



Department of Mathematics

Interpretable Anomaly Detection on AIS Spatial Data using Generative Deep Learning

Aldous Birchall

CID: 02129174

Supervised by Dr. James Martin

4 September 2023

Submitted in partial fulfilment of the requirements for the
MSc in Machine Learning and Data Science of
Imperial College London

The work contained in this thesis is my own work unless otherwise stated.

Signed: Aldous Birchall

Date: 4th September 2023

Abstract

In this thesis we propose a method for interpretable machine learning of local marine vessel behaviour using Maritime Vessel Automatic Identification System (AIS) spatial data. By representing vessel trajectories as point patterns on a 2D grid and training a variational autoencoder to generate likely trajectories, we identify anomalous behaviour by measuring reconstruction losses for new observations. We demonstrate the effectiveness of this method using observations of AIS transmissions from vessels off the Gulf Coast of the U.S. and simulated anomalous behaviour. The 2D representation of AIS transmissions allows quick and intuitive analysis of vessel behaviour and the cause of reconstruction losses by human analysts. This makes the proposed method viable for practical anomaly detection applications where human operators must decide whether to act on anomalies raised by an automated system in domains such as marine safety, marine crime detection, and marine ecological monitoring.

Acknowledgements

I would like to thank my supervisor Dr James Martin for his guidance, insights and valuable feedback whilst writing this thesis. I would also like to thank Dr Zak Varty and Professor Nick Heard for their encouragement and support throughout the MSc programme. Finally, I would like to thank my wife Karen, and children Isobel and Sebastian who have endured my absence at weekends in order to study with grace and understanding.

Contents

1	Introduction	1
1.1	The Maritime Vessel Automatic Identification System (AIS)	1
1.1.1	AIS System	1
1.1.2	Overview of AIS Data	2
1.1.3	Uses of AIS Data	3
1.2	Motivation	3
1.3	Relevant Theory	4
1.3.1	Spatial Statistics	4
1.3.2	Machine learning and spatial data	5
1.3.3	Anomaly Detection	7
1.4	Related Work	7
1.4.1	AIS spatial data	7
1.4.2	Anomaly Detection in Marine Vessel AIS Data	8
1.4.3	Anomaly Detection with VAEs	9
1.5	Research Contribution	9
1.6	Research Objectives	10
1.7	Thesis Outline	11
2	Variational Autoencoders (VAEs): Theory	12
2.1	Background	12
2.2	Model Architecture	13
2.3	Loss function and the ELBO	14
2.3.1	Evidence Lower Bound (ELBO) Approximation	15
2.3.2	Derivation of the ELBO	16
2.4	Training and Optimization	17
2.4.1	The Reparametrisation Trick	17
2.4.2	Model training	17
2.5	Hyperparameters	18
3	Data Collection and Pre-processing	20
3.1	Data Collection Methodology	20
3.1.1	Data Sources	20
3.1.2	Choice of Data Set	20
3.2	Data Pre-processing	21
3.2.1	Spatial and Temporal Filtering	21
3.2.2	Data Quality	21
3.2.3	Representation of Observations	22

3.3	Synthetic Data	24
3.3.1	Spatial Anomaly Simulations	24
3.3.2	‘Switching Off’ via Transmission Count Masking	25
4	Anomaly Detection Framework	26
4.1	Anomaly Detection Task	26
4.2	Proposed Methodology	26
4.3	Anomaly Metric	27
4.4	VAE Architecture	28
4.5	Evaluation Metrics	28
5	Experimental Setup	30
5.1	Model Pipeline Summary	30
5.2	Data Pre-processing	30
5.3	Model Training	31
5.4	Model Pipeline Outputs	31
5.5	Implementation Details	31
6	Results and Analysis	33
6.1	Performance Evaluation	33
6.1.1	Visualisation & Qualitative Analysis	33
6.1.2	Quantitative Analysis	39
7	Discussion	41
7.1	Interpretation of Results	41
7.1.1	Qualitative Discussion	41
7.1.2	Quantitative Discussion	42
7.2	Limitations and Challenges	42
7.2.1	Synthetic Data	42
7.2.2	Practical Challenges	43
7.3	Future Research Directions	43
7.3.1	Implementation as Practical Tool	44
8	Conclusion	45
8.1	Summary of Findings	45
8.2	Contributions to Anomaly Detection in Marine Vessel AIS Data	45
8.3	Practical Applications and Implications	45

1 Introduction

1.1 The Maritime Vessel Automatic Identification System (AIS)

The automatic identification system (AIS) came into widespread global use after the Safety of Life at Sea (SOLAS) convention in 2002 stipulated that all maritime vessels over 300 GT (gross tonnage) were obliged to be equipped with an AIS transceiver (Svanberg et al., 2019). Ships equipped with AIS transceivers periodically broadcast their position to other vessels and authorities in their vicinity (Wolsing et al., 2022). AIS broadcasts contain vessel specific information such as cargo type and physical dimensions as well as geospatial location and velocity.

AIS data can be received by anyone with an AIS transceiver. Organisations such as coastguards and commercial data vendors systematically collect this data at varying geographic scales via terrestrial and satellite receivers.

1.1.1 AIS System

Figure 1.1 summarises the AIS communication architecture.

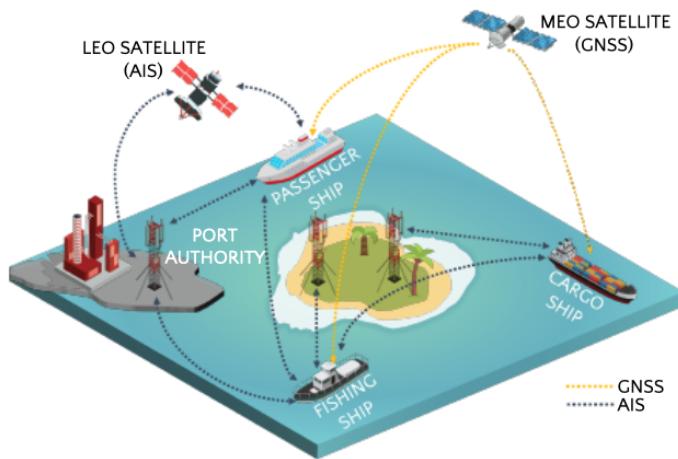


Figure 1.1: Overview of the AIS communication architecture (acronyms are explained in main text). Reproduced from Aziz et al. (2020).

AIS transceivers are located on individual vessels and mounted within ports and along

coastal tracks, to enhance ship tracking and assist in search-and-rescue operations. Vessel transmission frequency lies in the range of 2 to 12 seconds (3 minutes for ships at anchor) and depends on the vessel's velocity. AIS transmissions are in the VHF frequency band using two channels at 161.975 MHz and 162.025 MHz. The range of ground based transceivers is at the most around 70 km but in practice often reduced to around 40 km due to weather and terrain which can attenuate signals. In 2005 AIS transceivers were fitted to low-Earth orbit (LEO) satellites increasing coverage to around 400 km from these satellites. Vessels acquire their spatial position from the medium-Earth orbit (MEO) Global Navigation Satellite System (GNSS) for inclusion in their AIS transmissions (Aziz et al., 2020).

1.1.2 Overview of AIS Data

The contents of each AIS transmission are listed in table 1.1.

Table 1.1: Information contained in individual AIS transmissions. Reproduced from Wolsing et al. (2022).

Type	Data
Static	<ul style="list-style-type: none"> • MMSI number • Call sign & call name • Length & beam • Ship type • Antenna location (aft/bow; port/starboard)
Dynamic	<ul style="list-style-type: none"> • Ship position (PO), accuracy, and integrity • Time in UTC • Course over ground (COG) • Speed over ground (SOG) • Heading (HE) • Rate of turn (ROT) • Navigational status, e.g., at anchor (STA)
Voyage related	<ul style="list-style-type: none"> • Draught • Hazardous cargo (type) • Destination (DST), and estimated time of arrival
Safety related	<ul style="list-style-type: none"> • Text messages

The data elements used in this thesis are the Maritime Mobile Service Identity (MMSI) number which is a unique vessel identifier, ship position (PO) composed of geospatial latitude and longitude components, and time of transmission in coordinated universal time (UTC).

1.1.3 Uses of AIS Data

AIS data is a rich source of information about maritime vessel movements. This is illustrated in figure 1.2 which shows the spatial intensity of individual AIS transmissions in the year 2020 in an area off the U.S. Gulf coast. Shipping lanes and other areas of sea with concentrated shipping traffic can clearly be seen as spatial locations with a high transmission intensity. Consequently the data has many applications beyond collision avoidance, including surveillance for security and law enforcement, economic analysis, maritime planning activities, and mitigating environmental pollution (Svanberg et al., 2019).

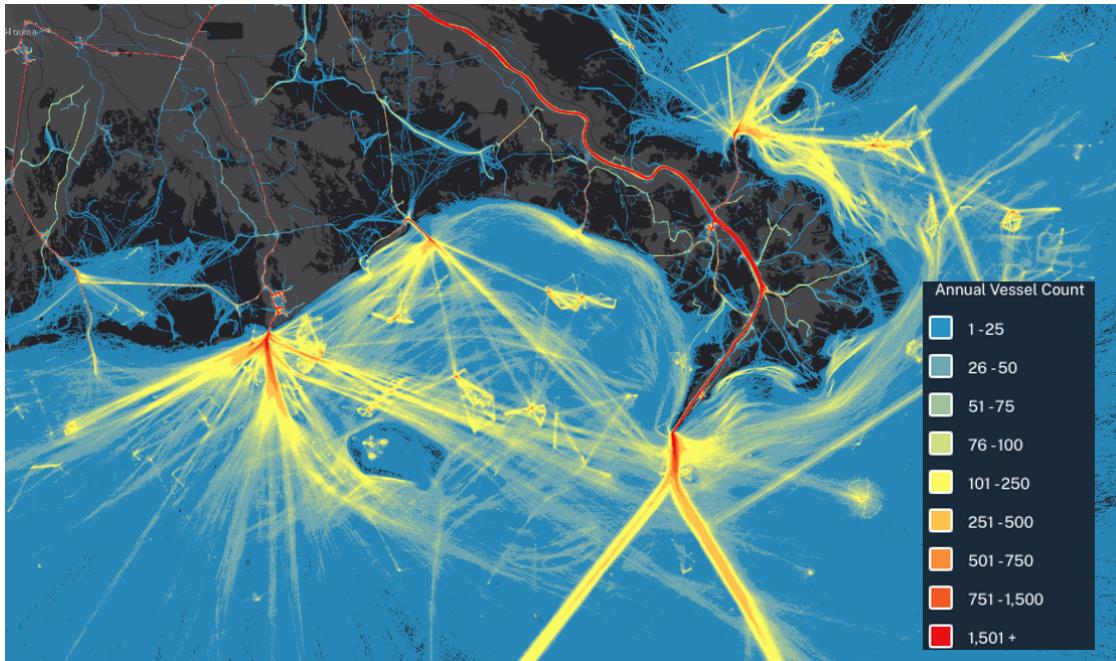


Figure 1.2: Plot of coastal AIS spatial data (illustrative) collected in 2020 south of New Orleans, USA. Source: Bureau of Ocean Energy Management (BOEM).

1.2 Motivation

Maritime freight transport accounts for 80 percent of all goods movement volume globally (Svanberg et al., 2019) so is of great economic significance. Industrial fishing occurs in over 55 percent of ocean area and has a spatial extent more than four times that of agriculture (Kroodsma et al., 2018). In the context of safety and the environment, maritime transport is of particular importance when one considers the volume of hazardous liquids such as crude oil which are transported at sea.

Consequently maritime transport and industrial activity entails significant economic and environmental risks, as well as being a globally important economic activity happening beyond national land borders.

AIS data can be used to monitor and assist these activities, mitigate risks, and help ameliorate problems when things go wrong. Of particular interest to applied mathematicians, data scientists, and marine safety engineers is statistical anomaly detection which can be used in this domain for safety support, monitoring criminal activities, and environmental monitoring (Wolsing et al., 2022).

1.3 Relevant Theory

Analysis in this thesis is grounded in spatial statistics given that the focus of the research is the spatial arrangement of points formed by AIS transmissions. The central objective is to model the process that generates the spatial data. With a suitable model we can distinguish between normal and anomalous behaviour. Given the quantity and complexity of AIS data, machine learning is an applicable class of algorithms that can be used to generate such models.

1.3.1 Spatial Statistics

Point pattern analysis is a subfield of spatial statistics whose object of study is the ‘spatial point pattern’. Baddeley et al. (2015) describe this as a dataset giving the observed spatial locations of things or events. As such, AIS geospatial data, for a given time interval in a particular location, can be considered a spatial point pattern given that it is primarily concerned with the spatial location of waterborne vessels. Given that the locations of individual vessels at any given time interval are random variables, these patterns are of interest as they are suitable for statistical analysis.

Collections of AIS data over temporal intervals long enough to capture multiple transmissions from the same vessel, constitute spatiotemporal data. This means that the point patterns evolve through time, allowing analysis of the behaviour of vessels through time (trajectories and events) or the evolution of a fixed spatial location (such as a port or navigational waypoint) with respect to vessel densities and positions. Baddeley et al. (2015) state: “Treating a spatial point pattern as a realisation of a spatial point process effectively assumes that the pattern is random (in the general sense, i.e. the locations and number of points are not fixed) and that the point pattern is the response or observation of interest.” This accurately describes the use of AIS spatial data to model maritime vessel behaviour in a particular location.

Point patterns can be modelled as statistical processes. Spatial Poisson process, specifically inhomogeneous Poisson process, are a commonly used approach for modelling spatial point patterns. This method defines an intensity function $\lambda(x)$ that gives the expected number of points in an infinitesimally small region around x , where x is typically a point in a 2D coordinate system. For example, we might model the spatial distribution of trees on forest by conditioning $\lambda(x)$ on some property of soil quality. The position of the trees would be determined by sampling from a spatial Poisson distribution with the expected number of points in region determined by $\lambda(x)$.

There many other methods commonly used to model point patterns. These include the Cox process (effectively a Poisson process in which the intensity function $\lambda(x)$ is itself

random), Gibbs models which capture interactions between points, and cluster models such as the Matérn process in which parent points generate offspring points, thereby creating clusters.

Whilst these methods have proved highly effective in many scientific domains such as ecology, microbiology and astronomy (Baddeley et al., 2015), they are probably not the most effective methods for modelling vessel movements given the relative complexity of their behaviour.

1.3.2 Machine learning and spatial data

Machine learning can be applied to spatial data for both supervised and unsupervised learning objectives. In a supervised problem setting, spatial coordinates may be the covariates of an observed measurement. A model could be trained to predict a target value at a particular coordinate.

In an unsupervised setting, machine learning is typically used to find clusters of observations. In spatial problems this usually means grouping observations by location but it can also involve clustering of features generated from spatial covariates such as vessel trajectories. These clusters typically represent normal behaviour with anomalous behaviours sitting beyond the cluster edges.

Figure 1.3 illustrates a drawback with clustering algorithms applied to raw AIS spatial data: whilst clustering is evident in locations such as ports, once on the open sea, vessel positions rapidly diffuse into complex patterns. Consequently spatial clustering has been used to automatically identify ports, anchorages and other locations where vessels converge with high density such as in Wang et al. (2019). However, when clustering is applied to the open sea, the topology of vessel spatial positions (on a quasi 2D surface) renders this approach ineffective. This is because vessels traverse large distances between ports. The resulting structure is web like network of shipping lanes and shortest paths between ports and important geographic waypoints. This type of network topology is in general not amenable to clustering techniques.

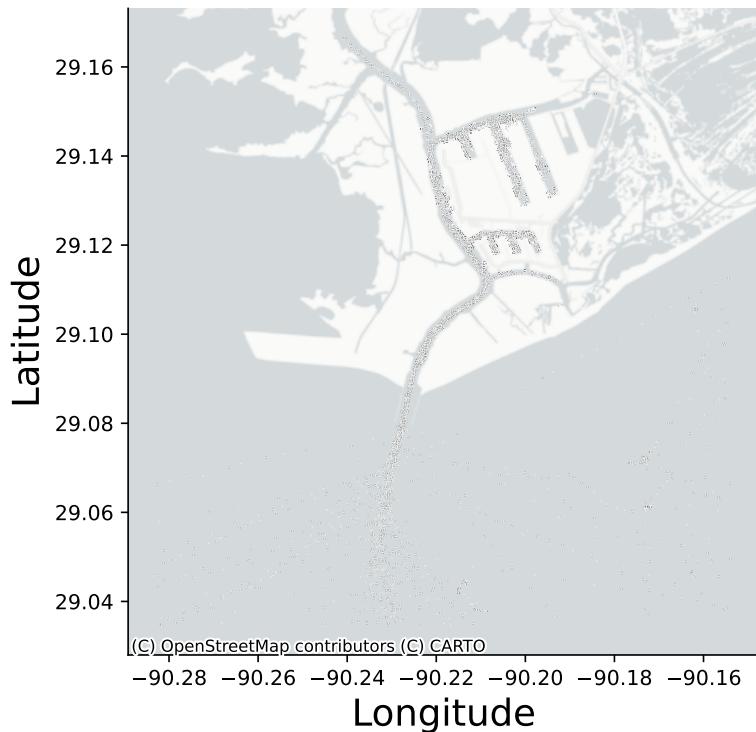


Figure 1.3: Plot of AIS spatial data (illustrative) collected in 2020 south of New Orleans, USA. Source: Bureau of Ocean Energy Management (BOEM).

We have alluded to the spatial complexity of AIS spatial data. In order to model complex phenomena, classical machine learning methods (supervised or unsupervised) typically transform complex observations into simpler representations through feature engineering: the creation of efficient representations of the input data (features) that improve model performance. This often allows relatively simple models to learn complex patterns in the data.

The deep learning paradigm of machine learning dispenses with this approach by allowing arbitrary complexity in the model structure. This reduces model bias at the cost of requiring very large data sets in order to properly train the model. Whilst theoretical treatment of the trade off between model complexity and training data is beyond the scope of this thesis, we can state that modelling complex phenomena without the imposition of domain specific bias through feature engineering requires a lot of data. Fortunately, AIS data is ubiquitous (if not always freely available).

Deep learning has been successful when applied to complex unstructured data such as images and natural language (Chollet, 2021). It is thus suitable for modelling complex spatial data, particularly if we would like to preserve the dimensionality of the training data. Mateu and Jalilian (2022) use a deep learning method called a variational autoencoder (VAE) to model ecological spatial data (plant locations). They claim that their method allows modelling of complex processes where no suitable statistical methods

exist.

1.3.3 Anomaly Detection

Chandola et al. (2009) state that “anomaly detection refers to the problem of finding patterns in data that do not conform to expected behaviour” and that it is widely used in fields such as fraud detection, insurance, health care, cyber-security, fault detection, and military surveillance. Whilst we can disambiguate anomaly detection methods into a number of approaches, principally; supervised, unsupervised, and semi-supervised (Chandola et al., 2009), all of these methods entail modelling the data generating process of interest from a series of observations and then using the trained model to discriminate between normal and abnormal behaviour. It is the approach to discriminating abnormal from normal observations that is specific to various anomaly detection methods. Fundamentally this is a classification problem, and we need not undertake a review of classification methods here. It is however worth noting that anomaly detection tasks often share the following problems (Chandola et al., 2009):

- Defining abnormal behaviour is difficult (clear cut decision boundaries in complex systems are hard to define).
- Normal vs. abnormal class imbalance: anomalies by their very nature are rare events.
- The lack of available labelled anomalies for training and validation.

The above considerations apply to anomaly detection on AIS spatial data. In section 4 we propose an approach that attempts to surmount these problems.

1.4 Related Work

1.4.1 AIS spatial data

A significant body of literature exists on the use of maritime AIS spatial data. In a review entitled “AIS in maritime research”, Svanberg et al. (2019) identify 189 papers covering topics such as “collisions, emissions, monitoring, traffic, noise, logistics and transport economy, whales, oil spills, fishing, and ice”. Several of the papers reviewed are concerned with mapping and modelling the global shipping network (or elements thereof). Zhang et al. (2019) use spatial statistics methods such as spatial autocorrelation analysis (Moran’s I index) of the vessel speed and traffic density to generate a spatial distribution of ship traffic to support decision-making in port operation.

Vespe et al. (2012) and Arguedas et al. (2014) use unsupervised learning on AIS data to map global shipping lanes. Kaluza et al. (2010) apply graph theory metrics to a graph constructed from port departures and arrivals generated from AIS data (the work is however devoid of spatial statistical analysis on the AIS data). Carlini et al. (2020) take this further and analyse the evolution of shipping network graphs over time. Wang et al. (2019) propose a bottom-up approach to generate a spatial representation of the

global shipping network entirely from AIS data using the DBSCAN clustering algorithm to identify nodes (ports and other stop locations).

1.4.2 Anomaly Detection in Marine Vessel AIS Data

Wolsing et al. (2022) summarise 44 papers on anomaly detection approaches for vessel trajectories derived from AIS data. 39 of the 44 papers reviewed use methods which they classify as: “DBSCAN” (6), “GMM and KDE” (6), “Geometry” (5), “Stochastic” (5), “ML & Clustering” (4), “Frameworks” (4), “Bayesian Network” (3), “Gaussian Process” (2) and “Miscellaneous” (4). Feature engineering of vessel trajectories is used in nearly all of these papers including the 5 remaining papers which are classified as “Neural Networks” and therefore of particular interest.

A small number of papers on anomaly detection exist with substantive statistical grounding. Laxhammar (2011) provides a comprehensive review of anomaly detection in trajectory data, contrasting point-based methods with trajectory-based approaches. Ristic (2014) models vessel velocity in Sydney Harbour using a Poisson point process. Li et al. (2018) propose a method that maps trajectories onto a low-dimensional spatial representation allowing point clustering with DBSCAN. Ford et al. (2018) use a binomial GAM to model the probability of receiving an AIS transmission in a one-hour window from given grid location to classify anomalous vessel transmission gaps that may indicate intentional switching off AIS transponders to hide illicit behaviour. The paper applies standard statistical methods to spatial point patterns, using geographic factors associated with failed AIS signal reception to characterise a point process intensity.

Machine learning for anomaly detection on maritime vessel automatic identification system (AIS) data has been extensively researched (Wolsing et al., 2022). The vast majority of this research models ‘normal’ vessel trajectories to identify anomalous movements by comparison. Vessel trajectories are typically modelled and constructed as discrete, continuous lines through 2D space from AIS data and passed as features to machine learning algorithms which learn their statistical properties. For example; Zhen et al. (2017) analyse trajectory patterns derived from AIS data by applying to hierarchical and k-medoids clustering, then use a naive Bayes classifier to discriminate between anomalous and normal vessel behaviour.

The process of trajectory construction constitutes feature engineering and typically involves fitting a line to spatiotemporal point patterns formed from individual vessel coordinates at different times. Malformed trajectories such as very short movements (e.g., less than 500 meters in length) are typically removed from the training data. All the trajectory construction methods impose some degree of modelling bias ranging from simple interpolation to the imposition of complex geometric structures informed by domain knowledge of vessel navigation. Consequently, most research in this domain is performed on a biased covariate. Further, most approaches in the literature discard observed data that fails to fit arbitrary constraints (such as removal of trajectories with length less than 500 meters).

Wolsing et al. (2022) thus represents a rich vein of literature on machine learning applied to spatial data. Most of this literature comes from the engineering community

with a significant proportion published by the Institute of Electrical and Electronics Engineers. Consequently, the epistemological framework is somewhat different from the more statistically (and scientifically) grounded approaches we see in biology, geography, and applied mathematics. The engineering literature tends to state a practical problem and propose a technical solution with varying degrees of theoretical background.

1.4.3 Anomaly Detection with VAEs

Xie et al. (2023) propose a Gaussian mixture model representation of the VAE latent space in supervised anomaly detection (see section 2 for an in depth analysis of VAEs). Their approach uses extensive trajectory pre-processing and filtering with manually labelled anomalous trajectories.

In two papers; Murray and Perera (2020) and Murray and Perera (2021), the authors engineer and split trajectories into past and future segments to predict the future trajectory based on the past trajectory for safety purposes. In the 2020 paper, VAEs are used to cluster past and future trajectory segments. Future segments are predicted by interpolation between clusters of past and future segments. The 2021 paper extends this approach to geographical clusters which uses a recurrent autoencoder and hierarchical clustering passed to a sequence-to-sequence network to learn past to future trajectory segment mappings.

Hu et al. (2023) build similarity graphs (essentially a multi-dimensional similarity measure) of engineered trajectories based on spatial correlations. VAE anomaly detection is performed both on the trajectories and similarity graphs and then combined with weightings determined by a reinforcement learning algorithm. The models are tested on a labelled river-based vessel anomaly dataset.

Duan et al. (2022) apply semi-supervised trajectory classification with a VAE to identify vessel types in places they shouldn't be. Their method includes generation of trajectories and extensive filtering of unusual trajectories for training (effectively the removal of anomalies). The trajectory representation uses 7-hot encoding to include both spatial, kinematic and static data about the vessel. The spatial position is recorded as (one hot encoded) latitudes and longitudes.

Singh and Heymann (2020) detect anomalous ‘switching off’ by automatically labelling historical trajectories as anomalous when there are gaps in transmissions and train an LSTM (long short-term memory network) classifier on this data by representing the trajectories as time series. The classification task is relatively straightforward given the rules based labelling approach. Whilst this work doesn’t use a VAE, it is worth noting as it is a rare example of the application of deep learning to this problem.

1.5 Research Contribution

In this research a ‘natural’ 2D spatial representation of vessel AIS transmissions is used as the unit of observation. The method uses quadrat counts on point patterns composed of the location of AIS transmissions. The counts are performed on AIS data that is partitioned by combinations of time, space, and vessel MMSI number. This grounds the

work in spatial statistical representations, avoids introducing unquantifiable bias into the training data, and avoids the potential removal of data based on arbitrary rules prior to model training. This differentiates the approach from the preceding literature on machine learning approaches to AIS anomaly detection.

The benefits of this approach are that the data representation is parsimonious, easily interpretable by humans when plotted in 2D, can be used at any spatial scale with variable resolution, and allows direct use of the many methods developed for deep learning on images. Similarly to multi-channel image representations; additional channels could be added, for example, kinematic, or static vessel characteristics. A multi-channel approach (essentially stacked 2D grids of observation data) would ensure all data is congruent with the vessel’s spatial position. A further benefit of this representation is it allows training on vessel traffic (simultaneous observation of multiple vessels) and thus has the potential to capture anomalous vessel interactions (such as illicit ship to ship transfers) unlike existing approaches which focus on single vessel trajectories.

The potential research impact is that the proposed method may allow a generally applicable, large-scale (albeit comprised of many local models), approach to vessel anomaly detection grounded primarily in spatial data by directly processing point patterns and thus removing the dependency on subjective feature engineering.

From a theoretical perspective, the elimination of trajectory feature engineering is important in the context of using deep learning methods such as VAEs to model complex behaviour: feature engineering is an essential method for the effective use of classical machine learning algorithms, but it could be considered somewhat incongruous when used with deep learning algorithms (e.g., Murray et al. 2020) because the defining innovation of deep learning is that feature engineering (or more formally representation learning) is left to the neural network (LeCun et al. 2015).

From a practical perspective, this approach has the benefit that it should work in any location; open sea, port or river. Many of the machine learning methods reviewed above such as Hu et al. (2023), Zhen et al. (2017) and Li et al. (2018) are tested on river traffic which significantly simplifies the anomaly detection problem by imposing tight spatial constraints on the behaviour of vessels. This does not imply that these methods are not effective in other locations, but many types of anomaly of interest occur on the open sea which is where we apply our analysis.

1.6 Research Objectives

The research question is “can variational autoencoders be used as an effective method for anomaly detection on 2D representations of AIS data?” In the process of answering the question the following objectives need to be accomplished:

- Define human interpretable 2D data representation(s) of AIS spatial data that can be passed directly to a VAE.
- Develop an effective VAE architecture.

- Design and implement a method for discriminating between anomalies and normal vessel behaviour.
- Design and implement a method for quantifying the model’s efficacy.

1.7 Thesis Outline

This thesis is organised into 8 chapters. This chapter (Chapter 1) introduces the research topic, relevant theoretical background, related work and the rationale for this research. Chapter 2 is dedicated to the theoretical and practical exposition of variational autoencoders given their central importance to the proposed methodology. Chapters 3 and 4 can be considered the ‘problem formulation’ with respect to defining the machine learning task and use of the model outputs; Chapter 3 is concerned with the sources of AIS data and how it was collected for this research. We discuss the rationale for the data preprocessing methods including interpretability and how maintaining similarity to image representations allows the use of existing practical implementations of deep neural networks. In Chapter 4 we explain the proposed anomaly detection method using reconstruction losses from a VAE trained on vessel observations. Chapter 5 covers the implementation of the model pipeline with respect to software and other practical details. The experimental results are presented in Chapter 6 and evaluated from both a quantitative and qualitative perspective. The results are discussed in Chapter 7 along with limitations of the proposed method and future research directions. Finally, Chapter 8 summarises the findings, research contribution, and discusses what the practical implications of this research might be.

2 Variational Autoencoders (VAEs): Theory

2.1 Background

Deep Learning is the class of machine learning algorithms based on artificial neural networks, where ‘deep’ refers to a many layered neural network architecture. The defining innovation of deep learning is that feature engineering (or more formally representation learning) is left to the neural network. This solves a serious limitation of classical machine-learning which is the requirement of careful engineering and often extensive domain knowledge to transform raw data into suitable feature vectors (LeCun et al., 2015).

Goodfellow et al. (2016) state that “Deep learning solves the central problem in representation learning by introducing representations that are expressed in terms of other simpler representations”. Chollet (2021) makes a similar point, claiming that deep learning could have been called “hierarchical representations learning”. The benefit of this approach to machine learning is that it provides a practical method for the modelling of highly complex phenomena.

Deep learning models excel at learning the statistical structure of complex data such as images and text (Chollet, 2021). This is often achieved by learning a low dimensional statistical latent space (often called the embedding space) of this data. This is a vector space where each point represents a condensed, abstract representation of input data that encodes its defining attributes. Particular directions in this space often represent important properties of the data.

Generative deep learning models can sample from such latent spaces to create entirely new examples of the data with similar statistical properties to the training data. VAEs are a type of latent variable generative model that has been used extensively in image generation. In this domain VAEs learn latent visual spaces. Sampling from them creates entirely new, often highly realistic images interpolated from real ones (Chollet, 2021). White (2016) demonstrates this by learning the latent space of celebrity faces. Entirely fictitious, yet realistic, faces can be generated by sampling from the visual latent space learnt by the VAE. Attribute vectors (directions in the latent space) are identified for many image properties such as a ‘smiling’, ‘young’ and ‘attractive’. Traversing and sampling from the latent space along these vectors generates images with incremental degrees of these properties.

Latent spaces for any kind of structured or unstructured data can in principle be learnt by deep learning models. For the purposes of this thesis we are concerned with AIS spatial data which is similar to visual data in that it can be represented as a two-

dimensional grid consisting of different values. In image processing this is pixel intensities in an image, in this thesis we are concerned with transmission counts in a spatial grid.

A common concern about deep learning models is that they are effectively black boxes because it is very difficult to say anything concrete about what they are doing *once trained for a particular task* at the level of neurons and layers. Whilst there are a number of methods that can help to ‘explain’ what is going on with respect to how features are transformed into outputs (see Ribeiro et al. (2016) and Lundberg and Lee (2017) for extensive treatment of model-agnostic interpretation methods), they are usually limited to feature (covariate) importance with respect to the relative sensitivity of model outputs to the various inputs.

One of the benefits of using image based models is that they are usually easier to interpret than other data modalities given our direct perception of images. For example White (2016) shows how a smile changes as the latent space is traversed along the ‘smile’ attribute vector. Whilst analysis of attribute vectors on AIS data is beyond the scope of this work, the use of VAE opens up the possibility of decomposing vessel behaviour into parsimonious, statistically measurable, attributes. However, the main benefit with respect to interpretability of the proposed approach is simply representing the AIS in a 2D plane that maps directly to the water surface under observation. This is effectively an image of where vessels have been, and thus easy to interpret by humans.

2.2 Model Architecture

In order to understand how VAEs work and the rationale for their architecture, it is instructive to consider a simpler, but in many ways similar, model architecture known as an autoencoder. Fig 2.1 shows a generic autoencoder architecture.

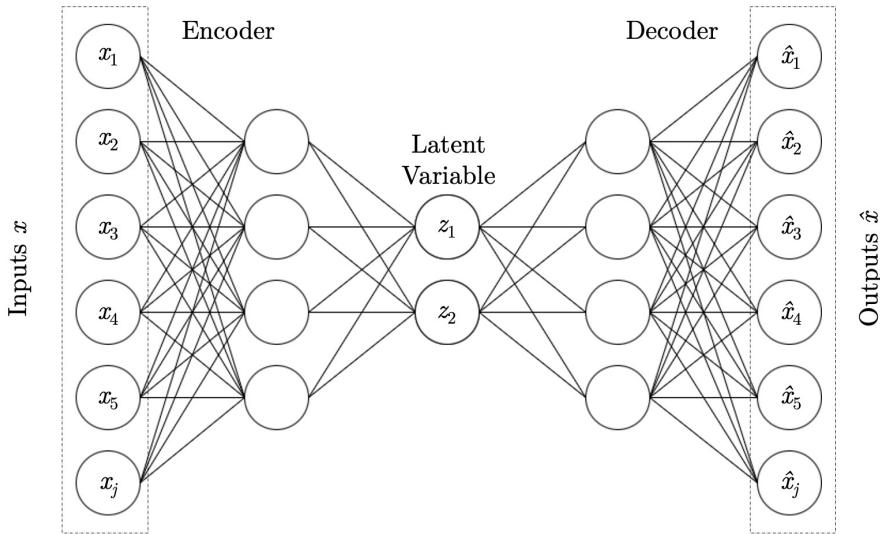


Figure 2.1: Network architecture of an autoencoder.

The autoencoder is comprised of an encoder, latent variable, and decoder. During training, the model objective is to accurately reproduce the input at the output. Forcing the representation of the input data through a bottleneck coerces the model into learning efficient, low dimensional, representations of the training data. In this sense the autoencoder can be considered a data compression algorithm. Each training example is effectively stored as a z -dimensional point in this latent space and reconstructed if this point is fed into the decoder.

Similarly to an autoencoder, a VAE is composed of a decoder, encoder and a latent space. The critical difference is that the VAE learns a well structured, statistical representation of the latent space that if *sampling* from, and passed to the encoder, will produce a realistic output. This is why VAEs are referred to as latent variable generative models.

VAEs get their name from the fact that whilst ostensibly similar to autoencoders, their latent space is structured and parametrised using variational Bayes: a method for learning approximations to the (posterior) distribution. In this case the posterior distribution is a continuous latent space representation that efficiently captures the properties of the training data. In variational Bayes it is represented by family of parametric distributions (in practice often Gaussian). This means the latent space is described in terms of distribution parameters such as mean and variance (in the Gaussian case at least). This approach to approximating latent space distribution not only captures the inherent randomness of the training data, but results in a smooth continuous space. This smoothness allows interpolation between training examples forcing structure into the latent space that gives rise to attribute vectors and the properties of latent spaces discussed in 2.1.

Variational Bayes is used in VAEs because it is a practical and effective method of approximating the (usually) intractable integrals that represent the true posterior distribution of the latent space. It does however mean that the model is not invertible owing to the necessity of sampling from the latent space and thus cannot be straightforwardly trained using maximum likelihood estimation.

2.3 Loss function and the ELBO

We can write the generative model (latent space and decoder) as:

$$p_{\theta}(z)p_{\theta}(x | z)$$

where $p_{\theta}(z)$ is latent space distribution with latent variable $z \in \mathbb{R}^l$, and $p_{\theta}(x | z)$ is the conditional distribution of possible observations defined by the neural network. The marginal likelihood (probability of observation) of an individual data point $x \in \mathbb{R}^D$ is given by

$$p_{\theta}(x) = \int p_{\theta}(z)p_{\theta}(x|z) dz \quad (2.1)$$

where θ are the model parameters.

Training a model entails finding the parameters θ that maximise the marginal likelihood, however, the integral in equation 2.1 is generally intractable, so an approximation is required.

2.3.1 Evidence Lower Bound (ELBO) Approximation

The marginal likelihood is the sum of marginal likelihoods of individual data points:

$$\log p_\theta(\mathbf{x}) = \sum_{i=1}^N \log p_\theta(x_i)$$

This can be re-written as per Kingma and Welling (2013):

$$\log p_\theta(x_i) = D_{KL}(q_\phi(z | x_i) \| p_\theta(z)) + \mathcal{L}(\theta, \phi; x_i) \quad (2.2)$$

where $q_\phi(z | x_i)$ is a parametric distribution (which may be chosen for computational efficiency or quality of approximation), D_{KL} denotes the Kullback-Leibler (KL) divergence, given by

$$D_{KL}(q_\phi(z | x_i) \| p_\theta(z)) = \int q_\phi(z | x_i) \log \left(\frac{q_\phi(z | x_i)}{p_\theta(z)} \right) dz$$

and the first term in 2.2 is the KL divergence of the approximate and true posterior. The second term; $\mathcal{L}(\theta, \phi; x_i)$ is the lower bound on the marginal likelihood of data point i . This can be written as:

$$\log p_\theta(x_i) \geq \mathcal{L}(\theta, \phi; x_i) = \mathbb{E}_{q_\phi(z|x)} [-\log q_\phi(z | x) + \log p_\theta(x | z)]$$

and can also be written as:

$$\mathcal{L}(\theta, \phi; x_i) = \underbrace{\mathbb{E}_{q_\phi(z|x_i)} [\log p_\theta(x_i | z)]}_{\text{reconstruction loss}} - \underbrace{D_{KL}(q_\phi(z | x_i) \| p_\theta(z))}_{\text{regulariser}} \quad (2.3)$$

where the terms on the right hand side can be interpreted as the reconstruction loss and a regulariser (Webster, 2023). The regulariser ensures the approximate distributions do not deviate too far from the prior distribution. Further, $q_\phi(z | x)$ can be interpreted as the encoder network that maps the input x to the latent variables z , and $\log p_\theta(x | z)$ as the decoder network that reconstructs the input corresponding to a value of z .

Figure 2.2 summarises the complete VAE architecture.

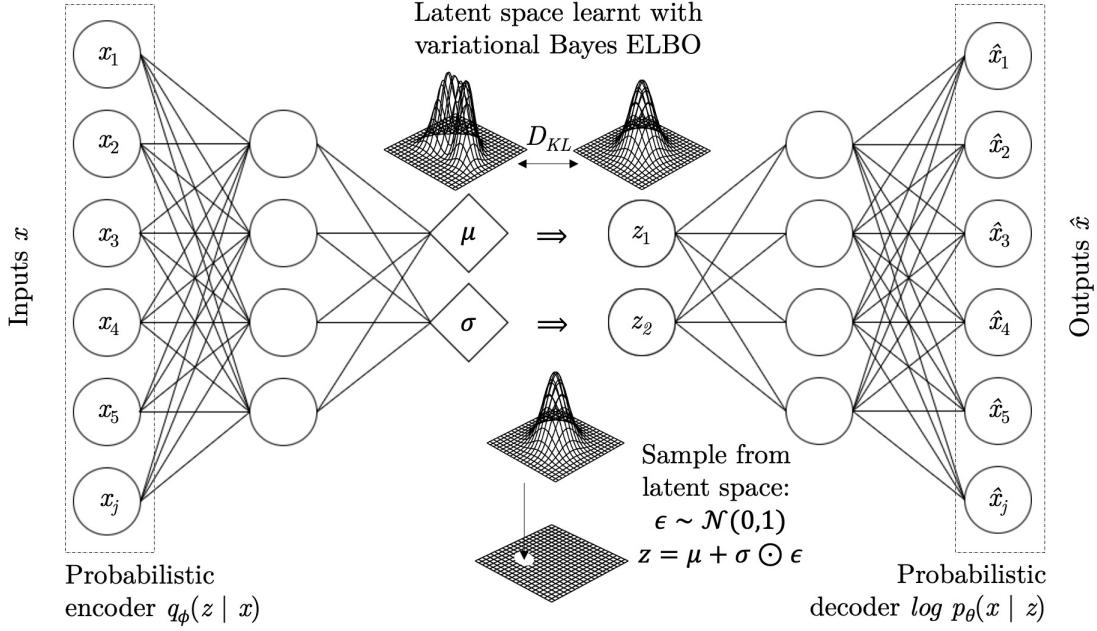


Figure 2.2: The variational autoencoder. The encoder and decoder are linked by the latent space: a low dimensional, probabilistic representation of the training data learnt using variational Bayes evidence lower bound (ELBO). New examples are generated by sampling from the latent space distribution.

2.3.2 Derivation of the ELBO

For completeness the derivation of the ELBO is presented below (Webster, 2023). As per (2.1), the marginal log-likelihood is given by:

$$\begin{aligned}
\log p_\theta(x_i) &= \log \int p_\theta(x_i|z)p_\theta(z) dz \\
&= \log \int p_\theta(x_i|z) \frac{p_\theta(z)}{q_\phi(z|x_i)} q_\phi(z|x_i) dz \\
&\geq \int \log \left(p_\theta(x_i|z) \frac{p_\theta(z)}{q_\phi(z|x_i)} \right) q_\phi(z|x_i) dz \\
&= \int q_\phi(z|x_i) \log p_\theta(x_i|z) dz - \int q_\phi(z|x_i) \log \left(\frac{q_\phi(z|x_i)}{p_\theta(z)} \right) dz \\
&= \mathcal{L}(\theta, \phi; x_i)
\end{aligned} \tag{2.4}$$

where 2.4 uses Jensen's inequality.

2.4 Training and Optimization

In order to train the VAE we need to maximise the ELBO objective function and thus maximise the lower bound on the marginal log-likelihood in equation 2.3. In deep learning, we typically compute the gradients of the loss function with respect to the model's parameters using backpropagation. These gradients are then used by an optimisation algorithm such as stochastic gradient descent (SGD) to minimize the loss function by iteratively updating the model parameters. This is not straightforward with the ELBO because taking gradient with respect to ϕ (the encoder network parameters) requires applying backpropagation through the stochastic nodes which randomly sample z_j from the posterior latent space.

2.4.1 The Reparametrisation Trick

The ‘reparametrisation trick’ addresses the difficulties of taking gradients of the random latent variable z . Instead of sampling from a distribution with learned parameters, we sample from a fixed ‘noise’ distribution and then deterministically transform this sample using the learned parameters. In the case of a Gaussian distribution, instead of sampling z from $\mathcal{N}(\mu, \sigma^2)$, we sample ϵ from $\mathcal{N}(0, 1)$ and compute $z = \mu + \sigma \odot \epsilon$. This way, the randomness is separated from the backpropagation process, and the gradients can flow through μ and σ deterministically.

Specifically we apply a differentiable transformation $g_\phi(\epsilon, x_i)$ of an auxiliary noise variable $\epsilon \sim p(\epsilon_j)$ to z_j :

$$z_j \sim q_\phi(z_j | x_i) z_j = g_\phi(\epsilon_j, x_i), \quad \epsilon \sim p(\epsilon_j)$$

In the case where $q_\phi(z_j | x_i)$ is a multivariate Gaussian distribution $N(\mu_\phi(x_i), \Sigma_\phi(x_i))$, then we could reparameterise the distribution as

$$p(\epsilon_j) = N(\mathbf{0}, \mathbf{I}), \quad g_\phi(\epsilon_j, x_i) = \mu_\phi(x_i) + L_\phi(x_i)\epsilon_j, \quad \text{where } L_\phi(x_i)L_\phi(x_i)^T = \Sigma_\phi(x_i).$$

The encoder network would then output the distribution parameters $\mu_\phi(x_i)$ and $L_\phi(x_i)$, which are both fully differentiable with respect to ϕ (Webster, 2023).

2.4.2 Model training

After applying the reparametrisation trick to loss function, we now have a practical approach to training the VAE. The reparameterised form of the ELBO is the stochastic gradient variational Bayes (SGVB) estimator and can be written as (Kingma and Welling, 2013):

$$\mathcal{L}(\theta, \phi; x_i) = \frac{1}{L} \sum_{l=1}^L (\log p_\theta(x_i | z_j) - D_{KL}(q_\phi(z_j | x_i) \| p(z_j))) \quad (2.5)$$

with reparametrisations:

$$\begin{aligned} z_j &= g_\phi(\epsilon_j, x_i), \\ \epsilon_j &\sim \mathcal{N}(0, 1). \end{aligned}$$

for the case where the KL-divergence term can be integrated analytically. When Gaussian distributions are used for the posterior approximation in variational Bayes, this is indeed the case (Kingma and Welling, 2013). Algorithm 1 shows how the model is trained using the SGVB estimator (adapted from Kingma and Welling (2013)).

Algorithm 1 Mini-batch version of the Auto-Encoding VB (AEVB) algorithm using the SGVB estimator in equation 2.5

```

1:  $\theta, \phi \leftarrow$  Initialize parameters
2: repeat
3:    $X^M \leftarrow$  Random mini-batch of  $M$  data points (drawn from full dataset)
4:    $\epsilon \leftarrow$  Random samples from noise distribution  $p(\epsilon)$ 
5:    $g \leftarrow \nabla_{\theta, \phi} \tilde{\mathcal{L}}^M(\theta, \phi; X^M, \epsilon)$ 
6:    $\theta, \phi \leftarrow$  Update parameters using gradients  $g$  (e.g., stochastic gradient descent)
7: until convergence of parameters  $(\theta, \phi)$ 
8: return  $\theta, \phi$ 
```

2.5 Hyperparameters

Variational Autoencoders have several hyperparameters that need to be set before training. In deep learning architectures such as this, setting the hyperparameters is often the most challenging element of building the model pipeline because architectural choices have a large and non-linear impact on results. The most important hyperparameters are listed below:

- Latent Dimension l : The number of latent variables $z \in \mathbb{R}^l$ used to represent the data. This determines the capacity of the VAE to capture the underlying structure of the data. This needs to be chosen to provide enough model complexity to effectively model the data whilst maintaining enough parsimony impose regularisation and ensure over-fitting is avoided.
- Encoder and Decoder Architectures $q_\phi(z | x)$ and $\log p_\theta(x | z)$: The VAE needs to efficiently capture the raw data and reduce it to a bottleneck around the latent space before reconstructing. We can consider this architecture as determining the compression and decompression rate and process in the model. A good architecture will allow efficient training and find effective compression transforms.

- Prior Distribution $p_\theta(x)$: This is the prior distribution used in variational Bayes. Given the rationale for this method, a simple, computationally tractable distribution such as a Gaussian is chosen to simplify the KL divergence term and regularise the output.
- Training parameters such as the learning rate of the gradient descent optimiser, training batch size and number of epochs (or early stopping parameters) need to be set.

3 Data Collection and Pre-processing

3.1 Data Collection Methodology

The main challenge in collecting AIS data for scientific research (primarily from a reproducibility perspective) is the limited availability of suitable public sources. A secondary challenge is the potential size of AIS data sets required to answer a specific research question. If the research is concerned with even a relatively modest spatial area (in the context of maritime navigation), or over more than week of observations, the data files can become large enough that specialist data-processing hardware and software may be required.

3.1.1 Data Sources

Whilst there are number of comprehensive sources of commercially available AIS data, comparatively little public data is available. The following three potentially usable public data sources were identified:

1. U.S. coastal AIS data provided by a partnership of the Bureau of Ocean Energy Management (BOEM), National Oceanic and Atmospheric Administration (NOAA) and the United States Coast Guard (USCG).
2. The IEEE Data Port website has a large AIS dataset consisting of observations in the Baltic Sea during years 2017-19. The data was collected by Hakola (2020). The AIS observations do not cover all months of the collection period.
3. Global Fishing Watch has a dataset designed for training machine learning models for detecting fishing events and fishing gear type by analysing AIS data. It is stored as individual CSV files, one for each of their seven different categories of fishing vessel.

3.1.2 Choice of Data Set

The U.S. coastal data set provided by the Bureau of Ocean Energy Management (BOEM) was chosen as the data source for this thesis for the following reasons:

- It provides raw AIS data.
- It is up to date and provides extensive historical data going back to 2009 (meaning there is more than enough training data available for this task).

MMSI	BaseDateTime	LAT	LON	SOG	COG	Heading	VesselName	IMO	CallSign	VesselType	Status	Length	Width	Draft	Cargo	TransceiverClass
366515000	2020-03-30T08:21:41	29.11389	-90.20422	0.0	149.1	77.0	BLUE TARPON	IMO9529906	WNSD	90.0	5.0	91.0	18.0	4.5	90.0	A
366515000	2020-03-30T08:27:41	29.11394	-90.20422	0.0	212.5	77.0	BLUE TARPON	IMO9529906	WNSD	90.0	5.0	91.0	18.0	4.5	90.0	A
366515000	2020-03-30T08:12:40	29.11391	-90.20425	0.0	216.6	77.0	BLUE TARPON	IMO9529906	WNSD	90.0	5.0	91.0	18.0	4.5	90.0	A
366515000	2020-03-30T08:18:40	29.11394	-90.20422	0.0	71.9	77.0	BLUE TARPON	IMO9529906	WNSD	90.0	5.0	91.0	18.0	4.5	90.0	A
366515000	2020-03-30T08:24:41	29.11388	-90.20424	0.0	247.7	77.0	BLUE TARPON	IMO9529906	WNSD	90.0	5.0	91.0	18.0	4.5	90.0	A

Figure 3.1: Sample of raw data CSV from AccessAIS tool. The key data elements are explained in the text.

- It provides an easy to use web application (AccessAIS) for defining and downloading AIS data sets covering specific coordinates and time periods.

The AccessAIS web application provided by the Bureau of Ocean Energy Management (BOEM) can greatly reduce download file sizes because spatiotemporal filtering can be applied at source rather than after the data is downloaded. This meant that the use of specialist big data hardware and software was not required to conduct this research.

The data used for model training and testing was covers the dates from 2019-01-02 to 2023-03-27. The spatial coverage was longitude -90.363 to -90.061 and latitude 28.842 to 29.094 where coordinates are given to three significant figures. This resulted in a comma separated value (CSV) download file with a size of 1.80 gigabytes. An example of the raw data is given in figure 3.1. As discussed in the introduction, the AIS data elements we are concerned with in this thesis are the Maritime Mobile Service Identity (MMSI) number which is a unique vessel identifier, vessel position composed of geospatial latitude (LAT) and longitude (LON) components, and time of transmission in coordinated universal time (BaseDateTime) from each AIS transmission.

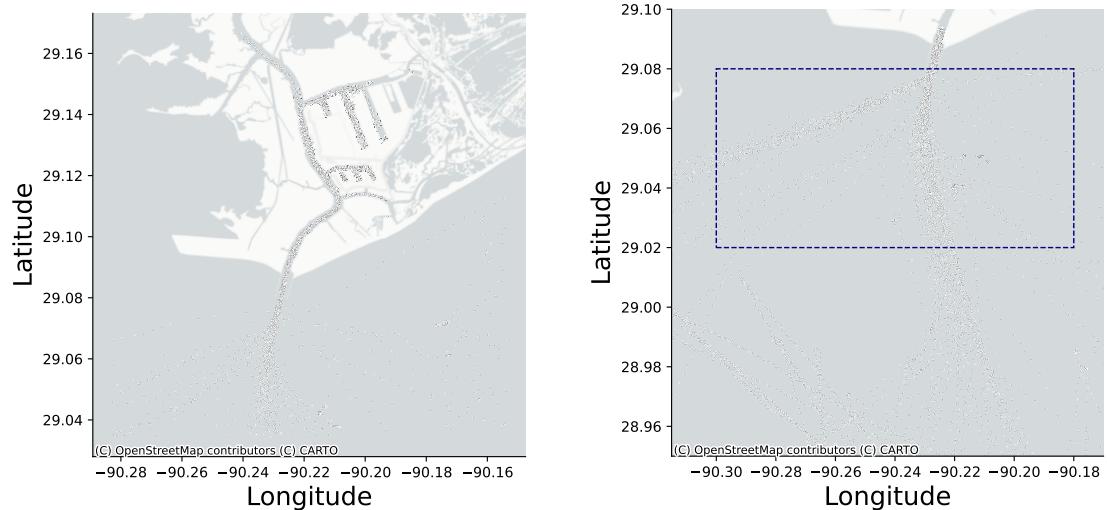
3.2 Data Pre-processing

3.2.1 Spatial and Temporal Filtering

Data was spatially filtered to limit the observation area to a box in the open sea outside of Port Fouchon (figure 3.2a) with bounding coordinates latitude top = 29.08, latitude bottom = 29.02, longitude west = -90.30, longitude east = -90.18. This is roughly an 12km by 7km box as shown in figure 3.2b. The data was not temporally filtered as the whole time period in the raw download was used.

3.2.2 Data Quality

No obvious data quality issues were found during exploratory data analysis so no specific data quality process were required.



(a) AIS spatial data (for geographic context) at Port Fouchon south of New Orleans.

(b) Region in which AIS data collected. Area in red box was used for analysis.

Figure 3.2: Observation area with nearest port for context. Each point is an individual AIS transmission. Plots are for geographical illustration.

3.2.3 Representation of Observations

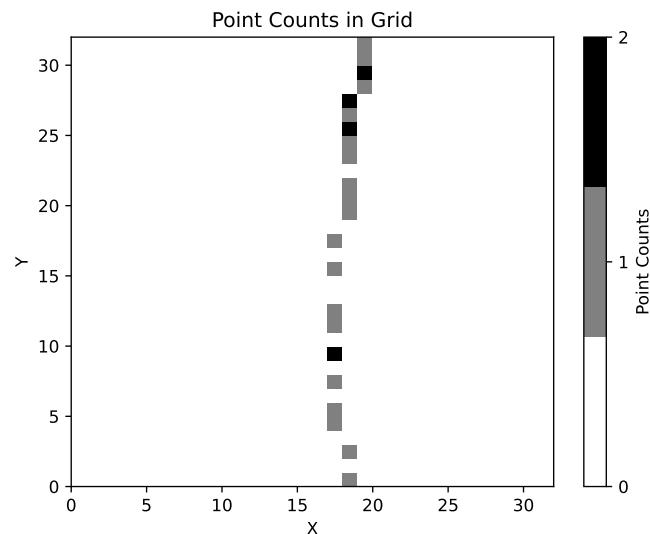


Figure 3.3: Example of quadrat count observation representation

Figure 3.3 shows a quadrat (2D grid) count representation of multiple AIS transmissions. Whilst no vessel trajectory is constructed (other than through basic data filtering), the representation is such that the pattern can be straightforwardly interpreted as vessel moving vertically across the grid space. The grid space shown in figure 3.3 maps directly to the box in figure 3.2b. It's worth noting the similarity of figure 3.3 to simple image representations such as the MNIST data set: this is intentional given the research objectives.

Two types of observation representations were generated. The first was a time-boxed ‘scene’. In this representation AIS transmissions in contiguous 60 minute time periods were grouped and the transmission locations counted using a 32×32 quadrat partition of the observation area. This representation includes data from any vessel passing through the observation area in the observation period. Hence the term ‘scene’ as this gives a representation of all marine traffic in a particular spatiotemporal partition.

The second method represents individual vessel trajectories. Transmissions were partitioned by vessel and then temporally ordered. A particular vessel’s transmissions were then split into single trajectory observations subject to the condition that a time gap greater than 180 minutes between the time-ordered transmissions was observed. As with the scene representations, the trajectory observations used to train the model are 32×32 quadrat (2D grid) counts. The counts are applied to the temporally partitioned point patterns created by individual vessel transmissions within the observation area.

To test the discriminatory power of the anomaly detection method on simulated anomalies, counts were binarised as either 0 or 1 (zero counts remained 0, any other count was set to 1). This was done to reduce the dimensionality of the discrimination task: the simulation method is relatively simple and focused on producing realistic spatial patterns. Leaving counts in the data may make it unnecessarily easy for the model to discriminate between simulated and real observations because of limitations in simulation of individual cell count distributions. By binarising the data, we force the model to operate purely on spatial location of the observations which is the primary objective of the anomaly detection task.

An alternative to binarising the data for the discrimination tests would have been to increase the grid granularity to the point where each quadrat contains at the most one transmission count. This would have introduced a number of complexities with respect to having to alter the VAE architecture to account for the different data shape. Binarising the data avoids having to make changes to the model pipeline, significantly simplifying the problem of devising a reasonable quantitative test for anomaly detection efficacy.

In summary, in most of the experiments, the VAE was trained only on full quadrat counts of both multi-vessel scenes and single vessel trajectory representations. The binarised representations were only used in quantitative testing of the discriminatory power of the trained models between real observations and simulated anomalies. This is discussed in detail in section 6.1.2.

3.3 Synthetic Data

In order to do any sort of quantitative analysis of the trained model(s), either real anomalies or proxy anomalies are required. Given that labelled anomalies are not available and finding and labelling them by hand is beyond the scope of this thesis (given the significant effort required), synthetic data was generated in order to represent possible anomalous behaviour. Two types of anomaly were simulated: spatial anomalies representing unusual trajectories and ‘switching off’ representing vessels choosing to ‘go dark’ by switching off their AIS transceivers to avoid detection.

3.3.1 Spatial Anomaly Simulations

A simple simulation was designed to create random walks from one side of the observation grid to any other as in algorithm 2.

Algorithm 2 Vessel Simulation

```

1: procedure VESSELSIMULATOR
2:   Initialize simulation attributes
3:   RandomPath:
4:     Determine start and end sides of grid
5:     Randomly assign start and end points on chosen sides
6:   PathGeneration:
7:     while Vessel not at end point do
8:       Randomly decide if next move is random
9:       if Next move is random then
10:        Choose any adjacent cell as next position and increase grid count by 1
11:       else
12:        Move to adjacent cell closest to end point and increase grid count by 1
13:       end if
14:       Update vessel position
15:     end while
16:   AnomalyEffects:
17:     Introduce random effects in some cells
18:     if Faulty transponder then
19:       Randomly set counts to zero across the grid
20:     end if
21:   end procedure

```

An additional parameter (‘faulty transponder’) is added to set random cells on the path traced in the simulation to zero to make the point pattern more realistic (most observations are not a continuous path of cells with positive counts). Examples can be seen in the results section.

3.3.2 ‘Switching Off’ via Transmission Count Masking

Synthetic ‘switching off’ anomalies were created by taking real observations from the single vessel trajectory test set then removing transmissions from a randomly placed square in the observation area covering roughly half of the observation area. Whilst this is a fairly simple and somewhat arbitrary approach, it could be said to reasonably represent an area vessels did not want to be seen in whilst maintaining normal behaviour outside of the ‘switching off’ area.

4 Anomaly Detection Framework

In this section we describe the problem formulation; specifically how the VAE is used to perform the anomaly detection task.

4.1 Anomaly Detection Task

The objective of the framework is to be able to discriminate between normal vessel behaviour and unusual vessel behaviour. In this research we represent unusual behaviour with simulated vessel activity designed to approximate trajectories taking unusual paths (spatial anomalies) and vessels that purposefully deactivate their AIS transponders ('switching off' anomalies). We would also like to be able to identify real anomalous behaviour by ranking and examining observations with the highest reconstruction errors.

Vessel behaviour is represented by transmission counts in a spatial grid. So the anomaly detection task is discriminating between normal and abnormal distributions of counts in a two dimensional spatial grid. More formally we can describe the anomaly detection task as follows:

Given a dataset $\mathcal{D} = \{x_1, x_2, \dots, x_n\}$, where each $x_k \in \mathbb{Z}_+^{l \times m}$ represents a 2D grid of non-negative integer counts of AIS transmissions. Let $p(x)$ be the underlying probability distribution from which these grids are drawn.

Objective: Identify grids that deviate significantly from the expected distribution of grids in \mathcal{D} . This deviation is an indication of an anomaly. Formally, the objective is to learn a function:

$$f : \mathbb{Z}_+^{l \times m} \rightarrow \mathbb{R}$$

which assigns to each grid x_i a scalar score $s_i = f(x_i)$ such that:

1. s_i reflects the deviation of x_i from the typical grids in \mathcal{D} .
2. Higher values of s_i indicate a higher likelihood of x_i being an anomaly.

4.2 Proposed Methodology

The goal is to model the distribution $p(x)$ based on the observed grids in \mathcal{D} and then compute the likelihood $p(x_i)$ for a given grid. The inverse likelihood (or some function thereof) can be used as the anomaly score:

$$s_i = -\log p(x_i) \quad (4.1)$$

The grids with scores significantly higher than a certain threshold are flagged as anomalies. The exact threshold can be determined based on the desired trade-off between precision and recall in the context of the application.

Given the proposed solution is to use a VAE in order to model $p(x)$, an appropriate function for s_i in 4.1 would be the reconstruction loss.

The rationale is that a VAE trained mostly on ‘normal’ data will reconstruct normal observations effectively, resulting in a low reconstruction error. Anomalous observations on the other hand should generally have higher reconstruction errors by which they can be identified.

4.3 Anomaly Metric

The reconstruction loss, s_i , can be computed as:

$$s_i = \frac{1}{lm} \sum_{i=1}^l \sum_{j=1}^m (x_i(j, k) - \hat{x}_i(j, k))^2$$

Where:

- l and m are the dimensions of the 2D grid representing the observation.
- $x_i(j, k)$ denotes the count at position (j, k) in the observation.
- $\hat{x}_i(j, k)$ denotes the count at position (j, k) in the reconstructed observation.

Given the sparsity of observations (many zero counts in a typical observation), we need to update this loss measure to calculate the average loss per element, otherwise observations with very few transmissions tend to show low reconstruction losses owing to lack of data rather than reconstruction accuracy.

Define n_i as the number of non-zero grid counts in the i -th observation; n_i can be calculated as:

$$n_i = \sum_{j=1}^l \sum_{k=1}^m \mathbb{1}(x_i(j, k) \neq 0)$$

Where $\mathbb{1}(\cdot)$ is the indicator function, returning 1 if the condition inside the parenthesis is true, and 0 otherwise.

The adjusted reconstruction loss s'_i for sparse data would then be:

$$s'_i = \frac{1}{n_i} \sum_{j=1}^l \sum_{k=1}^m \mathbb{1}(x_i(j, k) \neq 0) (x_i(j, k) - \hat{x}_i(j, k))^2$$

This is the normalised mean squared (reconstruction) error, or just MSE for this model. To operationalise this model, a threshold value for is set for s'_i above which the observation is classed as anomalous.

4.4 VAE Architecture

The proposed method uses an encoder, latent space, decoder architecture as described in section 2.2. The encoder input layer flattens the 32 x 32 observations. It is composed of three layers of 512, 256 and 128 neurons with rectified linear unit (ReLU) activation functions. The decoder layer mirrors the encoder layers except that the output layer has sigmoid activation functions and reconstructs the pattern back into a 32 x 32 pattern. The latent space dimension is 128, and comprised of Gaussian distributions with priors set to a mean of zero and standard deviation of 1. See 5.1 for details of training methodology.

4.5 Evaluation Metrics

As discussed above, our objective is to discriminate between anomalous and normal observations. Given we have formulated the problem as a binary classification task, a suitable evaluation metric is the ROC AUC score (receiver operator characteristic area under curve).

This method is a suitable evaluation metric because it gives a probabilistic measure of the model's ability to discriminate on a randomly chosen anomalous observation vs. a randomly chosen normal observation based on rank ordering. In the context of anomaly detection we are usually concerned relative abnormality rather than any absolute measure. The main reasons for this are that abnormality is typically a function of rarity i.e. relative frequency of observation in a population. Secondly, even if we wanted to define an absolute threshold from first principles (incorporating domain knowledge) this is usually very difficult in practice and may not generalise.

A further benefit of the ROC AUC metric is that it is invariant to class distribution, meaning experimental results can be compared to practical results where the number of anomalous observations may be very small. The measure is also independent of the classification threshold. This is useful because different users may require different thresholds based on the utility of true vs. false positives. ROC AUC gives us a metric that applies across all thresholds.

Specifically, the ROC curve is a graphical representation of a binary classifier's performance across varying discrimination thresholds. It plots the true positive rate against the false positive rate for the probabilistic binary classification threshold across an interval of [0,1].

Mathematically, the AUC can be expressed as:

$$\text{AUC} = \int_0^1 \text{TPR}(t) dt$$

Where t is the decision threshold and TPR is the true positive rate for a given threshold t .

An AUC value of 0.5 is equivalent to random guessing, while a value of 1.0 is completely accurate.

In practice the user would define an anomaly threshold given the relative utility of model specificity vs. sensitivity.

5 Experimental Setup

5.1 Model Pipeline Summary

The experimental pipeline for training the VAEs can be summarised as follows:

1. Data pre-processing to transform groups of AIS observations into 32 x 32 grid counts (details can be found in section 3). The grid counts represent two types of observations:
 - Individual vessel trajectories.
 - Individual vessel trajectories on binarised counts (set to one if non-zero).
 - Multi-vessel time-boxed scenes.
2. The pre-processed data was split in training, validation, and test sets. ‘Switching off’ anomalies were simulated using the test set of individual vessel trajectories on binarised counts. Spatial anomalies were simulated from scratch (i.e. without the use of any other data).
3. Model training. Separate models were trained (using training and validation sets only) on each of the pre-processed training sets above resulting in:
 - A VAE trained on individual vessel trajectories.
 - A VAE trained on individual vessel trajectories with binarised counts.
 - A VAE trained on multi-vessel time-boxed scenes.

5.2 Data Pre-processing

As discussed in section 3, quadrat counts of AIS transmissions are generated for time boxed multi-vessel scenes and single vessel trajectories. In this experiment 32 x 32 grid representations were used to keep the complexity of the representations relatively low whilst providing enough detail from a potential user to make sense of vessel behaviour. On different spatial scales grid partitions of higher or lower granularity may be more suitable. The principle of introducing enough complexity to achieve the modelling objective, and not more, was followed.

Other preprocessing transforms were applied to remove extraneous observations and transform the data onto a scale which improves ease of training:

1. Removal of observations with no transmissions in the case of time-boxed multi-vessel scenes.

2. Transformation of transmission counts to a log scale: $x' = \log(1 + x)$.
3. Scaling of the resulting log transformed counts across the interval [0, 1] for model training using the scikit-learn ‘MinMaxScaler’ function (Pedregosa et al., 2011).

Each of the three pre-processed data sets were split into train, validation and test sets with 60:20:20 ratios. The differing methods of partitioning observations for individual trajectories and multi-vessel scenes resulted in differently sized data sets when measured in terms of observation counts. The total number of observations of multi-vessel scenes was $\approx 37K$. For single vessel trajectories it was $\approx 159K$. Whilst there are no rules or guarantees for required data set sizes in deep learning, these data sets are likely of sufficient scale to effectively train the models. This is most likely the case for single vessel trajectories which are simpler and more numerous than multi-vessel scenes. For comparison the commonly used MNIST data set of 28 x 28 pixel images has 70K examples (Deng, 2012) and has been used to successfully train and test many deep learning models of comparable complexity to this problem.

5.3 Model Training

Each model was trained with the following method:

- The data was partitioned into mini-batches of 256 observations each.
- The Adam (adaptive moment estimation) optimizer algorithm was used to implement stochastic gradient descent.
- The loss function is the SGVB estimator as in equation 2.5.
- The models are trained on the training set (model parameters are updated vs. training set loss) and evaluated on the validation sets (used to monitor and halt the training process by evaluating loss on the validation set).
- Early stopping was used to end training.

5.4 Model Pipeline Outputs

The pipeline produced trained models which were saved in a Keras (Chollet et al., 2015) format for ease of saving and loading. Other information was produced to aid development and setting of hyperparameters. This included learning curves that plotted training and validation loss at each training epoch.

5.5 Implementation Details

The code was developed and run on a Jupyter notebook (Kluyver et al., 2016) using Python version 3.9 and variety of open source Python packages including TensorFlow

(Abadi et al., 2015) and Keras (Chollet et al., 2015) for deep learning as well as standard data analysis and mathematical packages such as Pandas (pandas development team, 2020) and Numpy (Harris et al., 2020). Scikit-learn (Pedregosa et al., 2011) was used for a number of tasks in the model pipeline and Matplotlib (Hunter, 2007) was used to produce graphics. The hardware used was a MacBook M1 Pro with 16GB of RAM. The Apple ‘Metal’ compute API was used and found to significantly reduce model training and inference times by allowing TensorFlow direct access to the MacBook’s GPU. End to end data pre-processing, model training, and model analysis took around 45 minutes for the most compute intensive pipeline.

6 Results and Analysis

Evaluation of the trained models can be summarised as follows:

- A qualitative comparison between observations and reconstructions was performed on a sample of test observations for both individual vessel trajectories and multi-vessel scenes. Examples are plotted in figures 6.1 and 6.2. Subplots for reconstructions with median errors and the largest errors are presented.
- A qualitative comparison between observations and reconstructions was performed for both simulated spatial anomalies (figure 6.3) and switching off anomalies (figure 6.4). These are randomly sampled and do not represent reconstructions from any particular part of the error distribution.
- Histograms of the reconstruction losses on the training and test set plus either the simulated spatial anomaly (figure 6.5a) or the simulated switching off sets (figure 6.5b).
- Receiver-operator curve (ROC) plots with the resulting area under curve (AUC) scores were generated for each model to measure their discriminative power between the binarised test observations and simulated anomaly data (figure 6.6).

6.1 Performance Evaluation

6.1.1 Visualisation & Qualitative Analysis

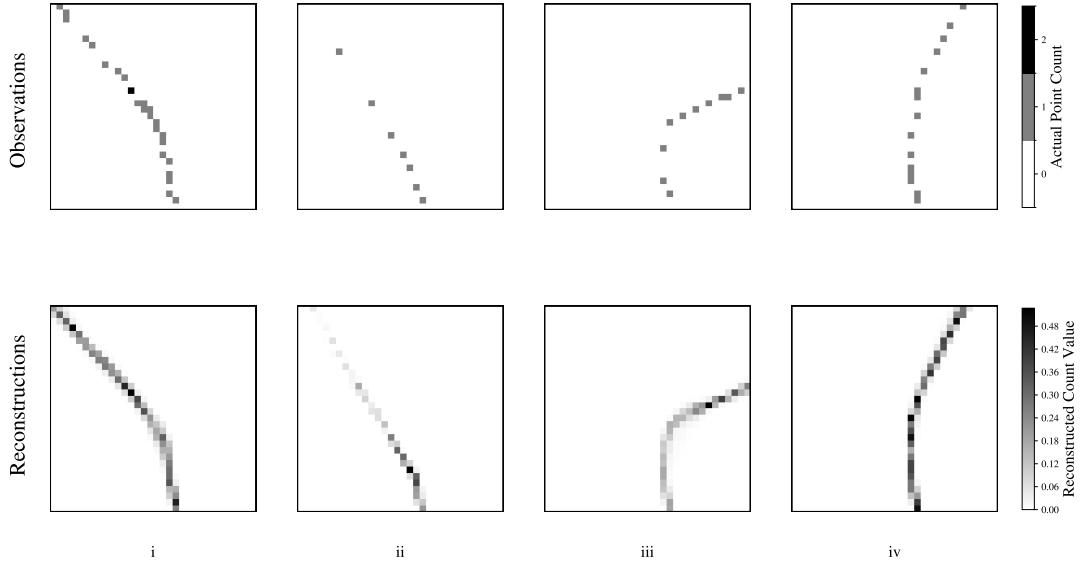
Before a quantitative analysis of the results are performed, visualisation of a small sample of results is provided in order to develop an intuition for the model's behaviour with respect to reconstructing observations.

First we look at reconstructions of the test data set. The examples are chosen by selecting observations closest to the median reconstruction error and those with the worst reconstruction errors. Figure 6.1a shows trajectory observations of real AIS transmissions in the test set (top row) and the corresponding reconstructions (bottom row) with median reconstruction errors. They are intended to represent ‘typical’ reconstructions, albeit with the proviso that there is significant variation in observations with similar reconstruction errors. Note that the observations are oriented with the nearby river mouth at the bottom of the grid which is the reason why so many trajectories pass through this area.

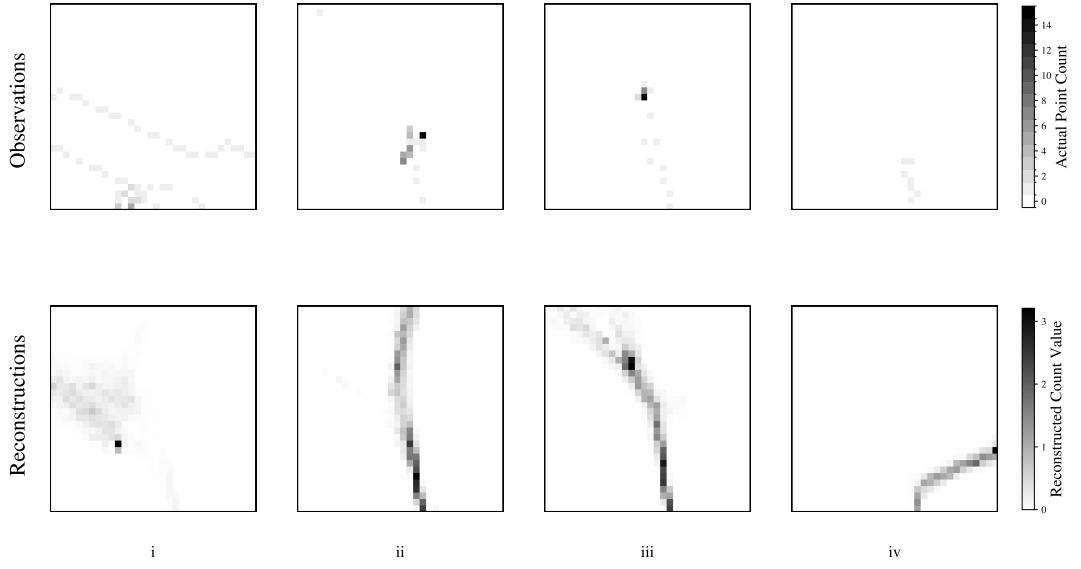
In contrast, figure 6.1b shows a sample of the observations with the highest reconstruction errors. The observations do not look like typical vessel trajectories; that is

to say vessels that appear to be efficiently traversing the observation area as if heading for a particular destination. The VAE reconstructs these observations with relatively diffuse plumes representing the uncertainty around it's prediction. Qualitatively, we could describe these observations as anomalous in the sense they are not typical vessel movements across the observation area.

This cursory analysis implies that the trained VAE is doing something useful with respect to producing good reconstructions of typical vessel behaviour and bad reconstructions of unexpected vessel behaviour. This is what is required for anomaly detection.



- (a) Reconstructed trajectory test data with median reconstruction errors. The plots show observations of normal trajectories, some with quite significant gaps in transmissions. The VAE reconstructs these as continuous tracks which correspond closely to the shape of the observations. This is what we'd expect from a well trained VAE.

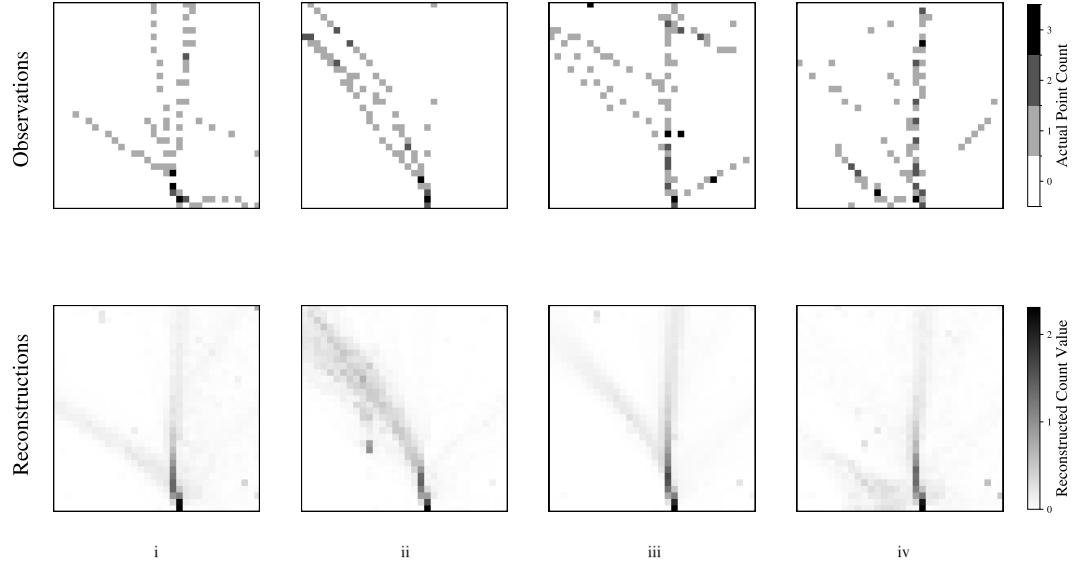


- (b) Reconstructed trajectory test data with high reconstruction errors. For observations ii, iii, and iv the model constructs plausible yet unlikely (given lack of evidence) trajectories with relatively diffuse structure. The model fails to generate a plausible trajectory for i given the highly anomalous observation.

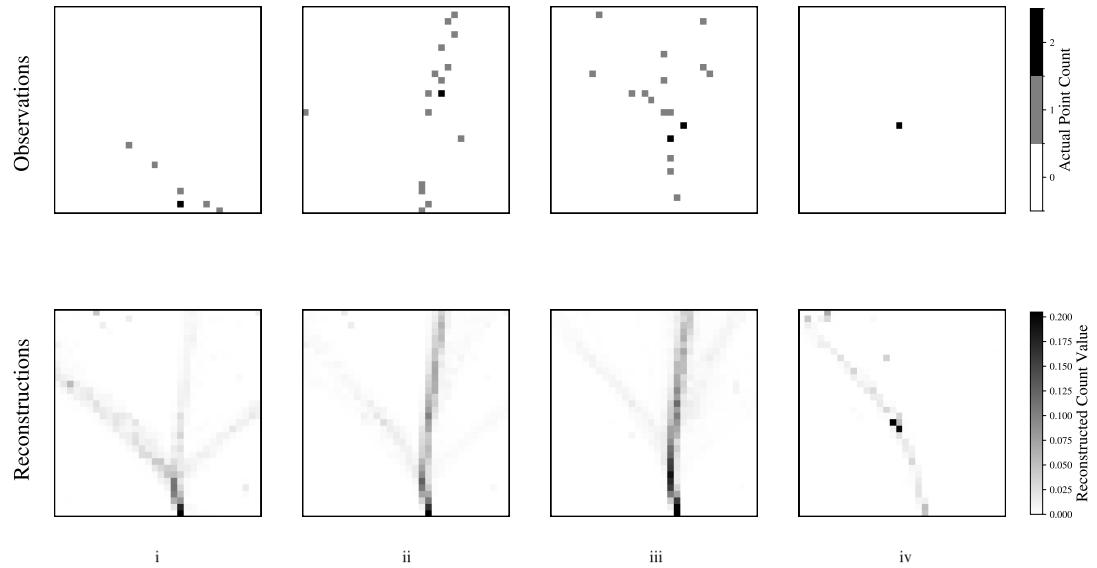
Figure 6.1: Single vessel trajectory test data: observations with reconstructions.

Similar results are produced for multi-vessel observations in figure 6.2. It can be seen that the relative sharpness of multi-vessel scenes is lower than that for single vessel trajectories, however the model appears to do a reasonable job of capturing realistic distributions of vessel traffic for median reconstruction losses (figure 6.2a).

Conversely, figure 6.2b shows reconstructions with the highest losses. The observations are somewhat unusual in that they don't correspond to reasonably well ordered traffic patterns. None-the-less, the model tries to fit traffic distributions to these observations. Some are plausible (with little evidence) whereas others highly unlikely.



(a) Reconstructed multi-vessel scene test data with median reconstruction errors. The model generates plausible traffic distributions for the given observations,



(b) Reconstructed multi-vessel scene test data with the high reconstruction errors. The model generates plausible traffic distributions for ii and iii but they have high reconstruction losses given the sparseness of the observations. The distributions for i and iv have very little evidence: i seems very implausible and observation iv could support any of many possible trajectories that could pass through it.

Figure 6.2: Multi-vessel scene test data: observations with reconstructions.

Next we look at reconstructions of synthetic anomaly data. Figure 6.3 shows reconstructions of simulated spatial anomalies. The simulated single-vessel trajectory observations have random start and end points meaning they do not follow typical traffic paths in the observation area. This results in some obviously low quality reconstructions (with correspondingly high error). The VAE clearly struggles to generate plausible trajectories when vessels do not traverse the highest traffic density (such as the area close to the river mouth just below the bottom middle of the observation area).

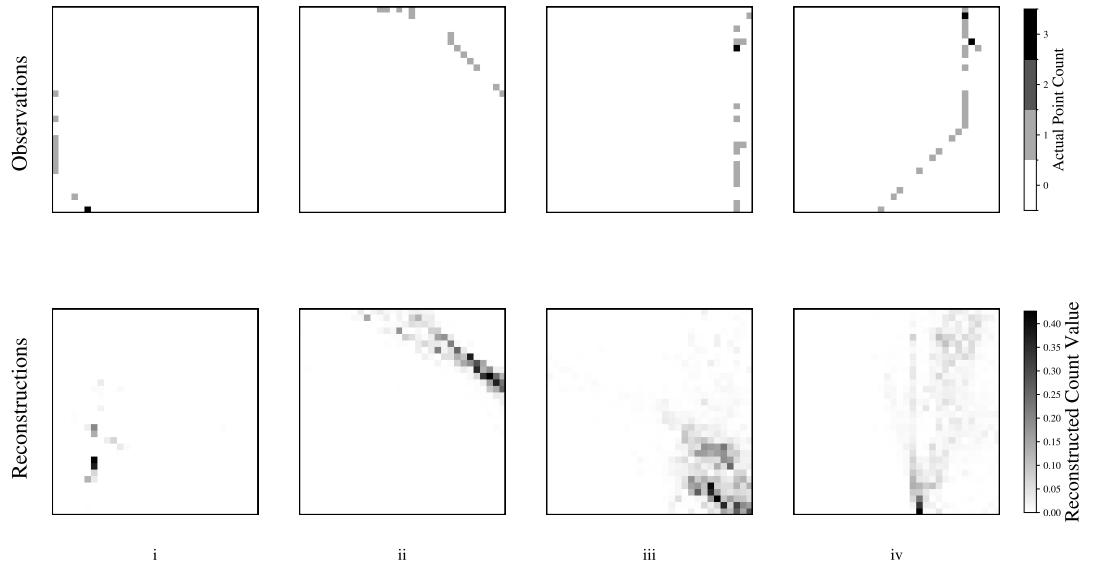


Figure 6.3: Reconstructed anomaly data (simulated spatial anomalies on single vessel trajectories). iii is a good example of the model’s inability to reconstruct trajectories that do not pass though the central main traffic area of the observation area. This is what we expect to see for such an anomalous trajectory.

Synthetic ‘switching off’ anomalies are created by taking real observations then removing transmissions from a central square in the centre of the observation as explained in section 3.3.2. Figure 6.4 shows reconstructions on simulated ‘switching off’ observations. It can be seen that the VAE is effective at generating plausible reconstructions when the start and the end of the vessel trajectory is in the observation. The model effectively fills in the blanks generated by the random mask.

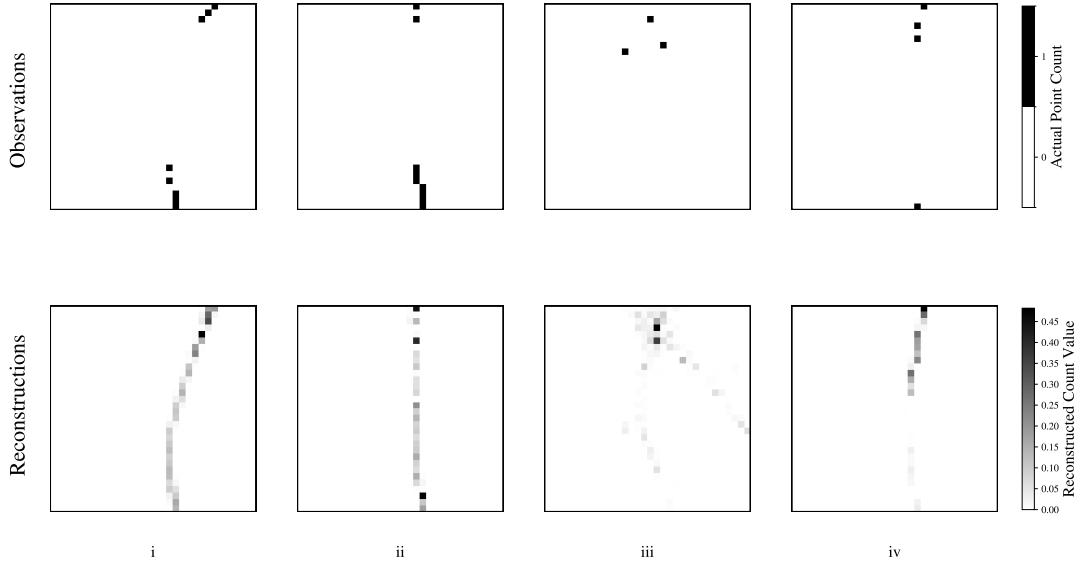
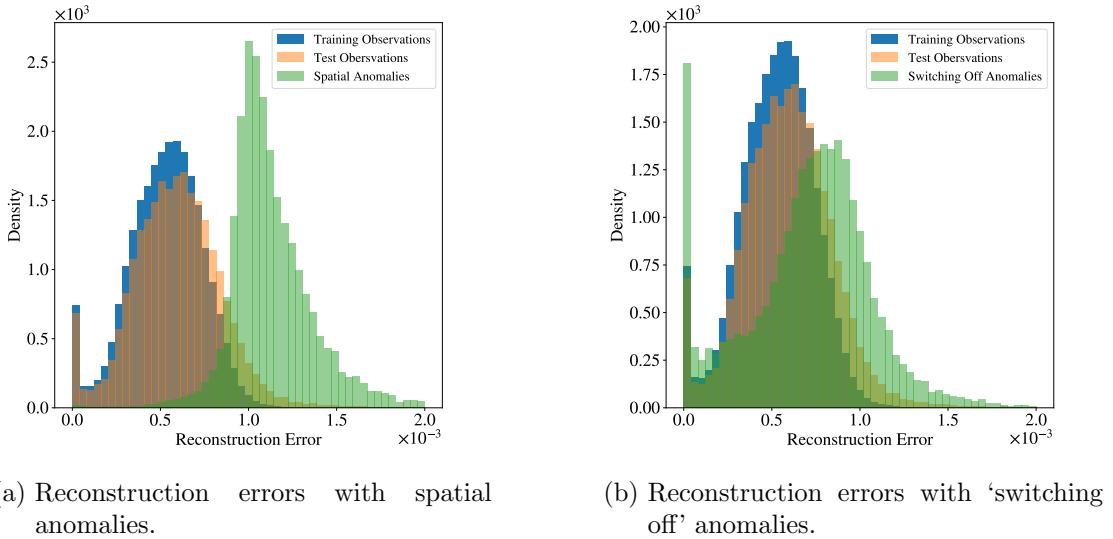


Figure 6.4: Reconstructed simulated 'switching off' anomaly observations. In i, ii and iv the VAE fills in the masked area with highly plausible trajectories giving rise to a relatively high reconstruction loss indicative of an anomaly. Observation iii is highly ambiguous resulting in a relatively implausible reconstruction.

6.1.2 Quantitative Analysis

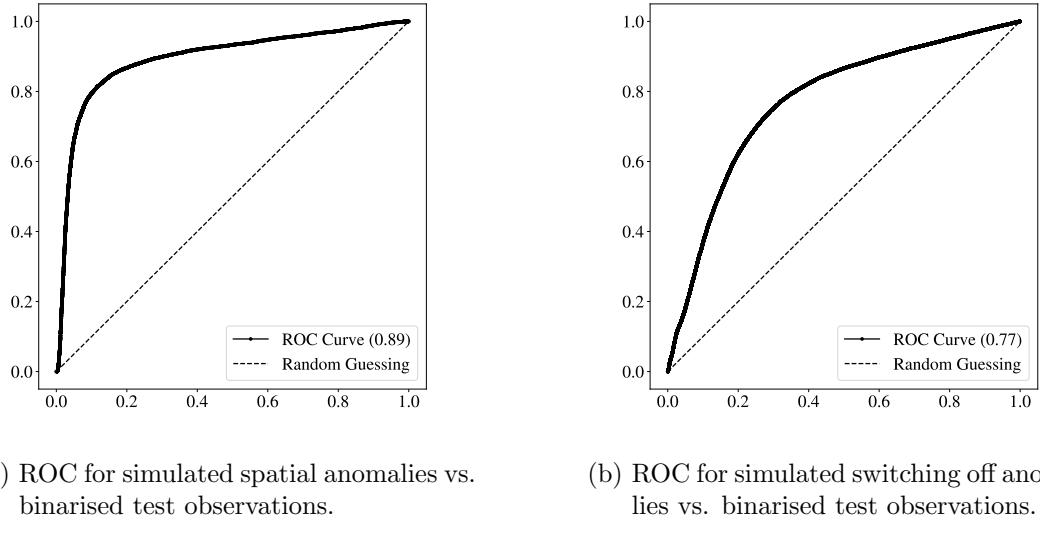
We can summarise the results by plotting histograms of normalised reconstruction errors. Figure 6.5 shows reconstruction errors on the train and test sets alongside reconstruction errors on spatial anomalies and switching off anomalies. The reconstruction errors are higher for the synthetic anomaly data with good separation for the spatial anomalies.



(a) Reconstruction errors with spatial anomalies.

(b) Reconstruction errors with 'switching off' anomalies.

Figure 6.5: Reconstruction errors on binarised train & test observations and simulated anomalies.



(a) ROC for simulated spatial anomalies vs. binarised test observations.

(b) ROC for simulated switching off anomalies vs. binarised test observations.

Figure 6.6: ROC AUC curves for model discrimination between observed and synthetic anomalies (binarised observations).

The VAE achieves ROC AUCs of 0.89 and 0.77 for spatial and 'switching off' anomalies respectively demonstrating good discriminative power between the synthetic anomalies and observed data (see figure 6.6).

7 Discussion

7.1 Interpretation of Results

The ROC AUC scores for the trained models ability to discriminate between normal behaviour and simulated anomalies are evidence that the proposed method operates in the way it is designed to, and is able to discriminate between these data sets. This evidence could justify further work to operationalise a practical system, but doesn't allow us to make any strong claims about the efficacy of this approach. It could be said that the simulated data provides a 'proof of concept' of this approach, but little more, especially when we critique the simulation methods (see section 7.2.1 below).

7.1.1 Qualitative Discussion

The qualitative results show that the models behave as expected, generating accurate reconstructions of normal observations and inaccurate reconstructions of anomalous behaviour. Whilst these results are very limited for the sake of brevity, they are parsimonious yet informative representations of the input and output data allowing us to reason over potential failure modes and discriminative weakness in the model. Possible further work involving generating reconstructions by traversing the latent space would offer a principled yet intuitive method to analyse the model behaviour.

The analysis presented here is limited to a relatively small, vessel rich location. Applying the method to sparsely traversed areas of the open sea would most likely require rescaling the problem both spatially and temporally. The results shown above give no indication as to impact this may have on model efficacy. Increasing training data by lengthening the observation period may have implications for validity of training data given marine traffic is likely to be statistically non-stationary (for example, the war in Ukraine has triggered a major re-configuration of global cargo routes given sanctions and risks to vessels). The use of more fine or coarse grained observation grids has several implications for the method: the encoder and decoder networks directly interface with the processed data. Going from a 32×32 to a 64×64 grid changes the machine learning task in a number of ways. The obvious one is the number of inputs increases non-linearly. This would require dimensional adjustment to the whole network except perhaps the number of latent dimensions.

A more fine grained representation of the data could have subtle implications for what the model learns. The classic example in deep learning for image classification is the sensitivity of models to fine grained object surface texture rather than large scale structure. Often networks need to be coerced into learning large scale structure because of a tendency to greedily over-fit to textures. In the case of AIS data, we may be more or

less interested in large scale trajectory shapes which are more effectively captured with low resolution grids than fine grained behaviour that may capture (for example) events of interest such as ship to ship transfers. In the case of ship to ship transfers we may be more interested in a multi-vessel scene than individual trajectories. The point is that different representations are more suitable for both different geographical regions and for specific anomaly types. These preferences can all be considered hyperparameters of a fully generalisable system, but finding them is non-trivial and likely to come down to empirically determined heuristics.

7.1.2 Quantitative Discussion

Despite ostensibly strong discriminative performance as evidenced by the ROC AUC metric on the anomaly sets, caution must be invoked before making strong claims based on synthetic data. It may be that the simulated anomalies contain unintentional attributes that make them easy to differentiate vs. real observations. It is also possible that the simulated anomalies do not capture the differentiating attributes we would see in real anomalies. Further, the results may not hold for the real anomalies this data is intended to represent because we don't know exactly what they look like, and the simulations are idealised.

Ideally a data set of real anomalous events would be curated and used to test the approach, but this is beyond the scope of this thesis. There is a deeper epistemological issue with anomaly detection: we define anomalies in negative terms i.e. it is *not* normal behaviour so the set of possible anomalies is potentially astronomical. This means that any strong claims about any anomaly detection system need strong evidence, which in this instance is lacking.

7.2 Limitations and Challenges

7.2.1 Synthetic Data

The algorithm used to generate simulated anomalies was relatively simple with some questionable assumptions, especially with respect to random transmission occlusion and random movements of vessels. The ‘switching off’ area shape and position were completely arbitrary. Significantly more effort could be put into the simulation quality of both of these approaches. Whilst it is beyond the scope of this thesis to suggest a comprehensive approach to generating simulated anomalies, two approaches could have been tried:

- A set of real anomalies could have been identified from reconstruction losses as they were in the qualitative analysis. Such a data set of real anomalies could have been used as the basis to generate similar anomalies using various statistical or machine learning methods (even perhaps another VAE).
- Domain knowledge of how vessels operate and what actually happens when anomalies are generated could have been used in an attempt to generate more accurate

simulations.

7.2.2 Practical Challenges

The main practical challenge to implementing this approach to anomaly detection is the scale of the problem which can result in the requirement for large specialist big data tools and compute requirements. A global AIS dataset over time periods required for model training could run into many terabytes. Many local models would drive up compute requirements. Whilst modern cloud infrastructure ameliorates these challenges somewhat, it would still be costly, and training and inferencing many models simultaneously would likely be quite complex and potentially resource intensive.

Even in cases where users of this system are only concerned with a single location or small region, there remains the challenge of selecting the appropriate grid size and possibly having to re-scale the VAE architecture accordingly. Whilst it may be possible to apply simple heuristics to this, this was not systematically investigated in this research. Increasing the grid size could have a significant impact on training time and training data requirements causing further practical issues.

7.3 Future Research Directions

There are a number of open research directions that this research invites:

- A relatively simple VAE architecture was used in this research. The use of, for example, a 2D convolutional layer on first and last layers of the encoders and decoders respectively may improve accuracy, or be more effective for specific anomaly types. The use of different distributions to approximate the posterior in the latent space may be more effective.
- The addition of new channels to the basic grid count such as vessel velocity data would inject more information into the model, possibly improving performance.
- As discussed in the previous section, determining how to automatically choose the grid size and alter the VAE architecture for an arbitrary observation region would be necessary for widespread practical use of this method.
- Also discussed in the previous section: there are a number of ways in which trained model evaluation could be improved with respect to the use of synthetic or real anomalies.
- Use of latent space to understand attributes of the data and how these relate to specific anomalies. This could allow anomaly classification as well a detection with the possibility of improving detection of specific anomalies.

7.3.1 Implementation as Practical Tool

Turing scientific research into software that not only works, but people actually use, is incredibly challenging. Such software has to clear a number of hurdles: improve on whatever method is currently in use, have enough utility to justify the cost, and make life easier for the end user. The functional factors that will determine if these requirements are met are:

1. The system accuracy: if the model generates too many false positives users become saturated with investigations. If the model generates too few true positive, the utility of the system is too low to bother with.
2. The interpretability of the output once a ‘case’ is raised: this approach is designed to be intuitive given the spatial point plots. The reconstruction and original observation can be compared side by side (as in section 6.1.1). These could be superimposed on geographical maps, political maps, fisheries, shipping lanes etc. Historical analysis of previous anomalies could be integrated.
3. The ability to take action: a practical system should generate information and artefacts that allow the user to take action. This could be a communication, physical intervention or even evidence for a legal action.

8 Conclusion

8.1 Summary of Findings

In this thesis we have demonstrated an interpretable anomaly detection method with which we are able to discriminate between qualitatively normal and abnormal observations of vessel trajectories as well as synthetically generated anomalies using a trained VAE. Anomalous behaviour can be detected by looking at the observations with the highest reconstruction errors subject to some user defined threshold given utility preferences with respect to model specificity vs. sensitivity. The representation of the AIS data allows direct, intuitive, analysis of observed vessel dynamics and the VAE's attempt at reconstructing such observations.

8.2 Contributions to Anomaly Detection in Marine Vessel AIS Data

The method is highly interpretable and requires very little domain knowledge for feature engineering because it uses only general methods from spatial statistics by treating the observations as point patterns. This has the further benefit of keeping the approach firmly in the domain of applied mathematics rather than imposing domain specific engineering knowledge onto the problem. Practically it remains to be seen if this is a good thing, but from a scientific perspective it keeps method transparent, parsimonious and allows others to build on the work without extensive implementation details.

8.3 Practical Applications and Implications

The interpretability and general applicability of this method means that it has potential to be developed into a practical tool for detecting general or specific anomalies in any location with enough historical marine traffic data to train a VAE. Whilst VAEs are not new technologies, industrial applications of generative machine learning models remain limited, so this method may be useful in other spatial analysis contexts such as road and air traffic monitoring and potentially for monitoring crowds for safety and security applications.

References

- Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL <https://www.tensorflow.org/>. Software available from tensorflow.org.
- AccessAIS. URL <https://coast.noaa.gov/digitalcoast/tools/ais.html>.
- Virginia Fernandez Arguedas, Giuliana Pallotta, and Michele Vespe. Automatic generation of geographical networks for maritime traffic surveillance. In *17th International Conference on Information Fusion (FUSION)*, pages 1–8, July 2014.
- Ahmed Aziz, Pietro Tedeschi, Savio Sciancalepore, and Roberto Pietro. SecureAIS - Securing Pairwise Vessels Communications. June 2020. doi: 10.1109/CNS48642.2020.9162320.
- Adrian Baddeley, Ege Rubak, and Rolf Turner. *Spatial point patterns: methodology and applications with R*. CRC press, 2015.
- National Oceanic and Atmospheric Administration (NOAA) Bureau of Ocean Energy Management (BOEM). Marinecadastre.gov accessais web tool. URL <https://marinecadastre.gov/accessais/>.
- Emanuele Carlini, Vinícius Monteiro, Amilcar Soares, Mohammad Etemad, Bruno Brandoli, and Stan Matwin. Uncovering vessel movement patterns from AIS data with graph evolution analysis. January 2020.
- Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):1–58, July 2009. ISSN 0360-0300, 1557-7341. doi: 10.1145/1541880.1541882. URL <https://dl.acm.org/doi/10.1145/1541880.1541882>.
- F. Chollet. *Deep Learning with Python, Second Edition*. Manning, 2021. ISBN 9781617296864. URL <https://books.google.co.uk/books?id=XHpKEAAQBAJ>.

- Francois Chollet et al. Keras, 2015. URL <https://github.com/fchollet/keras>.
- Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine*, 29(6):141–142, 2012.
- Hongda Duan, Fei Ma, Lixin Miao, and Canrong Zhang. A semi-supervised deep learning approach for vessel trajectory classification based on AIS data. *Ocean & coastal management*, 218:106015–, 2022. ISSN 0964-5691. doi: 10.1016/j.ocecoaman.2021.106015. Publisher: Elsevier Ltd.
- Jessica H. Ford, David Peel, David Kroodsma, Britta Denise Hardesty, Uwe Rosebrock, and Chris Wilcox. Detecting suspicious activities at sea based on anomalies in Automatic Identification Systems transmissions. *PLOS ONE*, 13(8):e0201640, August 2018. ISSN 1932-6203. doi: 10.1371/journal.pone.0201640. URL <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0201640>. Publisher: Public Library of Science.
- Global Fishing Watch. Global fishing watch. 2016-2023. URL www.globalfishingwatch.org.
- Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016. <http://www.deeplearningbook.org>.
- Ville Hakola. Vessel tracking (ais), vessel metadata and dirway datasets, 2020. URL <https://dx.doi.org/10.21227/j3b5-es69>.
- Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL <https://doi.org/10.1038/s41586-020-2649-2>.
- Jia Hu, Kuljeet Kaur, Hui Lin, Xiaoding Wang, Mohammad Mehedi Hassan, Imran Razzak, and Mohammad Hammoudeh. Intelligent Anomaly Detection of Trajectories for IoT Empowered Maritime Transportation Systems. *IEEE transactions on intelligent transportation systems*, pages 1–10, 2023. ISSN 1524-9050. doi: 10.1109/TITS.2022.3162491. Publisher: IEEE.
- J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.
- IEEE Data Port. URL www.ieee-dataport.org.

- Pablo Kaluza, Andrea Kölzsch, Michael T. Gastner, and Bernd Blasius. The complex network of global cargo ship movements. *Journal of The Royal Society Interface*, 7(48):1093–1103, January 2010. doi: 10.1098/rsif.2009.0495. URL <https://royalsocietypublishing.org/doi/10.1098/rsif.2009.0495>. Publisher: Royal Society.
- Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Safia Abdalla, and Carol Willing. Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides and B. Schmidt, editors, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press, 2016.
- David A. Kroodsma, Juan Mayorga, Timothy Hochberg, Nathan A. Miller, Kristina Boerder, Francesco Ferretti, Alex Wilson, Bjorn Bergman, Timothy D. White, Barbara A. Block, Paul Woods, Brian Sullivan, Christopher Costello, and Boris Worm. Tracking the global footprint of fisheries. *Science*, 359(6378):904–908, February 2018. doi: 10.1126/science.aoa5646. URL <https://www.science.org/doi/10.1126/science.aoa5646>. Publisher: American Association for the Advancement of Science.
- Rikard Laxhammar. *Anomaly detection in trajectory data for surveillance applications*. PhD thesis, Örebro universitet, 2011.
- Yann LeCun, Y. Bengio, and Geoffrey Hinton. Deep Learning. *Nature*, 521:436–44, May 2015. doi: 10.1038/nature14539.
- Huanhuan Li, Jingxian Liu, Kefeng Wu, Zaili Yang, Wen Liu, and Naixue Xiong. Spatio-Temporal Vessel Trajectory Clustering Based on Data Mapping and Density. *IEEE Access*, PP, August 2018. doi: 10.1109/ACCESS.2018.2866364.
- Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017. URL <http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf>.
- Jorge Mateu and Abdollah Jalilian. Spatial point processes and neural networks: A convenient couple. *Spatial Statistics*, 50:100644, August 2022. ISSN 2211-6753. doi: 10.1016/j.spasta.2022.100644. URL <https://www.sciencedirect.com/science/article/pii/S221167532200029X>.
- Brian Murray and Lokukaluge Prasad Perera. A dual linear autoencoder approach for vessel trajectory prediction using historical AIS data. *Ocean Engineering*, 209: 107478, August 2020. ISSN 0029-8018. doi: 10.1016/j.oceaneng.2020.107478. URL <https://www.sciencedirect.com/science/article/pii/S0029801820304959>.

- Brian Murray and Lokukaluge Prasad Perera. An AIS-based deep learning framework for regional ship behavior prediction. *Reliability engineering & system safety*, 215:107819–, 2021. ISSN 0951-8320. doi: 10.1016/j.ress.2021.107819. Place: Barking Publisher: Elsevier Ltd.
- The pandas development team. pandas-dev/pandas: Pandas, February 2020. URL <https://doi.org/10.5281/zenodo.3509134>.
- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. ” why should i trust you?” explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.
- Branko Ristic. Detecting Anomalies from a Multitarget Tracking Output. *IEEE Transactions on Aerospace and Electronic Systems*, 50(1):798–803, January 2014. ISSN 1557-9603. doi: 10.1109/TAES.2013.130377. Conference Name: IEEE Transactions on Aerospace and Electronic Systems.
- Sandeep Kumar Singh and Frank Heymann. Machine Learning-Assisted Anomaly Detection in Maritime Navigation using AIS Data. *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pages 832–838, April 2020. doi: 10.1109/PLANS46316.2020.9109806. URL <https://ieeexplore.ieee.org/document/9109806/>. Conference Name: 2020 IEEE/ION Position, Location and Navigation Symposium (PLANS) ISBN: 9781728102443 Place: Portland, OR, USA Publisher: IEEE.
- Martin Svanberg, Vendela Santén, Axel Hörteborn, Henrik Holm, and Christian Finnsgård. AIS in maritime research. *Marine Policy*, 106:103520, August 2019. ISSN 0308-597X. doi: 10.1016/j.marpol.2019.103520. URL <https://www.sciencedirect.com/science/article/pii/S0308597X18309667>.
- Michele Vespe, Ingrid Visentini, Karna Bryan, and Paolo Braca. Unsupervised learning of maritime traffic patterns for anomaly detection. volume 2012, pages 1–5, May 2012. doi: 10.1049/cp.2012.0414.
- Zhihuan Wang, Christophe Claramunt, and Yinhai Wang. Extracting Global Shipping Networks from Massive Historical Automatic Identification System Sensor Data: A Bottom-Up Approach. *Sensors*, 19(15):3363, January 2019. ISSN 1424-8220. doi: 10.3390/s19153363. URL <https://www.mdpi.com/1424-8220/19/15/3363>. Number: 15 Publisher: Multidisciplinary Digital Publishing Institute.
- Kevin Webster. Course lecture notes: Deep Learning (week 10) Variational autoencoder. Machine Learning and Data Science MSc. Imperial College. January 2023.

- Tom White. Sampling generative networks. *arXiv preprint arXiv:1609.04468*, 2016.
- Konrad Wolsing, Linus Roepert, Jan Bauer, and Klaus Wehrle. Anomaly Detection in Maritime AIS Tracks: A Review of Recent Approaches. *Journal of Marine Science and Engineering*, 10(1):112, January 2022. ISSN 2077-1312. doi: 10.3390/jmse10010112. URL <https://www.mdpi.com/2077-1312/10/1/112>. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute.
- Lei Xie, Tao Guo, Jiliang Chang, Chengpeng Wan, Xinyuan Hu, Yang Yang, and Changkui Ou. A Novel Model for Ship Trajectory Anomaly Detection Based on Gaussian Mixture Variational Autoencoder. *IEEE Transactions on Vehicular Technology*, pages 1–10, 2023. ISSN 1939-9359. doi: 10.1109/TVT.2023.3284908. Conference Name: IEEE Transactions on Vehicular Technology.
- Liye Zhang, Qiang Meng, and Tien Fang Fwa. Big AIS data based spatial-temporal analyses of ship traffic in Singapore port waters. *Transportation Research Part E: Logistics and Transportation Review*, 129:287–304, September 2019. ISSN 1366-5545. doi: 10.1016/j.tre.2017.07.011. URL <https://www.sciencedirect.com/science/article/pii/S1366554516309516>.
- Rong Zhen, Yongxing Jin, Qinyou Hu, Zheping Shao, and Nikitas Nikitakos. Maritime Anomaly Detection within Coastal Waters Based on Vessel Trajectory Clustering and Naïve Bayes Classifier. *The Journal of Navigation*, 70(3):648–670, May 2017. ISSN 0373-4633, 1469-7785. doi: 10.1017/S0373463316000850. URL <https://www.cambridge.org/core/journals/journal-of-navigation/article/abs/maritime-anomaly-detection-within-coastal-waters-based-on-vessel-trajectory-clustering-and-naive-bayes-classifier/650D7D7672E0053FA532D9294A734F09>. Publisher: Cambridge University Press.