

SYSTEMANALYSE UND
-OPTIMIERUNG

Carl von Ossietzky Universität Oldenburg

Thema:

MaGellan

Abschlussdokumentation

Abteilung Wirtschaftsinformatik:
Systemanalyse und -optimierung

Themensteller: Prof. Dr. Axel Hahn

Betreuer: Arne Lamm

vorgelegt von: Omid Brunzlow
Jean Cedric Envina Ondoa
Dennis Jankowski
Lennart Krühsel
Jan Mentjes
Rabea Müller
Simon Münster
Andreas Saum

Abgabetermin: 07. April 2019

Inhaltsverzeichnis	
Inhaltsverzeichnis	III
Abbildungsverzeichnis.....	II
Tabellenverzeichnis	V
1 Einleitung	- 1 -
1.1 Motivation.....	- 1 -
1.2 Problemstellung	- 1 -
1.3 Zielsetzung.....	- 1 -
1.4 Vorgehensweise	- 1 -
2 Projektmanagement.....	- 3 -
2.1 Vorgehensmodell	- 3 -
2.2 Sprint.....	- 3 -
2.2.1 Sprint-Erzeugung.....	- 4 -
2.2.2 Weekly Workflow	- 4 -
2.2.3 Evaluation.....	- 4 -
2.3 Meilensteine.....	- 5 -
2.4 Projektablaufpläne	- 5 -
2.5 Querschnittsfunktionen	- 5 -
2.6 Rollenverteilung.....	- 5 -
3 Anforderungen	- 8 -
4 Grundlagen	- 9 -
4.1 AIS und Radar	- 9 -
4.2 Rechtliche Grundlagen	- 10 -
4.3 KNIME	- 14 -
4.3.1 KNIME Einleitung	- 14 -
4.3.2 Nutzung von KNIME	- 15 -
5 Technische Infrastruktur	- 16 -
5.1.1 Aufzeichnung der AIS Daten	- 16 -
5.1.2 Aufzeichnung der Radar Daten	- 17 -
6 Aufteilung des Projektes	- 18 -
7 Umsetzung.....	- 19 -
7.1 Wetterdaten.....	- 19 -
7.1.1 Datenbank Schema	- 21 -
7.1.2 Wetterdaten Verarbeitung.....	- 27 -
7.1.3 Update Workflow	- 28 -
7.1.4 Abfrage der Wetterdaten	- 29 -
7.2 Datenmodell.....	- 30 -

7.2.1	Szenario	- 31 -
7.2.2	Plot.....	- 31 -
7.2.3	Measurement	- 32 -
7.2.4	Target.....	- 33 -
7.3	ETL	- 33 -
7.3.1	Extract Raw Data and set ID	- 34 -
7.3.2	Divide the data (AIS, Radar)	- 34 -
7.4	Datenassoziation und -fusion.....	- 39 -
7.4.1	Datenassoziation.....	- 39 -
7.4.2	Datenfusion.....	- 40 -
7.4.3	Fusionsergebnisse.....	- 40 -
7.5	Bereinigung der Daten	- 42 -
7.6	Destination Aufbereitung.....	- 44 -
7.7	Kartenmaterial	- 49 -
7.8	Wetterkonzept.....	- 50 -
7.9	Regelkonzept	- 54 -
7.10	Schiffsklassen	- 56 -
7.11	Schiffsbegegnungssituationen filtern.....	- 60 -
7.11.1	Schiffsbegegnungstyp bestimmen	- 63 -
7.12	Klassifikation von Schiffsbegegnungssituationen zur Detektion von anomalen Begegnungssituationen.....	- 72 -
7.13	Klassifikation von Schiffsbegegnungssituationen zur Erkennung von Parametereinflüssen	- 76 -
7.14	Statistiken zu den Begegnungssituation	- 79 -
7.15	Normalmodell zur Repräsentation üblicher Manöver und Tracks.....	- 82 -
7.15.1	Detektion von Manöverpunkte	- 82 -
7.15.2	Erstellung eines Manövernetzes	- 87 -
7.15.3	Nutzung eines Manövernetzes.....	- 89 -
7.16	Heatmaps	- 91 -
7.17	Verkehrsaufkommen.....	- 94 -
7.18	Bayessches Netz	- 94 -
7.19	Entscheidungsbaum (Decision Tree)	- 99 -
8	Xtreme Programming und Code Conventions	- 102 -
8.1.1	Xtreme Programming	- 102 -
8.1.2	Maven Conventions	- 108 -
9	Fazit und Ausblick	- 114 -
9.1	Fazit	- 114 -
9.2	Ausblick.....	- 116 -

Literatur	i
10 Anhang	iv
10.1 Datenmodell	iv
10.2 Entscheidungsbaum	v
10.3 Heatmaps der verschiedenen Begegnungssituationen	xiii
10.4 Benutzerhandbuch	xvi
10.4.1 Benutzerhandbuch ETL-Prozess	xvi
10.4.2 Begegnungssituationen	xvii
10.4.3 Erstellung eines Manövernetzes	xxiv
10.4.4 Benutzung eines Manövernetzes	xxvii
10.5 Anforderungen	xxx
10.5.1 Allgemeine Anforderungen	xxx
10.5.2 Allgemeine Anforderungen (GA).....	xxxvi
10.5.3 Anforderungen an den Hauptprozess	xxxvii
10.5.4 Anforderungen für die Überführung in die Simulation	xlvi
10.5.5 Datenbank-Anforderungen	xlviii
10.5.6 Allgemein	xlviii
10.5.7 Data Collection	liii
10.5.8 Anforderung an die Datenbank (Wetterdaten)	liii
10.5.9 Anforderung an ein Datenbank-Tool:.....	lxiii
10.5.10 Scenario Mining	lxx
10.5.11 Data Collection: Anforderungen	lxxii
10.5.12 Funktionale Anforderungen:.....	lxxii
10.5.13 Nicht Funktionale Anforderungen:.....	lxxvi
10.5.14 Anforderungen Data Processing	lxxix
10.5.15 Anforderungen.....	lxxix
10.5.16 Datenanreicherung	lxxxiii
10.5.17 Anforderungen.....	lxxxiii
10.5.18 Datenfusion.....	lxxxvii
10.5.19 Scenario Mining	xc
10.5.20 Nichtfunktionale Anforderungen:.....	cxxiv
10.6 Meilensteine	cxxv
10.7 Projektablaufpläne des ersten halben Jahres	cxxxiii
10.8 Eigenständigkeitserklärungen	cxxxvi

Abbildungsverzeichnis

Abbildung 2-1: Vorgehensmodell Scrum (WinRed.es 2018)	- 3 -
Abbildung 5-1: Aufzeichnung AIS Daten	- 16 -
Abbildung 5-2: Aufzeichnung Radar Daten	- 17 -
Abbildung 7-1: Datenbank Schema	- 22 -
Abbildung 7-2: Wetterschema Aufbau	- 22 -
Abbildung 7-3: Wetter Rest-Service Anfrage	- 29 -
Abbildung 7-4: Wetterdaten Beispielabfrage Zeiten	- 29 -
Abbildung 7-5: Wetterdaten Beispielabfrage Bereiche	- 30 -
Abbildung 7-6: EER Modell	- 31 -
Abbildung 7-7: Beispiel Track Fusion	- 41 -
Abbildung 7-8: Ungeglättete AIS-Daten	- 44 -
Abbildung 7-9: Durch Gaussian-Kernel-Filter geglättete AIS-Daten	- 44 -
Abbildung 7-10: Harbour Tabelle	- 48 -
Abbildung 7-11: Geojson-Datei mit einem Gebietsobjekt	- 49 -
Abbildung 7-12: Wahrscheinlichkeitsdichtefunktion von Windgeschwindigkeit- -	52 -
Abbildung 7-13: Winkeleinzeichnung für die Windrichtung	- 53 -
Abbildung 7-14: Neuer Inputeintrag mit dazugehörigem Fuzzysset	- 54 -
Abbildung 7-15: exemplarische Kerndichteschätzung der Länge und des Tiefgangs einer Klasse	- 58 -
Abbildung 7-16: Schiffsbegegnungstypen	- 60 -
Abbildung 7-17: Ablauf zur Filterung von Begegnungssituationen	- 61 -
Abbildung 7-18: Detektion eines CPA und Filterung der zeitlich nahen Trackpoints	- 63 -
Abbildung 7-19: Winkelbereiche Begegnungstypen	- 64 -
Abbildung 7-20: Beispiele Wahrscheinlichkeitsberechnung	- 65 -
Abbildung 7-21: Winkelbereiche für Differenzen der COG-Werte	- 67 -
Abbildung 7-22: Erforderliche Kombinationen für einen Begegnungstyp	- 68 -
Abbildung 7-23: Weitere Unterscheidung der Begegnungstypen	- 69 -
Abbildung 7-24: Richtung der Schiffe der beispielhaften Begegnungssituation- -	70 -
Abbildung 7-25: Wahrscheinlichkeitsverlauf Track 455 mit Differenz der COG- Werte	- 71 -
Abbildung 7-26: Wahrscheinlichkeitsverlauf Track 455 mit Bearing	- 72 -
Abbildung 7-27: Betrachtete Gewässerbereiche	- 73 -
Abbildung 7-28: Die zehn häufigsten Klassen	- 74 -
Abbildung 7-29: Kerndichteschätzung der Klasse mit den meisten Begegnungssituationen	- 75 -
Abbildung 7-30: Kerndichteschätzung häufigste Klasse mit Flussgebiet als Gewässertyp	- 76 -
Abbildung 7-31: Kerndichteschätzungen zu veränderten Wetterbedingungen..	- 77 -
Abbildung 7-32: Kerndichteschätzung zu veränderter Windbedingung	- 78 -
Abbildung 7-33: Kerndichteschätzung bei veränderter Tageszeit	- 79 -
Abbildung 7-34: Verteilung der Gewässerarten	- 80 -
Abbildung 7-35: Verteilung der Begegnungstypen	- 80 -
Abbildung 7-36: Verteilung der Sensitivitätsklassen	- 81 -
Abbildung 7-37: Eingehaltene Regeln in den Monaten Dezember 2018 und Januar 2019	- 81 -
Abbildung 7-38: Exemplarisches Manövernetz im Bereich der Elbe	- 82 -

Abbildung 7-39: Pseudocode vom CuSum Algorithmus	- 83 -
Abbildung 7-40: Detektion von steuerbordseitigen Manövern	- 85 -
Abbildung 7-41: Detektion von backbordseitigen Manövern	- 86 -
Abbildung 7-42: Exemplarischer COG-Graph mit dazugehörigem CuSum Graph.- 86 -	
Abbildung 7-43: CuSum Algorithmus angewendet an einem Track auf der Elbe...- 87 -	
Abbildung 7-44: exemplarisches Cluster mit der dazugehörigen zweidimensionalen Kerndichteschätzung.....	- 90 -
Abbildung 7-45: Heatmap der Durchschnittsgeschwindigkeiten mit 500000 Datensätzen.....	- 92 -
Abbildung 7-46: Heatmap der Positionsdaten mit 500000 Datensätzen.....	- 93 -
Abbildung 7-47: Heatmap der Szenarien	- 94 -
Abbildung 7-48: Bayessches Netz.....	- 97 -
Abbildung 7-49: Confusion Matrix	- 98 -
Abbildung 7-50: Graph des Bayesschen Netzes	- 98 -
Abbildung 7-51: Eintrittswahrscheinlichkeitstabelle für die Betrachtung von KVR 1 von Track 1	- 99 -
Abbildung 7-52: Entscheidungsbaum in KNIME	- 100 -
Abbildung 7-53: Entscheidungsbaum	- 101 -
Abbildung 7-54: Confusion Matrix, Entscheidungsbaum.....	- 101 -
Abbildung 8-1: TDD Verfahren (Machado 2018).....	- 104 -
Abbildung 8-2: Hauptpakete KNIME-Plugin in MaGellan (Beispiel GribWeather Knoten)	- 105 -
Abbildung 8-3: Exportieren Pakete innerhalb von KNIME-Plugin	- 106 -
Abbildung 8-4: Importieren Pakete aus einem Plugin	- 107 -
Abbildung 8-5: allgemeine Pakete des Plugins Gribweather	- 110 -
Abbildung 8-6: Pakete mit package-info.....	- 112 -
Abbildung 10-1: Datenmodell	iv
Abbildung 10-2: Vaterknoten mir erster und zweiter Ebene.....	v
Abbildung 10-3: Linke Seite	vi
Abbildung 10-4: Linke Seite Fortsetzung	vi
Abbildung 10-5: Linke Seite Ende	vii
Abbildung 10-6: Rechte Seite des linken Pfades	vii
Abbildung 10-7: Rechte Seite des linken Pfades Ebene 2	viii
Abbildung 10-8: Rechte Seite des linken Pfades Ebene 2 links.....	viii
Abbildung 10-9: Rechte Seite des linken Pfades Ebene 2 rechts	ix
Abbildung 10-10: Linke Seite des rechter Pfades Ebene 2 links	x
Abbildung 10-11: Linke Seite des rechter Pfades Ebene 2 rechts.....	xi
Abbildung 10-12: Rechte Seite der zweite Ebene linker Pfad Ebene drei rechts..	xii
Abbildung 10-13: Linke Seite der zweite Ebene linker Pfad Ebene drei rechts ...	xii
Abbildung 10-14: Rechte Seite der zweiten Ebene des rechten Pfades Ebene drei links	xiii
Abbildung 10-15: Heatmap Crossing	xiii
Abbildung 10-16: Heatmap Following.....	xiv
Abbildung 10-17: Heatmap Head-On.....	xiv
Abbildung 10-18: Heatmap Overtaking	xv
Abbildung 10-19: Heatmap Passing	xv
Abbildung 10-20 Workflow-Variablen des ETL-Prozesses.....	xvi
Abbildung 10-21: Konfiguration eines Date&Time Inputs.....	xviii

Abbildung 10-22: Bedienung des Knotens MinDistanceFilter	xix
Abbildung 10-23: Einstellen der Workflowvariablen	xxi
Abbildung 10-24: EncounterVisualizationWorkflow	xxiii
Abbildung 10-25: Workflow zur Generierung eines Manövernetzes – Maneuvernet Creator	xxv
Abbildung 10-26: Ausschnitt aus dem Metaknoten Data preparation	xxv
Abbildung 10-27: Workflow zur Detektion von Anomalien - Maneuernet Creator	xxvii
Abbildung 10-28: Dialog des CuSum Knotens	xxviii
Abbildung 10-29: Metaknoten zum Laden des Manövernetzes	xxix
Abbildung 10-30: Output des “Anomal Route Detection” Workflows.....	xxx

Tabellenverzeichnis

Tabelle 7-1: Maritime Wetterdaten	- 19 -
Tabelle 7-2: allgemeine Wetterdaten.....	- 20 -
Tabelle 7-3: Eingetragene Wetterdaten	- 21 -
Tabelle 7-4: Allgemeine Wetter Tabellen	- 24 -
Tabelle 7-5: Hilfstabellen	- 25 -
Tabelle 7-6: Providerspezifische Tabellen	- 27 -
Tabelle 7-7: Vergleich Datenfusion	- 42 -
Tabelle 7-8: Veränderung Destination cleaner.....	- 48 -
Tabelle 7-9: Verwendete Wetterparameter.....	- 51 -
Tabelle 7-10: Einteilung der Sichtweite in Klassen	- 53 -
Tabelle 7-11: Einteilung der umgesetzten Regeln in Soft- und Hardrules.....	- 55 -
Tabelle 7-12: Einteilung der Schiffe nach Manövrierbarkeit mit dazugehörigem prozentualen Anteil der Klassen.....	- 58 -
Tabelle 7-13: Median der Länge pro Klasse	- 59 -
Tabelle 7-14: Median des Tiefgangs pro Klasse	- 59 -
Tabelle 7-15: Mögliche Parametrisierung von CuSum	- 84 -
Tabelle 7-16: Kriterien für das Bayessche Netz	- 96 -
Tabelle 10-1: Erfüllte Meilensteine	cxxxix
Tabelle 10-2: Nicht oder nur teilweise erfüllte Meilensteine	cxxxix

1 Einleitung

Der Bereich Kooperative Mobile Systeme (KMS) des OFFIS e.V. erfasst seit dem Jahr 2016 von Schiffen gesendete Daten für Teile des norddeutschen Gewässers. Ferner werden dem OFFIS vom Bundesamt für Schifffahrt und Hydrographie (BSH) und vom Deutschen Wetterdienst (DWD) historische Wetterdaten zur Verfügung gestellt.

1.1 Motivation

Ungefähr 80% der Kollisionen im Schiffsverkehr basieren auf menschlichen Fehlern (vgl. Pietrzykowski und Uriasz 2009, 93). Um solche Kollisionen zu vermeiden, werden Assistenzsysteme wie das *Maritime Traffic Alert and Collision Avoidance System* (MTCAS) vom OFFIS entwickelt.

1.2 Problemstellung

Für das Testen solcher Assistenzsysteme werden reale Szenarien von Schiffen benötigt. Im Idealfall werden den Assistenzsystemen solche Szenarien zur Verfügung gestellt, welche Beinahekollisionen von Schiffen simulieren. Das Bereitstellen von Beinahekollisionen stellt sich hierbei als eine Herausforderung dar, da diese in einer historischen Datenmenge von Schiffsdaten zunächst identifiziert werden müssen.

1.3 Zielsetzung

Die im Jahr 2018 startende, ein Jahr laufende Projektgruppe **Maritime Gruppe** für die **Entwicklung intelligenter Simulationsszenarien** hat sich zum Ziel gesetzt mithilfe von historischen AIS-, Radar- und Umweltdaten aus dem maritimen Umfeld abweichendes Schiffsverhalten zu detektieren, welches auf Beinahekollisionen hinweisen kann.

1.4 Vorgehensweise

Zunächst werden die historischen AIS-, und Radardaten, welche unstrukturiert vorliegen, mithilfe eines ETL-Prozesses in ein neues Datenmodell übertragen und

fehlerhafte Daten markiert. Ferner sollen die vom BSH und DWD durch Dateien bereitgestellten Wetterdaten in der Datenbank gespeichert werden, sodass diese für weitere Analysen abgefragt werden können. Zuletzt soll abweichendes Schiffsverhalten durch verschiedene aufgestellte Normalmodelle und die Abweichung von Normen detektiert werden.

2 Projektmanagement

Dieses Dokument enthält Struktur, Aufbau, Vorgehensmodell und weitergehende Informationen für die Projektgruppe MaGellan bzgl. des Projektmanagements.

2.1 Vorgehensmodell

Als Vorgehensmodell wird das *ScrumBut*-Modell genutzt. In der Abbildung 2-1 ist *Scrum* zu sehen. Dabei werden die ersten drei Artefakte von *Scrum* nicht umgesetzt. Und beginnen direkt mit der Sprint Planung. Dabei gibt es einen inkrementellen Abschnitt, welcher für die dynamische Umsetzung des Projektes sorgt.

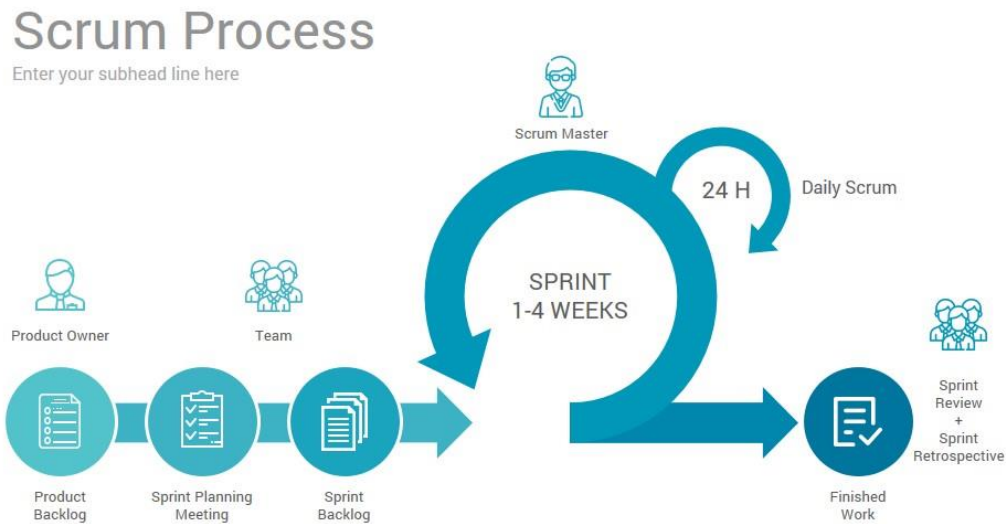


Abbildung 2-1: Vorgehensmodell Scrum (WinRed.es 2018)

2.2 Sprint

Ein Sprint ist ein zeitlich begrenzter, in sich geschlossener Aufgabenkatalog, welcher Aufgaben umfasst, die aus den Anforderungen hervorgehen (Kriegisch o. J.). Dabei ist zu beachten, dass lieber kleinere und mehrere Aufgaben erstellen, als große, komplexe *Tasks*. Wenn *Tasks* nicht erfüllt werden, können diese Aufgaben entweder in den nächsten Sprint gezogen oder der Sprint verlängert werden. Falls während des Sprints Mitglieder merken, dass sie ihre Aufgaben

nicht erfüllen können, so melden diese sich beim Projektleiter. Ein Sprint kann verlängert oder verkürzt werden. Dies entscheidet der Projektleiter.

2.2.1 Sprint-Erzeugung

Ein neuer Sprint wird vom Projektmanager gestartet, nachdem die neuen *Tasks* vom Team zusammengetragen worden sind. Dies geschieht immer, wenn ein vorheriger Sprint beendet wurde. Die neuen *Tasks werden* in Jira eingebettet und der Sprint gestartet.

2.2.2 Weekly Workflow

Im wöchentlich stattfindenden *Weekly Workflow* stellen die Projektmitglieder, ähnlich wie beim *Daily Scrum*, den jeweiligen Fortschritt vor. Mit dieser Methode soll der Sprint-Fortschritt erfasst werden. So kann im Notfall eingegriffen werden. (vgl. (Kriegisch o. J.))

2.2.3 Evaluation

Der Sprint geht in die Evaluation, wenn die Zeit, welche zuvor festgelegt wurde, abgelaufen ist. Der Projektleiter schaut anhand der *Definition of Done* (DoD), ob das Ergebnis eines *Tasks* dem entspricht, was abgemacht wurde. Nun können drei Szenarien entstehen:

- Alle *Tasks* sind gemäß DoD als fertig zu erachteten:
 - Projektleiter geht weiter zur Sprint-Definition der nächsten Iteration
- Die allermeisten *Tasks* (bis auf ca. 2 – 3 Aufgaben) wurden gemäß DoD als fertig erachtet
 - *Tasks*, welche nicht fertig gestellt wurden, werden in den neuen Sprint überführt, welcher vom Projektleiter definiert wird.
- Es wurden mehr Aufgaben nicht erledigt, als zulässig und es ist nicht zu gewährleisten, dass der nächste Sprint durch diese Aufgaben nicht in Verzug gerät
 - Der Projektleiter verlängert den Sprint um jeweils eine Woche

2.3 Meilensteine

Die Meilensteine befinden sich im Anhang 10.6 und besitzen ab dem zweiten Halbjahr einen Verantwortlichen. Dies sorgt für ein erhöhtes Verantwortungsgefühl eines einzelnen Projektgruppenmitglied der Gruppe gegenüber. Ebenfalls gibt es dadurch einen direkten Ansprechpartner bei Rückfragen oder Problemen mit Inhalt eines Meilensteines.

2.4 Projektablaufpläne

Die Projektablaufpläne für das erste halbe Jahr befinden sich ebenfalls im Anhang 10.7 und bilden den Zeitplan der Projektgruppe MaGellan ab.

2.5 Querschnittsfunktionen

Die Dokumentations-, Recherche- und nicht-produkt-bezogene Aufgaben werden nicht einem Sprint zugeordnet, sondern parallel erstellt. Ggf. können hier *Tasks* in Jira erstellt werden, welche keinem Sprint zugeordnet werden. Diese Aufgaben können somit eine Bearbeitungszeit aufweisen, welche länger oder kürzer als ein Sprint ist.

2.6 Rollenverteilung

Abschließend werden die Rollen beschrieben, welche zur Durchführung des Projektes benötigt werden.

Anforderungsbeauftragter:

In den Aufgabenbereich des Anforderungsbeauftragten fällt das Erstellen des Anforderungsdokuments und die Pflege dessen. Dazu zählt zum einen das Erstellen einer Vorlage des Dokumentes, mitsamt seinem Inhalt sowie Einleitung, Anforderungen und Glossar und zum anderen das Festlegen des Aufbaus der Anforderungen. Das Dokument muss außerdem stetig aktualisiert werden.

Außendarstellungsbeauftragter:

Die Aufgabe des Außenbeauftragten ist es, die Außendarstellung der Projektgruppe zu pflegen. Die Hauptaufgabe ist hierfür der Aufbau und die Pflege einer Webseite für die Projektgruppe. Bei der Webseite ist das Ziel, das Projekt und die Mitglieder näher vorzustellen sowie neugierig auf die Thematik der Projektgruppe zu machen.

Architekturbeauftragte:

Der Softwarearchitekt kümmert sich um die Integration und den Aufbau verschiedener Programme und Tools. Für die Datenanalyse von Schiffsdaten werden verschiedenste Prozesse der Teilgruppen erstellt, welche koordiniert, synchronisiert und integriert werden müssen. Daher erstellt der Architekturbeauftragte eine geeignete Struktur, um die Verwaltung und Erweiterung der Programme zu vereinfachen.

Git-Admin:

Die Aufgabe des Git-Admins ist die Verwaltung der Git-Repositories für die Projektgruppe MaGellan. Daher gibt dieser Nutzungshinweise für die *Repositories*, erstellt eine geeignete Ordnerstruktur und kümmert sich um das *Mergen* von *Branches*. Gegebenenfalls hält der Git-Admin zusätzlich Kontakt zum Server-Admin für die Erstellung von neuen *Repositories* und deren Nutzerverwaltung (bspw. Vergabe von Rechten).

Dokumentenbeauftragte:

Die Dokumentenbeauftragte kümmert sich um die Erstellung einer Dokumentationsvorlage sowie die Organisation der Projektdokumentation.

Projektleiter:

Der Projektleiter legt in Absprache mit den Projektgruppenmitgliedern ein Vorgehensmodell für das Projektmanagement fest und führt eine Struktur zur Umsetzung des Vorgehensmodells ein. Der Projektleiter plant in Absprache mit

den anderen Gruppenmitgliedern die Sprints und die Aufgaben, die innerhalb der Sprints erledigt werden müssen. Des Weiteren überprüft der Projektleiter, ob die Projektgruppenmitglieder ihre Aufgaben im Jira pflegen und ordnungsgemäß dokumentieren. In einem wöchentlichen Treffen diskutiert der Projektleiter den aktuellen Stand der Kleingruppen und deren Fortschritt mit den Teilprojektleitern. Nähere Informationen über das Projektmanagement selbst können im Kapitel Projektmanagement nachgelesen werden.

Terminbeauftragte:

Die Aufgabe der Terminbeauftragten unterteilt sich in die Urlaubsplanung und die Planung von wichtigen Terminen wie die Zwischenpräsentation.

Qualitätsbeauftragter:

Die Aufgabe des Qualitätsbeauftragten ist es die Qualität des Codes hochzuhalten. Hierzu gehört die Verwendung von Standards beim Schreiben des Quellcodes, Reviews für den geschriebenen Code, das Konfigurationsmanagement, Verwalten von Releases, die erfolgreiche Integration von verschiedenen Komponenten, und die Verifizierung des Systems beim Testen.

Protokollverantwortlicher:

Die Rolle des Protokollverantwortlichen beinhaltet die Erstellung und Weiterentwicklung einer Protokollvorlage sowie eine kontinuierliche Vereinheitlichung und Kontrolle der erstellten Protokolle.

Während des Projektes fand ein Wechsel des Projektmanagers statt. Dieser hat eine Verschlankung des Projektmanagements vorgenommen. Confluence wird im zweiten Halbjahr lediglich für die Retrospektive benutzt. Sämtliche Aufgaben und Meilensteine werden mit Jira geplant und kontrolliert. Die Meilensteine werden als *Story* abgelegt. Wenn das Fälligkeitsdatum erreicht wurde, wird überprüft ob alle Aufgaben, welche dem Meilenstein angehören fertiggestellt wurden. Erst nach Abnahme wurden diese in Jira auf *Done* gesetzt.

3 Anforderungen

In den ersten zwei bis drei Monaten der Projektphase wurden die Anforderungen an die Ergebnisse der Projektgruppe und der Teilgruppen sowie der zu Nutzenden Entwicklungsumgebungen definiert. Hierfür wurde ein Anforderungsdokument/Anforderungskatalog erstellt, der sich im Anhang befindet. Das Anforderungsdokument gliedert sich dabei wie folgt: Anforderungen an die Gesamtarchitektur, Data Collection, Data Processing, Scenario Mining und an die Datenbank (vgl. Anhang 10.5).

Die Anforderungen an ein Projekt stellen einen wichtigen Teil für die Erarbeitung von Ergebnissen dar. Ohne ein gut strukturiertes und formuliertes Anforderungsdokument, ist es nicht möglich ein Projekt ordnungsgemäß abzuschließen. Im Verlauf des Projektes wurden die Anforderungen an die Teilgruppen sowie die an die Gesamtarchitektur in den Gruppenmeetings immer mal wieder durchgesprochen und je nachdem, ob sie erfüllt wurden oder nicht, gekennzeichnet bzw. ergänzt, verworfen oder angepasst. Kontrolliert wurde dies durch den Anforderungsbeauftragten der Projektgruppe.

4 Grundlagen

Im Folgenden werden AIS- und Radar-Daten vorgestellt und auf die rechtlichen Grundlagen eingegangen. Abgeschlossen wird dieses Kapitel mit einem Überblick über KNIME.

4.1 AIS und Radar

Im maritimen Bereich wird mit Daten aus dem *Automatic Identification System* (AIS) und dem Radarbereich gearbeitet. AIS ist hierbei ein Funksystem, welches im Ultrakurzwellenbereich kommuniziert und seit 2000 als Standard dient. Das System ermöglicht eine präzise Verfolgung von Schiffen und eine Kommunikation zwischen Schiffen. Nicht jedes Schiff ist verpflichtet einen AIS Sender an Bord zu haben. Ab einer gewissen Größe oder bei speziellen Schiffstypen (Passagierschiffe) ist ein Sender jedoch verpflichtend. (Fischer 2016, 32)

Schiffe mit AIS senden unterschiedliche Daten. Diese lassen sich in statische, dynamische und reisespezifische Daten aufteilen, welche in unterschiedlichen Frequenzen versendet werden. Statische und reisebezogene Daten werden beispielhaft alle 6 Minuten versendet. Zu den statischen Daten gehören MMSI-Nummer, IMO-Nummer, der Schiffsname, die Abmessungen (Länge, Breite) des Schiffs, sowie der Schiffstyp. Außerdem lagen der Projektgruppe noch die Positionsdaten des AIS Senders vor. Zu den Reisebezogenen Daten gehören das Reiseziel, der Tiefgang sowie die erwartete Ankunftszeit. Zudem gehört die Gefahrgutklasse der Ladung dazu, welche der Projektgruppe jedoch nicht vorlagen. Dynamische Daten (Position, Kurs über Grund, Geschwindigkeit über Grund, Steuerkurs, Navigationsstatus, Kursänderungsrate) werden in kürzeren Abständen gesendet und hängen mit der Geschwindigkeit des Schiffes zusammen. (Fischer 2016, 33ff.)

Als Alternative bzw. Ergänzung zu AIS wird in der Schifffahrt mit Radardaten gearbeitet. Im Vergleich zu AIS, werden keine genauen Positionen versendet, sondern jedes Schiff kann durch seine Radar-Antenne umliegende Schiffe und Gegenstände detektieren und deren Abstand und Kurs ermitteln. (Fischer 2016,

36) Innerhalb der Projektgruppe lagen die Radardaten soweit vor, dass die Daten auf korrekte Position, Geschwindigkeit und Kurs runtergebrochen wurden und somit mit den AIS Daten verglichen werden konnten.

4.2 Rechtliche Grundlagen

Durch die in Kapitel 5 vorgegebene Infrastruktur können nicht alle rechtlichen Bestimmungen, die auf See gelten, in der Analyse berücksichtigt werden. Im Folgenden wird auf die umzusetzenden Regeln eingegangen und auf welche Organisation und rechtliche Grundlage diese gründen.

Dabei werden nur die Regeln umgesetzt, dessen Nichteinhaltung per Beobachtungsdaten wahrgenommen werden kann. Im genaueren, welche die eigene Position, die Position zu anderen und die Geschwindigkeit betreffen. Hierbei wird zwischen den unterschiedlichen Größen der Schiffe unterschieden, da diese durch ihre unterschiedliche Manövrierfähigkeit anderen Regeln unterliegen. Für die Festlegung der Regeln in dem Untersuchungsgebiet ist für den Bereich der Nordsee die „Internation Maritim Organization“ zuständig (bmvit 2018, 3). Dort gelten die COLREGS, bei denen auf die Regeln:

- **6: „Sichere Geschwindigkeit“** mit den Einflussparametern:
 - i. Die Sichtverhältnisse
 - ii. Die Verkehrsdichte
 - iii. Die Manövrierfähigkeit
 - iv. Die Wind-, Seegangs- und Strömungsverhältnisse
 - v. Der Tiefgang im Verhältnis zur vorhandenen Wassertiefe
- **8: „Kollisionsvermeidendes Verhalten“** mit den Möglichkeiten:
 - i. Änderung des Kurses
 - ii. Geschwindigkeit reduzieren bzw. Fahrt komplett stoppen
 - iii. Beide Maßnahmen unter der Prämisse, dass die Änderung des Kurses oder die Geschwindigkeitsänderung klar für das andere Fahrzeug erkennbar sind

- **9: „Enges Fahrwasser“** mit dem Vorgehen:
 - i. Sich im engem Fahrwasser weitmöglichst am steuerbordseitigen Rand zu halten
 - ii. Segelfahrzeuge und Fahrzeuge < 20 Meter dürfen Fahrzeuge, die auf das Fahrwasser angewiesen sind, nicht behindern
 - iii. Fischende Fahrzeuge dürfen Fahrzeuge, die in einem solchen Fahrwasser fahren, nicht behindern
 - iv. Enges Fahrwasser querende Fahrzeuge dürfen Fahrzeuge, die dem Fahrwasserverlauf folgen, nicht behindern
- **10: „Verkehrstrennungsgebiete“** mit dem Vorgehen:
 - i. Sich in die richtige Richtung auf dem entsprechenden Einbahnweg zu bewegen
 - ii. Sich von der Trennlinie so weit wie möglich fern zu halten
 - iii. Einfahren und Ausfahren nur am Ende des Gebietes oder in einem möglichst kleinen Winkel zum Fahrwasser (<10°)
 - iv. Queren vermeiden, wenn doch dann im 90° Winkel

Gute Sichtverhältnisse:

- **13: „Überholen“**
 - i. Es gilt als Überholen, wenn man sich einem anderen Fahrzeug in einem Winkel von mehr als 22,5° achterlicher querab nähert
 - ii. Durch eine spätere Änderung der Peilung bleibt die Situation ein Überholmanöver
- **14: „Entgegengesetzte Kurse“**
 - i. Es gilt als entgegengesetzter Kurs, wenn man in einer Linie zu einem entgegenkommenden Fahrzeug steht und sich annähert
 - ii. Beide Fahrzeuge müssen dann nach Steuerbord ausweichen und begegnen sich dann auf der Backbordseite
- **15: „Kreuzende Kurse“**
 - i. Das Fahrzeug, welches das andere zur Steuerbordseite hat, muss ausweichen
 - Bugseitiges Kreuzen soll vermieden werden

- **16: „Ausweichpflichtige Schiff“**
 - i. Frühes und durchgreifendes Handeln

- **17: „Kurshaltende Schiff“**
 - i. Kurs und Geschwindigkeit beibehalten
 - ii. Darf Ausweichen, wenn das andere Fahrzeug sich nicht an Regel 16 hält

- **18: „Ausweichpflicht“**
 - i. Ein Maschinenfahrzeug muss ausweichen:
 - einem manövrierunfähigen Fahrzeug
 - einem manövrierbehinderten Fahrzeug
 - einem fischenden Fahrzeug
 - einem Segelfahrzeug
 - ii. Ein Segelfahrzeug muss ausweichen:
 - einem manövrierunfähigen Fahrzeug
 - einem manövrierbehinderten Fahrzeug
 - einem fischenden Fahrzeug
 - iii. Ein fischendes Fahrzeug muss ausweichen:
 - einem manövrierunfähigen Fahrzeug
 - einem manövrierbehinderten Fahrzeug

Schlechte Sichtverhältnisse:

- **19: „Veränderung des Fahrverhaltens“**

eingegangen wird („COLREG“ 2018).

Diese Regeln werden durch die SeeSchStrO für Flüsse ergänzt und im Einzelfall sogar ersetzt. Dabei richtet sich der Fokus auf den vierten Abschnitt „Fahrregeln“ (ELWIS 2002). Die beiden Paragraphen, die im Besonderen an dieser Stelle Erwähnung finden, sind:

- **§23 „Überholen“**
 - i. Linksseitiges Überholen ist vorgesehen
 - ii. Das zu überholende Fahrzeug muss seine Fahrt verringern

- iii. Überholverbote gelten bei:
 - Engen Stellen und unübersichtlichen Krümmungen
 - Vor und innerhalb von Schleusen und der Einfahrt in den **NOK**
- **§25 „Vorfahrt der Schifffahrt im Fahrwasser“**
 - i. Fahrwasserverlaufsfolgende Fahrzeuge haben Vorrang vor Fahrzeugen, die:
 - Das Fahrwasser queren
 - In das Fahrwasser einlaufen
 - In dem Fahrwasser drehen
 - ii. Fahrzeuge im Fahrwasser haben Vorrang vor Fahrzeugen, die in das Fahrwasser einlaufen wollen
(ELWIS 2002)

Die Geschwindigkeitsbegrenzungen auf der Elbe werden aus der 9. Elbvertiefung vom 23.04.2012 unter II. Anordnungen Nr. 5.3.1 ff., (Seite 65) gezogen (AG 2014).

4.3 KNIME

Es folgt eine kurze Einleitung in das Data Mining Tool KNIME, sowie KNIME Server.

4.3.1 KNIME Einleitung

KNIME (Konstanz Information Miner) ist ein von der Universität Konstanz entwickeltes Data Mining Tool. Dies ist ein Open-Source-Projekt und basiert auf der Entwicklungsumgebung Eclipse. Mit KNIME können graphisch Abläufe (Workflows) erstellt werden. Diese werden durch eine aneinander-reihung von Knoten erstellt. („KNIME Open Source Story“ o. J.) Jeder Knoten stellt eine einzelne Einheit dar und führt eine einzelne Aktion aus. Jeder Knoten beinhaltet eine Beschreibung und einen Knoten-Typen, welche beschreiben, welche Aktion dieser durchführt. Als Typ Beispiele sind zu nennen Manipulation (Manipulieren), Source (Daten-Quelle) und Visualizer (Visualisierung).

Workflows werden genutzt, um Daten vorzuverarbeiten (*data preprocessing*), Daten-Analysen durchzuführen (*data mining*) und zu visualisieren. Durch die Nutzung von Eclipse als Basis von KNIME, wird die von Eclipse genutzte OSGi Implementierung (Equinox) als Plugin-Framework genutzt. Somit kann KNIME erweitert werden, indem neue (oder eigene) Knoten als Plugin geladen werden. (Bernardo Najlis 23:37:26 UTC)

Neben KNIME wurde zusätzlich die Server-Variante genutzt. KNIME-Server ist eine kommerzielle Software zur Nutzung von KNIME auf einem Server. Es vereinfacht die Zusammenarbeit von Projektmitgliedern sowie die Automatisierung und Verwaltung von erstellten *Workflows*. Durch ein zentrales *Workflow-Repository* können *Workflows* hochgeladen, geteilt und ausgeführt werden. („KNIME Server | KNIME“ o. J.) Zusätzlich können *Workflows* automatisiert werden, indem diese zu bestimmten Zeiten oder festgelegten Zeitintervallen ausgeführt werden.

4.3.2 Nutzung von KNIME

KNIME ist das zentrale Data Mining Tool der Projektgruppe. Mit diesem werden *Workflows* zur Datenanalyse, ETL Prozesse und Visualisierungen erstellt. Nachdem die Funktionalität erstellt und getestet wurde, werden die erstellten Workflows auf KNIME Server geladen, um große Datenmengen verarbeiten zu können.

Obwohl neben KNIME weitere Data Mining Tools existieren (vgl. RapidMiner), wurde KNIME aufgrund der Programmiersprache (Java), der Integration mehrerer Programmiersprachen und der Eclipse Basis gewählt. In der Universität Oldenburg wird für den Bereich Informatik hauptsächlich Java gelehrt, sodass alle Projektmitglieder im Stande sind in Java zu programmieren. Da KNIME durch Java Plugins erweitert werden kann, können alle Mitglieder Erweiterungen für KNIME schreiben und warten. Ein weiterer Vorteil von KNIME ist die Integration mehrerer Programmiersprachen, unter anderem JavaScript, R und Python. Somit können alle Implementierungen der jeweiligen Programmiersprachen genutzt werden und machen KNIME als Data Mining Tool flexibel. Der letzte Grund für die Nutzung von KNIME ist die Eclipse Basis. Da die meisten Projektmitglieder Eclipse als Entwicklungsumgebung genutzt haben, können sie diese einfach weiterverwenden, indem KNIME als Plugin installiert wird. Zudem bietet Eclipse eine reichhaltige Anzahl an Erweiterungen.

5 Technische Infrastruktur

Dieses Kapitel beschreibt die genutzte Infrastruktur zur Datenaufzeichnung. Aufgeteilt wird diese in die Aufzeichnung der AIS- und der Radardaten.

5.1.1 Aufzeichnung der AIS Daten

Die AIS-Daten werden von Schiffen über einen AIS-*Tranceiver* gesendet. Diese Daten sind im NMEA0183-Format codiert und werden über Funk versendet. Die Nachrichten werden daraufhin von mehreren AIS-Stationen empfangen (Brunsbüttel, Cuxhaven und Wilhelmshaven). Nach dem Empfang werden die Nachrichten in ein ECore-S100-Modell („The IMO E-Navigation Concept and the IHO S-100 Data Standard“ o. J.) geschrieben, um die Daten durch das Internet an einen RabbitMQ-Server („Messaging that just works — RabbitMQ“ o. J.) zu senden. Auf diesem horcht wiederum ein Prozess, welcher die Daten in die Datenbank schreibt.

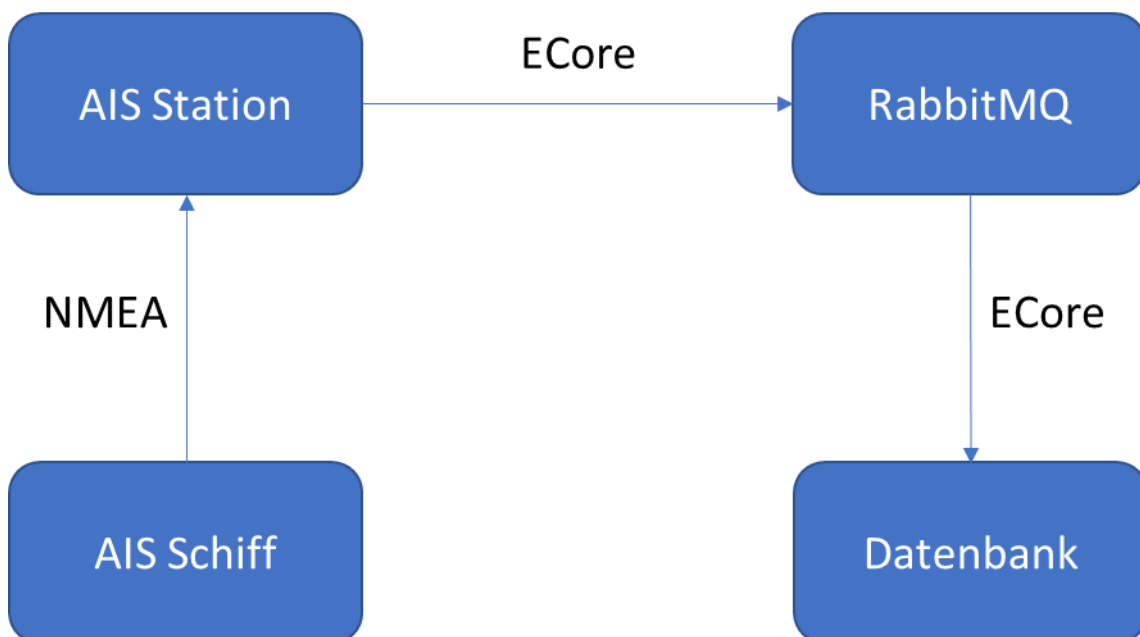


Abbildung 5-1: Aufzeichnung AIS Daten

Abbildung 5-1 verdeutlicht diesen Prozess und zeigt die Austauschformate zwischen den einzelnen Elementen.

5.1.2 Aufzeichnung der Radar Daten

Die Radardaten werden über mehrere Stationen entlang der Elbe aufgezeichnet (Brunsbüttel und Cuxhaven). Eine Radarstation nimmt die Daten als Video im Asterix-Format („ASTERIX“ 2012), welche nicht nur die gewünschten Objekte beinhalten, sondern alle Objekte in der Nähe. Aus diesem Grund müssen die Radardaten vorverarbeitet werden, umstehende Objekte und *Noise* herauszufiltern. Dies wird durch SPX Cambridge Pixel („SPx Tracking Server (Tracking, Plot Extraction, Radar Video Distribution, Recording)“ o. J.) erledigt, welches Schiffe erkennt und tracken kann. Somit können Position, Kurs und Geschwindigkeit extrahiert werden. Wie auch bei der AIS-Aufzeichnung werden die Daten daraufhin in einem ECore-Modell dargestellt und über den RabbitMQ-Server versendet. Schließlich werden diese in die Datenbank eingetragen.

SPX Verarbeitung

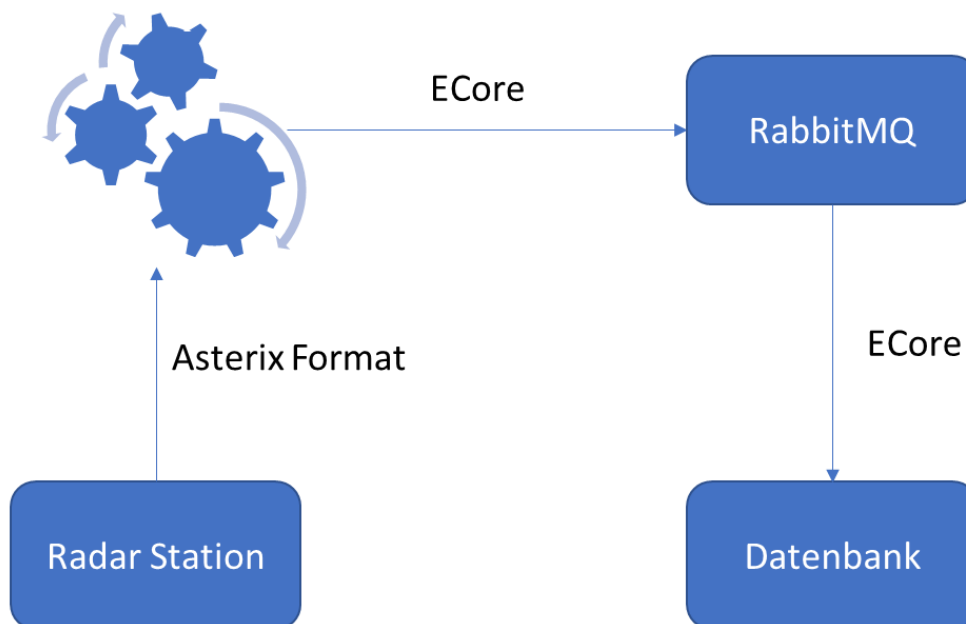


Abbildung 5-2: Aufzeichnung Radar Daten

Abbildung 5-2 verdeutlicht diesen Prozess mit den dazugehörigen Austauschformaten.

6 Aufteilung des Projektes

Das Projekt MaGellan besteht aus mehreren Bereichen. Zunächst müssen die benötigten Daten gesammelt und zugänglich gemacht werden, dies wird im Bereich „Data Collection“ übernommen. Bei den Daten handelt es sich um die AIS, Radar und Wetterdaten.

Die Daten liegen in ihrer Rohform vor und werden im Bereich „Data Processing“ in die benötigte Form gebracht. Für dieses Projekt bedeutet dies die Assoziation und Fusion der AIS- und Radar-Daten sowie die Kennzeichnung von fehlerhaften und bzw. oder nicht vorhandenen Daten.

Das Projekt schließt mit der Erkennung von kritischen Begegnungen zwischen zwei Schiffen sowie die Erkennung von Parametern, die das Risiko für eine Schiffsbegegnung erhöhen.

Die Gruppen wurden gleichzeitig gestartet, obwohl sie in Abhängigkeit stehen. So sollten die Daten gleichzeitig gesammelt und zur Verfügung gestellt werden sowie Wissen für die späteren Probleme gesammelt und Konzepte entworfen werden.

7 Umsetzung

In diesem Kapitel werden die Ergebnisse der Projektgruppe MaGellan vorgestellt.

7.1 Wetterdaten

Zur Analyse der AIS und Radar Daten wurden zusätzlich Wetterdaten genutzt. Hierbei wird zwischen maritimen und allgemeinen Wetterdaten unterschieden. Die maritimen Daten wurden vom BSH (Bundesamt für Seeschifffahrt und Hydrographie) und die allgemeinen vom DWD (Deutscher Wetterdienst) zur Verfügung gestellt.

Art	Beschreibung	Gemessen in
Wasserstand	Die Höhe des Wasserstandes abhängig von einem definierten Referenzsystem.	Meter
Strömung	Die Strömung in einem Raster mit einer zusätzlichen Strömungsrichtung.	Meter pro Sekunde
Windgeschwindigkeit	Die Windgeschwindigkeit in einem Raster mit einer zusätzlichen Windrichtung.	Meter pro Sekunde und Grad (0-360)
Wellenhöhe	Die Höhe der Wellen nach einem definierten Referenzsystem.	Meter
Salzgehalt/potentielle Wassertemperatur	Der Salzgehalt des Wassers und die daraus resultierende potentielle Wassertemperatur.	??, Grad Celsius

Tabelle 7-1: Maritime Wetterdaten

Art	Beschreibung	Gemessen in
Lufttemperatur	Die Lufttemperatur in einem bestimmten Raster.	Grad Celsius

Luftdruck	Der Luftdruck in einem bestimmten Raster.	
Bodentemperatur	Die Bodentemperatur im Wasser.	Grad Celsius
Schneehöhe	Die bemessene Schneehöhe in einem Gebiet.	Meter
Solar	Messung der Solarstrahlung. Dauer und Strahlungsmessung.	
Niederschlag	Gemessene Wassermenge in einem Gebiet.	
Frost	Die Bodentemperatur und der gefrorene Anteil.	
Bodenfeuchte	Der Wassergehalt des Bodens.	
Evapotranspiration und Wasserdampfgehalt	Die Summe der Verdunstung von Wasser im Gebiet.	
Bedeckungsgrad	Der Grad der Bewölkung.	

Tabelle 7-2: allgemeine Wetterdaten

Tabelle 7-1 und Tabelle 7-2 zeigen Wetterdaten, die gemessen werden und dementsprechend für Analyseverfahren genutzt werden können. Im Zuge des Projektverlaufs wurden allerdings nicht alle Wetterdaten in die Datenbank eingetragen oder in den Analyseverfahren berücksichtigt.

Name	Beschreibung
Wind Richtung	Die Richtung, in welche der Wind weht.
Wind Geschwindigkeit	Die Geschwindigkeit des Windes.
Wellenhöhe	Durchschnitt des höchsten Drittels der Wellen. (US Department of Commerce o. J.)

Niederschlag	Gemessene Niederschlagsmenge pro Quadratmeter.
Sichtweite	Sichtweite in Metern an einer Messstation.

Tabelle 7-3: Eingetragene Wetterdaten

Tabelle 7-3 zeigt alle Wetterdaten, welche in die Datenbank eingetragen wurden. Bis auf Niederschlag wurden alle Wetterdaten in den Analyseverfahren genutzt. Niederschlag sollte ursprünglich als Faktor für eine Berechnung der Sichtweite genutzt werden, allerdings hat sich herausgestellt, dass es separate Daten zur Sichtweite gibt, welche stattdessen genutzt wurden.

Einige maritime Wetterdaten wurden weder in den Analyseverfahren berücksichtigt noch in die Datenbank eingetragen. Das Problem ist, dass diese Wettertypen sehr viele Daten beinhalten und viel Zeit benötigen, um vorverarbeitet zu werden (siehe Kapitel Wetterdaten Verarbeitung). Der Zeitaufwand war zu groß, um 2 Jahre Daten in die Datenbank einzutragen. Aus diesem Grund wurde ein *On-Demand*-Eintrage-Service erstellt. Durch die KNIME-Rest-API kann der Zeitraum und der Wettertyp angegeben werden, welcher eingetragen werden soll. Daraufhin wird das erstellte Programm zum Eintragen der Wetterdaten gestartet und die gewünschten Wetterdaten werden nachgetragen (Kapitel 7.1.3).

Zudem wurden viele allgemeine Wetterdaten vom DWD nicht von der Projektgruppe berücksichtigt, da diese keinen direkten Nutzen für die Analyseverfahren beinhalten. Demensprechend hat sich die Projektgruppe vermehrt auf die maritimen Daten konzentriert. Allerdings stellt der DWD die Daten zur Verfügung, sodass diese im Datenmodell und in den Analyseverfahren erweitert werden können.

7.1.1 Datenbank Schema

Das folgende Bild stellt das aktuelle Datenbank Schema für die Wetterdaten bereit.

Abbildung 7-2 zeigt den Zusammenhang zwischen den Tabellen-Typen, welche im Folgenden näher beschrieben werden.

7.1.1.1 Meta-Daten

Die *File*-Tabelle speichert Meta-Daten über die Ursprungsdatei. Da die Daten in diesem Schema aus Dateien gewonnen werden, werden daher der *Name*, *Zuletzt aktualisiert* und der *Ursprungspfad* gespeichert. Der KNIME-Update-Workflow nutzt diese Daten, um den Aktualisierungsbedarf zu ermitteln, indem *Zuletzt aktualisiert* mit den neuen Dateien verglichen wird.

Zudem gilt diese Tabelle als "unsichtbar" für die Nutzer der Datenbank, da sie für die Abfragen auf Wetterdaten nicht genutzt werden müssen. Es werden lediglich Informationen über die Ursprungsdateien gespeichert, um zu überprüfen welche Dateien bereits eingetragen wurden und welche Daten aus welcher Datei stammen.

7.1.1.2 Allgemeine Tabellen

Die folgenden Tabellen (Tabelle 7-4) beinhalten allgemeine Informationen über die Wetterdaten. Daher der Typ, das verfügbare Zeitfenster, die Einheit und den Bereich der Wetterdaten.

Tabelle	Beschreibung
weather	Die <i>Weather</i> -Tabelle beinhaltet alle verfügbaren Wettertypen. Diese werden durch den Namen und die Beschreibung angegeben. Diese Tabelle ist der Einstiegspunkt für die Wetterdaten, da hierdurch direkt nach bestimmten Typen gefiltert werden kann. (Beispiel: <i>wind_speed</i>)

area	<p><i>Area</i> beinhaltet Informationen über den verfügbaren Bereich für bestimmte Wetterdaten Typen. Diese Tabelle speichert somit eine Geometrie, welche für einen Wettertyp den verfügbaren Bereich definiert. Diese Tabelle kann genutzt werden, um zu prüfen, ob Daten für einen bestimmten Bereich verfügbar sind.</p>
time_window	<p>Das Zeitfenster speichert den Zeitraum der Wetterdaten. Daher kann überprüft werden, in welchem Zeitraum Daten für einen bestimmten Wetter-Typ vorhanden sind. Zudem gibt die Spalte <i>granularity</i> an, wie weit die Messzeitpunkte im Durchschnitt auseinander liegen (Beispiel: Messungen alle 15 Minuten).</p>
measured_unit	<p>Diese Tabelle speichert die Messeinheit der Wetterdaten. Zudem gibt die Spalte <i>original_unit</i> an, welche Ursprungseinheit die Daten hatten, falls Konvertierungen im Zuge der Verarbeitung der Daten gemacht wurden. Somit können die Einheiten standardisiert werden, falls unterschiedliche Quellen und Einheiten genutzt wurden.</p>
weather_station	<p><i>weather_station</i> beinhaltet Informationen über die Wetterstation(„Stationsliste der 78 Messstationen (nach Stationsname sortiert)“ 2019). In Verbindung mit den providerspezifischen Tabellen kann hierüber die Position der Station und der Daten in Erfahrung gebracht werden.</p>

Tabelle 7-4: Allgemeine Wetter Tabellen

7.1.1.3 Hilfstabellen

Die folgenden Tabellen (Tabelle 7-5) sind Hilfstabellen und speichern in der Regel keine Werte. Da PostgreSQL-Datenbanken das Prinzip der Vererbung

nutzen können, wurden einige Tabellen als Interfaces erstellt. Somit können neue Wetterdaten als Erweiterungen eingefügt werden, indem die neuen Tabellen von den Interfaces erben. Zudem stellt die Nutzung der Hilfstabellen sicher, dass alle Tabellen eine einheitliche Benamung haben, um Abfragen zu vereinfachen.

Tabelle	Beschreibung
data	<p><i>Data</i> definiert die grundlegendsten Spalten für alle Wetterdaten. Daher definiert diese Tabelle nur die ID (<i>auto_increment</i>), Wetter ID (<i>foreign_key weather</i>) und einen Zeitstempel für den Messzeitpunkt.</p>
table_data	<p><i>Table Data</i> wird genutzt für die Erstellung von Tabellen, mit einzelner Messung an einer bestimmten Position. Daher werden keine Bereiche für Messungen genutzt, sondern einzelne Punkte mit einem bestimmten Wert.</p> <p>Daher definiert <i>table_data</i> nur eine zusätzliche Spalte <i>stationid</i>, welche die Referenz zu <i>weather_station</i> bildet.</p>
raster_data	<p><i>Raster Data</i> beinhaltet Bereiche von Daten. Im Unterschied zu <i>table_data</i> liegen diese Daten in der Form Wert (<i>number</i>), Bereich(<i>geometry</i>) vor. Im Vergleich zur Tabelle <i>data</i> wurde ein Zeitstempel für den Gültigkeitsbereich und Geometrie Spalten eingefügt. <i>Geometrie</i> definiert hierbei den Bereich, in welchem der Wert (<i>value</i>) gilt und <i>holes</i> definiert Bereiche in der Geometrie, in welchem der Wert nicht gilt.</p>

Tabelle 7-5: Hilfstabellen

7.1.1.4 Providerspezifische Tabellen

Neben den bereits vorgestellten Tabellen (Tabelle 7-6) sind die folgenden Tabellen spezifisch für die Provider der Daten erstellt. Somit werden je nach Wassertyp unterschiedliche Tabellen erstellt, welche speziell für die Speicherung des Typs genutzt werden. Zudem ermöglicht die Nutzung dieser Tabellen eine Art von Partitionierung zur Verbesserung der Performance. Da Abfragen auch auf die Hilfstabellen gemacht werden können, sind diese Tabellen für die Nutzer meistens unsichtbar.

Tabelle	Beschreibung
visibility	Diese Tabelle speichert die Sichtweite an einer Messstation gemessen in Meter.
precipitation	Diese Tabelle speichert die gemessene Niederschlagshöhe sowie die Art des Niederschlages an der jeweiligen Station. Angegeben wird die Höhe des Niederschlags in Millimeter. Die Art wird in WR-Codierung festgehalten.
air_temperature	Speichert die Lufttemperatur auf Zwei-Meter-Höhe der jeweiligen Messstation. Die Angabe wird in Celsius gespeichert.
cloudiness	<i>Cloudiness</i> speichert den Bedeckungsgrad durch Wolken. Ist der Himmel nicht erkennbar, wird eine -1 gespeichert. Ansonsten wird der Bedeckungsgrad zwischen 1 und 8 angegeben.
humidity	In dieser Tabelle wird die Luftfeuchte gespeichert. Die Luftfeuchtigkeit wird in Prozent angegeben.
wind	In <i>wind</i> werden Daten des Windes gespeichert. Dabei werden die Wind Geschwindigkeit und die -richtung angegeben. Die mittlere Windgeschwindigkeit wird in m/s und die Richtung in

	Grad gespeichert.
current	<i>Current</i> speichert Werte über die Stärke der Strömung und dessen Richtung.
wind_waves	<i>wind_waves</i> speichert sowohl Werte über Winddaten als auch Wellenhöhe. Somit bestehen die Winddaten aus einer Geschwindigkeit (m/s) und der Richtung (Grad). Die Wellenhöhe wird in Meter gemessen und beschreibt den Durchschnitt des höchsten Drittels der Wellen.
water_level	Diese Tabelle speichert Werte über den Wasserstand, gemessen in Metern.

Tabelle 7-6: Providerspezifische Tabellen

7.1.2 Wetterdaten Verarbeitung

Damit die Wetterdaten effizient verarbeitet und gespeichert werden können, werden diese in der Datenbank gespeichert. Ursprünglich liegen die Wetterdaten in verschiedenen Dateiformaten vor, sodass diese vorverarbeitet und eingetragen werden müssen. Viele Dateien beinhalten dabei mehrdimensionale Arrays, welche umgewandelt werden müssen, damit diese in der Datenbank gespeichert werden können. Hierfür wurde ein Konzept entwickelt, das im Folgenden vorgestellt wird.

Anstelle der Speicherung von rohen mehrdimensionalen Arrays, werden diese in Polygone umgewandelt. Ein Polygon beschreibt den Bereich, in welchem ein Wert und der Zeitstempel gültig ist. Für die Erstellung der Geometrien wurde ein Algorithmus entwickelt. Dieser geht Zeile für Zeile und Spalten von links nach rechts durch und überprüft, ob der Wert links oder der darüber den gleichen Wert haben. Gleiche Werte werden in einem Polygon zusammengefasst. Falls kein gleicher Wert gefunden wurde, wird ein neues Polygon erzeugt. Danach wurde ein Concave-Hull-Algorithmus (Schl [2016] 2019) verwendet, welcher basierend auf den Abständen zwischen Punkten die Hülle erzeugt. Die Concave-Hull gibt nur

die äußere Hülle einer Liste von Punkten zurück und löscht alle, welche im Inneren des Polygons liegen. Somit wird gewährleistet, dass der Speicheraufwand geringgehalten wird.

7.1.3 Update Workflow

Da vom DWD und BSH regelmäßig neue Wetterdaten veröffentlicht werden, müssen die Daten in der Datenbank aktualisiert werden. Vor allem ist dies wichtig, da die Extraktion von Szenarien regelmäßig durchgeführt wird und daher die Wetterdaten zur Verfügung stehen müssen.

Um die Wetterdaten zu aktualisieren, wurden daher zwei Workflows erstellt. Der erste Workflow dient zum Aktualisieren der DWD Daten. Neue Daten werden auf einen FTP-Server („CDC (Climate Data Center)“ o. J.) hochgeladen und aktualisiert. Der Workflow durchsucht, basierend auf einer vordefinierten Liste von Ordnern, den Server nach Dateien und vergleicht diese mit den bereits eingetragenen Dateien in der Datenbank. Falls ein Aktualisierungsbedarf festgestellt wurde, werden die betreffenden Dateien heruntergeladen und daraufhin eingetragen.

Der zweite Workflow zum Aktualisieren der BSH Daten wurde als On-Demand Service erstellt. Wie bereits beschrieben bedeutet die Verarbeitung dieser Dateien erheblichen Zeit- und Rechenaufwand, sodass diese nur bei Bedarf eingetragen werden. Der Service nimmt eine Anfrage über die KNIME-Rest-API an.

```
{
  "json-input-1": {
    "value": [
      {
        "Type": "tempsalt",
        "Time": "20160101"
      },
      {
        "Type": "current",
        "Time": "20180101"
      },
      {
        "Type": "current",
        "Time": "20170101"
      }
    ]
  }
}
```

Abbildung 7-3: Wetter Rest-Service Anfrage

Abbildung 7-3 stellt eine beispielhafte Rest-Anfrage dar. Übergeben werden der Zeitraum und Wettertyp. Der Workflow durchsucht daraufhin einem lokalen Ordner nach Dateien und trägt diese in die Datenbank ein.

7.1.4 Abfrage der Wetterdaten

Diese Kapitel beschreibt einige typische SQL-Abfragen auf die Wetterdaten.

```
SELECT *
FROM wind_waves
WHERE date(time) between '2018-03-10' and '2018-03-11'
      and date(to_time) between '2018-03-10' and '2018-03-11'
```

Abbildung 7-4: Wetterdaten Beispielabfrage Zeiten

Abbildung 7-4 zeigt eine Abfrage von Wetterdaten zwischen gegebenen Zeitstempeln. *Time* ist der Startzeitpunkt und *to_time* der Endzeitpunkt für die Wetterdaten. Die Konvertierung (*date()*) wurde gemacht, da in diesem Fall nur der Tag wichtig ist. Alternativ können auch Abfragen mit Zeitzonen und Zeit (Sekunden, Minuten, Stunden) erstellt werden.

Da diese Abfrage sehr üblich ist, wurde ein Index auf die Zeitstempel gesetzt. Dementsprechend können Ergebnisse schnell geliefert werden.

```
SELECT *, ST_AsText(current.geometry)
FROM weather, current
WHERE current.geometry
      && -- intersects, gets more rows -- CHOOSE ONLY THE
      ST_MakeEnvelope (
        xmin, ymin, -- bounding
        xmax, ymax, -- box limits
        my_srid)
OR current.geometry
  @ -- contained by, gets fewer rows -- ONE YOU NEED!
  ST_MakeEnvelope (
    xmin, ymin, -- bounding
    xmax, ymax, -- box limits
    my_srid)
```

Abbildung 7-5: Wetterdaten Beispielabfrage Bereiche

Die zweite übliche Abfrage ist eine Abfrage nach Bereichen (Abbildung 7-5). Durch die Nutzung von PostgreSQL kann *ST_MakeEnvelope* genutzt werden, um ein Rechteck zu erstellen. Die Datenbank wird daraufhin prüfen, ob die Wetter-Geometrien in dem definierten Bereich liegen oder ob diese den Bereich schneiden.

7.2 Datenmodell

Das Ziel im abschließenden Datenmodell war es Szenarien abzuspeichern. Hierfür musste ein neues Datenmodell entwickelt werden. Bei der Erstellung wurde darauf geachtet, dass das Datenmodell erweiterbar ist und Abfragen performant durchgeführt werden können. Um dies zu ermöglichen wurde mit Indizes und der Vererbung von PostgreSQL gearbeitet. Diese Indizes wurden auf Attribute gesetzt, auf die häufig Suchen stattfinden. Dabei handelt es sich meistens um den Zeitstempel. Bildhaft wird das Datenmodell einmal mit einem schlanken EER Modell (Siehe Abbildung 7-6) beschrieben. Außerdem kann das Datenmodell über Enterprise Architect eingesehen und erweitert werden. Es befindet sich im Anhang 10.1.

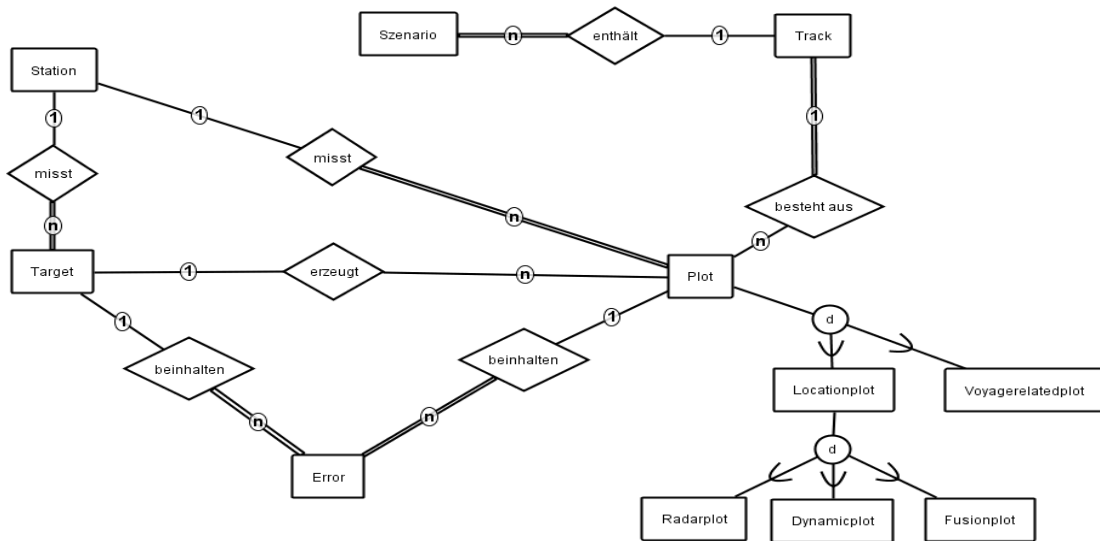


Abbildung 7-6: EER Modell

Damit das Datenmodell etwas klarer wird, wird es im Folgenden näher beschrieben. Hierfür werden die vier wichtigsten Entitäten Szenario, *Plot*, *Measurement* und *Target* näher beleuchtet.

7.2.1 Szenario

Ein Szenario besteht stets aus zwei *Tracks*. Ein *Track* ist als eine Abfolge von aufeinanderfolgenden Positionsdatenpunkten definiert. Zudem werden mehrere Szenarien beschreibende Daten abgespeichert. Zu diesen gehören die Art der Begegnungssituation, der geringste Abstand der beiden Schiffe in Metern, sowie statistische Daten (Wetterdaten, Tageszeit, Jahreszeit, Wochentag, Gebiet). Neben diesen statistischen Daten wird in einer Tabelle abgespeichert, inwieweit sich die Szenario-Beteiligten an die Regeln halten. Für das abspeichern der *Tracks* wird eine Tabelle mit Startzeit, Endzeit, Fahrendes Schiff und *Tracktyp* benötigt. Wie sich der *Track* zusammensetzt, wird in der Verbindungstabelle *Track-Position* zwischen einem *Track* und einem *Plot* abgespeichert.

7.2.2 Plot

Ein *Plot* ist ein Datenpunkt, der beispielhaft einen dynamischen AIS-Datensatz, bestehend aus Position, Kurs, Geschwindigkeit, *heading* und *rate of turn*,

abspeichert. Ein *Plot* kann sehr vielseitig sein und je nach Quelle unterschiedliche Attribute aufweisen. Deswegen hat die Projektgruppe mit der Vererbung von PostgreSQL gearbeitet. Diese ermöglicht es Daten nach ihrer Art zu unterteilen. Die Vartabelle *Plot* besteht somit lediglich aus einer ID, einem *Timestamp* und Daten zum Schiff und der Messstation, welche die Daten gemessen hat. Abgeleitet vom *Plot* sind dann die Tabellen *Locationplot* und *Voyagerelatedplot*. Der *Locationplot* speichert die Daten, welche sowohl bei Radar- als auch bei AIS-Daten vorkommen (Position, Geschwindigkeit, Kurs). *Locationplot* ist somit gleichzeitig das Kind von *Plot* und der Vater von *Radarplot*, *Dynamicplot* und *Fusionplot*. Der *Radarplot* weißt dieselben Daten wie der *Locationplot* auf und ist lediglich dazu da, um Radardaten separat abfragen zu können. Im *Dynamicplot* werden außerdem die Datenausrichtung, *rate of turn* und der Navigationsstatus abgespeichert. Der *Fusionplot* speichert Position, Geschwindigkeit, Kurs sowie die beiden *Quellplots* (Ein *Radar-* und ein *Dynamicplot*). Durch diese Architektur besteht die Möglichkeit, ohne komplexe SQL Statements, mit allen *Plots*, allen *Plots* mit Positionsdaten oder auch nur Radar oder AIS-*Plots* zu arbeiten. Im *Voyagerelatedplot* werden die reisebezogenen Daten (geplante Ankunftszeit, Tiefgang und Reiseziel) abgespeichert. Da die Projektgruppe mit Rohdaten gearbeitet hat und diese verarbeitet hat, besteht außerdem die Möglichkeit über die Tabelle *plot_data_quality* die Datenqualität jedes einzelnen *Plots* einzusehen. Dieser setzt sich aus Attributen wie Vollständigkeit oder Korrektheit zusammen.

7.2.3 Measurement

Jeder *Plot* hat eine Messstation als Quelle. Diese wurden ebenfalls im Datenmodell abgespeichert. Hierfür wurde das gesamte LabskausDB Schema übernommen. Für die Projektgruppe relevant waren hierbei die Messgeräte in Wilhelmshaven, Cuxhaven und Brunsbüttel. Jede Station besteht aus einer achtstelligen Reference-ID. Diese ist in vier Blöcke von je zwei Ziffern aufgeteilt.

- Der erste Block zeigt auf um welche Station es sich handelt (z.B. Brunsbüttel, Cuxhaven).
- Der zweite Block zeigt auf den Sensor (z.B. Radar, AIS)

- Der dritte Block zeigt auf den *Sensorport* (ObjectPort, VideoPort)
- Der vierte Block zeigt auf die *Portinstance* (dynamisch, statisch)

Datenbanktechnisch wird in der allgemeinen Tabelle „*Measurement*“ die achtstellige ID gespeichert, sowie vier ID’S zu den Tabellen Station, Sensor, *Sensorport* und *Portinstance* in denen wiederum die Bezeichnungen abgespeichert sind.

7.2.4 Target

Das *Target* ist das Objekt, welches die Daten erzeugt. Das *Target* bei AIS-Daten ist meistens das Schiff. Es gibt jedoch auch spezielle *Targets*, wie beispielsweise Helikopter, welche ebenfalls Daten senden. Damit zwischen normalen Schiffen und speziellen Sendern unterschieden werden kann, wurde ebenfalls mit der Vererbung von PostgreSQL gearbeitet. Dadurch können die beiden unterschiedlichen AIS-*Targets* in verschiedenen Tabellen (*ais_vessel* & *ais_special_mmsi*) abgespeichert werden. In der AIS Vessel Tabelle werden sämtliche Daten gespeichert, bei der die MMSI mit einer Ziffer zwischen 2 und 7 beginnt. Es ist jedoch möglich über die Tabelle *ais_target* sämtliche Daten abzurufen. Auf Radarebene werden den einzelnen Radartargets Radar-ID’s zugeordnet. Diese werden in der Tabelle *radartarget* abgespeichert.

7.3 ETL

Der Inhalt dieses Kapitels ist es, die wichtigsten Aspekte des Prozesses, welcher für die Übertragung der Daten von der Rohdatenbank *nbdatabase* zur Zieldatenbank MaGellan zuständig ist, vorzustellen. Zudem wird erläutert an welchen Stellen der Prozess sinnvoll erweitert werden kann.

Nach Öffnen des Prozesses können acht Fenster eingesehen werden, in welche der Prozess aufgeteilt ist:

- Extract Raw Data and set ID
- Divide the Data (AIS, Radar)
- Process Static AIS Data
- Process Dynamic AIS Data

- Process Voyage Data
- Process Radar Data
- Process Merged Data
- Save all results in database

Im Folgenden wird jeder dieser Unterpunkte vorgestellt.

7.3.1 Extract Raw Data and set ID

Die Aufgabe dieses Bereiches ist es, dafür zu sorgen, dass die gewünschte Datenmenge an das Zieldatenmodell übertragen wird. Hierfür muss zum einen festgelegt werden, wie viele Daten bei jedem Durchlauf verarbeitet werden soll und zum anderen muss bestimmt werden, bis zu welcher ID die Daten bereits verarbeitet worden sind. Um dies herauszufinden, wird in der Tabelle Cache die zuletzt verarbeitete ID abgespeichert. Da die Tabelle für mehrere temporäre Daten verwendet wird, muss bei der Abfrage eine Einschränkung auf die *owner_id = 3* gewählt werden. Dieser Wert wird in KNIME als *Flow Variable* „max_dynamic“ abgespeichert. Daraufhin werden die Rohdaten aus der Tabelle *tracks* der Datenbank *nbdatabase* extrahiert. Abschließend wird überprüft, ob die Ausführung des Workflows nach der gerade ausgeführten Iteration beendet werden soll, indem nachgeschaut wird, ob alle Datensätze aus der Rohdatenbank verarbeitet wurden.

7.3.2 Divide the data (AIS, Radar)

Der Input dieses Bereiches sind die Rohdaten, welche über verschiedene Prüfungen in Radardaten, statische-, dynamische- und reisebezogene-AIS-Daten unterteilt werden.

Die erste Überprüfung, die dabei stattfindet, ist auf die Variable „Theme“ welche abprüft, ob es sich um AIS oder Radardaten handelt. Bei den AIS-Daten findet danach eine Überprüfung auf die *Reference-ID* statt, durch die herausgefunden werden kann, ob es sich um dynamische- oder statische-/reisebezogene-Daten handelt. Dies ist möglich da dynamische AIS-Daten mit der ID 00 und statische/reisebezogene Daten mit der ID 01 enden. Bei dieser Überprüfung

landen Datensätze ohne *Reference-ID* zuerst bei den dynamischen Daten. Dort findet nochmals eine zweite Überprüfung auf die Variable *callsign* statt. Alle Datensätze mit einem Rufzeichen, werden mit den statischen Daten konkateniert. Für die Unterteilung zwischen statischen und reisebezogenen Daten werden mit einem JavaSnippet die Spalten *MMSI*, *Timestamp*, *ReferenceID* und *ID* dupliziert. Dies wurde so umgesetzt, da mit einem einfachen „Column Splitter“ die reisebezogenen Daten aus den Datensätzen rausgeschnitten werden können und sowohl die statischen als auch die reisebezogenen Daten diese Variablen aufweisen.

Der Bereich liefert vier verschiedene Outputs: unverarbeitete Radar-Daten, unverarbeitete statische AIS Daten, unverarbeitete dynamische AIS-Daten und unverarbeitete reisebezogene AIS-Daten.

7.3.2.1 Process Radar Data

Der erste Knoten filtert überflüssige Spalten aus dem Input der unverarbeiteten Radardaten. Da innerhalb der Rohdatenbank jeder Datensatz sämtliche Attribute besitzt, wurden hier Attribute wie die *MMSI*, das Rufzeichen oder der Schiffstyp entfernt. Am Ende besteht ein Radardatensatz lediglich aus einer *ID*, einem *Timestamp*, der Position, der Geschwindigkeit, dem Kurs, dem Target und der Messstation.

Danach werden die Spalten so umbenannt, dass es mit dem neuen Datenmodell konsistent ist. Innerhalb dieses Bereichs finden vier verschiedene Verarbeitungsschritte statt.

Im Metaknoten „parse positions“ wird die Position in das PostGIS Geometry-Format konvertiert.

Der Metaknoten „process radar plots“ wird lediglich dazu verwendet die *ReferenceID* in die *Measurement-ID* aus dem neuen Datenmodell umzuwandeln. Dieser Knoten kann noch erweitert werden, um beispielhaft Kurs und Geschwindigkeiten zu glätten.

Der Metaknoten „process radar targets“ ist dazu da, um alle aufgetretenen Radartargets für das wegspeichern vorzubereiten.

Der Metaknoten „radar errorcodes“ ist dazu da, auffällige Datensätze zu markieren. Als auffällig gelten Datensätze die einen Kurs von genau 90 oder 0 aufweisen oder eine Geschwindigkeit von 0 vorliegt. Hierfür wurde sich entschieden, da die anderen Datensätze meist aus 8 Nachkommastellen bestehen und keine glatten Zahlen vorkommen. Außerdem werden Daten markiert, welche keine *RefereceID* oder kein *target* vorweisen.

Die verarbeiteten Radardaten werden an den letzten Abschnitt „Save all results in database“ weitergegeben.

7.3.2.2 Process Static AIS Data

Die Rohdaten werden im ersten Schritt im Metaknoten „clean up data“ bereinigt.

Folgende Bereinigungs-schritte werden durchgeführt:

- Entfernen von Leerzeichen am Anfang und Ende von Strings
- Werte auf null setzen, welche keinen Eintrag bzw. ein String „null“ aufweisen
- Werte auf null setzen, wenn die Werte negativ (-1) sind – Default-Werte

Nach Bereinigung der Daten, werden im Metaknoten „determine new vessel and occurence“ Duplikate gesucht. Dafür werden die Daten, welche bereits in der Datenbank sind mit den neuen Daten zusammengeführt und gruppiert. Der Mechanismus funktioniert so, dass es zu einem Schiff mehrere Datensätze geben kann. Beispielhaft würde es zwei unterschiedliche Datensätze geben, wenn die Länge eines Schiffes unterschiedlich ist und der Rest identisch. Über einen Zähler kann ein Indiz über die Richtigkeit der einzelnen Datensätze gewonnen werden. Der Zähler spiegelt hierbei die Häufigkeit der einzelnen Datensätze dar. Der Zähler kann dies jedoch nicht zu 100% sicherstellen. Eine mögliche Erweiterung an dieser Stelle wäre es, über einen externen Dienstleister wie *vesselfinder* oder *marinetraffic* die Korrektheit der Datensätze zu überprüfen.

Im letzten Schritt werden die statischen Daten auf Auffälligkeiten überprüft. Als Auffälligkeiten identifiziert wurden:

- MMSI hat eine unübliche Länge
- Kein Rufzeichen

- Das Schiff ist zu lang oder zu breit
- Das Schiff hat einen nicht vorhandenen Schiffstyp
- IMO hat eine unübliche Länge
- Das Schiff hat keine Länge oder Breite
- Kein Name
- Keine Information über die Position des AIS-Senders

7.3.2.3 Process voyage Data

Wie bei den statischen Daten werden die reisebezogenen Daten im ersten Schritt bereinigt. Hierfür wird die Ankunftszeit durch mehrere *JavaSnippets* angepasst. Innerhalb der Rohdatenbank werden die Ankunftszeiten mit dem Jahr 1970 abgespeichert. Der Prozess sorgt dafür, dass das Jahr sich dem des *Timestamps* anpasst. Lediglich, wenn die Ankunftszeit im Januar ist und der erzeugte Datensatz aus November oder Dezember des Vorjahres kommt wird das kommende Jahr verwendet. Außerdem werden die Default-Werte der Reiseziele und des Tiefgangs auf null gesetzt.

Zur Sicherstellung, ob die Reiseziele korrekt sind, kann der *Destination Cleaner* (Kapitel 7.6) eingebunden werden. Dies wurde im endgültigen Prozess nicht vorgenommen, da es ein zu hohes Risiko war diesen kurz vor Projektende zu integrieren und der Ankunftsort für unser Projektergebnis nicht relevant war.

Daten, bei denen die Ankunftszeit vor dem Zeitstempel liegt, keine Ankunftszeit, kein Reiseziel oder kein Tiefgang vorliegt werden als auffällig markiert.

7.3.2.3.1 Process dynamic AIS Data

Analog zu den Radardaten werden die Positionsdaten ebenfalls in ein Geometry-Format umgewandelt. Außerdem werden die Geschwindigkeiten von Daten aus Wilhelmshaven mit einer ID kleiner als 1.586.530.520 umgewandelt, damit sämtliche Geschwindigkeitsdaten in m/s vorliegen, da die Geschwindigkeit bis zu diesem Punkt in Knoten gespeichert wird.

Folgende AIS Daten werden als auffällig markiert:

- Fehlende Messstation

- Fehlender Navigationsstatus
- Fehlende *rate of turn* Werte
- *Heading* ist über 360 oder nicht vorhanden
- Kurs ist unter 0 oder über 360
- Geschwindigkeit ist unter 0
- Positionswerte sind nicht plausibel

Im Metaknoten „create tracks“ werden sämtliche Schritte vollzogen, um *Tracks* in der Datenbank abzuspeichern. Hierfür werden im ersten Schritt Routen gebildet aus den eingehenden Daten. Positionsdaten werden zusammengeführt zu einer Route, wenn der Zeitabstand zwischen zwei Punkten kleiner 180 Sekunden ist. Dieser Abstand wurde gewählt, da der Sendeintervall von AIS-Daten von der Geschwindigkeit des Schiffes abhängt. Wenn ein Schiff steht, sendet es alle 180 Sekunden Daten.

Als *Track* weggespeichert werden lediglich Routen, die mehr als 10 *Track* Positionen aufweisen. Da der Prozess stets eine feste Anzahl an Datensätzen verarbeitet, kann es sein das ein *Track* abgeschnitten wird. Um dies zu verhindern wurde der Knoten „Route Identification Distributor“ geschrieben. Dieser sorgt dafür, dass alte *Tracks* mit neuen zusammengeführt werden. Die letzte *Track* Position von unvollendeten *Tracks* werden in der Datenbank, in der Tabelle *cache* abgespeichert.

7.3.2.3.2 Process Merged Data

In diesem Bereich findet die Datenfusion zwischen AIS und Radardaten statt. Im Knoten *DataFusion* kann der minimale *precision*-Wert eingestellt werden.

Außerdem kann im Knoten *TrackMerger* eingestellt werden, welchem Datensatz (AIS oder Radar) mehr vertraut wird. Hierfür gibt es die Variable *Tendency*, welche einen Wert zwischen 0 und 1 aufweist. Je näher die Zahl an der 1 ist, desto eher wird den Radar-Daten vertraut. Die neuen *Tracks* werden in die Tabelle *Fusionplot* gespeichert.

7.3.2.4 Save all results in database

Der Prozess endet mit dem Bereich des Abspeicherns. Hier werden sämtlich verarbeitete Daten in das neue Datenmodell gespeichert.

7.4 Datenassoziation und -fusion

Der Projektgruppe MaGellan wurden sowohl AIS- als auch Radardaten zur Verfügung gestellt. Dabei wurden die Radar Daten ausgewertet und aufbereitet, sodass in der Datenbank Positionen, Kurs und Geschwindigkeitsinformationen eingetragen wurden. Damit auch die Radardaten nutzbar gemacht werden konnten, wurden diese fusioniert. Dieser Prozess ist aufgeteilt nach Datenassoziation und Datenfusion.

7.4.1 Datenassoziation

Bei der Datenassoziation handelt es sich um ein Verfahren, welches einem AIS-*Track* einen oder mehrere Radar-*Tracks* zuordnet. Hierbei müssen zwei *Tracks* gefunden werden, welche einen ähnlichen Verlauf aufweisen.

Hierzu werden zeitdiskret die Distanzen zwischen den *Track*-Punkten ermittelt. Über diese Zeitreihe werden daraufhin statistische Kennzahlen ermittelt. Zuerst wird der Median ermittelt, welcher Aufschluss darüber gibt, ob die *Tracks* sich nah aneinander befinden. Geringe Distanzen erhöhen die Assoziations-Chance. Allerdings entsteht hierbei das Problem, dass parallel fahrende Schiffe assoziiert werden könnten. Daher können zwei verschiedene Schiffe nah einander fahren und eine hohe Chance haben assoziiert zu werden. Um dieses Problem zu lösen, wurde zudem die Standardabweichung der Distanzen ermittelt. Eine geringe Standardabweichung erhöht dementsprechend die Assoziationschance und verringert das Risiko parallel fahrende Schiffe als dieselben zu identifizieren. Nachdem die Kennzahlen ermittelt wurden, wird eine Chance ermittelt. Diese berechnet sich durch:

$$F(x) = \frac{D - \sigma - \mu}{D}$$

Durch diese Formel wird die Assoziationschance zweier *Tracks* berechnet. D steht hierbei für die maximale Distanz zwischen den *Trackpoints*, σ für die

Standardabweichung und μ für den Median. Dabei liegt der Wertebereich der Formel zwischen 0.0 bis 1.0. Eine Assoziationschance über 0.5 wird als hoch erachtet. Angenommen die Ausgabe der Formel hat einen Wert von 0.5, dann summieren sich die Standardabweichung und Median zu 150m. Dieser Wert entspricht somit der maximalen Abweichung zwischen einem AIS- und Radar-*Track* mit einer Assoziationschance von 0.5.

7.4.2 Datenfusion

Nachdem zu einem AIS-*Track* ein oder mehrere Radar-*Tracks* gefunden wurden, können diese nun fusioniert werden. Dieser Prozess verschiebt die *Track*-Punkte des AIS-*Tracks* näher zum Radar-*Track*, um einen fusionierten *Track* zu erzeugen. Normalerweise wird dem Radar-*Track* mehr vertraut als dem AIS-*Track*, allerdings würde eine vollständige Nutzung des Radar-*Tracks* zu einer Diskrepanz in den COG-Werten führen, falls der AIS-*Track* nicht vollständig durch den Radar *Track* repräsentiert wird. Um dieses Problem zu beheben, werden die Punkte des fusionierten *Tracks* durch lineare Interpolation („Lineare Interpolation“ o. J.) berechnet. Hierbei werden zeitdiskret die Punkte zwischen den *Tracks* ermittelt und die Distanz zwischen den Punkten berechnet. Daraufhin wird die Distanz mit einem Tendenzwert [0.0;1.0] multipliziert und durch lineare Interpolation der neue Punkt berechnet. Die Tendenz gibt an, welcher der beiden *Tracks* bevorzugt werden soll (0 – Bevorzugung von AIS, 1 – Bevorzugung von Radar). Da Radarstationen eine höhere Genauigkeit aufweisen als GPS, wurde ein Wert von 0.8 gewählt, welches einer starken Bevorzugung von Radar zur Folge hat.

Nach der Fusion werden die neuen *Track*-Punkte und der Ursprung der Daten (MMSI, RadarID) in einem *Fusionplot* in der Datenbank abgespeichert. Dementsprechend können durch die IDs die Ursprungsdaten jederzeit wieder genutzt werden.

7.4.3 Fusionsergebnisse

Dieser Abschnitt beschreibt die Ergebnisse der Datenfusion.

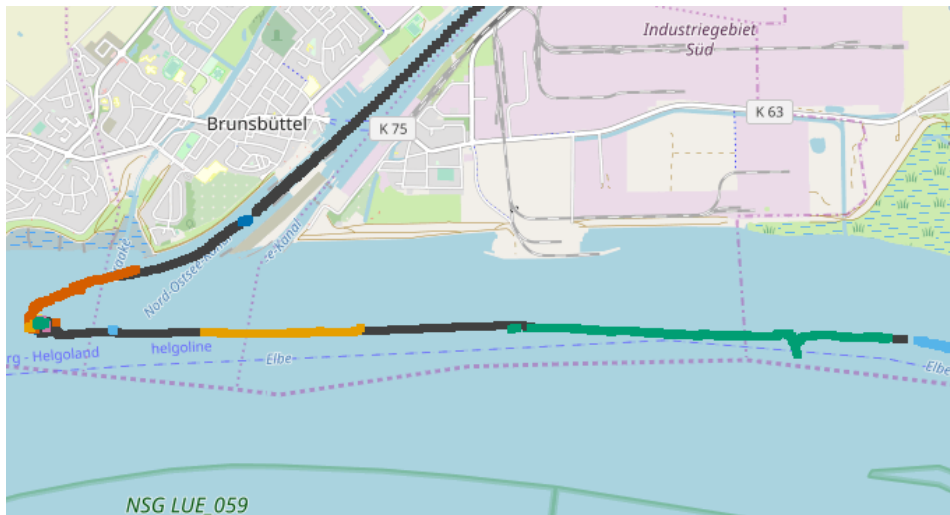


Abbildung 7-7: Beispiel Track Fusion

Abbildung 7-7 zeigt einen fusionierten AIS-Track, wobei die schwarzen Punkte Abschnitte markieren, zu denen keine Radar Tracks gefunden wurden und farbige Punkte markieren Abschnitte, welche fusioniert wurden. Der genutzte Confidence-Wert zur Track Assoziation beträgt 0.5 und der Tendenz Wert für die Daten Fusion 0.7. Dementsprechend wird der Radar-Track stark bevorzugt.

MMSI	confidence new	count new	radarid	confidence old	count radarid old
244690117	0.73	23		0.71	23
354552000	0.63	17		0.52	12
211772470	0.58	15		0.54	12
636014587	0.69	14		0.53	12
211181950	0.65	13		0.61	13
235108405	0.68	13		0.61	13
236265000	0.55	13		0.51	10
2447004	0.62	13		0.54	13
244870674	0.64	13		0.59	13
304752000	0.64	13		0.58	12
211721570	0.73	10		0.71	10

249206000	0.71	10	0.63	9
211181930	0.66	9	0.64	9
212702000	0.54	9	0.49	9
636017416	0.62	9	0.53	9
209777000	0.62	8	0.48	6
212518000	0.7	8	0.61	8
246397000	0.63	8	0.56	8

Tabelle 7-7: Vergleich Datenfusion

Im Verlauf der Projektgruppe wurden zu jedem Schiff zusätzlich die Position des GPS-Senders auf dem Schiff in die Datenbank eingetragen. Somit konnten genauere Distanz Berechnungen und bessere Assoziationsergebnisse erzielt werden. Tabelle 7-7 zeigt die Assoziationswahrscheinlichkeit (*confidence*) und Anzahl an fusionierten Radar *Tracks* (*count radarId*) mit und ohne Kenntnis von der Position des GPS Senders. Hierbei fällt auf, dass deutlich mehr Radar *Tracks* pro AIS *Track* (*MMSI*) gefunden wurden. Gleichzeitig steigt die durchschnittliche Assoziationswahrscheinlichkeit bei jedem *Track*.

7.5 Bereinigung der Daten

Aufgrund der Ungenauigkeit der erfassten Positionsdaten müssen diese durch Filter geglättet werden. Als Glättung wird in diesem Sinne die Ermittlung einer Kurve, die zwischen den gemessenen Punkten verläuft und damit eine möglichst genaue Annäherung an den ursprünglichen Kurs darstellt, verstanden. Insbesondere bei der Fusion der AIS- und Radardaten können die Filter zur Anwendung kommen, um einen Kurvenverlauf zwischen den gemessenen AIS-Positionen und den Radar-Positionen zu ermitteln. Im Wesentlichen wurden zwei verschiedene Arten von Filtern während der Projektgruppe getestet. Dabei handelt es sich um den Kalman-Filter, den Gaussian-Kernel-Regression-Filter. Umgesetzt wurden diese Filter als KNIME-Knoten, die im *git-Repository* zur Verfügung stehen.

Der Kalman-Filter schätzt die Position anhand von vorherigen Messungen und einem Systemmodell, welches das zu messende System simuliert (Beschorner 2012). Hierbei kann zum Beispiel ein Bewegungsmodell des Systems erstellt werden. Als Bewegungsmodell wurde das Constant Velocity Model verwendet. Der Filter eignet sich vor Allem für das *Live-Tracking*, indem es die aktuelle Position vorausschätzt und im Nachhinein weiter korrigiert (Beschorner 2012). Dadurch, dass im Rahmen der Projektgruppe allerdings kein *Live-Tracking* benötigt wird und die Daten erst nach dem Erfassen korrigiert werden sollen, waren die Ergebnisse des Filters für uns nicht geeignet. Darüber hinaus wurden auch Punkte, die nicht zu sehr abweichen zu stark verschoben, obwohl diese plausibel waren.

Als weiterer Filter wurde der Gaussian-Kernel-Regression-Filter umgesetzt. Dieser Filter basiert auf der Gaussian-Kernel-Regression, einer nichtlinearen Regression, die einen Kurvenverlauf zwischen den gemessenen Punkten erstellt. Dazu wird für jeden gemessenen Punkt der gewichtete Mittelwert aller umliegenden Punkte gebildet, wobei die Gewichtung abhängig von der Entfernung der Punkte ist (McCormick 2014). In unserem Fall stellt die Entfernung der Punkte nicht die tatsächliche Distanz zwischen den beiden Punkten dar, sondern die zeitliche Differenz zwischen den Messungen. Mit dem Filter werden wie in Schüssler et al. die Längen- und Breitengrade einzeln geglättet. In Formeln ausgedrückt sieht der Filter wie folgt aus (Schüssler und Axhausen 2008):

Für jede Koordinaten-Dimension $c \in x, y$ wird der geglättete Wert $\tilde{c}(t)$ zum Zeitpunkt t einzeln berechnet als:

$$\tilde{c}(t) = \frac{\sum_j (w(t_j) * c(t_j))}{\sum_j w(t_j)}$$

$c(t_j)$ ist der Rohwert der Koordinate c zum Zeitpunkt t_j und $w(t_j)$ ist die Gaußsche-Kernel-Funktion für jeden Zeitpunkt t_j berechnet aus:

$$w(t_j) = \exp\left(-\frac{(t - t_j)^2}{2\sigma^2}\right)$$

σ stellt die Kernel-Bandbreite dar, mit der angegeben wird, wie sehr die Punkte um den zu bereinigenden Punkt ins Gewicht fallen.

Die Ergebnisse sind in Abbildung 7-8 und Abbildung 7-9 zu sehen:

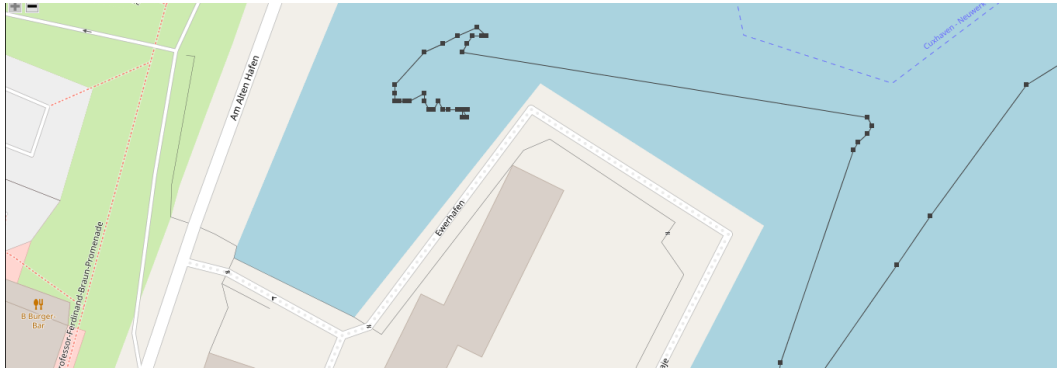


Abbildung 7-8: Unglätzte AIS-Daten



Abbildung 7-9: Durch Gaussian-Kernel-Filter geglättete AIS-Daten

Wie zu sehen ist, ist aus dem „Zick-Zack-Kurs“ aus Abbildung 7-8 ein wesentlich glatter verlaufender Kurs geworden. Allerdings lässt sich hier erkennen, dass die Punkte stark zusammengefasst wurden und der ursprüngliche Kurs, auch bei Anpassungen der Parameter, zu stark verändert wird.

Aufgrund der unbefriedigenden Ergebnisse wurde sich dazu entschlossen, auf eine Glättung der Daten zu verzichten.

7.6 Destination Aufbereitung

Wie in Kapitel 4.1 erwähnt, werden die AIS-Daten unterteilt in Angaben zu statischen, dynamischen und reisespezifischen Daten. Die reisespezifischen Daten

eines Schiffes enthalten neben den Attributen wie Tiefgang und geschätzter Ankunftszeit auch Angaben über den Zielhafen des Schiffes.

Für den Ablauf der Projektphase ist es nicht essenziell zu wissen, welchen Zielhafen das Schiff ansteuert. Allerdings bieten die Angaben zu Start- und Zielhafen eine weitere Möglichkeit festzuhalten bzw. zu erforschen, welche Route diesem Schiff in dem betrachteten Gebiet zuzuordnen ist. Angaben zum Zielhafen ermöglichen auch die Erforschung, an welcher Position das Schiff das Betrachtungsgebiet wieder verlässt bzw. es wieder befährt. Optimal wäre es, wenn sich auch für die Schiffe, bei denen kein Starthafen eingetragen wurde, der Starthafen ermitteln ließe. Die Möglichkeit einen Starthafen einzutragen ist bei allen gängigen AIS-Modulen gegeben, allerdings wird in vielen Fällen dieser Punkt bei der Dateneingabe nicht beachtet. Durch den *United Nations Code for Trade and Transport Locations* (UN/LOCODE) ist eine Angabe des Start- und Zielhafens verbindlich geregelt („UN/LOCODE Code List by Country - Trade - UNECE“ o. J.), jedoch wird nach Erkenntnis der Projektgruppe diese Vereinbarung nicht von allen eingehalten. Nach der ersten stichprobenartigen Sichtung der reisespezifischen Daten ist aufgefallen, dass häufig keine oder fehlerhafte Angaben zum Zielhafen gemacht werden. Dieses hat zum einen den Grund der abweichenden Schreibweise, wie z.B. beim Zielhafen „Bremerhaven“ für den Abkürzungen wie „BHV, DEBRV, Bremerhaven Germany“, usw. verwendet werden. Neben den nicht eindeutigen Abkürzungen sind viele Einträge mit Rechtschreibfehlern bzw. Buchstabendrehern vorhanden.

Damit aus diesen Daten ein sinnvoller und für das Projekt relevanter Datensatz wird, sollten die Angaben für die Zielhäfen nach Überprüfung und Überarbeitung der Eingaben wie folgt aussehen:

„Länderkürzel, Abkürzung des Starthafens → Länderkürzel, Abkürzung des Zielhafens“,

also z.B.

Deutschland Hamburg → Niederlande Rotterdam (DE HAM → NL RTM).

Da bei einigen Schiffen keine Angaben zum Starthafen gemacht wurden, gebe es die Möglichkeit, diesen Schiffen die letzte Position, an der sie sich länger

aufgehalten haben, als Starthafen zuzuordnen. Allerdings macht es nur Sinn, wenn an dieser Stelle auch ein Hafen ist und klar ausgeschlossen werden kann, dass das Schiff aus keinem anderen Grund an diesem Punkt gehalten hat. Innerhalb der Projektgruppe wurde sich darauf geeinigt, dass den Schiffen kein Starthafen zugeordnet wird. Gründe für diese Entscheidung sind zum einen, dass diese Angabe für das Projektergebnis nicht von Bedeutung ist. Zum anderen, dass es nur eine Annahme wäre, welche auf den letzten Haltepunkt des Schiffes basiert. Die Hafensliste wird in der Datenbank „Magellan“ unter der Tabelle „harbour“ abgespeichert und kann jederzeit bearbeitet werden. Die Tabelle setzt sich aus den Spalten „id, name, country, unctad, alias und position“ zusammen. Die offizielle Tabelle „harbour“ wurde der Projektgruppe vom OFFIS zur Verfügung gestellt, ergänzt wurde diese Tabelle um offizielle Binnenhäfen aus Deutschland und teils aus den Niederlanden. Zum jetzigen Zeitpunkt sind über 1.800 Häfen in der Datenbank erfasst. Betrachtet werden sowohl kleinere Binnenhäfen wie z.B. Oldenburg bis hin zu sehr großen Häfen wie Rotterdam oder Hamburg.

Um die Angaben der Zielhäfen aufbereiten und ggf. ergänzen zu können, werden die Daten mit dem Levenshtein-Algorithmus auf Tippfehler bzw. Rechtschreibfehler überprüft. Basierend auf der Editierdistanz, liefert der Levenshtein-Algorithmus zu je zwei Zeichenketten die zugehörige Distanz (Geduldig 2014, 5).

„Der Levenshtein Algorithmus basiert auf dem Prinzip der dynamischen Programmierung und liefert bei Bedarf nicht nur die Editierdistanz, sondern auch alle möglichen Editierskripte zweier Strings.“
(Geduldig 2014)

In dem KNIME-Workflow, den die Projektgruppe durch den Projektbetreuer zur Verfügung gestellt bekommen hat werden die Zielangaben überprüft und bearbeitet bzw. bereinigt. Der verwendete Workflow ist Teil einer Bachelorarbeit, die im OFFIS für die Abteilung „Systemanalyse und -optimierung“ entwickelt

wurde (Tsiroglou 2018). In dem Workflow, wird unter anderem anhand des Levenshtein-Algorithmus nach fehlerhaft eingetragenen Start- und Zielhäfen gesucht. Zudem werden die vorhandenen Start- und Zielhäfen in die oben erwähnt Form gebracht.

Berechnet werden mit dem Levenshtein-Algorithmus die minimalen Anzahlen an Änderungen, die nötig sind, um aus der ersten Zeichenkette eine zweite Zeichenkette zu generieren. Bei den Zielhäfen wird im Anschluss an die Überprüfung der ersten Zeichenkette, die neu generierte zweite Zeichenkette als ein möglicher Hafename dargestellt. Anhand einer offiziellen Hafenliste wird überprüft, ob der neu generierte Name in der Liste zu finden ist. Schlussendlich steht der Projektgruppe, durch den Destination Cleaner eine aufbereitete Liste der Start- und Zielhäfen zur Verfügung.

Der Workflow zu den Start- und Zielhäfen, wurde an einigen Punkten ergänzt. So ist zum Beispiel bei der ersten Sichtung der Ergebnisse aufgefallen, dass bei den Ergebnissen, die der Workflow liefert, nicht alle Daten korrekt erkannt wurden. Zudem müssen die offiziellen Hafenlisten um Binnen- oder kleinere Stadthäfen aus Deutschland und den Niederlanden ergänzt werden. Der Grund für die Ergänzung ist, dass sich in den Datensätzen auch Schiffe mit Zielhäfen innerhalb von Deutschland befinden, die z.B. kleinere Stadthäfen wie Oldenburg oder Groningen ansteuern.

Ergänzt wurde die Hafenliste um mehrere Binnenhäfen, sowie alle Häfen der ostfriesischen Inseln. Siehe hierzu folgende Abbildung als Beispiel.

	id smallint	name character varying (50)	country character varying (50)	unctad character varying (50)	alias character varying (50)	position point
92	1868	Borkum	Germany	DE BOR	Borkum (DE BOR)	(53.600983,6.70587)
93	1869	Bant	Germany	DE BAN	Bant (DE BAN)	(52.766339,5.75225)
94	1870	Brauerplate	Germany	DE BRA	Brauerplate (DE BRA)	(53.63248,6.74697)
95	1871	Kachelotplate	Germany	DE KAC	Kachelotplate (DE KAC)	(53.642998,6.832268)
96	1872	Luetje Hoern	Germany	DE LUE	Luetje Hoern (DE LUE)	(53.590836,6.866801)
97	1873	Memmert	Germany	DE MEM	Memmert (DE MEM)	(53.638296,6.882891)
98	1874	Juist	Germany	DE JUI	Juist (DE JUI)	(53.682511,7.008312)
99	1875	Buise	Germany	DE BUI	Buise (DE BUI)	(52.351621,4.872486)
100	1876	Norderney	Germany	DE NOR	Norderney (DE NOR)	(53.70664,7.145552)
101	1877	Baltrum	Germany	DE BAL	Baltrum (DE BAL)	(53.728016,7.369783)
102	1878	Langeoog	Germany	DE LAN	Langeoog (DE LAN)	(53.745586,7.488382)
103	1879	Spiekeroog	Germany	DE SPI	Spiekeroog (DE SPI)	(53.770953,7.693411)
104	1880	Wangerooge	Germany	DE WAN	Wangerooge (DE WAN)	(53.789824,7.903767)
105	1881	Minsener Oog	Germany	DE MIN	Minsener Oog (DE MIN)	(53.754614,8.014517)
106	1882	Mellum	Germany	DE MEL	Mellum (DE MEL)	(53.723255,8.148407)

Abbildung 7-10: Harbour Tabelle

Die Tabelle „harbour“ auf der Datenbank „Magellan“ enthält zum jetzigen Zeitpunkt natürlich nicht alle offiziellen Hafennamen, die weltweit existieren. Die Tabellen lassen sich über einfache Update-Funktion in SQL um neue Häfen ergänzen. Ein Datensatz der harbourtabelle setzt sich aus einer fortlaufenden Nummerierung „id“, dem Namen des Hafens „name“, dem dazugehörigen Land „country“, einem Kürzel, welches sich aus dem offiziellen Länderkürzel und den offiziellen Hafenkürzeln zusammensetzt „unctad“, einem Alias Namen „alias“, der sich aus dem Namen des Hafens und dem Kürzel „unctad“ zusammensetzt und einer Position „position“, bestehend aus den Koordinaten des Hafens, zusammen. Die Projektgruppe hat den Workflow an folgenden Punkten überarbeitet. Die Daten, die vorher aus CSV-Dateien geladen wurden, Windparks und Länderkürzel, befinden sich nun in der Datenbank „Magellan“ unter folgenden Tabellen:

CSV-Datei	Datenbanktabelle
WindparkListe	Offshore_wind_farm
ISO_Alpha_2	countrycode

Tabelle 7-8: Veränderung Destination cleaner

7.7 Kartenmaterial

Zur Bestimmung der örtlichen Gegebenheiten im Untersuchungsgebiet wurde manuell eine Geojson-Datei erstellt, welche die verschiedenen Bereiche beinhaltet. Dabei wird bei den Bereichen zwischen der Nordsee, Weser, Elbe und dem Nord-Ostsee-Kanal unterschieden. Die Gebiete sind unter dem *geometry*-Objekt als Polygone abgelegt. Zusätzlich enthält jedes Gebiet ein *rule*-Objekt, welches die Maximalgeschwindigkeit sowie die Mindestlänge eines Schiffes, für die die Geschwindigkeit gilt, beinhaltet. Außerdem muss das Regelgebiet ergänzt werden, welches die Regeln, die vor Ort gelten, festlegt. Diese Datei kann jederzeit erweitert werden, dafür wird, wie in Abbildung 7-11 zu sehen, ein Objekt angelegt, welches *type*-, *name*-, *rule*-, *properties*- und *geometry*-Objekte enthält.

```
{
  "type" : "FeatureCollection",
  "features" :
    [
      {
        "type" : "OPENSEAAREA",
        "name" : "Nordsee",
        "rule": {
          "maxspeed": "2147483647",
          "minLength": "1",
          "rulearea": "KVR"
        },
        "properties": {
          "SCAMIN": "2147483647",
          "SCAMAX": "0"
        },
        "geometry": {
          "type": "Polygon",
          "coordinates": [
            [[8.682334, 53.891950, 0],[8.603830, 53.876599, 0],
            [8.551192, 53.790858, 0],[8.682334, 53.891950, 0]]
          ]
        }
      }
    ]
}
```

Abbildung 7-11: Geojson-Datei mit einem Gebietsobjekt

Das *rule*-Objekt dient der Überprüfung der sicheren Geschwindigkeit, sowie der Bestimmung, welche Regeln für das jeweilige Szenario Anwendung finden. Darauf wird im anschließenden Kapitel Regelkonzept eingegangen.

Des Weiteren standen Kartendaten vom Chartserver zur Verfügung. Aus diesen Kartendaten wurden die Informationen zu den Standorten der Gates (Schleusen) entnommen, diese diente als Geojson-Datei dem SeaChartRules-Knoten als Input. Der Knoten kann mit dieser Information ausgeben, ob eine Überholverbotszone vorliegt oder nicht. Es können weitere Karten von diesem Server, die beispielweise Brücken enthalten, ergänzt werden, indem ein weiterer Inputparameter im SeaChartRules-Knoten angelegt wird. Des Weiteren muss diese zusätzliche Information in der zugrundeliegenden Methode ergänzt werden, sodass Brücken ebenfalls als Überholverbotszonen gelten.

Auch der SeaChartFilter-Knoten nutzt Geojson-Dateien. Hier können bewusst Schiffspositionen rausgefiltert werden, welche nicht betrachtet werden sollen. Beispielsweise mit Hilfe einer Karte, welche die Hafenstandortinformationen enthält, können Koordinaten in diesem Bereich rausgefiltert werden.

7.8 Wetterkonzept

Ein weiterer Risikofaktor, der betrachtet werden muss, ist das Wetter. Dabei muss unterschieden werden, welche Wetterparameter einen tatsächlichen Einfluss auf das Verhalten eines Schiffes haben. Wann liegt schlechtes bzw. gutes Wetter für eine Begegnungssituation vor? Diese Frage soll im Folgenden geklärt werden. Zur Verfügung stehen die in Tabelle 7-1 und in Tabelle 7-2 aufgeführten Wetterdaten. In der Projektgruppe MaGellan wurden die Wetterparameter aus Tabelle 7-9 verwendet.

Parametername	Beschaffungsort
Windgeschwindigkeit	BSH
Wellenhöhe	BSH
Windrichtung	BSH

Sichtweite	DWD
------------	-----

Tabelle 7-9: Verwendete Wetterparameter

Dabei hat die Windgeschwindigkeit, sowie die Wellenhöhe einen Einfluss darauf, ob ein Schiff rausfahren kann, sodass diese beiden Werte mit aufgenommen wurden. Da Szenarien von sich bewegenden Schiffen untersucht werden und angenommen wird, dass daher der Wasserstand hoch genug ist, spielt dieser im Verlauf der Projektgruppe keine Rolle mehr. Ebenfalls wird die Windrichtung betrachtet, bei der untersucht werden muss, welchen Einfluss diese auf das Verhalten von Schiffen in einer Situation hat. Die Daten vom DWD wurden gesondert betrachtet, da diese an eine Wetterstation gebunden sind und sich daher nicht speziell auf den maritimen Wetterzustand abbilden lassen. Insgesamt wurde die Sichtweite in den Vordergrund der DWD-Daten geschoben, da erwartet wurde, dass sich Schiffe bei nebligem Wetter anders verhalten.

Die Einteilung, welcher Wetterzustand vorliegt, wurde mittels einer Fuzzy-Logic umgesetzt, welche die beiden maritimen Wetterparameter Windstärke und Wellenhöhe für die Position des Schiffes als Inputparameter erhält. Dabei wurden die Parametereinteilungen in die *Fuzzysets* von einer Wahrscheinlichkeitsdichtefunktion abgeleitet, welche die Häufigkeit der einzelnen Wetterparameter im Untersuchungsgebiet zeigt.

In Abbildung 7-12 ist die Schätzung der Wahrscheinlichkeitsdichtefunktion der Windgeschwindigkeit zu sehen. Da der höchste detektierte Wert bei 18 liegt, ist dies der Endwert des Sets. Begonnen wird bei 0. Es wird pro Begegnungssituation ein Wetterzustand bestimmt. Die Fuzzy-Regeln stellen dabei die verschiedenen Kombinationen der Input-Werte dar und berechnen einen Wetterzustand, welcher im Bereich zwischen [0;1] liegt. Dabei entspricht 1 einem sehr guten Wetterzustand.

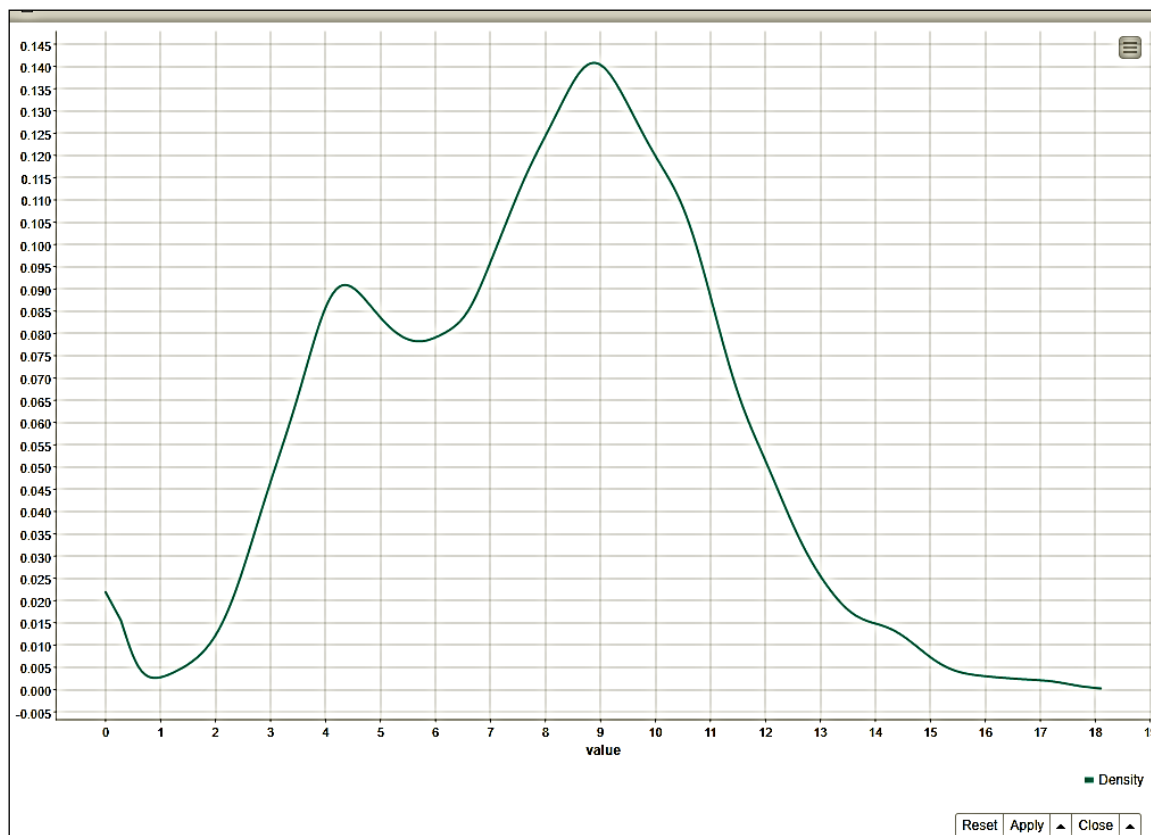


Abbildung 7-12: Wahrscheinlichkeitsdichtefunktion von Windgeschwindigkeit

Danach wird der ermittelte Wetterzustand gleichmäßig auf fünf Klassen eingeteilt. Die Windrichtung wird unterschieden in landein- und landauswärts. Die Einteilung richtet sich nach der Küstenlinie der Nordsee im Untersuchungsbereich. In Abbildung 7-13 sind die zwei Bereiche eingezeichnet. Wenn der Wind zwischen 270° und 40° weht wird er als landeinwärts deklariert, andernfalls als landauswärts. Diese Methode ist im *Node-Model* des Wetterzustands-Knoten implementiert und dort erweiterbar. An dieser Stelle kann, als mögliche Erweiterung, der Winkelwert der Windrichtung mit der Ausrichtung des Schiffes in Beziehung gesetzt werden. Je nach Kombination dieser beiden Werte kann die Windrichtung in Kategorien eingeteilt werden, welche das Risikolevel abbilden.



Abbildung 7-13: Winkeleinzeichnung für die Windrichtung

Neben der Windrichtung wird ebenfalls die Sichtweite einzeln betrachtet. Diese wird, wie in Tabelle 7-10 zu sehen, eingeteilt. Der Bereich für eine sehr gute Sichtweite beginnt bei 3704 Metern (zwei Seemeilen), obwohl dieser keine klare Sicht im eigentlichen Sinne darstellt. Ab dieser Weite sollte die Sicht kein Risiko in Begegnungssituationen darstellen, da diese erst ab 3704 Meter detektiert werden.

Sichtweitenklasse	Sicht in Metern
Sehr gut	≥ 3704
Gut	< 3704 und ≥ 2000
Mittel	< 2000 und ≥ 1000
Schlecht	< 1000 und ≥ 100
Sehr schlecht	< 100

Tabelle 7-10: Einteilung der Sichtweite in Klassen

Wenn die Strömungsdaten eingetragen sind, können diese ebenfalls zur Betrachtung des Risikos hinzugezogen werden. Diese sind durch das zugrundeliegende Untersuchungsgebiet, auch in Bezug auf den Wasserstand und den Wind von großem Interesse.

Damit man Wetterdaten hinzufügen kann, müssen diese als Input für den Weatherstate-Knoten ergänzt werden, dies geschieht im Dialog. Ebenfalls muss ein neuer Input, wenn dieser mit Wind oder Wellen korreliert, in der weatherstate_rules.fcl ergänzt werden. Diese befindet sich unter dem Pfad de.emir.magellan.emirweather.resources in Git. Dafür wird, wie in Abbildung 7-14 dargestellt, ein neuer Input reingeben und ein *Fuzzysset* für die verschiedenen Klasse, die der Input annehmen kann, erzeugt. Schlussendlich muss der neue Input mit den anderen Inputfaktoren in den *Fuzzyrules* verknüpft werden.

```
VAR_INPUT                // Define input variables
  newInput : REAL;
END_VAR

VAR_OUTPUT               // Output variable
  weatherstate : REAL;
END_VAR

FUZZIFY newInput        // Fuzzify input variable
  TERM very_low := (0, 1) (2, 1) (6, 0);
  TERM low := (2, 0) (7, 1) (9, 0);
  TERM medium := (7, 0) (10, 1) (14, 0);
  TERM high := (10, 0) (15, 1) (18, 0);
  TERM very_high := (15, 0) (18, 1);
END_FUZZIFY
```

Abbildung 7-14: Neuer Inputeintrag mit dazugehörigem Fuzzysset

7.9 Regelkonzept

Damit im Folgenden überprüft werden kann, ob eine Regel eingehalten wurde, muss zuerst kontrolliert werden, ob diese momentan aktiv ist. Dafür werden die Regelgebiete den eingespeisten Seekarten entnommen. Unter einem Regelgebiet wird ein Bereich verstanden, in dem die gleichen Gesetze gelten, beispielsweise gilt auf der Nordsee die KVR. Die Geschwindigkeitsbegrenzungen (KVR 6) werden ebenfalls als Parameter in den Kartendateien abgelegt. Wenn es

Geschwindigkeitsänderungen innerhalb eines Gebietes gibt, werden diese mittels eines Punktes, ab dem die Änderung stattfindet, festgehalten. Dieser Punkt liegt auf einer Linie, die sich durch die Mitte des Gebietes zieht. Von dem Geschwindigkeitsänderungspunkt wird eine Linie zum Gebietsrand gebildet. So entstehen kleinere Bereiche, in denen die gleiche Geschwindigkeit gilt. Wenn ein Szenario erfasst wurde, wird überprüft, welche Regeln in dem Gebiet gelten und anschließend werden die aktiven Regeln mit dem tatsächlichen Verhalten des Schiffes verglichen. Dabei gibt es eine Unterteilung der Regeln in Soft- und Hardrules. Diese Einteilung ist in Tabelle 7-11 festgehalten.

Hardrules sind Regeln, an die sich Schiffe strikt halten müssen. Softrules hingegen werden genutzt, wenn eine Regel keine konkrete Auslegung hat. So ist beispielsweise die Breite von engem Fahrwasser nicht klar definiert. Eine weitere Softrule besagt, dass Schiffe standardmäßig links überholen müssen, in Ausnahmefällen aber auch rechts überholen dürfen. Hier ist zu überprüfen, ob der Verstoß gegen diese Regel überhaupt ein Problem darstellt.

Regel	Softrule	Hardrule
KVR 6		X
KVR 9	X	
KVR 14		X
SeeSchStrO 23.a	X	
SeeSchStrO 23.b	X	
SeeSchStrO 23.c		X

Tabelle 7-11: Einteilung der umgesetzten Regeln in Soft- und Hardrules

Die KVR 9 beinhaltet, wie sich Schiffe verhalten müssen, wenn sie sich in engem Fahrwasser befinden. Da enges Fahrwasser nicht klar definiert ist, wurde dafür die Breite des Schiffes in Abhängigkeit zur Breite des Gebietes bestimmt. Wenn diese

Regel aktiv ist, dann müssen sich die Schiffe am steuerbordseitigen Rand halten, dies wird über die Distanz rechts und links vom Schiff überprüft. Die Distanz muss auf der rechten Seite kleiner sein.

Die KVR 14 überprüft zu welcher Seite zwei Schiffe, welche sich in einer Head-On-Situation befinden, aneinander vorbeifahren. Hierzu wird sich dem Bearing der beiden Schiffe zueinander bedient.

SeeSchStrO 23.a nutzt ebenfalls das Bearing der Schiffe, um zu überprüfen, zu welcher Seite die beiden Schiffe aneinander vorbeifahren. Dabei liegt eine Overtaking-Situation zugrunde. Wiederum gibt 23.b an, dass die Geschwindigkeit des zu überholenden Schiffes verlangsamt werden muss, daher wird der Trend des SOG-Wertes per linearer Regression beurteilt. Bei der Regel 23.c wird überprüft, ob der Standort der Overtaking-Situationen innerhalb einer Überholverbotszone liegt. Überholverbotszonen sind *Gates* (Schleusen), welche als Geojson-Datei eingebunden werden und enge Fahrwasser, welche über die Regel 9 der KVRs festgelegt sind.

Den einzelnen Regeln ist ein RuleChecker übergeordnet, welcher die Group, die RuleID und das Ergebnis der Regeln benötigt. Die Group ist der Bezeichner für den Regelbereich, wie KVR oder SeeSchStrO und die RuleID gibt die Regelnummer zurück, beispielsweise bei KVR 9 eine 9. Soft- und HardRuleChecker erben von dieser Klasse.

Weitere Regeln können als neue Klassen unter `de.emir.magellan.rules.model.impl` angelegt werden, diese müssen von Soft- oder HardRuleChecker erben und eine Group und RuleID besitzen.

Unter dem RulesNodePlugin können Regeln ausgeblendet werden, indem diese auskommentiert werden. Neue Regeln müssen hier ergänzt werden.

7.10 Schiffsklassen

Innerhalb der Projektgruppe wurden diverse Modelle entwickelt und Ergebnisse erarbeitet. Da es naheliegt, dass sich unterschiedliche Schiffe auch unterschiedlich verhalten, wurde die Menge an Schiffen in sieben Klassen eingeteilt. So können die einzelnen Modelle und Ergebnisse für unterschiedliche Klassen erhoben und

verglichen werden. Dadurch können auch Verhaltensunterschiede zwischen unterschiedlichen Schiffen aufgedeckt werden.

Als Basis für die Unterteilung wurde ein für jedes Schiff individuell berechneter Wert verwendet, der sich aus der Größe und dem Tiefgang des Schiffes ergibt. Diese beiden Werte haben neben vielen weiteren Einflussfaktoren, wie dem Ruder oder der Anzahl an Schiffspropellern, einen Einfluss auf die Manövrierbarkeit eines Schiffes (vgl. American Bureau of Shipping 2006, 29). Da für die genaue Berechnung der Manövrierbarkeit nicht alle Informationen in den Daten vorhanden sind, wurde diese mit Hilfe der Länge und dem Tiefgang approximativ bestimmt.

Die Berechnung der Manövrierbarkeit basiert auf einer Fuzzy-Logik. Als Inputfaktoren dienten die bereits oben erwähnten Variablen – Länge und Tiefgang. Als Ergebnis wird ein Wert zwischen 0 und 1 errechnet, welcher Aussagen darüber erlaubt, wie variabel ein Schiff seinen Kurs verändern kann. So haben Schiffe, die prinzipiell kleiner sind und einen geringeren Tiefgang besitzen eine höhere Manövrierbarkeit, als ein großes Schiff mit einem hohen Tiefgang.

Basierend auf dem Wert der Manövrierbarkeit wurde anschließend versucht, Schiffsklassen zu bilden, sodass die Varianz innerhalb einer Klasse möglichst gering, die Varianz zwischen den Klassen aber möglichst hoch ist. Hierzu wurde zunächst fünf gleichgroße Klassen gebildet. Die fünf Klassen wurden anschließend mit Hilfe einer Kerndichteschätzung jeweils in Abhängigkeit zu jeder Inputvariable untersucht. Das Ziel war es dabei Klassen zu bilden, deren zwei Kerndichteschätzungen normalverteilt sind. Die einzelnen Klassen sollten dabei zwischen einander aber dennoch einen abweichenden Median der Länge und des Tiefgangs besitzen.

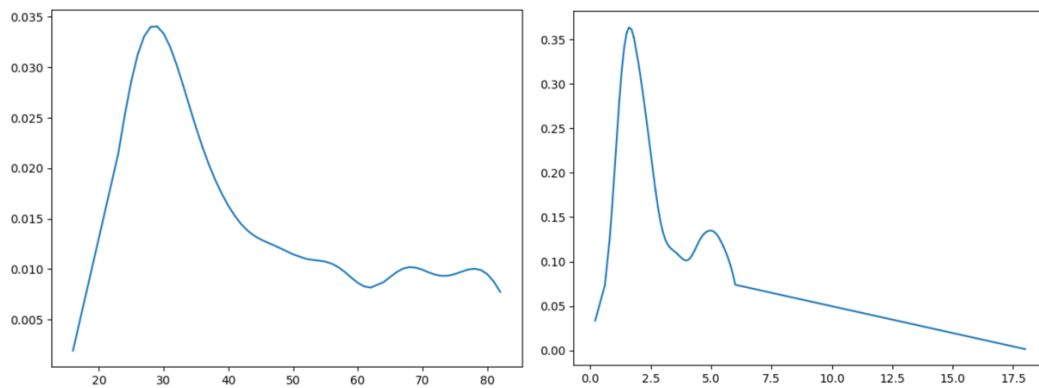


Abbildung 7-15: exemplarische Kerndichteschätzung der Länge und des Tiefgangs einer Klasse

Abbildung 7-15 zeigt die Kerndichteschätzungen von Länge und Tiefgang einer Klasse. Wie zu erkennen ist, sind die beiden Verteilungen annähernd normalverteilt, sodass diese Klasse nicht weiter aufgeteilt werden sollte. Dies traf bei den initial fünf erstellten Klassen allerdings nicht direkt zu, sodass eine weitere Aufteilung der betroffenen Klassen nötig war, sodass schlussendlich sieben Klassen erstellt worden sind.

	Von	Bis	Anteil
Klasse 1	0.0	0.1	0,022
Klasse 2	0.1	0.2	0,034
Klasse 3	0.2	0.4	0,048
Klasse 4	0.4	0.6	0,100
Klasse 5	0.6	0.8	0,181
Klasse 6	0.8	0.9	0,442
Klasse 7	0.9	1.0	0,173

Tabelle 7-12: Einteilung der Schiffe nach Manövrierbarkeit mit dazugehörigem prozentualem Anteil der Klassen

Tabelle 7-12 zeigt die finalen sieben Klassen, die als Basis für die Erstellung von Modellen und Evaluation von Ergebnissen verwendet werden. Klasse 1 enthält die

Schiffe mit der geringsten Manövrierbarkeit, Klasse 7 die mit der höchsten Manövrierbarkeit. Ebenfalls wird deutlich, dass es wesentlich mehr Schiffe mit einer guten Manövrierbarkeit (Klasse 5, 6, 7), als Schiffe mit einer schlechten Manövrierbarkeit (Klasse 1, 2, 3) gibt.

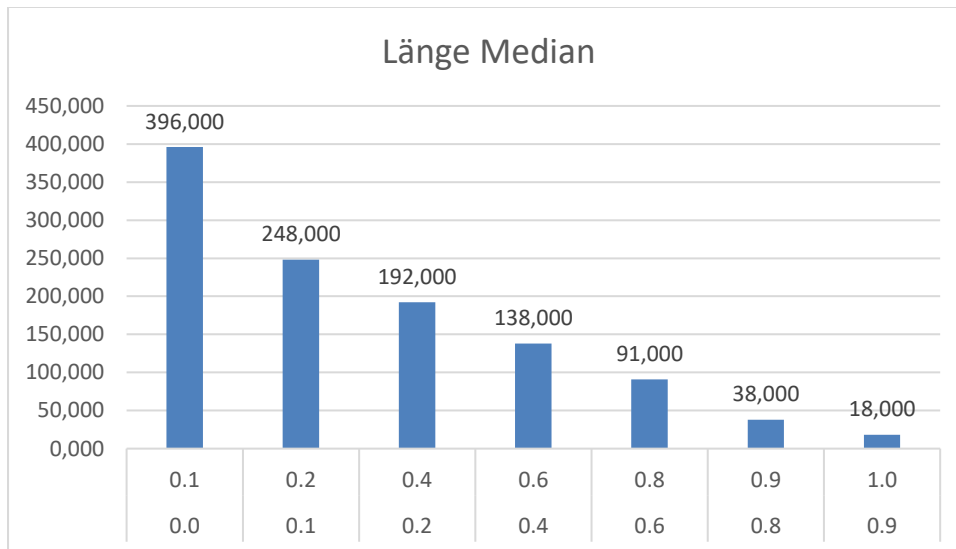


Tabelle 7-13: Median der Länge pro Klasse

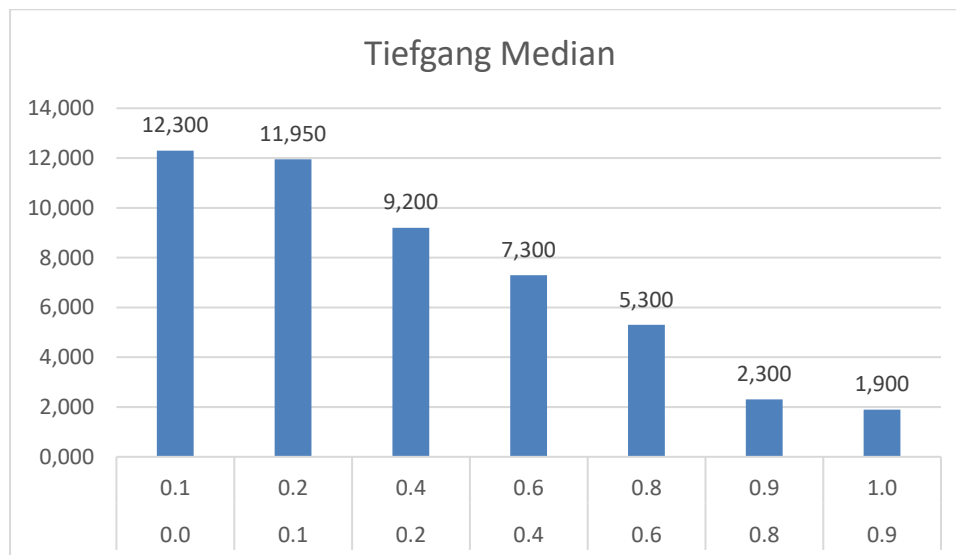


Tabelle 7-14: Median des Tiefgangs pro Klasse

Tabelle 7-13 und Tabelle 7-14 stellen den Median der Länge und des Tiefgangs pro Klasse dar. Bei beiden Einflussfaktoren lässt sich erkennen, dass diese mit einer zunehmenden Manövrierbarkeit abnehmen. Ebenfalls wird deutlich, dass sich die Klassen alle voneinander unterscheiden, sodass eine solche Unterteilung sinnvoll erscheint. Innerhalb der Klassen sind die Werte hingegen stets normalverteilt.

In den folgenden Kapiteln wird erklärt, wie Schiffsbegegnungssituationen aus der gegebenen Datenmenge herausgefiltert werden und bestimmt wird, um was für eine Schiffsbegegnung es sich hierbei handelt. Die detektierten Begegnungssituationen werden anschließend durch verschiedene Parameter klassifiziert und für die erstellten Klassen eine kritische Distanz bestimmt, welche Begegnungssituationen dieser Klasse i.d.R. nicht unterschreiten sollten.

7.11 Schiffsbegegnungssituationen filtern

Die Kollisionsverhütungsregeln (KVR) (vgl. Wasserstraßen- und Schifffahrtsverwaltung des Bundes 1977) unterscheiden in ihren Regeln 13, 14 und 15 zwischen drei Typen von Schiffsbegegnungen, welche u.a. in Abbildung 7-16 dargestellt werden.

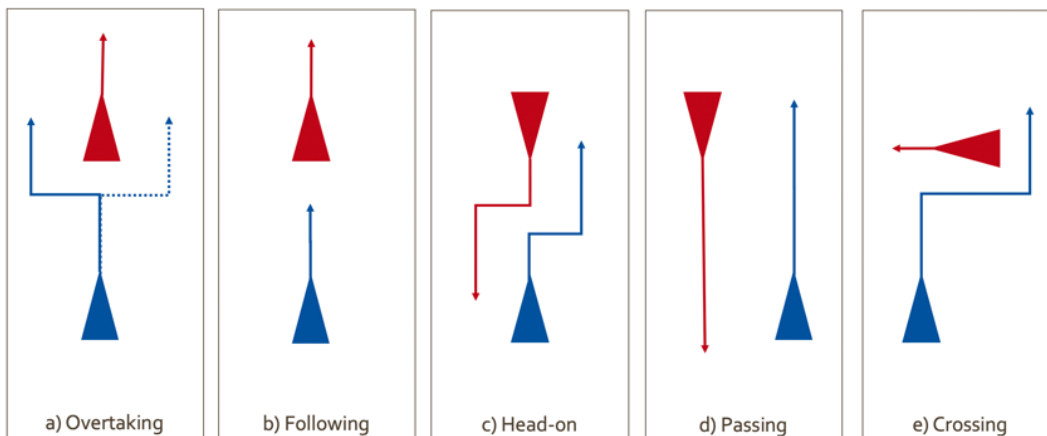


Abbildung 7-16: Schiffsbegegnungstypen

Abbildung 7-16 a) zeigt die Schiffsbegegnung des Überholens (engl. *Overtaking*), bei dem sich das blaue Schiff dem roten Schiff in einer Ausrichtung nähert, in der

beide Schiffe kollidieren könnten (vgl. Wasserstraßen- und Schifffahrtsverwaltung des Bundes 1977, 22). Es wird weder in den KVR, als auch in den COLREGS nicht näher definiert zu welcher Seite hin das überholende Schiff ausweichen muss. In der SeeSchStrO § 23 (1) (vgl. ELWIS 2018) wird jedoch darauf hingewiesen, dass zur Backbordseite überholt werden muss, sofern es die Umstände zulassen.

Abbildung 7-16 c) zeigt einen weiteren Typ einer Schiffsbegegnung, bei der jedoch die Kurse der beiden Schiffe entgegengesetzt sind (engl. *Head-on*). Hierbei müssen die beiden Schiffe nach den KVR ihren Kurs steuerbordseitig anpassen, sodass sie einander backbordseitig passieren (vgl. Wasserstraßen- und Schifffahrtsverwaltung des Bundes 1977, 23).

Abbildung 7-16 e) zeigt eine Schiffsbegegnung, bei der sich die Kurse der beiden Schiffe kreuzen (engl. *Crossing*). Hierbei ist das blaue Schiff ausweichpflichtig, da sich das rote Schiffe auf der Steuerbordseite befindet. Es wird empfohlen auf bugseitiges Kreuzen zu verzichten, somit soll steuerbordseitig ausgewichen werden (vgl. Wasserstraßen- und Schifffahrtsverwaltung des Bundes 1977, 24). Ergänzt wurden das Folgen (engl. *Following*) und das Passieren (engl. *Passing*). Beim Folgen fährt ein Schiff dem anderen hinter her, ohne zu überholen. Beim Passieren fahren die Schiffe aneinander vorbei, ohne aufeinander zu reagieren. Diese beiden Begegnungstypen sind vor allem in Fluss- und Kanalbereichen zu beobachten.

Abbildung 7-17 zeigt den Ablauf, um die vorher genannten Begegnungssituationen aus der gegebenen Datenmenge zu extrahieren.



Abbildung 7-17: Ablauf zur Filterung von Begegnungssituationen

Beim Filtern der fünf genannten Schiffsbegegnungstypen werden zunächst Begegnungssituationen zwischen zwei Schiffspfaden (*Tracks*) herausgefiltert, bei dem die Schiffe mindestens einmal eine bestimmte, zuvor angegebene Distanz zueinander unterschritten haben. Die Distanz von 2 Seemeilen (3704 Metern) wird als relevante Distanz zwischen zwei Schiffen angenommen, hierbei wurde sich u.a. an der Forschungsarbeit von Wang (vgl. Wang 2012, 61) zum Thema der *ship domain* orientiert. Darum kümmert sich ein KNIME-Knoten mit dem Namen `DistanceFilter`.

Für all die dadurch rausgefilterten Begegnungssituationen wird der dichteste Punkt der Annäherung (Closest Point of Approach, kurz CPA) bestimmt.

Im letzten Schritt werden nur noch diejenigen Datenpunkte (*Trackpoints*) berücksichtigt, welche sich zeitlich nah um den CPA befinden. Die Dauer einer Begegnungssituation wird hier mit 30 Minuten definiert. Aus dem Grund werden *Trackpoints* rausgefiltert, welche sich entweder 15 Minuten vor oder nach dem CPA befinden. Diesen Prozess erledigt ein KNIME-Knoten mit dem Namen `MinDistanceFilter`.

Abbildung 7-18 visualisiert die Detektion eines beispielhaften CPA und die Berücksichtigung der zeitlich nahen *Trackpoints* bei einer Begegnungssituation.

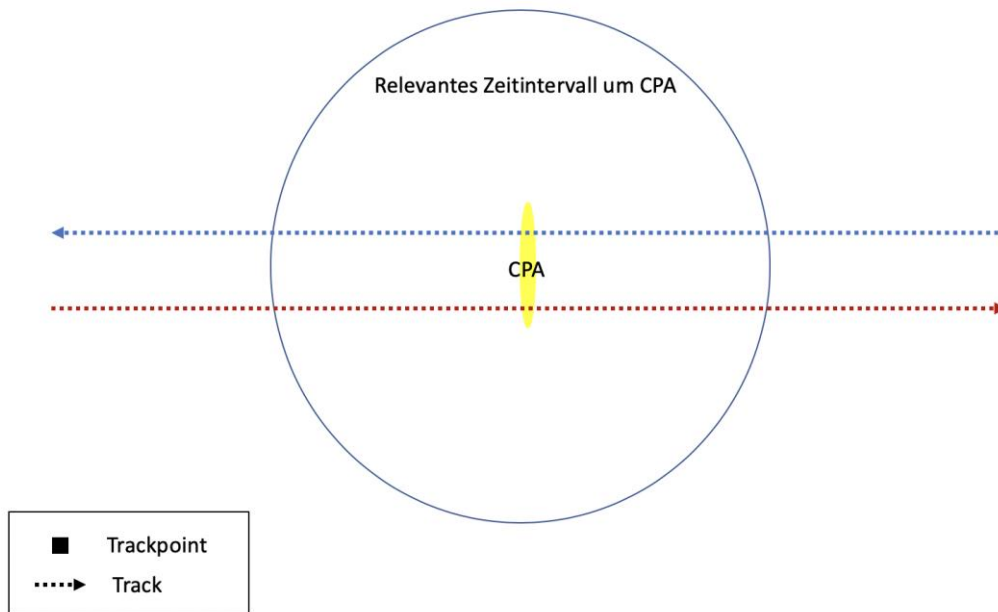


Abbildung 7-18: Detektion eines CPA und Filterung der zeitlich nahen Trackpoints

7.11.1 Schiffsbegegnungstyp bestimmen

Nachdem die relevanten Abschnitte einer Begegnungssituation rausgefiltert wurden, wird im nächsten Schritt bestimmt, um was für Begegnungstypen es sich handelt. Zum Bestimmen der Begegnungstypen wurde ein KNIME-Knoten mit dem Namen *EncounterLabeling* erstellt. Zum Erkennen der Begegnungstypen wird zum einen das *Bearing* des gerade zu betrachtenden Schiffes zum anderen Schiff genommen und zum anderen die Differenz der COG-Werte der beiden Schiffe. Das *Bearing* bezeichnet die Peilung eines anderen Schiffes zum gerade betrachteten Schiff.

Bei der Betrachtung der Bearingwerte werden die jeweiligen Begegnungstypen in jeweils charakteristische Winkelbereiche eingeteilt, wie aus Abbildung 7-19 zu entnehmen ist.

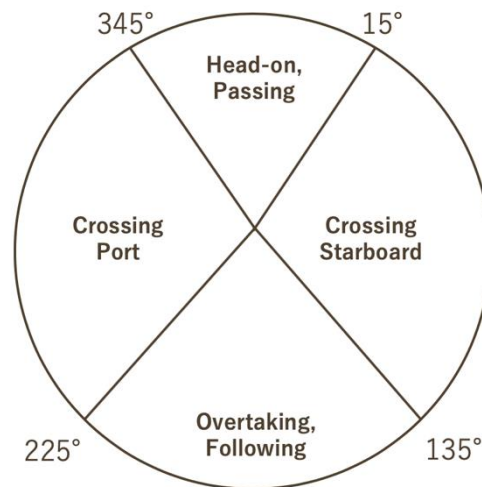


Abbildung 7-19: Winkelbereiche Begegnungstypen

Mit dieser Einteilung wurden in den Forschungsarbeiten von Kuwata, Wolf et. al. (vgl. Kuwata u. a. 2014) für ein autonom fahrendes Schiff, welches selbständig auf Begegnungssituationen mit anderen Schiffen reagieren musste, gute Ergebnisse erzielt. Zunächst werden *Head-on* und *Passing* in einem Winkelbereich zusammengefasst und *Overtaking* und *Following* in einem anderen Winkelbereich.

Anschließend wird für jeden *Trackpoint* eines *Tracks* das *Bearing* zum anderen Schiff berechnet und überprüft, mit welcher Wahrscheinlichkeit es sich bei dem einen Datenpunkt um eine bestimmten Begegnungstyp handelt. Für das Berechnen des *Bearings* werden jeweils die Position der GPS-Sender und die COG-Werte der beiden Schiffe genommen.

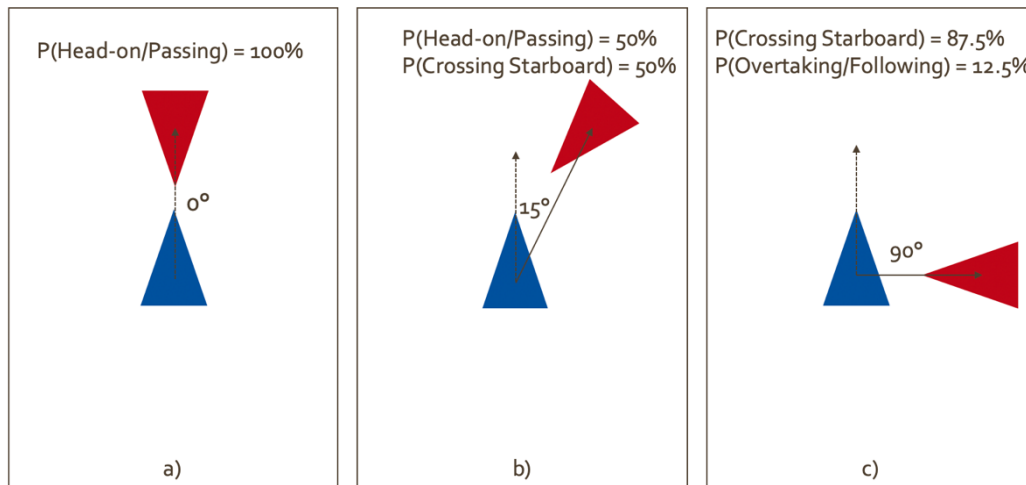


Abbildung 7-20: Beispiele Wahrscheinlichkeitsberechnung

In Abbildung 7-20 wird anhand von Beispielen gezeigt, wie die Wahrscheinlichkeiten berechnet wurden. Abbildung 7-20 a) verdeutlicht, dass die Wahrscheinlichkeit für eine Begegnungssituation bei 100% liegt, wenn sich der Bearingwert in der Mitte eines Winkelbereichs befindet, in diesem Fall in der Mitte des Bereichs *Head-on/Passing*. Abbildung 7-20 b) wiederum verdeutlicht, dass, wenn ein Bearingwert sich zwischen zwei Winkelbereichen befindet, die Wahrscheinlichkeit pro betroffenem Begegnungstyp bei 50% liegt. Die Formel, welche sich hinter der Wahrscheinlichkeitsberechnung befindet, ist:

$$p = 0.5 + \frac{\min(|\beta - \alpha|, |\beta - \gamma|)}{\varepsilon}$$

Hierbei bezeichnet p die Wahrscheinlichkeit, β das *Bearing*, α den minimalen Winkel des betroffenen Bereichs, γ den maximalen Winkel des betroffenen Bereichs und ε die Bereichslänge. Bei dem Beispiel aus Abbildung 7-20 c) ist β bei 90°, α bei 15°, γ bei 135°, da 90° in den Bereich *Crossing Starboard* fällt. ε beträgt 120°, da dies die Differenz zwischen dem minimalen Winkel und maximalen Winkel im Bereich *Crossing Starboard* ist. Setzt man nun die jeweiligen Werte ein, ergibt sich folgendes Ergebnis:

$$p = 0.5 + \frac{\min(|90 - 15|, |90 - 135|)}{120} = 0.875$$

Demnach liegt die Wahrscheinlichkeit, dass es sich für diesen *Trackpoint* um ein *Crossing Starboard* handelt, bei 87.5%. Die Restwahrscheinlichkeit von 12.5% fällt in den Bereich *Overtaking/Following*, da dieser Bereich dem Wert 90° noch am nächsten ist.

Diese Wahrscheinlichkeitsberechnung wird nun für jeden *Trackpoint* bis zum CPA durchgeführt, da sich die Schiffe nach dem CPA voneinander wegbewegen. Am Ende werden die durchschnittlichen Wahrscheinlichkeiten der jeweiligen Bereiche berechnet.

Wenn die durchschnittliche Wahrscheinlichkeit für einen Bereich mindestens den Wert 50% erreicht, wird die jeweilige Begegnungssituation weiter betrachtet. Der Wert von 50% hat sich nach einigen Versuchen mit verschiedenen Schwellenwerten als geeignet erwiesen.

Das gleiche Verfahren wird nun für die Differenz der COG-Werte der beiden Schiffe durchgeführt, jedoch ändern sich die Winkelbereiche der jeweiligen Begegnungssituationen (Abbildung 7-21). Das Verfahren wird nun auch mit der Differenz der COG-Werte gemacht, weil das alleinige Betrachten des *Bearings* die Begegnungstypen nicht zuverlässig genug klassifiziert hat. Das Problem beim *Bearing* ist, dass dies zwar korrekt ausgibt, in welchem Bereich sich das Schiff andere Schiff befindet, jedoch nicht in welcher Richtung sich das Schiff im Vergleich zum gerade zu betrachtenden Schiff befindet.

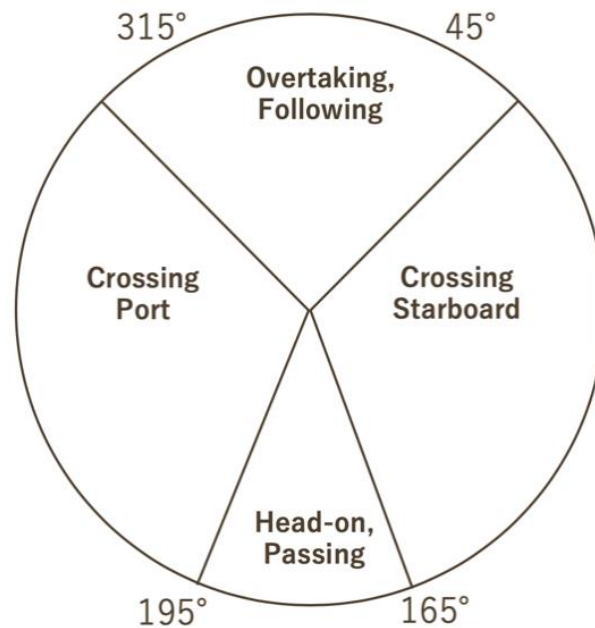


Abbildung 7-21: Winkelbereiche für Differenzen der COG-Werte

Mithilfe der Differenz kann jedoch eine Aussage getroffen werden, in welcher relativen Richtung das andere Schiff fährt. Wenn beispielsweise für die Differenz der COG-Werte ein Wert zwischen 165° und 195° ausgegeben wird, bewegt sich das andere Schiff in einer entgegengesetzten Richtung.

Für die Berechnung der COG-Differenzen wird folgende Formel verwendet:

$$\delta = \theta_1 - \theta_2 + 360 \% 360$$

Hierbei bezeichnet δ die Differenz und θ der COG-Wert eines Schiffes. Demnach ändert sich die Berechnung für die jeweiligen Wahrscheinlichkeit wie folgt:

$$p = 0.5 + \frac{\min(|\delta - \alpha|, |\delta - \gamma|)}{\varepsilon}$$

Bei der Berücksichtigung der Differenzen der COG-Werte muss im Durchschnitt für einen Winkelbereich mindestens die Wahrscheinlichkeit von 40% betragen, damit die Begegnungssituation weiter betrachtet wird. Wie vorher beim *Bearing*

hat sich nach einigen Versuchen mit verschiedenen Schwellenwerten der Wert von 40% bei der Differenz als geeignet erwiesen.

Es kann allerdings durchaus passieren, dass für den einen *Track* die Bedingungen für einen Begegnungstyp wie *Head-on/Passing* erfüllt sind, jedoch nicht für den anderen *Track*. Aus dem Grund wird eine Begegnungssituation erst dann einem Begegnungstyp zugeordnet, wenn die *Tracks* bei den durchschnittlichen Wahrscheinlichkeiten keine Widersprüche liefern. In Abbildung 7-22 werden die erforderlichen Kombinationen für die einzelnen Begegnungstypen als Pseudocode dargestellt.

```
def checkEncounterType():
    if p_overtaking_following_bearing_track1 >= 0.5
       and p_head_on_passing_bearing_track2 >= 0.5
       and p_overtaking_following_difference_track1 >= 0.4
       and p_overtaking_following_difference_track2 >= 0.4:

        return is_overtaking_following

    elif p_head_on_passing_bearing_track1 >= 0.5
       and p_head_on_passing_bearing_track2 >= 0.5
       and p_head_on_passing_difference_track1 >= 0.4
       and p_head_on_passing_difference_track2 >= 0.4:

        return is_head_on_passing

    elif p_crossing_right_bearing_track1 >= 0.5
       and p_crossing_left_bearing_track2 >= 0.5
       and p_crossing_right_difference_track1 >= 0.4
       and p_crossing_left_difference_track2 >= 0.4:

        return is_crossing

    else:
        return is_undefined
```

Abbildung 7-22: Erforderliche Kombinationen für einen Begegnungstyp

Beim *Overtaking/Following* muss beim *Bearing* des einen Schiffes die Wahrscheinlichkeit *Overtaking/Following* größer 0.5 sein, beim anderen Schiff jedoch *Head-on/Passing* größer 0.5, da überholende Schiff bis zum CPA hinter dem zu überholenden Schiff ist. Bei der Differenz der COG-Werte muss bei beiden die Wahrscheinlichkeit *Overtaking/Following* größer 0.4 sein, da dies mit

hoher Wahrscheinlichkeit aussagt, dass beide Schiffe ungefähr in die gleiche Richtung fahren. Beim Fall *Head-on/Passing* müssen beide *Tracks* die gleichen Bedingungen für *Head-on/Passing* erfüllen.

In einer *Crossing-Situation* muss bei einem Schiff die Wahrscheinlichkeit für *Crossing Port* hoch sein, bei dem anderen Schiff allerdings *Crossing Starboard*, damit es sich um ein *Crossing* handelt. Erst wenn die im Pseudocode aufgeführten Bedingungen erfüllt wurden, werden die Begegnungssituation dem jeweiligen Typ zugewiesen. Ansonsten werden die Begegnungssituationen als *undefined* definiert.

Zuletzt muss noch zwischen *Head-on* und *Passing* und zwischen *Overtaking* und *Following* unterschieden werden. Des Weiteren muss überprüft werden, ob es sich bei den herausgefilterten *Crossing-Situationen* wirklich um *Crossing-Situationen* handelt. Denn mithilfe der Winkelwerte kann nicht eindeutig festgestellt werden, ob sich die Schiffe wirklich gekreuzt haben. In manchen Fällen wechselt ein Schiff das Fahrwasser zu einem anderen Schiff und es wird durch die Winkelwerte fälschlicherweise ein *Crossing* ausgegeben.

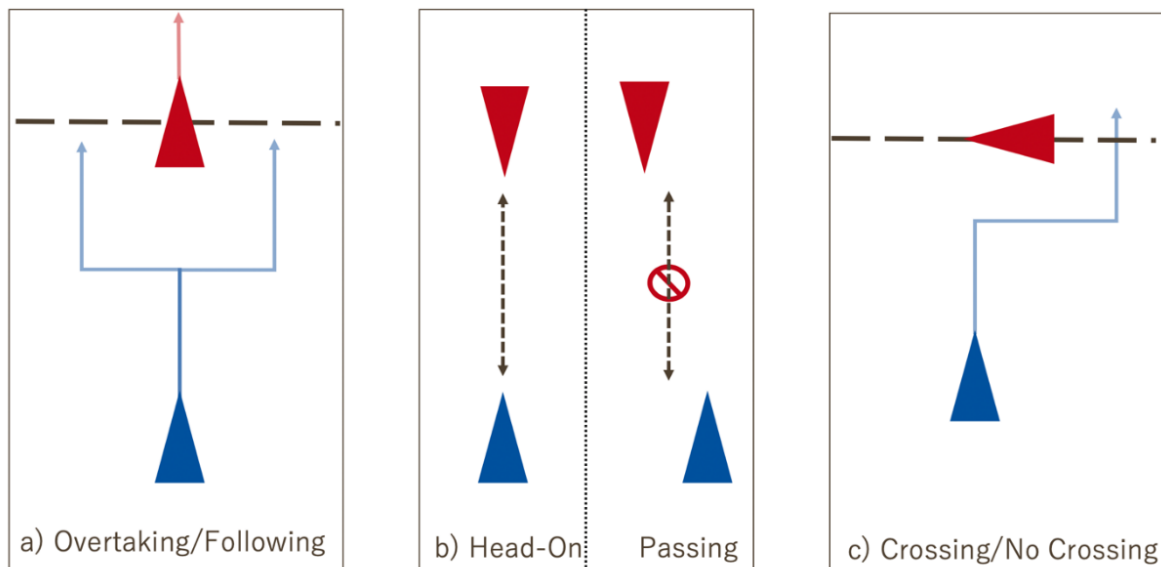


Abbildung 7-23: Weitere Unterscheidung der Begegnungstypen

Abbildung 7-23 visualisiert die weitere Unterscheidung der Begegnungstypen. Bei der Unterscheidung zwischen *Overtaking* und *Following* Abbildung 7-23 a)

wird im 90°-Winkel zur Mitte des roten Schiffs eine Linie gezogen und überprüft, ob diese vom blauen Schiff überquert wird. Falls diese vom blauen Schiff überquert wird, handelt es sich um eine *Overtaking*-Situation, andernfalls um *Following*. Um diese Aufgabe kümmert sich ein KNIME-Knoten mit dem Namen *OvertakingDifferentiator*. Bei einer *Head-on*-Situation müssen sich die Bahnen der beiden Schiffe schneiden wie in Abbildung 7-23 b), andernfalls handelt es sich um *Passing*. Hierfür wird der Knoten *Encounterlabeling* genutzt, welcher neben den Wahrscheinlichkeitsberechnungen pro *Trackpoint* überprüft, ob sich die Schiffslinien frontal schneiden. Zuletzt müssen sich bei einer *Crossing*-Situation die Schiffslinien mindestens einmal kreuzen, wie es in Abbildung 7-23 c) demonstriert wird. Hierbei wird geschaut, ob ein eines der beiden Schiffe die bisher abgefahrene Schiffsroute des anderen Schiffes schneidet, in Abbildung 7-23 c) wird das blaue Schiff die abgefahrene Route des roten Schiffes schneiden. Für diese Aufgabe wird ein Knoten mit dem Namen *CrossingDifferentiator* genutzt.

Im Folgenden wird ein Beispiel einer Begegnungssituation gezeigt, welche durch das oben genannte Verfahren aus der gegebenen Datenmenge herausgefiltert und einem Begegnungstyp zugewiesen wurde. In Abbildung 7-24 wird die beispielhafte Begegnungssituation gezeigt. Die Distanz am CPA beträgt ungefähr 568 Meter.

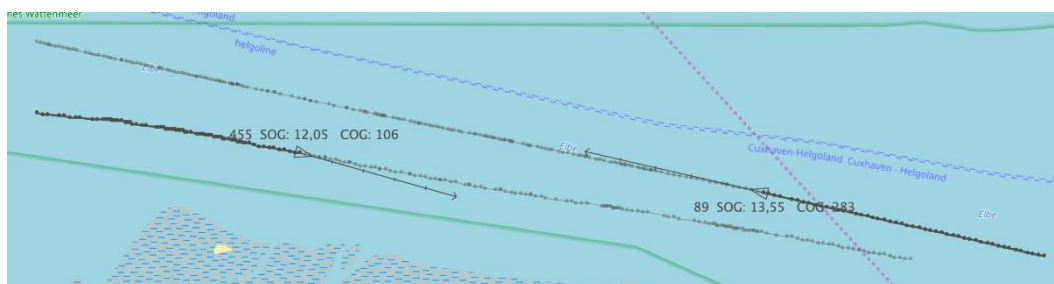


Abbildung 7-24: Richtung der Schiffe der beispielhaften Begegnungssituation

In Abbildung 7-25 sieht man den Verlauf der Wahrscheinlichkeiten für den *Track* 455 für die einzelnen Begegnungstypbereiche unter Berücksichtigung der Differenz der COG-Werte bis zum CPA. Hierbei bleibt die Wahrscheinlichkeit für

Head-on/Passing konstant bei über 0.6. Im Durchschnitt ergibt sich hier ein Wert von 0.9.

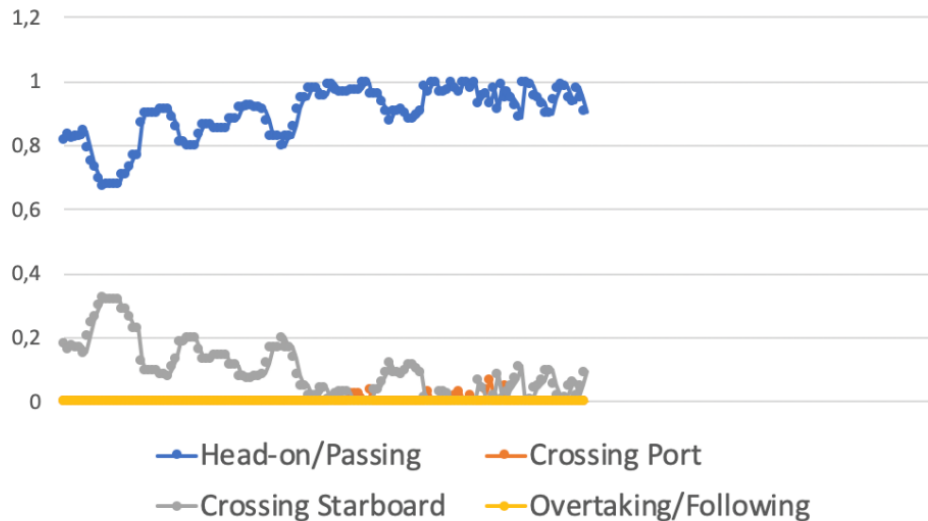


Abbildung 7-25: Wahrscheinlichkeitsverlauf Track 455 mit Differenz der COG-Werte

Wird nun der Wahrscheinlichkeitsverlauf unter Berücksichtigung des *Bearings* betrachtet fällt auf, dass sich der Wahrscheinlichkeitswert für *Head-on/Passing* bis zum CPA auf den Wert 0 reduziert. Dies passiert aus dem Grund, dass sich nach dem CPA das andere Schiff hinter dem ersten Schiff befindet (*Track 455*). Im Durchschnitt ergibt sich jedoch ein Wert von 0.793 für *Head-on/Passing*. Für den anderen *Track* werden auch die jeweiligen Bedingungen für *Head-on/Passing* durch die Wahrscheinlichkeitsverläufe erfüllt.

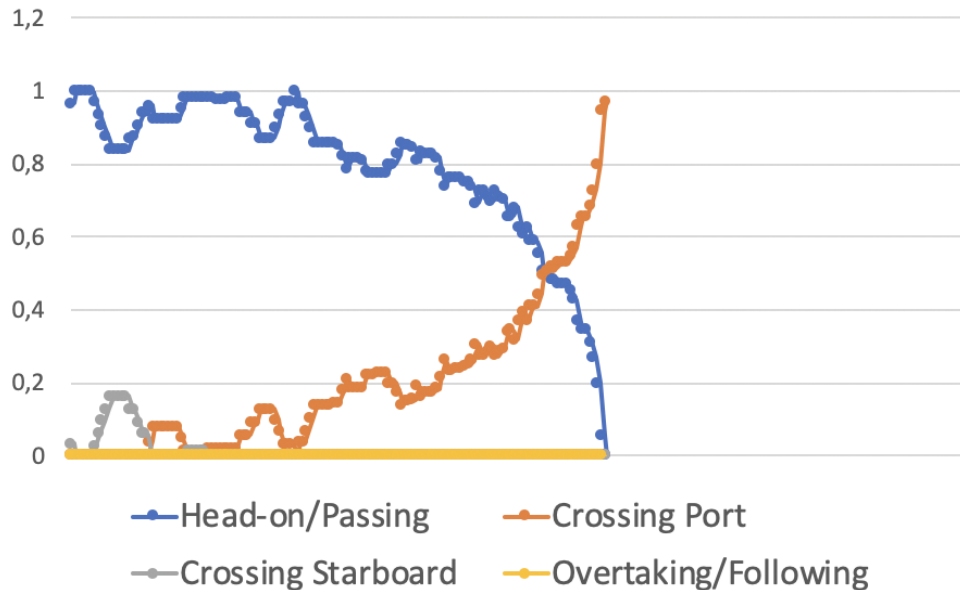


Abbildung 7-26: Wahrscheinlichkeitsverlauf Track 455 mit Bearing

Zuletzt wurde diese Situation durch den Prozess als *Passing* klassifiziert, da die beiden Schiffe zu keinem Zeitpunkt frontal zueinander ausgerichtet sind.

7.12 Klassifikation von Schiffsbegegnungssituationen zur Detektion von anomalen Begegnungssituationen

Die extrahierten Schiffsbegegnungssituationen wurden nicht nur nach dem Schiffsbegegnungstyp unterschieden, sondern durch zusätzliche Parameter klassifiziert.

Folgende Parameter wurden berücksichtigt, um die Schiffsbegegnungssituationen zu klassifizieren:

- Schiffsbegegnungstyp
- Gewässergebiet
- Tageszustand
- Wetterzustand
- Sichtweite
- Windrichtung
- Manövrierfähigkeit

Bei dem Gewässergebiet wird zwischen Nordsee, Flussgebiet und Kanalgebiet unterschieden. Die betrachteten Bereiche werden in Abbildung 7-27 dargestellt.

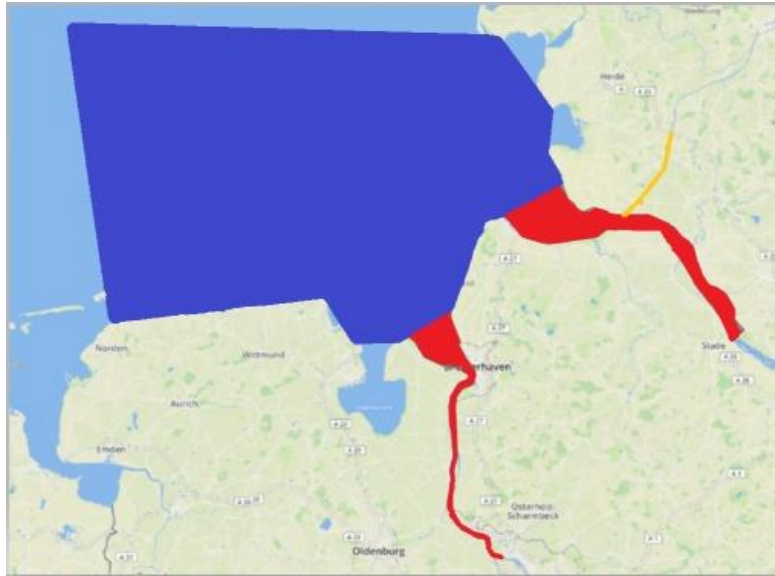


Abbildung 7-27: Betrachtete Gewässerbereiche

Blau eingefärbt ist hierbei der betrachtete Bereich der Nordsee, rot eingefärbt die betrachteten Flussbereiche und orange eingefärbt die betrachteten Kanalbereiche. Bei dem Tageszustand wird zwischen Tag und Nacht unterschieden. Der Wetterzustand teilt sich auf in die Zustände sehr schlecht, schlecht, mittel, gut und sehr gut und berücksichtigt hierbei die aktuelle Wellenhöhe und Windgeschwindigkeit. Nähere Beschreibungen über den Wetterzustand können in Kapitel 7.8 nachgelesen werden. Bei der Sichtweite wird wie bei dem Wetterzustand zwischen den Zuständen sehr schlecht, schlecht, mittel, gut und sehr gut unterschieden. Bei der Windrichtung wird zwischen landeinwärts und -auswärts unterschieden. Wie bei dem Wetterzustand können nähere Beschreibungen für die Sichtweite und die Windrichtung in Kapitel 7.8 nachgelesen werden. Die Manövrierfähigkeit wird in die Klassen 1-7 aufgeteilt und trifft eine Aussage darüber, wie beweglich Schiffe bei Manövern sind und wird betrachtet hierbei die Schiffslänge und den Tiefgang des Schiffs. Nähere Informationen zur Manövrierfähigkeit können Kapitel 7.15 nachgelesen werden. Die Ausprägungen der oben genannten Parameter werden miteinander kombiniert und bilden Klassen, denen die Schiffsbegegnungssituationen zugeordnet werden. Durch die verschiedenen Kombinationen können 42000 Klassen gebildet werden.

Nachdem die extrahierten Begegnungssituationen ihren jeweiligen Klassen zugeordnet wurden, werden die minimalen Distanzen der Begegnungssituationen einer Klasse näher betrachtet. Für alle minimalen Distanzen einer Klasse wird eine Kerndichtschätzung erstellt, um die Wahrscheinlichkeitsdichtefunktion der minimalen Distanzen zu schätzen.

Als kritische Distanz wird diejenige Distanz angesehen, welche nur 10% der Dichten der minimalen Distanzen innerhalb des Integrals der Kerndichtfunktion unterschreiten.

Mithilfe der gebildeten kritischen Distanzen der Klassen wird ein Maß zur Verfügung gestellt, um anomale Begegnungssituationen erkennen zu können. Wenn nun Begegnungssituationen die kritische Distanz ihrer Klasse unterschreiten, ist dies Zeichen für ein höheres Kollisionsrisiko.

Umgesetzt wurde die Klassifikation an ermittelten Begegnungssituationen vom 01.11.2018-15.02.2018. Es wurden nur diejenigen Begegnungssituationen berücksichtigt, welche für alle erforderlichen Parameter Werte hatten. Es wurden dadurch 3207 Klassen gebildet. Jedoch bestehen nur 158 Klassen aus mindestens 100 Szenarien. Die häufigsten Klassen werden in Abbildung 7-28 dargestellt.

Begegnungstyp	Gewässertyp	Wetterzustand	Sichtweite	Windrichtung	Tageszustand	Manövrierfähigkeit Schiff 1	Manövrierfähigkeit Schiff 2	Kritische Distanz	Berücksichtigte Begegnungssituationen
Passing	Nordsee	Gut	Sehr gut	Landauswärts	Nacht	4	5	246	1102
Passing	Nordsee	Gut	Sehr gut	Landauswärts	Tag	4	5	255	988
Passing	Nordsee	Gut	Sehr gut	Landauswärts	Tag	4	6	266	739
Passing	Flussgebiet	Sehr gut	Sehr gut	Landeinwärts	Nacht	4	5	180	718
Passing	Nordsee	Gut	Sehr gut	Landauswärts	Tag	5	6	254	713
Passing	Nordsee	Gut	Sehr gut	Landauswärts	Nacht	5	6	285	691
Passing	Nordsee	Gut	Sehr gut	Landauswärts	Nacht	4	6	255	662
Passing	Flussgebiet	Sehr gut	Sehr gut	Landeinwärts	Tag	4	5	187	617
Passing	Flussgebiet	Sehr gut	Sehr gut	Landeinwärts	Nacht	5	6	154	568
Passing	Kanalgebiet	Sehr gut	Sehr gut	Landeinwärts	Nacht	4	5	18	541

Abbildung 7-28: Die zehn häufigsten Klassen

Auffällig an den zehn häufigsten Klassen ist, dass nur der Begegnungstyp Passing vertreten ist. Dies ist darauf zurückzuführen, dass von den gefilterten

Begegnungssituationen die meisten zum Begegnungstyp Passing zugewiesen werden. Die genaue Verteilung der Begegnungssituationen für die Monate Dezember 2018 und Januar 2019 kann in Kapitel 7.14 betrachtet werden. Ferner ist der Gewässertyp häufig die Nordsee, der Wetterzustand entweder gut oder sehr gut, die Sichtweite immer sehr gut und die Windrichtung immer landeinwärts. Der Tageszustand ist in sechs Klassen Nacht und in den restlichen Tag, somit sehr ausgeglichen verteilt. Die Manövrierfähigkeit der beiden Schiffe ist immer zwischen 4 und 6. Die Verteilung der Gewässertypen und der Manövrierfähigkeiten in den Szenarien kann außerdem 7.14 nachgelesen werden. Die Kerndichtschätzung der Klasse mit den meisten berücksichtigten minimalen Distanzen (1102 berücksichtigte minimale Distanzen, siehe Abbildung 7-28) wird in Abbildung 7-29 dargestellt.

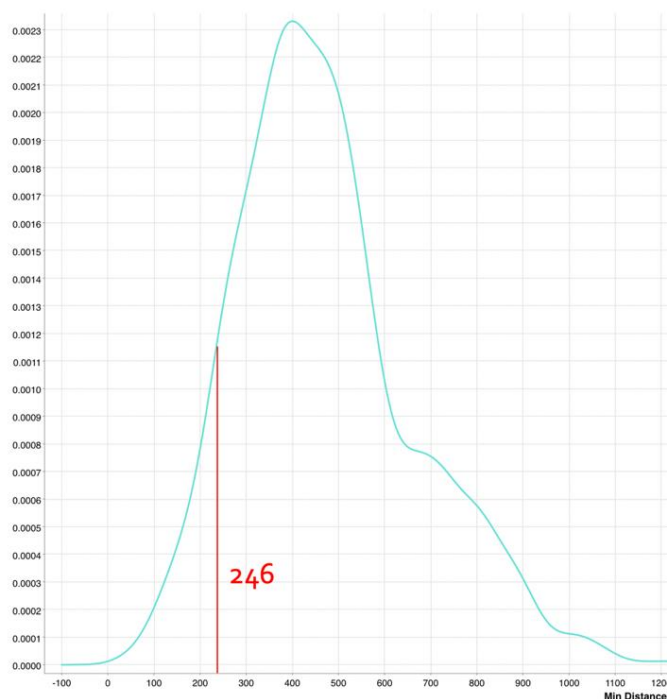


Abbildung 7-29: Kerndichtschätzung der Klasse mit den meisten Begegnungssituationen

Die kritische Distanz ist mit einer roten Linie markiert in Abbildung 7-29, diese beträgt 246 Meter. Alle Begegnungssituationen, welche dieser Klasse angehören und die kritische Distanz unterschreiten, werden demnach als anomal betrachtet.

7.13 Klassifikation von Schiffsbegegnungssituationen zur Erkennung von Parametereinflüssen

Im Folgenden wird exemplarisch der Einfluss der verschiedenen Parameter auf die kritische Distanz der im vorherigen Abschnitt vorgestellten häufigsten Klasse untersucht. Die einzigen beiden Parameter, welche unverändert bleiben sind der Begegnungstyp (Passing) und die Manövrierfähigkeit der beiden Schiffe (4 und 5).

Verändert sich der Parameter Nordsee zu Flussgebiet, so verringert sich die kritische Distanz von 246 Meter auf 186 Meter, die entsprechende Kerndichtschätzung ist in Abbildung 7-30 dargestellt. Für diese Klasse wurden 123 Begegnungssituationen berücksichtigt.

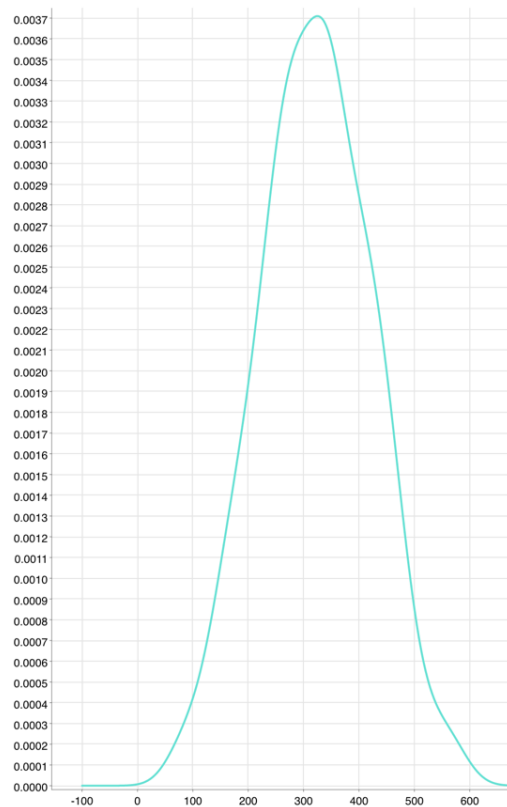


Abbildung 7-30: Kerndichtschätzung häufigste Klasse mit Flussgebiet als Gewässertyp

Neben einer geringeren kritischen Distanz fällt auf, dass der Graph gestauchter ist, das daran liegt kann, dass im Flussgebiet weniger Platz vorhanden als in offenen

Gewässern. Wird nun der Parameter Wetterzustand verändert, fällt auf, dass bei einem besseren Wetterzustand die kritische Distanz sinkt. Somit kommen sich für die Beispielklasse die Schiffe näher bei besseren Wetterbedingungen. Wenn der Wetterzustand sehr gut ist, sinkt die kritische Distanz auf 201 Meter bei 293 berücksichtigten Begegnungssituationen, bei einem Wetterzustand von ausschließlich mittel erhöht sich die kritische Distanz auf 260 Meter bei 143 berücksichtigten Begegnungssituationen. Die Graphen der jeweiligen Kerndichtschätzungen werden in Abbildung 7-31 gezeigt. Abbildung 7-31 a) zeigt die Kerndichtschätzung, wenn der Wetterzustand sehr gut ist und in Abbildung 7-31 b) wird die Kerndichtschätzung dargestellt, wenn der Wetterzustand mittel ist.

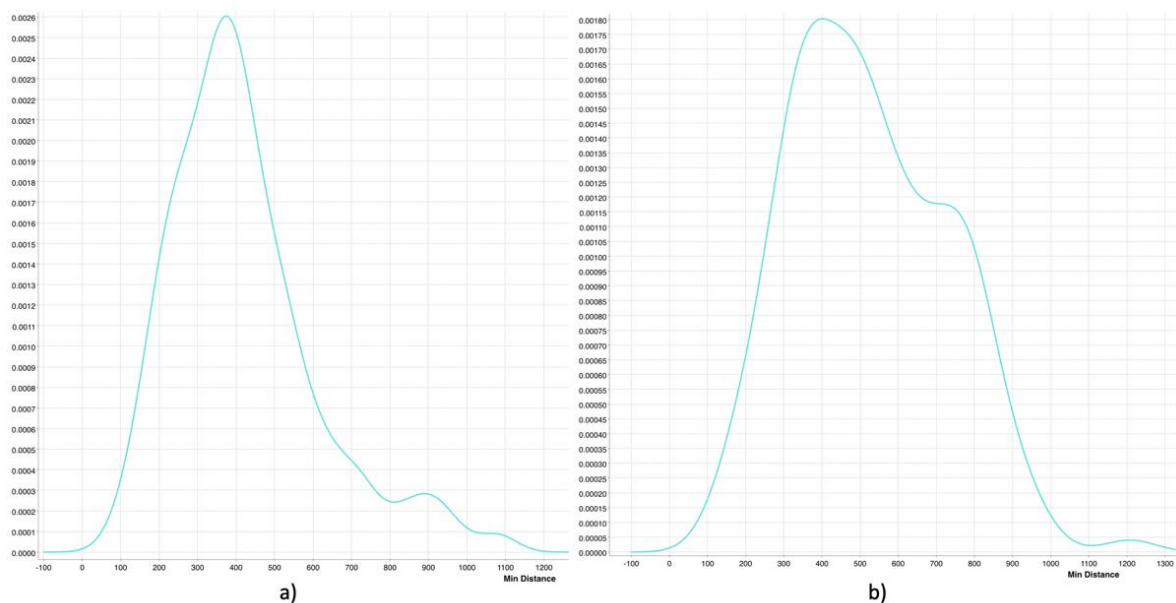


Abbildung 7-31: Kerndichtschätzungen zu veränderten Wetterbedingungen

Über den Einfluss des Parameters Sichtweite kann keine Aussage getroffen werden, da bei einer Verringerung des Werts keine der resultierenden Klassen mehr als 100 berücksichtigte Begegnungssituationen beinhalten. Wird die Windrichtung betrachtet, so fällt auf, dass kritische Distanz etwas höher ist, wenn der Wind landauswärts ist und liegt bei 255 Meter bei 249 berücksichtigten

Begegnungssituationen. Die zugehörige Kerndichtschätzung wird in Abbildung 7-32 dargestellt.

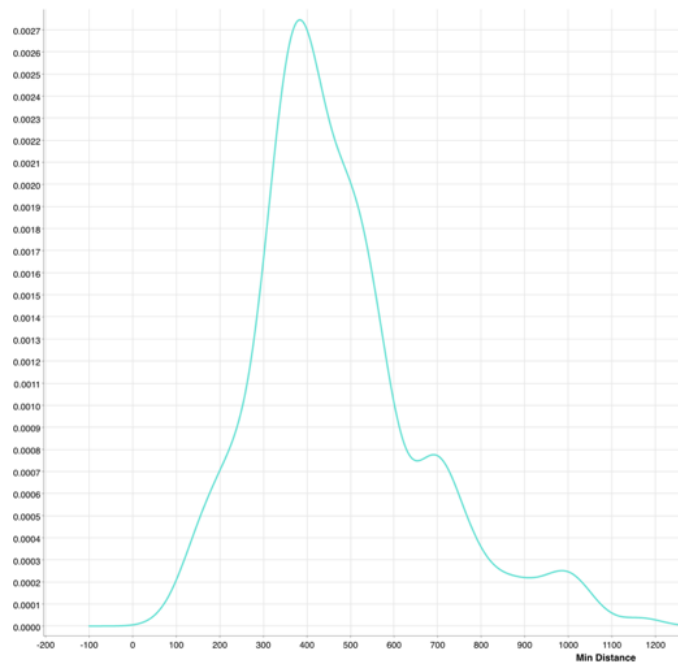


Abbildung 7-32: Kerndichtschätzung zu veränderter Windbedingung

Ändert man den Parameter Tageszeit von Nacht auf Tag, so steigt die kritische Distanz auf 255 Meter bei 988 berücksichtigten minimalen Distanzen. Die zugehörige Kerndichtschätzung wird in Abbildung 7-33 gezeigt.

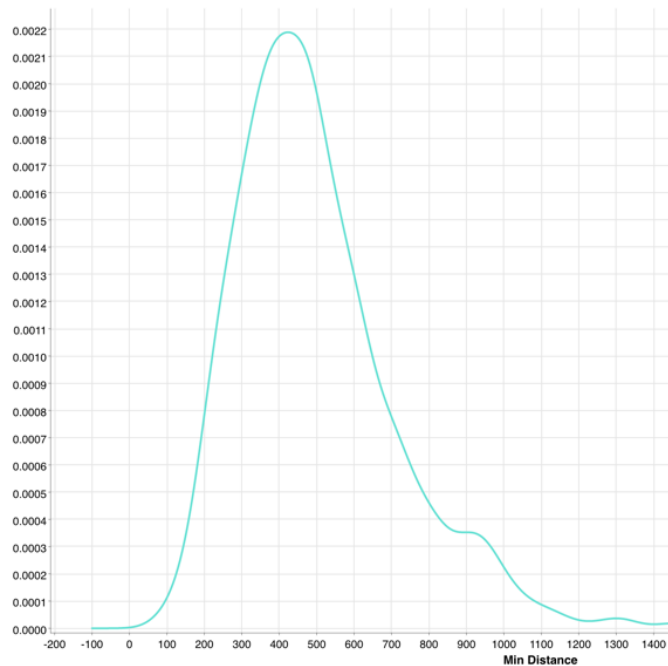


Abbildung 7-33: Kerndichtschätzung bei veränderter Tageszeit

7.14 Statistiken zu den Begegnungssituation

In diesem Abschnitt werden Statistiken zu den extrahierten Begegnungssituationen aus den Monaten Dezember 2018 und Januar 2019 gezeigt.

Die meisten Begegnungssituationen finden demnach im Flussgebiet statt, dicht gefolgt von der Nordsee und weit abgeschlagen kommt das Kanalgebiet. *Undefined* heißt, dass der zum Zeitpunkt des CPA erfasste Bereich nicht abgedeckt wurde durch die erstellten Geometrien für die jeweiligen Bereiche. In Abbildung 7-27 sind die durch die erstellten Geometrien abgedeckten Bereiche dargestellt. Blau eingefärbt ist der abgedeckte Bereich der Nordsee, rot eingefärbt die abgedeckten Flussgebiete und zuletzt orange eingefärbt der abgedeckte Kanalbereich.

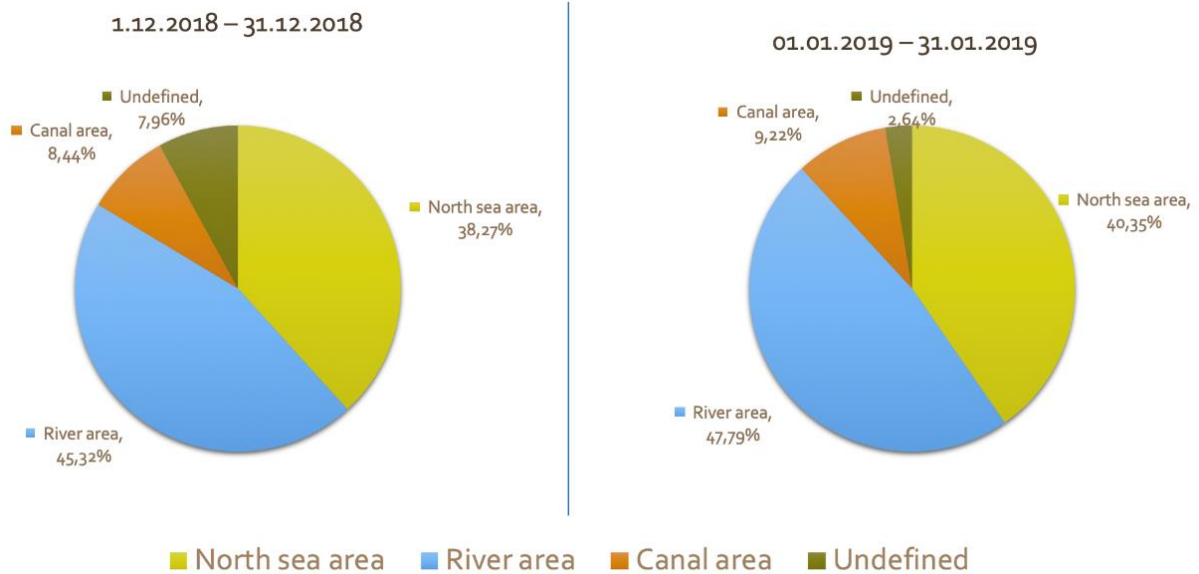


Abbildung 7-34: Verteilung der Gewässerarten

In Abbildung 7-35 wird Verteilung der Begegnungstypen abgebildet. Den größten Anteil hat das *Passing*, welches in beiden betrachteten Zeiträumen größer als 65% ist. Anschließend folgt das *Following* mit 18,5 % im Dezember und 19,9 % im Januar. *Overtaking* nimmt in beiden Monaten ungefähr 13% ein. Den geringsten Anteil hat *Crossing* und *Head-on*.

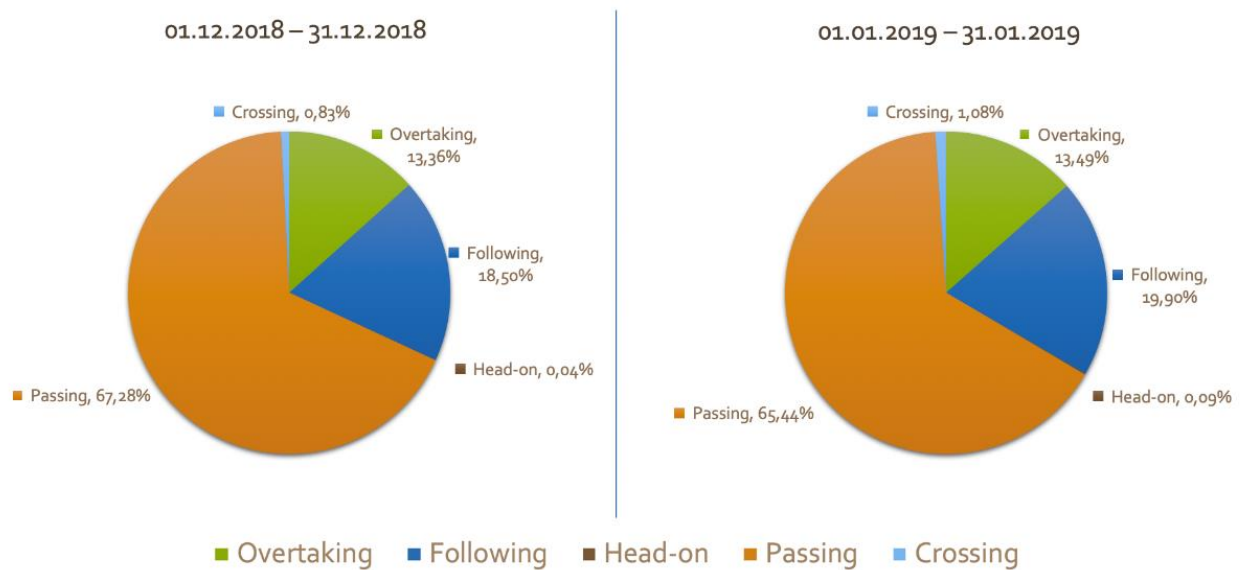


Abbildung 7-35: Verteilung der Begegnungstypen

In Abbildung 7-36 wird die Verteilung der Sensitivitätsklassen dargestellt. Die Klassen 4, 5 und 6 nehmen jeweils mehr als 20% ein in den beiden Monaten, mit fast 30% in beiden Monaten sind Schiffe der Sensitivitätsklasse 5 beteiligt. Die restlichen Klassen sind viel seltener beteiligt. Am meisten sind jedoch Schiffe der Klasse 3 beteiligt, mit jeweils ungefähr 7% in den beiden Monaten.

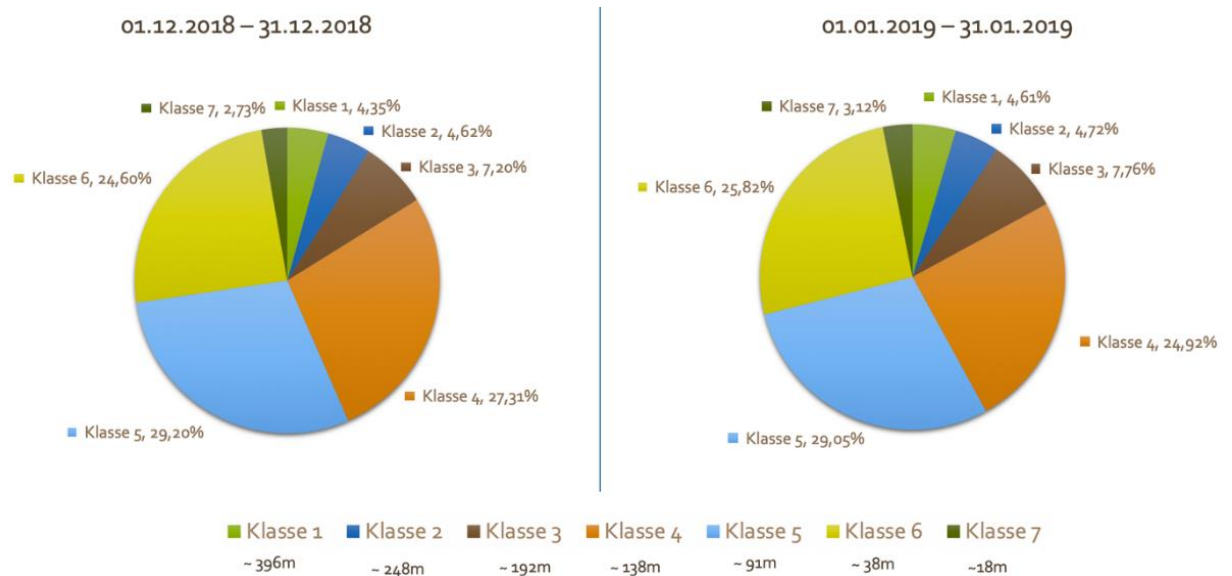


Abbildung 7-36: Verteilung der Sensitivitätsklassen

In Abbildung 7-37 werden die prozentual eingehaltenen Regeln in den beiden Monaten gezeigt. Zu SeeSchStrO 23 iii) können keine Aussagen getroffen werden, da keine Begegnungssituationen, welche als *Overtaking* und *Following* deklariert wurden, in einem Bereich mit Überholverbot stattfanden.

Regel	Constraint	Eingehalten Dezember 2018 (%)	Eingehalten Januar 2019 (%)
KVR 6	Hard	93.304	93.976
KVR 14	Hard	67.68	68.49
SeeSchStrO 23 i)	Soft	65.646	65.513
SeeSchStrO 23 ii)	Soft	60.07	58.277
SeeSchStrO 23 iii)	Hard	-	-

Abbildung 7-37: Eingehaltene Regeln in den Monaten Dezember 2018 und Januar 2019

7.15 Normalmodell zur Repräsentation üblicher Manöver und Tracks

Das Normalmodell zur Repräsentation üblicher Manöver ist ein Modell, welches Regionen beinhaltet an denen Schiffe Manöver vermehrt tätigen (vgl. Kapitel 7.15.2.1). Die Abfolgen in denen Regionen oftmals hintereinander befahren werden, werden durch das Modell ebenfalls dargestellt. (vgl. Kapitel 7.15.2.2) Damit können beliebige *Tracks* hinsichtlich ihrer Normalität untersucht werden. So weisen Abweichungen vom Normalmodell auf Anomalien in den einzelnen *Tracks* hin. (vgl. Kapitel 7.15.3)



Abbildung 7-38: Exemplarisches Manövernetz im Bereich der Elbe

In der Abbildung 7-38 wurde ein exemplarisches Manövernetz für den Bereich der Elbe abgebildet. Dabei stellen die gelben Punkte, die Mittelpunkte der üblichen Regionen dar. Die Verbindungen zwischen den Punkten visualisieren hingegen die Reihenfolge, in der die Manöver häufig hintereinander getätigt werden.

7.15.1 Detektion von Manöverpunkte

Als Manöver werden im Folgenden Punkte verstanden, an denen sich die Ausrichtung des Schiffes signifikant verändert, sodass die Weiterfahrt des Schiffes in eine abweichende Richtung fortgesetzt wird. Als Basis für die Erkennung dieser Punkte wird der *Course-over-Ground*-Wert, oder auch COG-Wert, verwendet.

Bei der Fahrt eines Manövers verändert sich der COG-Wert meistens von einem Datensatz zum anderen nur minimal. Häufig sind Manöver ein schleichender Prozess, bei dem sich die COG-Werte nach und nach kontinuierlich erhöhen oder verringern. Damit also ein Manöver detektiert werden kann, reicht es nicht aus einzelne Datensätze miteinander zu vergleichen. Vielmehr wird ein Verfahren benötigt, welches in der Lage ist, solche langsamen Veränderungen des COG-Wertes zu detektieren. Ein solches Verfahren ist der CuSum Algorithmus.

CuSum ist ein Algorithmus, mit dem geringe Veränderungen von einem definierten Zielwert entdeckt werden können. Dabei werden die Differenzen von dem aktuellen Wert und dem Zielwert über die Zeit aufaddiert. Diese Summe wird auch *cumulative sum*, kurz CuSum, genannt. Überschreitet die Summe einen bestimmten Grenzwert (*action limit*) wird die Abweichung vom Zielwert als relevant angesehen. (McNeese 2014, 1)

Damit CuSum für die Detektion von Manövern genutzt werden kann, müssen einige kleine Anpassungen getroffen werden. So existiert ein fester Zielwert bei der Detektion von Manövern nicht, da die Detektion eines Manövers auf Basis des aktuellen COG-Wertes erfolgt. Dieser ist allerdings für jedes Schiff und für jede Situation unterschiedlich, sodass es keinen festen Zielwert für alle Schiffe geben kann. Da COG-Werte meistens unpräzise sind und Schwankungen aufweisen, wird hier ein gleitender Mittelwert der letzten COG-Werte als Zielwert definiert.

```
def calculateTargetValue(k: slack, H: action_limit, ml: maneuverlist):
    for dataRow in data: #
        t = calcCurrentTarget() # Berechnet aktuellen Zielwert
        cs = calcCusum(t, k) # Berechnet neue kummulierte Summe

        if cs > H:
            ml.append(dataRow) # Datensatz d ist Manöver m
    return ml
```

Abbildung 7-39: Pseudocode vom CuSum Algorithmus

Abbildung 7-39 skizziert die grundlegende Funktionsweise des abgewandelten CuSum Algorithmus. Zu Beginn müssen zwei Grenzwerte festgelegt werden. Zum einen ist das der *Slack*, der die erlaubte Schwankung vom Zielwert angibt.

Zum anderen ist es das *Action Limit*, der welches definiert ab welchem Betrag der kumulierten Summe die Abweichungen als ein Manöver interpretiert werden. In der Literatur wird für den *Slack* ein Wert in Höhe der einfachen Standardabweichung und für den das Action Limit, ein Wert in Höhe der vierfachen Standardabweichung empfohlen (vgl. McNeese 2014, 3).

Für die Detektion von Manöverpunkten haben sich die Richtwerte allerdings als zu sensibel herausgestellt, sodass hierfür eigene empirische Werte verwendet werden. Es hat sich herausgestellt, dass die Stärke der Manöver in der Regel von der Größe und dem Tiefgang der Schiffe abhängig ist. (vgl. MARITIME SAFETY COMMITTEE 2002, 3 ff.) So können sich Schiffe der Schiffsklasse 6 wesentlich flexibler fortbewegen, als Schiffe aus der Schiffsklasse 1 (vgl. Kapitel Schiffsklassen). Dies spiegelt sich auch in den Manövern selbst wider. Große Schiffe können keine sonderlich scharfen Manöver fahren, bei denen sie ihre Ausrichtung in kurzer Zeit stark verändern. Kleinere Schiffe sind hingegen durchaus dazu in der Lage. Um durch den CuSum Algorithmus also die charakteristischen Manöverpunkte für jedes Schiff zu extrahieren, erfolgte die Parametrisierung der CuSum Parameter für jedes Schiff individuell auf Basis der Manövrierbarkeit. Hierzu wurde ein minimaler (für sehr große Schiffe) und ein maximaler Wert (für sehr kleine Schiffe) festgesetzt. Je nach Manövrierfähigkeit (vgl. Kapitel 7.10) des einzelnen Schiffes, wird linear ein Wert zwischen den beiden Grenzwerten für die Manöverdetektion verwendet. Je nach Wahl des *Slacks* und des Action Limits kann reguliert werden, ob jegliche Abweichungen vom aktuellen Kurs oder nur besonders starke Abweichungen als Manöver interpretiert werden sollen.

	Sensibel	Normal
Slack	0.5 – 2.0	2.0 – 4.0
Action Limit	3.0 – 8.0	15.0 – 30.0

Tabelle 7-15: Mögliche Parametrisierung von CuSum

Tabelle 7-15 zeigt zwei mögliche Parametrisierungen des CuSum Algorithmus auf. Wenn möglichst viele und auch leichte Abweichungen vom Kurs identifiziert werden sollen, empfiehlt es sich sowohl für den *Slack* als auch für das Action Limit eher geringe Werte zu verwenden. Sollen hingegen nur die prägnantesten Manöver einer Route detektiert werden, hat sich die zweite Variante „*Normal*“ sehr bewährt. Aufgrund des höheren *Slacks* werden viele Abweichungen gar nicht in die Berechnung der kumulierten Summe aufgenommen. Außerdem wird eine Ausrichtungsveränderung aufgrund der höheren Action Limits erst wesentlich später als ein Manöver erkannt. Sollte die vorgeschlagene Parametrisierung für einen Anwendungsfall nicht die erhofften Manöver detektieren, kann durch die Variation der Parameter dies erreicht werden.

Nachdem die beiden Parameter festgelegt und der Zielwert als gleitender Mittelwert der letzten zehn COG-Werte berechnet wurde, wird die kumulierte Summe berechnet. Dabei kann CuSum sowohl positive als auch negative Veränderungen vom Zielwert detektieren. Hierzu gibt es dementsprechend auch zwei verschiedene kumulierte Summen. Eine positive Abweichung vom Zielwert lässt sich hier als ein steuerbordseitiges und eine negative Veränderung als ein backbordseitiges Manöver interpretieren.

Die kumulierten Summen lassen sich, wie in Abbildung 7-40 und Abbildung 7-41 darstellt berechnen. Dabei müssen:

- X_i = COG-Wert an der i -ten Stelle
- t = Gleitender Mittelwert, Zielwert
- k = *Slack*

gesetzt werden.

$$SH(i) = \text{Max} [0, SH(i - 1) + X_i - t - k]$$

Abbildung 7-40: Detektion von steuerbordseitigen Manövern

$$SL(i) = \text{Min} [0, SL(i - 1) + X_i - t + k]$$

Abbildung 7-41: Detektion von backbordseitigen Manövern

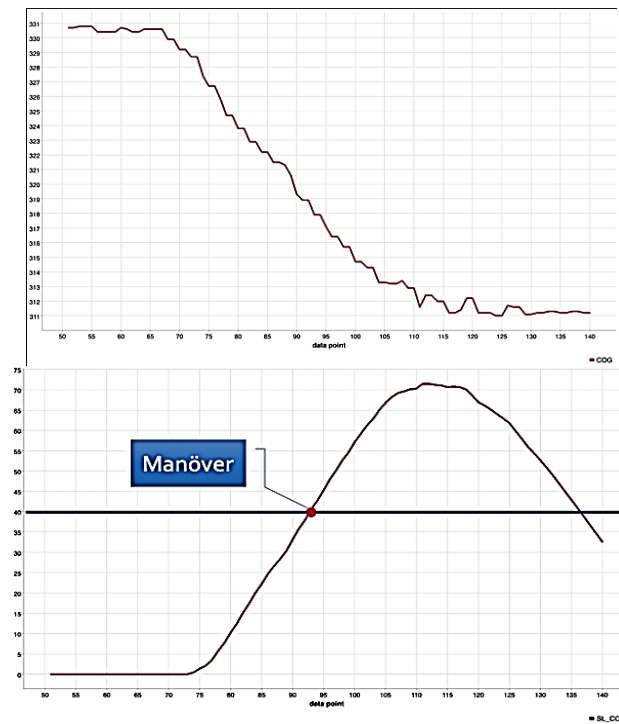


Abbildung 7-42: Exemplarischer COG-Graph mit dazugehörigem CuSum Graph

In der Abbildung 7-42 ist die Detektion eines Manövers dargestellt. Im linken Graphen kann der Abfall des COG-Wertes beobachtet werden, was auf ein backbordseitiges Manöver hinweist. Anfangs bleibt der COG-Wert größtenteils stabil, weshalb der CuSum Wert nicht ausschlägt. Ab dem siebzigsten Datensatz lässt sich eine kontinuierliche Veränderung der Ausrichtung erkennen. Mit der Formel zur Detektion von backbordseitigen Manövern ergibt sich durch die Eingabe der COG Zeitreihe ein Ausschlag, der sich zunächst unter dem Action Limit Niveau befindet. Dies deutet zunächst auf eine Veränderung der Ausrichtung hin, die aber noch nicht stark genug ist, um als ein Manöver zu gelten. Erst bei einem weiteren Abfall des COG-Wertes und damit Erhöhung der kumulierten Summe, wird ein Manöver detektiert, sobald der CuSum-Graph das Action Limit überschreitet.



Abbildung 7-43: CuSum Algorithmus angewendet an einem Track auf der Elbe

Abbildung 7-43 zeigt die Manöver (rote Punkte), die von dem CuSum Algorithmus auf einer Route eines Schiffes auf der Elbe detektiert wurden. CuSum detektiert dabei zuverlässig die charakteristischen Punkte dieser Route an denen sich die Ausrichtung des Schiffes verändert hat.

7.15.2 Erstellung eines Manövernetzes

Wie in Kapitel 7.15 beschrieben können nun die Manöverpunkte eines *Tracks* gefunden werden. Das Normalmodell soll allerdings jegliche Stellen, an denen Schiffe oftmals Manöver fahren, repräsentieren, sodass idealerweise mindestens 1000 *Tracks* zur Erstellung des Netzes einfließen sollten. Die Masse an sich daraus ergebenden Manövern muss nun auf Stellen untersucht werden, an denen Manöver häufiger gefahren werden, als an anderen Stellen.

7.15.2.1 Clustering der Manöverpunkte

Die Untersuchung auf Stellen, an denen häufiger Manöver gefahren wurden, wurde mit einem DBSCAN Clustering durchgeführt. Da das DBSCAN Clusterverfahren dichte basiert, erhält man nach dem Clustering die Stellen, an denen die meisten Manöver gefahren wurden. Stellen an denen Schiffe nur sporadisch ihre Ausrichtung veränderten, fallen als Rauschen weg.

Als eine Problematik hat sich allerdings dabei das unterschiedlich starke Verkehrsaufkommen an unterschiedlichen Orten erwiesen. So gab es in den vorliegenden Daten wesentlich weniger *Tracks* im Bereich der Elbe, als auf der Nordsee, sodass folglich auch absolut gesehen mehr Manöver auf der Nordsee detektiert wurden, als auf der Elbe. Sollen nun die Stellen extrahiert werden, bei denen es am häufigsten zu Manövern kommt, führt dieses Ungleichgewicht innerhalb der Dichte von Manövern zu unerwünschten Ergebnissen. Werden die DBSCAN Parameter *Epsilon* und *MinPoints* für das Clustering locker eingestellt (großes *Epsilon*, geringe *MinPoints*), werden die Cluster auf der Elbe zwar sehr gut extrahiert, im Bereich der Nordsee hingegen führt dies zur Detektion von sehr vielen Clustern die häufig keine besondere Relevanz besitzen. Wird das Clustering hingegen etwas strenger eingestellt (kleines *Epsilon*, hohe *MinPoints*) kommt es auf offener See zu guten Ergebnissen. Im Bereich der Elbe werden aufgrund der zu hohen *MinPoints* allerdings gar keine Cluster mehr gebildet. Es hat sich daher bewährt zwischen diesen beiden Bereichen zu differenzieren und zweimal separat zu Clustern, um so die beste Parametrisierung basierend auf der vorhandenen Datenmenge nutzen zu können. Eine mögliche Alternative dazu wäre es sicherzustellen, dass in beiden Bereichen etwa gleichviele *Tracks* betrachtet werden, so könnte für beide Bereiche dieselbe Parametrisierung verwendet werden. Basierend auf 15.000 Manöverpunkten hat sich ein *Epsilon* von 0.004 – 0.006 und *MinPoints* von 25 bewährt. Bei einer steigenden Anzahl an Manöverpunkten bietet es sich an das *Epsilon* weiter zu reduzieren, da sich mit einer steigenden Anzahl an Manöverpunkten auch der Abstand zwischen den Manöverpunkten weiter reduziert.

Beim Clustering sollte zudem auf die Menge des Rauschens geachtet werden. Werden bei dem Clustering prozentual viele Manöverpunkte als Rauschen herausgefiltert, bildet das tatsächliche Normalmodell Stellen, an denen vergleichsweise wenige Manöverpunkte gefahren wurden, nicht ab. Dies führt dazu, dass bei der Anwendung des Manövernetzes mehr Manöver als anomal detektiert, als bei einem Modell welches nur wenige Punkte als Rauschen identifiziert hat. Um somit bei der Analyse nach anomalen Manövern nur sehr

unübliche Manöver zu erhalten, sollte darauf geachtet werden, dass das Modell beim Clustering ein nicht zu hohes Rauschen aufweist.

7.15.2.2 Verbindung extrahierter Manöverregionen

Wurden die einzelnen Stellen, an denen Schiffe zahlreich Manöver fahren, extrahiert, werden anschließend die Abhängigkeiten zwischen den einzelnen Clustern analysiert. Die Manöver, die durch den Clustervorgang noch nicht als Rauschen herausgefiltert wurden, können nun eindeutig einem Cluster zugewiesen werden. Dadurch kann *Track* für *Track* durchgegangen und ausgewertet werden in welcher Reihenfolge die einzelnen Cluster normalerweise besucht werden. Hat ein Schiff beispielsweise zunächst ein Manöver im Cluster 1 ausgeführt und als nächstes im Cluster 4 wird diese Verbindung gespeichert. Außerdem existiert für jede Verbindung ein Zähler, wie häufig die einzelne Verbindung bereits befahren wurde.

7.15.3 Nutzung eines Manövernetzes

Das Manövernetz verfügt über zwei verschiedene Anwendungsmöglichkeiten. Zum einen kann das Manövernetz dazu verwendet werden grundlegend *Tracks* herauszufiltern in denen anomale Manöver gefahren worden sind. Zum anderen können *Tracks* extrahiert werden, in denen Schiffe *Tracks* wählen, die andere Schiffe üblicherweise nicht abgefahren sind. Beide Möglichkeiten dienen somit dazu risikoerhöhte *Tracks* zu finden, in denen sich Schiffe nicht an das erhobene Normalmodell gehalten haben.

Die Manöverpunkte, die nach dem Clustering aus dem Kapitel 7.15.2.1 noch übrigbleiben, bilden die Grundlage für die Einschätzung von Manövern. Soll überprüft werden, ob es sich bei einem neuen Manöver, um ein übliches oder unübliches Manöver handelt, kann dies auf Basis der Cluster und der Position des zu analysierenden Manövers geschehen. Fällt ein Manöver direkt in ein Cluster, kann es als ein übliches Manöver interpretiert werden, da dort zuvor andere Schiffe bereits auch häufig ihre Ausrichtung verändert haben. Fällt ein Manöver hingegen in keines der gebildeten Cluster, scheint es ein Manöver zu sein, was

selten oder gar nicht so gefahren wird, sodass es als ein anomales Manöver betrachtet werden kann. Um zu analysieren, ob ein Manöver in eines der Cluster fällt oder nicht, wird für jedes Cluster eine zweidimensionale Kerndichteschätzung berechnet.

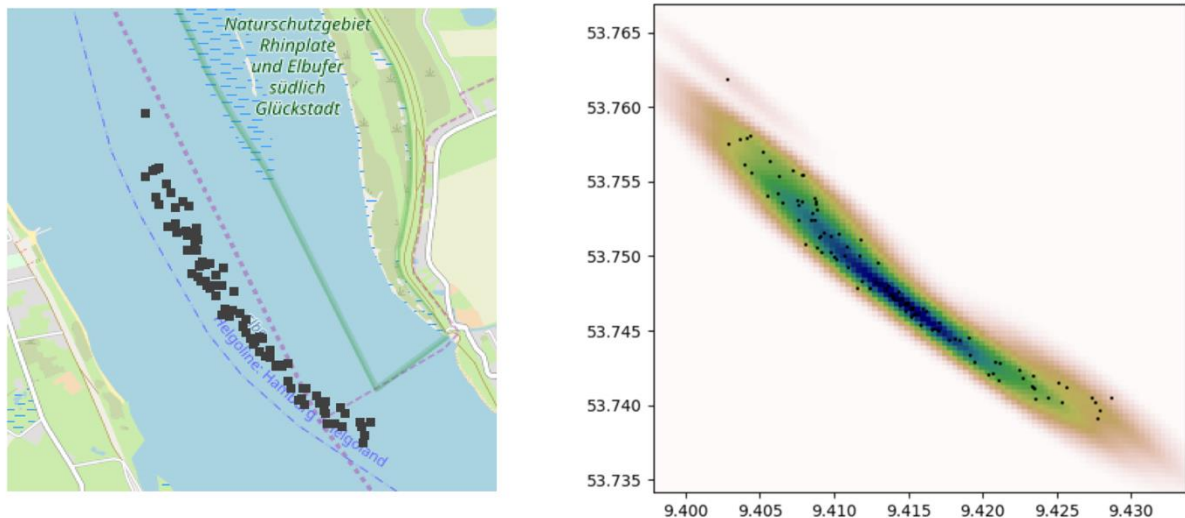


Abbildung 7-44: exemplarisches Cluster mit der dazugehörigen zweidimensionalen Kerndichteschätzung

Die zweidimensionale Kerndichteschätzung spiegelt die Form und die Position des jeweiligen Clusters sehr gut wider (vgl. Abbildung 7-44). Soll nun ein Manöver auf Anomalität untersucht werden, wird die Dichte für jedes Cluster von der Position des Manövers berechnet. Liefert eine der zweidimensionalen Kerndichteschätzungen für das Manöver einen Wert von über 0.00 als Dichte, scheint das Manöver in einem Bereich gefahren zu sein, welches durch ein Cluster repräsentiert worden ist. Somit handelt es sich bei dem gefahrenen Manöver, um ein übliches Manöver. Liefert keine der Kerndichteschätzungen einen Wert über 0.00, wurde das Manöver an einem Ort gefahren, welches von keinem der Cluster repräsentiert wird. Dadurch kann es als ein anomales Manöver interpretiert werden. Bei üblichen Manövern kann die Höhe des Dichtewerts Aussagen darüber treffen, wie üblich das gefahrene Manöver ist. Stellen, an denen sehr viele Manöver innerhalb eines Clusters gefahren wurden, liefern ebenfalls einen

höheren Dichtewert zurück. Regionen, an denen wenige Manöver gefahren wurden, aber dennoch zum Cluster dazugehören (häufig Randregionen) liefern eher eine geringe Dichte zurück.

Neben dem Analysieren von einzelnen Manövern ist es ebenfalls möglich die Route, die ein Schiff gefahren ist, als Ganzes zu untersuchen. Dafür werden für jede Route zunächst alle Manöverpunkte ermittelt. Diese Manöverpunkte lassen sich dann, wie im oben beschrieben, mit der Kerndichteschätzung eindeutig einem Cluster zuordnen, sofern es sich nicht um ein anomales Manöver handelt. Somit ist bekannt in welcher Reihenfolge das Schiff die einzelnen Cluster besucht hat. Da im Normalmodell die Verbindungen zwischen Clustern enthalten sind, die häufiger befahren wurden. Kann nun die Reihenfolge der Verbindungen der zu überprüfenden Route mit den Verbindungen aus dem Normalmodell verglichen werden. Sind alle Verbindungen aus der Route im Normalmodell vertreten, so wird die Route als eine übliche Route eingeschätzt. Sind hingegen Verbindungen enthalten, die das Normalmodell nicht enthält, deutet dies auf einen *Track* hin, die so üblicherweise nicht gefahren wurde, und somit als ein anomaler *Track* deklariert wird.

7.16 Heatmaps

Zur Visualisierung der Durchschnittsgeschwindigkeit, Positionsdaten und Szenarien in bestimmten Bereichen werden *Heatmaps* zur Verfügung gestellt. Die *Heatmap* für die Durchschnittsgeschwindigkeit ist ein Gitter bestehend aus Zellen, welche eine Länge und Breite von 400 Metern haben.

Eine Länge und Breite von 400 Metern wurde ausgewählt, da ebenfalls ein Gitter zur Erstellung des Route-Topology-Models für die Zählung von Positionsdaten genutzt wurde und sich diese Größe geeignet hat (vgl. Oltmann 2015). In Abbildung 7-45 wird eine *Heatmap* der Durchschnittsgeschwindigkeiten basierend auf 500000 Datensätzen gezeigt. Die Datensätze stammen aus dem Mai 2016. Diese *Heatmap* wurde mittels eines eigens geschriebenen KNIME-Knoten entwickelt.

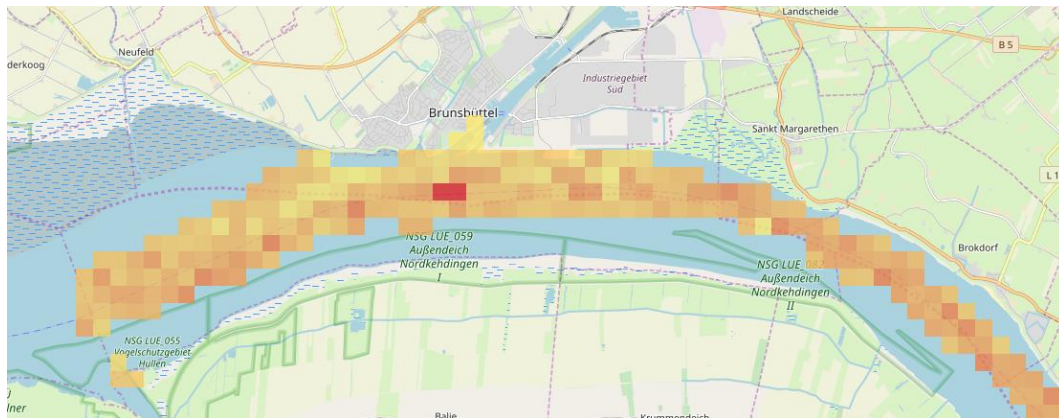


Abbildung 7-45: Heatmap der Durchschnittsgeschwindigkeiten mit 500000 Datensätzen

Je roter eine Zelle ist, desto höher ist die Durchschnittsgeschwindigkeit für eine Zelle. In den beiden roten Zellen wurde eine Durchschnittsgeschwindigkeit von 17,33 m/s aufgezeichnet. Neben der Visualisierung kann mithilfe dieser *Heatmap* in Zukunft die aktuelle Geschwindigkeit eines Schiffes mit der durchschnittlichen Geschwindigkeit für denjenigen Bereich verglichen werden, um zu überprüfen, ob die Geschwindigkeitswerte des betrachteten Schiffes anomal sind oder nicht. Dies muss jedoch noch umgesetzt werden. Die *Heatmap* für die Visualisierung der Positionsdaten wird in Abbildung 7-46 dargestellt. Die Datensätze stammen ebenfalls vom Mai 2016. Für die *Heatmap* der Positionsdaten wurde die Google Maps API genutzt.

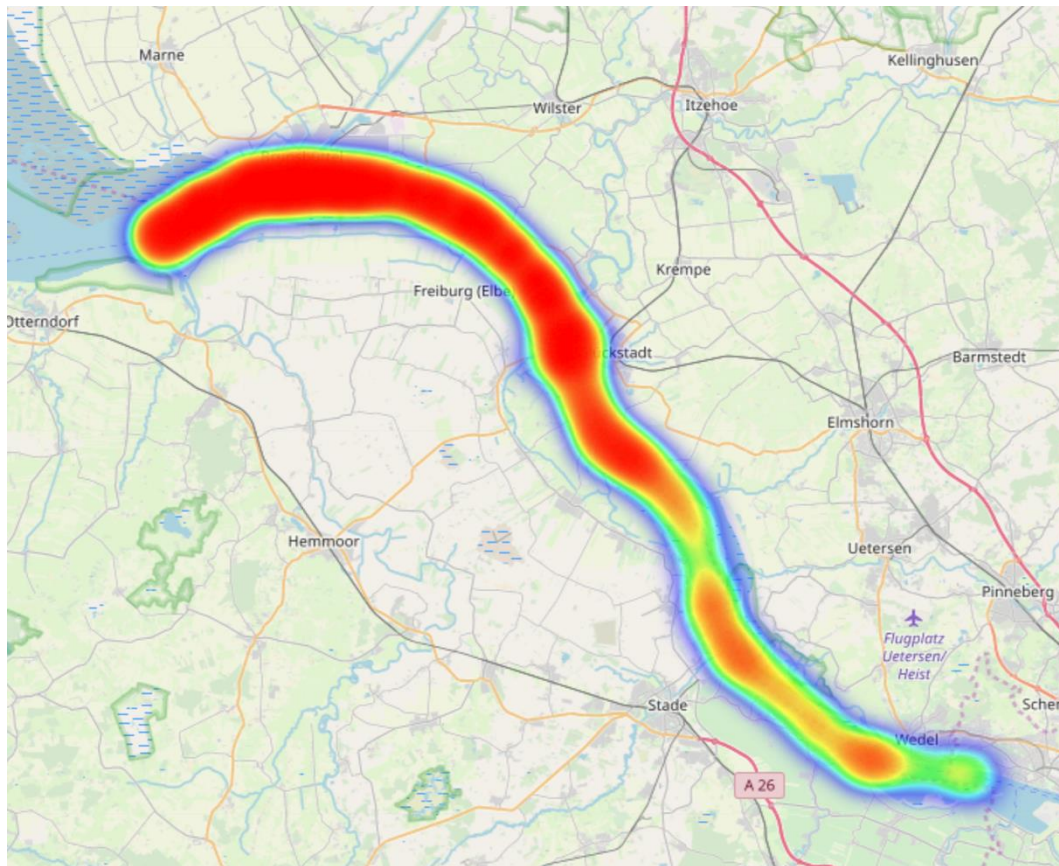


Abbildung 7-46: Heatmap der Positionsdaten mit 500000 Datensätzen

Zuletzt wird in Abbildung 7-47 eine *Heatmap* der Szenarien dargestellt. Hierbei wird die Position des CPA berücksichtigt bei den Szenarien. Es wurden hierfür Begegnungssituationen vom 01.03.2018 bis 31.03.2018 berücksichtigt, dies waren 24117. Für die *Heatmap* der Szenarien wurde Folium genutzt.

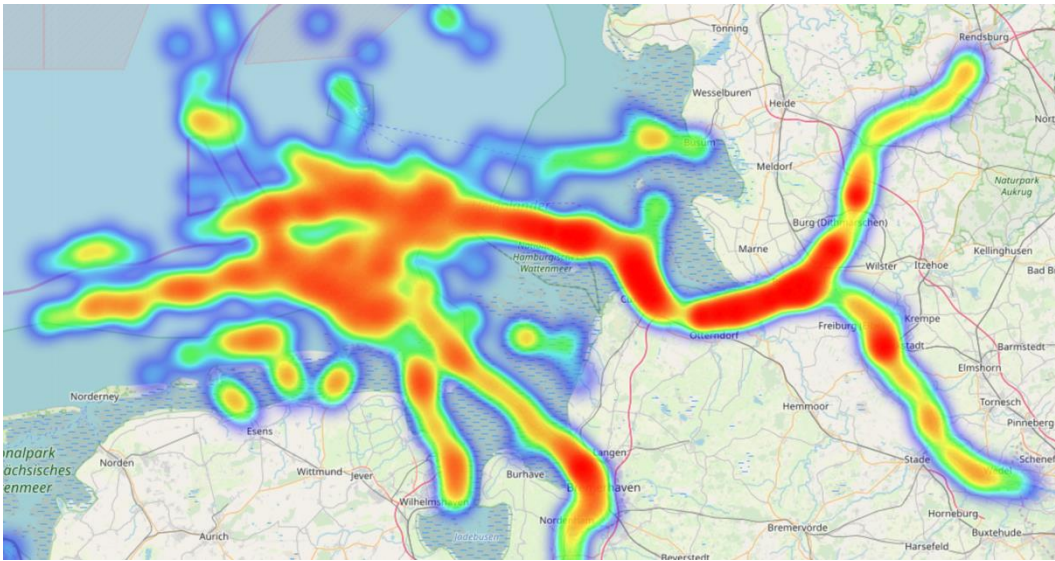


Abbildung 7-47: Heatmap der Szenarien

Weitere Heatmaps können im (Anhang 10.3) betrachtet werden.

7.17 Verkehrsaufkommen

Das Verkehrsaufkommen ist für die Projektgruppe MaGellan ein Maß, welches aussagt wie viele Schiffe sich zu einem Zeitpunkt durchschnittlich um ein Schiff befinden. Verwendet werden kann der Wert als ein weiterer Einflussfaktor für die Risikoeinschätzung von Begegnungssituationen. Je höher das Verkehrsaufkommen bei einer Begegnungssituation ausfällt, desto höher ist das Risiko einer Situation einzuschätzen. Dabei kann für jeden Track die Anzahl an Schiffe innerhalb einer zuvor bestimmten Bounding Box ermittelt werden. Berechnet werden kann das Verkehrsaufkommen mit dem Workflow „Traffic Volume“.

7.18 Bayessches Netz

Das Bayessche Netz ist eine in der Informatik sehr vielseitig einsetzbare Struktur, die aber auch außerhalb, Anwendung findet. Die Struktur der Netze dient zur Speicherung und Verarbeitung unsicheren Wissens vgl. (Wagner 2000, 1). Anwendung findet die Struktur des Bayesschen Netzes z.B. bei Ärzten, Piloten oder bei Finanzberatern. Experten der verschiedensten Bereiche treffen

Entscheidungen aufgrund von guten oder schlechten Resultaten aus vorherigen Aufgaben oder Experimenten. Zudem muss häufiger eine Entscheidung aufgrund von unvollständigen oder sich widersprechenden Informationen getroffen werden vgl.(Wagner 2000). Wagner nach zu urteilen, gilt folgendes:

„In einem solchen Fall, sollte entsprechend den bisherigen Erkenntnissen, die wahrscheinlich beste Entscheidung getroffen werden, in der Regel diejenige, die das Risiko minimiert.“

(Wagner 2000, 1)

Beobachtungen haben jedoch gezeigt, dass Experten in vielen Fällen anders entscheiden, daher wird angestrebt, dass es sogenannte Expertensysteme geben muss, die die Experten bei ihrer Arbeit durch Vorschläge unterstützen vgl. (Wagner 2000). Bayessche Netze sind neben der Einschätzung von Wahrscheinlichkeiten auch in der Lage solche Expertensysteme zu realisieren.

Der Einsatz von Bayesschen Netzen im maritimen Umfeld, stützt sich auf die Erkenntnisse des Papers „Influence of variable on ship collision probability in a Bayesian belief network model“. (Hänninen und Kujala 2012) In diesem Kapitel der Dokumentation, wird das Bayessche Netz für die Erkenntnisgewinnung über die Zusammenhänge der verschiedenen Einflussfaktoren auf die Begegnungssituationen zweier Schiffe beschrieben. Das Netz wird unter Berücksichtigung der folgenden Einflussfaktoren einer Schiffsbegegnung erstellt.

Attribut	Übersetzung	Datentyp
min_distance	Minimale Distanz	Number (double)
is_daytime	Tageszeit	Boolean value
sensitivity_track 1	Sensitivität Track 1	Number (integer)

sensitivity_track 2	Sensitivität Track 2	Number (integer)
below_critical_distance	Unterschreiten der kritischen Distanz	Boolean value
rule_6_satisfied_track_1	Hält sich Track 1 an Regel 6	Boolean value
rule_14_satisfied_track_1	Hält sich Track 1 an Regel 14	Boolean value
rule_1_satisfied_track_1	Hält sich Track 1 an Regel 1	Boolean value
rule_6_satisfied_track_2	Hält sich Track 2 an Regel 6	Boolean value
rule_14_satisfied_track_2	Hält sich Track 2 an Regel 14	Boolean value
rule_1_satisfied_track_2	Hält sich Track 2 an Regel 1	Boolean value

Tabelle 7-16: Kriterien für das Bayessche Netz

Das Bayessche Netz dient zur Einschätzung der Abhängigkeit der verschiedenen Einflussfaktoren unter Berücksichtigung der Unterschreitung der kritischen Distanz zwischen zwei Schiffen. Betrachtet wird hier die Abhängigkeit der Einflussfaktoren untereinander. Faktoren wie die Einhaltung der Regeln oder die Sensitivität der Schiffe spielen hier eine wichtige Rolle. Die betrachteten Szenarien berücksichtigen die Einhaltung einiger Regeln (Kollisionsverhütungsregel 1, 6 und 14) sowie die Betrachtung der Tageszeit (Tag oder Nacht) und die Berücksichtigung der Sensitivitätsklassen der Schiffe.

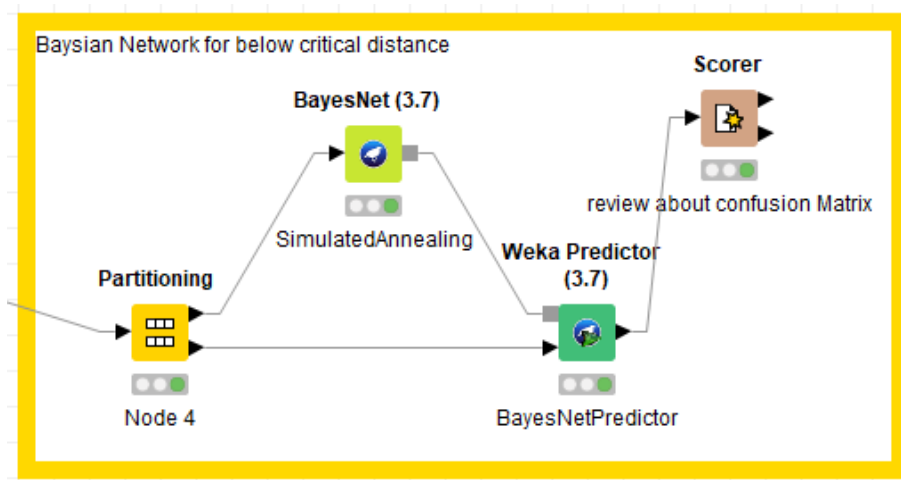


Abbildung 7-48: Bayessches Netz

Abbildung 7-48 stellt das Bayessche Netz in Abhängigkeit der Zielvariablen „below critical distanz“ dar. Der Partitionierungs-Knoten teilt die 25.310 erkannten Szenarien in 20 Prozent Lern- und 80 Prozent Trainingsdaten auf, um so ein selbst lernendes Bayessches Netz zu erstellen, welches die Szenarien im Nachhinein selbst klassifiziert und zwar unter Berücksichtigung der Unterschreitung der kritischen Distanz. Der Learner „BayesNet(3.7)“ des Netzes verwendet den SimulatedAnnealing-Algorithmus zur Analyse der Abhängigkeiten bei der Unterschreitung der kritischen Distanz, in Verbindung mit den Verstößen gegen die Regel 1, 6 oder 14 und der Sensitivität der Schiffe. Zum Testen wurde auch der TabuSearch-Algorithmus verwendet, welcher von den Ergebnissen her auf einer Ebene mit den des SimulatedAnnealing-Algorithmus ist. Allerdings wurde sich im Laufe der Projektphase gegen den TabuSearch entschieden, da dieser nicht so tiefgehend analysiert wie der SimulatedAnnealing. Über den Scorer lässt sich nach erfolgreichem Durchlaufen des Bayesschen Netzes und anhand der *confusion matrix* überprüfen, inwieweit die Trainingsdaten korrekt klassifiziert werden. Gegenübergestellt werden in der confusion matrix in diesem Fall die Unterschreitung der kritischen Distanz „below_critical_distance“ mit der Prediction (below_critical_distance). Unter Verwendung des oben beschriebenen SimulatedAnnealing-Algorithmus zur Klassifizierung der Szenarien, werden 94,89 Prozent der Situationen korrekt klassifiziert, siehe hierzu Abbildung 7-49.

Confusion Matrix - 0:770 - Scorer (review about confusion Matrix)		
File	Hilite	
below_criti...	0	1
0	18053	422
1	611	1162

Correct classified: 19.215 Wrong classified: 1.033
 Accuracy: 94,898 % Error: 5,102 %
 Cohen's kappa (κ) 0,665

Abbildung 7-49: Confusion Matrix

Der Graph des Bayesschen Netzes ist der Abbildung 7-50 zu entnehmen. Je nach verwendetem Klassifizierungs-Algorithmus ändert sich der Graph des Netzes etwas. Dies hängt mit der Suchtiefe des verwendeten Algorithmus zusammen. Anhand der einzelnen Kinder-Knoten des Graphen, lässt sich die Abhängigkeit des jeweiligen Kinder-Knoten in Bezug zum Zielknoten („Vaterknoten“) „below_critical_distance“ auswerten.

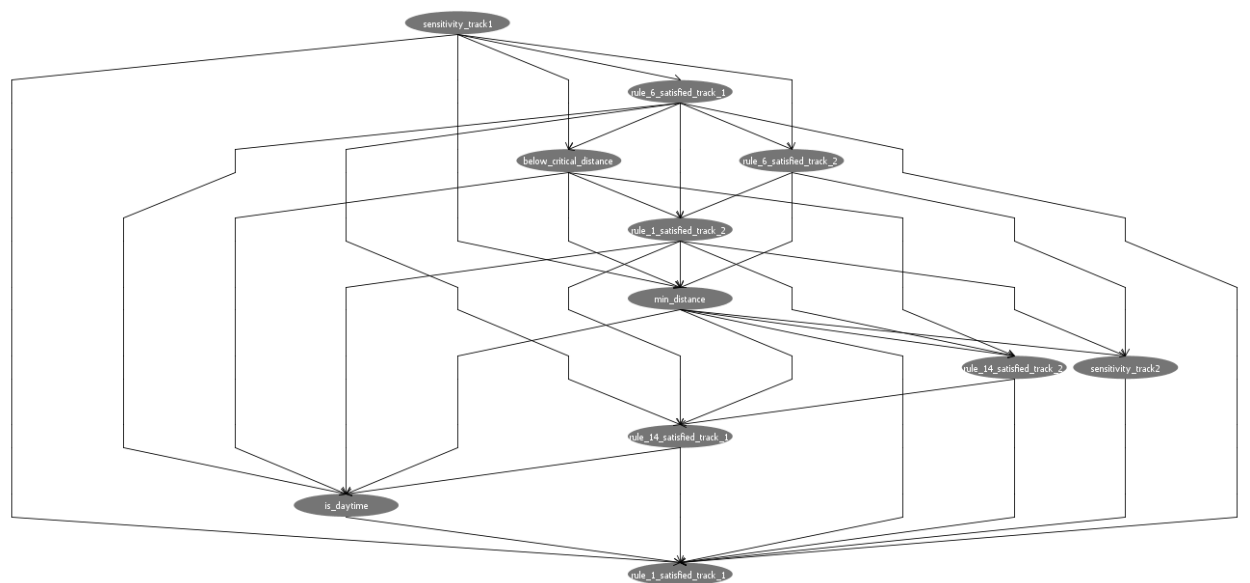


Abbildung 7-50: Graph des Bayesschen Netzes

Die Auswertung des Graphen zeigt, die Wahrscheinlichkeitsverteilungstabellen in Abhängigkeit der verschiedenen Einflussfaktoren wie Unterschreitung der kritischen Distanz oder der Missachtung der KVR in der maritimen Umgebung.

Die Abbildung 7-51 zeigt als Beispiel die Wahrscheinlichkeitsverteilung für die Einhaltung bzw. Missachtung von KVR 1 unter Betrachtung der kritischen Distanz, Einhaltung bzw. Missachtung von KVR 14 sowie die Einhaltung bzw. Missachtung von KVR 1 des zweiten Tracks.

Probability Distribution Tabel for rule_1_satisfied_track_1			0	1
rule_14_satisfied_track_1	belwo_critical_distance	rule_1_satisfied_track_2		
0	0	0	30,3	69,7
0	0	1	26,5	73,5
0	1	0	18,5	81,5
0	1	1	22,6	77,4
1	0	0	66,7	33,3
1	0	1	62,6	37,4
1	1	0	46,1	53,9
1	1	1	43,5	56,5

Abbildung 7-51: Eintrittswahrscheinlichkeitstabelle für die Betrachtung von KVR 1 von Track 1

Anhand der Auswertung des Bayesschen Netzes, lässt sich erkennen, dass das Beispiel in Abbildung 7-51 folgende Abhängigkeit aufweist. Hält sich *Track 1* nicht an Regel 14 und *Track 2* nicht an Regel 1 und wird zudem noch die kritische Distanz zwischen den beiden *Tracks* unterschritten, liegt die Wahrscheinlichkeit einer Schiffsbegegnung bzw. die Abhängigkeit dieser drei Faktoren bei 81,50 Prozent. Halten sich jedoch beide *Tracks* an die Regeln und die kritische Distanz wird nicht unterschritten, so beträgt die Abhängigkeit nur noch 37,40 Prozent.

7.19 Entscheidungsbaum (Decision Tree)

Zusätzlich zu dem Bayesschen Netz hat die Projektgruppe einen Entscheidungsbaum in KNIME aufgebaut. Der Entscheidungsbaum basiert auf den gleichen 25.310 Szenarien, die auch dem Bayessche Netz zugrunde liegen. Ebenfalls bekommt dieser über einen Partitionierungs-Knoten die Einteilung der Szenarien in 20 Prozent Lern- und 80 Prozent Trainingsdaten zugewiesen. Die Abbildung 7-52 zeigt den *Decision Tree*, wie dieser in KNIME aufgebaut ist.

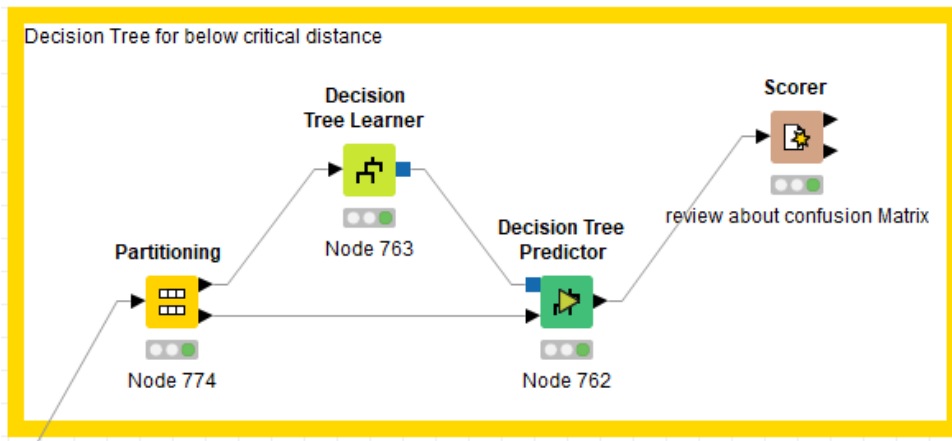


Abbildung 7-52: Entscheidungsbaum in KNIME

Wie schon aus dem Bayesschen Netz bekannt, wird auch bei dem Entscheidungsbaum erst partitioniert, dann lernt der Baum aus 20 Prozent der Daten und sagt anhand des Gelernten die anderen 80 Prozent der Szenarien voraus. Über den Scorer lässt sich anhand der *confusion matrix* überprüfen, wie korrekt der Entscheidungsbaum die Szenarien klassifizieren konnte.

Ein Ausschnitt des Entscheidungsbaumes für die Klasse „Unterschreitung der kritischen Distanz“ ist in Abbildung 7-53 dargestellt. Der gesamte Entscheidungsbaum kann dem Anhang 10.2 entnommen werden.

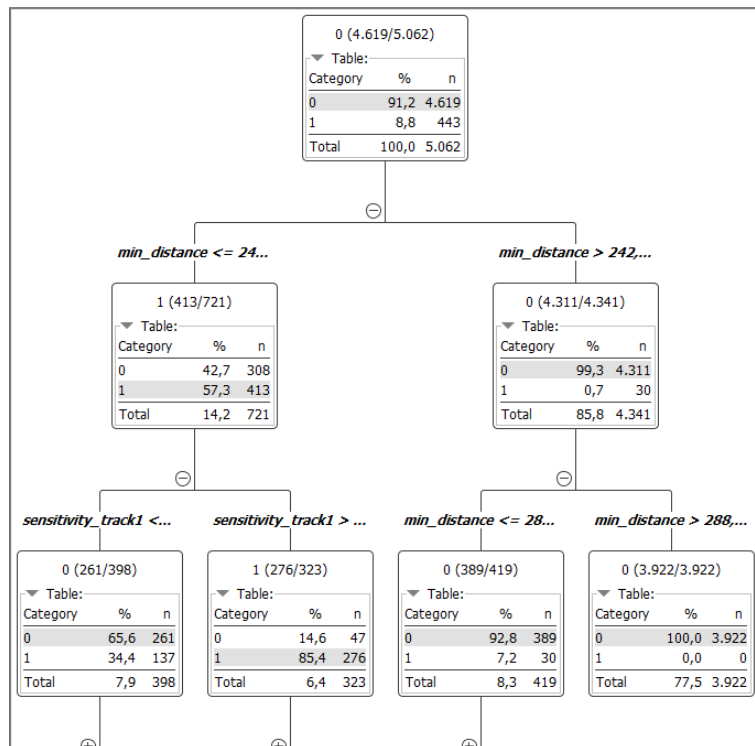


Abbildung 7-53: Entscheidungsbaum

Die *confusion matrix* in Abbildung 7-54 zeigt, dass sich durch den Entscheidungsbaum rund 99 Prozent der Szenarien korrekt klassifizieren lassen.

Confusion Matrix - 0:771 - Scorer (review about confusion Matrix)		
below_critical_distance \ ..	0	1
0	18366	109
1	84	1689

Correct classified: 20.055 Wrong classified: 193
 Accuracy: 99,047 % Error: 0,953 %
 Cohen's kappa (κ) 0,941

Abbildung 7-54: Confusion Matrix, Entscheidungsbaum

8 Xtreme Programming und Code Conventions

Im Rahmen der Projektgruppe MaGELLan wurde eine große Anzahl an Plugins entwickelt. Diese wurden von unterschiedlichen Entwicklern programmiert, die gute Kommunikation benötigten, um möglichst effizient und effektiv zu arbeiten. Die Unterstützung wurde einerseits durch kollaborative Arbeitsmittel, wie beispielsweise Jira, Outlook und Git gewährleistet und andererseits durch die Anwendung von Praktiken aus der Entwicklungsmethode *Xtreme Programming*, wie *Coding Standards*, *Collective Ownership* und *Pair Programming*.

In den folgenden Abschnitten wird zunächst das Verfahren des *eXtreme Programming* vorgestellt. Anschließend wird beleuchtet, welche *Coding Standards* in der Projektgruppe angewendet wurden.

8.1.1 Xtreme Programming

Xtreme Programming (Abkürzung XP) ist ein Entwicklungsprozess, der eine intensive Kommunikation zwischen Kunden und Entwicklern durch ein iteratives und inkrementelles Vorgehen ermöglicht (Winter 2017). Ein Vorteil des Prozesses ist die Entwicklung von schnellen und einfachen Lösungen und der Anwendung von vielen nützlichen Praktiken, wie vielen „Best Practices“. Die charakteristischen Mehrwerte des XP, im Sinne von Schnelligkeit und Einfachheit waren entscheidend, um die XP-Praktiken im Rahmen von der Projektgruppe MaGELLan zu übernehmen. Der folgende Abschnitt beschreibt die Praktiken, die während der Entwicklung des Projekts angewandt wurden.

8.1.1.1 Xtreme Programming Praktiken für MaGELLan

Im Projekt MaGELLan wurden drei XP-Praktiken am meistens verwendet:

- Programmieren in Paaren
- Testen
- Programmierrichtlinien (Coding Standards)

8.1.1.2 Testing

Ein der Ziele des Projekts MaGELLan ist es Begegnungssituation aus historischen Daten extrahieren zu können. Mit einer fortgeschrittenen Analyse, die mit

KNIME umgesetzt wurde, konnte die benötigten Daten gewonnen werden. KNIME selbst stellt die gängigsten Verfahren zur Datenanalyse, als sogenannte Knoten, zur Verfügung. Diese Knoten bieten unterschiedliche Verfahren für Data-Science und Data-Analysis, die bereits von KNIME getestet wurden. Daher wurde auf weitere Testverfahren von den KNIME inklusiven Knoten verzichtet. Die Funktionen, die nicht mit KNIME Knoten direkt umgesetzt werden können, werden durch eigene Plugins, umgesetzt werden. Die Umsetzung in KNIME wurde durch die Entwicklungsmethode *Test Driven Development* (TDD) durchgeführt, mit der die Qualität der eigenen Knoten gewährleistet werden kann. In den folgenden Abschnitten wird zuerst das TDD-Konzept erklärt, anschließend werden unsere unterschiedlichen Konzepte zur Umsetzung von Tests beschrieben und danach werden die unterschiedlichen Werkzeuge, die verwendet wurden, um das Konzept zu implementieren, vorgestellt.

8.1.1.3 Test Driven Development (TDD)

TDD-Entwicklungsmethode ist eine von vielen agilen Methoden. Ein großer Vorteil des TDD ist es, eine einzige Funktionalität gleichzeitig zu entwickeln und zu testen. Außerdem ermöglicht dieser Ansatz durch den Zwang von Tests, Fehler früh erkennen zu können und dadurch eine hohe Qualität für ein System zu garantieren (Beck 2002). Da das Projekt MaGellan im Bereich Datenanalyse liegt, wo die Qualität eine entscheidende Rolle spielt, kommt diese Methode zum Einsatz, um das Projekt möglichst mit den besseren Ergebnissen umzusetzen.

Es wäre wichtig, den Ansatz TDD zu definieren und seine Funktionsweise zu erklären, bevor es konkret beschrieben wird, wie die Methode im MaGellan angewendet wurde.

8.1.1.4 Definition und Funktionsweise

Test Driven Development ist ein Ansatz, bei dem der Test geschrieben wird, bevor der Quellcode beschrieben wird. In dieser Methode können die Tests schon in der Entwurfsphase vor der Implementierung gedacht werden. In der

Implementierung wird zuerst der Test hinzugefügt und danach der Code hinzugefügt und angepasst, solange der Test nicht erfolgreich ausgeführt wird.

8.1.1.5 Testkonzepte für das Projekt MaGellan

Ein Plugin ist eine KNIME Erweiterung, welche unter der Java OSGI Plattform entwickelt wird. Da jedes Plugin eine oder mehrere voneinander unabhängige Funktionen enthalten kann, können die Plugins mittels von Unit Tests getestet werden. Damit die unterschiedlichen Plugins für das Projekt MaGellan auf die gleiche Weise automatisiert und getestet werden können, wurde einerseits eine bestimmte Struktur für jedes Plugin definiert und andererseits die passenden OSGI-Konfigurationen hinzugefügt, um den Test zu erlauben. Dieser Prozess im TDD passiert in drei Schritten, wie es in der Abbildung 8-1 veranschaulicht wird:

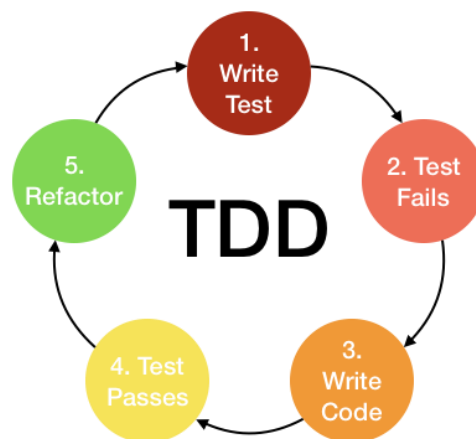


Abbildung 8-1: TDD Verfahren (Machado 2018)

8.1.1.6 Struktur eines Plugins für das Projekt MaGellan

Ein KNIME-Plugin wird bereits mit einer bestimmten Struktur generiert. Für einige Zwecke schien diese Struktur nicht ideal, weshalb eine eigene Struktur entwickelt wurde. Die Struktur wurde nicht nur geändert, um eine bessere Gliederung bezüglich der Code-Conventions zu gewährleisten, sondern auch um Tests effizient ausführen zu können. Die Abbildung veranschaulicht die Struktur eines Plugins im Projekt MaGellan.

Der Knoten (oder auch Plugin genannt) Gribweather, der für die Bearbeitung von Wetter Dateien aus dem GRIB-Format zuständig ist, wurde in die Pakete „knime“, „logic“ und „util“ gegliedert. Bezüglich der Tests, wird nur das Paket „logic“ betrachtet, weil dieses die Hauptmethoden des Knotens enthält. Damit das Paket im Test-Projekt aufgerufen werden kann, müssen auch noch bestimmte Plugin-Konfigurationen hinzugefügt werden, die im nächsten Abschnitt beschrieben werden.

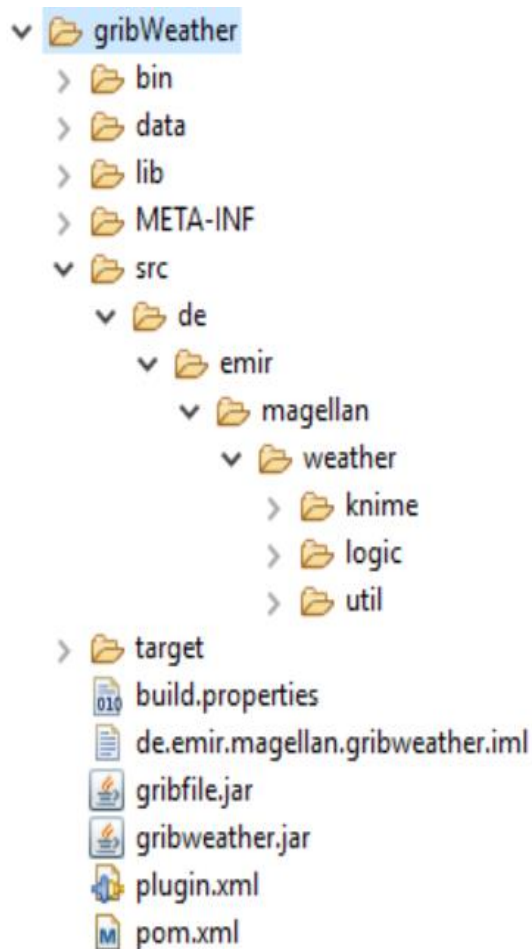


Abbildung 8-2: Hauptpakete KNIME-Plugin in MaGellan (Beispiel GribWeather Knoten)

8.1.1.7 Plugin-Konfiguration für Tests

```
Require-Bundle: org.eclipse.core.runtime,  
org.knime.workbench.core,  
org.knime.workbench.repository,  
org.knime.base,  
de.emir.magellan.core;bundle-version="1.0.0"  
Bundle-ActivationPolicy: lazy  
Import-Package: org.knime.core.data.time.localdatetime,  
org.knime.core.data.uri  
Export-Package: de.emir.haggis.magellan.crap,  
de.emir.haggis.magellan.crap.model,  
de.emir.haggis.magellan.crap.util;uses="de.emir.haggis.magellan.crap.model",  
de.emir.haggis.magellan.dcdatabase;uses="de.emir.haggis.magellan.dcdatabase.database,de.emir.haggis.magellan.dcdatabase.impl,de.emir.  
de.emir.haggis.magellan.dcdatabase.database;uses="org.postgresql.core",  
de.emir.haggis.magellan.dcdatabase.impl;uses="de.emir.haggis.magellan.dcdatabase,de.emir.haggis.magellan.crap.model",  
de.emir.haggis.magellan.dcdatabase.utils,  
de.emir.magellan.weather.knime,  
de.emir.magellan.weather.logic,  
de.emir.magellan.weather.util.conf,  
de.emir.magellan.weather.util.constant,  
de.emir.magellan.weather.util.exception,  
net.sourceforge.jgrib
```

Abbildung 8-3: Exportieren Pakete innerhalb von KNIME-Plugin

Da die Tests nicht zur reinen Funktionalität des Plugins beitragen, wurde ein Paket daneben zur Ausführung von Tests erstellt. Allerdings die zu testen Methoden sind erst sichtbar innerhalb von Test-Klassen nur, wenn sie in dem Plugin exportiert wird, wie es in der Abbildung 8-3 veranschaulicht wird und in der Test-Klassen importiert wird, wie es in der Abbildung 8-4 veranschaulicht wird. Exportieren und Importieren finden im Allgemeinen in der Datei MANIFEST.MF. Die Befehle dafür sind:

- Export-Package, um zu spezifizieren, welche Pakete exportiert werden.
- Import-Package, um zu spezifizieren, welche Pakete importiert werden.

```
Bundle-ActivationPolicy: lazy
Import-Package: de.emir.haggis.magellan.crap,
de.emir.haggis.magellan.crap.model,
de.emir.haggis.magellan.crap.util;uses:="de.emir.haggis.magellan.crap.model",
de.emir.haggis.magellan.dcdatabase;uses:="de.emir.haggis.magellan.dcdatabase.database,de.emir.haggis.magellan.dcdatabase.impl,de.emir.haggis.magellan.dcdatabase.database;uses:="org.postgresql.core",
de.emir.haggis.magellan.dcdatabase.impl;uses:="de.emir.haggis.magellan.dcdatabase,de.emir.haggis.magellan.crap.model",
de.emir.haggis.magellan.dcdatabase.utils,
de.emir.magellan.weather.knime,
de.emir.magellan.weather.logic,
de.emir.magellan.weather.netcdf,
de.emir.magellan.weather.util.conf,
de.emir.magellan.weather.util.constant,
de.emir.magellan.weather.util.exception,
net.sourceforge.jgrib,
de.emir.magellan.emirutilities,
de.emir.magellan.emirutilities.cusum,
de.emir.magellan.emirutilities.scenario,
de.emir.magellan.emirutilities.statistics,
de.emir.magellan.emirutilities.track,
de.emir.magellan.core,
de.emir.magellan.core.generic,
de.emir.magellan.core.helpers,
de.emir.magellan.core.model.core,
de.emir.magellan.core.model.core.impl,
de.emir.magellan.core.model.provider.datacell,
de.emir.magellan.core.model.provider.property,
de.emir.magellan.core.model.utils,
de.emir.magellan.core.utils
Require-Bundle: org.junit;bundle-version="4.12.0"
```

Abbildung 8-4: Importieren Pakete aus einem Plugin

Es wurde auch für MaGellan in der MANIFEST.MF alle Plugins importiert, damit alle gleichzeitig getestet werden können. Der Vorteil dafür ist es, der Test-Prozess automatisieren und den gesamten Code zu können.

In dem nächsten Abschnitt werden die Werkzeuge zur Test-Automatisierung und Code-Bewertung vorgestellt.

8.1.1.8 Werkzeuge zum Testen

Da Maven eine sehr lange Erfahrung zur Test-automatisierung hat und in der OSGI-Plattform durch das Tycho-Plugin verwendet werden kann, wurden nur die Werkzeuge für Test und Bewertung verwendet, die kompatibel mit Maven und Tycho sind. Nach der Überprüfung der Kompatibilität können die folgenden Tools genannt werden:

- Junit Version 4.12 (Unit-Tests schreiben und auszuführen)
- Sonar Cubes (Code-Qualität messen)
- Hamcrest-Framework (Vereinfachung von Tests)
- Tycho-Plugin (Maven Funktionalität)

Sobald diese Werkzeuge zusammen aufgestellt wurden, können Maven-Befehle aufgerufen werden, um Funktionen zu testen und Berichte bezüglich der Code-Qualität zu erzeugen.

In dem anschließenden Abschnitt wird auf die erhaltenen Ergebnisse der Messung der Qualität und Test-Abdeckung eingegangen.

8.1.1.9 MaGellan Programmierrichtlinien (Code Conventions)

Im folgenden Abschnitt werden die verschiedenen Standards vorgestellt, die für die Entwicklung von KNIME-Knoten für das Projekt MaGellan verwendet wurden. Diese Standards wurden in Übereinstimmung mit den Werkzeugen, die zur Entwicklung der Knoten verwendet wurden, erstellt. Dies sind die folgenden Werkzeuge:

- Die Programmiersprache Java
- Das Werkzeug zur Ausführung, Testing, Build Maven
- Die Plattform OSGI zur Entwicklung von Eclipse-Plugins

Für jedes Werkzeug gibt es Konventionen, die genauso übernommen wurden, oder für eine bessere Kommunikation und Einfachheit angepasst wurden.

8.1.2 Maven Conventions

Das Tycho Plugin ermöglichen es, Eclipse Plugins mit Maven zu erstellen (Sievers 2019). Der Knoten (Eclipse-Plugins) wurde mit Tycho entwickelt, um die Ausführung von Maven-Befehlen (Tests, Build, Run) zu ermöglichen und Tools für die Codequalität für die Entwicklung zu integrieren. Außerdem können die Werkzeuge für die Codequalität so leicht in Maven integriert werden. Es wird in diesem Teil die Maven-Konventionen nötig für das Projekt dargestellt.

Basierend auf den Maven Standards wurden die folgenden Namen-Konventionen verwendet (Maven 2019) **Parent Group Id:** de.emir.haggis

- **Parent Artifact Id:** emir-knime-nodes
- **Group Id:** de.emir.haggis
- **Artifact Id:** de.emir.magellan.<the name of the node>

Diese Konventionen wurde in jedem Projekt u.a. in jedem Plugin umgesetzt, damit jedes Projekt in Maven eindeutig identifizierbar ist.

Darüber hinaus werden von MaGellan-Entwickler selbst weitere Konventionen bezüglich Pakete-Struktur und -Organisation definiert, um die Maven befehle effizient auszuführen.

8.1.2.1 Package Structure and organization

Die Struktur jedes Plugins oder Knotens sollte die folgenden Pakete enthalten:

- **knime:** Dieses Paket sollte alle von KNIME generierten Dateien enthalten, wenn ein Knoten mit dem KNIME SDK erstellt wird.
- **logic:** Dieses Paket sollte alle Dateien oder Pakete enthalten, für die tatsächliche Funktionalität jedes einzelnen Knotens benötigt wird. In diesem Paket werden getrennte Unter-Pakete die Funktionen und Daten des Plugins erstellt, um eine eventuelle Erweiterung des Plugins zu erleichtern.
- **util:** Dieses Paket enthält allgemeine Klassen für Logging, Exception und Konstanten. Es kann allgemeine untere Pakete im Paket erstellt werden, damit die allgemeinen Klassen vernünftig unterteilt werden.

Die Abbildung 8-5 veranschaulicht die Pakete knime, logic und util innerhalb eines Plugins.



Abbildung 8-5: allgemeine Pakete des Plugins Gribweather

Zusätzlich zu Pakete-Konventionen werden Codierungskonventionen definiert, die die Regeln beschreiben, die bei der Entwicklung eines Knotens für das Projekt zu beachten sind. Der folgende Abschnitt beschreibt die Code-Konventionen, die an das Projekt MaGellan angewandt wurden.

8.1.2.2 Code Conventions

Die Code Konventionen dienen zu einer besseren Kommunikation und einem besseren Verständnis des Quellcodes während des Entwicklungsprozesses (Sun Microsystems 1999).

Da die Knoten mit der Programmiersprache Java implementiert wurden, werden die üblichen Java-Konventionen verwendet. Die folgenden JAVA-Konventionen wurden im Projekt angewandt:

- **Konventionen für Pakete:** Das Präfix eines Paketnamens wird immer in Kleinbuchstaben geschrieben und sollte einer der Top-Level-Domännennamen sein, derzeit com, edu, gov, mil, net, org oder einer der

englischen Zwei-Buchstaben-Codes zur Identifizierung von Ländern, wie in der ISO-Norm 3166, 1981 angegeben (Sun Microsystems 1999). Der Top-Level Domännennamen eines Plugin für MaGellan ist: **de.emir.magellan**.

- **Konventionen über Klassen:** Klassennamen sollten Substantive sein, wenn viele Substantiven im Namen verwendet werden, muss jeder mit einem großen Buchstaben anfangen. Außerdem sollten die Klassennamen beschreibend und simpel sein. Abkürzungen werden möglichst vermieden (Sun Microsystems 1999). Die Konvention wird beispielweise für das Plugin zur Bearbeitung von Wetter aus dem GRIB-Format verwendet: **GRIBRepository**.
- **Konventionen über Methoden:** Methoden sollten Verben sein. Wenn mehrere Verben nacheinander verwendet werden, sollten alle inneren Verben mit einem großen Buchstaben anfangen (Sun Microsystems 1999). Zum Beispiel `createGRIB(createGRIB (PostgreSQL.CopyConnection connection))`.
- **Konventionen über Variablen:** hier die Namenkonventionen für java Variablen.
- **Konventionen über Konstanten:** die Namenkonventionen für Java Konstanten.

Neben der Anwendung von Code-Koventionen, werden auch im Projekt „best practices“ angewendet, um die Kommunikation und das Verständnis des Codes weiter zu erleichtern.

8.1.2.3 Best practices

Best Practices oder Programmierrichtlinien sind Praktiken, die das Verständnis des Codes erleichtern und die Kommunikation vereinfachen (Winter 2017) .

Einige Beispiel für „best practices“, die in MaGellan zu finden sind:

- Verwaltung von Versionen in jeder POM
- Verwaltung von Repositorien in der Haupt-POM-Datei
- Dependency management in Haupt-POM-Datei
- Unit tests
- Verwaltung von Fehler und Logging

...

8.1.2.4 Dokumentation

Für die unterschiedlichen Plugins wurde die Datei „package-info“ erstellt. Diese Datei dient dazu das Paket zu beschreiben. Außerdem wurden alle Klassen und Methoden nach den üblichen Konventionen mittels Kommentare im Code dokumentiert.

Die Abbildung veranschaulicht die Struktur eines Plugins des Projekts MaGellan.

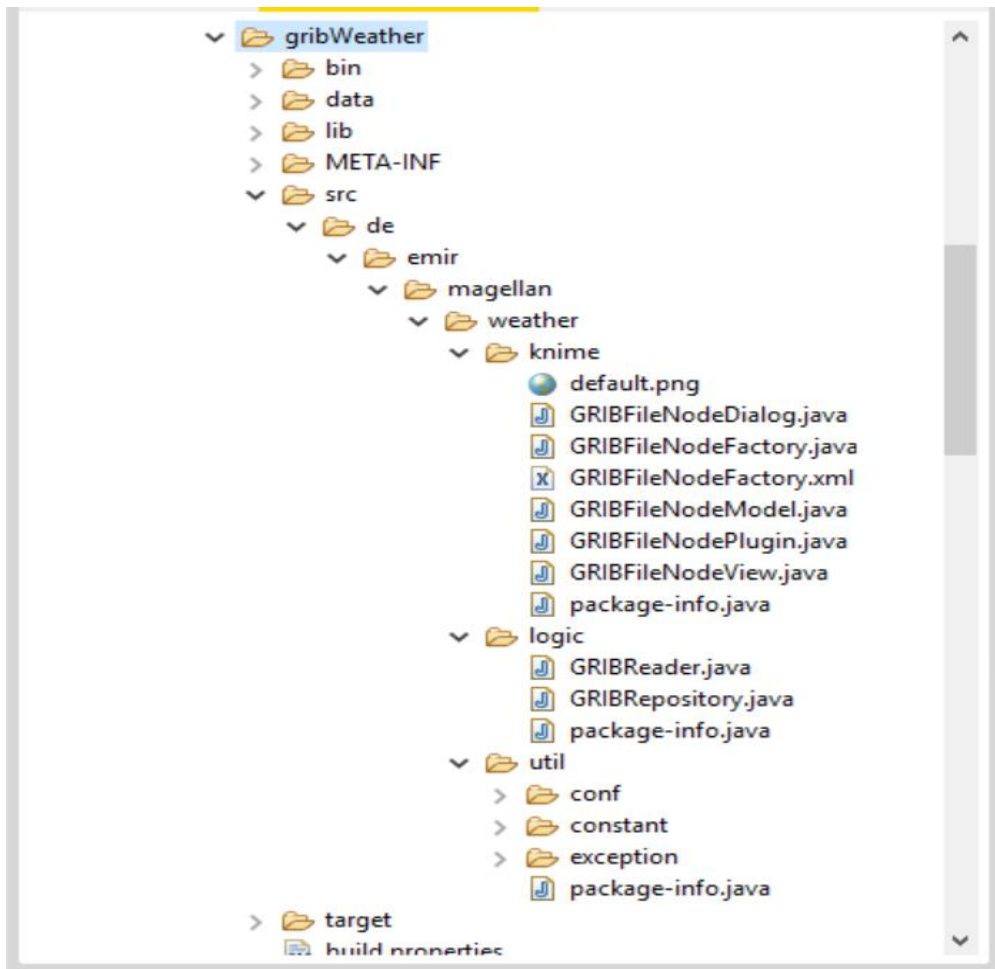


Abbildung 8-6: Pakete mit package-info

8.1.2.5 Regeln zur Entwicklung

- Jedes Plugin wurde im Allgemeinen in seinem eigenen Projekt implementiert. Es gibt jedoch Ausnahmen wo bei denen es mehrere Plugins in einem Projekt gibt. Die verschiedenen Plugins können als Maven-Module entwickelt.
- Jedes Plugin wird als Maven-Projekt erstellt, indem die POM-Datei in dem globalen Projekt referenziert wird, um Maven-Befehle im Projekt nutzen zu können.
- Es wurde bei der Entwicklung des Plugins einen try-catch-Block verwendet, in dem der catch-Teil die Fehler in KNIME-Log Datei schreibt, wenn sie auftreten.
- Alle Fehler wurden in den generierten KNIME-Klassen verwaltet, damit die Fehlernachrichten leicht in der KNIME-log Datei geschrieben werden können.

9 Fazit und Ausblick

Im Folgenden werden die Ergebnisse zusammengefasst und in einem Ausblick mögliche Erweiterungen vorgestellt.

9.1 Fazit

In diesem Abschnitt werden die Ergebnisse der Projektgruppe MaGELLan zusammengefasst.

Gegebene historische AIS- und Radardaten, welche unstrukturiert in einer Rohdatenbank gespeichert sind, werden mithilfe eines erstellten ETL-Prozesses in ein neues Datenmodell übertragen. Duplikate werden entfernt und möglicherweise fehlerhafte AIS- und Radardaten werden mit *Errorcodes* versehen. Des Weiteren werden aufgezeichnete AIS-Tracks zu Radartracks assoziiert und fusioniert, wenn sichergestellt werden kann, dass die *Tracks* einen ähnlichen Schifffspfad wiedergeben. Außerdem werden die AIS-Trackpoints auf zeitliche Differenzen überprüft und darauf basierend *Tracks* erstellt.

Wetterdaten, welche vom DWD und BSH im NetCDF-Format zur Verfügung gestellt wurden und die betrachteten Seegebiete zum größten Teil widerspiegeln, werden in einer eigenen Wetterdatenbank gespeichert, sodass diese für weitere Analysen genutzt werden können. Die u.a. in der Wetterdatenbank gespeicherte Sichtweite und der allgemeine Wetterzustand (gebildet aus Windgeschwindigkeit und Wellenhöhe) werden in fünf Klassen eingeteilt, welche eine Aussage darüber treffen sollen, wie gut bzw. schlecht die aktuelle Sichtweite und der Wetterzustand für einen Kapitän sind.

Übliche durchgeführte Manöver werden durch den CuSum-Algorithmus erkannt und anschließend geclustert, um Bereiche zu detektieren, an denen häufig Manöver von Schiffen gefahren werden. Die gebildeten Clusterpunkte werden in einem Netz miteinander verbunden, um die zeitliche Abhängigkeit der Manöver zu erkennen. Das gebildete Manövernetz ist richtungsabhängig, d.h. es gibt für jede Schiffsrichtung und -gegenrichtung ein eigenes Manövernetz. Mithilfe des Manövernetzes können schließlich anomale *Tracks* erkannt werden, wenn innerhalb der *Tracks* Manöver gefahren wurden, welche sich positionstechnisch von den Clusterpunkten des Manövernetzes unterscheiden.

Begegnungssituationen zwischen zwei Schiffen, welche eine Distanz von zwei Seemeilen unterschreiten, werden aus der Datenmenge herausgefiltert. Die Begegnungssituationen werden einem der Begegnungstypen *Overtaking*, *Following*, *Head-on*, *Passing* oder *Crossing* eingeteilt oder als *undefined* bezeichnet, wenn nicht ganz genau festgelegt werden kann, um was für einen Begegnungstypen es sich handelt.

Die Begegnungssituationen werden neben dem Begegnungstyp durch den Gewässertyp, den Wetterzustand, die Sichtweite, die Tageszeit, die Windrichtung und die Manövrierfähigkeit der beiden Schiffe klassifiziert. Für jede gebildete Klasse werden die minimalen Distanzen der zugehörigen Begegnungssituationen betrachtet und mithilfe einer Kerndichtschätzung eine kritische Distanz bestimmt. All die Begegnungssituationen, welche die kritische Distanz unterschreiten, werden als anomal und somit risikobehafteter angesehen. Neben der Erkennung von anomalen Begegnungssituationen dient die Klassifikation der Begegnungssituationen dazu, den Einfluss der verschiedenen Parameter auf die kritischen Distanzen ermitteln zu können. So sinkt beispielsweise die kritische Distanz bei *Passingsituationen*, wenn zwei Schiffe sich nicht im offenen Gewässer, sondern im Flussgebiet begegnen.

Neben der Klassifikation der Begegnungssituationen werden Regeln der KVR und SeeSchStrO überprüft, um Anomalitäten erkennen zu können. Die Regeln KVR 6, KVR 14, SeeSchStrO § 23 a), SeeSchStrO § 23 b) und SeeSchStrO § 23 c) werden bei den Begegnungssituationen überprüft. Das Risiko einer Kollision wird bei den Begegnungssituationen als höher angesehen, wenn eine der Regeln verletzt wird.

Verschiedene Statistiken und Verteilungen lassen sich durch verschiedene Workflows ansehen. Durch *Heatmaps* können die Durchschnittsgeschwindigkeit, Verkehrsdichte und die Dichte von Begegnungssituationen visualisiert werden. Ferner lassen sich innerhalb der Begegnungssituationen die prozentuale Verteilung der Begegnungstypen, Gewässerarten, Schiffsklassen und eingehaltenen Regeln betrachten.

9.2 Ausblick

Die gesammelten Schiffsdaten können noch durch externe Datenbestände ergänzt werden, denn es stehen für manche Schiffe keine Informationen wie beispielsweise die Länge und Breite zur Verfügung. Zudem können durch externe Quellen bisher nicht betrachtete Daten (bspw. Antriebstyp) eingetragen werden und in den Analyseprozessen benutzt werden.

Beim ETL-Prozess kann noch ein Teilprozess zur Bereinigung/Glättung der Positionsdaten eingefügt werden, welcher zuverlässig fehlerhafte Positionen erkennt und bereinigt.

Bei der Erstellung eines Manövernetzes sollten die Parameter des DBSCAN-Algorithmus zur Bildung der Clusterpunkte basierend auf dem Verkehrsaufkommen der betrachteten Region eingestellt werden. Des Weiteren kann an einem Ansatz gearbeitet werden, durch welchen mithilfe des Manövernetzes die geplanten Ziele von Schiffen prädiziert werden, basierend auf den bisher abgefahrenen Manöverpunkten.

Für die Berechnung der Wahrscheinlichkeiten der Begegnungstypen sollten bei der Berechnung des *Bearings* und der Differenzen der COG-Werte in Zukunft anstatt des COG-Werts der *Heading*-Wert genommen werden, da die Ausrichtung der Schiffe für die Berechnungen wichtig sind und nicht der *Drift*, den der COG-Wert noch berücksichtigt. Im Moment werden die COG-Werte genutzt, da die *Heading*-Werte zu häufig fehlerhaft sind. Ferner sollte noch evaluiert werden, welche Schwellenwerte für die Festlegung der Begegnungstypen die besten Ergebnisse liefern, bisher wurden die Schwellenwerte nur durch Ausprobieren festgelegt. Zusätzlich sollte noch untersucht werden, warum beim gegenwärtigen Verfahren so wenig Head-on und Crossingsituationen identifiziert wurden. Außerdem können noch weitere Begegnungstypen erstellt werden (bspw. Schleppermanöver), um die Situationen weiter differenzieren zu können. Zuletzt kann zur Erkennung von Begegnungssituationen ein Verfahren entwickelt werden, welches dies mit überwachendem Lernen macht.

Bei der Klassifikation können noch weitere Parameter hinzugefügt werden, um die Begegnungssituationen noch genauer zu beschreiben. Gerade die Wetterparameter können noch um die Strömung, den Wasserstand, Bewölkungsgrad und den Niederschlag ergänzt werden. Des Weiteren kann untersucht werden, ob es neben einer Kerndichtschätzung noch weitere Möglichkeiten gibt, für die einzelnen Klassen kritische Distanzen zu bestimmen bzw. den Grad der Anomalität noch genauer zu bestimmen. Bisher werden alle Begegnungssituationen einer Klasse, welche die kritische Distanz ihrer Klasse unterschreiten, als anomal gesehen, jedoch wird keine Aussage darüber getroffen, wie anomal diese Begegnungssituation genau ist. Bei der Analyse der Einflüsse der einzelnen Parameter auf die kritischen Distanzen kann nach Verfahren Ausschau gehalten werden, welche den Einfluss der Parameter interpretieren.

Die Regeln, welche innerhalb der Begegnungssituationen überprüft werden, können noch ergänzt werden. So fehlen beispielsweise noch die Regeln KVR 15, 16 und 17 für die Betrachtung von *Crossing*-Situationen. Die integrierten Seekarten können noch mehr Informationen beinhalten. Es werden nicht alle Brückenbereiche und generell keine Bojen berücksichtigt. Solche Informationen sind jedoch relevant bei der Überprüfung mancher Regeln wie die Überprüfung von KVR 9 (enges Fahrwasser).

Literatur

- AG. 2014. „Geschwindigkeitsbegrenzung | Hamburg für die Elbe“. 2014. <http://www.hamburg-fuer-die-elbe.de/?tag=geschwindigkeitsbegrenzung>.
- American Bureau of Shipping. 2006. „GUIDE FOR VESSEL MANEUVERABILITY“. https://ww2.eagle.org/content/dam/eagle/rules-and-guides/current/conventional_ocean_service/145_vesselmaneuverability/Vessel_Maneuverability_Guide_e-Feb17.pdf.
- „ASTERIX“. 2012. 1. EUROPEAN ORGANISATION FOR THE SAFETY OF AIR NAVIGATION. https://www.eurocontrol.int/sites/default/files/field_tabs/content/documents/single-sky/specifications/20120401-asterix-spec-v2.0.pdf.
- Beck, Kent. 2002. *Test-Driven-Development*. 1. Aufl. Addison-Wesley Professional.
- Bernardo Najlis. 23:37:26 UTC. „Introduction to Knime“. Technologie. <https://de.slideshare.net/bnajlis/introduction-to-knime>.
- Beschorner, Jenny. 2012. „Seminar ASidA - Kalman Filter“. <https://www.uni-koblenz-landau.de/de/koblenz/fb4/ist/AGZoebel/Lehre/ss2012/seminar/jBeschorner>.
- bmvit. 2018. „Seeschifffahrtsrecht“.
- „CDC (Climate Data Center)“. o. J. CDC (Climate Data Center). Zugegriffen 5. April 2019. https://www.dwd.de/DE/klimaumwelt/cdc/cdc_node.html.
- „COLREG“. 2018. 2018. <http://www.imo.org/en/About/conventions/listofconventions/pages/colreg.aspx>.
- ELWIS. 2002. „ELWIS - Vierter Abschnitt“. 2002. <https://www.elwis.de/DE/Schifffahrtsrecht/Seeschifffahrtsrecht/SeeSchStrO/Vierter-Abschnitt/Vierter-Abschnitt-node.html;jsessionid=DAD3ED5B12F7C79EE184970148676B03.server1t1>.
- . 2018. „Seeschifffahrtsstraßen-Ordnung (SeeSchStrO)“. 2018. https://www.elwis.de/DE/Schifffahrtsrecht/Verzeichnis-Rechtsverordnungen-Gesetze/SeeSchStrO.pdf;jsessionid=A81676A574BFCA39DFD34B1FD2C459F5.server2t1?__blob=publicationFile&v=10.
- Fischer, Yvonne. 2016. „Wissensbasierte probabilistische Modellierung für die Situationsanalyse am Beispiel der maritimen Überwachung“. PhD Thesis, Karlsruher Institut für Technologie (KIT). <https://doi.org/10.5445/KSP/1000051065>.
- Geduldig, Alena Tabea. 2014. „Parallelisierung des Levenshtein Algorithmus“. <http://www.spinfor.phil-fak.uni-koeln.de/sites/spinfor/geduldia/Levenshtein.pdf>.
- Hänninen, Maria, und Pentti Kujala. 2012. „Reliability Engineering and System Safety“. Elsevier.

- „KNIME Open Source Story“. o. J. Knime Open Source Story. Zugegriffen 5. November 2018. <https://www.knime.com/knime-open-source-story>.
- „KNIME Server | KNIME“. o. J. Knime Server. Zugegriffen 5. November 2018. <https://www.knime.com/knime-software/knime-server>.
- Kriegisch, Alexander. o. J. „Daily Scrum Meeting“. Zugegriffen 5. April 2019a. https://scrum-master.de/Scrum-Meetings/Daily_Scrum_Meeting.
- . o. J. „Sprint“. Zugegriffen 4. April 2019b. <https://scrum-master.de/Scrum-Meetings/Sprint>.
- Kuwata, Yoshiaki, Michael Wolf, Dimitri Zarzhitsky, und Terrance Huntsberger. 2014. „Safe Maritime Autonomous Navigation With COLREGS, Using Velocity Obstacles“, Januar.
- „Lineare Interpolation“. o. J. Interpolation in Wertetabellen. Zugegriffen 5. April 2019. https://elearning.physik.uni-frankfurt.de/data/FB13-PhysikOnline/lm_data/lm_281/modul_2/teil_5/node13.html.
- Machado, Luis. 2018. „Swift — Test Driven Development (TDD)“. *Luis Machado* (blog). 3. April 2018. <https://medium.com/@luisfmachado/swift-test-driven-development-tdd-810add46a1b9>.
- MARITIME SAFETY COMMITTEE. 2002. „STANDARDS FOR SHIP MANOEUVRABILITY“. MARITIME SAFETY COMMITTEE. [http://www.imo.org/en/KnowledgeCentre/IndexofIMOResolutions/Maritime-Safety-Committee-\(MSC\)/Documents/MSC.137\(76\).pdf](http://www.imo.org/en/KnowledgeCentre/IndexofIMOResolutions/Maritime-Safety-Committee-(MSC)/Documents/MSC.137(76).pdf).
- Maven. 2019. „Maven – Guide to Naming Conventions“. 2019. <https://maven.apache.org/guides/mini/guide-naming-conventions.html>.
- McCormick, Chris. 2014. „Kernel Regression“. *Chris McCormick* (blog). 26. Februar 2014. <http://mccormickml.com/2014/02/26/kernel-regression/>.
- McNeese, Bill. 2014. „CUSUM Charts“. [https://www.spcforexcel.com/publications/CUSUM-Charts/mobile/index.html?doc=D0AA4E49DA34265475BC23EFA9EE1FF8#{"docid":"D0AA4E49DA34265475BC23EFA9EE1FF8","page":0}](https://www.spcforexcel.com/publications/CUSUM-Charts/mobile/index.html?doc=D0AA4E49DA34265475BC23EFA9EE1FF8#{).
- „Messaging that just works — RabbitMQ“. o. J. Zugegriffen 5. April 2019. <https://www.rabbitmq.com/>.
- Oltmann, Jan-Hendrik. 2015. „ACCSEAS North Sea Region Route Topology Model (NSR-RTM) Description and contribution to an international generic Route Topology Model definition“.
- Pietrzykowski, Zbigniew, und Janusz Uriasz. 2009. „The Ship Domain – A Criterion of Navigational Safety Assessment in an Open Sea Area“. *Journal of Navigation* 62 (01): 93. <https://doi.org/10.1017/S0373463308005018>.
- Schl, Udo. (2016) 2019. *Concave Hull Java Implementation*. Java. <https://github.com/Merowech/java-concave-hull>.
- Schüssler, Nadine, und Kay W. Axhausen. 2008. „Processing GPS raw data without additional information“. Research Collection.
- Sievers, Jan. 2019. „Tycho Home | The Eclipse Foundation“. 2019. <https://www.eclipse.org/tycho/>.
- „SPx Tracking Server (Tracking, Plot Extraction, Radar Video Distribution, Recording)“. o. J. Zugegriffen 5. April 2019. <https://www.cambridgepixel.com/products/SPx-Server/>.

- „Stationsliste der 78 Messstationen (nach Stationsname sortiert)“. 2019. Stationsliste. 5. April 2019. <https://www.dwd.de/DE/leistungen/klimadatendeutschland/stationsuebersicht.html>.
- „The IMO E-Navigation Concept and the IHO S-100 Data Standard“. o. J. Zugegriffen 5. April 2019. <https://hydro-international.com/content/article/the-imo-e-navigation-concept-and-the-iho-s-100-data-standard>.
- Tsiroglou, Christina Maria. 2018. „Automatic Identification System (AIS) - Probleme und Ansätze sie zu lösen“.
- „UN/LOCODE Code List by Country - Trade - UNECE“. o. J. Zugegriffen 6. April 2019. <https://www.unece.org/cefact/locode/service/location>.
- US Department of Commerce, NOAA. o. J. „Significant Wave Height“. Zugegriffen 5. April 2019. https://www.weather.gov/key/marine_sigwave.
- Wagner, Marc. 2000. „Bayes-Netze Eine Einführung“. <https://th.physik.uni-frankfurt.de/~mwagner/talks/Bayes.pdf>.
- Wang, Yue Ying. 2012. „An empirical model of ship domain for navigation in restricted waters“. Dalian University of Technology.
- Wasserstraßen- und Schifffahrtsverwaltung des Bundes. 1977. „Internationale Regeln von 1972 zur Verhütung von Zusammenstößen auf See (Kollisionsverhütungsregeln - KVR)“. https://www.elwis.de/DE/Schifffahrtsrecht/Verzeichnis-Rechtsverordnungen-Gesetze/KVR.pdf;jsessionid=68249BC762DB2B3355F8DCF6439B6812.server2t2?__blob=publicationFile&v=6.
- Winter, Andreas. 2017. „Softwaretechnik Lecture.“ Universitaet von Carl Ossietszky, Niedersachsen.

10.2 Entscheidungsbaum

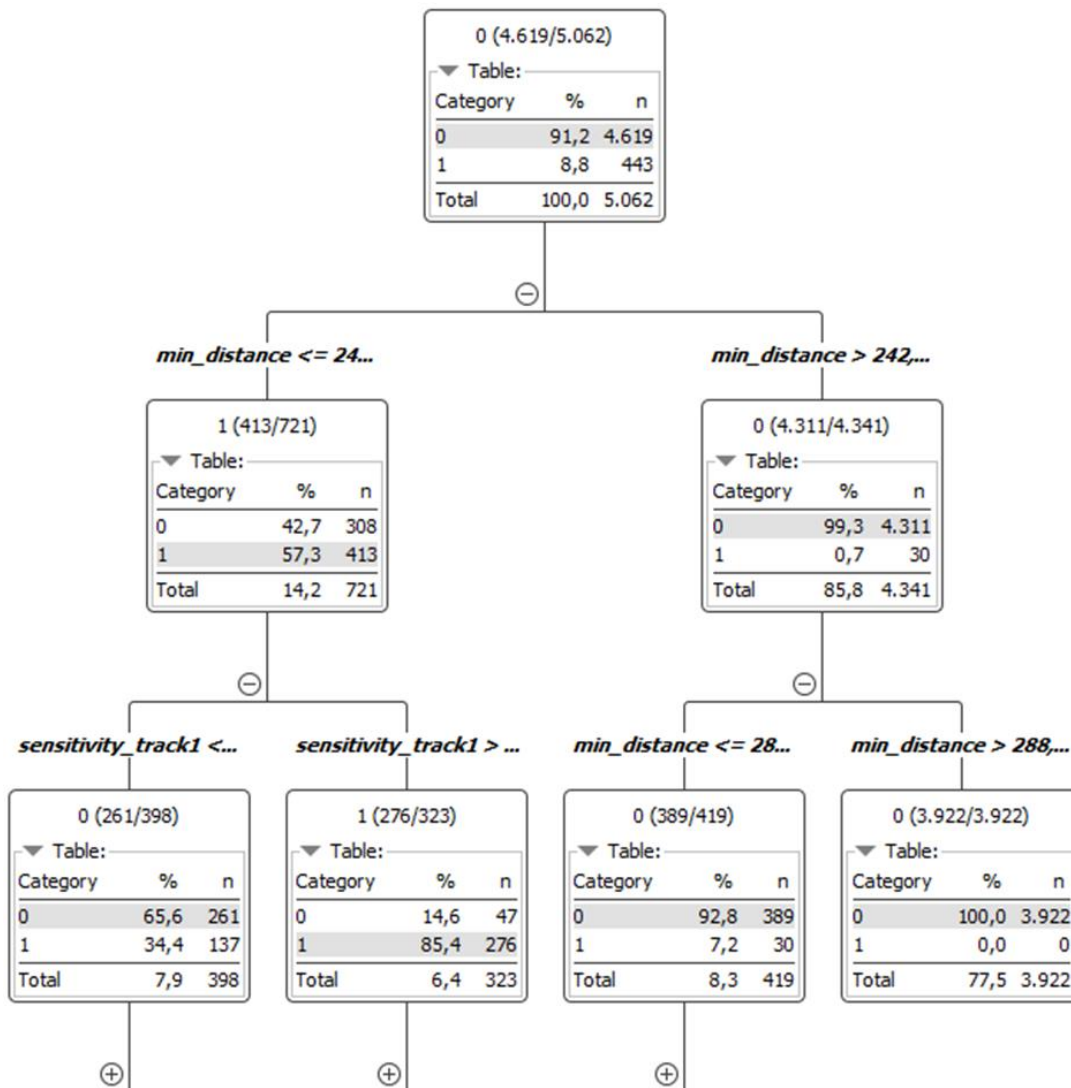


Abbildung 10-2: Vaterknoten mit erster und zweiter Ebene

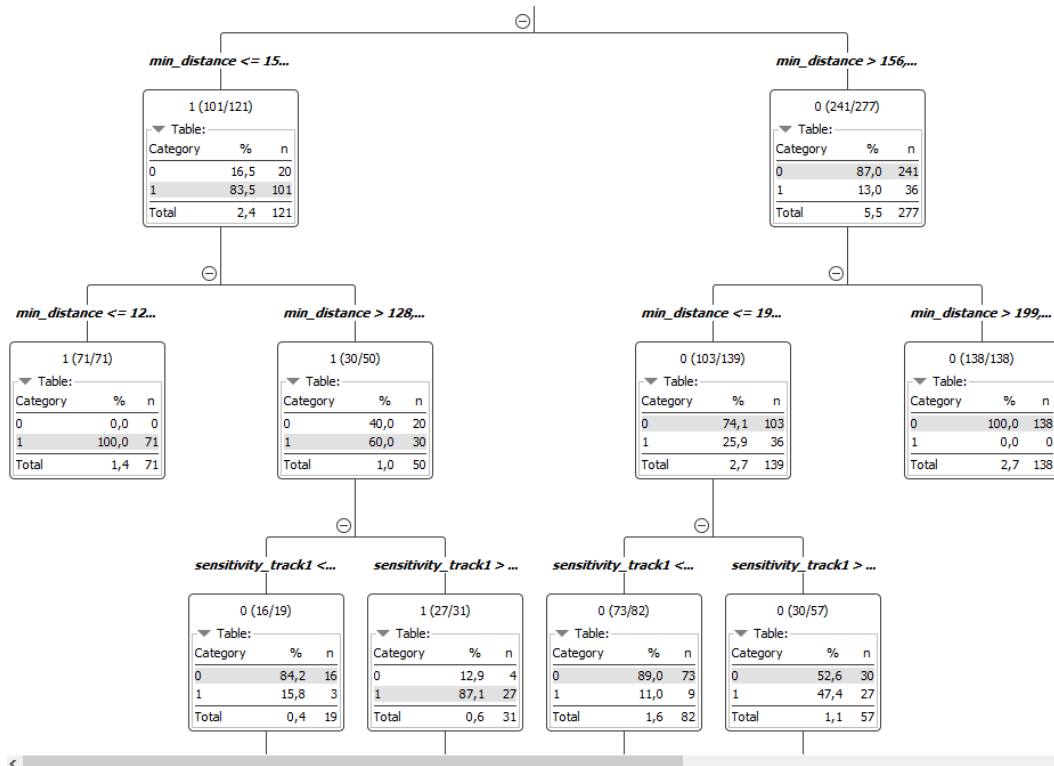


Abbildung 10-3: Linke Seite

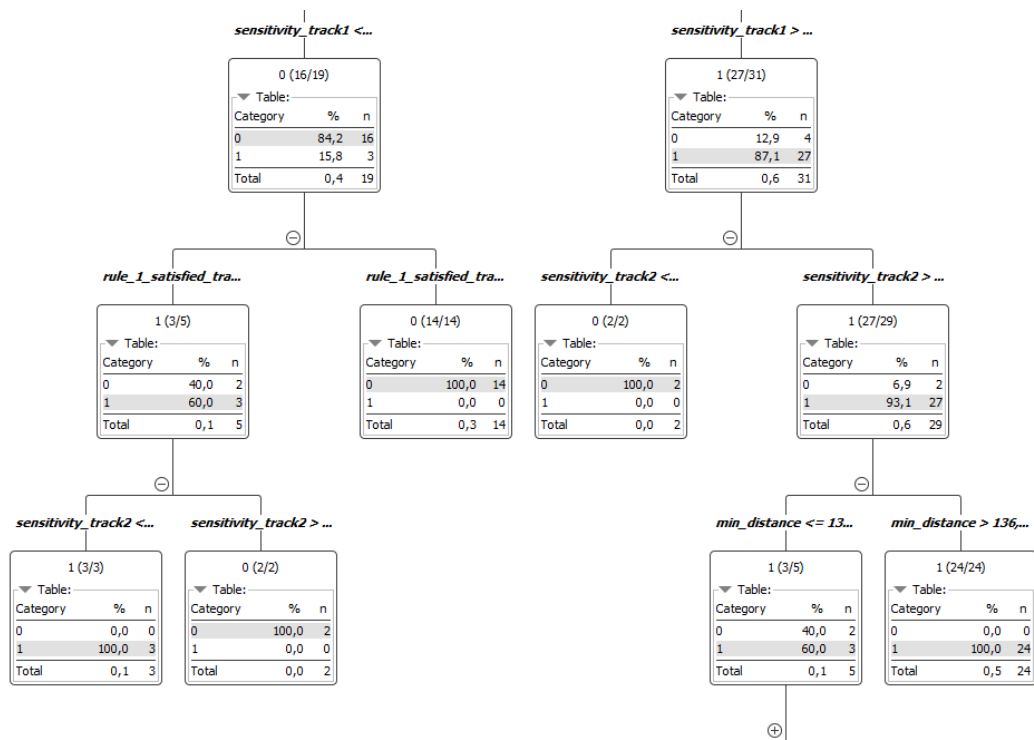


Abbildung 10-4: Linke Seite Fortsetzung

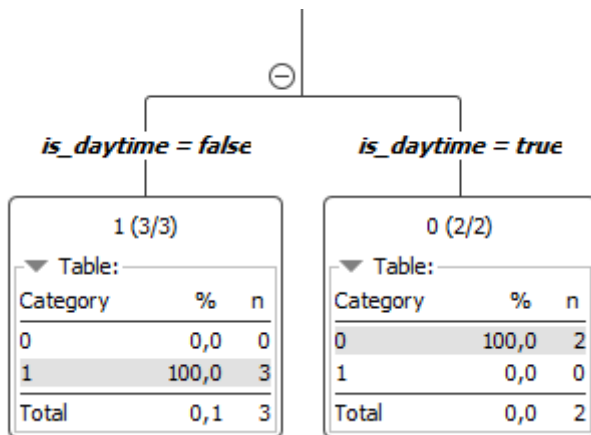


Abbildung 10-5: Linke Seite Ende

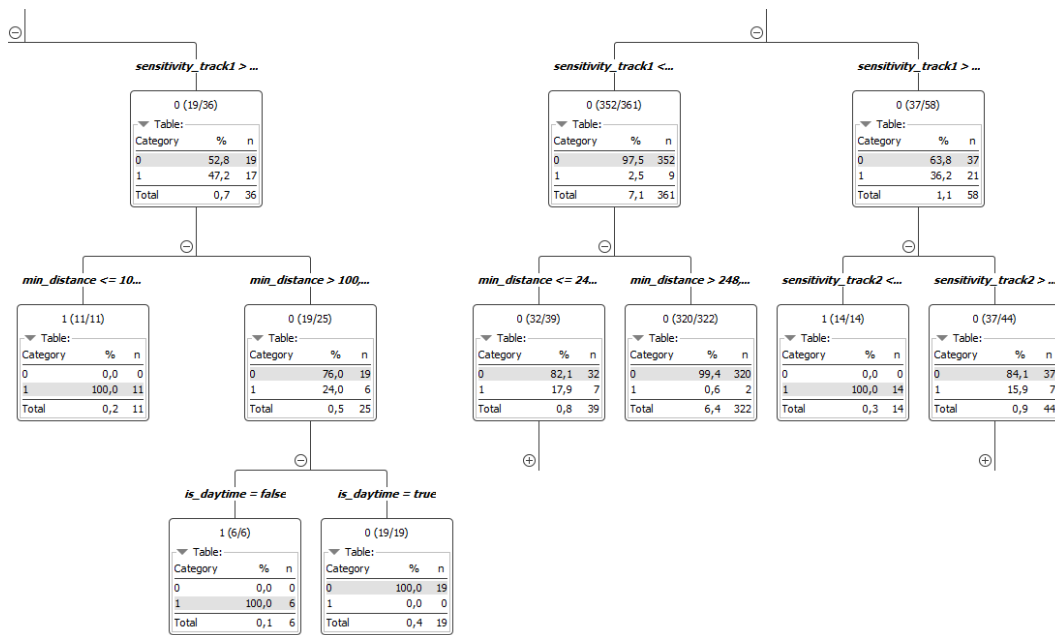


Abbildung 10-6: Rechte Seite des linken Pfades

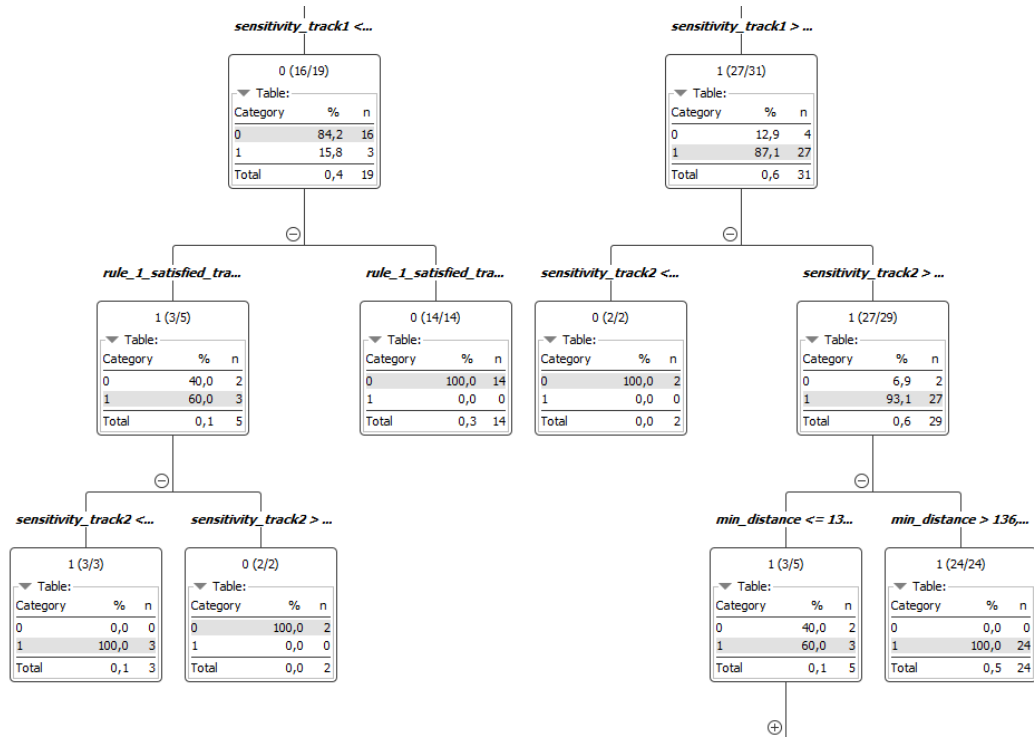


Abbildung 10-7: Rechte Seite des linken Pfades Ebene 2

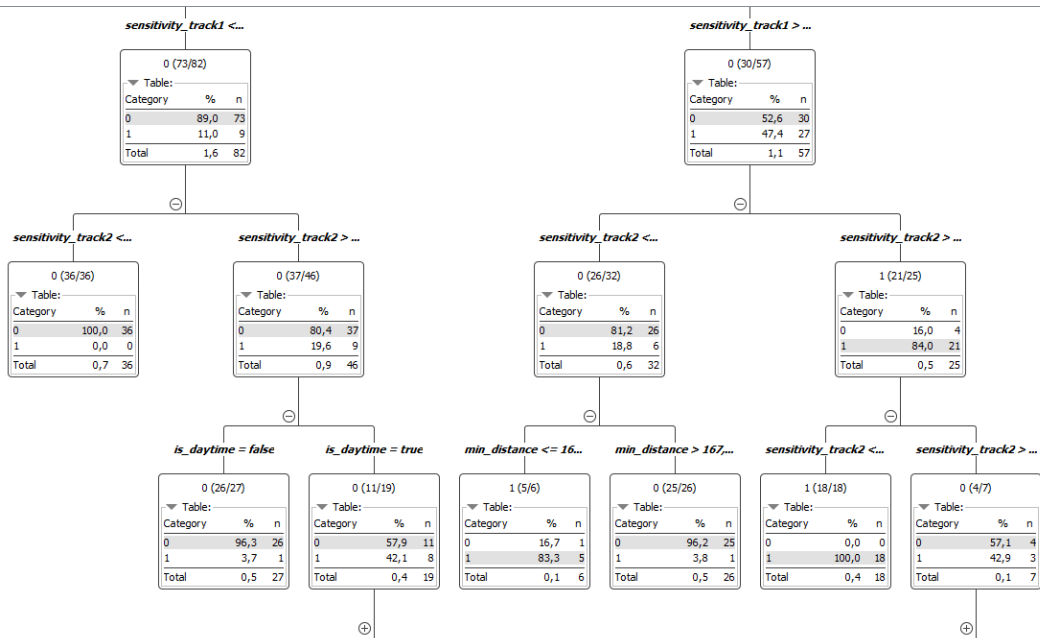


Abbildung 10-8: Rechte Seite des linken Pfades Ebene 2 links

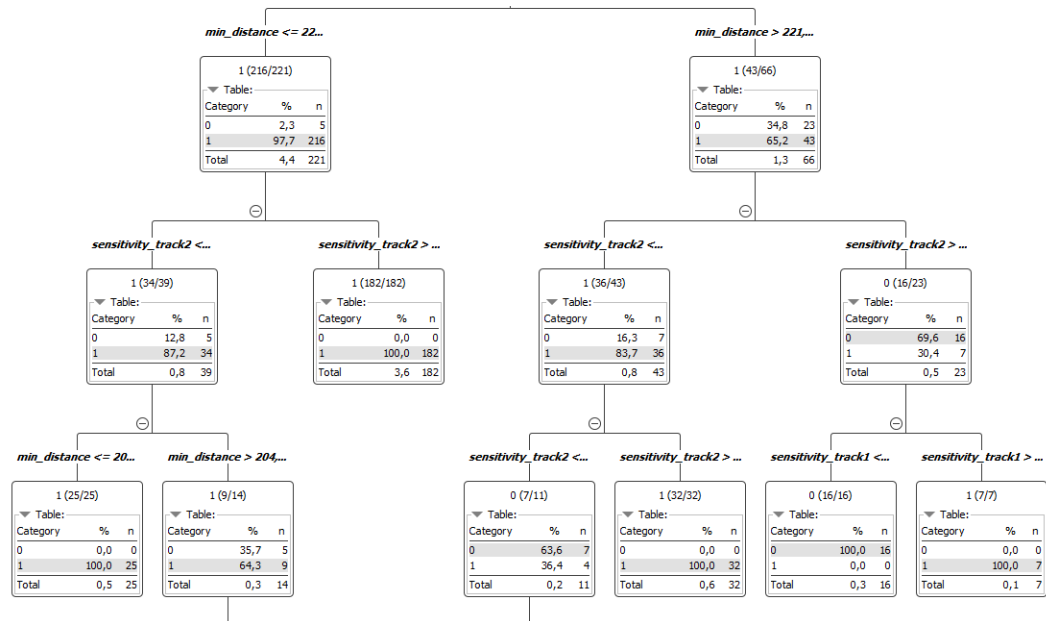


Abbildung 10-9: Rechte Seite des linken Pfades Ebene 2 rechts

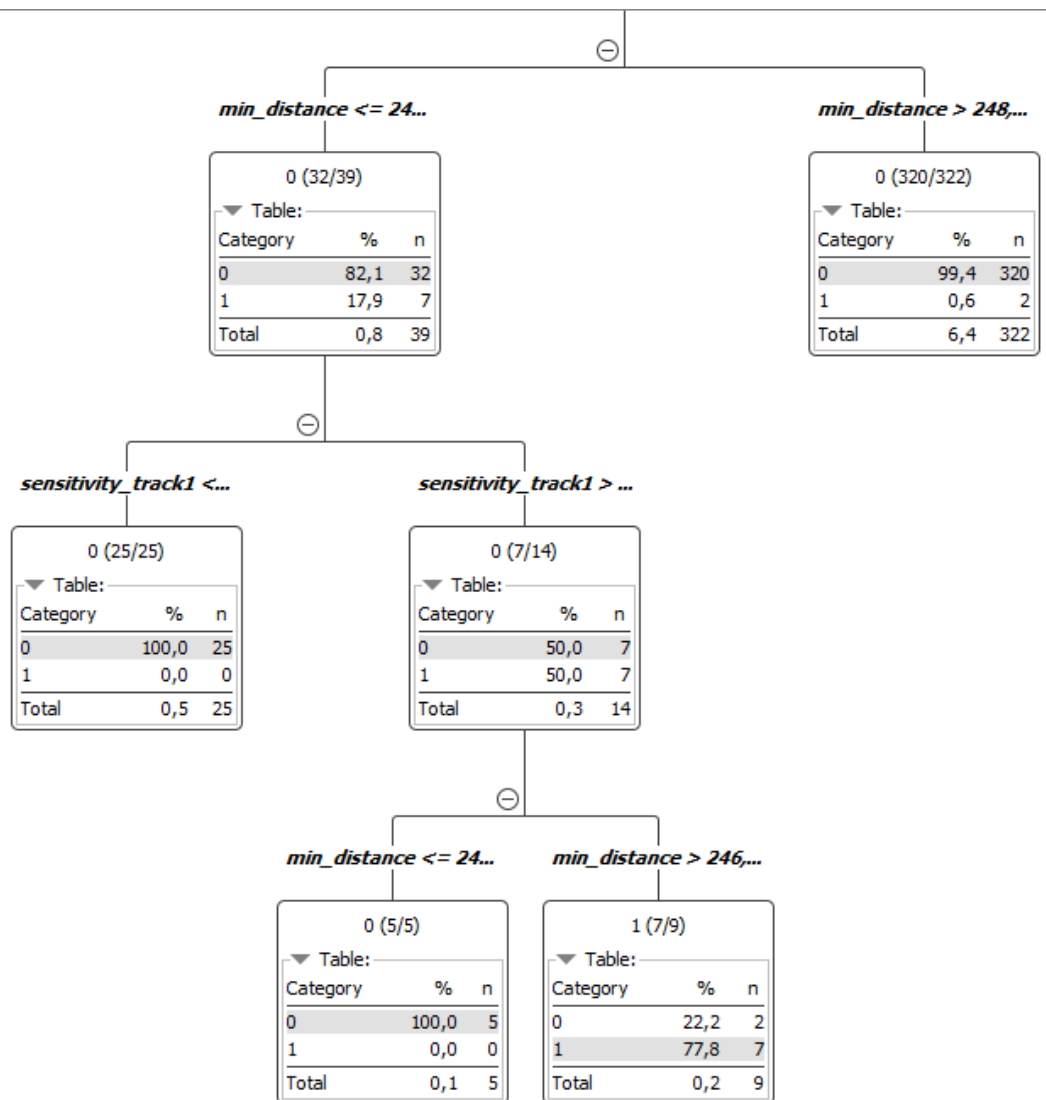


Abbildung 10-10: Linke Seite des rechter Pfades Ebene 2 links

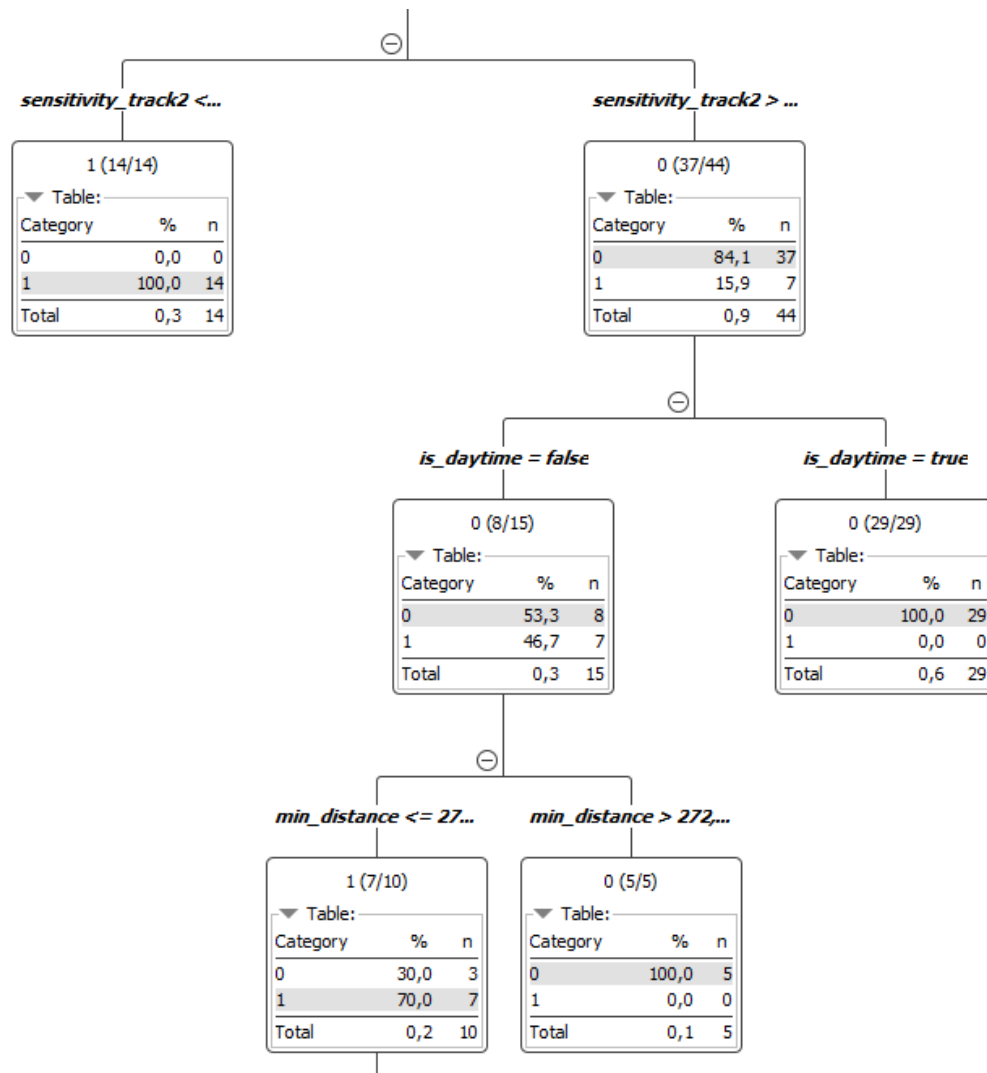


Abbildung 10-11: Linke Seite des rechter Pfades Ebene 2 rechts

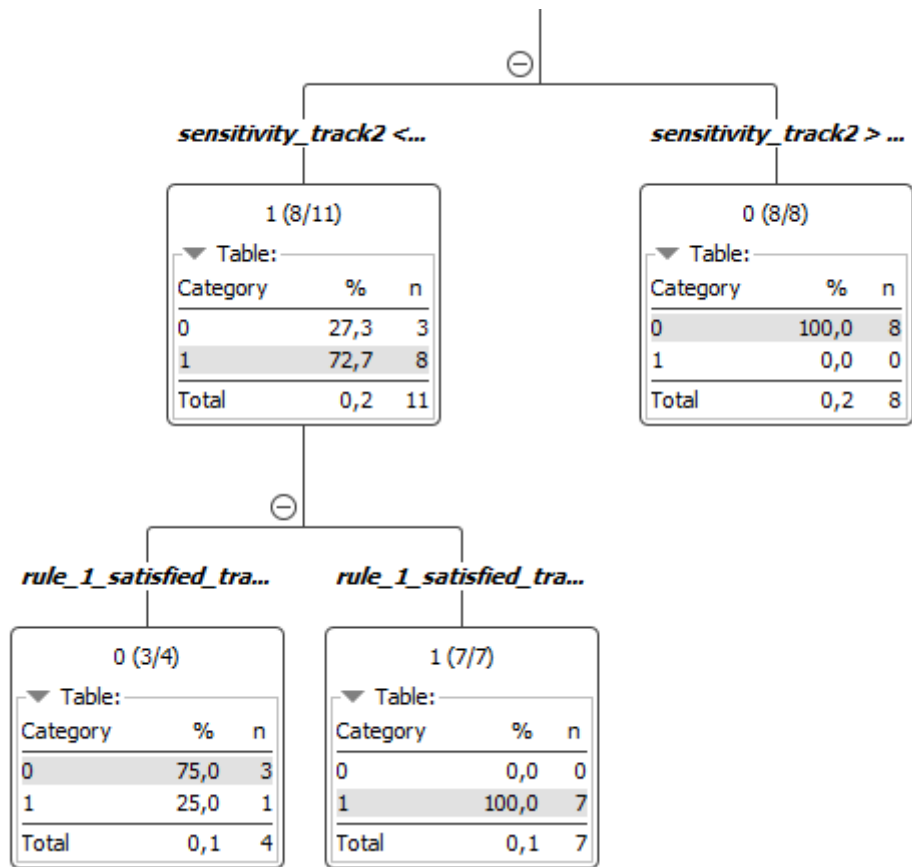


Abbildung 10-12: Rechte Seite der zweite Ebene linker Pfad Ebene drei rechts

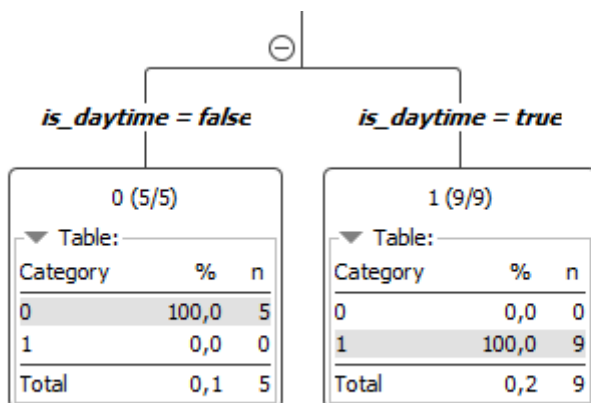


Abbildung 10-13: Linke Seite der zweite Ebene linker Pfad Ebene drei rechts

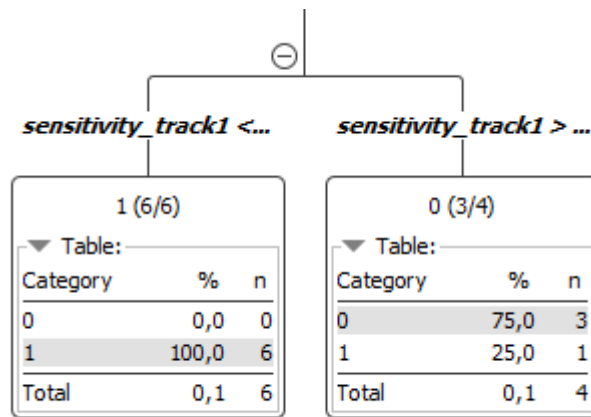


Abbildung 10-14: Rechte Seite der zweiten Ebene des rechten Pfades Ebene drei links

10.3 Heatmaps der verschiedenen Begegnungssituationen

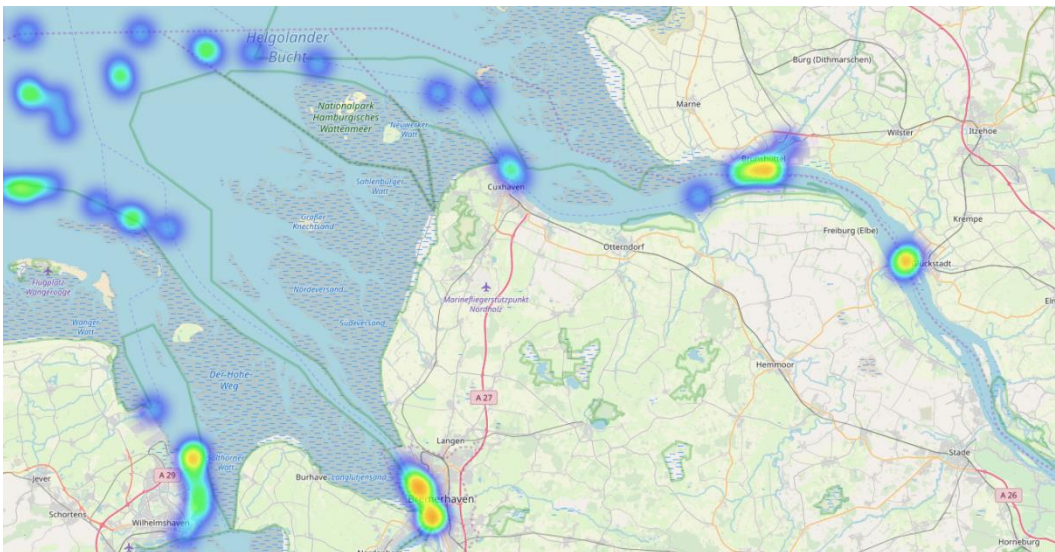


Abbildung 10-15: Heatmap Crossing

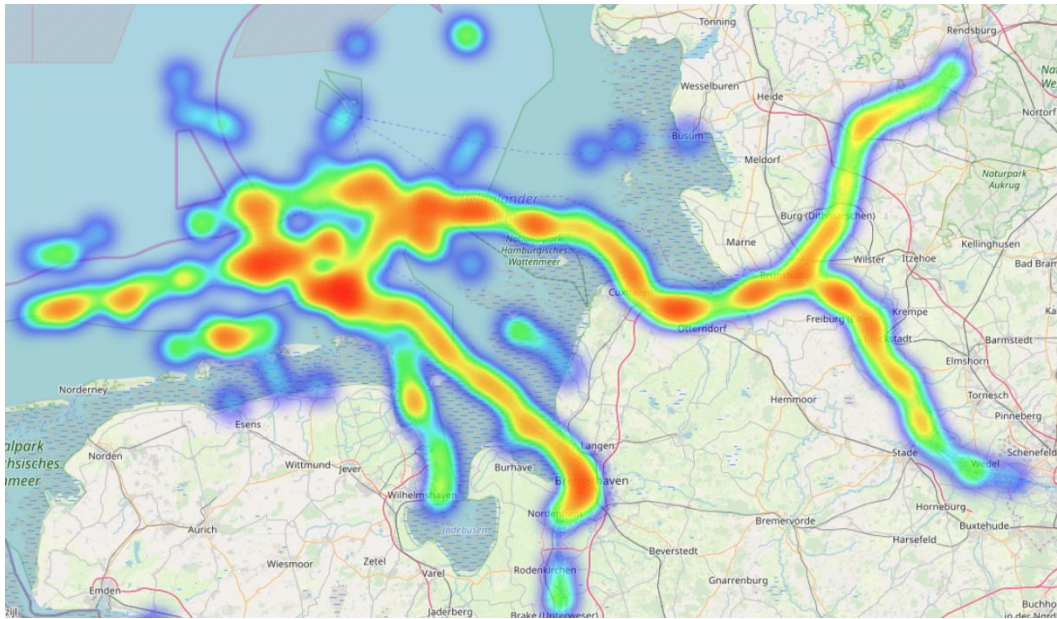


Abbildung 10-16: Heatmap Following

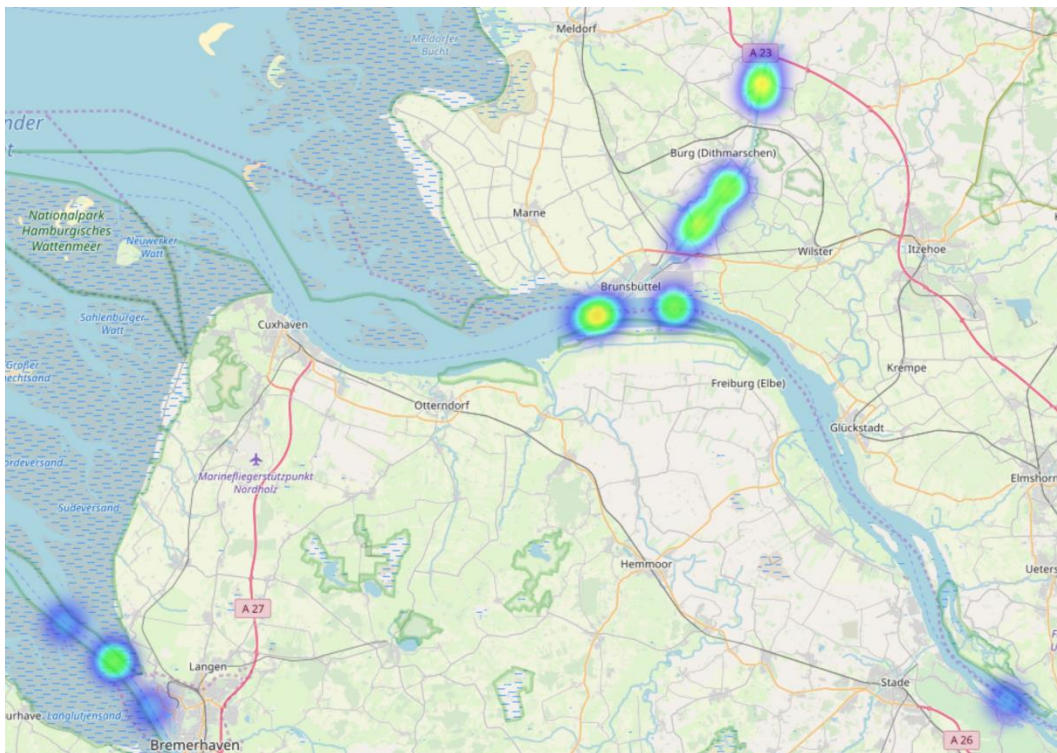


Abbildung 10-17: Heatmap Head-On

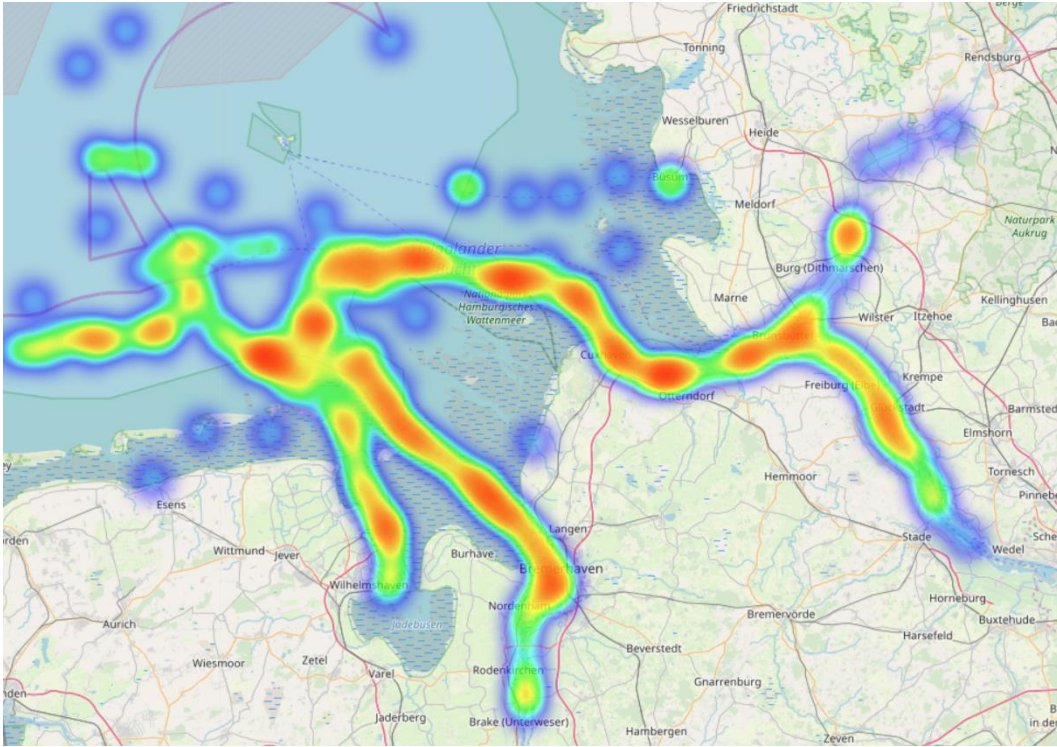


Abbildung 10-18: Heatmap Overtaking

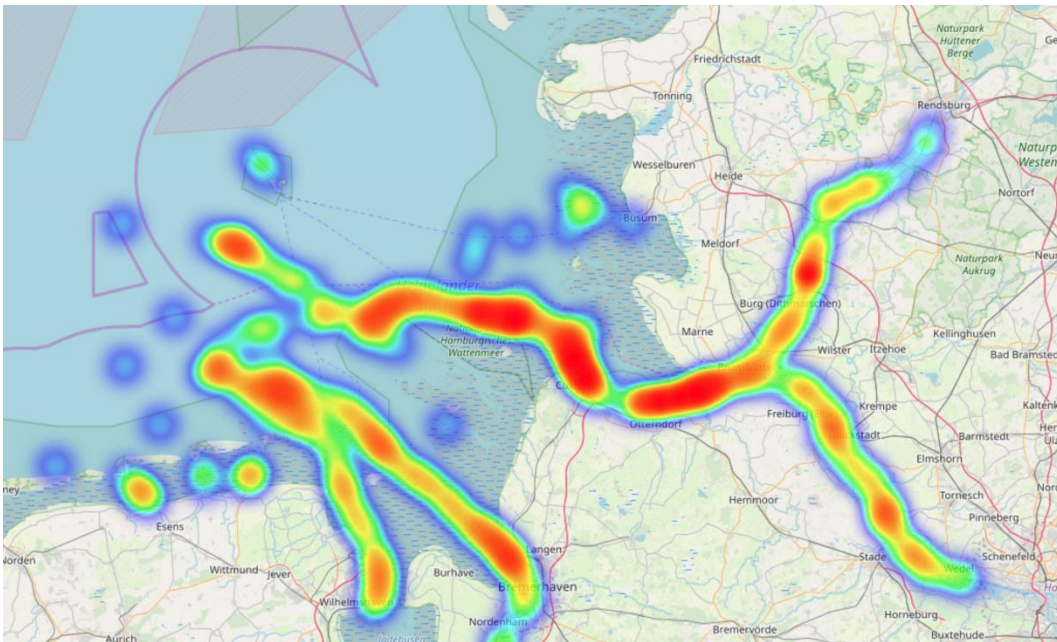


Abbildung 10-19: Heatmap Passing

10.4 Benutzerhandbuch

10.4.1 Benutzerhandbuch ETL-Prozess

Der ETL-Prozess steht auf KNIME Server unter dem Namen *ETL_MaGellan* im Ordner *MaGellan_Final_Workflows/ETL* sowie im git-Repository zur Verfügung.

Die Konfiguration des ETL-Prozesses erfolgt über die Anpassung der Workflow-Variablen, wie in Abbildung 10-20 zu sehen. Die Variablen *database_name*, *database_host* und *database_port* steuern dabei die Einstellungen der „Database Connector“-Knoten. Über die Variable *batch_size* wird gesteuert, wie viele Daten gleichzeitig bei den „Database Writer“-Knoten übertragen werden. Mit der Variable *amount_of_data* wird die Anzahl der Daten, die durch den ETL-Prozess bei jeder Iteration verarbeitet wird, gesteuert. Darüber hinaus kann mit der Variable *amount_of_iterations* die Anzahl der Loop-Iterationen, die bei jedem Ausführen des Workflows durchgeführt werden, gesteuert werden. Die Anzahl der Daten, die bei jeder Ausführung verarbeitet werden, ergibt sich daher aus dem Produkt der beiden Variablen *amount_of_data* und *amount_of_iterations*.

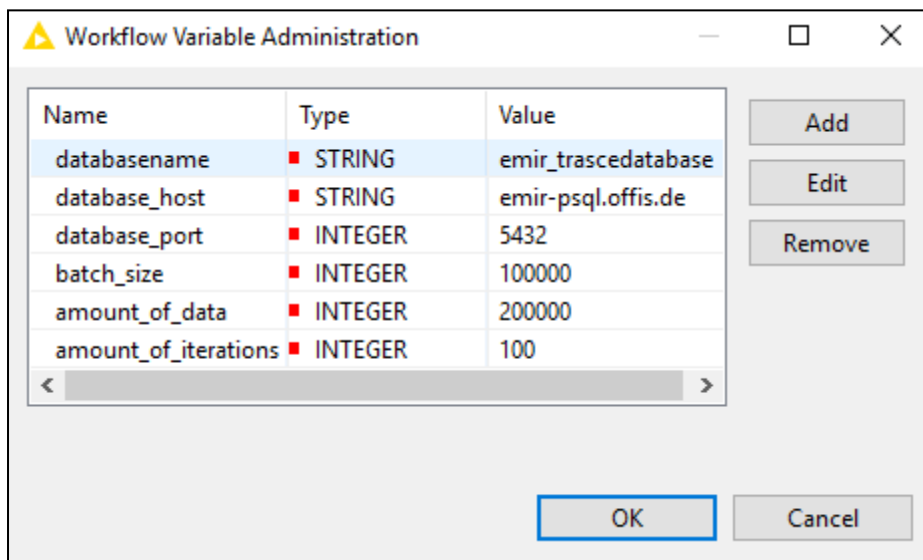


Abbildung 10-20 Workflow-Variablen des ETL-Prozesses

Der ETL-Prozess ist so gestaltet, dass die Ausführung automatisch stoppt, sobald der Datensatz mit der maximalen ID der Rohdaten aus der Tabelle *nbdatabase* erreicht und verarbeitet wurde oder sobald alle Iterationen ausgeführt wurden.

Um weitere Errorcodes in den ETL-Prozess zu integrieren, können diese in den Metaknoten *voyagerrelated ais errorcodes*, *static ais errorcodes*, *dynamic ais errorcodes* und *radar errorcodes* hinzugefügt werden.

10.4.2 Begegnungssituationen

Es gibt insgesamt vier zentrale Workflows, welche zur Analyse von Begegnungssituationen genutzt werden. Die wichtigsten Bedienelemente der Workflows werden nun vorgestellt.

Zunächst wird ein Workflow mit dem Namen *Encountersworkflow* genutzt, um zwischen den gebildeten *Tracks* Szenarien zu extrahieren. Am Anfang des Workflows können über die ersten beiden *Date&Time-Inputs* die Start- und Endzeit konfiguriert werden (Abbildung 10-21). Wichtig hierbei ist, dass der Starttag vor oder gleich dem Endtag ist.

The screenshot shows a configuration window for a 'Date & Time Input' variable. The window has four tabs: 'Control' (selected), 'Flow Variables', 'Job Manager Selection', and 'Memory Policy'. The configuration fields are as follows:

- Label:** Label
- Description:** Enter Description
- Variable Name:** start_time
- Parameter Name:** date-time-input
- "Now" Button in Wizard:**
- Granularity in Wizard:** Show minutes
- Type:** Date
- Earliest:** Date: 2019-02-21 Time: 12:38:33 Time Zone: Europe/Berlin Use execution time
- Latest:** Date: 2019-02-21 Time: 12:38:33 Time Zone: Europe/Berlin Use execution time
- Default Value:** Date: 2018-03-10 Time: 00:00:00 Time Zone: Europe/Berlin Use execution time

Abbildung 10-21: Konfiguration eines Date&Time Inputs

Über den ausgewählten Zeitraum werden Begegnungssituationen aus den gebildeten *Tracks* des ETL-Prozesses gebildet. Hierbei werden jeweils in zwei Stunden-Abschnitten *Tracks* aus der Datenbank abgefragt und miteinander bezüglich der Distanz verglichen. Demnach gib es 12 Iterationen bis *Tracks* aus einem Tag miteinander verglichen wurden.

Mit den *DB Query Readern* mit den Beschreibungen *get tracks from current time range*, *get track positions* und *get dynamic data from tracks* werden die jeweiligen *Tracks* und deren dynamischen Daten wie z.B. Positions- und Geschwindigkeitsdaten abgefragt. Es könnten die zugehörigen Tabellen *track*, *track_position* und *dynamic_plot* ausgetauscht werden, wenn beispielsweise nicht nur AIS-Daten betrachtet werden sollen. Im Metaknoten mit dem Namen *get*

static ship data werden statische Schiffsdaten abgefragt, um diese anschließend mit den dynamischen Daten zu joinen.

Weiter rechts im Workflow sind die vorher vorgestellten Knoten *DistanceFilter*, *MinDistanceFilter* und *Encounterlabeling* zu sehen.

Fast Mode (High RAM Usage)

Scenario ID

Track ID

MMSI

Time

Longitude

Latitude

COG

Heading

SOG

Shiptype

Length

Width

Bow

Starboard

Stern

Port

Sensitivity

Relevant distance

Relevant time interval

Abbildung 10-22: Bedienung des Knotens *MinDistanceFilter*

Bei den drei Knoten kann ausgewählt werden, ob diese in einem *fast mode* ausgeführt werden sollen, welcher schneller ist als der normale Modus, jedoch viel Arbeitsspeicher einnimmt und wenn der Rechner nicht genügend Arbeitsspeicher hat, kann der Workflow abstürzen. Bei den Knoten *DistanceFilter* und *MinDistanceFilter* kann auch noch die relevante Distanz einer

Begegnungssituation ausgewählt werden, standardmäßig ist 3704 Meter (2 Seemeilen) eingestellt. Beim *MinDistanceFilter* kann auch noch das relevante Zeitintervall der *Tracks* rund um den CPA in Sekunden eingestellt werden, standardmäßig sind es 900 Sekunden (15 Minuten). Exemplarisch werden die sieben vorgestellten Bedienungsmöglichkeiten am *MinDistanceFilter* in Abbildung 10-22 gezeigt.

Unter dem Metaknoten *Detect overtaking, following, head-on, passing, crossing* können unter dem Metaknoten *filtering out overtaking/following, filtering out head-on/passing* und *filtering out crossing* die Schwellenwerte für die Durchschnittswahrscheinlichkeiten der einzelnen Begegnungssituationen eingestellt werden. Beispielsweise können beim Metaknoten *filtering out head-on/passing* gezeigt die Schwellenwerte bei den Knoten *head-on with bearing* und *head-on with heading* geändert werden.

Ferner kann es sinnvoll sein die Tabelle *statistic* (Kapitel 7.2) um weitere Parameter zu erweitern. Wenn dies gemacht wird, können in dem *Encounterworkflow* im Metaknoten *statistics* (Abbildung 10-22) noch weitere Parameter hinzugefügt werden, dort kann als Input entweder der Knoten mit der Beschreibung *get only rows at the cpa* oder der Knoten mit der Beschreibung *get only one cpa row* genommen werden, je nachdem, ob die dynamischen Informationen von beiden Schiffen am CPA oder nur der Zeitstempel des Szenarios gebraucht wird.

Wichtig ist, dass vor dem Joinen mit den bisherigen Parametern der Name des neuen Parameters konsistent zum Namen der jeweiligen Spalte in der Tabelle *statistic* gehalten wird, hierzu kann der Knoten *Column Rename* genutzt werden.

Sehr wichtig beim Einstellen des Workflows ist das richtige Einrichten der Workflowvariablen (Abbildung 10-23).

Name	Type	Value
database_host	■ STRING	emir-p...
database_port	■ INTEGER	5432
database_name	■ STRING	Magell...
rules_geojson_file	■ STRING	/Users...
seachart_gates_geojson	■ STRING	/Users...
seachart_geo_geojson	■ STRING	/Users...
seachart_filter_geojson	■ STRING	/Users...

Abbildung 10-23: Einstellen der Workflowvariablen

Durch die Parameter *database_host*, *database_port* und *database_name* kann die zu benutzende Datenbank ausgewählt werden. Durch die Parameter *rules_geojson_file*, *seachart_gates_geojson*, *seachart_geo_geojson* und *seachart_filter_geojson* können die Pfade zu den jeweiligen geojson-Dateien geändert werden, welche standardmäßig in den Projekten Rules und emirFilter abgespeichert sind. Die Version auf KNIME-Server beinhaltet bereits eingestellte Pfade zu den Geojson-Dateien, sodass diese normalerweise nicht geändert werden müssen.

Zum abschließenden Klassifizieren der Begegnungssituationen gibt es einen Workflow mit dem Namen *EncounterClassificationWorkflow*.

Hier kann beim *DB Query Reader* mit der Beschreibung *get statistics from scenarios* der Zeitraum eingestellt werden, welcher bestimmt, welche Szenarien bei der Klassifikation berücksichtigt werden sollen.

Im Metaknoten *loops* wird für jeden berücksichtigten Parameter ein eigener Knoten *Group Loop Start* erstellt, dadurch werden alle möglichen, vorhandenen Kombinationen der Parameter der Szenarien erstellt. Soll somit ein neuer Parameter eingeflossen werden, muss u.a. dort ein neuer *Group Loop Start* erstellt werden.

Eingetragen werden die gebildeten Klassen in der Tabelle *classification_with_weather*. Welche Parameter genau berücksichtigt werden, kann in (Verweis auf Kapitel 7.12) nachgelesen werden. Des Weiteren gibt es in der Tabelle *classification_with_weather* eine id mit dem Verweis auf die Tabelle *density_function*, in der Tabelle sind die Koeffizienten der Regressionsfunktion

gespeichert, welche die jeweilige Kerndichtschätzung der minimalen Distanzen der Begegnungssituationen einer Klasse abbildet.

Ferner kann über die Workflowvariablen *min_probability_bearing* und *min_probability_heading* eingestellt werden, wie groß die Schwellenwerte für die Wahrscheinlichkeiten sein müssen für die beiden Verrechnungsweisen. Als Standardeinstellung ist bei *min_probability_bearing* 0.5 und bei *min_probability_heading* 0.4 eingestellt, wie auch initial überhaupt erst Begegnungssituationen gebildet wurden.

Zur Visualisierung einzelner Kerndichtschätzungen gibt es einen Workflow mit dem Namen *VisualizeKDEGraphWorkflow*. Zunächst sollte bei dem *DB Query Reader get statistics from scenarios* der Zeitraum ausgewählt werden. Des Weiteren sollte das *JavaSnippet choose classification paramters* bearbeitet werden, sodass die Werte der Parameter den Werten der Parameter gewünschten, zu betrachtenden Klasse entsprechen. Zuletzt kann beim letzten Knoten des Workflows, welcher ein *Line Plot (JavaScript)* ist, der Graph der Kerndichtschätzung betrachtet werden.

Zum Betrachten von Statistiken gibt es einen Workflow mit dem Namen *EncounterStatisticsWorkflow*.

Unter dem *DB Query Reader get statistics from scenarios* kann der Zeitraum bearbeitet werden, zu welchem Statistiken erstellt werden sollen. Bei den Knoten mit der Beschreibung *percentages* kann die prozentuale Verteilung einzelner Parameter angeschaut werden, wie beispielsweise die prozentuale Verteilung eingehaltener Regeln. Es gibt hier auch wieder die beiden Workflowvariablen *min_probability_bearing* und *min_probability_heading* zum Einstellen der Schwellenwerte für die Wahrscheinlichkeiten.

Zuletzt können in einem Workflow mit dem Namen *EncounterVisualizationWorkflow* (Abbildung 10-24) die in der Datenbank gespeicherten Begegnungssituationen angeschaut werden.

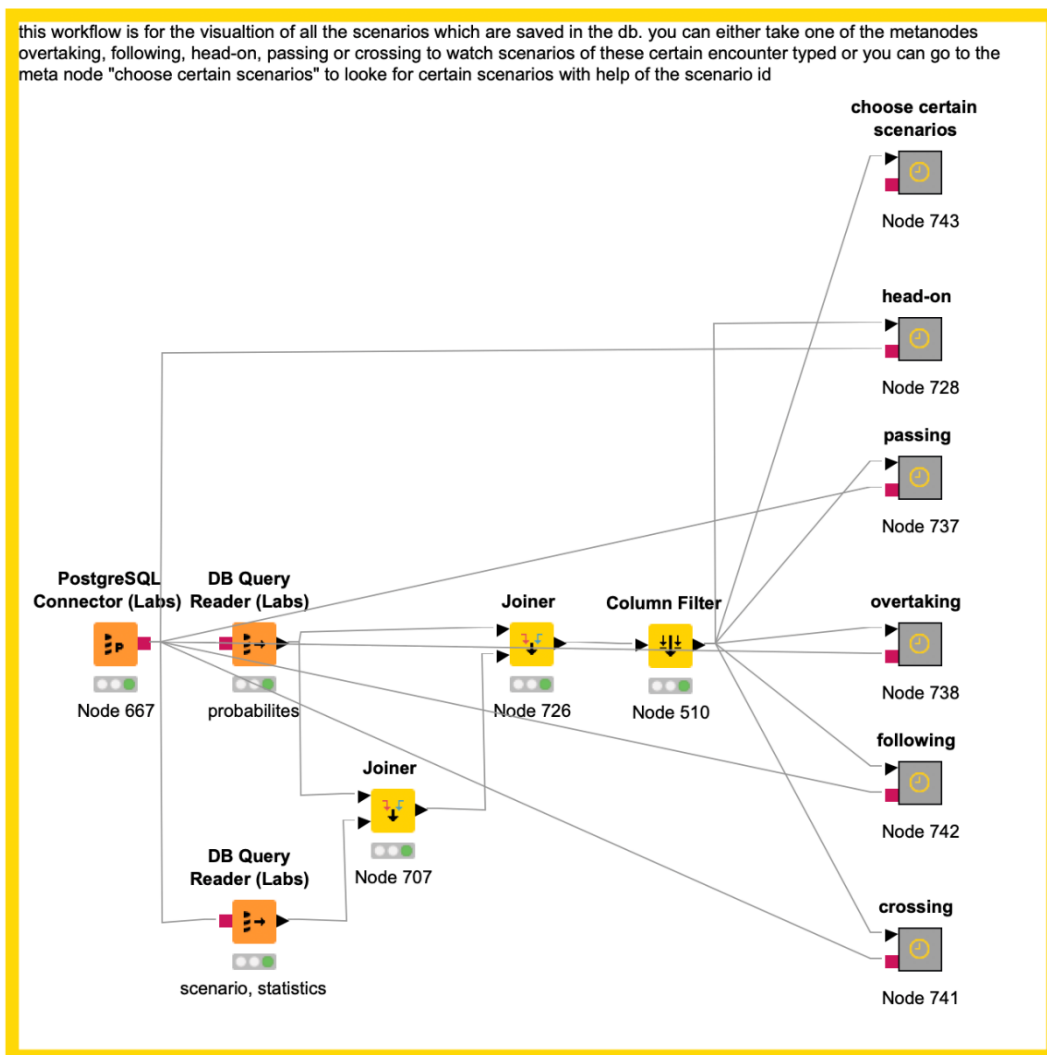


Abbildung 10-24: EncounterVisualizationWorkflow

Entweder wird in die Metaknoten *head-on*, *passing*, *overtaking*, *following* oder *crossing* gegangen und es werden sich Begegnungssituationen dieser Begegnungstypen angeschaut oder es wird in den Metaknote *choose certain scenarios* gegangen, um sich bestimmte Szenarien mithilfe der Auswahl einer oder mehrerer Szenario-IDs anzuschauen. Hierzu wird innerhalb des Metaknotens *choose certain scenarios* der *JavaSnippet Row Filter* mit dem Namen *certain scenarios* bearbeitet, sodass dort die jeweiligen Szenario-IDs der gewünschten Szenarien als Bedingung hinzugefügt werden. Bei den anderen Metaknoten sollte der *JavaSnippet Row Filter* mit dem Namen *certain scenarios* bearbeitet werden. Hier wird die Anzahl der Szenarien ausgewählt, zu welchen die dynamischen

Daten abgefragt werden soll, standardmäßig ist die Zahl bei 20. Je höher diese Zahl ist, desto länger dauert die Durchführung des Prozesses, da die Abfrage nach den dynamischen Daten etwas Zeit in Anspruch nimmt. Um sich nun die Szenarien anschauen zu können, gibt es in jedem der vorher vorgestellten Metaknoten wiederum einen Metaknoten *Visualization Scenarios*. Dort gibt es zum einen den Knoten *OSM Map View* zum Betrachten von den einzelnen *Trackpoints* der *Tracks* einer Begegnungssituation, welche miteinander verbunden sind. Zum anderen gibt es den Knoten *ScenarioViewer*, mit welchem die Begegnungssituation in bewegten Bildern nachgespielt werden.

In diesem Kapitel wurde vorgestellt, wie Begegnungssituationen zwischen zwei Schiffen aus der gegebenen Datenmenge erkannt wurden und wie bestimmt wurde, um was für einen Begegnungstyp es sich wahrscheinlich handelt. Ferner wurden die Begegnungssituationen durch verschiedene Parameter klassifiziert und für jede gebildete Klasse mithilfe einer Kerndichtschätzung eine kritische Distanz bestimmt. Diese kritische Distanz ist ein weiteres Maß, um anomale Begegnungssituationen zu erkennen und deren Risiko einschätzen zu können.

10.4.3 Erstellung eines Manövernetzes

Der Workflow „Maneuvernet Creator“ erstellt basierend auf einer beliebigen Menge an Daten ein Manövernetz (vgl. Kapitel 7.15), welches als CSV und GeoJson exportiert werden kann.

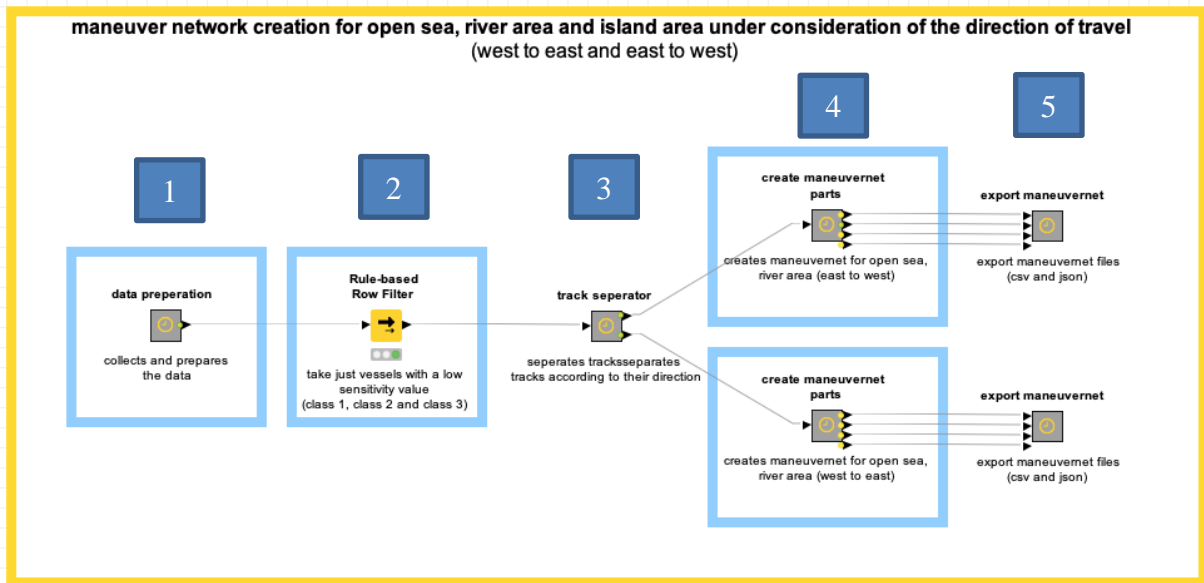


Abbildung 10-25: Workflow zur Generierung eines Manövernetzes – Maneuverneter Creator

Abbildung 10-25 zeigt den Workflow, der ein Manövernetz generiert. Die blauen Kästen symbolisieren dem Benutzer, dass der Knoten von ihm weiter eingestellt werden kann.

Zu Beginn werden im ersten Metaknoten alle benötigten Daten eingelesen auf deren Basis das Manövernetz erstellt werden soll. Dies können Daten einer bestimmten Region, zu einem bestimmten Zeitpunkt oder generell alle verfügbaren Daten sein.

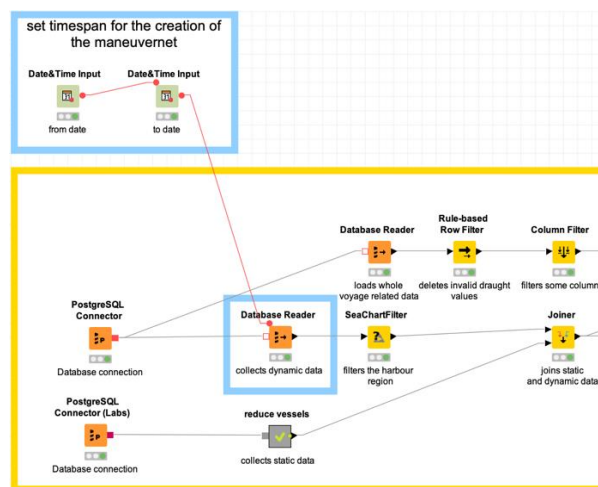


Abbildung 10-26: Ausschnitt aus dem Metaknoten Data preparation

Abbildung 10-26 zeigt einen Ausschnitt aus dem ersten Metaknoten. Für den Benutzer sind lediglich die blau markierten Knoten relevant. Zum einen kann in dem oberen blauen Kasten der Zeitraum eingestellt werden, für den das Manövernetz erstellt werden soll. Dadurch werden alle dynamischen Daten, die für den angegebenen Zeitpunkt vorhanden sind, vom blauen Database Reader abgefragt. Sollte Bedarf bestehen die Datenbankanfrage zu verändern, kann dies an dieser Stelle gemacht werden.

Beim zweiten Knoten können die gesammelten Daten anhand von Schiffsklassen (vgl. Kapitel 7.10) gefiltert werden. Soll beispielsweise ein Manövernetz für nur schlecht manövrierfähige Schiffe generiert werden, so können Schiffe mit einer hohen Manövrierbarkeit herausgefiltert werden.

Anschließend wird zwischen Schiffen differenziert, die von Westen nach Osten und von Osten nach Westen fahren. Dies ermöglicht es beispielsweise Manövernetze für lediglich aus der Elbe ein- bzw. ausfahrende Schiffe zu kreieren. Wird diese Differenzierung nicht benötigt, kann dieser Knoten gelöscht werden. Ab diesem Zeitpunkt ist der Workflow zweigeteilt, da sowohl ein Manövernetz für einfahrende als auch für ausfahrende Schiffe erstellt wird.

Beim vierten Metaknoten kann der Benutzer durch die Parametrisierung des Clustering beeinflussen, wie streng das Manövernetz bei der Bewertung von anomalen Manövern ist. Welche Auswirkungen die Wahl von *Slack* und *MinPoints* vom DBSCAN Clustering auf das Manövernetz haben wurde bereits in Kapitel 7.15.2.1 beleuchtet.

Abschließend wird das Manövernetz als GeoJson und als CSV exportiert. Dabei werden eine GeoJson und zwei CSV Dateien erstellt. Die GeoJson Datei enthält dabei alle Clustermittelpunkte und deren Verbindungen zueinander. Die Datei kann zur einfachen Visualisierung des Manövernetzes mit Hilfe von *geojson.io* genutzt werden. Eine der beiden CSV Dateien enthält ebenfalls, wie die GeoJson, die Clustermittelpunkte mit den zugehörigen Verbindungen. Zusätzlich verfügt die CSV aber auch über die Intensität, die angibt wie häufig eine Verbindung vom Manövernetz befahren wurde. Die zweite CSV hingegen enthält alle

Manöverpunkte mit dazugehörigen Clustern. Die beiden CSV Dateien werden für die Verwendung des Workflows „Anomal Route Detection“ benötigt.

10.4.4 Benutzung eines Manövernetzes

Der Workflow „Anomal Route Detection“ extrahiert auf Basis eines, durch den Workflow „Maneuvernet Creator“ zuvor erstellten, Manövernetzes anomale Manöver und *Tracks* aus einer Menge von Daten.

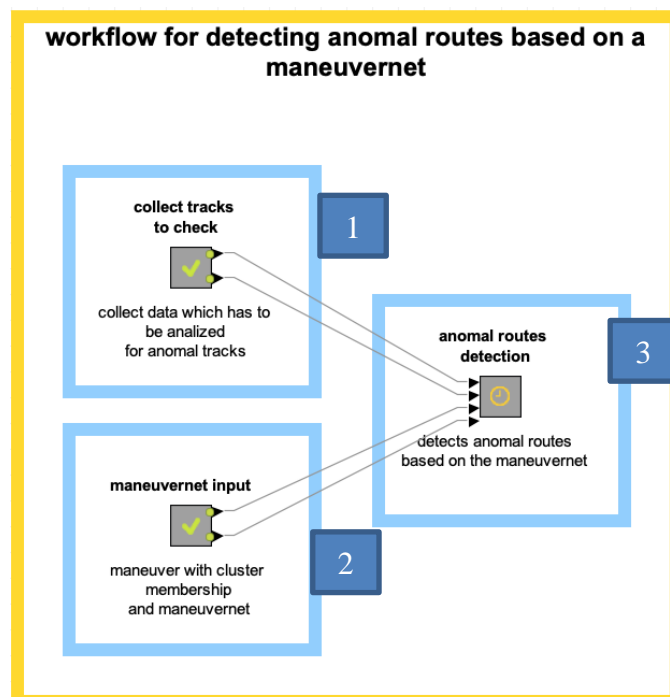


Abbildung 10-27: Workflow zur Detektion von Anomalien - Maneuvernet Creator

Abbildung 10-27 zeigt den dreiteiligen Aufbau des Workflows zur Detektion von Anomalien. Auch in diesem Workflow suggerieren die blauen Kästen, die Möglichkeit den entsprechenden Knoten selbst zu parametrisieren.

Der erste Teil des Workflows „collect tracks to check“ sammelt und transformiert zunächst dynamische Daten, an denen die Analyse auf Anomalien durchgeführt werden soll. Analog zur Abbildung 10-25 kann hier zunächst bestimmt werden, aus welchem Zeitraum die Daten zum Analysieren stammen sollen. Zusätzlich kann hier aber auch der CuSum Knoten auf seine Sensitivität eingestellt werden.

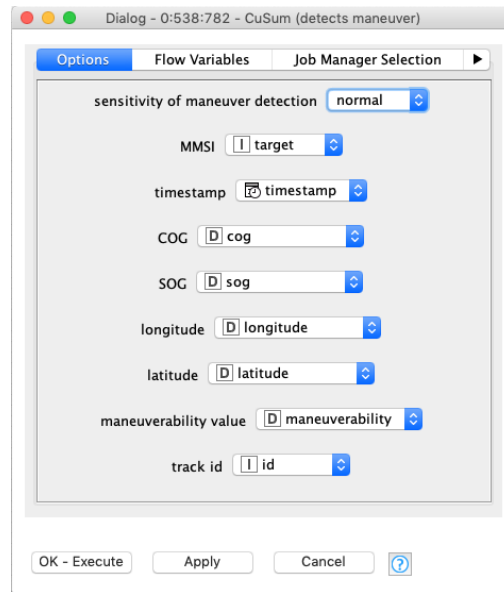


Abbildung 10-28: Dialog des CuSum Knotens

In Abbildung 10-28 ist das Dialog Fenster des CuSum Knotens zu sehen. Neben einigen erforderlichen Daten zum Detektieren von Manövern, wie der MMSI und dem Zeitstempel, kann direkt als erster Parameter die Sensitivität eingestellt werden, mit der CuSum die Manöver detektiert. Dabei gibt es drei Stufen mit denen bestimmt werden kann, wie stark die detektierten Manöver sein sollen. Die Anzahl der detektierten Manöver nimmt somit mit abnehmender Sensitivität ab. Für die meisten Fälle ist die mittlere Parametrisierung „normal“ ausreichend. Sie liefert die charakteristischen Manöver einer Route. Sollen hingegen mehr Manöver oder nur starke Manöver detektiert werden, bietet sich auch eine der beiden anderen Optionen an.

Mit dem zweiten Teil des Workflows wird das Manövernetz eingelesen (vgl. Abbildung 10-29)

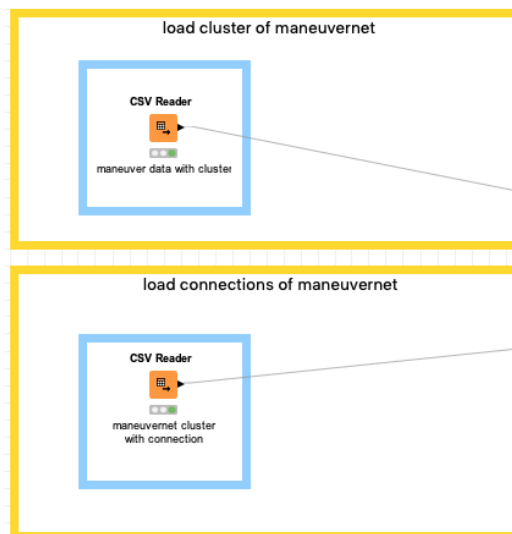


Abbildung 10-29: Metaknoten zum Laden des Manövernetzes

Hier werden die beiden CSV Dateien, die durch den Workflow „Maneurnet Creator“ erstellt worden sind, eingelesen. Hierfür wird für den oberen CSV Reader die CSV mit den Manöverpunkten und dazugehörigen Clustern ausgewählt. Der untere CSV Reader erwartet hingegen die CSV mit dem einzelnen Cluster und Verbindungen.

Wurden die Daten, auf denen die Analyse stattfinden soll, aufbereitet und das Manövernetz eingelesen, beginnt der dritte Metaknoten mit der tatsächlichen Anomalie Detektion. Die Ausführung kann einige Zeit in Anspruch nehmen, da jeder einzelne Manöverpunkt analysiert wird. Somit nimmt die Dauer der Analyse auch mit einer steigenden Anzahl an Manövern zu. Grundsätzlich detektiert der Workflow drei verschiedene Arten von Anomalien.

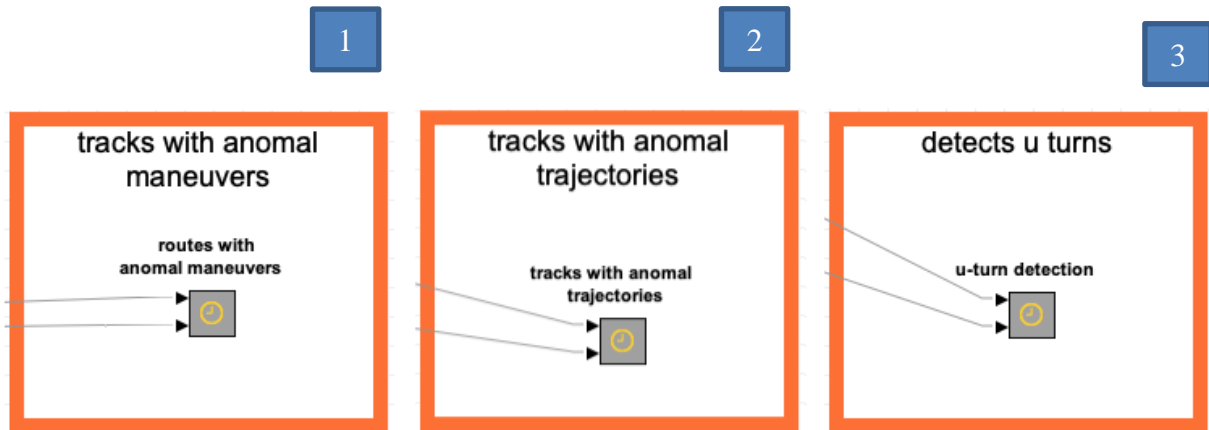


Abbildung 10-30: Output des “Anomal Route Detection” Workflows

Zum einen wären dies die einfachen anomalen Manöver, die keinem Cluster des Manövernetzes zugeordnet werden konnten (vgl. Abbildung 10-30, 1. Kasten). Die zweite Art von Anomalie, die erkannt wird, sind die anomalen *Tracks*, bei denen Trajektorien gefahren wurden, die nicht in den Verbindungen vom Manövernetz enthalten sind (vgl. Abbildung 10-30, 2. Kasten). Als letztes werden auch jegliche U-Turns von Schiffen detektiert (vgl. Abbildung 10-30, 3. Kasten), die nicht zwangsweise Anomalien sein müssen. Auf Basis des Manövernetzes lassen sich aber auch diese *Tracks* zuverlässig aus den Daten extrahieren. Alle Ergebnisse können im jeweiligen Meta-Knoten eingesehen werden und als CSV exportiert werden.

10.5 Anforderungen

10.5.1 Allgemeine Anforderungen

Die Anforderungen mit dem Kürzel AA sind Allgemeine Anforderungen an die gesamte Projektgruppe.

ID	AA1000
Beschreibung	Die erstellten Programme müssen getestet werden.

Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Workflows enthalten Test-Knoten (Für Knime Tests). • Die Test-Knoten-Ausführung ist erfolgreich (Für Knime Tests). • Der Java-Code enthält JUnit Klasse für jeden Knoten (für Java Tests). • Die Code Abdeckung nach einer erfolgreichen Test-Ausführung (alte und neue Tests) ist $\geq 80\%$ (für Java-Tests).
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	AA1001
Beschreibung	Die Projektgruppe muss eine Softwaredokumentation bereitstellen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Dokumentation muss in Englisch verfasst sein. • Die wichtigsten Programmfunktionen sind dokumentiert.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	<p>Unter den wichtigsten Programmfunktionen werden alle Funktionen verstanden, die für die vollständigen Durchlauf des gesamten Prozesses, d.h. Datensammlung, -aufbereitung und -auswertung, benötigt werden.</p> <p>Unter einem KNIME-Workflow wird eine aufeinanderfolgende mehrerer KNIME-Knoten (kleinste Verarbeitungseinheit/führt eine einzelne Aufgabe durch) verstanden. Durch die Kombination mehrerer Knoten entsteht also ein Prozess.</p>
ID	AA1002

Beschreibung	Die Projektgruppe muss für jede Programmiersprache eine einheitliche Codekonvention bereitstellen.
Akzeptanzkriterien	<ul style="list-style-type: none"> Für jede genutzte Programmiersprache wurde ein Dokument erstellt, in welchem die Codekonvention beschrieben ist.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	AA1003
Beschreibung	Das Programm muss erweiterbar sein.
Akzeptanzkriterien	<ul style="list-style-type: none"> Für den Entwurf eines Programms wurden Entwurfsmuster genutzt. Die erstellten Programme wurden als Plugins erstellt. In der Software-Dokumentation wurde beschrieben, wie das Programm erweitert werden kann.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	<p>Bei dem Programm handelt es sich um eine Kombination aus mehreren KNIME-Workflows und Java/Python-Anwendungen. Die Funktionalität von KNIME wird dabei durch selbst erstellte Knoten erweitert.</p> <p>Ein Programm heißt erweiterbar, wenn es ohne übergroßen Aufwand an andere Situationen angepasst werden kann.</p>
ID	AA1004
Beschreibung	Das Programm muss eine maximale Verfügbarkeit aufweisen.

Akzeptanzkriterien	<ul style="list-style-type: none"> • Bei Programmabstürzen müssen sinnvolle Logs bereitstehen. • Für jedes Programm wurden Tests geschrieben. => Tests verworfen • Das Programm muss 90% des Jahres verfügbar sein.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	Die Programme der Projektgruppe sollen über eine lange Zeit Datenanalysen vollziehen. Für diesen Zweck müssen die erstellten Programme fehlerfrei sein maximal Verfügbar sein. Test sollen dabei Fehlerquellen aufdecken, welche die Verfügbarkeit beeinflussen. Bei dem Programm handelt es sich um eine Kombination aus mehreren KNIME-Workflows. Die Funktionalität von KNIME wird dabei durch selbst erstellte Knoten erweitert. Aus den Logs muss hervorgehen, wann, warum und in welchem Zustand (z.B. aktueller Fortschritt) das Programm abgestürzt ist.
ID	AA1005
Beschreibung	Das entwickelte Programm muss die Möglichkeit bieten unter Windows 10 lauffähig zu sein.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Sämtliche implementierte Programmelemente laufen unter Windows 10.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	<p>Der Großteil der Offis Infrastruktur basiert auf Windows 10, dies beinhaltet unter anderem Server und PCs. Auf dem Server wird das Programm ausgeführt und auf den Rechnern soll das Programm erweitert werden können.</p> <p>Bei dem Programm handelt es sich um eine Kombination aus mehreren KNIME-Workflows und Java-/Python-Anwendungen. Die Funktionalität von KNIME wird dabei durch selbst erstellte Knoten erweitert.</p>

ID	AA1008
Beschreibung	Die Projektgruppe muss wichtige Softwarebausteine mit Hilfe der UML bzw. anderen Visualisierungsmöglichkeiten beschreiben.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Diagramme wurden vor der Implementierung erstellt. • Diagramme wurden mit Enterprise Architect erstellt.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	Ablaufdiagramme/Aktivitätsdiagramme für die Workflows zum Beispiel
ID	AA1009
Beschreibung	Die Projektgruppe muss Eclipse, JetBrains, Netbeans oder Visual Studio als IDE verwenden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurde keine andere IDE verwendet. • Es wurde Eclipse Indigo verwendet • Es wurde JetBrains 2017/2018 verwendet • Es wurde Visual Studio 10 verwendet
Status	ERLEDIGT
Priorität	hoch
Bemerkung	
ID	AA1010

Beschreibung	Die Projektgruppe muss für die Programmiersprache Java, Maven als Build-Tool verwenden.	
Akzeptanzkriterien	<ul style="list-style-type: none"> • Jedes Programm enthält eine Maven Pom, durch welches eine Jar erstellt werden kann. • Die erstellten Programme können exportiert werden (Eclipse Repository) (nachträglich ergänzt) 	
Status	ERLEDIGT	
Priorität	hoch	
Bemerkung	Die in Java geschriebenen Projekte der Abteilung nutzen Maven. Dies hat den Grund, dass ein Jenkins Server alle im Git Repository befindlichen Projekte baut und diese in einem Snapshot-Repository deployed. Damit die Programme der PG in diese Struktur integriert werden können muss Maven genutzt werden.	
ID	AA1011	
Beschreibung	Die Projektgruppe muss für die Versionskontrolle Git verwenden.	
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurde so wenig wie möglich lokal gearbeitet. • Jeder Commit hat einen sprechenden Titel/Beschreibung. • Es wurden englische Titel/Beschreibungen verwendet. 	
Status	ERLEDIGT	
Priorität	Hoch	
Bemerkung	-----	
ID	AA1012	

Beschreibung	Die Projektgruppe darf das Programm lediglich in Java, Python und C++ entwickeln.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der Kern der Anwendung wurde in Java geschrieben. • C++ wurde nur von Data Collection verwendet. • Es muss eine plausible Argumentation vorliegen, wenn Python oder C++ verwendet wird. • Python Version 3.x • Java Version 1.8
Status	ERLEDIGT
Priorität	hoch
Bemerkung	Bei dem Programm handelt es sich um eine Kombination aus mehreren KNIME-Workflows und Java-/Python-Anwendungen. Die Funktionalität von KNIME wird dabei durch selbst erstellte Knoten erweitert.

10.5.2 Allgemeine Anforderungen (GA)

ID	GA1000
Beschreibung	Für jedes genutzte Tool muss es eine Bedienungsanleitung geben
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurde beschrieben, wie der Prozess gestartet werden kann • Es wurde beschrieben, wie man den aktuellen Fortschritt des Prozesses erhält • Es wurde beschrieben, wie man die Fehler des erstellten Prozesses erhält. • Es wurde beschrieben, wie der Prozess manuell terminiert werden kann. • Es wurde beschrieben, welcher Input vor Beginn des Prozesses benötigt wird • Es wurde beschrieben, welcher Output am Ende des Prozesses erwartet wird
Status	OFFEN

Priorität	Hoch
Bemerkung	Bei dem Prozess handelt es sich um eine Kombination aus mehreren KNIME-Workflows und Java-/Python-Anwendungen. Die Funktionalität von KNIME wird dabei durch selbst erstellte Knoten erweitert. Der Prozess ist so gestaltet, dass er beim Start eigenständig die Teilprozesse anstößt.
ID	GA1001
Beschreibung	Eine Erweiterung innerhalb eines Tools muss in den gewählten Programmiersprachen Java und Python geschrieben werden
Akzeptanzkriterien	<ul style="list-style-type: none"> • Alle Erweiterungen wurden in Java oder Python geschrieben
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Die Erweiterungen in Form von KNIME-Knoten werden in Java oder Python geschrieben.

10.5.3 Anforderungen an den Hauptprozess

Damit alle erstellten Prozesse der Projektgruppe MaGellan gesteuert und synchronisiert werden, wird ein Hauptprozess erstellt. Der Hauptprozess stellt die übergeordnete Ebene der Teilprozesse dar und wird diese steuern. Die Steuerung beinhaltet das Starten, Stoppen und die Synchronisation der Teilprozesse. Daher stellt der Hauptprozess den Ablauf dar, in welchem die Teilprozesse gestartet werden. Beispielsweise müssen die Wetterdaten aktualisiert werden, bevor ein Teilprozess gestartet wird, welcher die Wetterdaten benötigt. Die Synchronisation beinhaltet zudem das Starten von Prozessen in festgelegten Zeitintervallen, sodass ein Prozess beispielsweise einmal pro Woche gestartet wird.

ID	GA2000
Beschreibung	Das Projektteam muss einen KNIME-Workflow schreiben, welcher die erstellten Abläufe der Teilgruppen steuert
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es können Wetterdaten via KNIME-Workflow eingetragen und geupdatet werden (Data Collection). • Es können Kartendaten via KNIME-Workflow eingetragen und geupdatet werden (Data Collection). • Es können Rohdaten über einen KNIME-ETL-Prozess in ein Datenbankschema übertragen werden (Data Processing). • Es können AIS- und Radar-Daten über einen KNIME-Prozess fusioniert und in ein Datenschema für Scenario Mining übertragen werden (Data Processing). • Es wurde ein Netz aus geclusterten Manöverpunkten über einen KNIME-Workflow erstellt (Scenario Mining). • Es besteht die Möglichkeit Schiffsverhalten über einen KNIME-Workflow zu clustern (Scenario Mining). • Es besteht die Möglichkeit unterschiedliche Schiffbegegnungssituationen über einen KNIME-Workflow zu klassifizieren (Scenario Mining). • Es können Risikolevel für anomale Manöverpunkte, über einen KNIME-Workflow zugeordnet werden. • Es können Risikolevel für anomales Schiffsverhalten über einen KNIME-Workflow zugeordnet werden. • Es können Risikolevel für anomale Begegnungssituationen über einen KNIME-Workflow zugeordnet werden. • Es können Risikolevel für Extraktion von Beinahe-Kollisionen über einen KNIME-Workflow zugeordnet werden.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	GA2001

Beschreibung	Der Prozess des Scenario Mining benötigt bereinigte, historische Daten für das Training der Algorithmen zur Szenarioextrahierung, welche den Zeitraum von den letzten 24 Monate, angefangen 01.01.2016, abbilden
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es stehen AIS-Daten (statisch, dynamisch, reisebezogen), die einen Zeitraum von 24 Monaten abdecken, zur Verfügung. • Es stehen Radar-Daten, die einen Zeitraum von 24 Monaten abdecken, zur Verfügung. • Es stehen Wetterdaten, die einen Zeitraum von 24 M. abdecken, zur Verfügung. => Wind & Wellen, Sichtweite sind vorhanden beim Rest dauert das Eintragen zu lange • Es stehen Kartendaten der Nord- und Ostsee sowie der Flüsse Norddeutschlands, zur Verfügung.
Status	OFFEN
Priorität	Hoch
Bemerkung	<p>Es werden 2 Jahre Daten genommen, damit lernende Algorithmen nicht negativ von einmaligen (einmal in mehreren Jahren) Events beeinflusst werden.</p> <p>Als Beispiel können Baustellen genannt werden, welche bestimmte Wasserstraßen sperren.</p>
ID	GA2002
Beschreibung	Der Hauptprozess muss mit KNIME realisiert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der Hauptprozess wurde mit KNIME realisiert => Es gibt keinen Hauptprozess sondern mehrere einzelne Prozesse
Status	OFFEN
Priorität	Hoch

Bemerkung	
ID	GA2003
Beschreibung	Der Hauptprozess muss die Teilprozesse steuern.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der Hauptprozess steuert die Ausführung der Teilprozesse, unter Berücksichtigung der Abhängigkeiten, in einer festgelegten Reihenfolge. • Der Hauptprozess stößt in festgelegten Zeitintervallen die Teilprozesse an • Der Hauptprozess soll die Teilprozesse überwachen und Fehler abfangen (Try/Catch) • Bei dem Abfangen eines Fehlers wird eine automatisierte Email mit der Fehlerursache an den Anwender gesendet • Bei dem Abfangen eines Fehlers müssen die Informationen über die Fehlerursache geloggt werden. • Falls der Prozess abstürzt, werden die bisherigen transformierten Daten des Prozesses gespeichert. • Je nach Fehlerart kann der Prozess neu gestartet werden. • Fehlerinformationen werden teilprozessübergreifend weitergeleitet. • Der Hauptprozess koordiniert und synchronisiert alle untergeordneten Prozesse (Erstellung eines zeitlichen und zusammenhängenden Ablaufs)
Status	OFFEN
Priorität	hoch
Bemerkung	Je nach Fehlerart gibt es die Möglichkeit den Fehler zu beheben. Beispielsweise bei Verbindungsproblemen kann der Prozess weitergeführt werden, sobald die Verbindung wieder hergestellt wurde. Bei schwerwiegenden Fehlern, welche nicht automatisiert behandelt werden können wie eine NullPointerException, terminiert der Prozess. In diesem Fall muss die Fehlerursache manuell durch den Benutzer untersucht und behoben wird und der Prozess erneut gestartet werden. Dabei soll der Prozess, falls es der Fehler ermöglicht, an dem Punkt weitermachen, wo er unterbrochen wurde.

ID	GA2004
Beschreibung	Die Schnittstellen zwischen den verschiedenen Teilprozessen müssen beschrieben werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Für jeden Teilprozess wurden die KNIME-Input-Ports beschrieben • Für jeden Teilprozess wurden die KNIME-Output-Ports beschrieben
Status	OFFEN
Priorität	hoch
Bemerkung	Es gibt eine Schnittstelle zwischen Data Collection und Data Processing und eine weitere Schnittstelle zwischen Data Processing und Scenario Mining. Die Daten werden über die KNIME-Verbindungen zwischen Knoten übertragen.
ID	GA2005
Beschreibung	Übertragene Daten zwischen Teilprozessen müssen auf Korrektheit überprüft werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Für jeden erstellten Teilprozess wurden die eingehenden Spalten beschrieben. • Für jeden erstellten Teilprozess wurden die ausgehenden Spalten beschrieben. • Für jeden erstellten Teilprozess wurden die Spalten überprüft, ob alle benötigten Spalten vorhanden sind.
Status	OFFEN
Priorität	hoch
Bemerkung	Die benötigten Attribute eines Teilprozesses müssen auf Vollständigkeit (alle benötigten Spalten sind vorhanden) überprüft werden. Des Weiteren muss überprüft werden, ob jede Zelle einer Spalte mit einem Wert versehen werden. Ein fehlender Wert wird durch MISSING versehen.

ID	GA2006
Beschreibung	Der Hauptprozess muss automatisiert sein.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der Hauptprozess kann nach einem festgelegten Zeitintervall automatisiert starten • Der Hauptprozess kann sich nach einem festgelegten Zeitintervall beenden
Status	OFFEN
Priorität	hoch
Bemerkung	Die Zeitintervalle sollen durch den Benutzer festlegbar sein.
ID	GA2007
Beschreibung	Der Hauptprozess muss bei Bedarf manuell konfiguriert und gestartet werden können
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurde dokumentiert wie der Hauptprozess konfiguriert und gestartet werden kann
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	GA2008
Beschreibung	Der Hauptprozess muss nach der Ausführung einen Bericht über die Ergebnisse erstellen
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der Bericht wird in ein gängiges Dateiformat konvertiert
Status	OFFEN

Priorität	hoch
Bemerkung	Als gängiges Format wird PDF verwendet. Der Bericht enthält Informationen zu Startzeitpunkt, Endzeitpunkt, Fehler, Status (fertig/nicht fertig), wo die Fehler aufgetreten sind und Fortschritt bei Abbruch.
ID	GA2009
Beschreibung	Der Hauptprozess muss den aktuellen Status der Datenverwertung anzeigen
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es kann an der Oberfläche eingesehen werden, wie weit der Prozess fortgeschritten ist.
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	GA2010
Beschreibung	Der Hauptprozess muss manuell terminiert und fortgesetzt werden können
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der Hauptprozess konnte erfolgreich manuell terminiert werden • Der Hauptprozess konnte danach erfolgreich fortgesetzt werden • Es liegt eine Dokumentation vor wie der Hauptprozess manuell terminiert werden kann • Es liegt eine Dokumentation vor wie der Hauptprozess manuell fortgesetzt werden kann
Status	OFFEN
Priorität	Hoch
Bemerkung	Falls der Prozess terminiert wird, werden die bisherigen transformierten Daten gespeichert und dem Anwender die Möglichkeit gegeben den Prozess an der terminierten Stelle bzw. den terminierten Knoten fortzusetzen.

ID	GA2011
Beschreibung	Nach Terminierung des Hauptprozesses muss der aktuelle Zustand gespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> Falls der Hauptprozess terminiert, wurde der aktuelle Stand bis zum letzten durchlaufenden Knoten gespeichert.
Status	OFFEN
Priorität	Hoch
Bemerkung	Falls der Hauptprozess terminiert, muss der aktuelle Stand bis zum letzten aktiven Knoten gespeichert werden, damit der Workflow nach einem Bugfix weitergeführt werden kann.
ID	GA2012
Beschreibung	Die erstellten Prozesse müssen auf den OFFIS-KNIME-Server integriert werden
Akzeptanzkriterien	<ul style="list-style-type: none"> Die erstellten KNIME-Workflows wurden auf den KNIME-Server hochgeladen Die KNIME-Workflows wurden korrekt konfiguriert Die KNIME-Workflows wurden erfolgreich getestet => Tests verworfen
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Der Hauptprozess soll letztendlich in die Infrastruktur des OFFIS integriert werden und auf einem Server laufen.
ID	GA02013
Beschreibung	Der Hauptprozess muss in der Lage sein, zwei Jahre historischer Daten innerhalb von 48 Stunden zu verarbeiten
Akzeptanzkriterien	<ul style="list-style-type: none"> Nach Anstoß eines Prozesses ist die Dauer der Verarbeitung kleiner als 48 Stunden

Status	VERWORFEN
Priorität	Hoch
Bemerkung	Auf Grund der Größe der zu verarbeitenden Daten ist diese Anforderung unrealistisch.

10.5.4 Anforderungen für die Überführung in die Simulation

ID	GA3001
Beschreibung	Es muss eine Schnittstellenbeschreibung für die Simulation vorliegen
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es liegt ein Dokument mit der Beschreibung der Schnittstelle vor.
Status	OFFEN
Priorität	Niedrig
Bemerkung	
ID	GA3002
Beschreibung	Das extrahierten Szenarien müssen in eine Appdatei, .app oder .bapp, geschrieben werden
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Szenarien werden in ein Application Objekt abgebildet • Das erstellte Objekt wird durch den XMLSerializer in eine .app oder .bapp Datei geschrieben
Status	VERWORFEN
Priorität	Niedrig
Bemerkung	Es wurde sich für .rtz entschieden.
ID	GA3003
Beschreibung	Das extrahierten Szenarien müssen in eine .rtz-Datei, geschrieben werden
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das erstellte Objekt wird in eine .rtz-Datei geschrieben
Status	ERLEDIGT

Priorität	Niedrig
Bemerkung:	

10.5.5 Datenbank-Anforderungen

Die Datenbankanforderungen sind unterteilt in einen allgemeinen Teil, sowie in Anforderungen für die Untergruppen Data Collection und Scenario Mining. Die Teilgruppe Data Processing hat ihre DB Anforderungen (da die gesamten Aufgaben sehr DB nah sind) in ihrer Teilgruppe abgehandelt.

10.5.6 Allgemein

ID	DB001
Beschreibung	Zur Erstellung der Datenbank soll PostgreSQL verwendet werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Alle erstellten Datenbank Schemata wurden in einer PostgreSQL Datenbank eingetragen.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	----- ----- -----
ID	DB002
Beschreibung	Das Plugin PostGis soll verwendet werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden Werte aus der PostGis Extension verwendet
Status	ERLEDIGT
Priorität	hoch
Bemerkung	PostGIS bietet viele Datentypen für Geoinformationen (bspw. Point, Polygon etc.), worauf wiederum Operationen ausgeführt werden können. Diese Datentypen sollen verwendet werden, um deren Vorteile nutzen zu können.

ID	DB003
Beschreibung	Das Plugin PostGis soll die Speicherung der Daten in Rasterformat ermöglichen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden Daten im Rasterformat erfolgreich in der Datenbank gespeichert
Status	VERWORFEN
Priorität	hoch
Bemerkung	<p>Grund für Verworfen:</p> <p>Die Wetterdaten werden als Polygone und nicht als Rasterdaten abgespeichert.-----</p>
ID	DB004
Beschreibung	Die Integrität der Daten muss gewährleistet sein.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Ein Validierungsprozess zur Überprüfung der Daten wurde eingeführt und ist funktionsfähig.
Status	OFFEN
Priorität	hoch
Bemerkung	----- ----- -----
ID	DB005
Beschreibung	Die Datenbank-Schemas müssen die erste Normalform erreichen.

Akzeptanzkriterien	<ul style="list-style-type: none"> • Alle Datenbank Schemata erreichen die erste Normalform.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	<p>-----</p> <p>-----</p> <p>-----</p>

ID	DB006
Beschreibung	Die Datenbank-Schemas müssen die zweite Normalform erreichen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Alle Datenbank Schemata erreichen die zweite Normalform.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	<p>-----</p> <p>-----</p> <p>-----</p>

ID	DB007
Beschreibung	Die Datenbank-Schemas müssen die dritte Normalform erreichen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Alle Datenbank Schemata erreichen die dritte Normalform.

Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----

ID	DB008
Beschreibung	Das Datenbankschema muss unabhängig von den Daten erweiterbar sein.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Dokumentation zur Erweiterbarkeit wird bereitgestellt.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	DB009
Beschreibung	Ein Administrator kann durch zentrale Kontrolle die Datenbank verwalten.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Ein Administrator hat die nötigen Rechte zur Verwaltung
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----

ID	DB010
Beschreibung	Der Datenbearbeitungsprozess muss wiederholbar sein.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es existiert eine Funktion zum erneuten Ausführen der Bearbeitung
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----

ID	DB011
Beschreibung	Die Wiederholungen des Datenbearbeitungsprozess müssen automatisiert ablaufen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Zur Durchführung müssen keine manuellen Eingaben erfolgen
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----

ID	DB012
-----------	--------------

Beschreibung	Der Datenbearbeitungsprozess muss mindestens alle 24 Stunden durchführbar sein.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der Prozess wurde erfolgreich, ohne manuelle Eingabe, alle 24 Stunden durchgeführt
Status	ERLEDIGT
Priorität	hoch
Bemerkung	

10.5.7 Data Collection

10.5.8 Anforderung an die Datenbank (Wetterdaten)

ID	DBC001
Beschreibung	Es sollen Wetterdaten (DBC003 - DBC007) vom BSH bezogen werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Alle erstellten Datenbank Schemata wurden in einer PostgreSQL Datenbank eingetragen.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC002
Beschreibung	Es sollen Wetterdaten (DBC008 - DBC012) vom DWD bezogen werden.

Akzeptanzkriterien	
Status	OFFEN
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC003
Beschreibung	Es müssen Daten über die Strömung ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC004
Beschreibung	Es müssen Daten über den Wasserstand ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC005

ID	DBC005
Beschreibung	Es müssen Daten über Windgeschwindigkeiten ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC006
Beschreibung	Es müssen Daten über den Wellenhöhen ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC007
Beschreibung	Es müssen Daten über den Salzgehalt und potentielle Wassertemperatur ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	ERLEDIGT
Priorität	hoch

ID	DBC007
Bemerkung	----- ----- -----
ID	DBC008
Beschreibung	Es müssen Daten über die Lufttemperatur ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	VERWORFEN
Priorität	hoch
Bemerkung	Grund für Verworfen: Lufttemperatur ist nicht mehr relevant.
ID	DBC009
Beschreibung	Es müssen Daten über den Niederschlag ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC010
Beschreibung	Es müssen Daten über die Sichtweite ab dem 1.2.2016 vorliegen.

ID	DBC009
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten über die Sichtweite in der Datenbank gespeichert
Status	ERLEDIGT
Priorität	hoch
Bemerkung	
ID	DBC011
Beschreibung	Es müssen Daten über den Bedeckungsgrad ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	OFFEN
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC012
Beschreibung	Es müssen Daten über den Nebel ab dem 1.2.2016 vorliegen.
Akzeptanzkriterien	
Status	OFFEN
Priorität	hoch
Bemerkung	----- ----- -----

ID	DBC013
Beschreibung	In der Datenbank müssen Daten der Strömung abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten der Strömung in der Datenbank eingespeichert.
Status	VERWORFEN
Priorität	hoch
Bemerkung	Grund für Verworfen: Die Daten können on-Demand eingetragen werden.
ID	DBC014
Beschreibung	In der Datenbank müssen Daten des Wasserstandes abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten des Wasserstandes in der Datenbank eingespeichert.
Status	VERWORFEN
Priorität	Hoch
Bemerkung	Grund für Verworfen: Die Daten können on-Demand eingetragen werden.
ID	DBC015
Beschreibung	In der Datenbank müssen Daten über den Windgeschwindigkeiten abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten der Windgeschwindigkeiten in der Datenbank eingespeichert.
Status	ERLEDIGT

ID	DBC015
Priorität	hoch
Bemerkung	
ID	DBC016
Beschreibung	In der Datenbank müssen Daten über Wellenhöhe abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten über den Wellenhöhe in der Datenbank eingespeichert.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC017
Beschreibung	In der Datenbank müssen Daten über Salzgehalt und potentielle Wassertemperatur abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten über den Salzgehalt und potentielle Wassertemperatur in der Datenbank eingespeichert.
Status	VERWORFEN
Priorität	hoch
Bemerkung	Grund für Verworfen: Die Daten können on-Demand eingetragen werden.
ID	DBC018

ID	DBC017
Beschreibung	In der Datenbank müssen Daten über die Lufttemperatur abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten über die Lufttemperatur in der Datenbank eingespeichert.
Status	OFFEN
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC019
Beschreibung	In der Datenbank müssen Daten über den Niederschlag abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten über den Niederschlag in der Datenbank eingespeichert.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC020
Beschreibung	In der Datenbank müssen Daten über den Bedeckungsgrad abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten über den Bedeckungsgrad in der Datenbank eingespeichert.
Status	OFFEN

ID	DBC019
Priorität	Hoch
Bemerkung	----- ----- -----
ID	DBC021
Beschreibung	In der Datenbank müssen Daten über den Nebel abgespeichert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden erfolgreich Daten über den Wasserdampfgehalt in der Datenbank eingespeichert.
Status	OFFEN
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC022
Beschreibung	Die Wetterdaten sollen in eine Datenbank übertragen werden.
Akzeptanzkriterien	
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC023

ID	DBC023
Beschreibung	Das Datenbank-Schema für die Wetterdaten muss alle Informationen der Wetterdaten beinhalten (siehe: DC023 – DC028).
Akzeptanzkriterien	
Status	OFFEN
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC024
Beschreibung	Wenn neue Wetterdaten vorhanden sind, sollen diese in die Datenbank eingetragen werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Neue Wetterdaten können eingetragen werden.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC025
Beschreibung	Wenn neue Wetterdaten vorhanden sind, sollen diese in die Datenbank aktualisiert werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Wetterdaten können aktualisiert werden.
Status	ERLEDIGT
Priorität	hoch

ID	DBC025
Bemerkung	----- ----- -----
ID	DBC026
Beschreibung	Es muss eine PL/pgSQL-Funktion erstellt werden, mit welcher ein Bereich und Zeitpunkt der Wetterdaten abgefragt werden können.
Akzeptanzkriterien	
Status	VERWORFEN
Priorität	hoch
Bemerkung	(vgl. https://www.postgresql.org/docs/10/static/plpgsql.html) ----- ----- Grund für Verworfen: Funktionen sind bereits vorhanden.

10.5.9

Anforderung an ein Datenbank-Tool:

Die folgenden Anforderungen gelten für eine Tool, welches verschiedene Datenformate (in den Anforderungen definiert) einliest und deren Inhalte in eine Datenbank einträgt.

ID	DBC027
-----------	---------------

Beschreibung	Es muss ein Programm erstellt werden, dass Dateiinhalte verarbeiten kann.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Alle erstellten Datenbank Schemata wurden in einer PostgreSQL Datenbank eingetragen.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	----- ----- ----- -----
ID	DBC027.1
Beschreibung	Das Programm muss NetCDF-Daten verarbeiten können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es werden erfolgreich Daten aus einer NetCDF-Datei ausgelesen und in die DB eingetragen.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Das Programm sollte in der Lage sein, NetCDF Dateien einzulesen und die beinhalteten Daten zu verarbeiten. Daher müssen die Daten aus der Datei ausgelesen und in ein Format zu konvertieren werden, welches in das Datenbankschema übertragen werden kann.
ID	DBC027.2
Beschreibung	Das Programm muss GRIB-Daten verarbeiten können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es werden erfolgreich Daten aus einer GRIB-Datei ausgelesen und in die DB eingetragen.

ID	DBC027.2
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Das Programm sollte in der Lage sein, GRIB-Dateien einzulesen und die beinhalteten Daten zu verarbeiten. Daher müssen die Daten aus der Datei ausgelesen und in ein Format zu konvertieren werden, welches in das Datenbankschema übertragen werden kann.
ID	DBC027.3
Beschreibung	Das Programm muss TXT-Daten (in Tabellenform) verarbeiten können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es werden erfolgreich Daten aus einer TXT-Datei ausgelesen und in die DB eingetragen.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Einige Wetterdaten vom DWD sind in TXT-Dateien gespeichert. Allerdings beinhalten diese Dateien Daten im Tabellenformat. Dementsprechend gibt es eine Spalten Beschreibung und Reihen als Daten. Das Programm sollte dementsprechend in der Lage sein diese verarbeiten oder gegebenenfalls diese in ein anderes Format konvertieren zu können.
ID	DBC027.4
Beschreibung	Das Programm muss CSV-Daten verarbeiten können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es werden erfolgreich Daten aus einer CSV-Datei ausgelesen und in die DB eingetragen.
Status	ERLEDIGT
Priorität	hoch

ID	DBC027.4
Bemerkung	Wetterdaten im CSV Format sollen eingelesen und in ein Format konvertiert werden, welches in das Datenbankschema eingefügt werden kann. ----- ----- -----
ID	DBC027.5
Beschreibung	Das Programm muss ZIP-Dateien entpacken können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • ZIP-Daten werden erfolgreich entpackt und die darin enthaltenen Dateien verarbeitet.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Einige Wetterdaten sind in ZIP-Dateien zusammengefügt. Das Programm sollte dementsprechend in der Lage sein diese Dateien zu extrahieren und die extrahierten Daten zu verarbeiten, sofern diese einer der durch DBC026ff spezifizierten Formate ist.
ID	DBC027.6
Beschreibung	Das Programm muss RAR-Dateien entpacken können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • RAR-Daten werden erfolgreich entpackt und die darin enthaltenen Dateien verarbeitet.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Einige Wetterdaten sind in RAR-Dateien zusammengefügt. Das Programm sollte dementsprechend in der Lage sein diese Dateien zu extrahieren und die extrahierten Daten zu verarbeiten, sofern diese einer der durch DBC026ff spezifizierten Formate ist.

ID	DBC028
Beschreibung	Es muss ein Programm erstellt werden, dass Dateiinhalte automatisiert in die Datenbank eintragen kann.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurde ein Programm erstellt, dass Dateiinhalte automatisiert in die Datenbank einträgt.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC028.1
Beschreibung	Das Programm muss NetCDF-Daten in die Datenbank eintragen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Daten der NetCDF-Dateien wurden ohne Informationsverlust in die Datenbank übertragen.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	----- ----- -----
ID	DBC028.2
Beschreibung	Das Programm muss GRIB-Daten in die Datenbank eintragen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Daten der GRIB-Dateien wurden ohne Informationsverlust in die Datenbank übertragen.
Status	ERLEDIGT
Priorität	Hoch

ID	DBC028.1
Bemerkung	----- ----- -----
ID	DBC028.3
Beschreibung	Das Programm muss HDF5-Daten in die Datenbank eintragen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Daten der HDF5-Dateien wurden ohne Informationsverlust in die Datenbank übertragen.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	----- ----- -----
ID	DBC028.4
Beschreibung	Das Programm muss TxT-Daten (in Tabellenform) in die Datenbank eintragen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Daten der TxT-Dateien wurden ohne Informationsverlust in die Datenbank übertragen.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC028.5
Beschreibung	Das Programm muss CSV-Daten in die Datenbank eintragen können.

ID	DBC028.5
Akzeptanzkriterien	<ul style="list-style-type: none"> Die Daten der CSV-Dateien wurden ohne Informationsverlust in die Datenbank übertragen.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----
ID	DBC028.6
Beschreibung	Das Programm muss ZIP-Daten in die Datenbank eintragen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> Daten der ZIP-Dateien wurden entpackt und die beinhalteten Dateien werden verarbeitet.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	----- ----- -----
ID	DBC028.7
Beschreibung	Das Programm muss RAR-Daten in die Datenbank eintragen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> Daten der RAR-Dateien wurden entpackt und die beinhalteten Dateien werden verarbeitet.
Status	ERLEDIGT
Priorität	Hoch

ID	DBC028.7
Bemerkung	----- ----- -----

10.5.10 Scenario Mining

ID	DBA0001
Beschreibung	Es müssen AIS-Daten zur Erkennung einer Schiffssituation und für die Extraktion von Routen prozessiert sein
Akzeptanzkriterien	<ul style="list-style-type: none"> • MMSI steht zur Verfügung • IMO steht zur Verfügung • Schiffstyp steht zur Verfügung • Schiffbreite steht zur Verfügung • Schiffstyp steht zur Verfügung • Schiffspostion steht zur Verfügung • Zeitstempel steht zur Verfügung • HDG steht zur Verfügung • COG steht zur Verfügung • SOG steht zur Verfügung • ROT steht zur Verfügung
Status	ERLEDIGT
Priorität	hoch
Bemerkung	
ID	DBA0002
Beschreibung	Es müssen Radar-Daten zur Erkennung einer Schiffssituation und für die Extraktion von Routen prozessiert sein
Akzeptanzkriterien	<ul style="list-style-type: none"> • Schiffspostion steht zur Verfügung • COG steht zur Verfügung • SOG steht zur Verfügung • Zeitstempel steht zur Verfügung
Status	ERLEDIGT

Priorität	hoch
Bemerkung	
ID	DBA0003
Beschreibung	Es müssen Wetterdaten-Daten zur Erkennung einer Schiffssituation und für die Extraktion von Routen prozessiert sein
Akzeptanzkriterien	<ul style="list-style-type: none"> • Lufttemperatur steht zur Verfügung => Verworfen, da Lufttemperatur irrelevant ist • Windstärke steht zur Verfügung • Seegang steht zur Verfügung • Niederschlag steht zur Verfügung => Steht in der DB und kann genutzt werden, wurde aber nicht verwendet • Auf- und Untergangszeit steht zur Verfügung • Zeitpunkt der Daten steht zur Verfügung • Position der Daten steht zur Verfügung
Status	ERLEDIGT
Priorität	hoch
Bemerkung	

10.5.11 Data Collection: Anforderungen

Die folgenden Anforderungen wurden von den Stakeholdern der Wetterdaten definiert. Diese beziehen sich auf die graphische Abfrage der Wetterdaten durch ein eigenständiges Programm, welches sich mit der Datenbank verbindet und basierend auf den Nutzereingaben die Wetterdaten abfragt. Dieses Programm soll die Möglichkeit bieten Teilmengen der gespeicherten Wetterdaten abzufragen, durch eine visuelle Konfiguration der Abfrage, ohne SQL-Abfragen schreiben zu müssen. Die abgefragten Daten können im Nutzerinterface dargestellt werden und/oder in ein auswählbares Format exportiert werden.

Diese sind als optional anzusehen, je nachdem wie komplex oder zeitaufwendig die Implementierung wird.

10.5.12 Funktionale Anforderungen:

ID	DC012
Beschreibung	Es kann ein Nutzerinterface erstellt werden, mit welchem Wetterdaten aus der Datenbank abgefragt werden können.
Akzeptanzkriterium	<ul style="list-style-type: none"> • Es existiert ein Nutzerinterface zur Abfrage der Daten
Status	OFFEN
Priorität	Mittel
Bemerkung	Damit die Wetterdaten abgefragt werden können, soll ein Nutzerinterface erstellt werden, in welchem ein Bereich und ein Zeitfenster spezifiziert werden kann. Dies soll genutzt werden, um die Wetterdaten aus der Datenbank abzufragen und darzustellen.
ID	DC013

Beschreibung	Im Nutzerinterface muss ein Bereich für die Abfrage der Wetterdaten definiert werden können.
Akzeptanzkriterium	<ul style="list-style-type: none"> Über das Nutzerinterface wurden erfolgreich Wetterdaten abgefragt.
Status	OFFEN
Priorität	Mittel
Bemerkung	
ID	DC014
Beschreibung	Der Bereich der Abfrage muss durch Längen-, Breitengrade und Länge, Breite angegeben werden.
Akzeptanzkriterium	<ul style="list-style-type: none"> Der Bereich wurde erfolgreich visualisiert
Status	OFFEN
Priorität	Mittel
Bemerkung	Die Abfrage kann beispielsweise durch eine Kartendarstellung geschehen, worauf ein Bereich definiert werden kann.
ID	DC015
Beschreibung	Die abgefragten Wetterdaten sollten in dem Nutzerinterface dargestellt werden können.
Akzeptanzkriterium	<ul style="list-style-type: none"> Die abgefragten Daten werden im UI durch Listen/Arrays visualisiert
Status	OFFEN

Priorität	Mittel
Bemerkung	Darstellung der Wetterdaten in dem Programm durch eine Liste/mehrdimensionales Array.
ID	DC016
Beschreibung	Im Nutzerinterface muss spezifiziert werden können, in welchem Zeitfenster die Wetterdaten abgefragt werden sollen.
Akzeptanzkriterium	<ul style="list-style-type: none"> • Es gibt ein Interface mit welchem Nutzer ein Zeitfenster für die Wetterdaten eingeben können.
Status	OFFEN
Priorität	Mittel
Bemerkung	Durch die Auswahl eines Zeitfensters (bspw. vom 1.1.2017 - 31.12.2017) sollen die NutzerInnen einen Teil der Wetterdaten spezifiziert abfragen können.
ID	DC017
Beschreibung	Die abgefragten Wetterdaten sollen auf einer Seekarte dargestellt werden können.
Akzeptanzkriterium	<ul style="list-style-type: none"> • Die abgefragten Daten wurden erfolgreich auf einer Seekarte visualisiert
Status	OFFEN
Priorität	Mittel

Bemerkung	Die abgefragten Daten sollen auf die Kartendarstellung durch eine neue Layer gezeichnet werden.
ID	DC018
Beschreibung	Die abgefragten Wetterdaten sollen in ein auswählbares Format exportiert werden können.
Akzeptanzkriterium	<ul style="list-style-type: none"> • Es existiert eine exportierte Datei im gewünschten Format • Es existiert eine Export-Funktion
Status	OFFEN
Priorität	Mittel
Bemerkung	Nachdem die Daten abgefragt wurden, sollen die NutzerInnen die Möglichkeit haben diese in gewähltes Format zu exportieren.
ID	DC018.1
Beschreibung	Die Wetterdaten sollen in NetCDF exportiert werden können
Akzeptanzkriterium	<ul style="list-style-type: none"> • Es existieren exportierte Daten im NetCDF-Format • NetCDF kann als Exportformat ausgewählt werden
Status	OFFEN
Priorität	Mittel
Bemerkung	Abgefragte Wetterdaten sollen in eine NetCDF Datei verpackt werden und als Datei gespeichert werden können.
ID	DC018.2

Beschreibung	Die Wetterdaten sollen in GRIB exportiert werden können
Akzeptanzkriterium	<ul style="list-style-type: none"> • Es existieren exportierte Daten im GRIB-Format • GRIB kann als Exportformat ausgewählt werden
Status	OFFEN
Priorität	Mittel
Bemerkung	Abgefragte Wetterdaten sollen in eine GRIB Datei verpackt werden und als Datei gespeichert werden können.
ID	DC018.3
Beschreibung	Die Wetterdaten sollen in CSV exportiert werden können.
Akzeptanzkriterium	<ul style="list-style-type: none"> • Es existieren exportierte Daten im CSV-Format • NetCDF kann als Exportformat ausgewählt werden
Status	OFFEN
Priorität	Mittel
Bemerkung	Abgefragte Wetterdaten sollen in eine CSV Datei verpackt werden und als Datei gespeichert werden können.

10.5.13 Nicht Funktionale Anforderungen:

ID	DC019
Beschreibung	Das Nutzerinterface muss in Java-Swing geschrieben werden, um es in die bestehenden Strukturen integrieren zu können.
Akzeptanzkriterium	<ul style="list-style-type: none"> • Als Programmiersprache wurde Java-Swing verwendet

Status	OFFEN
Priorität	Mittel
Bemerkung	Da die meisten vom OFFIS geschriebenen Interfaces in Swing geschrieben wurden, soll das Interface auch in diesem geschrieben werden. Dies geschieht unter dem Hintergrund, dass diese Anforderung die Möglichkeit bietet das Programm in andere integrieren zu können
ID	DC020
Beschreibung	Das Nutzerinterface muss in der Programmiersprache Java geschrieben werden.
Akzeptanzkriterium	<ul style="list-style-type: none"> Als Programmiersprache wurde Java verwendet
Status	OFFEN
Priorität	Mittel
Bemerkung	
ID	DC021
Beschreibung	Das Programm soll eine Standalone-Variante sein.
Akzeptanzkriterium	<ul style="list-style-type: none"> Das Programm kann unabhängig von Software und Hardware operieren
Status	OFFEN
Priorität	Mittel

Bemerkung	Die Stakeholder haben sich dafür entschieden, dass das Programm eigenständig sein soll. Daher soll es in erster Linie gestartet und benutzt werden können, ohne dies als Plugin oder ähnliches Laden zu müssen
-----------	--

10.5.14 Anforderungen Data Processing

10.5.14.1 Datenaufbereitung

Unter dem Schritt der Datenaufbereitung versteht die Projektgruppe die Extraktion der Rohdaten (*nbdatabase*), die erste Verarbeitung der Daten, sowie das abspeichern in einem optimierten Datenmodell. Das Ziel der Datenaufbereitung ist es, dass in einem optimierten Datenmodell die Rohdaten abgelegt werden und offensichtliche Fehler einsehbar sind.

10.5.15 Anforderungen

ID	DP1000
Beschreibung	Das System muss fähig sein, Rohdaten, die in die Datenbank <i>nbdatabase</i> geschrieben werden in einem gewählten Intervall zu verarbeiten.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Rohdaten werden in festen Zeiten (alle x Stunden) oder festen Datenanzahl (alle y Datensätze) abgerufen • Die Rohdaten werden spätestens alle 12 Stunden abgerufen und verarbeitet • Die Rohdaten werden in ein Tool zur Verarbeitung geladen und in ein neues Datenschema geschrieben (siehe Anforderung DP1001)
Status	OFFEN
Priorität	hoch
Bemerkung	<p>-----</p> <p>-----</p> <p>-----</p>
ID	DP1001
Beschreibung	Das System muss ein performantes Datenbankschema für die Speicherung der Rohdaten aufweisen.

Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Datenbankschema kann nach AIS und Radardaten unterscheiden • Das Datenbankschema kann nach statischen, dynamischen und reisebezogenen Daten unterscheiden • Das Datenbankschema kann eine Verbindung zwischen verarbeiteten und Rohdaten herstellen. • Das Datenbankschema ermöglicht die Weiterverarbeitung der Daten
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	----- ----- -----
ID	DP1002
Beschreibung	Das System bietet die Möglichkeit auffällige Rohdaten, welche in einem Konzept definiert wurden, kennzuzeichnen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • <ul style="list-style-type: none"> ○ Das System kann zwischen unterschiedlichen Gründen der Auffälligkeit unterscheiden ○ Das Konzept ermöglicht es, dass mindestens 80% der Auffälligkeiten im ersten Schritt erkannt werden. ○ Das System erlaubt es nachträglich festgestellte Auffälligkeiten zu integrieren. ○ Das System kann zwischen auffälligen Radar und AIS Daten unterscheiden.
Status	OFFEN
Priorität	Hoch
Bemerkung	----- ----- -----
ID	DP1003

Beschreibung	Der ETL Prozess muss fähig sein duplizierte statische Daten auszudünnen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • <ul style="list-style-type: none"> ○ Der ETL Prozess erkennt automatisch Duplikate ○ Der ETL Prozess speichert keine Duplikate im neuen Datenmodell ○ Der ETL Prozess speichert die Anzahl der entfernten Duplikate ab ○ Der ETL Prozess speichert nach Möglichkeit nur ein Vessel. Bei unterschiedlichen Datensätzen zu einem Vessel, werden die Datensätze als auffällig markiert und je ein Datensatz gespeichert.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	
ID	DP1004
Beschreibung	Der ETL Prozess muss fähig sein duplizierte reisebezogene Daten auszudünnen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • <ul style="list-style-type: none"> ○ Der ETL Prozess erkennt automatisch Duplikate ○ Der ETL Prozess speichert die Anzahl der entfernten Duplikate ab ○ Der ETL Prozess speichert für jede Fahrt nur einen Datensatz (Bremerhaven -> Cuxhaven) und verwirft die anderen ○ Der ETL Prozess erkennt Datensätze zur Liegezeit und speichert nur einen Datensatz zur Liegezeit mit der Dauer ab.
Status	OFFEN

Priorität	Hoch
Bemerkung	----- ----- -----
ID	DP1005
Beschreibung	Der ETL Prozess muss fähig sein duplizierte dynamische Daten auszudünnen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der ETL Prozess erkennt automatisch Duplikate • Der ETL Prozess speichert keine Duplikate im neuen Datenmodell • Der ETL Prozess speichert die Anzahl der entfernten Duplikate ab
Status	OFFEN
Priorität	Hoch
Bemerkung	----- ----- -----
ID	DP1006
Beschreibung	Der ETL Prozess muss vorgenommene Bereinigungen nachvollziehbar im neuen Datenmodell abspeichern.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Datenschema zeigt die Anzahl der Fehler auf • Das Datenschema zeigt die Art der Fehler auf • Das Datenschema erlaubt den Zugriff zum Rohdatensatz
Status	ERLEDIGT
Priorität	hoch
Bemerkung	----- ----- -----

10.5.16 Datenanreicherung

Unter dem Schritt der Datenanreicherung versteht die Projektgruppe die Bereinigung des ersten Datenmodells. Das Ziel der Datenanreicherung ist es, die Datenqualität zu erhöhen und die Daten für das Szenario Mining vorzubereiten.

10.5.17 Anforderungen

ID	DP2000
Beschreibung	Die Dokumentation muss die Möglichkeit bieten für jeden Fehlercode einen möglichen Lösungsvorgang bereitzustellen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Datenbank hat eine Liste mit sämtlichen Fehlercodes • Die Dokumentation hat Konzepte mit einem klaren Vorgehen zu den einzelnen Fehlercodes
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	DP2001
Beschreibung	Das Datenmodell muss die Möglichkeit bieten die Daten nach einzelnen Fehlercodes zu filtern.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Datenbank speichert Fehlercodes • Eine Filterung ist möglich, wenn einem Datensatz mehr als ein Fehlercode zugeordnet ist
Status	ERLEDIGT
Priorität	Hoch

Bemerkung	
ID	DP2002
Beschreibung	Statische AIS-Daten müssen aus Werten bestehen, welche die Regeln der einzelnen Fehler beachten.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die MMSI ist bei allen Datensätzen neunstellig - Bei Umstellung auf 10-stellig MMSI anpassen. • Die IMO ist bei allen Datensätzen siebenstellig • Die ReferenceID muss eine der ID's aus dem LabskausDBschema sein.
Status	OFFEN
Priorität	hoch
Bemerkung	Die Anforderung (wie DP2003) soll sicherstellen, dass die verschiedenen Werte nur aus realistischen Werten bestehen. Das heißt es soll keine 5-stelligen MMSI's oder 20 Stelligen IMO's geben.
ID	DP2003
Beschreibung	Reisebezogene AIS-Daten müssen aus Werten bestehen, welche die Regeln der einzelnen Fehler beachten.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Ankunftszeit hat eine Jahreszahl • Die Namen der Reiseziele entsprechen zu mindestens 80% einem definierten Katalog.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	DP2004

Beschreibung	Dynamische AIS-Daten müssen aus Werten bestehen, welche die Regeln der einzelnen Fehler beachten.
Akzeptanzkriterien	<ul style="list-style-type: none"> Die Positionsdaten müssen grob im definierten Bereich sein
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	DP2005
Beschreibung	Radar-Daten müssen aus Werten bestehen, welche die Regeln der einzelnen Fehler beachten.
Akzeptanzkriterien	
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	DP2006
Beschreibung	Vorgenommene Änderungen müssen in der Datenbank nachvollziehbar und gekennzeichnet sein.

Akzeptanzkriterien	<ul style="list-style-type: none">• Änderungen aus Anforderung DP2002 (statische Daten) sind nachvollziehbar und gekennzeichnet• Änderungen aus Anforderung DP2003 (reisebezogene Daten) sind nachvollziehbar und gekennzeichnet• Änderungen aus Anforderung DP2004 (dynamische Daten) sind nachvollziehbar und gekennzeichnet• Änderungen aus Anforderung DP2005 (Radar-Daten) sind nachvollziehbar und gekennzeichnet
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	

10.5.18 Datenfusion

Unter dem Schritt der Datenfusion wird das bilden von einzelnen AIS und Radar Tracks verstanden, sowie das glätten und fusionieren der Daten.

ID	DP3000
Beschreibung	AIS Tracks müssen gekennzeichnet werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden Regeln festgelegt, ab wann ein neuer Track beginnt • Es können die einzelnen Wegpunkte zu einem Track eingesehen werden
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	DP3001
Beschreibung	AIS Daten müssen vorab geglättet werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden mehrere Verfahren bewertet • Es wurden mindestens zwei Verfahren ausgetestet • Es wurde eine Bewertung durchgeführt, ob das Glätten von AIS Daten lohnenswert ist.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	Es wurden mehrere Verfahren ausgetestet, allerdings hat sich herausgestellt, dass keines der Verfahren den gewünschten Zweck erfüllt hat und deswegen wird auf eine Glättung der Daten verzichtet.

ID	DP3002
Beschreibung	Radar Tracks müssen gekennzeichnet werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden Regeln festgelegt, ab wann ein neuer Track beginnt • Es können die einzelnen Wegpunkte zu einem Track eingesehen werden
Status	ERLEDIGT
Priorität	hoch
Bemerkung	
ID	DP3003
Beschreibung	Radar Daten müssen vorab geglättet werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden mehrere Verfahren bewertet • Es wurden mindestens zwei Verfahren ausgetestet • Es wurde eine Bewertung durchgeführt, ob das Glätten von Radar Daten lohnenswert ist.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	Es wurden mehrere Verfahren ausgetestet, allerdings hat sich herausgestellt, dass keines der Verfahren den gewünschten Zweck erfüllt hat und deswegen wird auf eine Glättung der Daten verzichtet.
ID	DP3004
Beschreibung	AIS Routen und Radarrouten eines Schiffes müssen zusammengeführt werden.

Akzeptanzkriterien	<ul style="list-style-type: none"> • Kurs und Geschwindigkeit stimmen grob überein • Routen wurden zur gleichen Zeit zurückgelegt • Radardaten wurden einem Vessel zugeordnet • Radardaten die nicht zugeordnet werden können, müssen identifiziert worden sein (statisches Objekt, Schiff ohne AIS)
Status	OFFEN
Priorität	hoch
Bemerkung	Der Unterschied zu Anforderung DP3003 ist, dass die Daten bei dieser Anforderung nicht zusammengeführt werden, sondern es vorab eine Möglichkeit geben muss, dass zuvor für die Zusammenführung Routen zum Schiff zusammengeführt werden.
ID	DP3005
Beschreibung	Radar und AIS Daten müssen zu einem Track zusammengeführt werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es wurden mehrere Verfahren bewertet • Es wurden mindestens zwei Verfahren ausgetestet • Es wurde eine Bewertung durchgeführt, inwieweit die Genauigkeit der Daten steigt • Es muss ermittelt werden, welcher Datenquelle mehr zu vertrauen ist • Bei einem Ausfall über einen gewissen Zeitraum einer Datenquelle, müssen die Daten durch die andere Quelle allein gehalten werden • Es muss klar werden aus welchen Datensätzen sich der erhaltene Track zusammensetzt
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	DP3006

Beschreibung	Sämtliche Daten sollen in ein für Datenanalyse optimiertes Schema übertragen werden
Akzeptanzkriterien	<ul style="list-style-type: none"> • Datenmodell wurde von Gesamtteam erstellt • Daten können in gewünschter Form in das Schemaübertragen werden • Rückschlüsse auf Rohdaten sind weiterhin möglich
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	DP3007
Beschreibung	Bei der Datenfusion sollen die vorhandenen Wetterdaten berücksichtigt werden
Akzeptanzkriterien	<ul style="list-style-type: none"> • Relevante Wetterdaten (werden im Konzept bestimmt) fließen in die Datenfusion zwischen AIS und Radar mit ein • Es besteht die Möglichkeit die Wetterdaten zu berücksichtigen oder sie zu ignorieren
Status	OFFEN
Priorität	Mittel
Bemerkung	

10.5.19 Scenario Mining

10.5.19.1 Funktionale Anforderungen

10.5.19.2 Rahmenbedingungen:

ID	SM0000
Beschreibung	Es müssen die rechtlichen Grundlagen für die relevanten Bereiche berücksichtigt werden
Akzeptanzkriterien	<ul style="list-style-type: none"> • SM0000.01 • SM0000.02 • SM0000.03
Status	OFFEN
Priorität	Hoch
Bemerkung	

ID	SM0000.01
Beschreibung	Auf der Nordsee müssen die Kollisionsverhütungsregel (KVR) 6,8,9,10,13,14,15,16,17,18,19 aus dem Teil B für das Scenario Mining berücksichtigt werden.

Akzeptanzkriterien	<ul style="list-style-type: none"> • Regel 6 eingebunden • Regel 8 eingebunden • Regel 9 eingebunden • Regel 10 eingebunden • Regel 13 eingebunden • Regel 14 eingebunden • Regel 15 eingebunden • Regel 16 eingebunden • Regel 17 eingebunden • Regel 18 eingebunden • Regel 19 eingebunden
Status	OFFEN
Priorität	Hoch
Bemerkung	Dabei wurden nur die rechtlichen Grundlagen berücksichtigt, welche durch die zur Verfügung stehenden Beobachtungsdaten wahrnehmbar sind_ _ _ _
ID	SM0000.02
Beschreibung	Auf der Elbe und dem Nord-Ostsee-Kanal muss die Seeschiffstraßenordnung (SeeSchStrO) berücksichtigt werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • §23 • §24 • §25
Status	OFFEN
Priorität	Hoch
Bemerkung	Dabei wurden nur die rechtlichen Grundlagen berücksichtigt, welche durch die zur Verfügung stehenden Beobachtungsdaten wahrnehmbar sind_ _ _ _ _ _ _ _ _ _
ID	SM0000.03
Beschreibung	Der Übergang zwischen den beiden Gebieten muss definiert werden.

Akzeptanzkriterien	<ul style="list-style-type: none"> An jedem Punkt der Karte ist fest definiert, welche Regeln dort gelten müssen
Status	OFFEN
Priorität	Hoch
Bemerkung	Wichtig, da die SeeSchStrO durch den §25 der KVR Regel 18 widerspricht

10.5.19.3 Schiffssituation

ID	SM1000
Beschreibung	Eine Schiffssituation muss separiert werden in eine Crossing-Situation, Head-on-Situation oder Overtaking-Situation.
Akzeptanzkriterien	<ul style="list-style-type: none"> SM1000.01 <ul style="list-style-type: none"> SM1000.01.01 SM1000.01.02 SM1000.02 <ul style="list-style-type: none"> SM1000.02.01 SM1000.02.02 SM1000.03 <ul style="list-style-type: none"> SM1000.03.01 SM1000.03.02 SM1000.04
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	SM1000.01

Beschreibung	Eine Crossing-Situation muss definiert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Parameter zur Erkennung einer solchen Situation müssen festgelegt sein
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	SM1000.01.01
Beschreibung	Die Randfälle einer Crossing-Situation bei 45° und 157,5° auf der Backbordseite und 202,5° und 315° auf der Steuerbordseite müssen durch die Distanz zwischen den Schiffen und der zeitlichen Abfolge des Manövers definiert werden, indem sie statistisch ermittelt werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • 1000.01.02 • prägnante Manöverpunkte wurden gefunden
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Die Winkelgrößen entstanden im Bezug zum MTCAS Code und bilden dabei insgesamt einen 360° Radius ums Schiff -> Am Ende wurde sich bei der Abfolge an die KVR orientiert.
ID	SM1000.01.02
Beschreibung	Für eine Crossing-Situation müssen die Distanz und der Zeitpunkt des Starts und des Endes definiert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Feste Distanz wurde ermittelt • Zeitrahmen wurde ermittelt
Status	ERLEDIGT

Priorität	Hoch
Bemerkung	Dabei ist der Verlauf des Manövers in der Zeit wichtig, um Zwischenfälle zuordnen zu können
ID	SM1000.02
Beschreibung	Eine Head-on-Situation muss definiert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Parameter zur Erkennung einer solchen Situation müssen festgelegt sein
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	SM1000.02.01
Beschreibung	Die Randfälle einer Head-on-Situation bei 45° und 315° müssen durch die Distanz zwischen den Schiffen und der zeitlichen Abfolge des Manövers definiert werden, indem sie statistisch ermittelt werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • 1000.02.02 • prägnante Manöverpunkte wurden gefunden
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Die Winkelgrößen entstanden im Bezug zum MTCAS Code und bilden dabei insgesamt einen 360° Radius ums Schiff
ID	SM1000.02.02
Beschreibung	Für eine Head-on-Situation müssen die Distanz und der Zeitpunkt des Starts und des Endes definiert werden.

Akzeptanzkriterien	<ul style="list-style-type: none"> • Feste Distanz wurde ermittelt • Zeitrahmen wurde ermittelt
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Dabei ist der Verlauf des Manövers in der Zeit wichtig, um Zwischenfälle zuordnen zu können -> Am Ende wurde sich bei der Abfolge an die KVR orientiert. Am Ende wurde sich bei der Abfolge an die KVR orientiert.
ID	SM1000.03
Beschreibung	Eine Overtaking-Situation muss definiert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Parameter zur Erkennung einer solchen Situation müssen festgelegt sein
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	SM1000.03.01
Beschreibung	Die Randfälle einer Overtaking-Situation bei 157,5° und 202,5 müssen durch die Distanz zwischen den Schiffen und der zeitlichen Abfolge des Manövers definiert werden, indem sie statistisch ermittelt werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • 1000.03.02 • prägnante Manöverpunkte wurden gefunden
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Die Winkelgrößen entstanden im Bezug zum MTCAS Code und bilden dabei insgesamt einen 360° Radius ums Schiff -> Am Ende wurde sich bei der Abfolge an die KVR orientiert.

ID	SM1000.03.02
Beschreibung	Für eine Overtaking-Situation müssen die Distanz und der Zeitpunkt des Starts und des Endes definiert werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Feste Distanz wurde ermittelt • Zeitrahmen wurde ermittelt
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Dabei ist der Verlauf des Manövers in der Zeit wichtig, um Zwischenfälle zuordnen zu können

Daten:

ID	SM2000.04
Beschreibung	Es soll bestimmt werden, welche Korrelationen zwischen den AIS-, Radar- und Umweltdaten bei der Betrachtung einer Crossing-Situation bestehen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • AIS-Daten und Umweltdaten sind direkt miteinander verknüpft. • Es wurde herausgefunden, welche Umweltdaten Einfluss auf die prägnanten Parameter haben
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM2000.08
Beschreibung	Es soll bestimmt werden, welche Korrelationen zwischen den AIS-, Radar- und Umweltdaten bei der Betrachtung einer Head-on-Situation bestehen.

Akzeptanzkriterien	<ul style="list-style-type: none"> • AIS-Daten und Umweltdaten sind direkt miteinander verknüpft. • Es wurde herausgefunden, welche Umweltdaten Einfluss auf die prägnanten Parameter haben
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM2000.12
Beschreibung	Es soll bestimmt werden, welche Korrelationen zwischen den AIS-, Radar- und Umweltdaten bei der Betrachtung einer Overtaking-Situation bestehen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • AIS-Daten und Umweltdaten sind direkt miteinander verknüpft. • Es wurde herausgefunden, welche Umweltdaten Einfluss auf die prägnanten Parameter haben
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM2000.16
Beschreibung	Es soll aufbereitet werden, welche Korrelationen zwischen den AIS-, Radar- und Umweltdaten bei der Extraktion einer Route bestehen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • AIS-Daten und Umweltdaten sind direkt miteinander verknüpft. • Es wurde herausgefunden, welche Umweltdaten Einfluss auf die prägnanten Parameter haben
Status	OFFEN

Priorität	Hoch
Bemerkung	

10.5.19.4 Verfahren – Scenario Mining:

ID	SM3000
Beschreibung	Ein Netz aus Manöverpunkten muss für das Scenario Mining mithilfe von Trainingsdaten Routen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • SM3000.01 <ul style="list-style-type: none"> ◦ SM3000.01.01 <ul style="list-style-type: none"> ▪ SM3000.01.01.01 • SM3000.02
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	SM3000.01
Beschreibung	Manöverpunkte müssen ermittelt werden aus den signifikanten Änderungen zwischen COG und SOG-Werten
Akzeptanzkriterien	<ul style="list-style-type: none"> • Relevanten Änderungsgrad zwischen COG und SOG definieren
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Es werden nur noch COG-Werte genutzt, da SOG-Werte zu stark schwanken und sich daher nicht eignen, um Manöverpunkte zu detektieren
ID	SM3000.01.01

Beschreibung	Ermittelte Manöverpunkte müssen geclustert werden
Akzeptanzkriterien	<ul style="list-style-type: none"> • Manöverpunkte sind geclustert
Status	ERLEDIGT
Priorität	hoch
Bemerkung	
ID	SM3000.01.01.01
Beschreibung	Zusammengehörende Manöverpunkte werden miteinander verbunden, dabei entsteht eine Routenauswahl
Akzeptanzkriterien	<ul style="list-style-type: none"> • Kriterien für die Zusammengehörigkeit wurden definiert • Routen wurden abgeleitet
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	SM3000.02
Beschreibung	Ein Netz von Manöverpunkten muss aus zwei Jahren Daten gebildet werden
Akzeptanzkriterien	<ul style="list-style-type: none"> • Der Testdatensatz muss aus zwei Jahren Daten bestehen
Status	OFFEN
Priorität	Hoch
Bemerkung	

ID	SM4000
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining muss mithilfe von Trainingsdaten eine Schiffssituation bei verschiedenen Bedingungen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • SM4000.01 • SM4000.02 • SM4000.03 • SM4000.04 • SM4000.05 • SM4000.06 • SM4000.07 • SM4000.08 • SM4000.09 • SM4000.10 • SM4000.11 • SM4000.12 • SM4000.13 • SM4000.14 • SM4000.15 • SM4000.16 • SM4000.17 • SM4000.18 • SM4000.19 • SM4000.20
Status	OFFEN
Priorität	Hoch
Bemerkung	Lernendes Verfahren wurde nicht mehr umgesetzt
ID	SM4000.01
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining muss mithilfe von Trainingsdaten Crossing-Situationen bei eingeschränkter und guter Sicht lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 334 Datensätzen durchgeführt.
Status	OFFEN

Priorität	Hoch
Bemerkung	Insgesamt 1000 sollten genutzt werden, dadurch wurde das auf die jeweiligen Situationen verteilt Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.02
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Crossing-Situationen soll ein Normalmodell für verschiedene Schiffstypen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 334 Datensätzen durchgeführt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.03
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Crossing-Situationen soll ein Normalmodell für verschiedene Manövriereigenschaften lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 334 Datensätzen durchgeführt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.04

Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Crossing-Situationen muss ein Normalmodell für unterschiedliche Wetterbedingungen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 334 Datensätzen durchgeführt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.05
Beschreibung	Das Klassifikationsverfahren muss Crossing-Situationen zuverlässig klassifizieren können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • 70% der als Crossing-Situationen klassifizierten Szenarien sind tatsächliche Crossing-Situationen.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM4000.06
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Head-on-Situationen muss ein Normalmodell für eingeschränkte und gute Sicht lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 333 Datensätzen durchgeführt.
Status	OFFEN
Priorität	hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.

ID	SM4000.07
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Head-on-Situationen soll ein Normalmodell für verschiedene Schiffstypen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 333 Datensätzen durchgeführt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.08
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Head-on-Situationen soll ein Normalmodell für verschiedene Manövriereigenschaften lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 333 Datensätzen durchgeführt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.09
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Head-on-Situationen muss ein Normalmodell für unterschiedliche Wetterbedingungen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 333 Datensätzen durchgeführt.

Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.10
Beschreibung	Das Klassifikationsverfahren muss Head-on-Situationen zuverlässig klassifizieren können.
Akzeptanzkriterien	<ul style="list-style-type: none"> 70% der als Head-on-Situationen klassifizierten Szenarien sind tatsächliche Head-on-Situationen.
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM4000.11
Beschreibung	Ein Klassifikationsverfahren für das Szenario Mining von Overtaking-Situationen muss ein Normalmodell für eingeschränkte und gute Sicht lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> Das Training wurde anhand von 333 Datensätzen durchgeführt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.12

Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Overtaking-Situationen soll ein Normalmodell für für verschiedene Schiffstypen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 333 Datensätzen durchgeführt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.13
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Overtaking-Situationen soll ein Normalmodell für verschiedene Manövriereigenschaften lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 333 Datensätzen durchgeführt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.14
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Overtaking-Situationen muss ein Normalmodell für unterschiedliche Wetterbedingungen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 333 Datensätzen durchgeführt.
Status	OFFEN
Priorität	hoch

Bemerkung	Macht die Anforderungen so noch Sinn? Es wird ja nicht gelernt, sondern einfach mit eigenen Parametern klassifiziert.
ID	SM4000.15
Beschreibung	Das Klassifikationsverfahren muss Overtaking-Situationen zuverlässig klassifizieren können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • 70% der als Overtaking-Situationen klassifizierten Szenarien sind tatsächliche Overtaking-Situationen.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM4000.16
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Routen muss ein Normalmodell für eingeschränkte und gute Sicht lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 200 Datensätzen durchgeführt.
Status	OFFEN
Priorität	hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.
ID	SM4000.17
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Routen soll ein Normalmodell für für verschiedene Schiffstypen lernen können.

Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 200 Datensätzen durchgeführt.
Status	OFFEN
Priorität	hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.
ID	SM4000.18
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Routen soll ein Normalmodell für verschiedene Manövriereigenschaften lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 200 Datensätzen durchgeführt.
Status	OFFEN
Priorität	hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.
ID	SM4000.19
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Routen muss ein Normalmodell für unterschiedliche Wetterbedingungen lernen können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Das Training wurde anhand von 200 Datensätzen durchgeführt.
Status	OFFEN
Priorität	hoch

Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.
ID	SM4000.20
Beschreibung	Das Klassifikationsverfahren muss die extrahierten Routen zuverlässig klassifizieren können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • 90% der Hauptrouten wurden korrekt geclustert.
Status	OFFEN
Priorität	Hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.
ID	SM4001
Beschreibung	Es müssen für das Scenario Mining bereinigte (u.a. Bereinigung von Datenanomalien) AIS- Radar- und Umweltdaten zur Verfügung stehen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die AIS- Radar und Umweltdaten erhalten ein Label bei dem der Grad der Korrektheit festgehalten wird.
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM4002
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining muss Szenarien für eine vorher definierte Schiffssituation extrahieren können.

Akzeptanzkriterien	<ul style="list-style-type: none"> • SM4002.01 • SM4002.02 • SM4002.03
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM4002.01
Beschreibung	Ein Verfahren für das Scenario Mining von Crossing-Situation muss automatisiert eine Instanz (Szenario) einer Crossing-Situation extrahieren können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Instanz muss die in Anforderung SM1000.01 beschriebenen Merkmale einer Crossing-Situation korrekt widerspiegeln.
Status	ERLEDIGT
Priorität	hoch
Bemerkung	
ID	SM4002.02
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Head-on-Situationen muss automatisiert eine Instanz (Szenario) einer Head-on-Situation extrahieren können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Die Instanz muss die in Anforderung SM1000.02 beschriebenen Merkmale einer Head-on-Situation korrekt widerspiegeln.
Status	ERLEDIGT
Priorität	Hoch

Bemerkung	
ID	SM4002.03
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining von Overtaking-Situationen muss automatisiert eine Instanz (Szenario) einer Overtaking-Situation extrahieren können.
Akzeptanzkriterien	<ul style="list-style-type: none"> Die Instanz muss die in Anforderung SM1000.03 beschriebenen Merkmale einer Overtaking-Situation korrekt widerspiegeln.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	
ID	SM4003
Beschreibung	Das Klassifikationsverfahren muss in der Lage sein, je nach Schiffstyp, dynamischen Daten des Schiffes und Wetterdaten ein anderes Normalmodell für eine Schiffssituation auszugeben.
Akzeptanzkriterien	<ul style="list-style-type: none"> Ausreißer wurden detektiert und entfernt.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM4003.01
Beschreibung	Das Klassifikationsverfahren für Crossing-Situations muss in der Lage sein, je nach Schiffstyp, dynamischen Daten des Schiffes und Wetterdaten ein anderes Normalmodell für eine Schiffssituation auszugeben.
Akzeptanzkriterien	<ul style="list-style-type: none"> Ausreißer wurden detektiert und entfernt.

Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM4003.02
Beschreibung	Das Klassifikationsverfahren für Head-On-Situations muss in der Lage sein, je nach Schiffstyp, dynamischen Daten des Schiffes und Wetterdaten ein anderes Normalmodell für eine Schiffssituation auszugeben.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Ausreißer wurden detektiert und entfernt.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM4003.03
Beschreibung	Das Klassifikationsverfahren für Overtaking-Situations muss in der Lage sein, je nach Schiffstyp, dynamischen Daten des Schiffes und Wetterdaten ein anderes Normalmodell für eine Schiffssituation auszugeben.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Ausreißer wurden detektiert und entfernt.
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM4003.04

Beschreibung	Das Klassifikationsverfahren für Schiffsrouten muss in der Lage sein, je nach Schiffstyp, dynamischen Daten des Schiffes und Wetterdaten ein anderes Normalmodell für eine Schiffssituation auszugeben.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Ausreißer wurden detektiert und entfernt.
Status	OFFEN
Priorität	Hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.

10.5.19.5 Verfahren – Messverfahren:

ID	SM5000
Beschreibung	Es muss für eine Schiffssituation der Grad der Abweichung von der Norm festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung von der Norm
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM5000.01
Beschreibung	Es muss für Crossing-Situations der Grad der Abweichung von der Norm festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung von der Norm
Status	OFFEN

Priorität	hoch
Bemerkung	
ID	SM5000.02
Beschreibung	Es muss für Head-On-Situations der Grad der Abweichung von der Norm festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung von der Norm
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM5000.03
Beschreibung	Es muss für Overtaking-Situations der Grad der Abweichung von der Norm festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung von der Norm
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM5000.04
Beschreibung	Es muss für Schiffsrouten der Grad der Abweichung von der Norm festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung von der Norm

Status	OFFEN
Priorität	hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.
ID	SM5001
Beschreibung	Es muss für eine Schiffssituation der Grad der Abweichung vom Normalverhalten festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung vom Normalverhalten
Status	OFFEN
Priorität	hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.
ID	SM5001.01
Beschreibung	Es muss für Crossing-Situations der Grad der Abweichung vom Normalverhalten festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung vom Normalverhalten
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM5001.02

Beschreibung	Es muss für Head-On-Situations der Grad der Abweichung vom Normalverhalten festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung vom Normalverhalten
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM5001.03
Beschreibung	Es muss für Overtaking-Situations der Grad der Abweichung vom Normalverhalten feststellen werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung vom Normalverhalten
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM5001.04
Beschreibung	Es muss für Schiffsrouten der Grad der Abweichung vom Normalverhalten festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung der Abweichung vom Normalverhalten
Status	OFFEN
Priorität	hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.

ID	SM5002
Beschreibung	Es muss für eine Schiffssituation der Grad des Risikos festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung des Grad des Risikos.
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM5002.01
Beschreibung	Es muss für Crossing-Situations der Grad des Risikos festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung des Grad des Risikos.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM5002.02
Beschreibung	Es muss für Head-On-Situations ein der Grad des Risikos festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung des Grad des Risikos.
Status	OFFEN
Priorität	hoch

Bemerkung	
ID	SM5002.03
Beschreibung	Es muss für Overtaking-Situations der Grad des Risikos festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung des Grad des Risikos.
Status	OFFEN
Priorität	Hoch
Bemerkung	
ID	SM5002.04
Beschreibung	Es muss für Schiffsrouten ein Messverfahren der Grad des Risikos festgestellt werden können.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Verschiedene Abstufungen bei der Messung des Grad des Risikos
Status	OFFEN
Priorität	Hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.

Schellenwerte:

ID	SM6000
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad der Abweichung ein Szenario als Anomalie detektiert wird.

Akzeptanzkriterien	<ul style="list-style-type: none"> • SM6000.01 • SM6000.02 • SM6000.03 • SM6000.04
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Kritische Distanz als Grad der Abweichung
ID	SM6000.01
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad der Abweichung eine Instanz einer Crossing-Situation als Anomalie detektiert wird.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es werden mithilfe des Schwellenwerts maximal 40% getesteten der Crossing-Situationen als anomal detektiert.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Kritische Distanz als Grad der Abweichung
ID	SM6000.02
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad der Abweichung eine Instanz einer Head-on-Situation als Anomalie detektiert wird.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es werden anhand des Schwellenwerts maximal 40% der getesteten Head-on-Situationen als anomal detektiert.
Status	ERLEDIGT
Priorität	Hoch

Bemerkung	Kritische Distanz als Grad der Abweichung
ID	SM6000.03
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad der Abweichung eine Instanz einer Overtaking-Situation als Anomalie detektiert wird.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es werden anhand des Schwellenwerts maximal 40% der getesteten Overtaking-Situationen als anomal detektiert.
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	Kritische Distanz als Grad der Abweichung
ID	SM6000.04
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad der Abweichung eine Instanz einer Schiffsroute als Anomalie detektiert wird.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Es werden anhand des Schwellenwerts maximal 20% der getesteten Schiffsrouten als anomal detektiert.
Status	OFFEN
Priorität	Hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.
ID	SM6001
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad des Risikos ein Szenario als Beinahe-Kollision detektiert wird.

Akzeptanzkriterien	<ul style="list-style-type: none"> • SM6001.01 • SM6001.03 • SM6001.04
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM6001.01
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad des Risikos eine Instanz einer Overtaking-Situation als Beinahe-Kollision detektiert wird.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Bei der Beinahe-Kollision betrug der minimale Abstand zwischen einem Schiff und einem anderen Schiff oder Hindernis höchstens 10 Meter.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM6001.02
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad des Risikos eine Instanz einer Head-on-Situation als Beinahe-Kollision detektiert wird.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Bei der Beinahe-Kollision betrug der minimale Abstand zwischen einem Schiff und einem anderen Schiff oder Hindernis höchstens 10 Meter.
Status	OFFEN
Priorität	hoch

Bemerkung	
ID	SM6001.03
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad des Risikos eine Instanz einer Overtaking-Situation als Beinahe-Kollision detektiert wird.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Bei der Beinahe-Kollision betrug der minimale Abstand zwischen einem Schiff und einem anderen Schiff oder Hindernis höchstens 10 Meter.
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM6001.04
Beschreibung	Es muss ein Schwellenwert festgelegt werden, ab welchem Grad des Risikos eine Instanz einer Schiffsroute als Beinahe-Kollision detektiert wird.
Akzeptanzkriterien	<ul style="list-style-type: none"> • Bei der Beinahe-Kollision betrug der minimale Abstand zwischen einem Schiff und einem anderen Schiff oder Hindernis höchstens 10 Meter.
Status	OFFEN
Priorität	hoch
Bemerkung	Routen werden in der Form gar nicht generiert, somit ist die Anforderung nicht mehr sinnvoll.

Verhaltensmuster

ID	SM7000
-----------	---------------

Beschreibung	Es müssen Bereiche ermittelt werden, an denen Schiffe immer ihr Verhalten ändern
Akzeptanzkriterien	<ul style="list-style-type: none"> • SM7000.01 • SM7000.02
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM7000.01
Beschreibung	Es müssen Bereiche ermittelt werden, an denen Schiffe immer ihre Geschwindigkeit reduzieren/erhöhen.
Akzeptanzkriterien	<ul style="list-style-type: none"> • ermittelten Bereiche müssen an die zugrunde liegenden Gegebenheiten geknüpft werden • der Bereich in der Nordsee muss untersucht sein • der Bereich auf der Elbe muss untersucht sein • der Bereich vor Schleusen und Liegeplätzen muss untersucht sein • der Bereich in Häfen muss untersucht sein
Status	OFFEN
Priorität	hoch
Bemerkung	
ID	SM7000.02
Beschreibung	Es müssen Bereiche ermittelt werden, an denen Schiffe generell keine Manöver fahren bzw. nicht überholen.

Akzeptanzkriterien	<ul style="list-style-type: none"> ermittelten Bereiche müssen an die zugrunde liegenden Gegebenheiten geknüpft werden der Bereich in der Nordsee muss untersucht sein der Bereich auf der Elbe muss untersucht sein der Bereich vor Schleusen und Liegeplätzen muss untersucht sein der Bereich in Häfen muss untersucht sein
Status	OFFEN
Priorität	hoch
Bemerkung	

10.5.20 Nichtfunktionale Anforderungen:

ID	SM8001
Beschreibung	Ein Klassifikationsverfahren für das Scenario Mining muss mithilfe des Java-Frameworks Knime und den dort zur Verfügung stehenden Funktionen, Programmiersprachen und integrierbaren Plugins umgesetzt werden.
Akzeptanzkriterien	<ul style="list-style-type: none"> Klassifikationsverfahren ist eingebunden
Status	ERLEDIGT
Priorität	Hoch
Bemerkung	

10.6 Meilensteine

Bezeichnung	Beschreibung	Fälligkeit	Status
Grundwissen maritimer Bereich	Es wurde sich ein Grundwissen im maritimen Bereich angeeignet	01.05.2018	Done
Anforderungen erstellt	Es wurden sämtliche Anforderungen definiert	31.05.2018	Done
DB-Schema	Datenbankschema definieren für abspeichern von Szenarien	24.06.2018	Done
Konzept benötigte Daten	Es wurde ein Konzept erarbeitet, welche Daten Scenario Mining von Data Processing benötigt	24.06.2018	Done
Konzept für Metadaten	Es wurde ein Konzept erarbeitet, welche Metadaten benötigt werden	30.06.2018	Done
Updatekonzepte für Wetterdaten	Es wurde ein Updatekonzept für neue Wetterdaten erarbeitet	30.06.2018	Done
DB-Schema für Wetterdaten	Es wurde ein DB-Schema für die Wetterdaten erarbeitet	30.06.2018	Done
Datenbereinigungskonzept	Es wurde ein Konzept erarbeitet, wie die Rohdaten	08.07.2018	Done

	bereinigt werden		
Konzept ETL-Prozess	Es wurde ein Konzept erarbeitet, wie der ETL Prozess erstellt wird	08.07.20 18	Done
Gesamtarchitektur-Konzept	Es wurde ein Konzept erarbeitet, wie die Gesamtarchitektur aussieht	22.07.20 18	Done
DB-Gesamtkonzept	Es wurde ein Konzept erarbeitet, wie das Datenbankgesamtkonzept aussieht	22.07.20 18	Done
Data-Fusion-Konzept	Es wurde ein Konzept erarbeitet, wie die Datenfusion vollzogen wird	22.07.20 18	Done
Konzept für Verfahrensweise	Es wurde ein Konzept erarbeitet, wie der Vorgang aussieht bezüglich der Minen von Szenarien	22.07.20 18	Done
Konzept E-Core/U-Core	Es wurde ein Konzept erarbeitet, wie die Umstellung von E-Core auf U-Core vorgenommen werden kann	22.07.20 18	Done
Gesamtkonzept	Es wurde ein Konzept erarbeitet, wie das Gesamtkonzept für die Projektgruppe aussehen	22.07.20 18	Done

	kann.		
Datenmodell erstellt	Das erste Datenmodell wurde erstellt	28.07.2018	Done
Eintrag Wetterdaten	Wetterdaten wurden in die Datenbank eingetragen	31.07.2018	Done
Geclusterte Manöverpunkte	Ein Netz aus geclusterten Manöverpunkten wurde erstellt	31.07.2018	Done
Übliche Verkehrsmuster wurden extrahiert	Übliche Verkehrsmuster wurden extrahiert	31.08.2018	Done
ETL Prozess	ETL-Prozess für das Übertragen der Daten wurde erstellt	31.8.2018	Done
AIS-Daten wurden geglättet	AIS Daten wurden mit einem Filter geglättet	30.9.2018	Verworfen
Klassifikation Begegnungssituationen	Ein Modell zur Klassifikation von Begegnungssituationen wurde erstellt	01.10.2018	Done
Radardaten wurden geglättet	Radar-Daten wurden mit einem Filter geglättet	20.10.2018	Verworfen
Zwischenpräsentation	Die Zwischenpräsentation wurde erstellt	25.10.2018	Done
Umstellung E-Core auf U-Core	Die Umstellung von E-Core auf U-Core ist erfolgt	31.10.2018	Verworfen

Normmodell	Es soll ein Normmodell umgesetzt werden, welches die vor Ort geltenden Regeln umfasst	31.10.20 18	Done
Extraktion Begegnungssituationen	Es sollen alle Begegnungssituationen aus den historischen Daten extrahiert werden.	19.11.20 18	Teilweise erfüllt
Risikobewertung Grad der Abweichung	Es gibt eine Risikobewertung anhand des Grades der Abweichung von der Norm und vom Normalverhalten	22.11.20 18	Done
Fusion von Radar und AIS-Datan ist erfolgt	AIS und Radardaten wurden fusioniert	28.11.20 18	Done
Anomales Verhalten erkennen	Es kann anomales Verhalten bei Begegnungssituationen erkannt werden	30.11.20 18	Done
Korrelationen zwischen Wetter und Schiffsverhalten wurden gezogen	Es wurden Korrelationen zwischen Wetter und Schiffsverhalten gezogen	01.12.20 18	Done
Knime Hauptprozess - Teilprozess	KNIME-Hauptprozess wurde um einen Teilprozess zum automatisierten Scenario Mining erweitert	01.12.20 18	Verworfen
Wetterdaten abspeichern	Alle Daten sind in der DB	03.12.20 18	Teilweise erfüllt

Datenbereinigung	Alle AIS und Radardaten sollen bereinigt im neuen Datenmodell gespeichert werden	03.12.2018	Teilweise erfüllt
Fusion von AIS und Radardaten	Es werden passende AIS und Radartracks gesucht und miteinander fusioniert	10.12.2018	Done
Anomales Schiffsverhalten	Anomales Schiffsverhalten von zwei Jahren Daten wurde herausgefiltert	15.12.2018	Teilweise erfüllt
Wahrscheinlichkeitsdichtefunktion	Wahrscheinlichkeitsdichtefunktion erstellen, um die verschiedenen Begegnungssituationen zu klassifizieren	14.01.2019	Done
IWRAP MK2 Theorie	Theoretische Ansätze für IWRAP MK2 erarbeiten	21.01.2019	Nicht erfüllt
Wetter klassifizieren können	Das Wetter muss in gutes und schlechtes Wetter klassifiziert werden können	21.01.2019	Done
Klassifizierungsfaktoren	Es müssen Klassifizierungsfaktoren für Begegnungssituationen festgehalten werden.	21.01.2019	Done
Statistische Auswertung der Begegnungssituationen	Es werden statistische Auswertungen der Begegnungssituationen	04.02.2019	Done

	vorgenommen		
Bayessche Netze	Bayessches Netz für Szenarien erstellen	04.02.2019	Done
Precision and Recall	Precision und Recall für Data Processing berechnen	11.02.2019	Nicht erfüllt
IWRAP MK2 Umsetzung	Umsetzung der theoretischen Ansätze für IWRAP MK2	25.02.2019	Nicht erfüllt
Verkaufsaufkommen betrachten	Das Verkehrsaufkommen bei einzelnen Szenarien muss abgerufen werden	25.02.2019	Done
Szenarien abspeichern	Szenarien können in der Datenbank abgespeichert werden und es ist eine Verbindung zwischen Szenarien und Daten	25.02.2019	Done
Szenarien in ein allgemeingültiges Format umwandeln	Die Szenarien sollen in einem allgemeingültigen Format wie beispielsweise GPX sein, dass sich in Appdateien umwandeln lässt	25.02.2019	Done
Code Dokumentation	Der Code wurde nach den gruppeninternen Regeln dokumentiert	04.03.2019	Done
Code Testen	Der Code wurde nach	04.03.2019	Teilweis

	Cedrics Vorgaben getestet	19	e
Fertiger Anforderungskatalog	Der Anforderungskatalog wurde fertig gestellt	11.03.2019	Done
Automatisierung des Hauptprozesses	Mit Knime soll ein Hauptprozess erstellt werden	11.03.2019	Verworfen
Dokumentation fertigstellen	Die Dokumentation soll fertig gestellt werden	25.03.2019	Done
Endpräsentation	Die Endpräsentation wurde erstellt	31.03.2019	Done

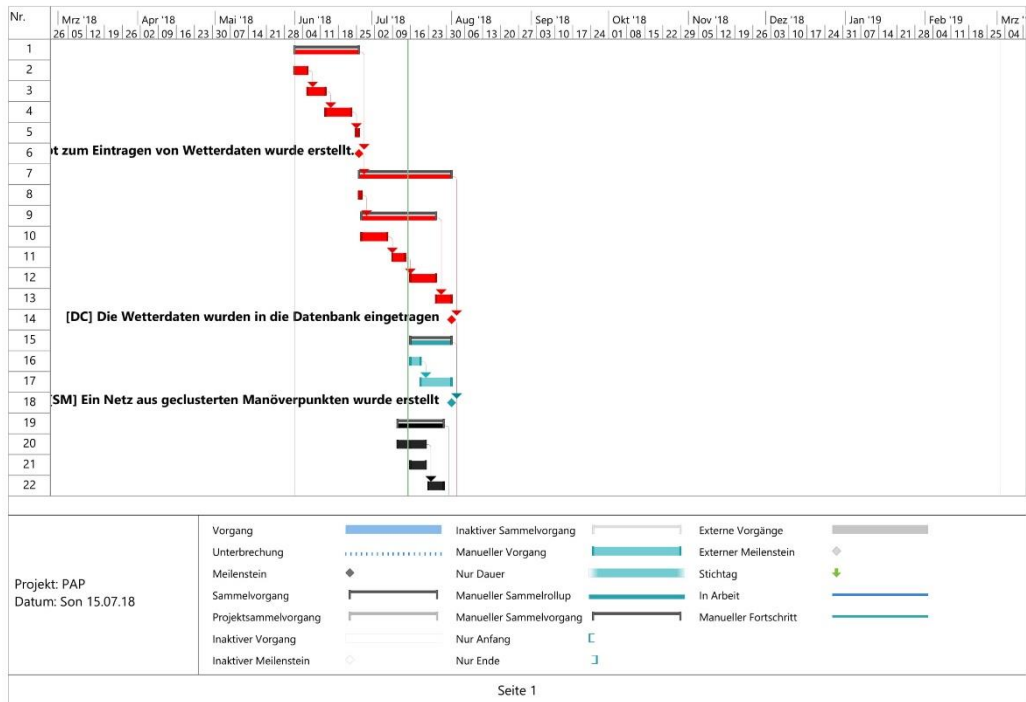
Tabelle 10-1: Erfüllte Meilensteine

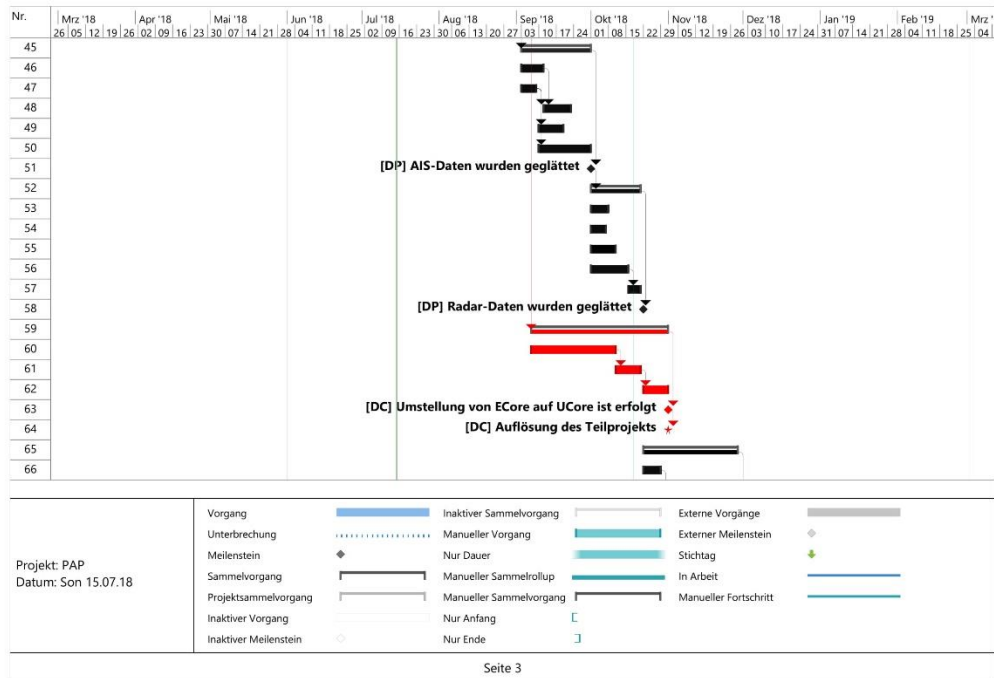
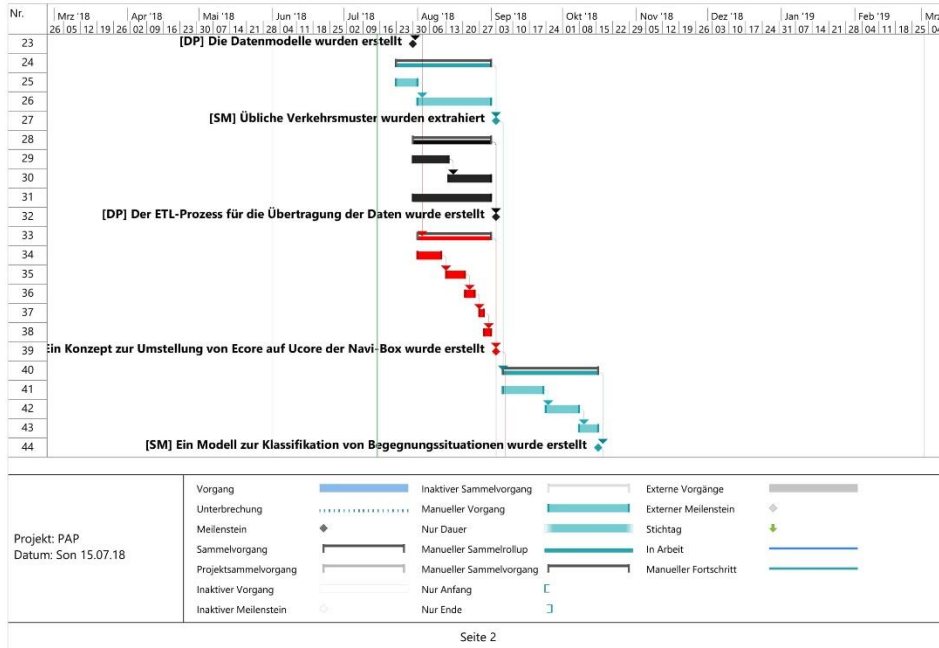
Bezeichnung	Status	Begründung
AIS-Daten wurden geglättet	Verworfen	Es wurden verschiedene Filter ausprobiert um die Daten zu glätten. Im abschließenden Prozess ist jedoch keine abschließende Filterung enthalten.
Radardaten wurden geglättet	Verworfen	Es wurden verschiedene Filter ausprobiert um die Daten zu glätten. Im abschließenden Prozess ist jedoch keine abschließende Filterung enthalten.
Umstellung E-Core auf U-Core	Verworfen	Meilenstein ist durch Projektauftraggeber hinfällig

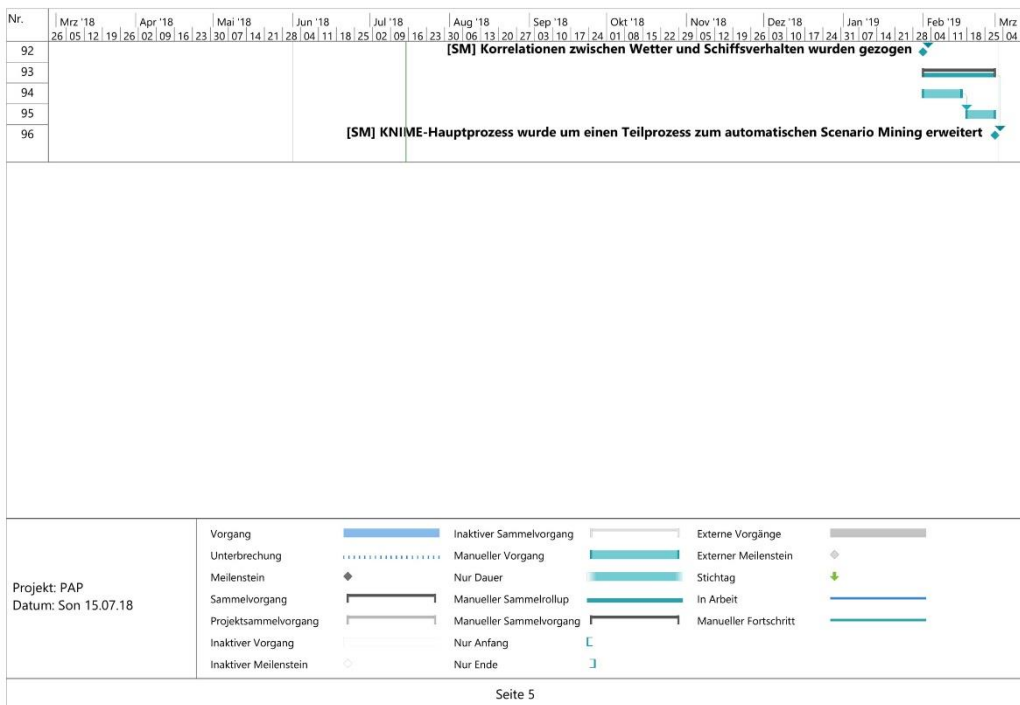
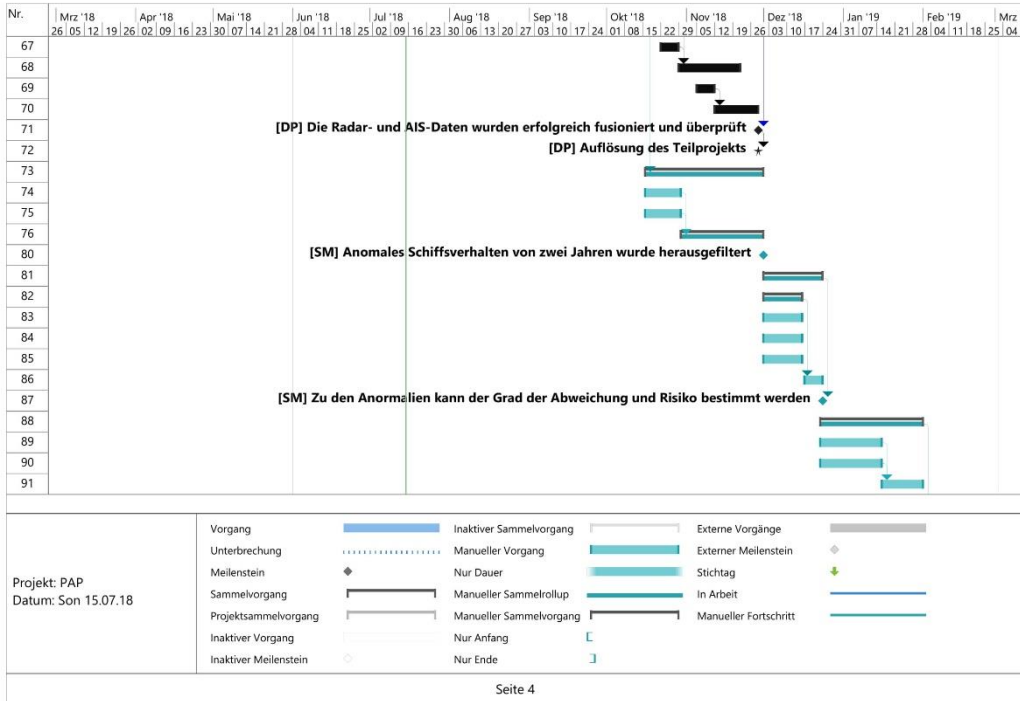
Extraktion Begegnungssituationen	Teilweise erfüllt	Es wurden lediglich 6 Monaten extrahiert
Knime Hauptprozess - Teilprozess	Verworfen	Es gibt keinen Hauptprozess, mit sämtlichen Inhalten, sondern mehrere kleine eigenständige Prozesse.
Wetterdaten abspeichern	Teilweise erfüllt	Es wurden nicht sämtliche Wetterdaten abgespeichert. Lediglich die relevanten Wetterdaten sind abgespeichert.
Datenbereinigung	Teilweise erfüllt	Es wurden lediglich 6
Anomales Schiffsverhalten	Teilweise erfüllt	Es wurden lediglich 6 Monate untersucht
IWRAP MK2 Theorie	Nicht erfüllt	Das Thema wurde bearbeitet, jedoch nicht schriftlich dokumentiert und somit ist es nicht erfüllt.
Precision and Recall	Nicht erfüllt	Das Thema wurde aus zeitlichen Gründen zurückgestellt.
IWRAP MK2 Umsetzung	Nicht erfüllt	Das Thema wurde aus zeitlichen Gründen zurückgestellt.
Code Testen	Teilweise	Die Testabdeckung ist sehr gering und unter den Erwartungen.
Automatisierung der Prozesse	Verworfen	Es gibt keinen Hauptprozess, mit sämtlichen Inhalten, sondern mehrere kleine eigenständige Prozesse

Tabelle 10-2: Nicht oder nur teilweise erfüllte Meilensteine

10.7 Projektablaufpläne des ersten halben Jahres







10.8 Eigenständigkeitserklärungen

Eigenständigkeitserklärung von Rabea:

Diese Erklärung bezieht sich auf die von mir verfassten Kapitel Einleitung, Projektmanagement, Aufteilung des Projektes, Kartenmaterial, Wetterkonzept und Regelkonzept.

Hiermit versichere ich, Rabea Müller, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis an der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.



Oldenburg, den 07.April 2019

Eigenständigkeitserklärung von Jan:

Diese Erklärung bezieht sich auf die von mir verfassten Kapitel KNIME, Technische Infrastruktur, Wetterdaten und Datenassoziation und -fusion.

Hiermit versichere ich, Jan Mentjes, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher

Praxis an der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.

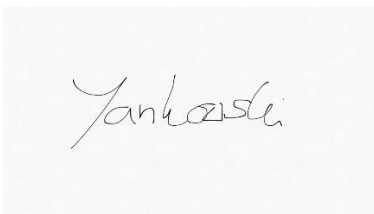


Oldenburg, den 07.April 2019

Eigenständigkeitserklärung von Dennis:

Diese Erklärung bezieht sich auf die von mir verfassten Kapitel Schiffsklassen, Normalmodell zur Repräsentation üblicher Manöver und Tracks, Verkehrsaufkommen. Ebenfalls wurden die Kapitel im Anhang, Erstellung eines Manövernetzes und Benutzung eines Manövernetzes von mir persönlich verfasst.

Hiermit versichere ich, Dennis Jankowski, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis an der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.

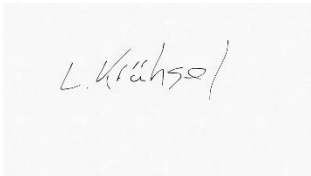


Oldenburg, den 07.April 2019

Eigenständigkeitserklärung von Lennart:

Diese Erklärung bezieht sich auf die von mir verfassten Kapitel Bereinigung der Daten und Benutzerhandbuch ETL-Prozess.

Hiermit versichere ich, Lennart Krühsel, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis an der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.



Oldenburg, den 07. April 2019

Eigenständigkeitserklärung von Simon:

Diese Erklärung bezieht sich auf die von mir verfassten Kapitel Destination Cleaner, Bayessches Netz und Decision Tree.

Hiermit versichere ich, Simon Münster, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis an der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.



Oldenburg, den 07. April 2019

Eigenständigkeitserklärung von Andreas:

Diese Erklärung bezieht sich auf die von mir verfassten Kapitel AIS und Radar, Datenmodell und ETL Prozess.

Hiermit versichere ich, Andreas Robert Saum, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis an der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.



Oldenburg, den 07.April 2019

Eigenständigkeitserklärung von Omid:

Diese Erklärung bezieht sich auf die von mir verfassten Kapitel Schiffsbegegnungssituationen filtern, Klassifikation von Schiffsbegegnungssituationen zur Detektion von anomalen Begegnungssituationen, Klassifikation von Schiffsbegegnungssituationen zur Erkennung von Parametereinflüssen, Statistiken zu den Begegnungssituation, Heatmaps, Fazit, Ausblick und Begegnungssituationen (Anhang).

Hiermit versichere ich, Omid Brunzlow, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis an der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.

A handwritten signature in black ink on a light background. The signature consists of the letters 'O. Brunn' followed by a long, sweeping horizontal stroke that extends to the right.

Oldenburg, den 07.April 2019

Eigenständigkeitserklärung von Cedric:

Diese Erklärung bezieht sich auf das von mir verfasste Kapitel Xtreme Programming und Code Conventions.

Hiermit versichere ich, Jean Cedric Evina Ondo, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis an der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.

A handwritten signature in black ink on a light background. The signature is highly stylized, starting with a large 'C' and ending with a long, sweeping horizontal stroke that extends to the right.

Oldenburg, den 07.April 2019