*Predicting User Demographics using Navigational Behaviour in a Video Game*

A study submitted in partial fulfilment

of the requirements for the degree of

*Data Science MSc*

at

THE UNIVERSITY OF SHEFFIELD

by

*Keir James Elliott-Spencer*

*220251958*

Word Count: 13779 August 2023

# Abstract

**Background:** Analysing trajectories has become an increasingly fascinating field of study, thanks to the exponential growth of available and increasingly accurate trajectory data in our lives. With the emergence of more complex machine learning methods, able to discern complex patterns of navigational behaviour, the potential to understand human movement and differences in cognition between demographics has significantly expanded. Whilst current literature has explored navigational pattern differences across demographics such as age and gender, a lack of research into the interplay of these factors is evident, limiting the comprehensive understanding of the factors influencing navigational behaviours and cognitive abilities.

**Aims:** Therefore, this project set out with the aim of using trajectory data to explore to what extent combined age/gender demographics can be predicted based on differences in user navigational behaviours. Additionally, this project aimed to utilise two separate models to allow for comparisons between the influence of temporal and spatial dependencies.

**Methods:** This study utilised anonymised data obtained from Sea Hero Quest (a mobile navigation game) in a multi-class classification approach, using a Convolutional Neural Network (CNN), and a Temporal 2 Space Neural Network (Temp2SpaceNN), to predict the 10 combined age/gender labels. The CNN analysed trajectory data at two scopes looking for immediate patterns, as well as longer-term patterns in the data using convolutional operations to achieve this. The Temp2SpaceNN built on the foundation of the temporal CNN however mapped these learned patterns to a graph structure, allowing for more complex spatial patterns to be learnt to help better predict demographic labels.

**Results:** The models resulted accuracies of 9% and 21% respectively, with the temporal CNN performing worse than that of guessing by random chance, and the Temp2SpaceNN performing over 2x better than random chance. The difference of architecture between the models, combined with the results, indicated that differences in spatial dependencies of navigation behaviour were more significant between combined age/gender populations. The lack of overall strong model accuracy suggested either a lack of significant differences in navigation patterns between groups, or a lack in the models’ abilities to uncover these.

**Conclusion:** Overall, this study highlights the complexity of analysing trajectory data, highlighting the complex interplay of biological and environmental factors influencing the ability to discern clean differences in patterns between demographic populations. The limited predictive success underscores the need for further research in this area.

**Key Words/Phrases:** Trajectory analysis, Machine learning, Convolutional Neural Networks, Sea Hero Quest.

# Acknowledgements

First, I would like to express my sincere gratitude to my supervisor, Mengdie Zhuang, the module staff, and my university in general for their help and support despite what has been quite a difficult and lengthy master’s dissertation project.

I want to thank my parents, who have supported me through this project and through all the emotions that I have faced during it, whether in happiness or frustration, for which there has been lots of both.

Finally, I am deeply grateful to all my friends, who have been my virtual companions throughout this project and my degree. Letting me sit in call with you whilst I am working away has been a great anchor for me, keeping me tethered to the joy of social interaction, and preventing a sense of isolation, despite my days consumed by coding, research, and writing.

Table of Contents

[Abstract 2](#_Toc160046641)

[Acknowledgements 3](#_Toc160046642)

[Figure Legend 6](#_Toc160046643)

[Table Legend 7](#_Toc160046644)

[1 Introduction 8](#_Toc160046645)

[1.1 Navigational Abilities 8](#_Toc160046646)

[1.2 Sea Hero Quest 9](#_Toc160046647)

[1.3 Alzheimer’s Disease: Cognition and Health 10](#_Toc160046648)

[1.4 The Predictive Potential of Navigational Behaviour: Age and Gender 11](#_Toc160046649)

[1.5 Methods of Trajectory Analysis 11](#_Toc160046650)

[1.6 Aims and Objectives 16](#_Toc160046651)

[1.7 Key Definitions 17](#_Toc160046652)

[2 Literature Review 17](#_Toc160046653)

[2.1 Existing SHQ Research and Cognition 17](#_Toc160046654)

[2.2 Deep Learning models and Trajectory Analysis 17](#_Toc160046655)

[2.3 Summary 18](#_Toc160046656)

[3 Methodology 20](#_Toc160046657)

[3.1 Data Search Strategy and Collection 20](#_Toc160046658)

[3.2 Build 20](#_Toc160046659)

[3.3 Structure of Sea Hero Quest Data 20](#_Toc160046660)

[3.4 Data Pre-processing and Cleaning 22](#_Toc160046661)

[3.41 Data streaming 22](#_Toc160046662)

[3.42 Demographic analysis 23](#_Toc160046663)

[3.43 Discretisation of demographic variables 24](#_Toc160046664)

[3.44 Reconstructing directional information 27](#_Toc160046665)

[3.45 Resampling trajectory data 28](#_Toc160046666)

[3.46 Data sampling 31](#_Toc160046667)

[3.5 Model Architecture 32](#_Toc160046668)

[3.51 Model Selection 32](#_Toc160046669)

[3.52 Temporal model 32](#_Toc160046670)

[3.53 Defining a graph 34](#_Toc160046671)

[3.54 Spatial model 38](#_Toc160046672)

[3.6 Model Training, Validation, and Testing 40](#_Toc160046673)

[3.61 Hardware 40](#_Toc160046674)

[3.62 Dataset sampling 40](#_Toc160046675)

[3.63 Training and validation 40](#_Toc160046676)

[2.64 Hyperparameter tuning 42](#_Toc160046677)

[3.65 Testing 43](#_Toc160046678)

[4 Results 45](#_Toc160046679)

[4.1 Training Performance: Cross Entropy Loss and Accuracy 45](#_Toc160046680)

[4.2 Temporal Convolutional Neural Network Results 46](#_Toc160046681)

[4.3 Spatial Temp2Space Neural Network Results 48](#_Toc160046682)

[5 Discussion 51](#_Toc160046683)

[5.1 Training Performance 51](#_Toc160046684)

[5.2 Temporal and Spatial Models: Insights and Comparisons 51](#_Toc160046685)

[6 Conclusion 54](#_Toc160046686)

[6.1 Summary 54](#_Toc160046687)

[6.2 Limitations of the Report 55](#_Toc160046688)

[6.3 Ethical Considerations 56](#_Toc160046689)

[References 57](#_Toc160046690)

[Appendices 63](#_Toc160046691)

[Appendix A 63](#_Toc160046692)

[Appendix B 65](#_Toc160046693)

# Figure Legend

*Figure 1: A section of the map of the London Underground.*

*Figure 2: A visual representation of the visual style of Sea Hero Quest, showing the user controlling a boat in navigation tasks. From: (alzheimersresearchuk, n.d.).*

*Figure 3: An image of a Sea Hero Quest map (specifically level 43’s map) that are shown to users at the beginning of each level. From: (Coutrot et al., 2019).*

*Figure 4: A visualisation of the structure of a neural network in deep learning, showing nodes arranged in 5 layers (1 input, 3 hidden, 1 output), with edges connecting the nodes of one layer to the next. From: (IBM, 2023).*

*Figure 5: An visual illustration of the convolutional layers of a Convolutional Neural Network, acted upon a raw signal. From: (Shenfield & Howarth, 2020).*

*Figure 6: An example illustration of a simple graph, including labels of both nodes and edges. The graph is composed of 5 nodes, with 7 edges connecting these nodes in the graph structure. From: (Ravikiran, 2023).*

*Figure 7: An image of the level 56 Sea Hero Quest map. From: (Coutrot et al., 2019).*

*Figure 8: A dual line plot comparison of 2 example trajectories (A and B) of the trajectory dataset studied in the project, plotting x vs y.*

*Figure 9: A histogram of the frequency distribution of ages ranging from 16-99.*

*Figure 10: A histogram of the frequency distribution of the amount of previous attempts, with a value of 1 indicating the user had no prior experience with the level.*

*Figure 11: A line plot of 5 equally spread gaussian distributions each with a Peak Probability Density (PPD) of 1, that represent each of the 5 age groups. The means of the distributions are 19.0 (blue), 31.5 (orange), 44.0 (green), 56.5 (red), and 69.0 (purple) respectively.*

*Figure 12: A histogram of the frequency distribution of 5 age bins (age\_0, age\_1, age\_2, age\_3, and age\_4).*

*Figure 13: A histogram of the frequency distribution of the 10 combined (2) gender and (5) age bins, for a total of 10 demographic sets: f-age\_0, f-age\_1, f-age\_2, f-age\_3, f-age\_4, m-age\_0, m-age\_1, m-age\_2, m-age\_3, and m-age\_4.*

*Figure 14: A plot of the original (green) and gaussian smoothed, reconstructed (orange) directional component cos(θ) of an example trajectory, over time (t).*

*Figure 15: A plot of the original (green) and gaussian smoothed, reconstructed (orange) directional component sin(θ) of an example trajectory, over time (t).*

*Figure 16: A histogram of the frequency distribution of the length of all trajectories in 100 bins. An x limit of 0 – 2000 was applied for ease of viewing, owing to the lack of any discernible bars past a length of 1250.*

*Figure 17: A multiplot of 2 original (blue) vs resampled (orange) example trajectories – the shortest and longest recorded from the trajectory dataset studied in the project, plotting x vs y.*

*Figure 18: A multiplot of an investigation into the shortest trajectory’s original (blue) vs resampled (orange) x and y components, individually, over time.*

*Figure 19: An example illustration of a 1-dimensional convolution, with the immediate patterns kernel shown in green, and the longer-term patterns kernel (with a dilation factor of 2) shown. An example of the output signal that the first 3 steps of the immediate patterns kernel is shown in pink. Padding applied to the input signal (orange) for the immediate kernel is highlighted in yellow, and for the longer-term patterns kernel is highlighted in blue.*

*Figures 20 (left): A Kernel Density Estimation (KDE) heatmap of the density distribution of the x and y components of all trajectories of the level 56 trajectory data. The KDE was applied on a grid of 100x100, and local maxima of the output were extracted, shown as blue markers (x).*

*Figure 21 (right): A heatmap of the density distribution of the x and y components of all coordinates, on the trajectory x, y grid. Markers extracted from the KDE are mapped to the coordinates of the heatmap, shown as blue markers (x).*

*Figures 22 (left): A plot of the macro regions defined from the watershed segmentation algorithm, with each region defined as a different colour on the level 56 map depicted. Centres of each region are shown as markers (x).*

*Figure 23 (right): A plot of the micro regions defined from the watershed segmentation algorithm within each macro region, with each macro regions defined as distinct colours, and micro regions defined as different gradient colours within each of these macro regions.*

*Figure 24: A plot of the centres of each of the macro regions (red markers – o) and micro regions (black markers – o), with the micro region centres acting as the basis for the graph structure.*

*Figure 25: An example visualisation of the graph structure formed from the micro regions. The graph is comprised of 141 nodes (blue). Example edges were constructed to better represent the graph as a visualisation, with the density of travel between nodes as edge weights.*

*Figure 26: An illustration of the k-fold cross validation technique, with k = 5. From: (SciKit-Learn, 2009).*

*Figure 27: A multiplot of a cross entropy loss plot over number of epochs and accuracy plot over epochs for the Convolutional Neural Network (left), and Temp2Space Neural Network (right) respectively.*

*Figure 28: A confusion matrix for the Convolutional Neural Network predictions on the test set, with the 10 predicted demographic labels shown on the horizontal axis, and the actual (true) labels on the vertical axis. Each cell denotes the count into each category.*

*Figure 29: A confusion matrix for the Temp2Space Neural Network predictions on the test set, with the 10 predicted demographic labels shown on the horizontal axis, and the actual (true) labels on the vertical axis. Each cell denotes the count into each category.*

*Figure 30: A plot of a bipartite graph, with micro nodes shown in blue and macro nodes shown in red. The graph is constructed with directed edges, representing the passing of information from micro to macro node in a single direction. As no node of the same graph layer (macro/micro) has an edge to another of its node, the graph is considered bipartite.*

# Table Legend

*Table 1: A table of 6 common deep learning techniques used in trajectory analysis, including a description of each, as well as brief lists of key advantages and disadvantages.*

*Table 2: A table of the 4 hyperparameters optimised in this project, namely learning rate, momentum, h (convolution signal dimension), and layers (number of hidden neurons in linear layers). The table notes the ranges of values tested in the 100 optuna hyperparameter optimisation trials, the rationale behind the ranges, and the model that these hyperparameter optimisations were applied to.*

*Table 3: A table of the 6 performance metrics used to analyse the performance of both the Convolutional Neural Network, and Temp2Space Neural Network, along with a description of each metric, and the benefit of using this performance metric – accuracy, confusion matrices, precision, recall, f1-score, and support.*

*Table 4: A table of the classification report results produced by the Convolutional Neural Network’s predictions, including precision, recall, f1-score, and support values for each of the 10 demographic labels, along with the overall accuracy, the average of each metric (macro average), and weighted average.*

*Table 5: A table of the classification report results produced by the Temp2SpaceNN’s predictions, including precision, recall, f1-score, and support values for each of the 10 demographic labels, along with the overall accuracy, the average of each metric (macro average), and weighted average.*

# 1 Introduction

## 1.1 Navigational Abilities

Navigational abilities refer to a human’s ability to identify their surroundings, orient themselves, devise a plan, and navigate themselves through an environment in order to reach a desired destination (Wolbers & Hegarty, 2010). These decisions are linked to various human cognitive functions (notably linked to the hippocampus (Weisberg & Ekstrom, 2021)) including spatial awareness, decision-making, and memory (Bermudez-Contreras et al., 2020), functions that are essential for thriving in day-to-day life. It remains a relatively under-explored area within the broader field of behavioural research, owing to the inherent complexity of studying navigational abilities and their dynamics (Spiers et al., 2021).

Taking the London Underground as an example (shown in figure 1), we can see how navigational abilities manifest in everyday life. Two individuals starting at Baker Street, aiming to reach Tower Hill, may travel vastly different routes to reach their destination, influenced by personal and environmental factors.

A map of a subway system

Description automatically generated

*Figure 1: A section of the map of the London Underground.*

Accurately measuring navigational abilities using trajectories recorded from real-world environments presents significant challenges, primarily owing to the difficulty of controlling for external variables such as weather, GPS signal variability, etc (Brügger et al., 2019). The use of virtual environments eliminates many of the challenges imposed by the various factors faced in studying real-world trajectory data whilst still effectively capturing the nuanced nature of spatial navigation and strategies (Coutrot et al., 2019) (Spiers et al., 2021) (Coutrot et al., 2018). This creates the potential for a more comprehensive understanding of the factors that may influence spatial abilities and the cognitive functions such as spatial awareness, decision-making, and memory.

## 1.2 Sea Hero Quest

Sea Hero Quest (SHQ) was created with the aim of discovering insights into how people navigate defined spaces and make decisions (alzheimersresearchuk, n.d.). The game’s founders aimed for the data gathered to be used in research into conditions such as dementia and the game was created by Glitchers with founding partners Alzheimer UK, University College London and University of East Anglia. The game was comprised of various tasks, segmented into different levels, the most frequent of which was a wayfinding challenge. This task involves controlling a boat, using the gyroscope in mobile phones, to navigate a virtual ocean environment to a series of waypoints, with the map and placement of the various waypoints unique to each level (shown in figure 2).



*Figure 2: A visual representation of the visual style of Sea Hero Quest, showing the user controlling a boat in navigation tasks. From: (alzheimersresearchuk, n.d.).*

At the start of each wayfinding level, the user was shown a map of the level including the checkpoints to memorise. Level design was generally constructed so more difficult levels contained waypoint paths that were non-linear in nature, with the user having to re-navigate after reaching a checkpoint. An example of the level 43 map is shown in figure 3.

A map of a game

Description automatically generated

*Figure 3: An image of a Sea Hero Quest map (specifically level 43’s map) that are shown to users at the beginning of each level. From: (Coutrot et al., 2019).*

The player could control factors such as boat direction and speed. Participation in Sea Hero Quest gameplay ultimately resulted in over 3.9 million participants’ trajectory data, along with demographic parameters such as age, gender, and location (Coutrot et al., 2022). The data provides an opportune medium for the analysis of trajectory data to uncover patterns between navigational abilities and demographics.

Moreover, deriving patterns that are informative of the player’s executive functions (e.g. working memory, planning, adaptable thinking, self-control, self-monitoring, time management, and organisation (Creative, 2022)), as well as the player’s ability to form cognitive maps and spatial memory ability (Ishikawa, 2022), offers potential new insights into understanding how these spatial abilities may correlate with broader cognitive health issues, such as Alzheimer's disease, something the developers of Sea Hero Quest intentionally set out to research (Spiers et al., 2021).

## 1.3 Alzheimer’s Disease: Cognition and Health

Alzheimer’s disease is a progressive, neurological condition affecting over 55 million people worldwide (World Health Organization, 2023). It is one form of dementia being the greatest cause of cognitive decline in humans (Kumar et al., 2022). Notably, spatial and navigational abilities are negatively impacted (Coughlan et al., 2019) (Levine et al., 2020) (Laczó et al., 2022) (Puthusseryppady et al., 2022). As dementia-related diseases currently have no cure, early diagnosis and identification of hallmark symptoms provide key advantages including finding ways of living with memory loss and confusion, and earlier access to clinical trials and support (Rasmussen & Langerman, 2019). Multiple studies have discerned that the decline in visuospatial abilities occurs early in the progression of the disease, before other declines in cognition become evident, illustrating that spatial ability assessments can be a potentially useful early detection tool (Quental et al., 2009) (Mandal et al., 2012) (Salimi et al., 2017).

In relation to demographic disparities, Alzheimer's disease does not affect populations uniformly. Age is the most universally acknowledged and significant risk factor for the development of the disease (Xia et al., 2018), with its prevalence increasing with age – most notably in those aged 65+ (NHS, 2018). Furthermore, gender differences also play a significant role in the risk of developing the disease, with women disproportionally (twice as likely (Alzheimer's society, 2023)) affected compared to men (Musicco, 2009).

## 1.4 The Predictive Potential of Navigational Behaviour: Age and Gender

A number of differences in navigation between males and females have been documented. Men tend to display better navigation abilities in tasks involving spatial orientation and maze navigation (Cazzato et al., 2010) (often taking more shortcuts (Boone et al., 2018)) and generally pausing less and making fewer mistakes in tasks overall (Cánovas et al., 2008). Conversely, women tend to display better navigation abilities in episodic memory tasks (Cazzato et al., 2010), and also tend to approach the learning phases of navigation tasks with more caution than men, displaying less direct route paths towards the goal (Schinazi et al., 2023). Gender related differences in cognition are not fully understood.

The potential of being able to analyse and then accurately predict age and gender demographics through navigational patterns, offers significant contributions to study, owing to its potential in advancing early Alzheimer's disease detection. Once normal ranges (within demographics) of navigational abilities have been established, further study can be conducted to identify abnormal navigational abilities (for example a middle-aged female exhibiting navigational characteristics of an older female) that could indicate abnormal decline.

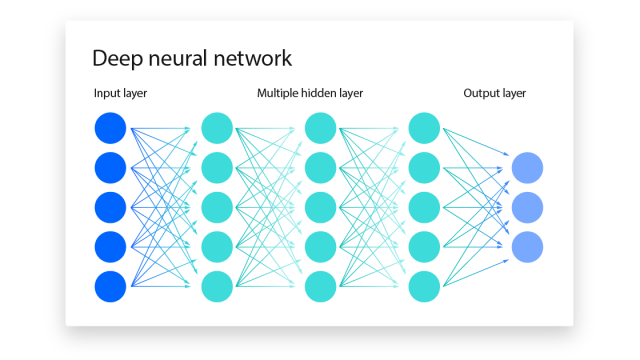
It is important to note that the investigation carried out only seeks to explore whether there is a correlation between navigation behaviours and age/gender, rather than establishing a causational relationship between them.

## 1.5 Methods of Trajectory Analysis

At its foundation, trajectories, such as those generated in SHQ, is built of spatial and temporal information. They represent sequences of coordinates (x, y) that hold the information as to where on the virtual environment a player has been, with these coordinates stored at specific time intervals (t), providing a timeline of movements that reflects the decision-making processes and navigational skills of the players. In addition, trajectory data often include more than just coordinates and time intervals, containing an array of feature data (e.g. speed, duration, direction, and areas visited as examples) that correspond to either the temporal or spatial dimensions, providing an additional source of analysis. There are different methods of carrying out the classification of trajectories, ranging from simpler traditional approaches to more advanced machine learning (ML) methods.

There are a number of different advanced methods that have been applied to the field of trajectory analysis, with machine learning, specifically deep learning, at the heart of this research. To understand the application of these models, it is first important to define what deep learning is and how it works. Deep learning is a subset of machine learning (systems that are able to learn without following explicit instructions (IBM, 2023)) that encapsulates the use of neural networks (strictly speaking those with 3 or more layers (IBM, 2023)) to identify, learn, and interpret patterns from the trajectory data and its features.

Neural networks are models that are built up of nodes (organised into layers – notably an input, hidden, and an output) and edges (the paths information takes between these nodes) shown in figure 4, mimicking the method of learning found in the neurons of the human brain (IBM, 2023). Nodes essentially act as individual neurons of the brain, designed to receive a signal (fed to the initial input layer directly from the dataset or relayed by other nodes), apply mathematics operations to said signal, and then pass this signal on to nodes in subsequent layers of the model, with edges dictating this flow of information (Islam et al., 2019).



*Figure 4: A visualisation of the structure of a neural network in deep learning, showing nodes arranged in 5 layers (1 input, 3 hidden, 1 output), with edges connecting the nodes of one layer to the next. From: (IBM, 2023).*

Notably, threshold values assigned to each node limit their activation (their ability to pass on information – decided by an activation function), with weights modulating the strength of the signals received (Islam et al., 2019). During learning, the outputs produced by the neural network are compared with expected results, with the error used in adjusting the weights, optimising the predictions, with this process repeated across multiple iterations (referred to as epochs), each time incrementally improving the model’s performance (Maennel et al., 2020). Overall, this process results in a neural network capable of making highly accurate predictions/classifications based on learned patterns of the data.

Deep learning models aim to extract new features from the trajectory data that are not inherently apparent in the trajectory dataset, providing advantages such as being able to learn highly complex patterns and relationships from an array of both temporal and spatial features that would otherwise have gone unnoticed. However, these advanced methods also contain disadvantages, such as the ease of development and challenges with the interpretability/explainability of the model, where understanding the decisions made by the model can often be difficult (Blouin, 2023).

There are a number of different deep learning approaches can be used in trajectory analysis(Graser et al., 2023)**.** A brief description, along with advantages and disadvantages of the most common approaches has been provided in table 1.

|  |  |  |  |
| --- | --- | --- | --- |
| Technique | Description | Advantages | Disadvantages |
| Transformers | Utilises a multi-head attention mechanism to process sequences of data | * Effective at modelling complex, long range temporal dependencies | * Very computationally expensive * Requires large volumes of input data for training |
| Recurrent Neural Networks (RNNs) | Utilises a memory mechanism that captures information from previous inputs to process sequential data | * Effective at modelling temporal dependencies in trajectory data * Suitable for sequence-to-sequence learning * Can handle variable-length inputs | * Suffers from vanishing gradients * Limited ability to capture longer range dependencies |
| Long Short-Term Memory Networks (LSTMs) | An advanced extension of RNNs that incorporate mechanisms to avoid the vanishing gradient shortcoming of traditional RNNs | * Effective at capturing longer range dependencies * Addresses vanishing gradient issue of RNNs | * More computationally expensive than regular RNNs |
| Gated Recurrent Unit Neural Networks (GRUs) | A simplified LSTM using fewer gates | * Faster * Less computationally expensive compared to LSTMs | * Can underperform in capturing highly complex temporal dependencies |
| Convolutional Neural Networks (CNNs) | Utilises filter (kernel) optimisation to apply convolutional operations to a sequence or spatial pattern | * Effective at modelling spatial dependencies in trajectory data * Can be applied to modelling temporal dependencies, with correct pre-processing | * Primarily optimised for grid-based data structures such as images * May not capture temporal dependencies as effectively as other models * May not be suitable for variable-input lengths |
| Graph Neural Networks (GNNs) | Utilises message passing in graph structured data to learn patterns and relationships in the data | * Highly effective at modelling spatial dependencies in trajectory data * Suitable for large-scale applications | * Requires data to be in a graph based format * Can become highly computationally expensive with the size of the graph |

*Table 1: A table of 6 common deep learning techniques used in trajectory analysis, including a description of each, as well as brief lists of key advantages and disadvantages.*

This highlights the difference of applicability of each model, depending on the task at hand and the form of the trajectory data. An interesting approach to demographic prediction is in modelling the temporal and spatial dependencies separately, allowing for the potential to compare the approaches and analyse if temporal or spatial signatures of trajectory data are more influential in predicting the combined age and gender demographics. Therefore, a Convolutional Neural Network approach to demographic prediction offers the most appropriate approach, owing to its ability to model both temporal and spatial dependencies (unlike the other techniques previously mentioned). Specifically, this study is interested in both a Convolutional Neural Network (CNN) approach to model temporal dependencies, as well as a Temporal 2 Space Convolutional Neural Network (Temp2SpaceNN) approach – based off a conventional CNN – to model spatial dependencies in a single model, offering a way to allow for comparisons of the models to be made in their predictive capabilities.

Convolutional Neural Networks demonstrate great potential for learning patterns from time-series data through 1-dimensional convolutions (Escottá et al., 2022). An illustration is shown in figure 5.

A diagram of a diagram of a diagram

Description automatically generated

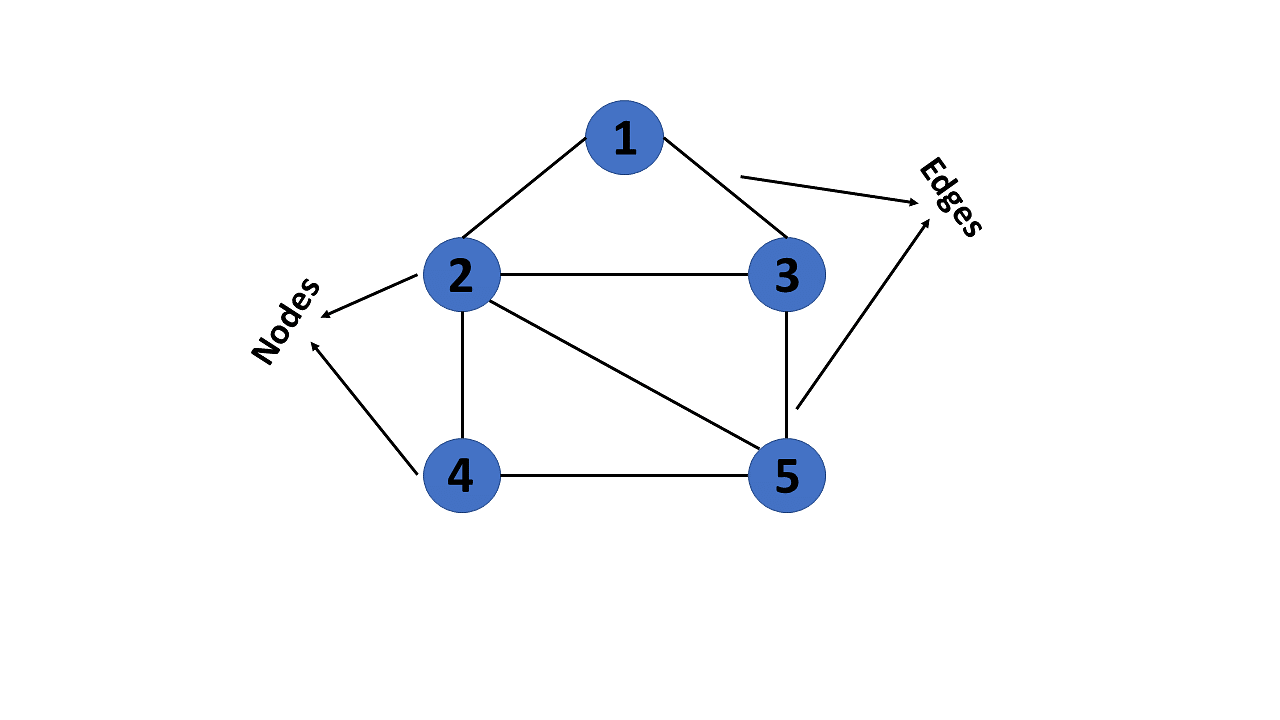
*Figure 5: An visual illustration of the convolutional layers of a Convolutional Neural Network, acted upon a raw signal. From: (Shenfield & Howarth, 2020).*

These convolutions involve sliding a kernel over the input signal to learn time-series progression dependencies and nuanced features, effectively combining the input signal and kernel in a process that results in the creation of a new derived signal (Smith, 1997). Key factors that influence this process include:

* The kernel size – the breadth of the convolution operation
* The dilation factor – the spacing of the kernel's elements, allowing for a wider scope of analysis
* The use of padding – applied before and after the input signal ensuring the kernel can be applied uniformly across the entire dataset

These parameters influence the CNN’s ability to interpret the time-series data to different extents. 1D CNNs are highly advantageous in learning both temporal patterns, as well as spatial progression patterns, however are inherently unable to model spatial patterns, as these patterns cannot be built into the 1-dimensional time-series representation of the signal, that 1D CNNs use.

In direct contrast to learning patterns solely from temporal aspects of trajectory data, the newly proposed Temp2SpaceNN (by (Dubois, 2023)) shows promise as a new, fascinating area of study, owing to its ability to model special dependencies from temporal convolutions, using graph pooling. A graph is a non-linear data structure, built up of nodes (also termed vertices *V*) and edges (*E*), mathematically defined as *G = (V, E)*. An illustrative example of a graph is shown in figure 6. A graph’s nodes and edges are analogous to the elements of neural networks previously discussed, with nodes acting as stores of information/signals, with each having a weight attribute that influences the signal’s strength in the graph’s network, and edges acting as the flow of information between these nodes.



*Figure 6: An example illustration of a simple graph, including labels of both nodes and edges. The graph is composed of 5 nodes, with 7 edges connecting these nodes in the graph structure. From: (Ravikiran, 2023).*

In the context of trajectory analysis, there are multiple different approaches to mapping the data to a graph structure.

The first approach maps the individual trajectories to graphs through spatial grid projection, enabling for clustering methods to capture grouped trajectories that share similar properties – with its advantage in its scalability and grouping properties (Sabarish et al., 2020). Conversely, an alternative approach builds a graph directly from the map topology itself, with the graph being a representation of the level design, incorporating its geography and landmarks into the data structure, potentially offering better insights into spatial relationships and behaviours and allowing for the better detection of patterns in the data that are related to map specific nuances (Kim et al., 2017).

Temp2SpaceNNs are similar to CNNs in their extraction of features through convolutional operations, however differ in the fact that these operations are then transformed through a pooling operation that maps the learned features directly onto the graph structure (Wu et al., 2019), which for this study represents the topology of the spatial environment. From this, the Temp2SpaceNN is able to learn complex spatial patterns from the trajectory signals, allowing for the understanding of spatial dynamics that CNNs lack.

Finally, with the use of these models, it is important to note that there are multiple challenges in applying these to the Sea Hero Quest data to successful extract patterns of behaviour needing to be accounted for:

1. Heterogeneity in the distributions of demographics may pose an issue as it may compromise the generalisability of the model to unseen data, with the model not performing equally well across demographic categories.
2. There is the potential for incorrect input of demographic data by the user, which could compromise the integrity of the results.
3. The Sea Hero Quest dataset is substantial in volume; training CNN and Temp2SpaceNN models on the entire dataset may not be feasible due to the limited resources and time available.
4. Convolutional Neural Networks require data to have the same sample size (length). Therefore, methods of resampling must be explored to allow for a CNN based approach to work with the Sea Hero Quest dataset, due to each trajectory being of a different length.
5. Sea Hero Quest levels are relatively simple in design, with all users carrying out the same task, on the same relatively simple map. This may limit the variability in behaviours observed, and hence may limit the amount of patterns that can be extracted from the data.

## 1.6 Aims and Objectives

Whilst current literature has highlighted distinct navigational behavioural differences across age groups and genders, there is a lack of understanding of how these differences converge and interact. With previous research having only examined these demographics in isolation to one another, this project aims to group them together, aiming to look into if any combined age and gender based patterns can be extracted.

This project aims to explore the differences in navigational abilities and behaviours across age and gender demographics, determining whether, and to what extent, we are able to accurately predict age and gender demographics based on navigational behaviours extracted from trajectory data.

To achieve this, this project aims to employ 2 models (a CNN and a Temp2SpaceNN ), with this dual-model approach allowing for the comparison of the differences in the contribution of temporal and spatial dependencies in predicting the joint age and gender demographics, contributing valuable insights into which factors of the differences in navigation behaviours may be responsible.

Additionally, in this dissertation, a number of specific research objectives will be explored, listed below:

1. Establish a methodological framework to feed the trajectory data into CNNs and Temp2SpaceNNs.
2. To utilise a convolutional neural network to look into whether any patterns of behaviour can be found from the temporal aspects of the SHQ data.
3. To utilise a Temp2SpaceNN to analyse whether any patterns of behaviour can be found from the temporal & spatial aspects of the SHQ data.
4. Use each model to predict a combination of age and gender demographics, extracting results that can be used to evaluate the effectiveness of each model in their ability to predict these factors or not.
5. Compare the results from each model to assess the relative importance of temporal and spatial dependencies on extracting patterns of behaviour in each demographic.

Summarising, for this project, multiple research questions can be devised, relevant to the aims and objectives of this project:

1. What features can be extracted from the Sea Hero Quest dataset, that are significantly able to model differences in navigational behaviours?
2. How accurately can the CNN and Temp2SpaceNN models predict grouped age and gender demographics based on their navigational behaviours, captured in trajectory data in Sea Hero Quest?

## 1.7 Key Definitions

A table of the key terms used throughout this project are defined in appendix A, for ease of reference.

# 2 Literature Review

## 2.1 Existing SHQ Research and Cognition

There have been multiple pivotal studies using the SHQ dataset, assessing and analysing spatial navigational abilities as a measure of cognitive function, with the studies using demographic data to discover key patterns across the demographics classes, including age, sex, and location – SHQ studies explored these demographic classes in isolation.

First, multiple studies showed that spatial navigational abilities generally declined with age (Coutrot et al., 2018), however some attributed this to selection bias (as participants aged 70+ that were able to participate in playing SHQ were more likely to express greater cognition skills (Spiers et al., 2021).

Multiple studies also noted key patterns in spatial navigational abilities and gender. Works of (Coutrot et al., 2018) and (Coutrot et al., 2019) found higher average performance metrics in males between the ages of 19 to 60 than females of the same age range. Despite this, (Coutrot et al., 2019) standardised gender of real-world pathing tasks against video game familiarity and found that gender differences were not significant, suggesting that the differences were linked to how familiar each sex was navigation tasks, rather than inherent biological differences. Similarly, (Coutrot et al., 2018) highlighted other potential explanations for variance in spatial abilities with sex, finding that both GGI (Gender Gap Index – a measure of inequality in genders) and GDP (Gross Domestic Product – a measure of the total market value of a country) significantly predicted gender spatial abilities, with differences in spatial abilities becoming more pronounced in countries with greater gender inequalities.

## 2.2 Deep Learning models and Trajectory Analysis

In terms of existing literature of trajectory analysis, a diverse number of deep learning techniques have been explored ranging from Transformers (Hong et al., 2022), RNNs (Kashyap et al., 2021) (Li et al., 2018), LSTMs (Liatsikou et al., 2021) (Mehri et al., 2021), GRUs (Fan et al., 2022), CNNs (Zamboni et al., 2022), GNNs (Yang et al., 2023) (Altan et al., 2022), to graph attention networks (Zhang et al., 2020), and multi-layer perceptrons (Simini et al., 2021). These techniques have been employed across a broad spectrum of research tasks involving trajectory data in the literature, including arrival time prediction, traffic prediction, destination prediction, location classification, trajectory prediction, mode of transport prediction, and demographic prediction (Graser et al., 2023).

Current research highlights multiple methods of representing trajectories to allow for them to be fed, these include methods as simple as using the raw trajectories as signals (Bas Jacob Buijse et al., 2021), using resampled trajectory sequences (Liatsikou et al., 2021), using discretised trajectories (trajectories that map to a grid structure) (Carroll et al., 2022), using rasterised trajectories (trajectories signals in the form of images) (Yang et al., 2022), and graphs (Kim et al., 2017). The feeding strategy is dependent on the type of neural network model used; for example, resampling or rasterization is used for models such as CNNs, due to their limitation in needing same length signal inputs. It should be noted that not all methods process full length trajectories, with papers such as (Chen et al., 2020), (Wang et al., 2021) and (Dubois, 2023) utilising subtrajectories (smaller trajectory sequences of an overall trajectory) to carry out classification tasks, enabling for a more detailed level of analysis.

Multiple studies such as (Wang et al., 2018) and (Derrow-Pinion et al., 2021) have emphasised improved model performance on typically temporally dominated tasks (such as arrival time prediction) through the incorporation of spatial dependencies (e.g. geographical information in a geo-convolutional operation in an RNN model), highlighting that even with temporally driven tasks, consideration of the spatial dynamics may further improve the accuracy of models in their predictive/classification tasks.

Finally, relevant to our scope of research in predicting user demographics from navigational behaviours, works of Hippolyte Dubois, Patrick Le Callet, and Antoine Coutrot ((Dubois et al., 2021) and (Dubois, 2023)) are of considerable interest, owing to their novel approach to understanding the importance of demographic characteristics, using a multi-label task to accomplish this. Their proposed method, a composite signal analyser (CompSNN) is an amalgamation of different techniques that combine to form one advanced model, with the advantage of this approach being in its ability to obtain better analysis performance over the individual sub-models, whilst retaining a strong level of explainability. This is important in understanding and plotting spatial dynamics, that is lost in more complex models. In reference to (Dubois, 2023), the project’s CompSNN was comprised of a 1-dimensional convolutional neural network to model the temporal dependencies, a Temp2Space neural network (a model that pools temporal information onto a graph structure) to model temporal and spatial dependencies, and a hierarchical model, that passes signals from a micro to macro scale in an attempt to learn complex spatiotemporal patterns in navigational behaviour. Additionally, the study found that age and gender were the two most prominent demographics that could be predicted from patterns learned from whole demographic profiles. Owing to the similarity of research, employing an approach akin to that explored in (Dubois, 2023) could be an interesting direction for our study, as this could allow for relative comparisons to be made between the performance of their models that analyse age and gender independently and our models that consider age and gender as a combined demographic variable.

## 2.3 Summary

There has been numerous studies conducted into the data produced by Sea Hero Quest, and other similar games, with multiple demographical patterns emerging. There has been an increasing use of deep learning techniques, with multiple new techniques and approaches to research having the potential to improve model performance. There has been new research proposed that models the importance of demographic labels in the task of predicting a user’s demographic profile based off of their navigation behaviours extracted from trajectory data.

A number of gaps in the literature can be observed, such as:

1. There is still a lack of evidence as to the mechanisms behind navigational differences, with studies unsure if navigational differences observed in demographic groups such as gender are as a result of biological differences, or as a result of their familiarity with spatial tasks through environmental factors and upbringings. Therefore, further research in respects to demographic groups could help clarify the contributions of factors influencing navigational behaviours, reducing potential bias in our assumptions about inherent navigational abilities.
2. Of the few studies that explored a similar topic to that of this research paper, a multi-label classification method was used, that lacks the isolation of demographic behaviours like that of a multi-class approach. Therefore, more multi-class classification research in this area could aid in understanding navigational behaviours between specific groups, to gain a clearer understanding of the interplay between the groups.
3. Other more complex factors such as socioeconomic status, cultural background, and educational differences have yet to be comprehensively explored. Whilst these are not the focus of this study, it is worthy to note as potential avenues for future research.

In summary, further research is needed to understand more of the underlying patterns and potential contributing factors to these differences.

# 3 Methodology

## 3.1 Data Search Strategy and Collection

Whilst this project was initially proposed with the SHQ dataset, a systematic search of readily available datasets was also conducted, for more readily available datasets that were better fit to the aims of this project.

The search strategy involved:

1. Searching the web using terms such as ‘trajectory datasets’, ‘traffic datasets’, ‘GPS datasets’, ‘taxi datasets’, ‘taxi trajectories’, ‘environment navigation’, ‘virtual environment navigation’, etc.
2. Exploring current literature relevant to our project, and the datasets that they used.
3. Searching known data repositories, such as Kaggle, GitHub, the UC Irvine Machine Learning Repository, and Microsoft Research Open Data for trajectory datasets with open licences to use

Through this, a number of datasets (e.g. (Ma et al., 2019) (Cross, 2018), (Zheng, 2011)) were discovered, with each evaluated on their relevance of use in our project. This evaluation provided us with the justification that the SHQ dataset was the most appropriate dataset for the study, as it provided the advantages such as the data being tightly controlled within a virtual environments (which other datasets such as the real-world taxi datasets lacked), or the advantage (pivotal to our study) of including demographic information for the trajectories. The vast quantity of data stored were also influential in the decision to use the SHQ dataset. A licence to use the SHQ dataset was requested from <https://seaheroquest.alzheimersresearchuk.org/> and approved in June 2023, granting access to the full 2019-12-16 SHQ data.

## 3.2 Build

This project was carried out using python 3.11, with its wealth of libraries/packages aiding in completing the tasks, in particularly to machine learning applications. A number of packages were used throughout this project, which are included in appendix B.

## 3.3 Structure of Sea Hero Quest Data

The SHQ data was comprised of a trajectory dataset (a 120GB .json.gz zipped file), and a user dataset (a 2GB .json.gz zipped file). The SHQ data can be divided into 3 main categories:

1. The player trajectory data and associated features.
   1. Trajectories were stored as x, y, and r (rotation) lists, with each sample taken every 0.5 seconds
   2. The duration spent to complete the level
2. The associated meta-data.
   1. Unique identifiers of each trajectory including UUID and User ID
   2. The SHQ level for the associated trajectory
   3. Number of previous attempts
   4. Whether the level was terminated early
   5. The amount of time spent viewing the map at the beginning of the level
   6. The platform (iOS/android) and the SHQ app version
3. The demographic data of users.
   1. Age
   2. Gender
   3. Recent activity
   4. Level of education
   5. Dominant hand
   6. Home environment
   7. Location (country)
   8. Self-rated navigation skills
   9. Average hours of sleep per night
   10. Daily time spent travelling

For the aims of this study, the player trajectory data, the associated meta-data, and the age and gender demographic variables were utilised.

The project focuses on level 56, shown in figure 7, comprised of 5 waypoints scattered in different areas the map.

A map of a maze

Description automatically generated

*Figure 7: An image of the level 56 Sea Hero Quest map. From: (Coutrot et al., 2019).*

This level was chosen for two main reasons. First, level 56 was chosen due to its relative complexity in navigation task in comparison to earlier levels of the game. Second, level 56 highlights an interesting spatial dynamic. On close inspection of the route one would take, checkpoints 1, 2, and 3 follow a linear path. Whilst this is the case, once the user has reached checkpoint 2, a barrier appears that prevents the user from reaching checkpoint 3 using the same linear pattern that they had likely thought about taking when looking at the map at the start of the level. This results in the level being a great indication of cognitive ability in the fact that the user has to remember the map and adapt and re-think their initial approach to reaching the 3rd checkpoint, with those with worse memories and abilities showing more uncertainty on the map. An example video of the level is provided: <https://www.youtube.com/watch?v=D7QoR2lc6vQ>.

It should be noted that the code for this project was constructed to allow for level 56 to be swapped with other levels of SHQ, ensuring that methodologies and findings from this study can be applied to other levels.

## 3.4 Data Pre-processing and Cleaning

### 3.41 Data streaming

Level 56 data was first extracted from the overall SHQ dataset for ease of analysis. Owing to hardware limitations, the entire SHQ dataset was unable to be stored in system memory due to its 120GB size. Therefore data streaming was utilised using the gzip, json, and csv libraries, feeding each line of the overall compressed json file through a check that looked at which level the trajectory was for, and wrote any level 56 trajectories to a new csv file. This resulted in a significantly smaller 5.5GB csv file of trajectories just assigned to level 56.

An example of 2 trajectories, showing different spatiotemporal patterns, are shown in figure 8. From visual inspection, it is evident that example trajectory B shows much weaker levels of certainty in their navigation path versus that of example trajectory A, visiting more unnecessary areas and showing far more backtracking across multiple different areas of the map.

A graph of a map of a person

Description automatically generated with medium confidence

*Figure 8: A dual line plot comparison of 2 example trajectories (A and B) of the trajectory dataset studied in the project, plotting x vs y.*

### 3.42 Demographic analysis

A number of pre-processing steps were taken to ensure the quality and integrity of the trajectory data, so that subsequent analysis would be more accurate. This included removing data which potentially indicated errors such as false input of demographic information by the user (an obvious example being someone stating they sleep for 25 hours a day).

First, any trajectories lacking age and gender demographic information were removed from the study. Trajectories with durations over 3 standard deviations (s.d.) from the mean were also removed, due to the high probability that these represented either data recording errors or severely atypical navigation (such as someone not interacting with the game, or exploring the map without engaging in the navigational task). The choice of 3 s.d. was that 99.7% data fell within this boundary, removing only extremely severe outliers that would potentially significantly affect subsequent analysis. This same approach was utilised in removing trajectories with associated sleep duration demographics 3s.d. away from the mean, as this indicated that input of demographic information was false, and therefore other demographic information was also likely to be wrongly inputted.

An examination of the distribution of the age demographic variable was conducted, shown in figure 9.

A graph of a number of people

Description automatically generated with medium confidence

*Figure 9: A histogram of the frequency distribution of ages ranging from 16-99.*

Upon observation, a large abnormal spike at both ages 18 and 99 is found (as well as an abnormal trough at age 17), deviating from the general distribution pattern observed. A likely explanation could be that, because of the required minimum 16 year age limit to the game, those under the age of 16 were false reporting their age to be able to play, using values the 18 and 99 ages most often. This explanation could also be applied to the trough seen at age 17, with those aged 17 self-recording their age as 18 instead.

Therefore, trajectories with associated 16-18 and 70-99 age ranges were removed from the study. A cut off of 70 was decided due to the lack of trajectory data in older populations, that would significantly impact the classification group distributions.

Additionally, inspection of the distributions of both genders was conducted, showing equal distributions of male and females. Groups other than male and female were removed from analysis for classification simplicity and owing to the lack of representation in other groups.

Finally, inspection of the distribution of previous attempts for level 56 is shown in figure 10.

A white line with black numbers

Description automatically generated with medium confidence

*Figure 10: A histogram of the frequency distribution of the amount of previous attempts, with a value of 1 indicating the user had no prior experience with the level.*

For simplicity of analysis, we decided to remove any trajectories that were not a user’s first attempt, as subsequent attempts were likely to introduce learning patterns that could skew the analysis of extracting navigational patterns. Finally, no trajectories that did not reach the 5th checkpoint were found in the dataset, and so no data related to this was removed – likely removed before access was given for file size implications.

Overall, the dataset was reduced from 80,455 trajectories to 26,608.

### 3.43 Discretisation of demographic variables

Another step in the pre-processing of the SHQ data was in the discretisation of the demographic variable age. Age was stored in an ordinal categorical form of integer values from 19-69, but the large amount of categories makes it unfeasible to be effectively used in the classification task. Therefore, age was discretised (binned) into 5 different groups.

A gaussian probability approach was considered in this task due to its advantage of not introducing strict age boundaries in the dataset e.g. by splitting the range of ages into 5 strict groups (e.g. 19-28, 29-38, etc.). As it is currently unknown at what age cognitive abilities start to decline, the gaussian probability approach allows for a more nuanced representation of age as a spectrum, accommodating for gradual transitions between each of the 5 groups, better reflecting the natural progression of aging without arbitrarily introducing cutoffs that may oversimplify the relationship between age and cognition. This method allows for broader trends across age groups to emerge, without overcomplicating the analysis from looking into the individual ages.

To carry out this approach, 5 gaussian distributions were applied equally spread across the demographic age values, with a peak and trough of 1 and 0 respectively, shown in figure 11. Each trajectory was assigned a vector of 5 values, one for each of the 5 gaussians. For example, a trajectory with an associated demographic age value of 19 would have a (estimated) vector [1, 0.08, 0, 0, 0].

A graph of different colored lines

Description automatically generated

*Figure 11: A line plot of 5 equally spread gaussian distributions each with a Peak Probability Density (PPD) of 1, that represent each of the 5 age groups. The means of the distributions are 19.0 (blue), 31.5 (orange), 44.0 (green), 56.5 (red), and 69.0 (purple) respectively.*

From this, a probability assignment function could be constructed that assigned each trajectory to one of the 5 age groups using the vector (normalised to 1) based on the probability of being in each category, with the distribution of these age bins shown in figure 12.

A graph of different colored rectangular shapes

Description automatically generated with medium confidence

*Figure 12: A histogram of the frequency distribution of 5 age bins (age\_0, age\_1, age\_2, age\_3, and age\_4).*

The distribution has the same general trend as the distribution of individual ages, with the exception of lower representation in the first and last age bins due to cutoff of gaussians by the data imposed minimum and maximum age limit. 10 categorical labels were constructed based on the two genders analysed (male and female), and five age groups (0,1,2,3,4), namely:

* Female-age\_group\_0 (F-age\_0)
* Female-age\_group\_1 (F-age\_1)
* Female-age\_group\_2 (F-age\_2)
* Female-age\_group\_3 (F-age\_3)
* Female-age\_group\_4 (F-age\_4)
* Male-age\_group\_0 (M-age\_0)
* Male-age\_group\_1 (M-age\_1)
* Male-age\_group\_2 (M-age\_2)
* Male-age\_group\_3 (M-age\_3)
* Male-age\_group\_4 (M-age\_4)

The distribution of these groups are shown in figure 13.

A graph of different colored bars

Description automatically generated

*Figure 13: A histogram of the frequency distribution of the 10 combined (2) gender and (5) age bins, for a total of 10 demographic sets: f-age\_0, f-age\_1, f-age\_2, f-age\_3, f-age\_4, m-age\_0, m-age\_1, m-age\_2, m-age\_3, and m-age\_4.*

The distribution shows that distributions are relevant evenly between genders, with the only notable exception being in the first age groups, with higher frequencies found in young females (f-age\_0) over males (m\_age-0). It’s apparent that age demographics are not evenly distributed, with the general trend being that frequency declined with subsequent age groups within the genders. This observation will necessitate consideration of the model training process, so that bias is not introduced in favour of higher frequency demographic groups.

### 3.44 Reconstructing directional information

Furthermore, owing to a lack of understanding of how directional information was stored in variable ‘r’, directional information was directly calculated from the (x, y) information of each trajectory. Direction was stored as cos(θ) and sin(θ) values, calculated using the deltas of x and y respectively and the hypotenuse of both (trigonometry).

An additional consideration of reconstructing directional information was made. As the trajectories recorded in Sea Hero Quest were stored as coordinates discretised to a grid, the resulting directional information resulted in a ‘snappy’ appearance. If we consider that natural changes in direction tend to happen smoothly (e.g. a person gradually changing their path direction to avoid an obstacle/turn a corner), this should be reflected in the trajectory data. Therefore, a gaussian smoothing operation was carried out, smoothing the change in direction using a sigma of 2. The original and reconstructed directional features are shown in figure 14 and 15 for cos(θ) and sin(θ) respectively.

A graph with green and orange lines

Description automatically generated

*Figure 14: A plot of the original (green) and gaussian smoothed, reconstructed (orange) directional component cos(θ) of an example trajectory, over time (t).*

A graph with green and orange lines

Description automatically generated

*Figure 15: A plot of the original (green) and gaussian smoothed, reconstructed (orange) directional component sin(θ) of an example trajectory, over time (t).*

The plots show the reintroduction of smooth changes in direction over time, with this reconstructed information tending towards natural observations seen in the real-world.

### 3.45 Resampling trajectory data

Both the CNN and the Temp2SpaceNN models used in this project use convolutional layers. As previously mentioned, a prerequisite to their use is that the trajectories are required to be of the same sample size, which is not an inherent characteristic of the raw trajectory data from the SHQ dataset. Therefore, a method of resampling was required in order to transform the trajectories so that they could be correctly fed into each of the models.

There are multiple methods of resampling trajectory data including linear interpolation, spline interpolation, cubic interpolation, and Fourier transform resampling, with each method excelling in different circumstances. Cubic interpolation was chosen as the resampling method for the trajectory data of this project, owing to its ability to consider smooth curves and the naturally smooth nature of changes in direction and space as previously mentioned (by using cubic function curves and using these curves to plot new resampled coordinates based off), that methods such as linear interpolation (connecting points using straight lines) cannot model. 1-dimensional cubic interpolation was applied to both the x and the y coordinates individually using scipy.interpolate to achieve this.

In retrospect, whilst a number of resampling methods were explored/evaluated, a more suitable approach would have been by using re-discretization methods to resample the data (such as the Traja package re-discretization tool (Traja, 2021)), owing to the trajectory data being discretised to our grid. Whilst the cubic resampling method better reconstructed trajectory data based on their smooth nature, it also resulted in non-integer values that, whilst completely valid in the temporal CNN, resulted in issues when mapping trajectory points to a graph (later described in the project) for the Temp2SpaceNN spatial model.

The sample size was calculated as the 75th percentile of the distribution of the length (the distribution shown in figure 16), resulting in all trajectories resampled to a sample size of 523.

A graph of a distribution of a number of people

Description automatically generated with medium confidence

*Figure 16: A histogram of the frequency distribution of the length of all trajectories in 100 bins. An x limit of 0 – 2000 was applied for ease of viewing, owing to the lack of any discernible bars past a length of 1250.*

The 75th percentile was chosen as the majority of data fell within this, striking a balance between ensuring sufficient detail in trajectories was maintained, limiting the amount of noise introduced to shorter trajectories, and limiting the complexity of the model training process, as choosing too large a sample size would significantly increase the computational demand on the model, increasing the time taken to train (and even potentially overbearing tensor size, introducing additional more complicated methods to account for tensors not fitting in system memory).

A plot of the original and resampled shortest and longest trajectories are shown in figure 17.

A screenshot of a graph

Description automatically generated

*Figure 17: A multiplot of 2 original (blue) vs resampled (orange) example trajectories – the shortest and longest recorded from the trajectory dataset studied in the project, plotting x vs y.*

Upon observation, it is evident that whilst the amount of samples increased/decreased quite drastically, the overall, the overall shape of the trajectory was preserved. This is further backed up by figure 18, that plots x and y individually for the longest trajectory of the dataset.

A graph of different colored lines

Description automatically generated with medium confidence

*Figure 18: A multiplot of an investigation into the shortest trajectory’s original (blue) vs resampled (orange) x and y components, individually, over time.*

Whilst resampling introduces noise to shorter trajectories and loses information in longer trajectories, we found that using the 75th percentile provided a good balance, ensuring not too much noise was introduced in shorter trajectories, whilst ensuring not too much information was lost for trajectories of long lengths.

### 3.46 Data sampling

The final pre-processing step was splitting the dataset into training and testing data; an 80/20 percent split was used as it allows for a substantial amount of data in training each model, whilst retaining enough for an accurate evaluation of the model’s performance on unseen data. Furthermore, this split was carried out using a stratified method of sampling, ensuring equal representation of the 10 age-gender groups in both the training and testing sets.

## 3.5 Model Architecture

### 3.51 Model Selection

For this project, a number of various convolutional based models were considered, to identify the most suitable architecture for the multi-class classification task of predicting demographic labels. After a thorough review of current similar literature to this project, two sub-models from (Dubois, 2023) stood out as the most applicable to our research objectives. Whilst the models of this paper were used in predicting demographic profiles using a multi-label classification approach, these models were also applicable to our multi-class classification task, and therefore were adapted and utilised in this project. This decision was driven by multiple key factors:

1. The first sub-model of the combined CompSNN model of (insert source) modelled the temporal and spatial progression dependencies in isolation, with the model not considering any non-temporally based spatial patterns that could contributed to differences in navigational behaviours. The second sub-model integrates the analysis of temporal, spatial progression, and spatial dependencies to discern patterns. With this, utilising both of these models for analysis could help understand the contribution of spatial patterns in the prediction of the combined age and gender labels, where using a singular model wouldn’t allow for this analysis.
2. The models were efficient in the processing of information, with the use of tensors to feed data into each model that could either be fed as a whole, or as batches if each tensor couldn’t be fed into system memory.
3. The temporal CNN of (Dubois, 2023) considered time series patterns at two different scales, by running 2 simultaneous convolutional sub-layers in parallel that looked for more immediate short-term pattern, and for longer-term patterns (using a dilation factor), with this proving for a more comprehensive analysis of patterns of the time-series data.
4. Utilising these models would allow for the relative comparison of findings between our paper and theirs, contextualising our results in a broader investigation of demographic patterns.

The models utilised were a Temporal Neural Network, and a Temporal 2 Space Neural Network.

### 3.52 Temporal model

The Temporal Convolutional Neural Network model is designed to take time-series trajectory data with the aim of capturing both immediate and longer-term patterns in the data. This dual-focus architecture provides multiple advantages, allowing for the detection of more complex temporal behaviours, better model generalisation, improved feature extraction, and its ability to help filter out noise introduced through the resampling pre-processing steps (by concatenating short-term trends, potentially influenced by noise, with more stable long-term trends that are less affected).

In constructing the CNN model, feature selection into the model was based on its ability to discern temporal patterns, as well as spatial progression patterns (dynamic aspects of movement over time) from the time-series data. The model processed a signal comprised of 8 features including the resampled x and y trajectory data, along with the derived features: speed, acceleration, cos(θ) and sin(θ) – the directional components, and delta cos(θ) and delta sin(θ) – the curvature components. These derived features were calculated from the resampled (x, y) data and stored as lists of 523 values.

The process of resampling trajectories for temporal analysis may seem counter intuitive, however by calculating the speed feature using the formula…

With:

speed

the derivative of two consecutive x coordinates

the derivative of two consecutive y coordinates

…the relative length of the trajectory is essentially baked into the speed feature. For example, consider a scenario with a pre-resampled trajectory comprised of 200 (x, y) points to reach completion equating 100 seconds at a rate of 0.5 seconds/point. When resampled to 523 points, the relative distance between two x coordinates significantly diminishes, with this reduction captured by the speed feature as the equation uses the x and y derivatives in calculation. The method inversely impacts longer trajectories, and hence speed retains a representation of a trajectory’s overall duration.

For model feeding, these trajectories were built into a 3-dimensional PyTorch tensor of shape (trajectories, length, features), a structure conducive of neural networks and convolutional processing.

Continuing onto the specific design of the temporal model, the tensor is first passed through a MultiScaleConv1d layer that process the data simultaneously through 2 convolutional sub-layers, corresponding to the immediate patterns (a 1D convolution with a kernel of 5 and dilation factor of 1) and long-term patterns (a 1D convolution with a kernel size of 7 and dilation factor of 2). An increase in kernel size with a doubling of dilation factor was selected as it balanced essentially more than doubling the scope of the convolutional filter (allowing for broader pattern learning), highlighted in the illustration of figure 19, with maintaining a good granularity so as to not overly dilute the temporal resolution of the signal. Additionally, padding was applied to both ends of the 1-dimensional signal to ensure that the edge information was not lost. Whilst technically not as relevant to the start of the signal (as all signals tended to have a number of the same points at the start), adding padding to the end of the signal was essential to preserve the signal’s terminal information). Padding was calculated using the optimal padding formula:

Therefore, a padding of 2 was used for the immediate pattern convolution, and a padding of 6 was used for the longer-term pattern convolution.

A screenshot of a computer

Description automatically generated

*Figure 19: An example illustration of a 1-dimensional convolution, with the immediate patterns kernel shown in green, and the longer-term patterns kernel (with a dilation factor of 2) shown. An example of the output signal that the first 3 steps of the immediate patterns kernel is shown in pink. Padding applied to the input signal (orange) for the immediate kernel is highlighted in yellow, and for the longer-term patterns kernel is highlighted in blue.*

The resulting signals are then concatenated across their temporal scales and passed a MaxPool1D layer that applies a max pooling over the signal to reduce the dimensionality of the data and improve feature emphasis. The product of this max pooling operation layer are then normalised using a BatchNorm1D layer to standardise the data, helping in the training process of the CNN by preventing the inputs from changing too much in an unwanted phenomenon known as interval covariate shift, and then processed through a ReLU (rectified linear activation function) layer that introduces non-linearity to the signal to allow for the model to learn more complex representations. Finally, this signal is passed through 2 fully connected layers to transform the data into the desired output size of 10, one for each of the age-gender labels to predict.

### 3.53 Defining a graph

To feed the trajectory data into a spatial Temp2SpaceNN model, it was first essential to define a graph that represented the level topology, using the trajectories’ spatial information. To do so, a segmentation method was performed to split the level 56 graph into distinct sections, with each section acting as a node on the graph. This was constructed based on the (x, y) feature coordinates of the non-resampled trajectories, using non-resampled data to ensure integrity of the spatial information.

First, a kernel density estimation (KDE) was performed using the density heatmap of all trajectories (shown in figure 20), from which local maxima (the points of the KDE map with the highest density – calculated with a minimum distance of 5 to ensure local maxima were sufficiently spread out) were identified. This allowed for areas of high congregation (areas that users tended to visit more) to be identified from the trajectory data.

A red and blue grid with blue lines

Description automatically generatedA red and blue graph

Description automatically generated

*Figures 20 (left): A Kernel Density Estimation (KDE) heatmap of the density distribution of the x and y components of all trajectories of the level 56 trajectory data. The KDE was applied on a grid of 100x100, and local maxima of the output were extracted, shown as blue markers (x).*

*Figure 21 (right): A heatmap of the density distribution of the x and y components of all coordinates, on the trajectory x, y grid. Markers extracted from the KDE are mapped to the coordinates of the heatmap, shown as blue markers (x).*

Whilst ideally a 1-to-1 scale would be used between the density heatmap and the KDE map, a higher granularity grid (100x100) was instead used, as using the same grid dimensions as the density heatmap resulted in an over-abundance of local maxima, that would 1. significantly add to the computation complexity of the model and 2. potentially obscure important patterns due to having too many points of interest. Owing to the disparity in grid systems, the local maxima coordinates of the KDE grid were mapped back to the original trajectory grid (shown in figure 21).

Next, a watershed segmentation method was used to segment the level 56 map into distinct regions, as this method worked best in the context of SHQ level design. Other clustering methods such as K-means and DBSCAN were also considered, however these harboured disadvantages when compared to the watershed segmentation approach:

* K-means clustering operates within Euclidean space through a process of minimising within-cluster variances, which does not account for the out-of-bounds areas of the level topology. This could lead to clusters (segments) forming that extend beyond the intended border of the level, such as a region spanning 2 regions that are not directly navigable between, which would be problematic for this study.
* DBSCAN clustering would be less likely to have the same issue presented by k-means clustering owing to its density-based methodology, however DBSCAN does not account for the fact that trajectories are strongly self-correlated.

Watershed segmentation works using the concept of flooding (Bhatia, 2023), taking multiple pre-defined basins and flooding out from these until the border of a growing segment reaches another segment.

The local maxima, extracted from the KDE, were used as basins, with the watershed algorithm spreading out from each to form unique segments, using density as a weight. Notably, as the density of the Sea Hero Quest level was considered when segmenting the map, areas of higher density ‘filled’ faster than areas of lower density, resulting in larger segments in high density areas – advantageous as it better represents regions of higher density.

Furthermore, a mask of the out-of-bound areas (areas of the map the player physically couldn’t reach) was used to prevent the watershed segmentation from extending out of the physical bounds of the level. The mask was constructed by taking all in bound trajectory coordinates and constructing a grid, with areas that had been visited being assigned a value of 1, and areas that had not been visited retaining a value of 0. Whilst this method doesn’t strictly exclude the possibility that there may have been in-bound areas that no user had visited, owing to the vast wealth of data this was not only unlikely but would also not affect the subsequent analyses performed in this project.

The watershed segmentation was performed on two different scales, a macro scale shown in figure 22 (segmenting the whole level map into segments), and a micro scale shown in figure 23 (segmenting each macro region further into smaller micro regions).

A screenshot of a game

Description automatically generated

*Figures 22 (left): A plot of the macro regions defined from the watershed segmentation algorithm, with each region defined as a different colour on the level 56 map depicted. Centres of each region are shown as markers (x).*

*Figure 23 (right): A plot of the micro regions defined from the watershed segmentation algorithm within each macro region, with each macro regions defined as distinct colours, and micro regions defined as different gradient colours within each of these macro regions.*

This approach allowed for the graph constructed from these regions (later detailed) to capture two different aspects of the data:

1. The macro scale – Identifying broader areas of the SHQ level that represent regions of high activity and interest
2. The micro scale – Focusing on the areas of interest within each macro region by identifying areas of low activity/areas that are avoided, i.e. the areas that are more likely to represent abnormal behaviours

Contrastingly to the macro region segmentation, micro region segmentation was calculated using local minima of each macro region as basins, segmenting each region into subsequent smaller regions that represent areas that people tended to avoid.

Building on this, the centres of each micro region was calculated, shown in figure 24, which were used as the basis for graph construction, with each micro region corresponding to a node on the graph.

A colorful pattern on a black background

Description automatically generated

*Figure 24: A plot of the centres of each of the macro regions (red markers – o) and micro regions (black markers – o), with the micro region centres acting as the basis for the graph structure.*

The constructed graph for this project, depicted in figure 25, is comprised of 141 nodes, one for each of the micro segments. The figure includes example edges that represent all possible paths taken by users between these nodes, with the weight of this edge signifying the frequency of traversal, and hence the most frequent node path taken. As each coordinate on level 56 maps to a micro region, this means that every trajectory coordinate (x, y) can be directly associated with a specific node, allowing for every trajectory to be represented on the graph and hence fed into the spatial Temp2SpaceNN model.

A graph with blue dots and lines

Description automatically generated

*Figure 25: An example visualisation of the graph structure formed from the micro regions. The graph is comprised of 141 nodes (blue). Example edges were constructed to better represent the graph as a visualisation, with the density of travel between nodes as edge weights.*

### 3.54 Spatial model

The spatial Temp2Space graph convolutional neural network model, utilised in this project, is designed to transform the temporal signal into a spatial representation. The model relies on transforming the same 3-dimensional PyTorch tensor of shape (trajectories, length, features) utilised in the previous temporal CNN model, and graft the signal to a spatial domain, so that spatial based patterns can be learned by the model. As the same temporal signal is used in the spatial model, identical to that of the one used in the temporal CNN model, the same 8 features are analysed: x, y, speed, acceleration, cos(θ), sin(θ), delta cos(θ), and delta sin(θ).

The model performs this through the use of a graph contribution matrix tensor – of shape (trajectories, length, graph node assignments) – which was constructed through the use of a get\_micro\_node\_from\_coord function. This function takes a trajectory coordinate and maps it to its associated micro region; as each micro region is associated with a node on the graph, the function essentially allows for the transformation of a coordinate to a node on the graph. Therefore, each trajectory was processed through a loop mapping each x and y pair to a node for the length (523) of the trajectory, storing each trajectory in the first dimension, each of the 523 samples from a trajectory in the second dimension, and the node assignments of each sample in the third dimension. Node assignments represented a vector of 141 elements (for each of the nodes of the graph), where 140 0s corresponded to the un-associated nodes, and a single one indicated the associated node.

One issue faced in the construction of the graph contribution matrix was imposed by the resampling cubic interpolation method used to resample trajectories to the same length, as the new trajectory points did not strictly align to the integer grid coordinates. To combat this, the get\_micro\_node\_from\_coord function rounded all resampled points to the nearest integer, essentially forcing the resampled points to align to the Sea Hero Quest level grid. In cases where this resulted in the coordinate being mapped to a non-existent node (the point falling out-of-bounds), the previous in-bound node was taken in its stead. It should be noted that this method resulted in a degradation of the original signal that could impact subsequent results obtained. Other methods such as a different resampling method such as that of a rediscretization method, or a non-convolutional based deep learning approach that entirely bypasses the need for resampling, could be explored in future research, to better preserve the integrity of the signal.

The Temp2SpaceNN model functions in three stages, first processing the temporal signal, then pooling the resulting signal to the spatial domain using the graph contribution matrix, and finally processing the spatial signal, producing 10 output classes, one for each of the demographic labels.

The first stage takes the input tensor and applies a series of two 1-dimensional convolutional layers, with each of these followed by a batch normalisation layer (normalising the outputs of the convolutional layers to stabilise and accelerate training) and a ReLU layer (introducing non-linearity to better learn complex patterns) to learn temporal patterns from the input signal. Unlike the CNN model that processes the two convolutional layers in parallel (one for learning immediate patterns and one for the longer-term patterns), the Temp2SpaceNN processes the two layers sequentially, ignoring the dual-learning approach, in favour of first learning local patterns in the first convolutional layer (transforming the 8 input channels (features) to 32 output channels), then learning high-level temporal patterns in the second layer (transforming the 32 channels to 64). A kernel size of 5 and optimal padding of 2 were used in both, the same as the immediate trends convolution in the CNN.

Following on, the output learnt from the convolutional layers are pooled to the spatial domain, with this performed through two operations, an einsum operation and a max pooling operation. The einsum operation () performs a multiplication operation between the graph contribution matrix () and the output signal of the temporal convolutions (), mapping the temporal features to each node on the graph, resulting in a tensor of shape (). With this, the max pooling operation reduces the produced tensor to by focusing on the most significant contributions of each node across the time (trajectory length) dimension , by picking out the maximum values across all time stamps. These operations in conjunction effectively condense the temporal dynamics into a spatial framework, taking the temporal changes and mapping them to a spatial representation.

Finally, the output signal from the pooling operation is fed into a set of two linear layers (fully connected layers) that, along with the help of a ReLU and batch normalisation layer, learn spatial patterns (by transforming the signal into a hidden 100-dimensional space) and transforms the data to the desired output size of 10, one for each of the demographic labels to predict.

## 3.6 Model Training, Validation, and Testing

To reach the research objectives of this project we define the model protocol as such: the parameters for both the CNN and Temp2SpaceNN models are determined through a multi-class classification task, utilising the trajectory data as an input and produces predictions for 10 combined age-gender demographic labels, reflecting the user's age and gender categories.

### 3.61 Hardware

Model training, validation, and testing was performed on local hardware. Model training was accelerated by leveraging CUDA cores. By pushing the tensors and models to the CUDA device, this significantly reduced the neural network models’ training times by parallelising computations. Notably, 24GB of dedicated GPU memory and 64GB of RAM were used in carrying out this project. The large amount of memory available allowed for tensors to entirely fit within available memory, and so batching was not used for this project. Whilst we found that the CNN training ran fully in dedicated GPU memory, we should note that the Temp2SpaceNN model exceeded the 24GB of dedicated GPU memory in training and validation, pulling from shared GPU memory that slowed down model training significantly. Whilst not carried out, in retrospect, applying a batching to model training would allow the Temp2SpaceNN to run solely on dedicated GPU memory, speeding up training significantly.

### 3.62 Dataset sampling

A sample of 1000 trajectories was selected from the overall dataset for the purposes of model training, validation, and testing; this sample size was deemed sufficient to demonstrate the model's performance capabilities, whilst ensuring efficient use of computational resources. Data was split using 80/20 (training/testing) stratified sampling to ensure equal representations of the 10 demographic labels in both the training and testing set.

The same method of training, validation, and testing was applied to both the temporal and spatial models to ensure that comparisons made between their results were because of the models themselves, and not the differences in training or validation procedures.

### 3.63 Training and validation

k-fold cross validation was implemented to ensure generalisability on unseen data. k-fold cross validation, illustrated in figure 26, is technique used wherein the training data is partitioned into k equal folds.

A screenshot of a computer

Description automatically generated

*Figure 26: An illustration of the k-fold cross validation technique, with k = 5. From: (SciKit-Learn, 2009).*

In each split, one of the five folds is allocated to be the validation set, whilst the other four folds collectively serve as the training set, with the validation set fold unique to the k splits to ensure that each of the k splits act as a validation set precisely once. Whilst an alternate method could have been to simply take a percentage of the training data to act as a validation set, utilising k-fold cross validation provides a number of significant advantages over either not using a validation set, or only using a singular pre-defined set, including improving the model’s generalisation to new unseen data, as well as reducing bias owing to performance across different splits. Additionally, this method also maximises data efficiency, as each trajectory of the dataset is used exactly once between all splits as a validation. For the use case of this project, k was set to 5, as this allowed for equal distributions of samples (160 sampled for each partition).

Both loss functions and optimisers are key components in the training and evaluation of machine learning models.

A loss function is a mathematic method used to quantify the error between a model’s predictions and the actual data, with a larger number indicating that the predictions were far off, and a smaller number indicating that the predictions were good (Hastie et al., 2009) (Alake, 2023). With relevance to model training, the loss function guides the optimisation of the model by indicating the direction of adjustment in model parameters to minimise model error, and hence improving the model’s performance. Whilst other loss functions such as hinge binary cross-entropy loss, hinge loss, and log loss were also suited for classification tasks, cross entropy loss was chosen as the loss function for this project, owing to its effectiveness in multi-class classification tasks, penalising incorrect predictions of labels and optimising probability estimates for each of the demographic labels.

In model training, optimisation refers to the process of adjusting the model parameters to minimise the error quantified by the loss function, with the aim of finding the optimal parameters that result in the most accurate classification of unseen data. Stochastic gradient descent (SGD) optimisation is one such optimisation technique, that updates model parameters incrementally for each instance of the training data, over the entire training set at once (Ruder, 2017), resulting in faster convergence to the optimal parameters, owing to its ability to navigate out of local minima from the inherent randomness in the update steps. Therefore, SGD was used as the optimisation method over the other optimisation methods such as traditional batch gradient descent.

Finally, manual exploration of the number of epochs (a complete pass of the training dataset through the model’s algorithm) was conducted, with plots of both accuracy over epochs, and cross entropy loss over epochs constructed to assess the optimal number of epochs among of epochs for each model. When training the models to 50 epochs, the same trends captured in the plots appeared for the validation set. Therefore, it was determined that 15 epochs provided a good balance of sufficient model training with not overfitting the data.

### 2.64 Hyperparameter optimisation

Hyperparameter optimisation was performed using the Optuna package, with a series of 100 trials performed to investigate the optimum parameters for each of the models, over 50 epochs. The optuna package was utilised, owing to its ability to automatically analyse and identify the optimal hyperparameter values through Bayesian optimisation (Budakoğlu, 2023), a process that narrows down the optimal values by learning from the performance of past trials. This approach was utilised over that of grid search, as it is a more computational efficient method and allows for a more targeted approach, with better overall performance. The hyperparameters tested for each model, the range tested for each, and rationale is detailed in table 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Hyperparameter | Description | Range | Rationale | Model |
| Learning rate | Rate at which the model learns | 0.00001 to 0.1 | Balances efficient + stable learning with overshooting/slow convergence | CNN + Temp2SpaceNN |
| Momentum | Parameter enhancing optimiser speed/stability by leveraging past gradients | 0.5 to 0.9 | Balances quick convergence with risk of overshooting | CNN + Temp2SpaceNN |
| h | Signal dimension at each convolutional layer | [8, 16, 32] or [8, 32, 64] | Balancing model learning + performance with overfitting | CNN |
| Layers | Number of hidden neurons in the full-connected layers | 64, 128, or 256 | Balancing model learning + performance with overfitting | CNN |

*Table 2: A table of the 4 hyperparameters optimised in this project, namely learning rate, momentum, h (convolution signal dimension), and layers (number of hidden neurons in linear layers). The table notes the ranges of values tested in the 100 optuna hyperparameter optimisation trials, the rationale behind the ranges, and the model that these hyperparameter optimisations were applied to.*

### 3.65 Testing

In evaluating both CNN and Temp2SpaceNN models, they were first set to evaluation mode, using model.eval(), to ensure that the model’s behaviours were set correctly, specifically that normalisation layers use running statistics and dropout layers a deactivated.

Each of the models were run on the test dataset to verify their generalisation performance on unseen data, with model performance evaluated using 3 key metrics: accuracy scores, a confusion matrix, and a classification report. The results were visualised as a confusion matrix heatmap, that provides class specific numeric insights of predicted vs actual, and a classification report table that includes detailed metrics such as precision, recall and F1 score by demographic label. Overall, a culmination of multiple evaluation methods was utilised for their ability to provide a comprehensive overview of the model’s ability to predict combined demographic age-gender labels. A description of each performance matrix along with its benefit of use is provided in table 3.

|  |  |  |
| --- | --- | --- |
| Performance Metric | Description | Benefit of Use |
| Accuracy | A simple measure of overall accuracy of the model’s prediction across all labels | Provides how accurate the model is, which can be used to compare with different models |
| Confusion Matrix | A detailed breakdown of the predictions in the predicted vs actual demographic labels, highlighting false positives + negatives | Provides a comprehensive overview of the model’s performance and errors across the different labels |
| Precision | The proportion of positive identifications that were actually correct | Provides a measure to minimise false positives, important for high-stakes positive prediction accuracy scenarios |
| Recall | The proportion of actual positives that were identified correctly | Provides insight into the model's ability to detect all true positives, important for scenarios where missing any is detrimental |
| F1-Score | The harmonic mean of precision and recall values | Provides a balanced metric of precision and recall, useful for model comparison in imbalanced datasets |
| Support | The number of actual occurrences in each demographic label | Provides data on class frequency, important for contextualising model metrics |

*Table 3: A table of the 6 performance metrics used to analyse the performance of both the Convolutional Neural Network, and Temp2Space Neural Network, along with a description of each metric, and the benefit of using this performance metric – accuracy, confusion matrices, precision, recall, f1-score, and support.*

# 4 Results

## 4.1 Training Performance: Cross Entropy Loss and Accuracy

In order to evaluate the effectiveness of the model training process, two performance indicators were measured. Cross entropy loss and accuracy plots, tracked across 15 epochs for both the CNN and Temp2SpaceNN models offer a representation of the models’ optimisation and learning behaviours, shown in figure 27.

A graph with a line

Description automatically generatedA graph with a blue line

Description automatically generatedA graph with a line

Description automatically generatedA graph with a line

Description automatically generated

*Figure 27: A multiplot of a cross entropy loss plot over number of epochs and accuracy plot over epochs for the Convolutional Neural Network (left), and Temp2Space Neural Network (right) respectively.*

The CNN cross entropy plot highlights the general trend that an increase in the amount of epochs reduced the cross entropy of both the training and validation, though to a limited extent. We can see (though minor) an increase in validation accuracy with the number of epochs, with the highest accuracy measured at 0.11 on the 14th epoch. The accuracy plot shows an ever increasing gap between training and validation sets over the number of epochs. Lowest recorded cross entropy loss for the validation set was recorded at ~2.3 (15th epoch), and highest recorded accuracy was recorded at 0.11 (14th epoch).

Furthermore, the Temp2SpaceNN cross entropy plot highlights the general trend that, with an increase in epochs used in model training, the cross entropy loss of the training set improved significantly, reducing by a cross entropy loss of > 1.2 over the 15 epochs shown in the plot. Whilst this is the case, an increase in epochs did not significantly reduce the cross entropy loss of the validation set, evidenced by a mostly straight validation (orange) line on the plot. Furthermore, this same pattern can be seen in the Temp2SpaceNN accuracy plot, with accuracy increasing in the training set, whilst remaining relatively constant in the validation over the number of epochs. Lowest recorded cross entropy loss for the validation set was recorded at ~2.25 (13th epoch), and highest recorded accuracy was recorded at 0.16 (15th epoch).

## 4.2 Temporal Convolutional Neural Network Results

The second phase of analysis focused on an evaluation of the temporal CNN model on the test trajectory set, looking to see how well the model performed in using temporally derived patterns to predict demographic labels. This comprehensive study was conducted looking at the overall accuracy of the model, the confusion matrix, and the classification report.

First, **model accuracy** was found to be **9.00%**. Blind prediction accuracy of a 10 label classification problem is statistically 10%, indicating that the model’s performance is worse than simply guessing at random.

Furthermore, a confusion matrix of the predicted labels vs actual (true) labels was constructed using sklearn.metrics confusion matrix and matplotlib (for visualisation), shown in figure 28.

A screenshot of a computer generated diagram

Description automatically generated

*Figure 28: A confusion matrix for the Convolutional Neural Network predictions on the test set, with the 10 predicted demographic labels shown on the horizontal axis, and the actual (true) labels on the vertical axis. Each cell denotes the count into each category.*

The confusion matrix indicates that the temporal CNN model tended to predict the majority of demographics as the f-age\_4 demographic, with the highest prediction of the confusion noted as predicting females of age group 3 as group 4, with a count of 17. Furthermore, there were seemingly little to no prediction for the entirety of the first 4 female age groups. When departing form the f-age\_4 prediction pattern, looking at the male demographic highlights a trend in young males (m-age\_0 and m-age\_1), predicted as the male age group 0 demographic.

Finally, a classification report was constructed using sklearn’s classification\_report, providing a detailed summary of the precision, recall, f1-score, and support values for each of the 10 demographic labels, shown in table 4.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-score | Support |
| f-age\_0 | 0.00 | 0.00 | 0.00 | 25 |
| f-age\_1 | 0.17 | 0.04 | 0.06 | 26 |
| f-age\_2 | 0.00 | 0.00 | 0.00 | 23 |
| f-age\_3 | 0.00 | 0.00 | 0.00 | 21 |
| f-age\_4 | 0.05 | 0.62 | 0.09 | 8 |
| m-age\_0 | 0.16 | 0.35 | 0.22 | 20 |
| m-age\_1 | 0.14 | 0.08 | 0.10 | 25 |
| m-age\_2 | 0.15 | 0.08 | 0.11 | 25 |
| m-age\_3 | 0.00 | 0.00 | 0.00 | 19 |
| m-age\_4 | 0.08 | 0.12 | 0.10 | 8 |
|  | | | | |
| Accuracy |  |  | 0.09 | 200 |
| Macro average | 0.08 | 0.13 | 0.07 | 200 |
| Weighted average | 0.08 | 0.09 | 0.06 | 200 |

*Table 4: A table of the classification report results produced by the Convolutional Neural Network’s predictions, including precision, recall, f1-score, and support values for each of the 10 demographic labels, along with the overall accuracy, the average of each metric (macro average), and weighted average.*

No significant precision values were found for the temporal model, with a peak precision value of 0.17 (in female age group 1) recorded across the demographic labels. Furthermore, f-age\_4 stood out with a significantly higher recall in comparison to the other demographics, with a value of 0.62. Similarly, m-age\_0 also showed significant contrast to the general trend of recall values across the demographic labels (with a value of 0.35), though to lesser extent than f-age\_4. Notably 4 demographic groups (f-age\_0, f-age\_2, f-age\_3, and m-age\_3) has a recall of 0.00, with the same 4 observed to have precision scores and f1-scores of 0.00 as well. Male age group 0 were found to have the highest f1-score, with a value of 0.22. Though not significantly high, the value trended against the other demographic label’s f1-scores.

Finally, the macro averages of 0.08, 0.13, and 0.07 highlights that, on average, the data shows low precision, recall, and f1-scores across all demographics, which when taking into account the support values (number of instances of each demographic label in the test data), the model’s performance is even poorer (evidenced by lower weighted averages.

## 4.3 Spatial Temp2Space Neural Network Results

The final analysis focused on an evaluation of the spatial Temp2SpaceNN model on the test trajectory set, looking to see how well the model performed in using spatial patterns to predict demographic labels, with accuracy, the confusion matrix, and the classification report all giving insights into the model’s performance.

First, **model accuracy** was found to be **21.00%**. As blind prediction accuracy of the 10 demographic groups would statistically be 10%, the model accuracy result suggests that the model performs over twice as well as random chance in correctly classifying the demographic groups.

Furthermore, a confusion matrix of the predicted labels vs actual (true) labels was constructed, shown in figure 29.

A screenshot of a computer screen

Description automatically generated

*Figure 29: A confusion matrix for the Temp2Space Neural Network predictions on the test set, with the 10 predicted demographic labels shown on the horizontal axis, and the actual (true) labels on the vertical axis. Each cell denotes the count into each category.*

The confusion matrix indicates that the graph Temp2SpaceNN model tended to predict trajectories into 1 of 3 demographic labels: female-age\_group\_0 (f-age\_0), female-age\_group\_1 (f-age\_1), and male-age\_group\_1 (m-age\_1), with the overwhelming majority of predictions made solely in these 3 demographics.

Additionally, within the confusion matrix, the cell intersection predicted and true labels for both females age group 0 and males age group 1 displayed significant amounts of correct predictions, with 16 and 14 respectively, highlighting that these were the most accurately predicted demographic labels by the Temp2SpaceNN model. It should be noted that the model frequently misclassified the f-age\_1 demographic as f-age\_0, with a count of 14, and also frequently misclassified the m-age\_2 demographic as m-age\_1

Finally, a classification report was constructed, providing a detailed summary of the precision, recall, f1-score, and support values for each of the 5 female and 5 male demographic labels with age, shown in table 5.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-score | Support |
| f-age\_0 | 0.19 | 0.64 | 0.29 | 25 |
| f-age\_1 | 0.21 | 0.35 | 0.26 | 26 |
| f-age\_2 | 0.00 | 0.00 | 0.00 | 23 |
| f-age\_3 | 1.00 | 0.05 | 0.09 | 21 |
| f-age\_4 | 0.00 | 0.00 | 0.00 | 8 |
| m-age\_0 | 1.00 | 0.05 | 0.10 | 20 |
| m-age\_1 | 0.21 | 0.56 | 0.31 | 25 |
| m-age\_2 | 1.00 | 0.04 | 0.08 | 25 |
| m-age\_3 | 0.00 | 0.00 | 0.00 | 19 |
| m-age\_4 | 0.00 | 0.00 | 0.00 | 8 |
|  | | | | |
| Accuracy |  |  | 0.21 | 200 |
| Macro average | 0.36 | 0.17 | 0.11 | 200 |
| Weighted average | 0.41 | 0.21 | 0.14 | 200 |

*Table 5: A table of the classification report results produced by the Temp2SpaceNN’s predictions, including precision, recall, f1-score, and support values for each of the 10 demographic labels, along with the overall accuracy, the average of each metric (macro average), and weighted average.*

First, four demographic labels were identified as having a 0.00 value for all three precision, recall, and f1-scores, these being f-age\_2, f-age\_4, m-age\_3, and m-age\_4. When drawing results from the precision statistic, three demographic labels have a value of 1.00 (f-age\_3, m-age\_0, m-age\_2). Whilst this is the case, these same values show low recall values of 0.05, 0.05, and 0.04 respectively. F1-score results reveal three demographics with higher values, that deviate from the general < 0.1 trend observed through the other seven demographics: f-age\_0 (f1-score = 0.29), f-age\_1 (f1-score = 0.26), and m-age\_1 (f1-score = 0.31).

A macro average precision of 0.36 indicates that 36% of the positive identification were actually correct. In addition a higher weighted average (0.41) suggests that, when taking the number of each demographic label in the test data, precision was higher, with a positive identification of 41%.

# 5 Discussion

## 5.1 Training Performance

When interpreting the cross entropy values observed on the plots, we can compare the specific results observed in both models against a baseline naïve model, that predicts each of the 10 demographic labels purely on random chance: , with the cross entropy of this model at ~2.3. Using this, it suggests that the temporal model performed worse than that of purely guessing, even over a sufficient number of epochs. In contrast, over a sufficient number of epochs, the Temp2SpaceNN training performance suggested that the model performed slightly better than that of random chance, with a loss of less than 2.3. Whilst this is the case, a validation cross entropy value over 2, as seen in both models, suggests poor overall performance, with the lack of significant improvement over epochs indicating that the models were both poor at identifying distinct behavioural patterns that could predict the labels to any accurate extent.

Despite the training performance metrics of the temporal CNN model suggesting an initial learning trend (an improvement in cross entropy loss), this is quickly cut, evidenced by both strong plateau trends of the cross entropy loss and accuracy plots, showing a lack of improvement in the model’s ability to learn from the SHQ trajectory data to any significant depth. As accuracy of the model tends to hover around that of random chance (an accuracy of 0.1), it seems to suggest that subsequent patterns learnt by the model were superficial and lacked the power to fully grasp the underlying complexities of the data necessary to predict the joint demographics to any significant level.

The Temp2SpaceNN model did show slight improvement to cross entropy loss over epochs, suggesting that the model was able to learn some patterns from the data, though to no great extent. An investigation of the Temp2SpaceNN cross entropy plot seemed to suggest a trend of underfitting, highlighted by the significant disparity between the training and validation sets in both the loss function and accuracy plots, even over a significant number of epochs. There are a number of potential explanations for this underfitting pattern found, the most likely of which were either that the input features fed into the model lacked the necessary variability to accurately capture the differences in navigation behaviour (a previously mentioned risk in using SHQ data due to the game’s relatively simple level design), or that the model engineered features were too generic to effectively differentiate between the labels during learning (or more likely a combination of both).

Whilst this is the case, even at baseline (one pass of the model), the model accuracy seemed to be comparable to random chance, with an increase in the number of epochs ever so slightly improving this (to an observed accuracy of 0.16), seeming to suggest that the model was able to perform better than just randomly guessing, and hence some patterns learnt (even if not significant), were able to aid in predicting the joint age/gender demographic labels.

## 5.2 Temporal and Spatial Models: Insights and Comparisons

An evaluation into the results of both models reveal interest insights. First, the temporal CNN’s model accuracy of 9% suggests that it was poor at distinguishing temporal patterns from the trajectory data that could accurately predict the age/gender demographic of a user based on their temporal behaviours. This could imply that temporal behaviours across age & gender are not sufficiently distinct to model their differences, with aspects such as a user’s speed or acceleration not inherently tied to the age or gender of a user. For example the model results imply that a young female and a young male, or a young female and old female, all possess the same temporal characteristics. Whilst this is the case, the overall low predictive capabilities seems to imply, if there are inherent demographic differences, the model was unable to learn these. Further research is needed to discern between the two implications.

On the other hand, the Temp2SpaceNN model’s accuracy off 21% demonstrated a significantly better classification ability than that of the strictly temporal model. With the Temp2SpaceNN ability to predict the joint demographic of a user over twice that of random chance, it implies that the model was able to discern navigational behaviours that differed in the demographic groups. Whilst this model did model both temporal and spatial patterns, when comparing this result to the poor performance of the CNN (that modelled only temporal patterns), the improvement in accuracy (with the Temp2SpaceNN model over 2x as accurate) implies that the patterns learned were from that of the spatial domain, not temporal domain. This implication could suggest that navigational behaviours between the demographics varied in terms of spatial patterns, such as the routes they took or the specific locations they frequented. Notably, these results coincided with the results of (Dubois, 2023) with their Temp2SpaceNN sub-model outperforming the CNN sub-model across all five levels tested.

A closer inspection of the confusion matrix of the Temp2SpaceNN highlights that the model was marginally able to discern males from females (with predicted female labels more commonly associated to true female labels, and predicted male labels more commonly associated to true female labels). Additionally the matrix also seems to suggest that the model was able to slightly discern between young vs old age groups (with the general trend that young predictions were more commonly mapped to younger true labels). It should be noted that the model frequently misclassified f-age\_1 demographics as f-age\_0 (a count of 14), and m-age\_2 as m-age\_1 (a count of 11), which could suggest that whilst slight discernible patterns between young and old demographics are evident, differences in navigational behaviours within the young and old populations themselves were too insignificant for the model to accurately assign the correct younger, or older, demographic label.

As general patterns emerged that suggested that the model was better at distinguishing young from old (even with misclassifications within these) a refinement to this study could be to simplify the study to predict 3 age ranges instead of 5 – whilst this provides for a less nuanced overall understanding, it would likely improve model performance.

Furthermore, a large contribution to the results of this study was from feature selection. Whilst the models used in this project were designed to learn from 8 trajectory based features, it is apparent that these selected features were unable to significantly capture navigational patterns required for age/gender prediction, suggesting that other feature utilisation (such as the direct use of duration in the models rather than relying on duration baked into the speed feature) could improve the model’s predictive capabilities.

Another consideration is in the sample size used. Whilst it was thought that 1000 samples would be sufficient to train the model, this may not have been the case, with a larger sample size needed to learn significant patterns. Therefore, increasing the sample size (using a batching process to allow for this), could uncover if there was insufficient data used in this study. In the same vein, not enough consideration was put into mitigating bias introduced through the imbalance of the demographic labels in training – instead of only using stratified sampling, future research could instead consider using a cutoff based method that equalises the amount of trajectories in each of the demographics, to mitigate this issue.

The overall relatively poor performance of both models (peaking at 21% accuracy) raises multiple important considerations regarding modelling navigational behaviours, specifically:

1. The inherent complexity of navigational patterns – Navigational patterns are ‘muddy’ in that there are so many factors influencing them. This makes strong predictions based on specific demographics challenging.
2. There may just be an inherent lack of differences in navigational patterns between the demographics studied. Whilst general research in the area (Cazzato et al., 2010) (Boone et al., 2018) (Cánovas et al., 2008) (Xia et al., 2018) tended to suggest cognitive differences between both age demographics and gender demographics, there is still a lack of definitive research and a lack of understanding into the underlying mechanisms, keeping the possibility open that there may just not be discernible differences within the demographics of this study, and that differences in navigational patterns are caused by a different, currently unknown factor.
3. The possibility that existing differences in the navigational behaviours of specific demographics do exist to a measurable a classifiable level, but that the data size, techniques, features, and models used in this project were insufficient to effectively capture these distinctions.

Nonetheless, it is clearly evident, through the misclassification results visually shown in the confusion matrices in addition to the low general precision results, that both of these models are not suitable for tasks requiring high precision in behavioural prediction, such as adapting the model for Alzheimer's detection, owing to the significant misclassification rate. As Alzheimer’s detection relies on identifying abnormal trajectories, a high misclassification rate undermines the reliability of the model, as it would be unclear if the trajectory was misclassified due to actually being anomalous, or just as a result of poor model performance. The personal consequences of misdiagnosis for dementia means that a high bar must be set for accuracy in real-world application.

# 6 Conclusion

## 6.1 Summary

Overall, this study set out to establish a methodological framework that was able investigate whether the combined age and gender demographics of Sea Hero Quest could be predicted, based on their navigational behaviours.

Two models were implemented to carry this out, the first (a conventional CNN) analysing temporal and spatial progression dependencies of the trajectory data, and the second (a Temp2SpaceNN) analysing temporal, spatial progression, and spatial dependencies. Both of these models were build using a convolutional neural network architecture, that took 8 features of the trajectory data (namely x, y, speed, acceleration, 2 directional components, and 2 curvature components) to learn new features that captured the underlying patterns and nuances of navigational behaviours across the 10 demographic groups – these demographic groups were comprised of 2 genders (male and female), and 5 age groups (0-4).

A multitude of techniques and testing parameters were used to ensure model integrity, with a number of metrics (including accuracy, precision, recall, f1-score) and visualisations (epoch plots and confusion matrices) used to investigate how the learning capabilities of each of the models, and the power of the learned models on unseen data. Results of the study showed overall poor predictive performance, with testing accuracies of both models at 9% and 21% respectively. Whilst the temporal CNN performed worse than randomly guessing, the Temp2SpaceNN performed considerably better, being over 2x as accurate at predicting a demographic label than from randomly guessing. The most significant demographic was in the Temp2SpaceNN model, showing slight discernability between demographic populations. Whilst this is the case, it is important to contextualise this with the information that it is currently unknown if the behavioural differences modelled are as a result of intrinsic biological factors, or just a product of environmental factors.

This study gave rise to a number of questions and considerations, that provide avenues for further research:

1. Investigating if the sample size was sufficient to learn comprehensive patterns
2. Investigating feature selection and if the addition/removal of new trajectory features could help improve model performance
3. Exploration of other less commonly explored demographic factors, such as socio-economic background, educational background, etc.
4. Delving into the models of this paper, looking at the learned weights and features of the models and mapping these to the level 56 map, that could provide visual understanding of where the information lies, and what areas of the level were signifiers of differences in navigational behaviours.
5. Although not covered in the scope of this project due to time limitations, a potential major piece of further research could be in adapting the Temp2SpaceNN to either:
   1. Not only pool the learned signal to the micro graph, but to simultaneously pool the signal to the macro graph, with an aggregation joining these signals (similar to what was carried out in the temporal CNN).
   2. Utilise a bipartite graph (shown in figure 30) to aggregate learned features to the macro graph to learn more broad patterns on location behaviours.

…with these methods potentially allowing for the model to capture a more unified and comprehensive understanding navigational patterns.

A diagram of a person with red dots and blue dots

Description automatically generated

*Figure 30: A plot of a bipartite graph, with micro nodes shown in blue and macro nodes shown in red. The graph is constructed with directed edges, representing the passing of information from micro to macro node in a single direction. As no node of the same graph layer (macro/micro) has an edge to another of its node, the graph is considered bipartite.*

## 6.2 Limitations of the Report

A number of limitations of this study can be identified:

1. Owing to a lack of ground truth in relation to behavioural differences in navigation and demographic populations, a lot of the results lacked conclusive findings, with only suggestions as to what could be the cause of the results produced.
2. The choice of resampling method introduced a degradation to the signal that was likely to have impacted the results of the study, and the model’s ability to learn patterns. Exploration of other resampling methods, or the use of non-convolutional based neural networks could resolve this.
3. Optimal number of epochs was decided using manual observation of loss plots. In retrospect, an alternative method to manually optimising the number of epochs would have been to use an early termination method, that automatically halts the training process when a consecutive number of validation sets show no improvement in test metrics, preventing overfitting and reducing computational complexity.
4. The study lacked comprehensive evaluation of additional parameters such as the optimal kernel size in convolutional layers for both models, limiting this study’s ability to reasonably assess if the poor model performance was as a result of the model itself being poor or other factors. Whilst the use of testing two models allowed for relative comparisons between each, the lack of a baseline metric to evaluate against meant results could only be contextualised within the confines of the study itself, with no clear benchmark of success/failure in the wider field.

## 6.3 Ethical Considerations

Ethical consideration was important as analysis was performed on real-world human data. The data used was completely anonymised, ensuring confidentiality of participants. Additionally, this study’s exploration of demographic differences carries the risk of unintentional discrimination – therefore we must question whether the differences found are as a result of biological differences, or just a result of environmental factors. It is important to contextualise findings to ensure that the findings of this research are not taken out of context, to be used to reinforce stereotypes/biases.

# References

Alake, R. (2023, November). *Loss Functions in Machine Learning Explained*. Datacamp. https://www.datacamp.com/tutorial/loss-function-in-machine-learning

Altan, D., Etemad, M., Marijan, D., & Kholodna, T. (2022, April 25). *Discovering Gateway Ports in Maritime Using Temporal Graph Neural Network Port Classification*. ArXiv.org. https://doi.org/10.48550/arXiv.2204.11855

Alzheimer's society. (2023, March 8). *Why is dementia different for women? | Alzheimer’s Society*. Www.alzheimers.org.uk. https://www.alzheimers.org.uk/blog/why-dementia-different-women#:~:text=In%20fact%2C%20around%20twice%20as

alzheimersresearchuk. (n.d.). *Sea Hero Quest*. Alzheimer’s Research UK. https://www.alzheimersresearchuk.org/research/for-researchers/resources-and-information/sea-hero-quest/

Bas Jacob Buijse, Vahideh Reshadat, & Enzing, O. (2021). A Deep Learning-Based Approach for Train Arrival Time Prediction. *Lecture Notes in Computer Science*, 213–222. https://doi.org/10.1007/978-3-030-91608-4\_22

Bermudez-Contreras, E., Clark, B. J., & Wilber, A. (2020). The Neuroscience of Spatial Navigation and the Relationship to Artificial Intelligence. *Frontiers in Computational Neuroscience*, *14*. https://doi.org/10.3389/fncom.2020.00063

Bhatia, J. (2023, July 15). *Exploring Image Segmentation Techniques: Watershed Algorithm using OpenCV*. Medium. https://medium.com/@jaskaranbhatia/exploring-image-segmentation-techniques-watershed-algorithm-using-opencv-9f73d2bc7c5a

Blouin, L. (2023, March 6). *AI’s mysterious “black box” problem, explained | University of Michigan-Dearborn*. Umdearborn.edu. https://umdearborn.edu/news/ais-mysterious-black-box-problem-explained

Boone, A. P., Gong, X., & Hegarty, M. (2018). Sex differences in navigation strategy and efficiency. *Memory & Cognition*, *46*(6), 909–922. https://doi.org/10.3758/s13421-018-0811-y

Brügger, A., Richter, K.-F., & Fabrikant, S. I. (2019). How does navigation system behavior influence human behavior? *Cognitive Research: Principles and Implications*, *4*(1). https://doi.org/10.1186/s41235-019-0156-5

Budakoğlu, G. (2023, March 26). *Hyper-parameter Tuning Through Grid Search and Optuna*. MLearning.ai. https://medium.com/mlearning-ai/hyper-parameter-tuning-through-grid-search-and-optuna-2bd89a2ece06#:~:text=However%2C%20grid%20search%20can%20be

Cánovas, R., Espínola, M., Iribarne, L., & Cimadevilla, J. M. (2008). A new virtual task to evaluate human place learning. *Behavioural Brain Research*, *190*(1), 112–118. https://doi.org/10.1016/j.bbr.2008.02.024

Carroll, M., Lin, J., Paradise, O., Georgescu, R., Sun, M., Bignell, D., Milani, S., Hofmann, K., Hausknecht, M., Dragan, A., & Devlin, S. (2022, December 9). *Towards Flexible Inference in Sequential Decision Problems via Bidirectional Transformers*. ArXiv.org. https://doi.org/10.48550/arXiv.2204.13326

Cazzato, V., Basso, D., Cutini, S., & Bisiacchi, P. (2010). Gender differences in visuospatial planning: An eye movements study. *Behavioural Brain Research*, *206*(2), 177–183. https://doi.org/10.1016/j.bbr.2009.09.010

Chen, X., Liu, Y., Achuthan, K., & Zhang, X. (2020). A ship movement classification based on Automatic Identification System (AIS) data using Convolutional Neural Network. *Ocean Engineering*, *218*, 108182. https://doi.org/10.1016/j.oceaneng.2020.108182

Coughlan, G., Coutrot, A., Khondoker, M., Minihane, A.-M., Spiers, H., & Hornberger, M. (2019). Toward personalized cognitive diagnostics of at-genetic-risk Alzheimer’s disease. *Proceedings of the National Academy of Sciences*, *116*(19), 9285–9292. https://doi.org/10.1073/pnas.1901600116

Coutrot, A., Manley, E., Goodroe, S., Gahnstrom, C., Filomena, G., Yesiltepe, D., Dalton, R. C., Wiener, J. M., Hölscher, C., Hornberger, M., & Spiers, H. J. (2022). Entropy of city street networks linked to future spatial navigation ability. *Nature*, *604*(7904), 104–110. https://doi.org/10.1038/s41586-022-04486-7

Coutrot, A., Schmidt, S., Coutrot, L., Pittman, J., Hong, L., Wiener, J. M., Hölscher, C., Dalton, R. C., Hornberger, M., & Spiers, H. J. (2019). Virtual navigation tested on a mobile app is predictive of real-world wayfinding navigation performance. *PLOS ONE*, *14*(3), e0213272. https://doi.org/10.1371/journal.pone.0213272

Coutrot, A., Silva, R., Manley, E., de Cothi, W., Sami, S., Bohbot, V. D., Wiener, J. M., Hölscher, C., Dalton, R. C., Hornberger, M., & Spiers, H. J. (2018). Global Determinants of Navigation Ability. *Current Biology*, *28*(17), 2861-2866.e4. https://doi.org/10.1016/j.cub.2018.06.009

Creative, C. (2022, March 8). *What Are the 7 Executive Functions?* Igniteachievementacademy.com. https://igniteachievementacademy.com/blog/what-are-the-7-executive-functions

Cross, C. (2018). *Taxi Trajectory Data*. Www.kaggle.com. https://www.kaggle.com/datasets/crailtap/taxi-trajectory

Derrow-Pinion, A., She, J., Wong, D., Lange, O., Hester, T., Perez, L., Nunkesser, M., Lee, S., Guo, X., Wiltshire, B., Battaglia, P. W., Gupta, V., Li, A., Xu, Z., Sanchez-Gonzalez, A., Li, Y., & Velickovic, P. (2021). ETA Prediction with Graph Neural Networks in Google Maps. *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. https://doi.org/10.1145/3459637.3481916

Dubois, H. (2023, June 16). *Apprentissage automatique pour l’analyse de trajectoires spatiales : extraction conjointe de caractéristiques démographiques et comportementales*. HAL Theses. https://theses.hal.science/tel-04131533

Dubois, H., Patrick Le Callet, Hornberger, M., Spiers, H. J., & Coutrot, A. (2021). Capturing and Explaining Trajectory Singularities using Composite Signal Neural Networks. *HAL (Le Centre Pour La Communication Scientifique Directe)*. https://doi.org/10.23919/eusipco47968.2020.9287403

Escottá, Á. T., Beccaro, W., & Ramírez, M. A. (2022). Evaluation of 1D and 2D Deep Convolutional Neural Networks for Driving Event Recognition. *Sensors*, *22*(11), 4226. https://doi.org/10.3390/s22114226

Fan, Z., Yang, X., Yuan, W., Jiang, R., Chen, Q., Song, X., & Ryosuke Shibasaki. (2022). *Online trajectory prediction for metropolitan scale mobility digital twin*. https://doi.org/10.1145/3557915.3561040

Graser, A., Jalali, A., Lampert, J., Weißenfeld, A., & Janowicz, K. (2023). *Deep Learning From Trajectory Data: a Review of Deep Neural Networks and the Trajectory Data Representations to Train Them*. https://ceur-ws.org/Vol-3379/BMDA\_2023\_paper\_7556.pdf

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning, second edition : data mining, inference, and prediction* (2nd ed.). Springer.

Hong, Y., Martin, H., & Raubal, M. (2022). How do you go where? *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*. https://doi.org/10.1145/3557915.3560996

IBM. (2023a). *What Are Neural Networks? | IBM*. Www.ibm.com; IBM. https://www.ibm.com/topics/neural-networks

IBM. (2023b). *What is Deep Learning?* Www.ibm.com. https://www.ibm.com/topics/deep-learning

IBM. (2023c). *What is Machine Learning?* IBM. https://www.ibm.com/topics/machine-learning

Ishikawa, T. (2022). Individual Differences and Skill Training in Cognitive Mapping: How and Why People Differ. *Topics in Cognitive Science*. https://doi.org/10.1111/tops.12605

Islam, M., Chen, G., & Jin, S. (2019). An Overview of Neural Network. *American Journal of Neural Networks and Applications*, *5*(1), 7. https://doi.org/10.11648/j.ajnna.20190501.12

Kashyap, A. A., Raviraj, S., Devarakonda, A., Nayak K, S. R., K V, S., & Bhat, S. J. (2021). Traffic flow prediction models – A review of deep learning techniques. *Cogent Engineering*, *9*(1). https://doi.org/10.1080/23311916.2021.2010510

Kim, J., Zheng, K., Corcoran, J., Ahn, S., & Papamanolis, M. (2017). *Trajectory Flow Map: Graph-Based Approach to Analysing Temporal Evolution of Aggregated Traffic Flows in Large-Scale Urban Networks*. https://arxiv.org/ftp/arxiv/papers/2212/2212.02927.pdf

Kumar, A., Sidhu, J., Goyal, A., & Tsao, J. W. (2022). *Alzheimer Disease*. PubMed; StatPearls Publishing. https://www.ncbi.nlm.nih.gov/books/NBK499922/#:~:text=Alzheimer%20disease%20(AD)%20is%20the

Laczó, M., Martinkovic, L., Lerch, O., Wiener, J. M., Kalinova, J., Matuskova, V., Nedelska, Z., Vyhnalek, M., Hort, J., & Laczó, J. (2022). Different Profiles of Spatial Navigation Deficits In Alzheimer’s Disease Biomarker-Positive Versus Biomarker-Negative Older Adults With Amnestic Mild Cognitive Impairment. *Frontiers in Aging Neuroscience*, *14*. https://doi.org/10.3389/fnagi.2022.886778

Levine, T. F., Allison, S. L., Stojanovic, M., Fagan, A. M., Morris, J. C., & Head, D. (2020). Spatial navigation ability predicts progression of dementia symptomatology. *Alzheimer’s & Dementia*, *16*(3), 491–500. https://doi.org/10.1002/alz.12031

Li, W.-Y., Li, Y., Zhang, C., Sun, F., Meng, F., Guo, A., & Jin, D. (2018). DeepMove. *The Web Conference*. https://doi.org/10.1145/3178876.3186058

Liatsikou, M., Papadopoulos, S., Apostolidis, L., & Ioannis Kompatsiaris. (2021). A Denoising Hybrid Model for Anomaly Detection in Trajectory Sequences. *EDBT/ICDT Workshops*.

Liatsikou, M., Papadopoulos, S., Apostolidis, L., & Kompatsiaris, I. (2021). *A Denoising Hybrid Model for Anomaly Detection in Trajectory Sequences*. https://ceur-ws.org/Vol-2841/BMDA\_4.pdf

Ma, Y., Zhu, X., Zhang, S., Yang, R., Wang, W., & Manocha, D. (2019). TrafficPredict: Trajectory Prediction for Heterogeneous Traffic-Agents. *Proceedings of the ... AAAI Conference on Artificial Intelligence*, *33*(01), 6120–6127. https://doi.org/10.1609/aaai.v33i01.33016120

Maennel, H., Alabdulmohsin, I., Tolstikhin, I., Baldock, R. J. N., Bousquet, O., Gelly, S., & Keysers, D. (2020, November 11). *What Do Neural Networks Learn When Trained With Random Labels?* ArXiv.org. https://doi.org/10.48550/arXiv.2006.10455

Mandal, P. K., Joshi, J., & Saharan, S. (2012). Visuospatial Perception: An Emerging Biomarker for Alzheimer’s Disease. *Journal of Alzheimer’s Disease*, *31*(s3), S117–S135. https://doi.org/10.3233/jad-2012-120901

Mehri, S., Alesheikh, A. A., & Basiri, A. (2021). A Contextual Hybrid Model for Vessel Movement Prediction. *IEEE Access*, *9*, 45600–45613. https://doi.org/10.1109/access.2021.3066463

Musicco, M. (2009). Gender differences in the occurrence of Alzheimer’s disease. *Functional Neurology*, *24*(2), 89–92. https://pubmed.ncbi.nlm.nih.gov/19775536/

NHS. (2018, May 10). *Alzheimer’s Disease*. Nhs.uk. https://www.nhs.uk/conditions/alzheimers-disease/#:~:text=Who%20is%20affected%3F

Puthusseryppady, V., Morrissey, S., Spiers, H., Patel, M., & Hornberger, M. (2022). Predicting real world spatial disorientation in Alzheimer’s disease patients using virtual reality navigation tests. *Scientific Reports*, *12*(1). https://doi.org/10.1038/s41598-022-17634-w

Quental, N. B. M., Brucki, S. M. D., & Bueno, O. F. A. (2009). Visuospatial function in early Alzheimer’s disease: Preliminary study. *Dementia & Neuropsychologia*, *3*(3), 234–240. https://doi.org/10.1590/s1980-57642009dn30300010

Rasmussen, J., & Langerman, H. (2019). Alzheimer’s Disease – Why We Need Early Diagnosis. *Degenerative Neurological and Neuromuscular Disease*, *Volume 9*(9), 123–130. https://doi.org/10.2147/dnnd.s228939

Ravikiran, A. S. (2023, July 26). *Graphs in Data Structure: Overview, Types & More | Simplilearn*. Simplilearn.com. https://www.simplilearn.com/tutorials/data-structure-tutorial/graphs-in-data-structure

Ruder, S. (2017). *An overview of gradient descent optimization algorithms \**. https://arxiv.org/pdf/1609.04747.pdf

Sabarish, B. A., Karthi, R., & Kumar, T. G. (2020). Graph Similarity-based Hierarchical Clustering of Trajectory Data. *Procedia Computer Science*, *171*, 32–41. https://doi.org/10.1016/j.procs.2020.04.004

Salimi, S., Irish, M., Foxe, D., Hodges, J. R., Piguet, O., & Burrell, J. R. (2017). Can visuospatial measures improve the diagnosis of Alzheimer’s disease? *Alzheimer’s & Dementia: Diagnosis, Assessment & Disease Monitoring*, *10*(1), 66–74. https://doi.org/10.1016/j.dadm.2017.10.004

Schinazi, V. R., Meloni, D., Grübel, J., Angus, D. J., Baumann, O., Weibel, R. P., Jeszenszky, P., Hölscher, C., & Thrash, T. (2023). Motivation moderates gender differences in navigation performance. *Scientific Reports*, *13*(1), 15995. https://doi.org/10.1038/s41598-023-43241-4

SciKit-Learn. (2009). *3.1. Cross-validation: evaluating estimator performance — scikit-learn 0.21.3 documentation*. Scikit-Learn.org. https://scikit-learn.org/stable/modules/cross\_validation.html

Shenfield, A., & Howarth, M. (2020). A Novel Deep Learning Model for the Detection and Identification of Rolling Element-Bearing Faults. *Sensors*, *20*(18), 5112. https://doi.org/10.3390/s20185112

Simini, F., Barlacchi, G., Luca, M., & Pappalardo, L. (2021). A Deep Gravity model for mobility flows generation. *Nature Communications*, *12*(1). https://doi.org/10.1038/s41467-021-26752-4

Smith, S. W. (1997). *The scientist and engineer’s guide to digital signal processing*. California Technical Pub.

Spiers, H. J., Coutrot, A., & Hornberger, M. (2021). Explaining World‐Wide Variation in Navigation Ability from Millions of People: Citizen Science Project Sea Hero Quest. *Topics in Cognitive Science*. https://doi.org/10.1111/tops.12590

Traja. (2021). *Resampling Trajectories — Traja 22.0.0 documentation*. Traja.readthedocs.io. https://traja.readthedocs.io/en/latest/rediscretize.html

Wang, D., Zhang, J., Cao, W., Li, J., & Zheng, Y. (2018). When Will You Arrive? Estimating Travel Time Based on Deep Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). https://doi.org/10.1609/aaai.v32i1.11877

Wang, L., Ciliberto, M., Hristijan Gjoreski, Lago, P., Murao, K., Okita, T., & Roggen, D. (2021). Locomotion and Transportation Mode Recognition from GPS and Radio Signals: Summary of SHL Challenge 2021. *Queen Mary Research Online (Queen Mary University of London)*. https://doi.org/10.1145/3460418.3479373

Weisberg, S. M., & Ekstrom, A. D. (2021). Hippocampal volume and navigational ability: The map(ping) is not to scale. *Neuroscience & Biobehavioral Reviews*, *126*, 102–112. https://doi.org/10.1016/j.neubiorev.2021.03.012

Wolbers, T., & Hegarty, M. (2010). What determines our navigational abilities? *Trends in Cognitive Sciences*, *14*(3), 138–146. https://doi.org/10.1016/j.tics.2010.01.001

World Health Organization. (2023). *Dementia*. World Health Organization. https://www.who.int/news-room/fact-sheets/detail/dementia

Wu, F., Zhang, T., Holanda De Souza, A., Fifty, C., Yu, T., & Weinberger, K. (2019). *Simplifying Graph Convolutional Networks*. https://arxiv.org/pdf/1902.07153.pdf

Xia, X., Jiang, Q., McDermott, J., & Han, J.-D. J. (2018). Aging and Alzheimer’s disease: Comparison and associations from molecular to system level. *Aging Cell*, *17*(5), e12802. https://doi.org/10.1111/acel.12802

Yang, T., Wang, X., & Liu, Z. (2022). Ship Type Recognition Based on Ship Navigating Trajectory and Convolutional Neural Network. *Journal of Marine Science and Engineering*, *10*(1), 84. https://doi.org/10.3390/jmse10010084

Yang, Z., Pang, C., & Zeng, X. (2023). Trajectory Forecasting Using Graph Convolutional Neural Networks Based on Prior Awareness and Information Fusion. *ISPRS International Journal of Geo-Information*, *12*(2), 77. https://doi.org/10.3390/ijgi12020077

Zamboni, S., Kefato, Z. T., Girdzijauskas, S., Norén, C., & Dal Col, L. (2022). Pedestrian trajectory prediction with convolutional neural networks. *Pattern Recognition*, *121*, 108252. https://doi.org/10.1016/j.patcog.2021.108252

Zhang, Y., Li, Y., Xun Yu Zhou, Kong, X., & Luo, J. (2020). *Curb-GAN: Conditional Urban Traffic Estimation through Spatio-Temporal Generative Adversarial Networks*. https://doi.org/10.1145/3394486.3403127

Zheng, Y. (2011). T-Drive trajectory data sample. *Www.microsoft.com*. https://www.microsoft.com/en-us/research/publication/t-drive-trajectory-data-sample/

# Appendices

## Appendix A

|  |  |
| --- | --- |
| Term | Definition |
| Trajectory | A path taken by an object through space (x, y) over time (t) |
| Features | An individual measurable property used in data analysis and machine learning to learn patterns from |
| Feeding | The process of supplying data to a model |
| Signal | A source of information |
| Normalisation | The re-scaling of data to fall between a defined value set |
| Machine Learning | Statistical algorithms that can learn from data and generalise to unseen data |
| Deep Learning | A branch of machine learning, using neural networks (comprised of at least 3 layers) to learn patterns from data |
| Convolutional Neural Network | A class of neural network that applies convolutional operations over the input signal |
| Graph | A structure of data organised into nodes and edges |
| Kernel Density Estimation | A method that estimates the probability density function of a variable through the use of kernels and weights |
| Watershed segmentation | A segmentation method to segment larger areas into sections through the use of a basin filling method |
| Matrix | A set of values arranged in a 2-dimensional structure, e.g.: |
| Tensor | A contained of N dimensions of matrices |
| Graph contribution matrix | A matrix containing the assignments of an object/features to the set of nodes on a graph |
| Graph pooling | The process of mapping non-graph data onto a graph structure |
| Temporal to Space Neural Network | A newly proposed class of neural networks, delineated from convolutional neural networks, that pools convolutional output signals and pools them onto a graph structure to learn graph based patterns |

*Appendix A: A table of the key terms discussed throughout this dissertation project, and a definition for each.*

## Appendix B

|  |  |  |
| --- | --- | --- |
| Package/s | Description | Use |
| Gzip, json, csv | Data structure tools | Processing whole dataset as a zipped .json file in batches to filter to only level 56 data and write this to a new csv file |
| Pandas | Data analysis & manipulation tool | Handling, pre-processing, and analysis of the trajectory data |
| Numpy | Numerical computation tool | Used in performing a multitude of mathematically based operations |
| Matplotlib | Visualisation tool | Foundation for visualisations produced for this study |
| Seaborn | Visualisation tool | Foundation for statistical visualisations, and better aesthetic production of matplotlib inc. use of built in themes |
| Scipy | Mathematical equation and algorithm tool | Used to carry out a number of different advanced statistical analyses and processes: normalisation, gaussian filtering, interpolation, and kernel density estimation |
| Skimage | 2D data and image processing tool | Foundation for mapping trajectory data to a graph structure through the use of plotting peak local maxima and watershed algorithms |
| networkx | Graph construction tool | Used to construct graphs for both constructing the graph contribution matrix for the Temp2SpaceNN, and for visualisations |
| PyTorch | Machine learning tool | Foundation of all feeding and machine learning components of the project inc. processing trajectory data to tensors, and building/running the neural network models |
| Optuna | Hyperparameter optimisation tool | Used to optimised multiple hyperparameters of both models for better model performance |
| Sklearn | Predictive data analysis tool | Foundation for splitting data into train/test, label encoding, and all results of the project inc. confusion matrices and classification reports |

*Appendix B: The packages used in carrying out this project, including a description of each, and the use case in the project.*