Bachelorarbeit
Einsatz von Prozessanalysen für maschinelle Lernverfahren zur Anomalieerkennung in Multibeam Daten

-

Sommersemester 2022

Erstbetreuer: Tobias Ziolkowski

Manuel Krebs

3. September 2022

Inhaltsverzeichnis

Abbildungsverzeichnis

Kapitel 1

Einführung

- Multibeam Daten -¿ Definition
- Prozessanalysen für maschinelle Lernverfahren -¿ Definition und warum wichtig
- Anomalien -¿ Definition, Beispiele und Erklärung, dass Anomalie = Outlier
- Sagen, was ich mit dem Paper erreichen möchte -¿ Forschungsfrage
- https://sci-hub.hkvisa.net/10.3182/20130902-3-cn-3020.00044 -¿ Sliding Winfdow

Outliers are the patterns which are not in the range of normal behavior [6]

Moving window pattern!
Chandola, V., Banerjee, A., Kumar, V.: Outlier detection: a survey

"what techniques were used in the past and how machine learning could help us now and in the future for a better bathymetric data processing"

Unterschied lokale und globale Outlier!

Outlier detection (or data cleaning) is an important step in data processing because, if an outlier data point is used during data mining, it is likely to lead to inaccurate outputs -¿ Ramírez-Gallego, S.; Krawczyk, B.; García, S.; Wo ´zniak, M.; Herrera, F. A survey on data preprocessing for data stream mining: Current status and future directions. Neurocomputing 2017, 239, 39–57. [CrossRef]

Kapitel 2

Literaturübersicht

Andere Anwendungsgebiete!

Erklären, wie Multibeamdaten erfasst werden Erklären, wie Outlier meistens erkannt werden.

Kapitel 3

Methodik

3.1 Datenerhebung

3.1.1 Erstellung der Roh-Daten

Die Datenerhebung fand durch eine Expedition statt. Diese startete am 20.12.2019 um 10:35 Uhr in Mindelo, Kap Verde. Die Expedition endete am 13.01.2020 um 15:33 Uhr. Über den Verlauf des Monats segelte das Expeditions-Schiff Maria S. Merian insgesamt 18233.7111 km und sammelte bathymetrische Multibeam-Daten 3.1.

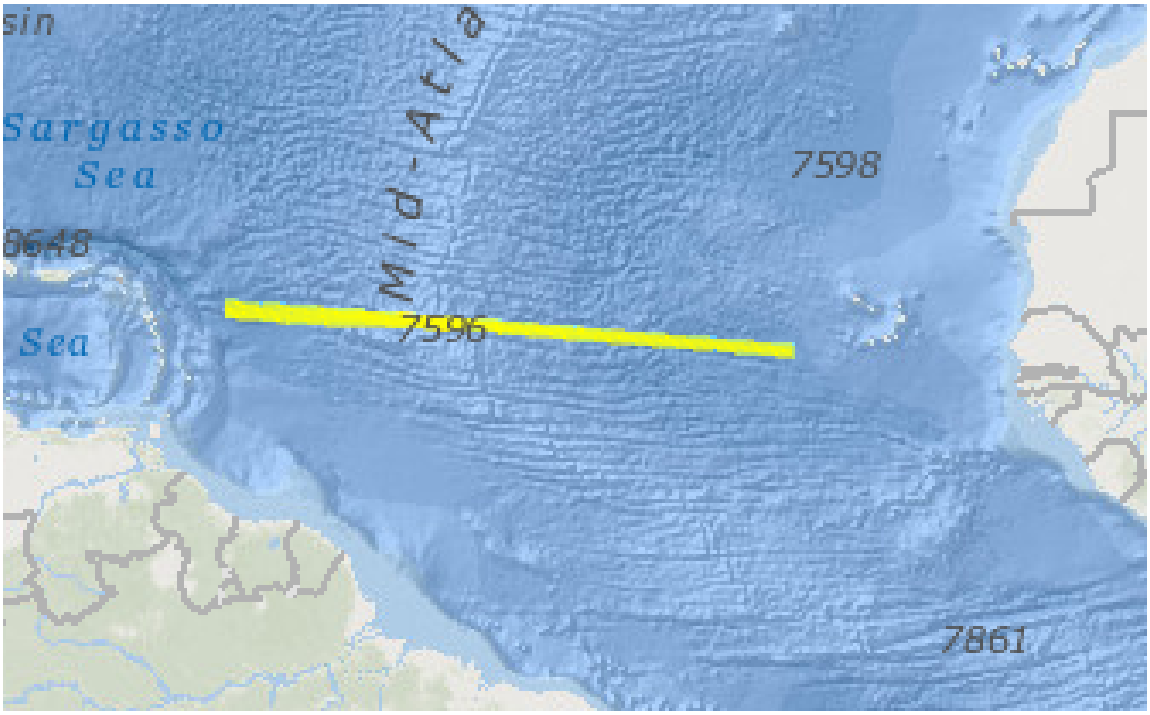

Abbildung 3.1: Messdaten

3.1.2 Datentransformation

Um nun Anomalien in den Rohdaten erkennen zu können, müssen diese zuvor in kleinere Subsequenzen unterteilt werden. Der Grund besteht darin, dass Anomalien nicht mit Blick auf den gesamten Datensatz erkannt werden können, da diese immer relational zu ihren näheren Nachbarn sind. Existiert beispielsweise ein Berg in den Messdaten, so unterscheiden sich die Tiefenwerte an der Spitze des Berges stark zu den Tiefenwerten, die am Boden des Berges aufzufinden sind. Trotz dieser Höhenunterschiede können demnach nicht zwingend Anomalien erkannt werden.
Folglich werden die Daten in Subsequenzen aufgeteilt, mittels dessen dann die Merkmals-Berechnung stattfinden wird.

3.2 Datenverarbeitung

3.2.1 Statistische Merkmale

Interquartilsabstand

Der Interquartilsabstand (im folgenden mit IQR angekürzt) ist ein Streuungsmaß, welcher in statistischen Analysen dabei hilft, durch Verteilungen Rückschlüsse über einen Datensatz zu ziehen. Visuell werden diese meist durch einen Boxplot dargestellt, welche zu den am weitesten verbreiteten Werkzeugen in der statistischen Praxis, insbesondere in der Phase der explorativen Datenanalyse [4].

Funktionsweise Zunächst wird das erste und das dritte Quartil berechnet. Der IQR bildet sich nun aus der Differenz des der beiden Quartile, also IQR = Quartil 3 - Quartil 1. Um dies als graphische Repräsentation zu verdeutlichen zeigt Grafik 3.2 einen Boxplot, welcher die 'z'-Koordinate der ersten 1000 Messdaten veranschaulicht. Der IQR lässt sich nun durch die blau eingefärbte Fläche zeigen, welche 50 Prozent der Messdaten enthält. Alle Messwerte, welche außerhalb oberen und unteren Grenze liegen, lassen sich nun als potenzielle Outlier identifizieren. Diese Grenzen werden typischerweise auf das 1,5-fache des IQR gesetzt. Die untere Grenze lässt sich also durch Quartil 1 - (1.5 * IQR) und die obere Grenze durch Quartil 3 + (1.5 * IQR) berechnen [6].

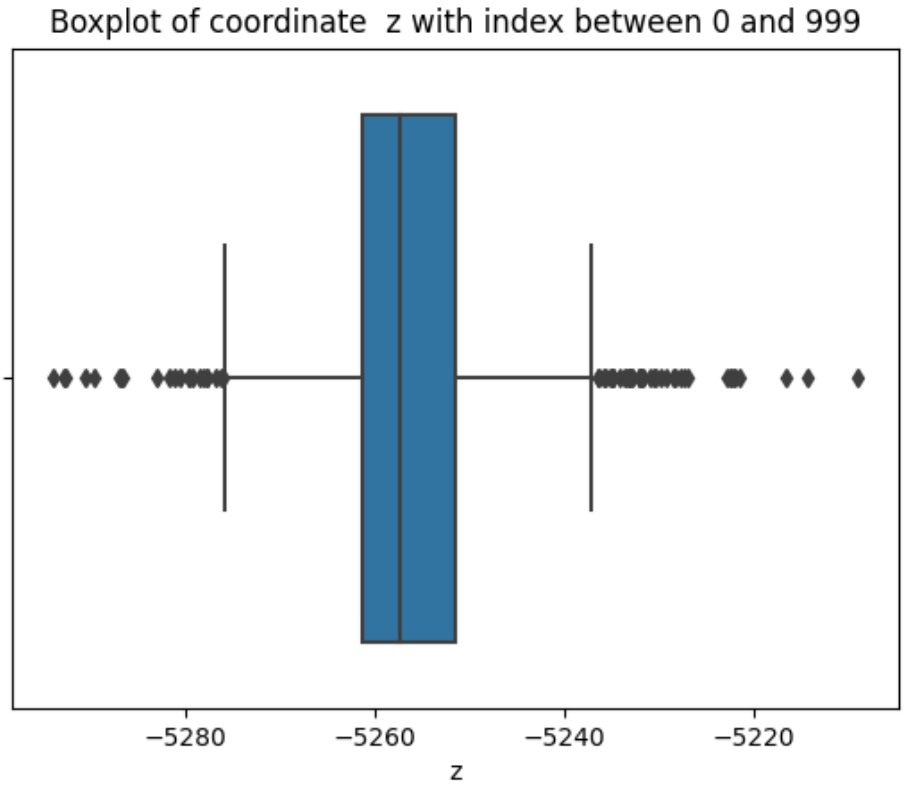

Abbildung 3.2: Beispiel eines Boxplots

Wie in der Grafik 3.2 nun zu erkennen ist, befinden sich einige der Messwerte nur knapp außerhalb der Grenzen. bei den Werten weit außerhalb der Grenzen ist die Wahrscheinlichkeit am höchsten, dass diese Ausreißer sind.

Anwendung Bevor die Methodik angewendet werden kann muss der Datensatz angepasst werden. Dies geschieht mithilfe dem zuvor beschrieben Ansatz des Moving-Window-Patterns. Nachdem der Datensatz aufgeteilt wurde, wird die Methode mit den einzelnen Chunks aufgerufen. Hier hat man nun mehrere Möglichkeiten:

- Händische Identifizierung von Outliern mithilfe der Ausgabe mehreren Boxplots.
- Berechnung der oberen und unteren Grenze und automatische Filterung aller Werte, welche sich außerhalb befinden.

3.2.2 Tiefenbasierende Merkmale

https://hands-on.cloud/using-python-and-isolation-forest-algorithm-for-anomalies-detection/

Isolation Forest

Um Anomalien in Datensätzen zu erkennen werden meist erst normale Instanzen profiliert, anhand deren man dann nicht übereinstimmende Instanzen als Anomalien zu identifizieren [5]. Der Isolation Forest konzentriert sich hingegen auf das Erkennen der Anomalien selbst, indem es zunutze nimmt, dass Anomalien 1) die Minderheit des Datensatzes sind, und 2) sich die Attributwerte stark von denen normaler Instanzen unterscheidet. Sie sind somit "wenig und unterschiedlich"[5]. Isolation Forests basieren wie Random Forests auf Entscheidungsbäumen QUELLE. Und da es hier keine vordefinierten Labels gibt, handelt es sich um ein unüberwachtes Modell QUELLE.

Funktionsweise Der Isolation Forest berechnet zu jeder Instanz einen Anomalie-Wert. Nach der zufälligen Auswahl eines Merkmals werden die Daten in einer Baumstruktur zufällig verarbeitet. Dies wird nun rekursiv fortgesetzt, bis alle Messwerte isoliert sind. Der Anomalie-Wert berechnet sich nun aus der Länge des Pfades zwischen der Wurzel und dem zu bewertenden Knoten. Anomalien werden daran erkannt, dass diese bereits nach wenigen Iterationen isoliert wurden und somit einen geringeren Anomalie-Wert besitzen als Knotenpunkte, die keine Outlier darstellen QUELLE..

Vorteile Firstly, building iTrees only need to select subset of the training set randomly, the research result show that the reasonable number of sub-samplings is set to 256, which is a relative small number and reduce the swamping and masking effects effectively. Secondly, iForest utilizes no distance or density measures to detect anomaly, this eliminates computational cost significantly compared to the distance-based methods and density-based methods. Thirdly, iForest has a linear time complexity with low constant and a low memory requirement. Last but not the least, iForest algorithm is based on the ensemble idea, even if the efficiency of some iTrees are not very high, the ensemble algorithm always can turn the weak algorithm into strong algorithm. https://towardsdatascience.com/isolation-forest-the-anomaly-detection-algorithm-any-data-scientist-should-know-1a99622eec2d

3.2.3 Clusterbasierende Merkmale

Local Outlier Factor

Local Outlier Factor (fortgehend mit LOF abgekürzt) beschreibt die lokale Suche nach Anomalien. Die Methode beruht auf einer dichtebasierten Technik [2]. Die Idee ist, jedem Objekt ein Ausreißergrad zuzuordnen, welcher davon Abhängig ist, wie isoliert besagtes Objekt in Bezug auf die umgebende Nachbarschaft ist [3].

Funktionsweise

3.2.4 ???

One-Class Support Vector Machine

https://datagy.io/python-support-vector-machines/

3.3 Prozessanalyse

3.3.1 Eventlog-Generierung

Um nun eine Prozessanalyse der generierten Anomalie-Erkennungs-Merkmale durchzuführen müssen zuvor Eventlogs generiert werden. Da es sich hier allerdings nicht um typische Events handelt, müssen die Datensätze dafür zuvor angepasst werden.

CaseID	Timestamp	Activity	Resource
-28.199190189471913 14.564852566063955 -523...	1900-01-01 00:00:00	LOF: Likely no Outlier	LOF: -1.0533796392823758
-28.199190189471913 14.564852566063955 -523...	1900-01-01 00:00:01	Iforest: Likely no Outlier	Iforest: -0.1612632467625628
-28.199190189471913 14.564852566063955 -523...	1900-01-01 00:00:02	IQR: Likely no Outlier	IQR: -1
-28.199190189471913 14.564852566063955 -523...	1900-01-01 00:00:03	SVM: likely no Outlier	SVM: 4.944487258270534
-28.19921213810358 14.564884852271142 -5251....	1900-01-01 00:00:04	IQR: Likely an Outlier	IQR: 1
-28.19921213810358 14.564884852271142 -5251....	1900-01-01 00:00:05	Iforest: Likely an Outlier	Iforest: 0.0264538683796972
-28.19921213810358 14.564884852271142 -5251....	1900-01-01 00:00:06	LOF: Likely no Outlier	LOF: -1.211642956710735
-28.19921213810358 14.564884852271142 -5251....	1900-01-01 00:00:07	SVM: Maybe Outlier	SVM: 22.84723981093644
-28.19922575052645 14.56456310925572 -5234.4...	1900-01-01 00:00:08	LOF: Maybe Outlier	LOF: -1.0205977903195351
-28.19922575052645 14.56456310925572 -5234.4...	1900-01-01 00:00:09	IQR: Likely an Outlier	IQR: 1
-28.19922575052645 14.56456310925572 -5234.4...	1900-01-01 00:00:10	Iforest: Likely no Outlier	Iforest: -0.1382158358088508
-28.19922575052645 14.56456310925572 -5234.4...	1900-01-01 00:00:11	SVM: likely no Outlier	SVM: 11.674844537910506
-28.19923885773837 14.56467679924186 -5257.7...	1900-01-01 00:00:12	IQR: Likely an Outlier	IQR: 1
-28.19923885773837 14.56467679924186 -5257.7...	1900-01-01 00:00:13	SVM: Maybe Outlier	SVM: 22.84759346741373

Abbildung 3.3: Eventlog

Case-ID Die Case-ID wird generiert, indem die Latitude, Longitude und die Tiefe der Messdaten zu einem String konkateniert werden.

Aktivität Die Aktivität wird, bestehend aus dem Merkmals-Namen und der zuvor bestimmten Klassifizierung (ja, nein, vielleicht), zusammengestellt.

Zeitstempel Da es sich nur um Koordinaten-Werte handelt, bietet der Zeitstempel nur eine Reihenfolge der zu betrachtenden Eventlogs und wird somit folgendermaßen erstellt: Beginne mit dem Start-Datum 01.01.1999 um 00:00:00 Uhr. Nun wird bei der Erstellung des Eventlogs zu jeden Folgedatum eine Sekunde hinzuaddiert, sodass eine künstliche Reihenfolge generiert wird.

Ressource Die Ressource baut sich zusammen aus dem Namen des berechneten Merkmals, konkateniert mit dem Anomalie-Score, welcher bei der Berechnung der Anomalieerkennung entstanden ist.

3.3.2 Analyse der Eventlogs

Zur Analyse der Eventlogs wurde das Programm Disco von Fluxicon verwendet [1]. Es erlaubt das automatische Erstellen von verschiedenen Prozess-Maps, welche direkt aus den Rohdaten generiert wird.

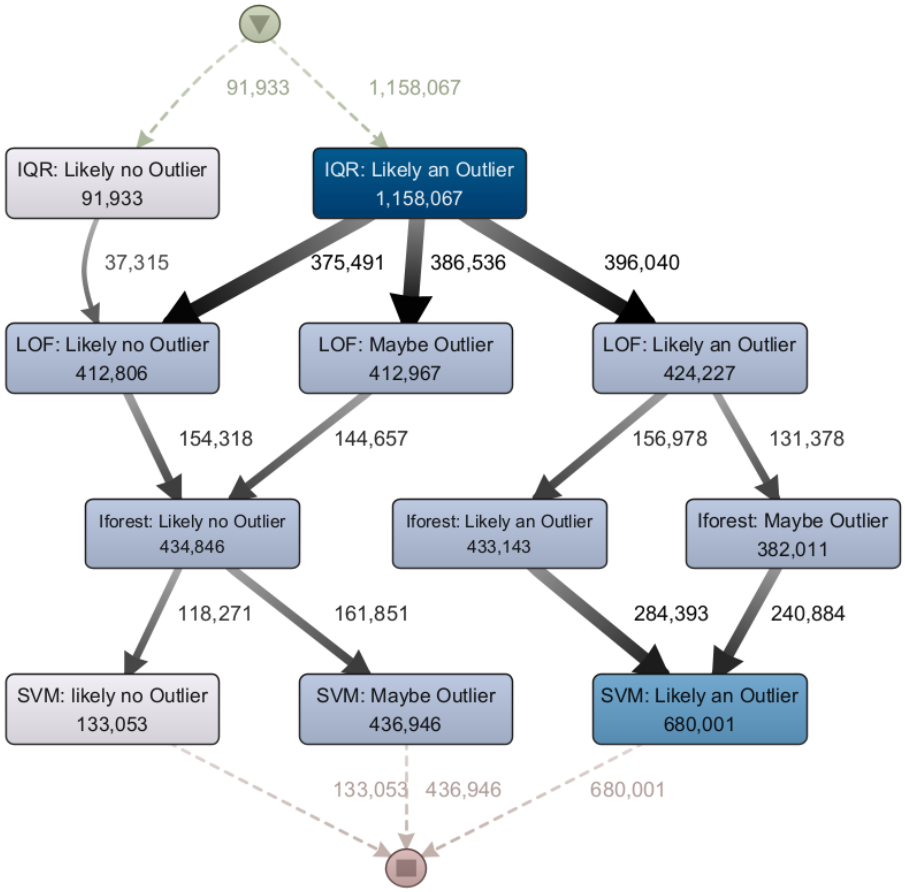

Abbildung 3.4: Prozessmodell

Kapitel 4

Ergebnisse

Erklärung der Ergebnisse und Analyse durch Disco im Anschluss.

Kapitel 5

Diskussion

Für Weiterführende Arbeit! https://sci-hub.hkvisa.net/10.1145/3338840.3355641

Kapitel 6

Fazit

Literaturverzeichnis

[1] Process Mining and Automated Process Discovery Software for Professionals - Fluxicon Disco.

[2] Omar Alghushairy, Raed Alsini, Terence Soule, and Xiaogang Ma. A Review of Local Outlier Factor Algorithms for Outlier Detection in Big Data Streams. 5(1):1.

[3] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. LOF: Identifying density-based local outliers. 29(2):93–104.

[4] Y. H. Dovoedo and S. Chakraborti. Boxplot-Based Outlier Detection for the Location-Scale Family. 44(6):1492–1513.

[5] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation Forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422. IEEE.

[6] H. P. Vinutha, B. Poornima, and B. M. Sagar. Detection of Outliers Using Interquartile Range Technique from Intrusion Dataset. In Suresh Chandra Satapathy, Joao Manuel R.S. Tavares, Vikrant Bhateja, and J. R. Mohanty, editors, *Information and Decision Sciences*, volume 701 of *Advances in Intelligent Systems and Computing*, pages 511–518. Springer Singapore.