# Anleitung zur Verwendung des System-Novelty-Detection-Systems

## Einführung

Der ASIC Use Case im SMTIH-Projekt beschäftigt sich mit der Frage, wie Messwerte und Daten, die von Patienten erhoben wurden, welche auf Intensivstationen behandelt wurden, dazu verwendet werden können, eine algorithmische Früherkennung von ARDS zu ermöglichen. Außerdem soll die automatisierte Untersuchung der Patientendaten den Arzt bei der Behandlung unterstützen.

Bei der Untersuchung solcher großen Datenmengen ist jedoch zu beachten, dass hin und wieder Fehler in den Daten auftreten können, wie beispielsweise Sensorfehler. Um eine hohe Qualität der Datenanalyse zu gewährleisten, müssen solche Fehler erkannt werden. Dafür existieren verschiedene Algorithmen, die nach einer vorhergehenden Annotation einiger Datenpunkte Fehler erkennen können.

In seiner Masterarbeit entwickelte Jan van Essen eine Software, welche eine grafische Oberfläche bereitstellt, in der Datenpunkte entsprechend annotiert werden können, die Ausführung verschiedener solcher Algorithmen möglich ist und das Ergebnis visualisiert wird. Im Folgenden wird die Bedienung der Software kurz erläutert.

## Start des Programms

Das Programm wurde mit der Programmiersprache Python geschrieben. Zu dem Zeitpunkt, als diese Anleitung erstellt wurde, liegt das Programm ausschließlich in einer nicht-kompilierten Version vor, d.h. es gibt keine ausführbare Datei. Um es auszuführen, muss daher Python auf dem Computer installiert sein. Die neuesten Versionen zu den jeweiligen Betriebssystemen können unter folgendem Link heruntergeladen werden: <https://www.python.org/downloads/>

Zudem werden einige Python-Module benötigt, die separat installiert werden müssen. In der Datei *requirements.txt* sind diese Pakete alle aufgelistet. Der Programmstart schlägt mit einer entsprechenden Fehlermeldung fehl, wenn eines der Pakete nicht installiert wurde. Ein Modul kann installiert werden, indem in der Kommandozeile folgender Befehl eingegeben wird: *python -m pip install <paketname>.*

Eine detaillierte Anleitung zum Installieren von Modulen findet sich hier: <https://docs.python.org/3/installing/index.html>

Da es bisher keine ausführbare Datei gibt, muss das entsprechende Python-Skript, welches als Einstiegspunkt des Programms dient, direkt über die Konsole aufgerufen werden. Dieses Skript heißt *ndas.py* (im Ordner NDAS). Um es aufzurufen, muss man also in der Kommandozeile in den Ordner NDAS wechseln und den Befehl *python ndas.py* ausführen. Wenn Python und alle benötigten Module erfolgreich installiert wurden, startet das Programm und es öffnet sich die Hauptansicht.

## Grundlegendes zur Bedienung

Nach dem Programmstart befindet man sich in der Hauptansicht. Zentraler Bestandteil ist ein Koordinatensystem, in der geladene Messwerte und Daten im zeitlichen Verlauf dargestellt werden.

Am oberen Bildschirmrand befindet sich eine Menüleiste, über welchen man Patientendaten importieren kann sowie der aktuelle Zustand gespeichert und der Plot in verschiedene Formate exportiert werden kann. Unter der Menüleiste befinden sich einige Registerkarten, mit denen zwischen den verschiedenen Modulen der Software umgeschaltet werden kann.

### Annotation

In der Annotationsansicht, die beim Programmstart geöffnet wird, befindet sich in der Mitte ein Koordinatensystem, indem die geladenen und ausgewählten Patientendaten dargestellt werden. Mithilfe der linken und rechten Maustaste sowie des Mausrads kann man die Ansicht beliebig verschieben, zoomen sowie strecken und stauchen. Mithilfe eines Linksklicks auf einen Datenpunkt wird dieser Punkt selektiert. Zieht man bei gedrückter STRG-Taste mit der rechten Maustaste ein Fenster über mehrere Punkte, werden alle diese Punkte ausgewählt. Oberhalb des Koordinatensystems befinden sich Steuerelemente, mithilfe derer ausgewählte Datenpunkte annotiert werden können.

Auf der rechten Seite, innerhalb der Box *Analysis Settings,* kann ein Novelty-Detection-Algorithmus ausgewählt und ausgeführt werden. Im Bereich *Data Slicing* kann die Menge der Datenpunkte auf ein bestimmtes Intervall eingegrenzt werden. Im Bereich *Visualization* können einige grafische Einstellungen am Plot vorgenommen werden. Zudem kann hier der darzustellende Plot ausgewählt werden, falls mehrere Parameter in den geladenen Daten verfügbar sind.

Nach dem Ausführen eines Novelty-Detection-Algorithmus werden die Datenpunkte verschiedenfarbig markiert. Die verschiedenen Datenpunkte bedeuten im Einzelnen folgendes:

* *Grau:* Vom Algorithmus nicht verwendete Datenpunkte,
* *Türkis:* Als Trainingsdaten verwendete Datenpunkte,
* *Blau:* Als normal erkannte Datenpunkte
* *Gelb* und *rot:* Als Anomalie erkannte Datenpunkte.

### Data Inspector

In der *Data Inspector-*Ansicht werden die importieren Patientendaten komplett in tabellarischer Form dargestellt. Die Datensätze können durch Klick auf die entsprechenden Spaltennamen sortiert werden.

### Data Generator

Hier können Testdaten generiert werden, mithilfe derer die Funktionalitäten des Programms ausprobiert werden können. Hierbei können verschiedene Anomalien eingebaut werden.

### Benchmark

Das Benchmarksystem baut auf dem Datengenerator auf und ermöglicht es, verschiedene Novelty-Detection-Algorithmen zu testen. Im ersten Schritt werden die Testdaten generiert, im zweiten Schritt werden die zu testenden Algorithmen ausgewählt und anschließend erhält man eine Übersicht, anhand derer zu erkennen ist, wie gut der Algorithmus Anomalien erkannt hat. Dies funktioniert logischerweise nur auf generierten Testdaten, bei denen die Anomalien bereits im Vorfeld bekannt sind.

### Statistics

In diesem Bereich werden einige statistische Informationen über die importierten Patientendaten grafisch dargestellt.