

# Anforderung und Bewertungsschlüssel für die Projektarbeit im Fach AML

**Stand: Wintersemester 2025, Bugra Turan**

*Die Dokumentation Ihrer Projektarbeit muss alle nachfolgenden Punkte umfassen und klar strukturiert sein. Die Dokumentation kann entweder als separates Dokument (Markdown/PDF) oder integriert in einem Jupyter Notebook vorliegen.*

## 1. Problembeschreibung

- Klare Definition des zu lösenden Problems (Klassifikation oder Regression)
- Relevanz und Anwendungskontext des Problems
- Formulierung der Zielsetzung: Was soll das System konkret leisten?
- Erwartete Herausforderungen und deren Bedeutung

## 2. Datenquelle und Datenbeschreibung

- Herkunft der Daten (Link/Referenz zum Dataset)
- Lizenz und Verfügbarkeit der Daten
- Anzahl der Datenpunkte und Features
- Beschreibung der wichtigsten Features (inkl. Datentypen)
- Verteilung der Zielklassen (bei Klassifikation) bzw. Wertebereich (bei Regression)

## 3. Explorative Datenanalyse (EDA)

- Visualisierung der Datenverteilung (Histogramme, Boxplots, etc.)
- Korrelationsanalyse zwischen Features und Zielvariable
- Identifikation von Mustern, Trends und Auffälligkeiten
- Analyse von Class Imbalance (falls vorhanden)

## 4. Datenvorverarbeitung

- Behandlung fehlender Werte (NaN, Inf)
- Erkennung und Behandlung von Outliers
- Normalisierung/Standardisierung
- Feature Engineering: Erstellung neuer Features (falls durchgeführt)
- Encoding kategorialer Variablen (One-Hot, Label Encoding, etc.)

## 5. Data Augmentation (optional)

- Beschreibung der Augmentation-Techniken
- Auswirkung auf die Datenmenge und -qualität

## 6. Modellauswahl und -architektur

- Begründung für die Auswahl der Modelle
- Detaillierte Beschreibung der finalen Modellarchitektur
- Hyperparameter und deren Wahl

## 7. Training

- Aufteilung der Daten (Train/Validation/Test-Split)
- Training-Konfiguration (Batch Size, Learning Rate, Epochs, etc.)
- Verwendete Loss-Function und Optimizer
- Techniken zur Vermeidung von Overfitting (Regularisierung, Dropout, Early Stopping, etc.)
- Visualisierung des Trainingsverlaufs (Loss- und Metrik-Kurven)

## 8. Evaluation und Ergebnisse

- Definition und Begründung der verwendeten Evaluation-Metriken
- Quantitative Ergebnisse aller getesteten Modelle (Vergleichstabelle)
- Detaillierte Analyse des besten Modells (Confusion Matrix, Classification Report, etc.)
- Fehleranalyse: Wo macht das Modell Fehler und warum?
- Visualisierung der Ergebnisse (z.B. Predictions vs. Ground Truth)

## 9. Diskussion und Fazit

- Interpretation der Ergebnisse: Wurden die Ziele erreicht?
- Kritische Reflexion: Was hat gut funktioniert, was nicht?
- Limitationen der gewählten Ansätze
- Mögliche Verbesserungen und Ausblick

## Zusätzliche Anforderungen

- **Code-Qualität:** Der Code muss gut strukturiert, kommentiert und nachvollziehbar sein
- **Reproduzierbarkeit:** Alle Schritte müssen im GIT Repository reproduzierbar dokumentiert sein
- **Requirements:** Eine requirements.txt Datei muss vorhanden sein
- **README:** Das Repository muss eine README.md mit Anleitung zur Ausführung enthalten

### **Wichtiger Hinweis:**

*Eine Projektarbeit ist nicht gescheitert, wenn die gewünschte Performance nicht erreicht wird. Entscheidend ist die wissenschaftlich fundierte Analyse und Diskussion der Hintergründe und Ergebnisse. Ein sauberer Vergleich verschiedener Ansätze mit kritischer Reflexion ist genauso wertvoll wie das Erreichen hoher Metriken.*