helfen dabei, das Modell transparenter und verständlicher zu machen.

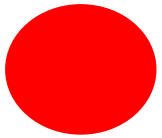
- Basisvorhersage: Der Durchschnitt aller Kreditentscheidungen des Modells ist 0.5 (50 % Chance auf Kredit). Das ist der Basiswert ohne Berücksichtigung der individuellen Features.
- Einfluss der Features: Um den SHAP-Wert für jedes Feature zu berechnen, schauen wir, wie sich die Vorhersage verändert, wenn das Feature entfernt oder hinzugefügt wird:
  - Ohne Berücksichtigung des Einkommens (nur Schulden und SCHUFA-Score betrachtet), ergibt das Modell eine Vorhersage von 0.6. Das Einkommen erhöht also die Vorhersage um +0.1.
  - Wenn wir das Feature Schulden weglassen (nur Einkommen und SCHUFA-Score betrachtet), ergibt das Modell eine Vorhersage von 0.8. Schulden verringern also die Vorhersage um -0.1.
  - Wenn wir den SCHUFA-Score weglassen, sinkt die Vorhersage auf 0.4. Der SCHUFA-Score hat also einen positiven Beitrag von +0.3.
  - Gesamtergebnis: Die Vorhersage für diese Person setzt sich wie folgt zusammen:
  - Basiswert: 0.5 Einkommen trägt +0.1 bei (positiv)Schulden tragen -0.1 bei (negativ)SCHUFA-Score trägt +0.3 bei (positiv)
  - Die endgültige Vorhersage ist also 0.5 + 0.1 - 0.1 + 0.3 = 0.7 (70 % Wahrscheinlichkeit, dass der Kredit bewilligt wird).

- **Interpretation:**
  - **Einkommen** hat einen SHAP-Wert von **+0.1** (es verbessert die Chance auf Kredit).
  - **Schulden** haben einen SHAP-Wert von **-0.1** (sie verschlechtern die Chance).
  - Der **SCHUFA-Score** hat einen SHAP-Wert von **+0.3** (er verbessert die Kreditbewilligung deutlich).
- Dieses Beispiel zeigt, wie SHAP transparent macht, wie stark jedes Feature eine Entscheidung beeinflusst. So kann die Bank nachvollziehen, warum die Entscheidung getroffen wurde.

- Programmierübung:
  - Wenden Sie SHAP auf dem IRIS und dem Breast Cancer Datensatz an.

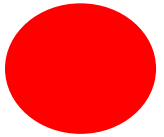
Counterfactual explanations

Counterfactual Explanations



- **Counterfactual Explanations** sind ein Ansatz zur Erklärung von Entscheidungen und Vorhersagen von maschinellen Lernmodellen, indem hypothetische Alternativen zu den gegebenen Eingabewerten erstellt werden.
- Diese Erklärungen zeigen auf, wie sich eine Entscheidung ändern würde, wenn einige der Merkmale des ursprünglichen Inputs geändert werden.
- **Wichtige Konzepte:**
  1. **Hypothetische Alternativen**: Eine counterfactual Erklärung vermittelt, wie eine Eingabe verändert werden müsste, um ein anderes Ergebnis zu erzielen. Beispielsweise könnte eine Kreditgenehmigung abgelehnt werden, und die Erklärung könnte lauten: "Wenn Ihr Einkommen 10.000 Euro höher wäre, wäre Ihr Antrag genehmigt worden."
  2. **Erläuterung des Entscheidungsprozesses**: Counterfactuals helfen dabei, die Entscheidung des Modells transparenter zu gestalten, indem sie dem Benutzer die Merkmale zeigen, die die Vorhersage beeinflussen.
  3. **Interaktive Entscheidungsfindung**: Sie ermöglichen es den Benutzern, mögliche Änderungen in den Eingabewerten zu verstehen und deren Auswirkungen auf das Ergebnis zu erkennen.

Counterfactual Explanations



- Die Erstellung von Counterfactual Explanations erfolgt typischerweise in mehreren Schritten:
  1. **Modelltraining**: Ein maschinelles Lernmodell wird auf einen verfügbaren Datensatz trainiert.
  2. **Auswahl der Zielinstanz**: Eine spezifische Eingangsinformation (Instanz), für die eine Erklärung erforderlich ist, wird ausgewählt.
  3. **Generierung von Alternativen**: Algorithmen werden verwendet, um Hypothetische Alternativen zu generieren. Die Änderungen an den Merkmalen können durch Optimierungstechniken, Datenintegration oder heuristische Ansätze erreicht werden.
  4. **Evaluierung**: Die vorgeschlagenen Änderungen werden analysiert, um sicherzustellen, dass sie zu dem gewünschten Ergebnis führen.

Counterfactual Explanations

- **Beispiel eines Counterfactuals**
- Nehmen wir an, dass wir ein Kreditgenehmigungsmodell haben:
- **Eingabewerte**: Einkommen: 40.000 €, Schulden: 5.000 €, Kreditbetrag: 20.000 €
- **Vorhersage**: Antrag wird abgelehnt.
- **Counterfactual**: "Wenn Ihr Einkommen 50.000 € und Ihre Schulden 3.000 € betragen würden, dann wäre Ihr Antrag genehmigt worden.„

- Programmierübung:
  - Wenden Sie Counterfactual explanations auf dem IRIS und dem Breast Cancer Datensatz an.

GradCam

GradCAM

Funktionsweise von Grad-CAM

- Grad-CAM nutzt die Gradienteninformationen, die durch ein neuronales Netzwerk fließen, um zu bestimmen, welche Bereiche eines Bildes besonders wichtig sind. Der Prozess kann in mehreren Schritten zusammengefasst werden:
  - **Rückpropagation der Gradienten**: Wenn ein Bild (oder ein anderes Eingangsdatenelement) durch das Modell geleitet wird, wird für jede Klasse eine Vorhersage getroffen. Um herauszufinden, wie wichtig jede Schicht im Netzwerk ist, wird der Gradient der Vorhersage in Bezug auf die Aktivierungen der letzten Convolutional-Schicht berechnet.
  - **Berechnung der gewichteten Aktivierungen**: Die Gradienten werden dann verwendet, um die Gewichtungen für die Merkmale in der letzten Convolutional-Schicht festzulegen. Je höher der Gradient für ein bestimmtes Merkmal, desto wichtiger ist es für die Entscheidung des Modells.
  - **Generierung der Heatmap**: Eine kombinierte Heatmap wird erstellt, indem die gewichteten Aktivierungen der letzten Schicht in einer Karte verarbeitet werden. Diese Heatmap zeigt an, welche Bereiche des Bildes zur Vorhersage beigetragen haben, wobei es oft verwendet wird, um Regionen hervorzuheben, die eine hohe Relevanz aufweisen (in warmer Farbe) im Vergleich zu niedrig relevanten Regionen (in kühler Farbe).

GradCAM – Was ist ein Gradient?

- Ein **Gradient** ist ein mathematisches Konzept, das in der mehrdimensionalen Analysis verwendet wird, um die Richtung und die Rate der Änderung einer Funktion zu beschreiben. Im Kontext des maschinellen Lernens und der Optimierung bezieht sich der Gradient hauptsächlich auf den Vektor, der die ansteigenden Richtungen der Funktion angibt.
- Geometrisch betrachtet zeigt der Gradient an:
  - **Richtung des maximalen Anstiegs**: Der Gradient zeigt, in welche Richtung sich die Funktion am schnellsten erhöht. Man kann sich den Gradient als einen Pfeil vorstellen, der vom Punkt, an dem die Funktion evaluiert wird, in die Richtung des steilsten Anstiegs zeigt.
  - **Beliebige Neigung**: Der Betrag des Gradienten beschreibt auch, wie schnell die Funktion in diese Richtung steigt. Ein größerer Betrag bedeutet eine steilere Kurve, während ein kleinerer Betrag eine flachere Kurve anzeigt.

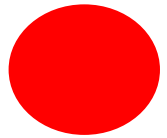
GradCAM – Was ist ein Gradient?

- **Gradienten in der Optimierung**
- Im maschinellen Lernen und der optimization werden Gradienten typischerweise verwendet, um **gradientenbasierte Abstiegstechniken** zu verbessern. Ein bekanntes Verfahren ist der **Gradient Descent** (Gradientenabstieg), das zur Minimierung von Verlustfunktionen verwendet wird:
  - **Gradientenabstiegsverfahren**
    - Hierbei handelt es sich um eine iterative Methode, um die Werte der Parameter eines Modells so anzupassen, dass die Verlustfunktion minimiert wird.
    - Der Gradient der Verlustfunktion wird berechnet, und die Parameter werden in die entgegengesetzte Richtung des Gradienten aktualisiert

GradCAM Funktionsweise

- Berechnung der Gradienten
  - Um zu bestimmen, wie stark jedes Merkmal in der letzten Convolutional-Schicht zur Vorhersage beigetragen hat, berechnet Grad-CAM den Gradienten der Vorhersage in Bezug auf die Aktivierungen der letzten Convolutional-Schicht:
  - Forward Pass: Durch das Bild wird ein Vorwärtsdurchlauf des Modells durchgeführt, um die Aktivierungen in den verschiedenen Schichten, insbesondere in der letzten Convolutional-Schicht, zu erfassen.
  - Loss-Berechnung: Der Loss wird für die vorhergesagte Klasse berechnet. Dies kann eine Verlustfunktion wie die Kreuzentropie sein, die den Unterschied zwischen der Vorhersage und der tatsächlichen Klasse misst.
  - Backward Pass: Der Gradient wird dann durch einen Rückwärtsdurchlauf des Modells in Bezug auf die Vorhersage berechnet.
- Der Gradient zeigt, wie sich die Vorhersage ändert, wenn sich die Aktivierungen der Merkmale in der letzten Convolutional-Schicht ändern.

GradCAM Funktionsweise



- Aggregation der Gradienten
  - Nachdem die Gradienten berechnet wurden, werden sie aggregiert (normalerweise durch Mittelung), um eine gewichtete Kombination der Aktivierungen in der letzten Convolutional-Schicht zu bestimmen. Die Aggregation führt dazu, dass wir herausfinden können, welche Merkmale in der letzten Schicht am signifikantesten zur Vorhersage der Klasse beigetragen haben:
- Erstellung der Grad-CAM-Heatmap
  - Die aggregierten Gewichte werden dann verwendet, um eine gewichtete Summe der Aktivierungen aus der letzten Convolutional-Schicht zu erstellen.
  - Dadurch wird eine Heatmap erzeugt, die die wichtigen Regionen im ursprünglichen Bild anzeigt: