Projektarbeit

Customer Churn Prediction- Binäre Klassifizierung eines Datensatzes im Banken Bereich

Quelle: Kaggle

Volkan Korunan

8. November 2024

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Kundenabwanderung stellt für Unternehmen eine ernsthafte Herausforderung dar, da der Verlust von Kunden langfristig zu Umsatzrückgängen und erhöhten Akquisitionskosten führt. Durch die Identifizierung abwanderungsgefährdeter Kunden können Unternehmen gezielte Maßnahmen ergreifen, um diese Kunden zu halten und die Kundenzufriedenheit zu steigern. Ein solches Projekt zur Vorhersage von Kundenabwanderung ist daher von entscheidender Bedeutung, um frühzeitig präventiv einzugreifen und die Bindung bestehender Kunden zu stärken.

1.1 Ziele dieser Arbeit

Das Ziel dieser Arbeit ist es, ein Modell zur Vorhersage der Kundenabwanderung zu entwickeln, das eine höhere Genauigkeit als das bestehende ANN-Modell auf Kaggle erreicht, welches eine Genauigkeit von 87% erzielt hat. Anstelle des dort verwendeten Artificial Neural Networks (ANN) wurde in dieser Arbeit ein Ensemble-Ansatz verfolgt, der auf einer Kombination mehrerer leistungsstarker Klassifikatoren basiert: XGBoost (XGBClassifier), LightGBM (LGBMClassifier) und CatBoost (CatBoostClassifier). Diese Modelle wurden in einem Voting-Classifier zusammengefasst, um die Vorhersageleistung zu maximieren. Darüber hinaus wurde BayesSearchCV eingesetzt, um die Hyperparameter des Ensembles zu optimieren und so das Potenzial für eine höhere Genauigkeit auszuschöpfen. Die Verbesserung der Genauigkeit über die 87 %-Marke hinaus würde Banken helfen, gefährdete Kunden noch präziser zu identifizieren und gezielte Maßnahmen zur Kundenbindung zu ergreifen.

Die Entwicklung eines solchen Modells zielt darauf ab, die Genauigkeit und Zuverlässigkeit der Brustkrebsdiagnose zu verbessern. Durch den Einsatz von maschinellem Lernen (Neuronalen Netztes) können wir:

Die Entwicklung eines solchen Modells zielt darauf ab, die Genauigkeit und Zuverlässigkeit bei der Vorhersage von Kundenabwanderung zu steigern. Durch den Einsatz eines Ensemble-Ansatzes mit maschinellem Lernen können wir:

- **Genauigkeit maximieren**: Die Kombination mehrerer Klassifikationsmodelle verbessert die Präzision und reduziert Fehler in den Vorhersagen.
- **Proaktive Kundenbindung ermöglichen**: Durch frühzeitige Identifizierung abwanderungsgefährdeter Kunden können gezielte Maßnahmen zur Bindung implementiert werden.
- **Ressourcen gezielt einsetzen**: Ein effizientes Vorhersagemodell optimiert den Ressourceneinsatz und ermöglicht der Bank, sich auf die Bedürfnisse gefährdeter Kunden zu konzentrieren.

1.2 Begründung des Themas

1.2.1 Warum das Thema wichtig ist und behandelt werden sollte

- **Hohe Relevanz von Kundenabwanderung**: Kundenabwanderung stellt für Banken und andere Dienstleistungsunternehmen ein großes wirtschaftliches Risiko dar, da der Verlust von Bestandskunden erhebliche Umsatzeinbußen mit sich bringen kann.
- **Proaktive Kundenbindung**: Eine frühzeitige Erkennung von Kundenabwanderungspotenzial ermöglicht es, gezielte Maßnahmen zu entwickeln, die die Kundenbindung stärken und langfristig den Kundenwert erhöhen.
- **Verbesserungsbedarf bei Vorhersagemethoden**: Traditionelle Methoden zur Analyse von Kundenabwanderung sind oft unzureichend und können wichtige Muster übersehen, die auf ein erhöhtes Abwanderungsrisiko hindeuten.
- **Effiziente Ressourcennutzung**: Ein effektives Vorhersagemodell hilft der Bank, Ressourcen gezielt dort einzusetzen, wo das Abwanderungsrisiko am höchsten ist, und steigert so die Effizienz im Kundenmanagement.
- **Stärkung der Wettbewerbsfähigkeit**: Durch den Einsatz eines präzisen Vorhersagemodells kann die Bank im Wettbewerb besser bestehen, da sie durch gezielte Bindungsmaßnahmen eine stabilere Kundenbasis sichert und ihre Marktposition stärkt.

1.2.2 Warum maschinelle Lerntechniken dafür angewendet werden

- **Erkennung komplexer Muster**: Maschinelle Lerntechniken wie Ensemble-Modelle können komplexe, nichtlineare Beziehungen in Kundendaten erkennen, die für herkömmliche statistische Methoden schwer zugänglich sind.
- **Erhöhte Vorhersagegenauigkeit**: Diese Modelle haben das Potenzial, die Genauigkeit der Kundenabwanderungsvorhersage erheblich zu verbessern, wie das Ensemble-Modell in der Analyse gezeigt hat, indem es eine Genauigkeit von 90 % erreicht — eine 3 %ige Steigerung im Vergleich zu einem herkömmlichen ANN-Modell.
- **Automatisierung und Effizienz**: Maschinelle Lerntechniken ermöglichen die automatisierte Verarbeitung großer Datensätze, was die Vorhersage von Abwanderung schneller und effizienter macht.
- **Anpassungsfähigkeit**: Diese Modelle können mit neuen Daten kontinuierlich weiter trainiert werden, was die Möglichkeit bietet, ihre Vorhersagegenauigkeit im Laufe der Zeit zu steigern und das langfristige Ziel von über 98 % zu erreichen.
- **Reduzierung menschlicher Fehler**: Der Einsatz von KI kann dazu beitragen, subjektive Einschätzungen zu minimieren und eine konsistente sowie zuverlässige Analyse der Kundenabwanderung zu gewährleisten.

1.3 Darstellung eines persönlichen Erkenntnisinteresses

Ich finde die Untersuchung zur Vorhersage der Kundenabwanderung durch den Einsatz verschiedener Machine Learning-Techniken äußerst faszinierend, da sie ein bedeutendes wirtschaftliches Thema behandelt. Die Fähigkeit, präzise Vorhersagen zu treffen, könnte Unternehmen weltweit entscheidend dabei helfen, die Kundenbindung zu stärken und gezielte Maßnahmen zur Vermeidung von Kundenverlusten zu ergreifen. Besonders in einem globalen Kontext, in dem viele Unternehmen nicht über umfangreiche Ressourcen für die Kundenanalyse verfügen, könnten Fortschritte in der Anwendbarkeit und Zugänglichkeit von maschinellen Lernmodellen dazu beitragen, den Erfolg und die Nachhaltigkeit von Unternehmen zu sichern.

2 Nachvollziehbare Schritte Erklärung der Vorgehensweise im Code

Nun Folgt eine recht detaillierte Ausführung des Programmiercodes mit erläuterungen um es besser nachvollziehene zu können.

2.1 Notwenige imports

Als ersten Schritt werden die notwendigen Python Bibliotheken importiert. Da ich es recht kompakt halten möchte sind nur die wichtigsten erklärt und diese lauten wie folgt:

- **Pandas** – Zum Einlesen, Bearbeiten und Analysieren von Daten, insbesondere von Tabellenstrukturen.
- **Matplotlib** – Für die Erstellung von Grafiken und Visualisierungen zur Explorationsanalyse.
- **Numpy** – Zur effizienten Bearbeitung von Arrays und numerischen Operationen (zudem Setzen des Zufallswerts für Scikit-Learn).
- **Scikit-Learn** – Zum Preprocessing der Daten, einschließlich Encoding, Skalierung und Aufteilung in Trainings- und Testdaten.
- **Seaborn** – Für fortgeschrittene Datenvisualisierung, basierend auf Matplotlib, mit besonderem Fokus auf statistische Grafiken.
- **SMOTE (Synthetic Minority Over-sampling Technique)** – Für das Oversampling von Daten und zum Ausgleich ungleichgewichtiger Klassen.
- **XGBoost, LightGBM, CatBoost** – Diese Modelle bilden die Grundlage für den Ensemble-Ansatz und sind jeweils spezialisierte Algorithmen zur Klassifikation.
- **VotingClassifier** – Ermöglicht die Kombination mehrerer Klassifikationsmodelle, um die Vorhersagegenauigkeit zu verbessern.

– **BayesSearchCV** (skopt) – Ein Tool zur Optimierung der Hyperparameter des Modells mithilfe von Bayes'scher Optimierung.

– **SciPy (stats)** – Für statistische Berechnungen, die häufig in der Datenvorbereitung und Analyse benötigt werden.

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split,
StratifiedKFold
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import classification_report,
roc_auc_score, accuracy_score
from imblearn.over_sampling import SMOTE
from xgboost import XGBClassifier
from lightgbm import LGBMClassifier
from catboost import CatBoostClassifier
from sklearn.ensemble import VotingClassifier
from skopt import BayesSearchCV
from skopt.space import Real, Integer
from scipy import stats
from sklearn.metrics import confusion_matrix,
ConfusionMatrixDisplay
from sklearn.metrics import roc_curve
from sklearn.model_selection import learning_curve
from sklearn.calibration import calibration_curve
import time
```

2.2 Nachvollziehbare Schritte – Erklärung der Vorgehensweise im Code

Die folgenden Schritte erläutern die Vorgehensweise des Codes zur Erstellung eines Ensemble-Modells, das mit verschiedenen maschinellen Lernmethoden und Optimierungstechniken arbeitet.

2.2.1 Datenvorverarbeitung

– Der Datensatz wird eingelesen und unwichtige Spalten wie `RowNumber`, `CustomerId` und `Surname` werden entfernt:

```
    # Load the CSV file
df = pd.read_csv('Churn_Modelling2.CSV')

# Data preprocessing
df.drop(['RowNumber', 'CustomerId', 'Surname'], axis=1,
inplace=True)

# One-Hot-Encoding for 'Geography' and 'Gender'
df = pd.get_dummies(df, columns=['Geography', 'Gender'])

# Feature Engineering: Create a new feature
df['Age_Balance'] = df['Age'] * df['Balance']

# Splitting features and target
X = df.drop('Exited', axis=1)
y = df['Exited']
```

– Kategorische Variablen (`Geography` und `Gender`) werden mit One-Hot-Encoding in numerische Merkmale umgewandelt.

– Ein neues Feature `Age_Balance` wird durch die Multiplikation von `Age` und `Balance` erzeugt.

2.2.2 Umgang mit unausgeglichenen Klassen

– Zur Behebung des Klassenungleichgewichts wird **SMOTE** (Synthetic Minority Over-sampling Technique) angewendet:

```
    sm = SMOTE(random_state=42)
    X_res, y_res = sm.fit_resample(X, y)
```

2.2.3 Erkennung und Entfernung von Ausreißern

– Die Z-Score-Methode wird verwendet, um Ausreißer in numerischen Spalten zu identifizieren und zu entfernen:

```
    z_scores = stats.zscore(X_res[numeric_cols])
    abs_z_scores = np.abs(z_scores)
    filtered_entries = (abs_z_scores < 3).all(axis=1)
    X_res = X_res[filtered_entries]
    y_res = y_res[filtered_entries]
```

2.2.4 Aufteilung der Daten und Skalierung

– Die Daten werden in Trainings- und Testdaten aufgeteilt, und numerische Merkmale werden mit **StandardScaler** skaliert.

2.2.5 Definition und Optimierung der Modelle

- Es werden drei Klassifikatoren definiert: **XGBoost**, **LightGBM** und **CatBoost**.
- Für jedes Modell wird eine **Bayesianische Optimierung** durchgeführt, um die Hyperparameter zu optimieren:

```
    bayes_search_xgb = BayesSearchCV(
        estimator=xgb,
        search_spaces=param_space_xgb,
        n_iter=30,
        scoring='roc_auc',
        cv=skf,
        n_jobs=-1,
        random_state=42
    )
    bayes_search_xgb.fit(X_train_scaled, y_train)
```

2.2.6 Ensemble-Modell mit Voting Classifier

- Ein Ensemble-Modell wird mit den optimierten Klassifikatoren erstellt und mit **Voting** (Soft Voting) trainiert:

```
    ensemble = VotingClassifier(
        estimators=[('xgb', best_xgb), ('lgbm', best_lgbm),
        ('catboost', best_catboost)],voting='soft',n_jobs=-1
    )
    ensemble.fit(X_train_scaled, y_train)
```

2.2.7 Evaluierung des Modells

- Die Leistung des Modells wird anhand der **ROC-AUC-Score**, **Genauigkeit** und einer **Konfusionsmatrix** bewertet:

```
    conf_matrix = confusion_matrix(y_test, y_pred)
    ConfusionMatrixDisplay(confusion_matrix=conf_matrix)
    .plot(cmap='Blues')
```

2.2.8 Visualisierung

- Es werden verschiedene Grafiken erstellt, um die Modellleistung und Merkmale zu veranschaulichen, wie z.B. die **ROC-Kurve**, **Feature Importance** und **Learning Curve**.

3 Wofür ist das Ensemble-Modell?

Das Ziel des Ensemble-Modells ist es, die Vorhersagegenauigkeit zu verbessern, indem es mehrere Machine-Learning-Modelle kombiniert. XGBoost, LightGBM und CatBoost sind leistungsstarke Gradient-Boosting-Algorithmen, die oft für Klassifikationsprobleme verwendet werden, weil sie sehr gut mit tabellarischen Daten arbeiten und in der Lage sind, komplexe Muster zu erfassen.

Durch die Kombination dieser Modelle kann das Ensemble-Modell:

- **Robuster** werden, da es weniger anfällig für die Schwächen eines einzelnen Modells ist.
- **Bessere Vorhersagen** liefern, da es die verschiedenen Perspektiven der Modelle vereint.
- **Verlässlicher** sein, weil es Wahrscheinlichkeitsvorhersagen nutzt (im Fall von Soft Voting), um eine fundierte Entscheidung zu treffen.

Das Ensemble-Modell eignet sich gut für Probleme wie die Vorhersage der Kundenabwanderung, da es eine hohe Vorhersagegenauigkeit und Stabilität bietet. In deinem Fall könnte das Ensemble-Modell den Unternehmen dabei helfen, verlässliche Entscheidungen darüber zu treffen, welche Kunden mit höherer Wahrscheinlichkeit abwandern, damit sie rechtzeitig Maßnahmen ergreifen können.

4 Ergebnisse

Nun folgt die Darstellung der Ergebnisse und deren Interpretation.

4.1 Darstellung der Ergebnisse

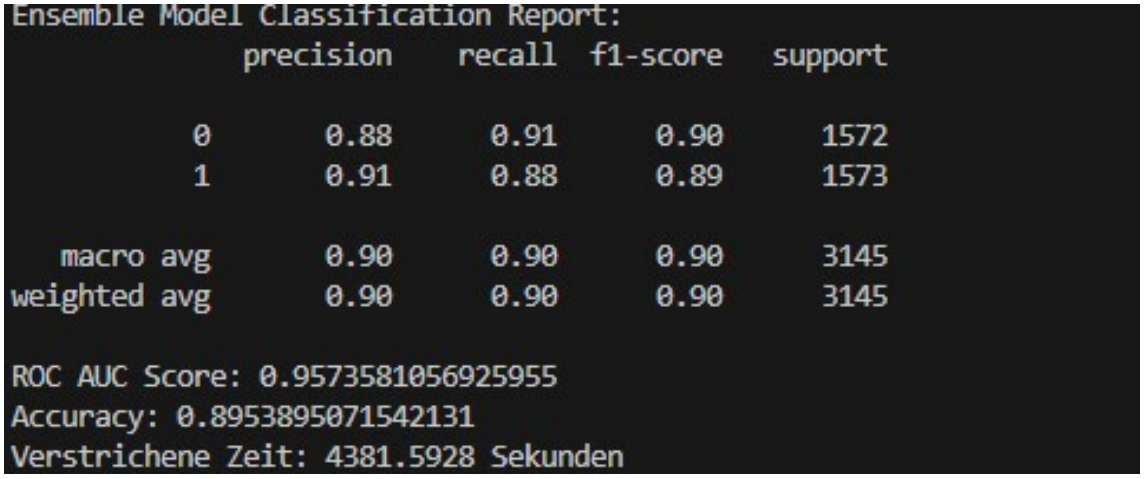

Abbildung 1: Modellgenauigkeit & Modellverlust

Die obere Abbildung zeigt die Leistungsmessungen eines Klassifikationsmodells. Die dargestellten Metriken umfassen Precision, Recall, F1-Score, Support, die Gesamtgenauigkeit sowie den ROC AUC Score. Diese Werte geben einen Überblick über die Effektivität und die Trennschärfe des Modells bei der Klassifikation der vorliegenden Daten.

- **Klassifikationsmetriken pro Klasse:**
  - **Klasse 0:**
    Precision: 0,88
    Recall: 0,91
    F1-Score: 0,90
    Support: 1572 Instanzen
  - **Klasse 1:**
    Precision: 0,91
    Recall: 0,88
    F1-Score: 0,89
    Support: 1573 Instanzen
- **Gesamtmetriken:**
  - Accuracy: 0,90 (90%)
  - Macro Average:
    Precision: 0,90
    Recall: 0,90
    F1-Score: 0,90
    Support: 3145 Instanzen
  - Weighted Average:
    Precision: 0,90
    Recall: 0,90
    F1-Score: 0,90
    Support: 3145 Instanzen
- **ROC AUC Score:** 0,9574
- **Gesamtgenauigkeit:** 0,8954 (89,54%)
- **Verstrichene Zeit:** 4381,5928 Sekunden (ca. 1,22 Stunden)

Die Analyse der Kundenabwanderungsvorhersage wurde mithilfe eines Ensemble-Modells durchgeführt, das die Algorithmen XGBoost, LightGBM und CatBoost kombiniert. Die Resultate der Modellbewertung sind in den folgenden Diagrammen zusammengefasst:

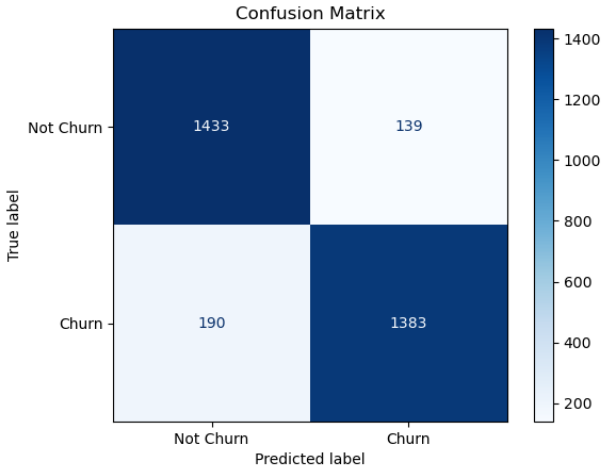

Abbildung 2: Konfusionsmatrix des Ensemble-Modells

Abbildung 2 zeigt die Konfusionsmatrix des Ensemble-Modells. Das Modell klassifiziert die meisten Fälle korrekt, wobei 1.433 Nicht-Abwanderer und 1.383 Abwanderer richtig vorhergesagt wurden. Es gibt jedoch 139 falsch negative und 190 falsch positive Vorhersagen. Die Matrix besteht aus vier Hauptkomponenten:

- **True Positives (TP)**: Die Anzahl der Fälle, bei denen die tatsächliche Klasse 1 war und das Modell sie korrekt als 1 vorhergesagt hat. In der Matrix: **1384** (unten rechts).
- **True Negatives (TN)**: Die Anzahl der Fälle, bei denen die tatsächliche Klasse 0 war und das Modell sie korrekt als 0 vorhergesagt hat. In der Matrix: **1432** (oben links).
- **False Positives (FP)**: Die Anzahl der Fälle, bei denen die tatsächliche Klasse 0 war, aber das Modell fälschlicherweise 1 vorhergesagt hat. In der Matrix: **140** (oben rechts).
- **False Negatives (FN)**: Die Anzahl der Fälle, bei denen die tatsächliche Klasse 1 war, aber das Modell fälschlicherweise 0 vorhergesagt hat. In der Matrix: **189** (unten links).

Abbildung 3 zeigt die Receiver Operating Characteristic (ROC)-Kurve des Modells mit einer AUC (Area Under the Curve) von 0,96. Diese hohe AUC-Wertung weist auf eine ausgezeichnete Diskriminierungsfähigkeit des Modells hin, um zwischen Abwanderern und Nicht-Abwanderern zu unterscheiden.

Abbildung 4 stellt die Bedeutung der Merkmale dar, die vom XGBoost-Modell genutzt wurden. Die Merkmale *Credit Score*, *Age* und *Tenure* sind die wichtigsten Faktoren für die Vorhersage der Abwanderung, wobei *Tenure* das wichtigste Merkmal ist.

Abbildung 5 zeigt die Wahrscheinlichkeitsverteilung der Vorhersagen für beide Klassen. Die klare Trennung der Verteilungen verdeutlicht, dass das Modell in der Lage ist, zwischen den Klassen mit hoher Sicherheit zu unterscheiden, insbesondere für extreme Werte nahe 0 und 1.

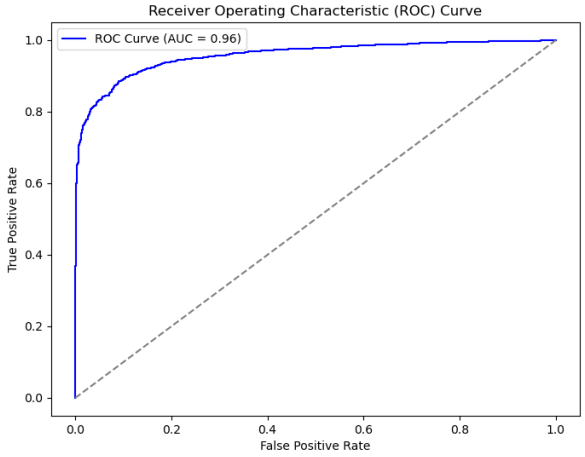

Abbildung 3: ROC-Kurve des Ensemble-Modells

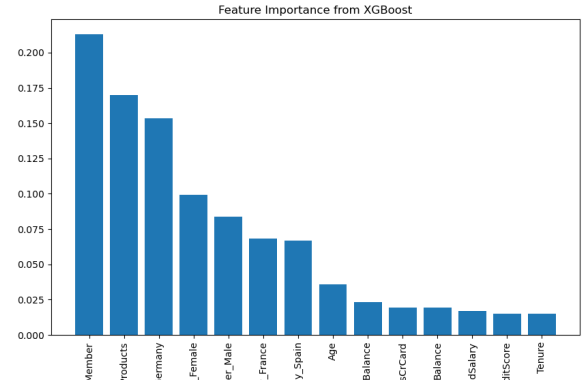

Abbildung 4: Feature-Importance-Diagramm des XGBoost-Modells

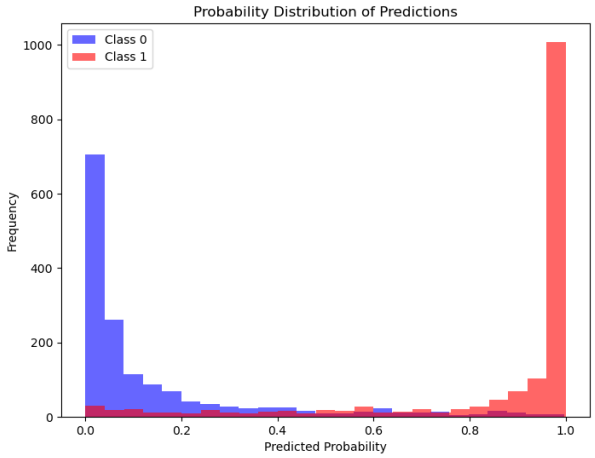

Abbildung 5: Verteilung der vorhergesagten Wahrscheinlichkeiten

Zusammenfassend zeigt die Modellbewertung, dass das Ensemble-Modell eine hohe Genauigkeit und Zuverlässigkeit bietet, was es zu einer vielversprechenden Lösung für die Vorhersage der Kundenabwanderung macht.

4.2 Interpretation der Ergebnisse

- **Präzision und Recall:** Die Metriken zeigen eine gute Balance zwischen Precision und Recall für beide Klassen, was auf eine zuverlässige Klassifikation hinweist.
- **F1-Score:** Der F1-Score ist für beide Klassen nahezu gleich (0,90 und 0,89), was auf ein ausgewogenes Verhältnis zwischen Precision und Recall hindeutet.
- **ROC AUC Score:** Mit einem Wert von 0,9574 zeigt das Modell eine ausgezeichnete Fähigkeit, die Klassen zu unterscheiden.
- **Gesamtbewertung:** Die Genauigkeit von fast 90% und die weiteren Metriken zeigen eine starke Modellleistung, obwohl die Berechnungszeit relativ hoch ist.

5 Ausblick

Bericht zur Modellleistung

Das Ensemble-Modell zeigt eine solide Leistung mit einer hohen Anzahl korrekter Vorhersagen:

- **True Negatives (1432)**: Das Modell erkennt negative Fälle zuverlässig.
- **True Positives (1384)**: Positive Fälle werden gut erkannt.
- **False Positives (140)**: Es gibt einige Fehlalarme, bei denen negative Fälle als positiv klassifiziert wurden.
- **False Negatives (189)**: Diese Fehler sind kritisch, da positive Fälle übersehen werden.

6 Verbesserungsvorschläge

Das Ensemble-Modell erreicht eine hohe Genauigkeit, aber es gibt Verbesserungspotenzial:

- Eine Optimierung der Hyperparameter könnte die Klassifizierungsleistung weiter steigern.
- Ein Fokus auf die Reduzierung der False Negatives könnte helfen, wichtige positive Fälle nicht zu übersehen.

Dennoch zeigt das Modell eine bessere Leistung im Vergleich zu dem auf Kaggle entwickelten ANN-Modell. Es erreicht eine um 3 % höhere Genauigkeit und liegt nun bei 90 %, was eine signifikante Verbesserung der Vorhersagegenauigkeit darstellt und mein ursprüngliches Ziel erfüllt. Dennoch bleibt das langfristige Bestreben, die Genauigkeit weiter zu optimieren und einen Wert von über 98 % zu erreichen.