VISUAL ANALYSIS OF MOBILE SENSING TIME-SERIES DATA:
IDENTIFYING INDIVIDUAL AND RELATIVE BEHAVIOURAL PATTERNS

by

Mohamed Muzamil H

Submitted in partial fulfillment of the requirements
for the degree of Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
July 2022

To my parents, Asifa Begum and Abdul Majeed. None of this would have been possible without your sacrifices and dedication to provide access to quality education.

Table of Contents

List of Tables

List of Figures

Abstract

Mental well-being is increasingly demanded due to growing concerns about mental health. At the same time, the Internet and smartphones are transforming the world in unprecedented ways. This pervasiveness opens up new avenues for research by providing access to an individual's behaviour and daily habits. Unobtrusive data collection and analysis from smartphone sensors is a promising approach to addressing mental health issues and have been the focus of many research studies. In this work, we explore this opportunity by analyzing data collected from smartphone usage and leveraging the advantages of data visualization and machine learning methods to possibly identify and compare behavioural indicators and patterns that can indicate mental health. We developed a visualization system to interact with extracted features about behavioural indicators like screen usage, calling, and sleep to assess the daily routine of participants under study. We also present two usage scenarios to demonstrate our visual approach's applicability in exploring the given dataset.

List of Abbreviations

CBT	- Cognitive Behavioural Therapy
AL	- Active Learning
DR	- Dimensionality Reduction
ML	- Machine Learning
SNE	- Stochastic Neighbor Embedding
t-SNE	- t-distributed stochastic neighbor embedding
PCA	- Principal Components Analysis
KNN	- k-nearest neighbours
PCP	- Parallel Coordinates Plot
EDA	- Exploratory Data Analysis
CBT	- Cognitive Behavioural Therapy
PROSIT	- Predicting Risks and Outcomes of Social InTeractions
GPS	- Global Positioning System

Acknowledgements

First, I thank Dr. Fernando Paulovich for supervising my master's research. Being your student offered me exceptional opportunities to learn and grow. Your support and warm, friendly presence, especially during the difficulties of a pandemic, made this journey much easier.

I thank Dr. Sandra Meier for providing me with this incredible opportunity to work with the PROSIT dataset. Your mentorship was essential in building an idea that eventually shaped this thesis. Your commitment to PROSIT study is inspiring in many ways. It is something I wish to emulate in my career ahead.

I thank Davi (postdoc), Chandramouli (PhD) and Leonardo (PhD) for your valuable suggestions and help to improve my coding and writing.

I thank Jeni for your sincere friendship. This journey would not have been pleasant without my brother Shoeb, my sister Sheema and a few of my friends: Hemanth, Nandish, Naveen, Adithya, Jhanvi, Josvin, Megha, Prishi, Souvik, Aman, Harpreet, Rajveen, Ruminder, Micaela, Disha, Evie, Aishwarya Kanchi.

Finally, I thank Dr. Fernando Paulovich, Dr. Sageev Oore, Dr. Qiang Ye, Dr. Stephen Brooks, Jennifer LaPlante, and Dr. Chris Whidden. Your courses and/or mentorship laid the foundations that enabled me to work on this thesis.

Chapter 1

Introduction

Anxiety, depression and substance abuse are youth's most common behavioural disorders. A third of men and more than half of women had an episode of prominent depressive and anxiety symptoms at least once during mid-to-late adolescence [76]. Anxiety is a common mental health problem, which typically onset at an early age and follows a chronic course [23]. Many patients experience a relapse or chronic course of anxiety, often resulting in substantial impairment across their lifespan [41, 42, 36, 28]. A steep increase in the prevalence of behavioural disorders including anxiety has been observed over the last three decades, and this is likely to continue in the future [24].

Cognitive behavioural therapy (CBT) is the first-line treatment option for youth anxiety-related disorders. CBT involves psycho-education about anxiety, teaching youth skills for managing fears (e.g., relaxation, cognitive restructuring, problem-solving), and provides a context for youth to encounter their fears gradually and minimize avoidance (e.g., exposure) [59]. The efficacy of CBT for youth anxiety has been demonstrated in several randomized control trials indicating large pre to post-treatment effects and demonstrating superiority to control conditions [108]. While CBT is just one of the many effective treatment options for anxiety disorders [26], most youths with anxiety disorders do not receive adequate mental health care [24, 40].

Notably, the delivery of treatment for anxiety disorders will change considerably over the next few years due to the widespread availability of the internet and smartphone applications and their utility in delivering CBT-based psychological interventions. This change will ease many barriers that stand in the way of youth seeking or receiving treatment under the current health care standards [57].

While mental health apps cannot replace professional clinical services, meta-analyses highlights the potential of mental health apps to serve as cost-effective, easily accessible, and low-intensity interventions for those who cannot receive standard psychological treatment [68, 63]. Mobile sensing technology can be ubiquitous

and offer unprecedented opportunities to gain valuable insights into areas where youth might experience problems with daily life behaviour. Today's smartphones have high-quality built-in sensors like GPS, Bluetooth, accelerometer, gyroscope, and ambient light/noise sensors. Analysts can use logs from these sensors combined with calling, app usage, and screen usage events to derive critical information about youth's social interactions, physical mobility and sleep routines.

1.1 Research Questions

In this thesis, we explore the possibility of combining feature engineering and established data visualization and machine learning techniques to visually identify individual behavioural patterns and compare the same with the remainder of the group or subgroups having participants with matching behavioural patterns. We attempted to answer the following research questions using a visual system.

1. **(RQ1):** Is this dataset collected as part of the PROSIT study adequate to identify distinct participant groups based on similarities in smartphone usage behaviour?

2. **(RQ2):** Is it possible to visualize this smartphone sensor-based dataset to identify behavioural trends or to preprocess it to extract meaningful features characterising a participant's behaviour and daily habits?

3. **(RQ3):** Can the abovementioned features be sub-selected and examined to better understand how different combinations aid in the identification of similar groupings or cluster formations?

This project is an exploratory study. It does not pursue the validation of an argument; instead, it aims to bring together diverse established techniques found in the existing literature to research the potential of mobile sensing data to identify behavioural disorders.

1.2 Contributions

In brief, the contributions of this thesis to explore research questions described in section 1.1 are listed below:

1. **(C1):** A data preprocessing pipeline and a feature engineering approach to extract meaningful information from raw mobile sensing data. With this, we investigated methods to extract features representing screen usage, calling habits and sleep routine that can help us answer RQ1 and RQ2.

2. **(C2):** An interactive visual system with glyph-based scatter plot to explore similar user groups or identify outliers. Further, allow comparison of aggregated features of individual participants with aggregated features of other users in the same or different groups. This contribution attempts to answer RQ3.

3. **(C3):** An interactive radial chart-inspired visualization with a brush filtering component to filter the days during the study period. This work attempts to address RQ2 by combining smartphone data (accelerometer, gyroscope, screen brightness, sleep noise, and screen lock state) to help visually identify participants' sleep and activity patterns.

1.3 Thesis Outline

The remainder of this thesis is organized as follows: Chapter 2 briefly introduces feature engineering process, dimensionality reduction and classification. These three strategies are used together in this project to identify subgroups or clusters visually. Chapter 3 presents the literature review of some of the research work previously done in mobile sensing for mental health, visualization of multidimensional data and visualization of time series data. In Chapter 4, we describe the data collection process used in the PROSIT project and briefly present the data-set's structure. Chapter 5 discusses the methodology and approach used to pre-process the raw data and explains the reasoning behind the design goals used for developing and implementing our visual system. Chapter 6 describes three use cases that represent the usefulness of our visual system, and also demonstrates our visual system's effectiveness in exploring and analyzing any given dataset. Here, we demonstrate two usage scenarios, first using the IRIS dataset and then the PROSIT dataset. Chapter 7 concludes our thesis work by briefly highlighting limitations and future directions.

Chapter 2

Background and terminology

This chapter presents the background and the relevant concepts for this work.

2.1 Feature Engineering

A feature is a numeric representation of the characteristics of the raw data. Features sit between data and models in the machine learning pipeline, and the process of formulating the most appropriate features given the data or the model is named feature engineering. ML models, such as decision trees, random forests, neural networks, and gradient boosting machines take feature vectors as inputs and make a prediction. These models learn in a supervised manner, with feature vectors mapped to the predicted output. The performance of such machine learning methods heavily depends on the choice of data representation (or features) to which they are applied. For that reason, much of the effort in deploying machine learning algorithms go into the design of preprocessing pipelines and data transformations that result in a representation of the data that can support effective machine learning[107, 19]. It is a crucial step that enables higher quality output results. Fig. 2.1 shows an example end-to-end data modelling pipeline, and the place features are introduced in the same.

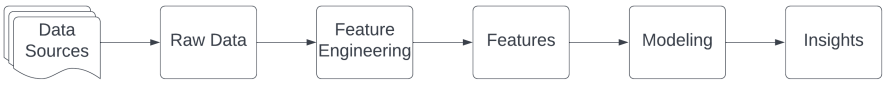


Figure 2.1: Feature engineering in a typical end-to-end data modelling pipeline.

2.2 Dimensionality Reduction

Real-world data often has high dimensions, meaning it has many features or attributes representing each data instance. In order to handle such data adequately, its dimensionality must be reduced. Dimensionality Reduction(DR) refers to the problem of

mapping high dimensional data points into meaningful representations of reduced dimensions [34, 73]. Ideally, the reduced representations should have a dimensionality corresponding to the data's intrinsic dimensionality. The intrinsic dimensionality of data is the minimum number of parameters needed to account for the observed properties of the data [39]. DR is important in many domains since it helps tackle the sparsity of raw data, which is a consequence of the curse of dimensionality. Traditionally, DR was performed using linear techniques such as Principal Components Analysis(PCA) [77], factor analysis [88], classical scaling [92]. These techniques keep the low-dimensional representations of dissimilar data points far from each other. For high-dimensional data that lie on or near a low-dimensional, nonlinear manifold, keeping the low-dimensional representations of very similar data points close together is usually more important, which is typically not possible with a linear mapping. Therefore, these linear techniques do not perform adequately for a nonlinear data set.

In the last few years, large variety of diverse alternative non linear DR techniques have emerged, including popular algorithms such as Locally Linear Embedding (LLE), Isomap, Isotop, Maximum Variance Unfolding (MVU), Laplacian Eigenmaps, Neighborhood Retrieval Visualizer, Maximum Entropy Unfolding, t-SNE, and many others [82, 90, 106, 18, 95, 97], see e.g. [96, 97, 64, 22] for overviews. t-SNE which is a modified method based on SNE technique has performed well in unsupervised experiments. t-SNE is capable of capturing much of the local structure of the high-dimensional data very well, while also revealing global structure such as the presence of clusters at several scales [95]. SNE starts by converting the high-dimensional Euclidean distances between data points into conditional probabilities that represent similarities. The similarity of datapoint x_j to datapoint x_i is the conditional probability, $p_{j|i}$, that x_i would pick x_j as its neighbour if neighbours were picked in proportion to their probability density under a Gaussian centred at x_i. For nearby datapoints, $p_{j|i}$ is relatively high, whereas for widely separated data points, $p_{j|i}$ will be almost infinitesimal (for reasonable values of the variance of the Gaussian, σ_i). Mathematically, the conditional probability $p_{j|i}$ is given by

$$p_{j|i} = \frac{\exp(-||x_i - x_j||^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-||x_i - x_k||^2/2\sigma_i^2)}, \tag{2.1}$$

where σ_i is the variance of the Gaussian that is centered on datapoint x_i. Because we are only interested in modeling pairwise similarities, we set the value of $p_{i|i}$ to zero. For the low-dimensional counter parts y_i and y_j of the high-dimensional datapoints x_i and x_j, it is possible to compute a similar conditional probability, which we denote by $q_{j|i}$. We set the variance of the Gaussian that is employed in the computation of the conditional probabilities $q_{j|i}$ to $\frac{1}{\sqrt{2}}$. Hence, we model the similarity of map point y_j to map point y_i by

$$q_{j|i} = \frac{\exp(-||y_i - y_j||^2)}{\sum_{k \neq i} \exp(-||y_i - y_k||^2)}. \tag{2.2}$$

Since we are only interested in modeling pairwise similarities, we set $qi|i = 0$.

If the map points y_i and y_j correctly model the similarity between the high-dimensional data points x_i and x_j, the conditional probabilities $p_{j|i}$ and $q_{j|i}$ will be equal. To measure the faithfulness with which $q_{j|i}$ models $p_{j|i}$ Kullback-Leibler divergence (which in this case equal to the cross-entropy up to an additive constant) is used. SNE minimizes the sum of Kullback-Leibler divergences over all datapoints using gradient descent method. The cost function C is given by

$$C = \sum_i KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}, \tag{2.3}$$

in which P_i represents the conditional probability distribution over all other datapoints given datapoint x_i, and Q_i represents the conditional probability distribution over all other map points given map point y_i. Because the Kullback-Leibler divergence is not symmetric, different types of error in the pairwise distances in the low-dimensional map are not weighted equally. In particular, there is a large cost for using widely separated map points to represent nearby datapoints (i.e., for using a small $q_{j|i}$ to model a large $p_{j|i}$, but there is only a small cost for using nearby map points to represent widely separated datapoints. This small cost comes from wasting some of the probability mass in the relevant Q distributions. In other words, the SNE cost function focuses on retaining the local structure of the data in the map (for reasonable values of the variance of the Gaussian in the high-dimensional space, σ_i).

Although SNE constructs reasonably good visualizations, it is hampered by a cost function that is difficult to optimize and by a problem we refer to as the "crowding

problem.". The cost function used by t-SNE differs from the one used by SNE in two ways: (1) it uses a symmetrized version of the SNE cost function with simpler gradients that were briefly introduced by Cook et al. [25] and (2) it uses a Student-t distribution rather than a Gaussian to compute the similarity between two points in the low-dimensional space. t-SNE employs a heavy-tailed distribution in the low-dimensional space to alleviate both the crowding problem and the optimization problems of SNE.

In this thesis, we have used t-SNE technique to compute projections of user selected features belonging to participants, and these projections are used for generating the chart.

2.3 Classification

Machine learning is a subfield of artificial intelligence that focuses on using data and algorithms to imitate learning whose accuracy improves with experience. The study of ML is essential for addressing fundamental scientific and engineering questions and for the efficient software systems it has produced across a broad range of domains. ML has progressed rapidly over the past two decades and is now a practical technology in widespread commercial use. A diverse array of machine-learning algorithms has been developed to cover the wide variety of data and problem types exhibited across different machine-learning problems [71, 45]. Conceptually, machine-learning algorithms can be viewed as searching through a vast space of candidate programs, guided by training experience, to find a program that optimizes the performance metric [55].

Instances in a dataset used by machine learning algorithms are represented using multiple features which may be continuous, categorical or binary. If the instances are given with known labels (the corresponding outputs), then the learning is called supervised; otherwise, unsupervised learning is where instances are unlabelled. The most widely used machine-learning methods are supervised learning methods [45]. Here, the training data forms a collection of (x, y) pairs, and the goal is to produce a prediction y^* in response to a query x^*. The inputs x may be classical vectors or more complex objects such as documents, images, DNA sequences, or graphs. Similarly, many different kinds of output y have been studied. Much progress has been made by focusing on the simple binary classification problem in which y takes one of two

values.

One example of such techniques, the k-nearest neighbours algorithm, often known as KNN or k-NN, is a non-parametric, supervised learning classifier that makes classifications or predictions about grouping individual data points based on closeness. While it can be used for both regression and classification problems, it is most commonly utilized as a classification technique based on the notion that comparable points can be discovered together. KNN classification has two stages, the first stage is to determine the nearest neighbours, and the second stage is to determine the class based on majority vote of classes of those neighbours [27, 29]. It is also worth mentioning that the KNN method belongs to the "lazy learning" family of models, which means it just saves a training dataset rather than going through a training step. All calculation takes place when a classification or prediction is produced. It is also known as an instance-based or memory-based learning approach because it significantly relies on memory to retain all of its training data. The distance between the query point and the other data points must be calculated to determine which data points are closest to the query point. These distance measures aid in forming decision borders that divide query points into distinct regions. Any of the several distance metrics like Euclidean, Manhattan, Minkowski or Hamming distance can be chosen to compute these distances

In this thesis, feature engineering is used to extract screen usage call-related and sleep-related features from raw sensor data, and the t-SNE or PCA techniques are used to compute the 2D projections. Later KNN algorithm is applied over the feature space to predict a class label of every data point. This information is visually embedded in the form of a glyph to allow data exploration. Chapter 5 further elaborates this in detail.

Chapter 3

Literature Review

This chapter outlines the literature that was reviewed during this thesis. We mainly focus on mental health issues such as anxiety and depression since it is commonly recognized that these two are strongly linked to the use of smartphones and social media, as well as inactivity [32]. We centred our review on earlier work that leveraged mobile sensing data to detect symptoms of anxiety or depression, as well as work on visualizing time-series multivariate data to find similar groups and patterns.

In the first section, we summarize work analyzing mobile sensing data to identify mental health issues like anxiety and depression. In the second section, we review work related to multidimensional data visualization, including techniques like dimensionality reduction and classification for multivariate data. The last section reviews popular methods to visualize time series data.

3.1 Mobile sensing data for Mental health

Smartphones are becoming increasingly popular among adults of all backgrounds around the world, and they are carried with them the majority of the time as one of their most important personal belongings. Smartphone subscriptions have surpassed six billion worldwide and are expected to increase by several hundred million soon [8]. As a result, a significant amount of research has been focused on gathering data (e.g., app usage logs, media and internet consumption, communication logs, screen activity, location and human activity detection) from these mobile devices to study user behaviour for targeted profiling. The goal of such targeting could be for business reasons like ads and digital marketing to increase brand awareness, but a similar approach has been used to study and understand user behaviour that correlates to mental well-being. Depression, for example, is associated with several behavioural changes like reduction in physical activity and changes in sleep routine, some of which can be detectable using mobile phone sensors [78, 94]. Studies have indicated that data

from phone sensors effectively analyzed the relationships between social interactions, screen usage and sleep with depression [30, 79] and such features significantly correlated with its severity [102]. Some research also describes excessive smartphone use as compulsive behaviour and has linked it to a few depressive symptoms [91, 65].

Physical activity has been shown to help prevent a variety of diseases, as well as improve mental health and general quality of life. Three out of every four teenagers and one out of every four adults do not currently fulfill WHO's worldwide physical activity recommendations [13]. Numerous studies have found links between physical activity and improvements in mental health symptoms in various populations [75]. Most smartphones today come in many high-quality sensors, such as an accelerometer, gyroscope, proximity sensors, and magnetometer, that can be used to measure the extent of physical activity.

Several studies have used data from smartphone usage to predict personality traits. Given that our thesis focused on identifying mental health issues, these related studies can be used to learn how to extract essential features from raw mobile sensing data and apply artificial intelligence approaches. A study by Stachl et al., worked extensively in this area of personality prediction [89]. This study recruited 743 volunteers and filtered out participants with less than 15 days of logging data, no app usage, and missing questionnaire data. The final sample (n=624) was used for analysis. The dataset consisted of logs of events which included calls, contact entries, texting, global positioning system (GPS) locations, app starts/installations, screen de/activations, flight mode de/activations, Bluetooth connections, booting events, played music, battery charging status, photo and video events, and connections to wireless networks (WiFi). The character length of text messages and technical device characteristics were also collected. Researchers extracted 1,821 behavioural predictors from this raw dataset in domains of 1) communication and social behaviour, 2) music consumption, 3) app usage, 4) mobility, 5) overall phone activity, and 6) day and night-time activity. The final dataset also consisted of 35 personality dimensions (five domains and 30 facets) assessed for each participant during the study. These extracted variables mainly included standard estimators (e.g., arithmetic mean, standard deviation). More complex variables containing information about the irregularity, the entropy, the similarity, and the temporal correlation of behaviours were

also computed (e.g. mobility data). Later, these variables were used to train and validate k-fold linear and non-linear regression models. Results showed that these models could successfully predict Big Five personality trait levels for more than half of the domains and facets.

Another study by Saeb et al., explored the detection of daily-life behavioural markers using mobile phone sensor data and its usage to identify the severity of depressive symptoms [83]. In this study, 40 adults were recruited for two weeks and were instructed to carry a mobile phone with a data acquisition app. An online assessment consisting of Patient Health Questionnaire-9 (PHQ-9), a commonly used measure for self-reported depressive symptom severity [61] was used at the beginning of the study period to label the severity of depressive symptoms in individual participants. Though this study mainly focused on features extracted from GPS points and their correlation to depression, there were also features indicating phone usage duration and frequency. The results of this study demonstrated that the severity of depressive symptoms has a moderate to strong negative association (r=-.63, P=.005) with GPS data properties, as well as a moderate positive correlation (r=.54, P=.011) with phone usage.

A study by Huckins et al., focused on college student's mental health and behaviour during the early phases of the Covid-19 pandemic [48]. Researchers here used mobile sensing data and self-reported mental health labels to study the changes in behaviour and mental health of students associated with the restrictions imposed due to the pandemic. Mobile sensing data was collected using a smartphone app from 217 undergrad students 18 to 22 years of age. These students had been a part of longitudinal research for the previous two years. Sensor logs, including GPS, accelerometer, and screen lock events, were collected during the Winter 2020 semester. This data was used to extract features representing the day-to-day and week-to-week impact of workload on stress, sleep activity, mood, sociability, mental well-being and academic performance of the students. Short surveys were conducted weekly in the form of Ecological Momentary Assessments(EMA) [85] of the Patient Health Questionnaire-4(PHQ-4), which is a brief measure of depressive and anxious symptoms [62]. The extracted features included Sedentary time, Sleep(i.e. sleep onset, wake time and sleep duration), location(i.e. distance travelled, number of locations

visited) and phone usage(i.e. screen duration). Results from the study show that individuals in the winter of 2020 were more sedentary, which was expected due to severe lockdown restrictions. Participants also reported being anxious and depressed relative to previous terms and subsequent breaks. The analysis showed various behavioural changes, including increased screen time, decreased physical activity, and fewer locations visited. The study discovered that these changes in behaviour were related to COVID-19 news variations at the time and that they were associated with anxiety (P<.001) and sadness (P=.03).

The PROSIT[10] study's mobile sensing data is similar to some of the research described above, and we have leveraged their findings, which show a strong link between behavioural changes and mental health difficulties. We used a modified version of their methods to extract these behavioural indicators and develop a tool to examine changes in those indicators during the study period visually. Other programs like RADAR-Base and mindLAMP work in areas similar to PROSIT, where they collect high-resolution data at scale and analyze it with the help of data visualization to correlate it with mental well-being [80, 93].

3.2 Visualization and analysis of multidimensional data

Visual Data Exploration is usually broken down into three-step process: *Overview first, zoom and filter, and then details-on-demand* [86]. Visualization techniques help provide a high-level overview and allow the user to discover relevant subgroups in the data. It is critical to preserve the overall picture while focusing on the subset with another visualization technique. An overview like this is necessary to represent the cluster patterns or groups that naturally arise in data.

Traditional methods such as scatterplots and scatterplot matrices can produce a view of a dataset's inherent structure. They remain one of the most popular and widely-used visual representations for multi-dimensional data due to their simplicity, familiarity and visual clarity. Work by Elmqvist et al., utilizes scatterplots for visual exploration of multi-dimensional datasets using structured navigation in data dimension space [33]. Fig. 3.1 shows the example of a scatterplot matrix of a car dataset having 8-dimensions. Techniques like GPLOM [49] are improvised versions of scatterplot matrices combining scatterplots for pairs of continuous variables, heatmaps

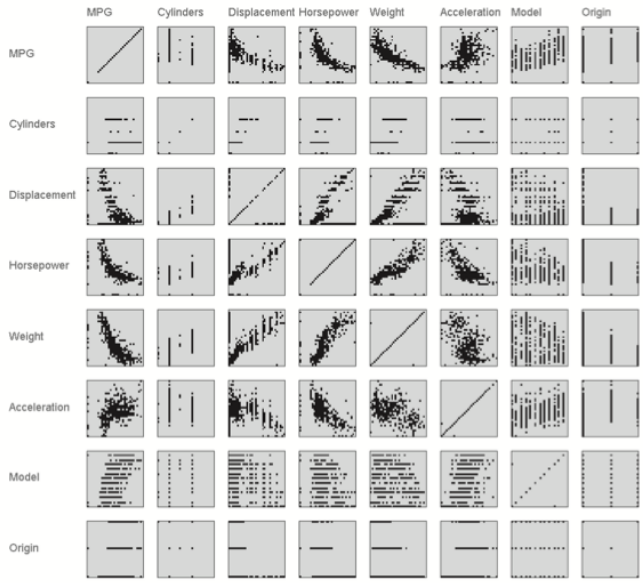


Figure 3.1: Example of a scatterplot matrix for a 8-dimensional car dataset. (Source: [33])

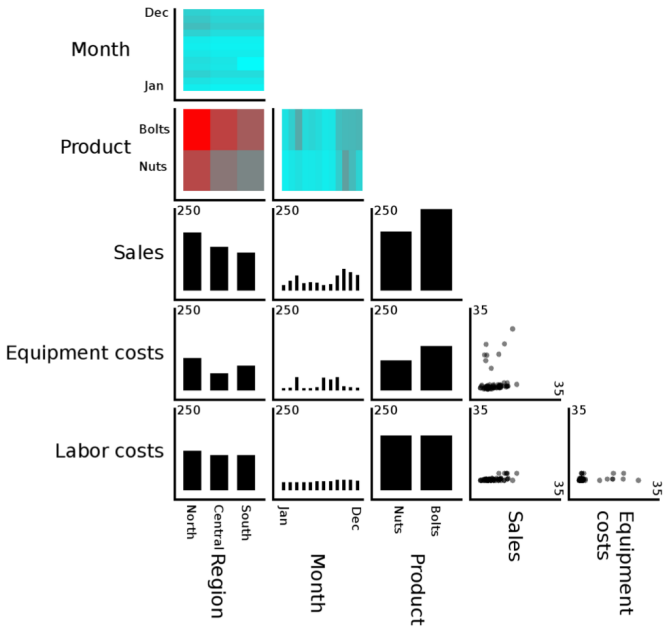


Figure 3.2: GPLOM visualization, which is a modified version of scatterplot matrix of a sample dataset. Barcharts and heatmaps show aggregated data. Heatmaps are also color coded to represent aggregated sales numbers. (Source: Im et al.[49])

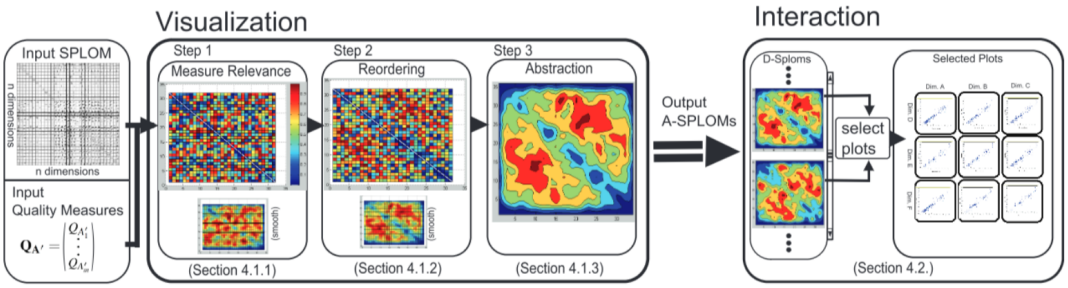


Figure 3.3: Schematic overview of a visual technique to scale scatterplot matrices (Source: Lehmann et al.[66])

for categorical variables, and bar charts for pairings of a categorical and continuous variable. This method is well suited for data with multiple categorical variables. Fig. 3.2 shows a GPLOM visualization of a sample dataset with 5-dimensions. However, the scatterplot matrix technique is not scalable and is ineffective when the size and dimensionality of data increase [70, 15]. The visualization technique proposed by Lehmann et al. [66] addresses the scalability problem of scatterplot matrix. They presented an interactive framework and tested the visual representations with a real dataset having more than 100 dimensions. Fig. 3.3 shows a schematic overview of the proposed visual technique. However, such visual representations can be complex for analysts to understand and navigate without guidance.

Many techniques have been devised to project high-dimensional data into 2D or 3D dimensional space. As the dimensionality increases, various techniques like *dimension subsetting*, *dimension reduction*, *dimension embedding*, *multiple displays* are used to handle the curse of dimensionality problem [104]. A more popular approach is to apply Machine Learning(ML) techniques to generalize the complexity of high dimensional data by employing Dimensionality Reduction(DR), density estimation and clustering/classification methods. Here, the structure of a dataset can be defined as the "geometric relationships among subsets of the data vectors in the L-space" [84], where vectors regard instances or data points, and L their dimensionality. Section 2.2 has a more detailed explanation of DR. A comparative study by Ventocilla et al., presents an empirical user study that compares eight multidimensional projection techniques for supporting the estimation of the number of clusters, k, embedded in six multidimensional data sets [99]. Results of this study suggested that t-SNE will likely

lead to estimates closer to the number of labels in a data set when compared to other Multidimensional Projections (MDPs). Point plots are visualizations that project data instances from an n-dimensional space to an arbitrary k-dimensional space, such that data records map to k-dimensional points. A graphical representation or a mark is drawn at the associated k-dimensional point for each data point. The objective of such graphical representation, also known as 'glyph,' is to represent multivariate data so that an investigator can quickly comprehend relevant information to apply suitable analysis. Fig. 3.4 shows a few examples of glyphs that have been previously used as markers. Glyphs have become a popular method for conveying information visually. Individual dimensions for each data point are mapped to attributes of a particular shape or symbol, and variations and anomalies among these graphical entities can easily be perceived. A taxonomy of glyphs and placement strategies by Matthew O Ward [103] presents an overview of multivariate glyphs, a list of issues regarding the layout of glyphs, and a comprehensive taxonomy of placement strategies to assist the visual design process.

Once the positions for the glyphs are computed, a possible post-processing step for data-driven techniques would involve adjusting these initial positions of glyphs to reduce clutter and overlap. This is required because overlap can distort the final image's interpretability. In literature, many solutions can be found to address this problem with occlusion due to the overlap of glyphs. A technique by Hilasaca et al., called Distance Grid (DGrid) proposed a novel approach to removing overlapping DR projections by combining a density-based strategy to generate auxiliary points with a novel space-partitioning method [46]. In this thesis, we used the DGrid technique to remove the occlusion caused due to overlap of glyphs. Fig. 3.5 shows this overlap removal process using Dgrid on a sample scatterplot. Python implementation of this algorithm can be found here.[1]

An example of another popular technique which allows the visualization of multi-dimensional data is the Parallel Coordinates or PCP (Parallel Coordinates Plot). This technique induces a non-projective mapping between N-Dimensional and 2-Dimensional sets [53]. A mathematician and computer scientist named Alfred Inselberg popularized this technique in 1985 for studying high-dimensional geometry

[1] `https://github.com/fpaulovich/dimensionality-reduction`

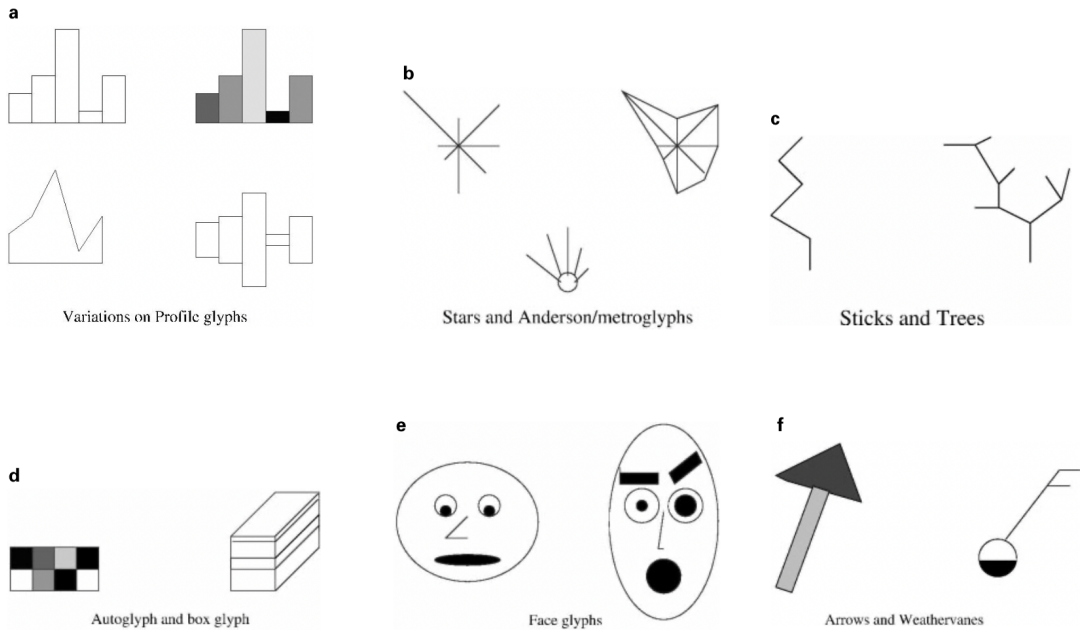


Figure 3.4: Examples of glyphs. Top row: (a) variations on profiles; (b) stars/metroglyphs; and (c) stick figures and trees. Bottom row: (d) autoglyphs and boxes; (e) faces; and (f) arrows and weathervanes (Source: [103])

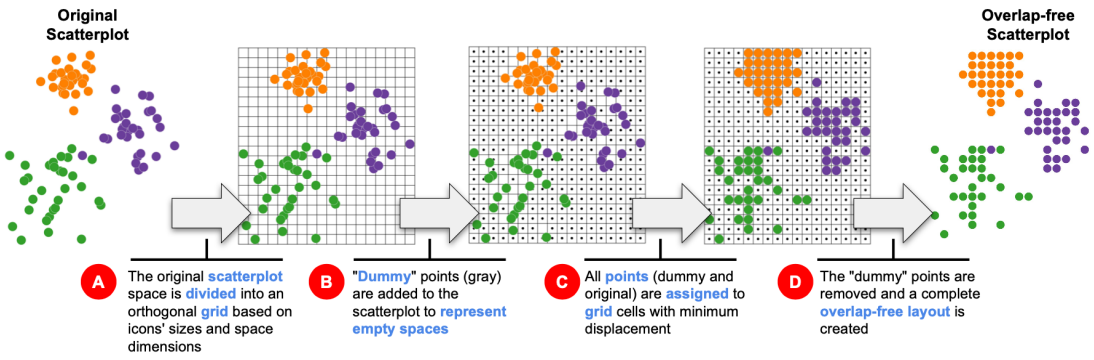
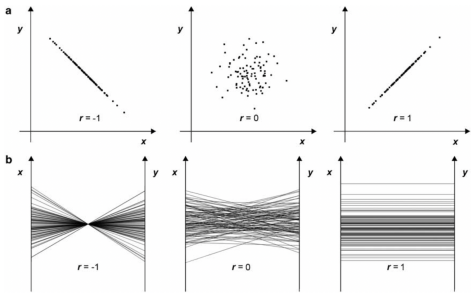


Figure 3.5: Overview of overlap removal process using *Distance Grid(DGrid)*[46]. The scatterplot area is first split into a grid (A), and "dummy" points (small black dots) are crafted to represent empty space (B). Finally, original and "dummy" points are assigned to grid cells (C), and the "dummy" points are removed (D), resulting in a completely overlap-free layout. (Source: [46])

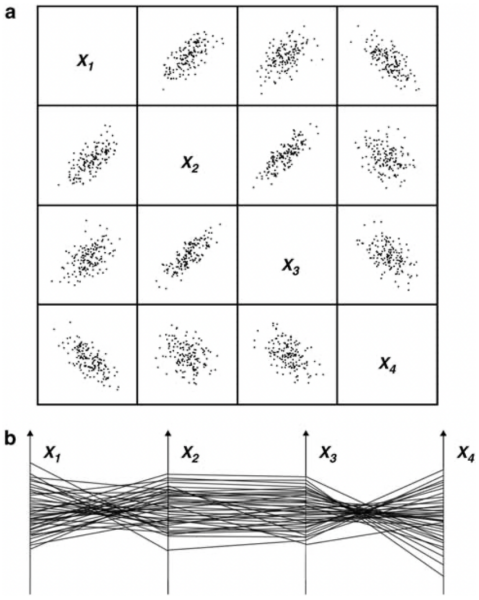
[51]. Parallel Coordinates display each multidimensional data item as a polygonal line which intersects the horizontal dimension axes at the position corresponding to the data value for the corresponding dimension [58]. To interpret the plot, users can look for clusters of similar lines (indicating the partial correlation between pairs of dimensions) and lines that are either isolated or have a slope that is significantly different from their neighbours (indicating outliers). Scatter plots and PCP can be used to assess the dataset's correlation visually. A user study by Li et al., compared these two visualizations and assessed their perception of correlation [67]. Although the study concluded that scatter plots are more practical for supporting visual correlation analysis, the ability of PCP to scale for the higher number of dimensions cannot be ignored. Fig. 3.6 shows the visual comparison of scatter plot matrix and PCP to perceive correlation in multidimensional data. It can be clearly seen that as the number of dimensions increases scatterplot matrix technique requires more visual space to display matrix combinations between individual dimensions, thus making this techniques less useful in cases where the number of dimensions is large.

Another relatively older study by Brunsdon et al. [20] examined PCP along with three other techniques for multi-dimensional data visualization: projection pursuit [38], Geographically Weighted Regression [21] and RADVIZ [47]. This study suggested that the PCP approach was the most intuitive of the four techniques and even commented that PCP was essentially a multidimensional variation of the scatterplot. Instead of just two axes, as in the case of a scatterplot, PCP has one axis for every dimension and can be used to draw relationships between those axes, which are depicted as parallel lines. The ordering of the axes, on the other hand, influences the depiction of relationships within the dataset; therefore, attention must be made when choosing one. The depiction of the data in parallel coordinates can get rather messy when large numbers of cases are involved. Chapter 10 of Inselberg's book provides a great discussion on exploiting interactivity in PCPs to understand large and complex data [50].

A study by Krause et al., presented a framework called *SeekAView* that allowed the analyst to build subspaces to analyze the structure of datasets having around 100 dimensions visually [60]. Fig. 3.7 shows different panels presented in SeekAView. This technique uses a combination of PCA projected scatter plot, parallel coordinates,



(a) Representation of sample data with three extreme values of r measuring correlation using (i)scatterplots, and (ii)parallel coordinate plots (PCP)



(b) Visualization of multivariate data: (i)scatterplot matrix and (ii)parallel cordinate plot.

Figure 3.6: Visual comparison representing the ability to display measure of correlation within multidimensional data(Source: [67])

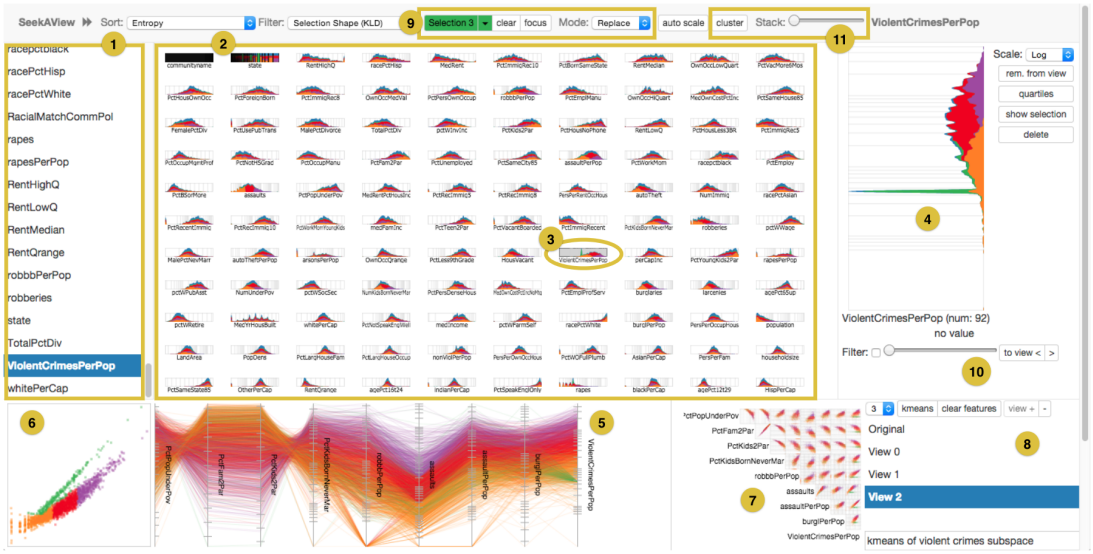


Figure 3.7: The user interface of *SeekAView*: framework to visually analyze the structure of multidimensional data (Source: [60])

and scatter plot matrix to analyze the structure and dependencies within the various dimensions in the dataset. Cluster labels from hierarchical clustering (DB-SCAN with single linkage) were coupled with visualizations to find groups of similar dimensions. Users also had the flexibility to brush filter various visualization plots. These brush filtering enabled users to specific targets that can be used to suggest different views. Brushes could also remove outliers or focus attention on a specific class.

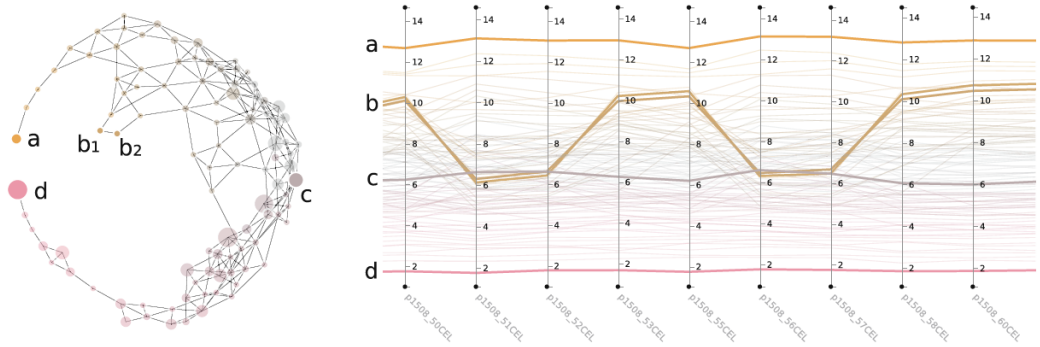
Another study by Ventocilla and Riveiro proposed an advanced visualization to explore the structure of large datasets [98]. This study also uses the strategy to analyze the data by building a visual overview of data in multi-dimensional space. Such an overview is built on a framework using Growing Neural Gas (GNG), and visual encodings with force-directed graphs (FDG). A progressive visualization approach was used here to study the structure of the dataset: a progressive algorithm generates early results based on the dataset and parameter settings and produces partial results that the user can analyze. This cycle repeats until the algorithm achieves a convergence. Fig. 3.8 shows an overview of this progressive evolution of GNG topology when applied to a sample medical dataset. User interactive features like hovering and clicking on nodes on the GNG plot highlighted the respective data line over the PC plot. Filtering was triggered by dragging up and down feature boundaries in the

Figure 3.8: Progressive evolution of GNG topology when applied to a sample medical dataset. From left to right represent the topology state after progression in the number of signal sampling. (Source: [98])

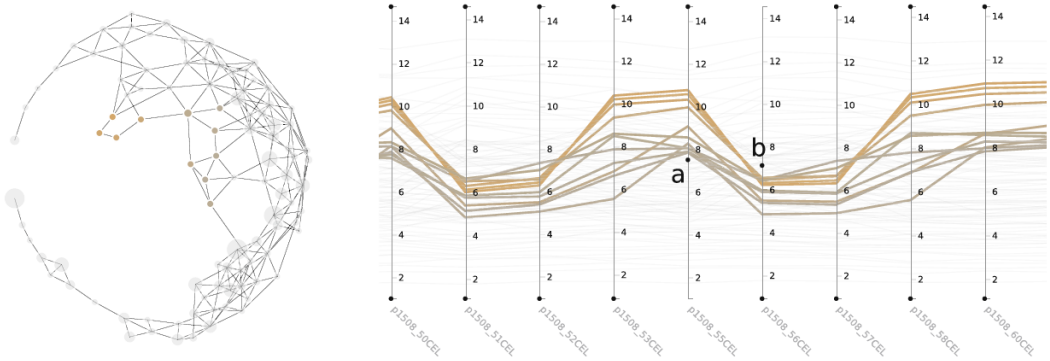
PC plot. Dragging feature boundaries caused prototype lines with values outside the boundaries and their corresponding nodes in the GNG to be demoted to a gray colour with lower opacity. K-means clustering method was also employed to cluster the data points and colour code the nodes in GNG and lines in PC plots to help analysts identify similar groups.

3.3 Visualization of time series data

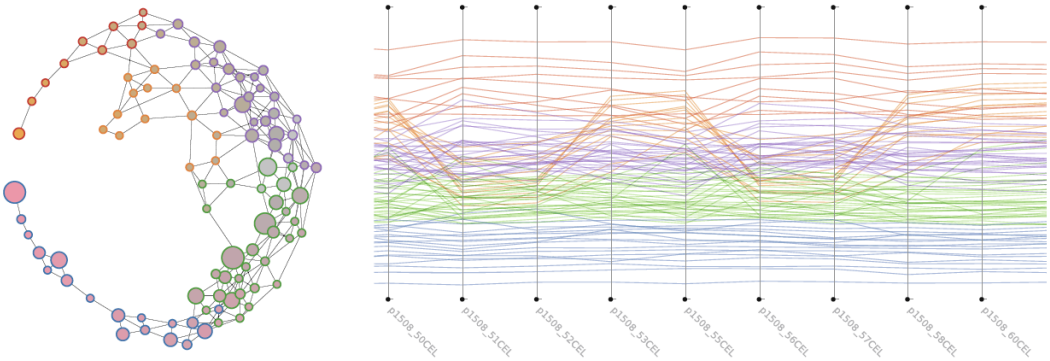
Time is an inherent data dimension that is central to the task of identifying trends and relationships within the data. It is a set of observations arranged in chronological order. Time and time-oriented data have distinct characteristics that make it worthwhile to treat such data as a separate data type [14, 16]. Analysis of time series data is quite diverse and can be seen in a wide variety of research involving the study of the behaviour of a subject over a specific time. There can be several examples like (1) studying the movement of a planet around its star, (2) predicting the net sales of a company in the next quarter based on previous trends, (3) recommending a product to a customer based on their previous purchase or (4) studying the progression of a chronic illness in a patient. Time series analysis has long been used in science and engineering and has contributed significantly to the most recent technological advancements (analog and digital communication, robotic control). Visualization plays



(a) Linked highlighting on hovering and selecting nodes in the FDG.



(b) Filtering on dragging filter boundaries (a and b) in PC plot.



(c) K-Means ($k = 5$) applied to the units' prototypes.

Figure 3.9: Views of interactions between the GNG and PC plots highlighting the reciprocity between overview and detailed view of multidimensional dataset. (Source: [98])

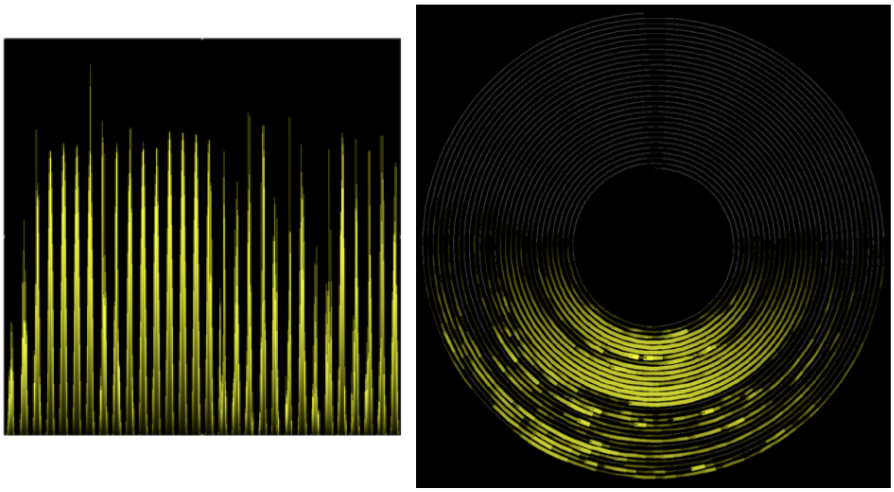


Figure 3.10: Linear vs Circular visualizations of intensity of sunlight measured over certain time. (Source: [105])

an essential role in such research involving time-series data. A variety of visual approaches have been used to explore time series data. Each of these techniques is significant in its own right. Each of these approaches has its significance [87, 35]. The choice of any of these visual representations depends on the characteristics of time series data and the overall goal of the visualization.

Classical point graph, line chart and bar chart have been proven to be very effective in visualizing serial data and is very popular even today to highlight trends and anomalies. Circular/spiral graphs, on the other hand, have been proven to help expose periodic behaviours over small subsets of time. We can demonstrate the advantage of circular visual representations in identifying patterns in time series data. Fig. 3.10 shows linear vs circular visual representations of the sunlight intensity measured over time. By comparing both visualizations, it can be seen that circular visual representation is much easier to compare data over individual days. A circular graph is also more effective in identifying important events like sunrise, sunset and cloudy periods during the daytime.

A visual framework by Graells and Jaimes[44] combined linear and spiral layouts to analyze time-series data visually. The time brushing feature was implemented as scroll bars to interact with the data and allow users to focus on a specific subset of time. Fig. 3.11 shows an overview of the prototype implementation. The linear part

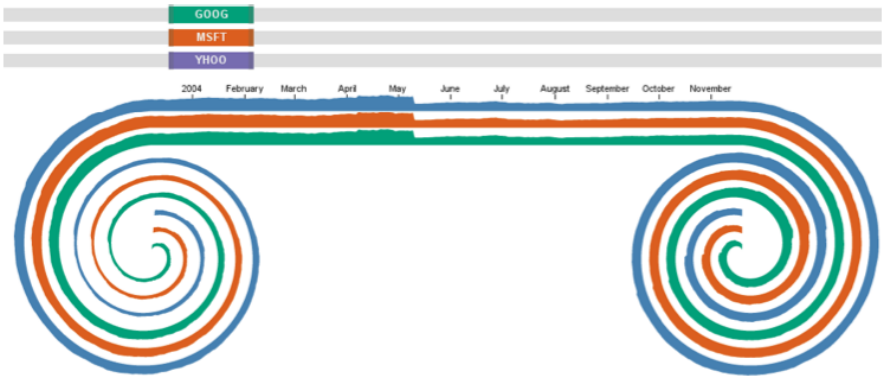


Figure 3.11: Prototype implementation of Lin-spiration. Visual technique that combines linear and spiral representations for time-series data analysis (Source: [44])

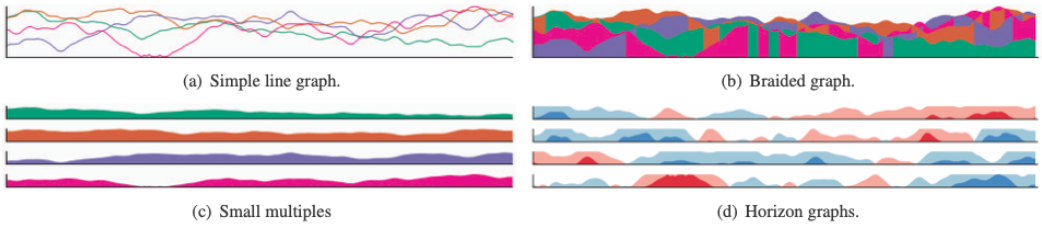


Figure 3.12: Overview of four techniques to visualize time series data. Example shows same time series data plotted using different representations to compare preception. (Source: [54])

shows the focused area of the data, whereas the spiral part shows the remainder of it. Scroll bars on the top allow users to interact with the visualization to change focus over time for each variable.

A study by Javed evaluated the visual perception of different time series data visualized together in a single chart [54]. Fig. 3.12 shows the four visual techniques that were evaluated for perception. Author divided the graphs into two categories: (i) shared space layouts which plot multiple time-series data over same region i.e. subfigures (a) and (b) in Fig. 3.12 and (ii) split space layouts which plots individual time-series data over sub-regions split within the complete available space i.e. subfigures (c) and (d) in Fig. 3.12. This study suggested that superimposed(shared space) techniques excel at comparisons within a local visual span. In comparison, juxtaposed(split space) techniques required the user to gaze vertically between different sub-spaces, making the comparison more difficult.

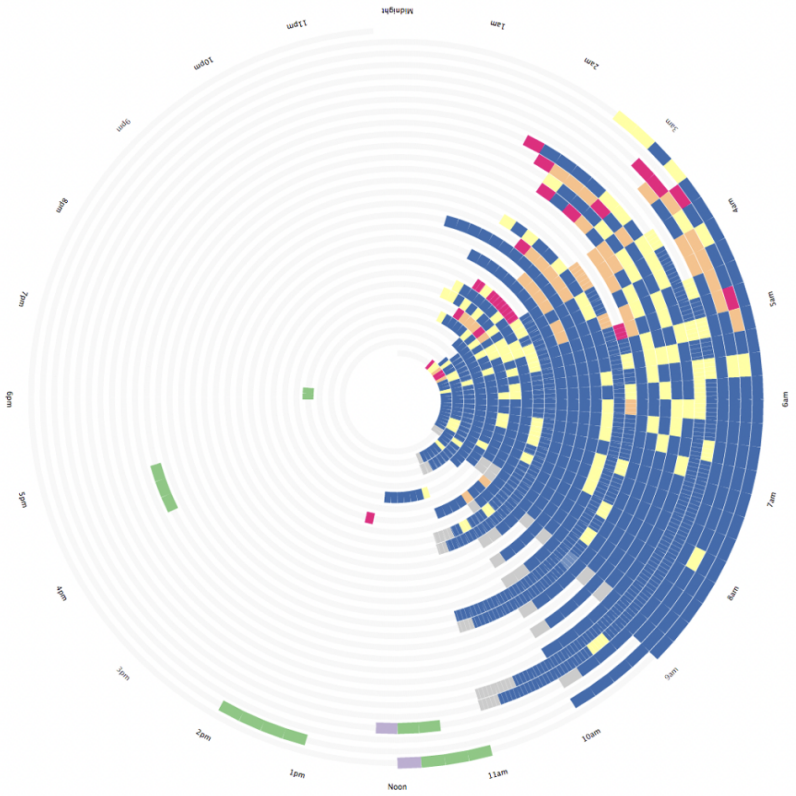


Figure 3.13: Circular visual design for effectively visualizing sleep data. (Source: [101])

A visual design study by Wallace et al., surveyed three sleep clinicians to validate the effectiveness of two different design types (linear vs circular) to plot self-tracked sleep data effectively [101]. Evaluation results of this study suggest that the spiral design is intuitive, reveals sleep patterns across long periods, and requires less effort than a standard line chart. Fig. 3.13 shows the final version of the visualization design that was voted best suited for sleep data visualization in a design user study.

For our system, we leveraged the data visualization design listed above and developed a circular clock-based visual chart that can be used to plot many sensor data layers on top of each other. The design goal here is to preserve the cyclical rhythmic nature of human behaviour and help users visually identify individual participants' sleep and activity patterns throughout the study period. Chapter 5, which outlines the design goals and methodology, will elaborate on this in more detail.

Chapter 4

Data

4.1 Data Collection

PROSIT is a research project conducted by Dalhousie University's Psychiatry Department [10]. This study aims to improve the mental well-being of youth. Researchers in this study sought a novel approach to providing clinical care to youth at risk of developing some of the most common mental health disorders or individuals who have a medical condition but have never been diagnosed or treated as a result of the current healthcare system. Given the potential of mobile sensing in providing unobtrusive logging of user activity that can be used to assess social behaviour, researchers from the PROSIT were keen on designing a mental health app that can serve as a cost-effective, easily accessible solution for those who cannot receive standard psychiatric treatment. An innovative mobile sensing app called PROSIT was developed to collect objective metadata data from the devices participants were using. Different versions of apps were designed to work on both Android and iOS versions since these were the two most popular mobile operating systems used by the general public. Data on social interactions were collected using this app for a period of a 6-week time window at the beginning of the study period. Participants were also asked to rate their social interactions during this time subjectively. Researchers were interested in measuring new onsets and trajectories of mental disorders in youth.

In many studies, researchers used the PROSIT Android and iOS apps to capture and store valuable data from participants' smartphones. Until this thesis was written, up to four studies had been conducted, each targeting specific user groups or periods. For instance, the *Covid study* focused on participants' mental health during the COVID-19 pandemic period and other studies like *Social media study* use focused on the impact of social media usage by adolescents on their mental well-being [69]. Although the structure and format of data collected from participants were the same, these studies mainly differed in the targeted participant groups and the set of

questionnaires that the participants had to answer during the study period. For this thesis, we worked only with data collected as part of the Covid study, representing the smartphone usage and behaviour patterns of participants during the Covid-19 pandemic. It should be noted that the system proposed as part of this thesis can replicate the processing and visualization of similar data collected in other studies using the same or a similar PROSIT app.

As part of the Covid study, we had data from participants using iOS and Android devices. We had 831 participants with iOS devices, and we used filtering criteria where only those participants who had data for more or equal to 14 days were considered for analysis. After filtering, we were left with 523 participants. Out of these 523, only 477 participants had labels about a mental health condition. As shown in the chapter5, these labels are essential in creating a custom glyph and colour-coding those glyphs in the visualization dashboard. This information is designed to help the dashboard user identify similar user groups and outliers, meaning to identify participants whose mental health label is different compared to their neighbours with similar smartphone usage patterns.

4.2 Data Structure

During the developmental phase of the PROSIT app, researchers looked for mobile sensor-based indices that may predict youth's behaviour in real life. These indices were selected based on their potential to make inferences about participants' mental health state [81, 43]. Table 4.1 and Table 4.2 lists few of the sensor data attributes that were captured by Android and iOS versions of the PROSIT app. Data consisted of 22 attributes for Android devices and 14 attributes for ios devices. Few of these attributes were not useful since they could not be linked to human behaviour. Hence we have not used all the captured data for our analysis and visualization. Fig. 4.1 show the distinct attributes representing measurements from various sensors and events collected from the devices used by the participants.

The PROSIT app is downloaded into the smartphone of a participant in the study. The app temporarily stores sensory data fetched from the device in a local SQLite database. This data is transmitted to a secure server over an HTTPS connection when the device is connected to WIFI. The app was carefully developed to take battery

ios

```
[
        "Accelerometer",
        "Analytics",
        "Brightness",
        "Call",
        "Device",
        "Gyroscope",
        "Location",
        "Lock state",
        "Magnetometer",
        "Power state",
        "Reachability",
        "Sleep_Noise",
        "Steps",
        "Survey",
        "Weather"
] _
```

android

```
[
        "accelerometer_m_s2__x_y_z",
        "bluetooth__bluetoothClass_bondState_deviceAdress_deviceName_id_type",
        "calls__callDate_callDurationS_callType_phoneNumberHash",
        "connectivity",
        "debug",
        "detectedActivityConfidence__inVehicle_onBicycle_onFoot_running_still_tilting_unknown_walking",
        "gyroscope_rad_s__x_y_z",
        "installedApps",
        "light_lux",
        "location__accuracyInM_altitudeAboveWGS84_bearingDeg_latitude_longitude_speedMperS",
        "magnetometer_muT__x_y_z",
        "notifications__action_flags_package",
        "powerState",
        "pressure_hPa",
        "proximity_cm",
        "rotationVector__cos_x_y_z",
        "sensorAccuracy__accuracy_sensor",
        "sms__numberLetters_phoneNumberHash_smsDate_smsType",
        "soundPressureLevel_dB",
        "stepCounter_sinceLastReboot",
        "systemInfo",
        "usageEvents"
] _
```

Figure 4.1: Distinct attributes representing sensor measurements for data collected from ios and android devices

Sl No	**Sensor**	**Description**	**capture in android**	**Data capture in ios**	**Usage in this thesis**
1	Accelerometer	Measure of acceleration in m/s^2 on three physical axes (x, y, z) including force of gravity.	yes	yes	yes
2	Gyroscope	Measure of angular velocity in rad/s around each of the three physical axes (x, y, z).	yes	yes	yes
3	Call	Timestamps of calling events i.e. call_incoming, call_dialling, call_connected, call_disconnected and call_hold.	yes	yes	yes
4	Lock state	Device screen locking states i.e. locked or unlocked.	yes	yes	yes
5	Light lux	Measure of ambient light levels around the device.	yes	no	yes
6	Brightness	Intensity of screen brightness.	no	yes	yes
7	Sleep Noise	Noise levels during sleep times detected by the device.	no	yes	yes
8	GPS	Current Location Coordinates of the device. This information encrypted for enhanced security.	yes	yes	no
9	Magnetometer	Measure of strength of Magnetic field around the device.	yes	yes	no
10	Connectivity	Device connectivity to internet.	yes	yes	no
11	Power state	Device charging and battery event i.e. charging, full, unplugged, PowerUnknown, power_connected, shutoff.	yes	yes	no

Table 4.1: List of sensor data which is captured by PROSIT app on android and ios devices

Sl No	**Sensor**	**Description**	**Data capture in android**	**Data capture in ios**	**Usage in this thesis**
12	Bluetooth	Bluetooth features used to periodically scan for nearby devices.	yes	no	no
13	Detected Activity	Workout or physical activity like walking, running, cycling etc. detected by the device	yes	no	no
14	installed apps	Time stamps of events when an app is installed or uninstalled in the device.	yes	no	no
15	Notifications	Type of notification received from an app	yes	no	no
16	Pressure	Information about pressure	yes	no	no
17	Proximity	Measure when user's face is close to the device.	yes	no	no
18	SMS	metadata captured from SMS sent or received by the device. i.e. hashed phone number, SMS data, length and type	yes	no	no
19	Sound Pressure Level	Background noise levels.	yes	on	no
20	Step Counter	Number of steps detected by the device.	yes	no	no
21	Weather	Weather information based on GPS coordinates.	no	yes	no

Table 4.2: List of sensor data which is captured by PROSIT app on android and ios devices continued...

usage into consideration. Continuously logging data at specific intervals would cause the app to consume a large amount of device battery, which would be undesirable for any user. Instead of periodic logging, data was logged only when there was a signification change in the last logged value. The threshold for this change was decided to be different for different sensors. For example, GPS coordinates were logged only when the locations changed in a range of more than 20 meters from the previously recorded point. On the server-side, all the data is stored in the MongoDB database. The flexibility of a no-SQL database like MongoDB is ideal in the case of PROSIT since not all the sensor data is recorded simultaneously. MongoDB allows data to be stored as JSON documents, and each sensory log is stored as an independent document. These documents representing a log of different sensor measurements of a device have a unique ObjectId assigned by the MongoDB database for indexing. Other than ObjectId, every document has five key-value pairs. A unique participantId is assigned to each participant before installing the PROSIT app on their device. All their data on the server has this unique participantId, which can be used to distinguish data from different participants. Apart from that, each document has a timestamp indicating the time when the log was measured and upload time indicating when the log was uploaded to the server. For all of the analyses, we only used measurement time, and the time here was in UTC, therefore we converted the time zone to Atlantic Standard Time (AST). The actual sensor data is stored under a key called "value," which has the measurement or the event log. Fig. 4.2 shows the structure of a document with single attribute value and Fig. 4.3 shows the structure of document with multiple values for a single attribute (ex: accelerometer or gyroscope data).

This raw data is further pre-processed to make it dashboard ready, the details of which are described in Chapter 5.

```
{
        "_id" : ObjectId("619713f32399ae3ae902b2c3"),
        "participantId" : "PROSITC00M",
        "attribute" : "light_lux",
        "value" : 141.73751,
        "measuredAt" : ISODate("2021-11-18T21:08:52.723Z"),
        "uploadedAt" : ISODate("2021-11-18T22:56:24.214Z")
}
```


Figure 4.2: the data structure of a document on a server database, i.e. MongoDB with a single value for each attribute


```
{
        "_id" : ObjectId("6198655e8ae7abaa50350954"),
        "participantId" : "PROSITC00M",
        "attribute" : "accelerometer_m_s2__x_y_z",
        "value" : [
                10.17668,
                -16.72702,
                -3.38165
        ],
        "measuredAt" : ISODate("2021-11-19T14:43:48.488Z"),
        "uploadedAt" : ISODate("2021-11-19T15:19:37.998Z")
}
```


Figure 4.3: The data structure of a document on a server database, i.e. MongoDB with multiple values for a single attribute

Chapter 5

Methodology

This chapter describes our data visualization system's design, preprocessing pipeline and implementation steps. We took an iterative approach to test and improve preprocessing and visualization techniques. Examining academic papers, journals, blogs, internet articles, and open source projects influenced our design choices and implementation directions. Due to its sensitivity, no part of the PROSIT dataset has been stored outside a secure server. Development and preprocessing of data was also done securely on this server. The complete code (without real data) developed during this thesis can be found here [1]. As an use-case of our visualization techniques, the same dashboard has been implemented to work with IRIS dataset, which can be found here[2].

5.1 Requirements

Domain experts from the psychiatry department helped us understand the data and identify the requirements expected from this data visualization system. We used to have regular weekly meetings to discuss the progress and develop ideas that could be implemented into a system. Due to Covid-19 restrictions imposed on everyone, these meetings were primarily online, with only a few occasions where in-person meetings were possible. The requirements resulting from brainstorming sessions with domain experts over these regular weekly meetings and discussions with supervisors, colleagues in the Visual Analytics and Visualization Lab [3] and friends eventually led to the ideas behind this work.

The research questions described in Section 1.1 translates to the technical requirements (**R1-R4**) as follows:

[1] `https://github.com/mohd-muzamil/flaskDashboard.git`
[2] `https://github.com/mohd-muzamil/IrisDashboard.git`
[3] `https://fpaulovich.wixsite.com/paulovich`

- **(R1):** The ability to manually explore feature subset selection to examine if it can help distinguish healthy participants from those diagnosed with a mental illness. We attempted to solve this task using **C1** and **C2** described in Section 1.2 under thesis contributions.
- **(R2):** The potential to use machine learning algorithms to group participants based on their mobile usage and behavioural patterns. **C2** described in Section 1.2 was the implementation of this requirement.
- **(R3):** The capability to select an individual participant and compare its features with a selected subgroup or remainder of the participant group. This requirement was implemented as part of **C2**, described in Section 1.2.
- **(R4):** The capability to visualize raw sensor data that allows recognition of behavioural patterns that might indicate physical activity or sleep. **C3** described in Section 1.2 attempts to implement this requirement.

5.2 Data Preprocessing

Visualization Dashboard is designed to identify individual participants' behavioural patterns and usage similarities w.r.t each other and other groups. Since the data on the MongoDB server is in NoSQL format, it had to be preprocessed in multiple steps to make it dashboard-ready.

Below is the list of data files that were available for analysis:

- CSV file containing demographic information and mental health labels of all the participants in PROSIT study. Attributes: ["participantId", "age", "gender", "device_type", "label1", "label2"]
- Access to MongoDB server, which hosts all the sensor data from the PROSIT study. This database has data from both Android and iOS devices. Data was stored in different MongoDB databases based on the study type or period when data was collected.

For file naming convention, the names of the raw data files were same as the name of the attribute which represented that data in mongoDB database. This naming convention helped us keep a track of data that exists within each data file.

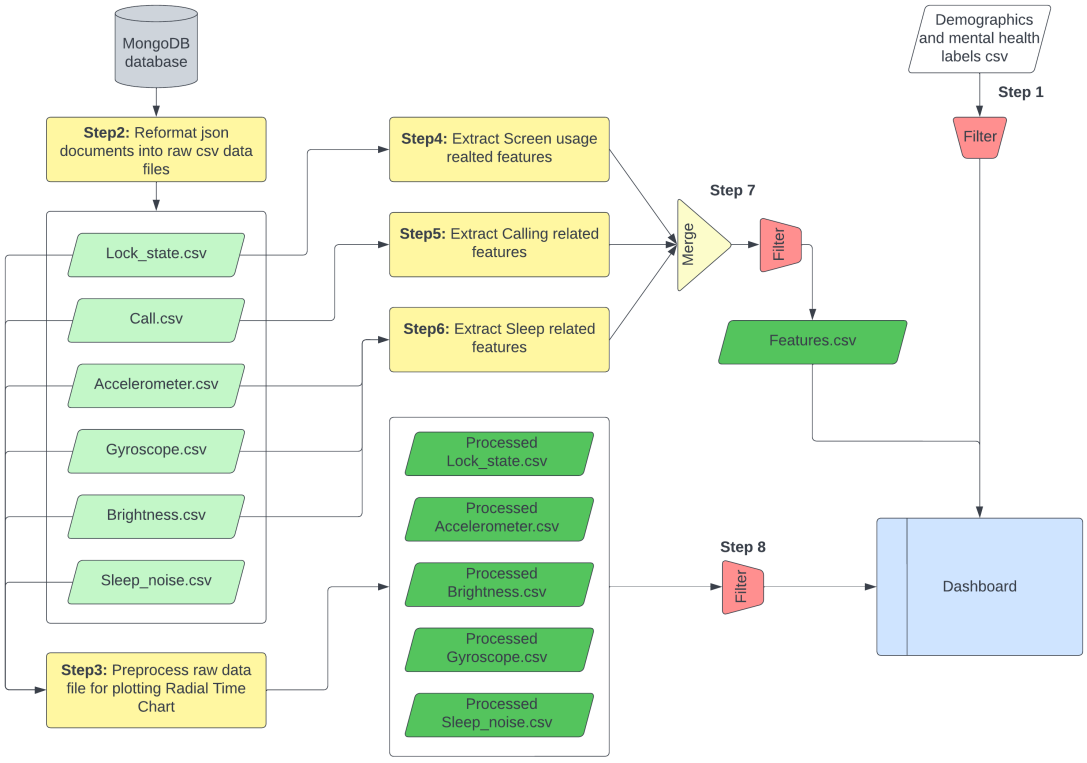


Figure 5.1: Block level diagram showing preprocessing steps needed to make the data Dashboard ready. (Color-coding: Yellow - Preprocessing step, Light Green - Intermediate files, Dark Green - Final preprocessed files, Red - Filtering)

Fig. 5.1 shows block level processing steps required to make the raw data dashboard ready. As seen in the figure, preprocessing is done in 8 steps: in Step1, we merge the two CSV file containing demographic information of all the participants. First file being merged in STEP 1 as shown in Fig. 5.1 contains the demographics and mental health labels of the participants in PROSIT study. The attributes in this files are as given below:

$$[\text{"}participantId\text{"},\text{"}age\text{"},\text{"}gender\text{"},\text{"}device\_type\text{"},\text{"}label1\text{"},\text{"}label2\text{"}]$$

Here, **device_type** indicates whether participant used an Android or iOS device during study. **label1** represents a four-category value indicating the symptoms identified during clinical assessment. **label2** is a binary value representing the diagnosis identified for the participant. label1 and label2 values are label encoded and true meaning of these value is unknown to us.

In **STEP 2** as shown in Fig. 5.1, the data needed for analysis and visualization is fetched from MongoDB server using python scripts having pymongo DB connectors and reformatted into CSV from JSON format. This reformatting was needed since we were using pandas library for data manipulation and preprocessing. Although pandas data frame could read the data directly from the original JSON documents in MongoDB, we decided not to use this approach and go with the idea where we create a reformatted CSV copy of MongoDB data on the same server which hosts the database and clean the data in these files for further processing. This initial preprocessing method allowed us to fetch data from multiple databases and collections inside a database, then merge them all to make further preprocessing easier.

In **STEP 3** as shown in Fig. 5.1, the raw CSV data files are processed so that the newly generated data files can be used to plot a radial time chart. The raw data from the accelerometer, gyroscope, brightness, and sleep noise sensors are cleaned and aggregated over one minute and saved in CSV files while preprocessing. There is a filtering step before the data gets used in the dashboard. Filtering is performed to exclude data from participants who were only in the study for 14 days or less. The decision to filter out such participants was made after discussions with domain experts and supervisors. More details about radial time chart can be seen under the

Visualisation techniques section below.

In **STEP 4** as shown in Fig. 5.1, *Lock_state.csv* data which represents the raw CSV data of screen lock state in iOS is used to extract screen usage related features. In the case of Android data, *power_state.csv* data is used to extract the same. Below is the list of screen-related features which were extracted using this data.

1. Number of screen locks in a day
2. Time in hours when the device screen was first unlocked in a day
3. Time in hours when the device screen was last locked in a day
4. Maximum duration in minutes where the screen was in continuous unlock state
5. Total duration in minutes of screen unlock duration

In **STEP 5** as shown in Fig. 5.1, *Call.csv* data which represents the raw CSV data of call related logs in iOS is used to extract a set of calling features. In case of Android data, *calls__callDate_callDurationS_callType_phoneNumberHash.csv* data is used to extract the same. Below is the list of call-related features which were extracted using this data.

1. Number of missed calls in a day
2. Number of dialled calls in a day
3. Number of incoming calls in a day
4. Minimum duration in minutes of any incoming call in a day
5. Maximum duration in minutes of any incoming call in a day
6. Total duration in minutes of all incoming calls in a day
7. Number of outgoing calls in a day
8. Minimum duration in minutes of any outgoing call in a day
9. Maximum duration in minutes of any outgoing call in a day
10. Total duration in minutes of all outgoing calls in a day

11. Total number of any type of calls in a day

12. Total duration in minutes of time spent on calls in a day

In **STEP 6** as shown in Fig. 5.1, *Accelerometer.csv*, *Gyroscope.csv*, *Brightness.csv* data which represents the raw CSV data for iOS participants is combined together and used to extract a set of sleep related features. In case of Android data, *accelerometer_m_s2__x_y_z.csv*, *gyroscope_rad_s__x_y_z.csv* data is used to extract the same. Screen brightness data is not captured in Android devices due to technical reasons. Below is the list of sleep related features which were extracted using this data.

1. Starting time of sleep in hour of the day

2. Ending time of sleep in hour of the day

3. Total sleep duration in hours

In **STEP 7** as shown in Fig. 5.1, the extracted screen usage, calling, and sleep-related features are merged and filtered to generate a Features.csv file which will be fed into the dashboard.

In **STEP 8** as shown in Fig. 5.1, the raw sensor data, which is preprocessed to remove duplicates and aggregate over 10 minutes time blocks is filtered for specific participants and fed into the dashboard.

In total, we extracted 20 features representing a participant's daily behaviour. Out of these 20, 5 features are screen usage related, 12 features are calling related, and 3 features are sleep-related. We chose to pre-process iOS and Android data separately due to a difference in the format in which data was kept for both participant device types. Also, during the development of this thesis, we experimented with extracting mobility-related features using the GPS coordinates of the participants. Due to the sensitivity of the dataset and the precision of the recorded GPS coordinates, it was decided not to use any GPS characteristics in the analysis. As a result, this has been left out of future development. These three categories of behavioural features are combined and filtered to remove participants with less than 14 days of data. The resulting dataset is stored as a *Features.csv* file, which will be used directly to plot visualizations on the dashboard.

5.3 Visualization techniques and design

Different visual components and their interactions were designed to meet the requirements listed in Section 5.1.

5.3.1 Design Goals

The design goals that were considered during the development of the visual system are listed below.

- **(G1):** Project all the participants in an overview plot that allows groups or cluster formations based on similarities in smartphone usage behaviour.
- **(G2):** Encode additional information within the overview plot to create visual markers that can provide a brief insight into the participant's features. The traditional approach of just using a dot or circle can be replaced with a custom glyph that presents this overview type. **G1 & G2** design goals will tend to requirements **R1 & R2** described in Section 5.1.
- **(G3):** Ability to allow for the selection of different participants or participant groups to narrow the scope of the analysis.
- **(G4):** Present a detailed view to provide an in-depth analysis of the data representing individual participants. This can be done by visualizing the sub-selected features representing participants and allowing the user to compare the feature of individuals w.r.t the remainder of the participant groups. **G3 & G4** design goals will tend to requirement **R3** described in Section 5.1.
- **(G5):** Visualize the raw data that was used to compute the extracted feature values and allow the user to validate the correctness of the computed feature value.
- **(G6):** Allow interactivity to filter the visualization of this raw data for a specific period during the study period. **G5 & G6** design goals will tend to requirement **R4** described in Section 5.1

Fig. 5.2 illustrates the dashboard implemented as part of this thesis. Dashboard consists of three Views each meant to accomplish the design goals mentioned above.

Glyph View(Fig. 5.2 2) implements a *Glyph based Scatter Plot* which achieves design goals **G1, G2 and G3**. Aggregated Features View(Fig. 5.2 7) implements a *Parallel Co-ordinate chart* which achieves design goals **G4**. Radial View(Fig. 5.2 5) implements a *Radial Time Chart* that achieves design goals **G5 & G6**. The dashboard also consists of a *Menu*(Fig. 5.2 1) which allows the user to select multiple options that vary the settings inside these views. Most important is the *Feature selection* drop-down, which allows users to select from a list of available features that represent behaviour of individual participants and generate a DR projection in Glyph View based on these selection. User can also choose from *T-SNE* or *PCA* type of DR technique from this Menu.

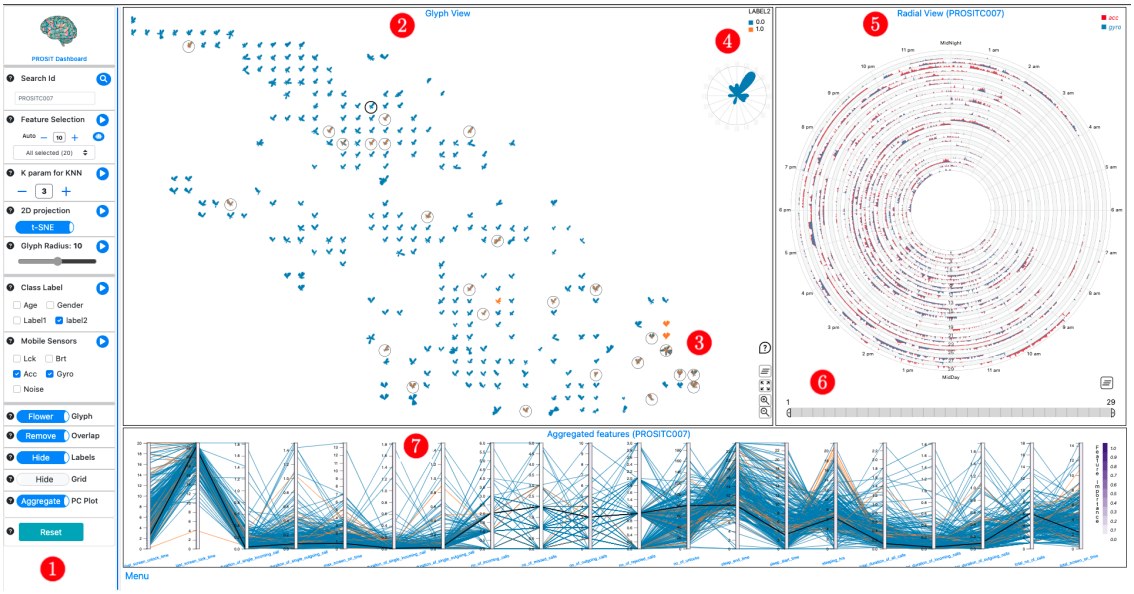


Figure 5.2: Annotated snapshot of proposed data visualization system with PROSIT dataset. Sidebar (1) has multiple options to sub-select feature combinations, alter the parameters to KNN classifier, select a DR method, and change few visual settings. User can select from 4 different Class Labels which colour codes the glyphs. Glyph View (2) projects the datapoint on a 2D space in the form of a flower or polygon glyph. Radial View (5) renders the continuous mobile sensor data over concentric circles to identity routine behavioural patterns. Radial View (6) also consists of a brush filter to interact and with PCP feature View to filter individual participants data. PCP feature View (7) has two different options to show either aggregated features of all the participants or daily features of individual selected participant.

5.3.2 Glyph based Scatter Plot

Scatter plots are ideal for showing overview of all the datapoints in a single chart. It uses two Cartesian coordinate values (x, y) to decide the position of the point over the chart. In case of high dimensional data, DR techniques can be used to reduce the larger number of dimensions to two dimensions which can then be used to identify the position of points over the chart. This approach allows the user to view the overall structure of the datapoints relative to each other. DR technique is usually combined with classification to colour code the data points to identify similar participant groups [17]. More details about this can be found in Section 2.2 which describes briefly about DR and Section 2.3 which describe briefly about classification techniques. DR technique combined with KNN classification to classify participants based on its neighbours was the implementation of design goal **G1** mentioned in Section 5.3.1.

In our visual system, we decided to use t-SNE and PCA as a choice of DR techniques. t-SNE works well in preserving both the local and global structure of the data by retaining the data's significant features [95], which is crucial to identify clusters and similar user groups. PCA on the other hand tries to preserve the global structure of data which is sometimes necessary to view the overall structures within the data. Also, most of the time, data points on scatter plots are represented using dots or circles, which do not convey any information. Often colour-coding the datapoint representations is used to indicate some feature, but the shape of the data point itself does not indicate any additional meaning. We were interested in using the shape of each DR projection of data points to represent meaningful information that can be used in the analysis. This was mentioned in design goal **G2** described in Section 5.3.1. The design of the glyphs-based scatter plot used in this thesis was inspired by work by Dietrich et al. [56]. Glyphboard interface is based on mapping each data item to a colour-coded glyph whose two-dimensional position on the plot is computed using DR. Fig. 5.3 shows the two different types of glyphs patterns used to represent 5-dimensional data in the Glyphboard interface. However, Fig. 5.4 shows two different types of glyphs, i.e. Flower glyph and Polygon glyph, used to represent the sub-selected features representing an individual participant's smartphone usage behaviour. In visualizing huge data sets, we need to reduce the size of glyphs to

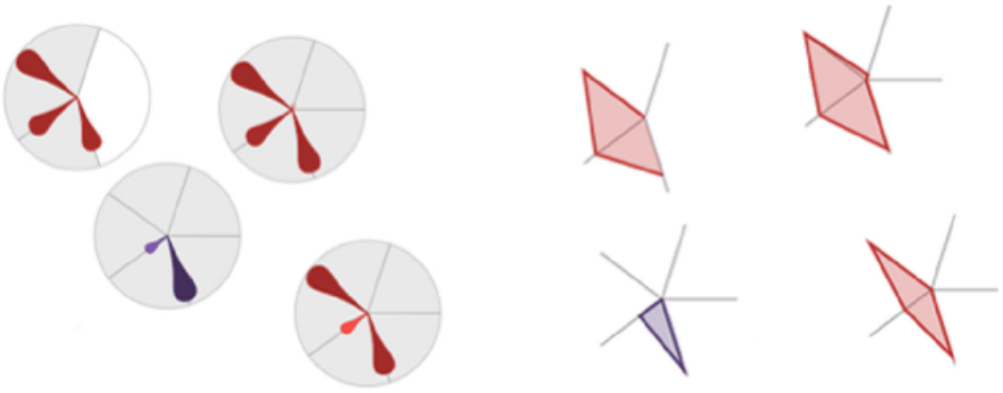


Figure 5.3: Glyphs used in Glyphboard interface by Dietrich et.al[56]

Figure 5.4: Flower glyphs and Polygon glyphs used in this thesis to represent individual participants. Shape of glyphs is indicative of sub-selected features representing individual participant's smartphone usage. In case of flower glyphs, the length of each petal represents the normalized feature value, and in case of polygon glyph, the distance of each edge from its center represents the normalized feature value of respective data instance.

remove occlusion that may occur with a high number of participants. To tackle this problem, our interface has an input slider in the menu that can be used to vary the size of each of the glyphs. We also have a brushing feature that allows the user to zoom into a specific region of interest, and the glyph outside the interested region will be eliminated from the visual canvas. A subset of participants in interested region can then be selected using lasso selection feature of the tool. This feature was implemented keeping design goal **G3** mentioned in Section 5.3.1.

5.3.3 Parallel Co-ordinate Chart (PCP)

Historically, the role of visualization has been to facilitate discovery and understanding of high-level structure of the data in ways impossible by direct examination of

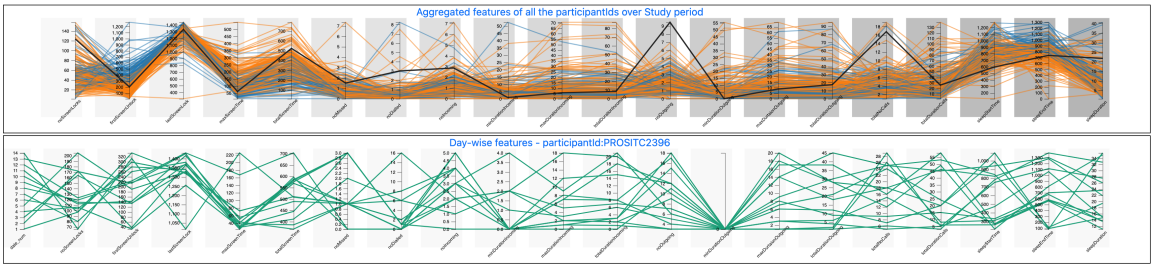


Figure 5.5: PCP charts implemented in our visual system. 2 PCP charts; one in the top visualizes mean aggregated feature values for individual participants over complete study period, and one in bottom illustrates daily-wise feature values of selected participant.

the data themselves. Increasingly, however, visualizations must also serve as effective interface to access details and refined statistical features of the data [31]. Parallel coordinates is a methodology for visualizing N-dimensional geometry and multivariate problems. Parallel coordinates transform multivariate relations into 2-D patterns, a property that is well suited for visual data exploration and analysis [52]. Parallel coordinate charts are usually used as an ideal supplement to the scatter plot DR mappings of high dimensional data. Where the scatter plots project an overview of the global attributes, PCP gives insights into the local attributes. The pattern found while analyzing PCP charts can help explain why two points are closer or farther from each other in a DR mapped glyphs scatter plot shown in Section 5.3.2. Fig. 5.5 illustrates the PCP charts developed as part of this visual analytic system. This visualization implements the design goal **G4** mentioned in Section 5.3.1.

5.3.4 Radial Time Chart

The approach we took for analysis was to implement DR coupled with KNN classification to identify similar participant groups based on smartphone usage. Since our data does not have features that can directly be used for analysis we need the part where features have to be extracted from raw data using simple algorithms. These features are then analyzed to identify behaviour of individual participants and compare the same with other participant groups. The quality or accuracy of these extracted features are very important here and since our data had lots of missing values, we needed a way to visually see the raw data for any participant and then cross-verify or edit the feature values extracted by generic algorithms. This approach

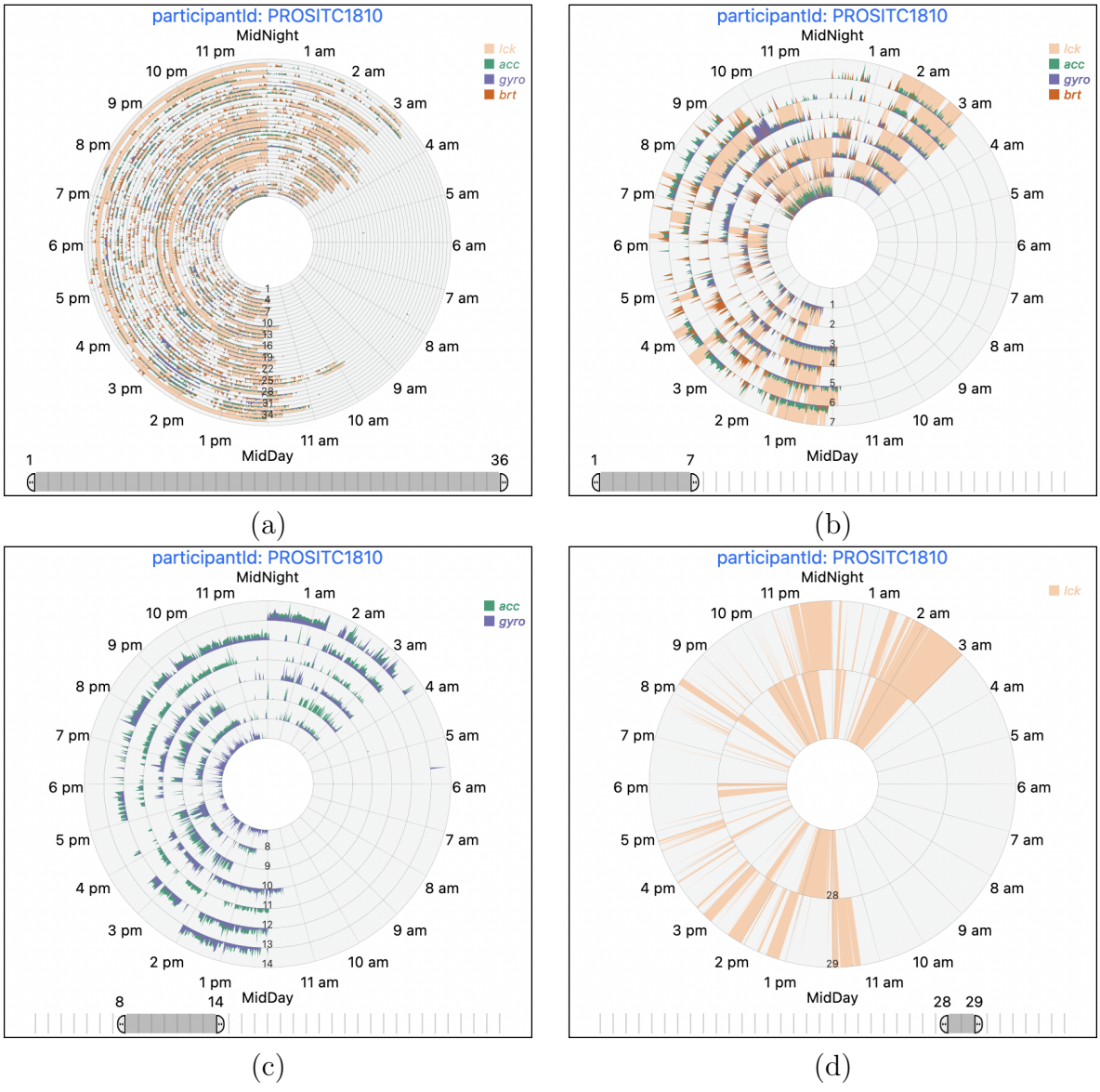


Figure 5.6: Screenshot illustrating Radial time chart implemented in this thesis. (a) shows four smartphone sensor's data (lockstate, accelerometer, gyroscope, brightness) visualized for complete study duration. (b) shows feature to brush filter data to visualize only one week's of data (day1-day7). (c) shows only accelerometer and gyroscope data for second week (day8-day14). (d) shows the imputed lock state data visualizing the time periods when the mobile screen was unlocked and being used.

adds a positive feedback where the user of visual system can use visualization charts to improve the accuracy of these engineered features. This is specifically true in case of sleep related features. As shown in Fig. 5.1 STEP 6 of data preprocessing dealt with extracting sleep related features.

The relationship of insomnia to psychiatric disorders is important for several reasons. Sleep disturbances may be an early sign or even the cause of some psychiatric disorders. A study by Daniel et.al. [37] which analysed data from 7954 adults indicated that more than 57% of those with insomnia and 64% of those with hypersomnia had a psychiatric diagnosis compared with 24.9% of those without a sleep complaint during 18-month study period. Other studies have indicated that about 21% and 13% of those with insomnia had symptoms resembling major depression and generalized anxiety, respectively [74]. Another study by Neckelmann et.al [72] researched about relationship of insomnia to the development of anxiety disorders and depression. Results from this study were consistent with earlier studies that suggested insomnia being a risk factor for the development of anxiety disorders.

Literature confirms the known fact that irregular sleeping routines highly correlate with mental instability which could be indicative of mental health disorders. Also extracting sleep related features is easier using mobile sensing data as the user ideally do not use their smart phones during sleep. The algorithm that we used in our feature engineering of sleep combined accelerometer, gyroscope, and screen brightness data to identify time blocks of non usage periods and then combine these non usage time blocks to estimate the sleep start, sleep end and sleep duration. Though this algorithm works well in ideal usage scenariOS combined with high quality of data logging, we observed that with PROSIT data we were observing significantly higher or inconsistent sleep features. We needed a visualization that allowed us to view the raw data and use this raw data to intuitively justify the extracted sleeping features and change them if the algorithm overestimated or underestimated sleep due to various reasons involving inconsistent data.

We developed radial time chart that allows users of visual system to select the raw data that they are interested using checkbox buttons present in the *Menu* and analyse the quality of features extracted so that they can justify the outliers seen in glyphs scatter plot. Extensive research online revealed that radial charts are particularly

useful for time series data, since they allow to visualize cyclical or seasonal trends that is seen in nature and in turn in human behaviour [105, 87, 35]. This cyclical behaviour is also famously known as circadian rhythm which is a natural, internal process as a result of evolution that regulates the synchronization of biological and behavioral processes to the external temporal environment. Meaning, it regulates sleep–wake cycle and repeats roughly every 24 hours [100]. The developmental goal of radial time chart was to allow the user to visualize all those mobile sensor data that would indicate non usage or usage and help identify sleeping patterns and how these patterns fluctuate over the course of study period. Fig. 5.6 illustrates the radial time chart developed as part of this thesis to identify and validate sleep features. This visualization implements the design goal **G5**. i.e. sleep start time, sleep end time and sleep duration. Fig. 5.6(a)-(d) illustrate the implemented features in the visual system to filter the timeline and select the data within a specific duration belonging to the study duration. This feature implements the design goal **G6** mentioned in Section 5.3.1.

Chapter 6

Use Case

This chapter shows our visual approach's utility in exploring a given dataset using hypothetical scenarios. We presents two usage scenarios to illustrate how our visual approach may be used to examine data to execute the following tasks.

6.1 Usage Tasks

1. **(T1):** Feature sub-selection to explore the different combinations of features that possibly create distinct, visually separated subgroups with similar data points.

2. **(T2):** Compare an individual data point's characteristics to those of the rest of the group or a previously chosen subgroup.

3. **(T3):** Identify an outlier or data points near the boundaries separating two subgroups.

4. **(T4):** Assess the accuracy of the extracted sleep-related features by eyeballing raw mobile sensor (brightness, accelerometer, and gyroscope) data patterns on a radial time chart (This specific scenario only applies to the PROSIT dataset, which has continuous time series mobile sensor data).

We start by presenting a simple introductory scenario involving the analysis of the well-known IRIS dataset [6], followed by a scenario of analyzing the PROSIT dataset. In both scenarios, we introduce John, a hypothetical user of the proposed tool.

6.2 Usage Scenario 1: Exploring Iris Dataset

John works as an analyst in a research team specializing in data analytics for mental healthcare. The company's R&D team has proposed a new data visualization system

to help analysts explore a given dataset about patients' daily behaviour and make decisions about developing intelligent ML-based applications that mental health professionals can recommend as a supplement to therapy. John was tasked to assess the efficacy of this new tool for exploratory data analysis (EDA). John chose the IRIS dataset for testing since it is perhaps the best-known dataset in the pattern recognition literature. This data set contains three classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other two; the latter are not linearly separable from each other.

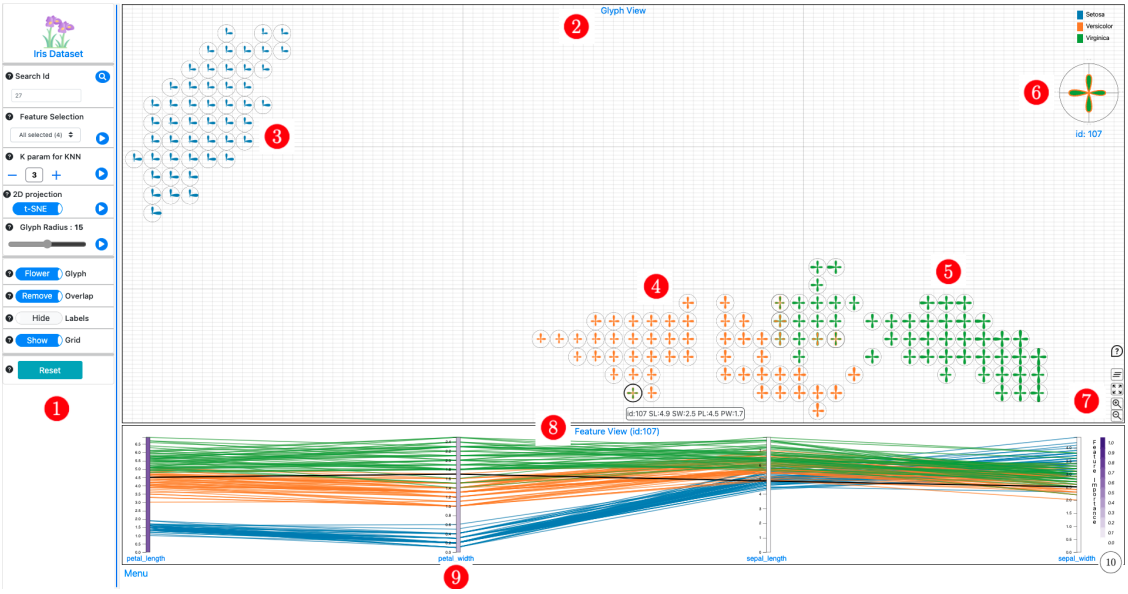


Figure 6.1: Annotated snapshot of proposed data visualization system with IRIS dataset. Sidebar (1) has multiple options to sub-select feature combinations, alter the parameters to KNN classifier, select a DR method, and change few visual settings. Glyph View (2) projects the datapoint on a 2D space in the form of a glyph.

John loads the IRIS dataset into the tool, and Fig. 6.1 shows the first look John observes on the screen. John notices that there are several options available in the sidebar (Fig. 6.1 1). John then focuses his attention on Glyph View (Fig. 6.1 2) of the dashboard. He notices two significant clusters of data points. First one being Setosa flowers (Fig. 6.1 3) and second one being Versicolor (Fig. 6.1 4) and Virginica (Fig. 6.1 5) flowers. He also notices that a few of the glyphs have different coloured outer strokes (Fig. 6.1 6) representing mislabelled data points from a KNN classifier. John shifts his attention to Feature View (Fig. 6.1 8), which is a

parallel coordinate plot having colour-coded lines representing individual data points. John also observes that each axis on the PCP plot has a rectangular colour-coded bar (Fig. 6.1 9) which represents the importance of that feature to distinctively separate data points into subgroups. The color-coded feature importance value of which is shown over a legend in (Fig. 6.1 10). John quickly notices that *petal_length* is the most important feature that can be used to classify the IRIS dataset, closely followed by *petal_width*. John also notices that *sepal_length* and *sepal_width* have a lower importance score and hence cannot be used to make distinct subgroups based on flower type.

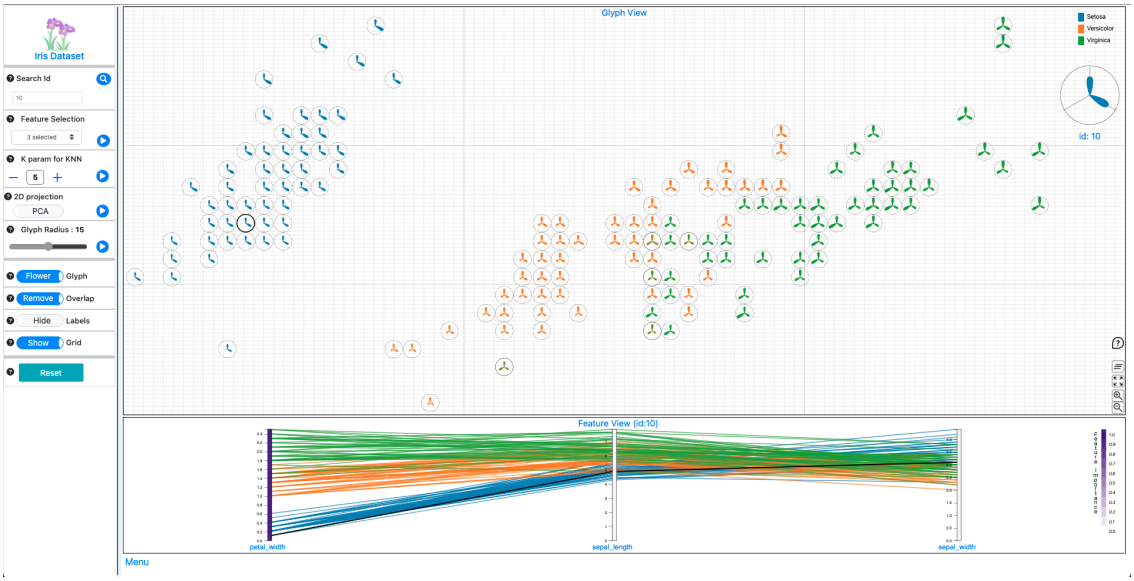


Figure 6.2: Annotated snapshot of glyph projections after only three features were selected from feature sub-selection, with a K parameter value of 5 for a KNN classifier and PCA as DR technique.

To test usage task **T1** described under Section 6.1, John clicks on *Feature Selection* drop-down from sidebar (Fig. 6.1 1) and selects three random features to explore how the projections change in the Glyph View. He also increases the *k param for KNN* classifier to 5 and selects *PCA* from *2D projection* Menu option. Fig. 6.2 shows the visualizations resulting from this selection. The individual glyphs over the Glyph view now have only three petals, and the PCP plot in the Feature view also has three axes representing data from three pre-selected attributes. John observes that with this combination of feature subselection, *petal_width* has significantly higher

importance in classifying subgroups out of the three feature selections.

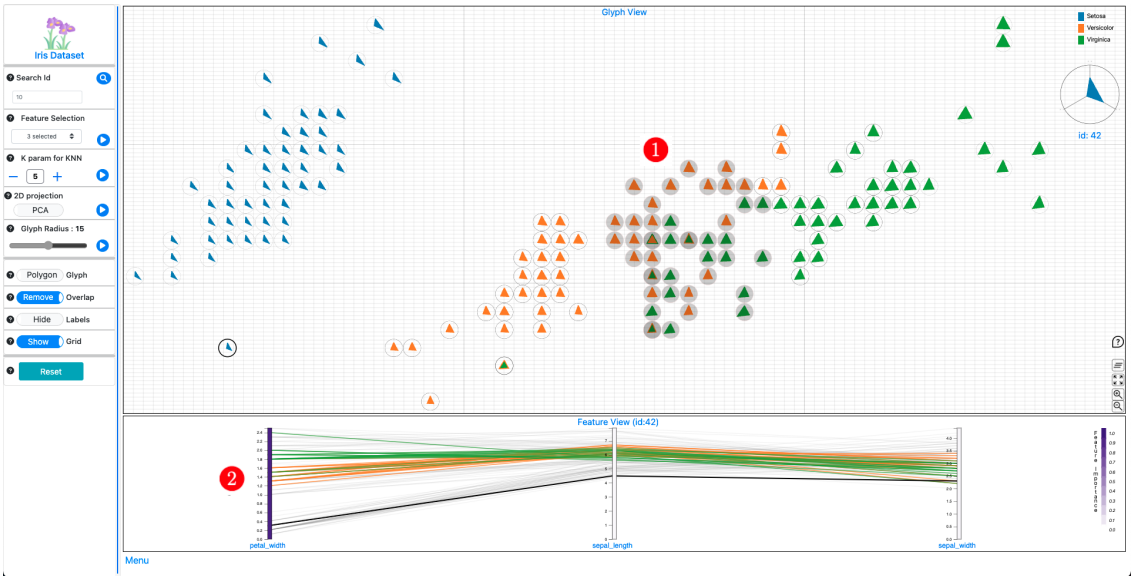


Figure 6.3: Annotated snapshot showing the changes in Glyph View and Feature View after few of the data points were selected using Lasso selection.

To test usage task **T2 & T3** described under Section 6.1, John now uses the Lasso selection tool to pre-select some of the data points from the central region of PCA projections. He notices that the PCP plot in Feature View instantly updates to show only these pre-selected data points. Fig. 6.3 (1) shows the glyphs selected using lasso selection and Fig. 6.3 (2) shows their respective data instances in PCP feature view. The position of glyphs over the Glyph view allows John to identify similar data points. John notices that the datapoint placed far away from any cluster represents dissimilar data points that may indicate an outlier. Also, the difference in Stroke colour for some of the glyphs indicates that these data points may lie in the boundary regions separating two distinct groups.

6.3 Usage Scenario 2: Exploring PROSIT Dataset

Having tested the visual tool with IRIS dataset, John feels confident about the tool since the findings from the visual exploration matches the preconceived knowledge about IRIS dataset. John now proceeds to use this tool to explore PROSIT dataset to understand mobile usage behaviour of different patients. John loads the PROSIT

dataset into the tool, and Fig. 5.2 shows the first look John observes on the screen. He notices that there are three new options available in the Menu as seen in (Fig. 5.2 1) to select the *Class Label* , *Mobile Sensors* and type of *PC Plot*. John also observes Radial View as seen in (Fig. 5.2 5) which renders raw mobile sensor data. This view also comes with a brush filtering feature as seen in (Fig. 5.2 6) to control the data being rendered in Radial View.

John is interested to explore the smartphone screen usage behaviour of participants of different age groups. This represents usage task **T1** described under Section 6.1, which is to sub-select features for data exploration. From the Menu options, John selects three features *'No of Unlocks', 'Max On time', 'Total On time'* and changes the *Class Label* from *Label2* to Age. He also changes the Glyph type from *Flower* to *Polygon*. Fig. 6.4 shows the first look of these selections from the Menu options. He notices that the participants near top right corner region marked as Fig. 6.4 1 have higher values for screen usage features compared participants in bottom near left corner marked as Fig. 6.4 2.

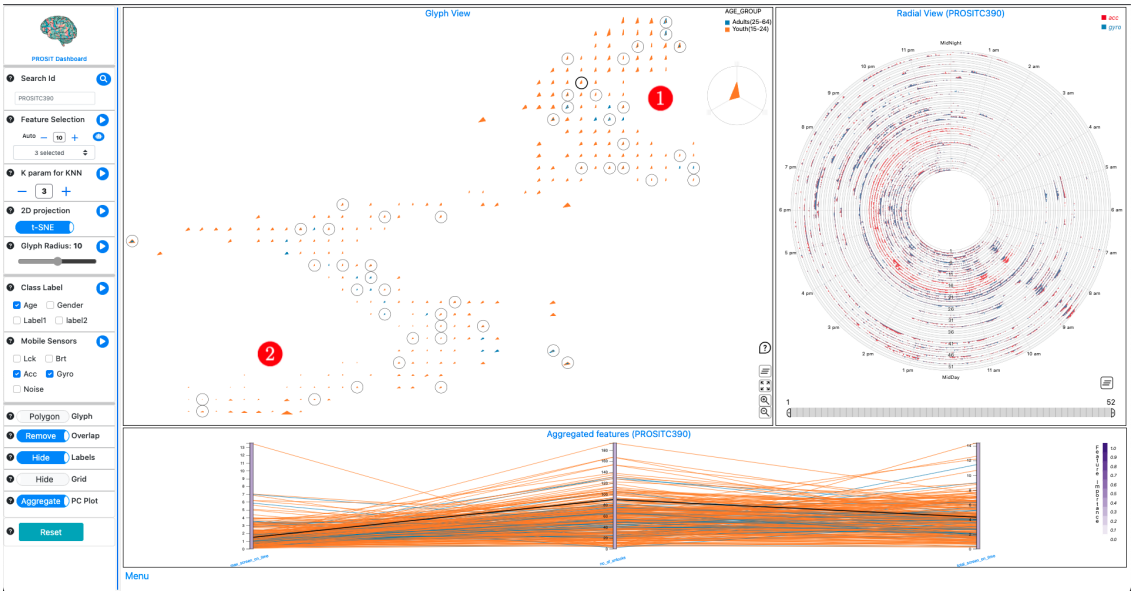
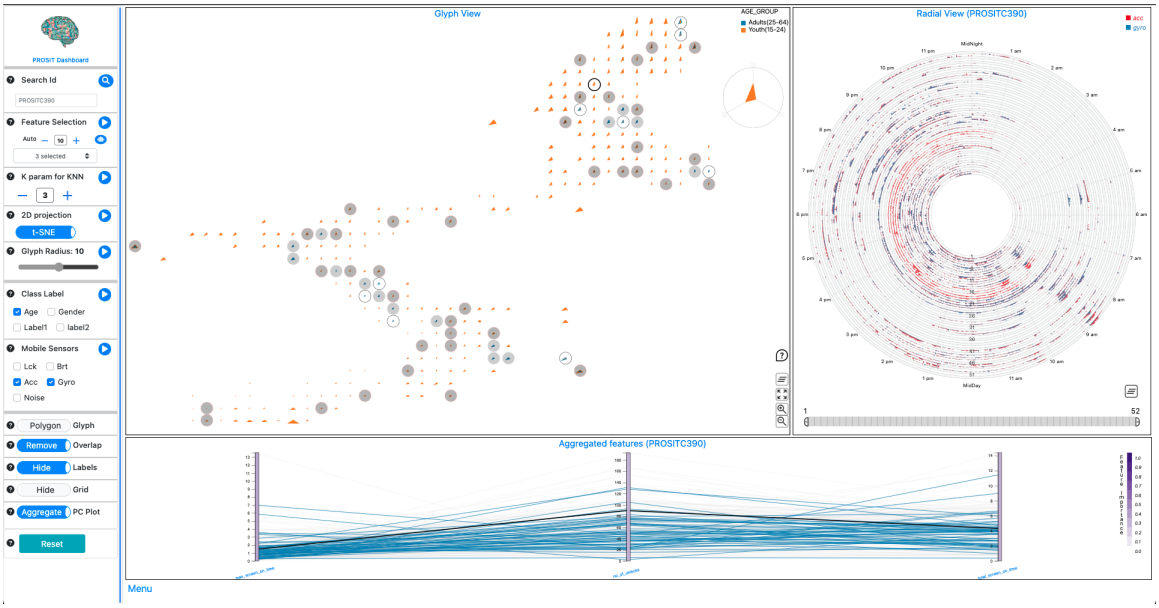
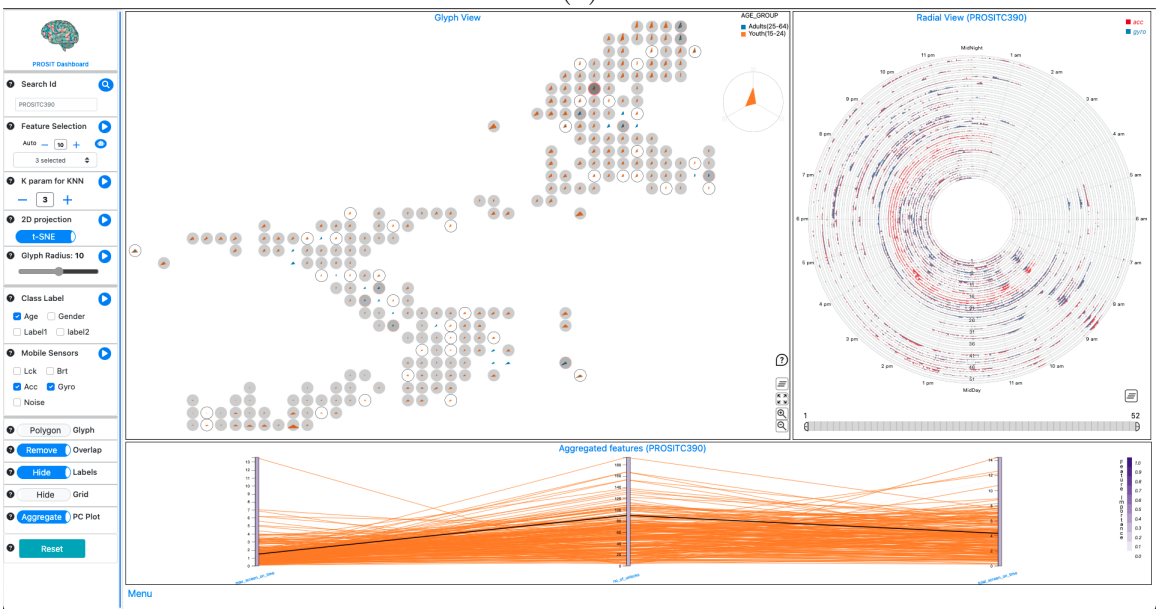


Figure 6.4: Annotated snapshot showing the resulting view from the dashboard when analyst selects 3 screen usage related features, with a selection to view polygon shaped glyphs.

John is now interested to see the screen usage behaviour of different age groups. He now wants to only see data belonging individual age groups. To do this, he clicks



(a)



(b)

Figure 6.5: Snapshots from the dashboard illustrating utility of tool to filter out subgroups belonging to same category. In this case Age group. Fig.a shows screen usage related data for Adults(25-64) age group. Fig.b. shows screen usage related data for Youth(15-24) age group.

on the legend on the top right corner of *Glyph View* marked as Fig. 5.2 4. Fig. 6.5 shows two different views he sees when he clicks on different legend buttons on the screen. John notices that there is no distinct user behaviour for all the participants in the same age group. Distribution of screen usage behaviour is almost identical for both Adults and Youth for users in this dataset, indicating that screen usage behaviour of participants in this user group doesn't depend on the age group of a person.

John now focuses on clusters which are formed on the projections and lasso selects one such cluster and compares the screen usage features of these sub-selected participants with a a participant in a different cluster. Circles with Gray fill colour in Fig. 6.6 1 shows the sub-selected participants in one cluster and glyph with Dark outer boundary circle in Fig. 6.6 2 highlights individual participant selected from a different cluster. John also observes that Dark line in PCP view as shown in Fig. 6.6 3 represents the features of individual participant and it can be seen that this participant has significantly higher *no_of_unlocks* values compared to lasso selected participants in different cluster. This usage task represents **T2** mentioned in Section 6.1.

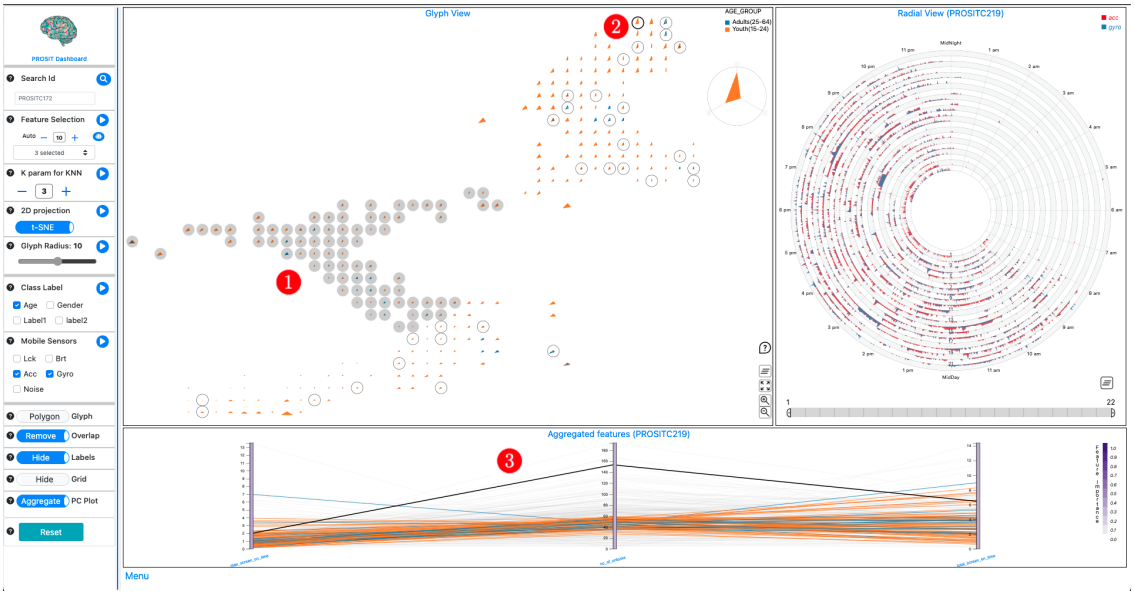
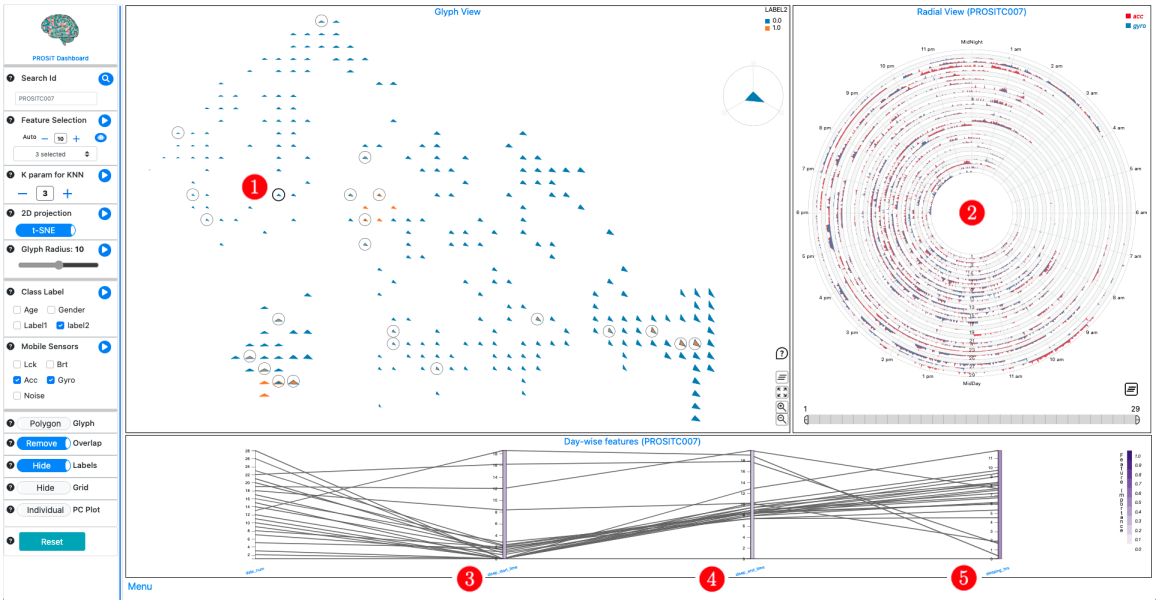


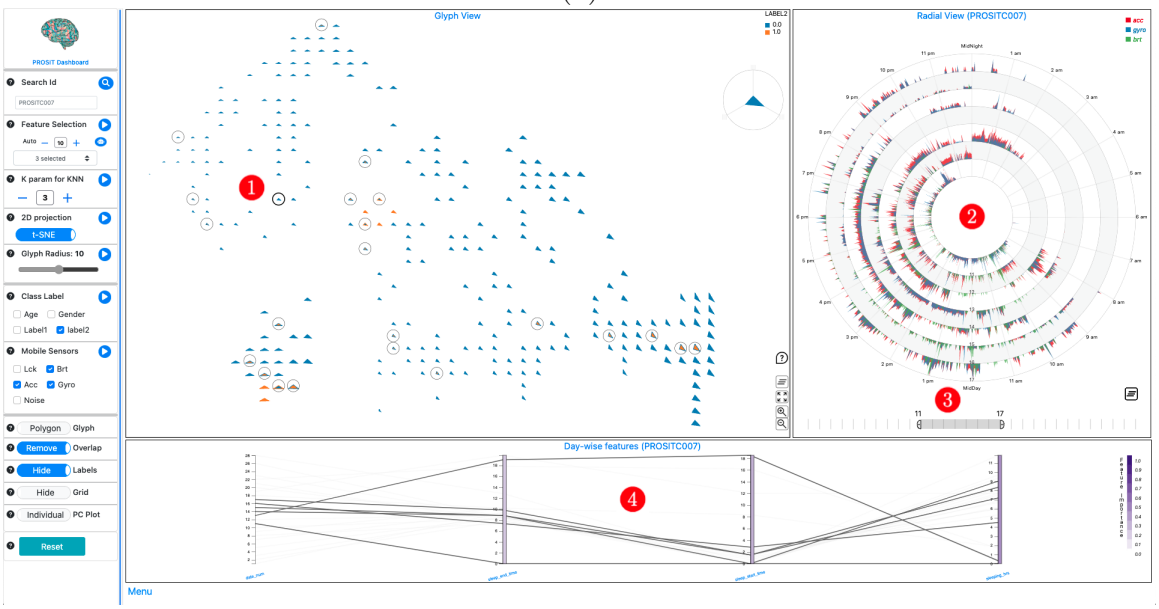
Figure 6.6: Snapshots from the dashboard illustrate a usage scenario where an analyst selects a subgroup using lasso selection and compares the features of a subgroup with features of individual participants from another cluster.

John now focuses on Sleep analysis and wants to assess the quality of sleep features extracted from the preprocessing algorithm. He now selects *'Sleep Start Time', 'Sleep End Time', 'Sleep Duration'* from *Feature Selection* drop-down in the Menu and changes the *Class Label* from *Age* to *Label2* which represents mental health label. John also toggles button to select type of PC plot from *Aggregate* to *Individual*. This will change the PC plot and show features belonging to participants selected in Glyph View. Fig. 6.7.Fig.a (1) shows the participant selected for Sleep Analysis. John focuses his attention on Fig. 6.7.Fig.a (2) which shows a pattern in accelerometer and gyroscope data. By eye balling over this pattern in Radial View, John estimates that selected participants usual sleeping time is from 2AM until 9AM. John looks at estimated *sleep_start_time* (Fig. 6.7.Fig.a (3)), *sleep_end_time*(Fig. 6.7.Fig.a (4)) and *sleeping_hrs*(Fig. 6.7.Fig.a (5)) from preprocessing algorithm. It can be seen that majority of lines in PCP plot cross over around 2 for *sleep_start_time* and around 9 for *sleep_end_time* with a majority of estimated sleep duration ranging anywhere from 5hours to 10hours. John also explores the filtering capability of Radial view to see data only for a some days during his study period. To do this, John drags the Brush filtering feature as shown in Fig. 6.7.Fig.b (3) and notices that only data for these filtered dates is show over the Radial View (Fig. 6.7.Fig.b (3)). John also notices that lines over PC plot also get update to show only data belonging to these filtered dates (Fig. 6.7.Fig.b (4)). This usage task represents **T4** described in Section 6.1.

By exploring interactive visualizations presented over Glyph View, Radial View and PC plots, John can understand individual profile of a participant under study and compare it's behavioural indicators with the remainder of the group or lasso selected participant group. From the analysis, John observes some of the mental health markers indicated in the literature and correlates them to the participants in PROSIT. Using this tool, John could filter out individual participants' sensor data and behavioural features and see how these features varied throughout the study duration. John noticed that some participants diagnosed with mental illness had higher screen usage and irregular sleep or activity patterns. Some participants also had fewer incoming and outgoing calls and calling duration, which might indicate depression and isolation from friends and family. John sees the Visualization dashboard's value in viewing participants' overall mobile usage and identifies those with unhealthy or



(a)



(b)

Figure 6.7: Snapshots from the dashboard illustrate the tool's utility of Radial View to assess sleep-related features. Fig.a shows mobile sensor data visualized over a Radial view for the entire time of the participant's study period. Fig.b. shows mobile sensor data visualized over a Radial view for a brush-filtered period.

irregular smartphone usage, which might indicate anxiety or depression.

Chapter 7

Conclusion

In this study, we developed a data visualization system to explore behavioural patterns extracted from mobile sensing data collected as part of the PROSIT project. We formulated our design goals to maximize the potential of visualization and machine learning to highlight critical mental health indicators. Both approaches help to identify similar user groups and individual daily behavioural patterns. Further, we drafted two usage scenarios to demonstrate the usability of this tool using the IRIS dataset and behavioural features extracted from the PROSIT dataset. In the following sections, we outline some of the limitations of this work and future developments, including preprocessing and visualization.

7.1 Limitations and Future Work

7.1.1 Data preprocessing

Preprocessing has been a major challenge with the PROSIT dataset. Much research was conducted to learn about features that can be extracted using mobile sensing data that strongly correlate with mental health and behavioural issues [48, 83, 89]. A significant amount of research described excessive screen time as compulsive behaviour and linked it to a few depressive symptoms [65, 91]. Irregular sleeping routines and reduced physical activity have also been strongly linked with mental illness [30, 78, 79, 94]. Extracting high-quality screen usage and sleep-related features like sleep duration, sleep start and end time, and sleep latency which is the length of time a person takes to fall asleep, using algorithms on mobile sensor data has also been difficult. A few reasons like missing data, irregular mobile usage behaviour of different participants, and change in timestamps post-midnight made it challenging to design a generic algorithm that can extract the above features for all the participants. These

concerns prompted the development of a radial chart-based visualization to help understand sleep behaviour by plotting continuously recorded data from mobile sensors like accelerometer and gyroscope. As part of future work, a utility similar to active learning can be developed where a learning algorithm can interactively query a user to label new data points with the desired outputs. In this tool users can be asked to visually analyze these continuous sensor data over Radial View and edit the sleep-related features, which can recursively improve the process of sleep detection. Some research [83] also indicates a strong correlation between depressive symptoms and features extracted from GPS data. Although some work in this area was done during this thesis, it was not implemented due to the sensitivity of the GPS dataset. High-quality physical movement-related features from GPS data can add immense value to mental health analysis.

The proposed visual approach strongly depends on the quality of extracted features since these features are used to generate projections to identify similar participant groups. Improved algorithms to extract research-backed behavioural indicators are crucial to designing an intelligent system that can indicate mental health. Although much effort was put in this direction, there is further scope to improve the preprocessing algorithms to extract quality features that can improve the analysis and can also be used to build a prediction model that can support intervention.

7.1.2 Scalability and Performance

We have utilized glyphs-based scatter plots and parallel coordinate plots to represent multidimensional data in our approach. Although this method works well when there are quite a few dimensions, it is not scalable. Both glyphs and PCP plots will get cluttered if the size of dimensions increases to more than 50 features. Some studies have engineered over 1000 features to train models that can predict participants' personalities based on smartphone usage [89]. In cases where the dataset has many features, this approach might prove less valuable, and other solutions like matrix-based visual techniques might be more relevant.

When this thesis was written, the PROSIT project collected well over 2500 participants as part of various research studies. Participants were recruited for 4-6 weeks, and their smartphone usage data included events-based data like call logs, screen

lock/unlock logs and continuous sensor-based data like accelerometer and gyroscope. This data was stored in a secure server at Dalhousie University and access to the same was controlled. The number of data records collected from mobile sensors was in the hundreds of millions. For example, the file size of accelerometer data for roughly 1000 participants was well over 15GB. This data had to be preprocessed and engineered to display charts on the dashboard, and due to the enormous file size, there is a slight lag in rendering the same. Future work can include big data solutions for preprocessing and improving the rendering of data over the dashboard to improve scalability regarding the number of participants that can be analyzed using this approach.

With the help of visualization combined with feature engineering and machine learning, we attempted to build an analytic system that can aid with exploring mobile sensing data. Feature engineering was a significant challenge in this thesis as the quality of these features represents a participant's behaviour. We also acknowledge the lack of formal user testing of the interface and methods used in our tool. We were not able to conduct standard user testing due to insufficient time. This evaluation can be conducted in two parts. First, interviews with domain experts from clinical psychology about the usability of this tool to identify mental health of a participant based on their behavioural indicators. Second, a user study can be designed and conducted with a group of non-expert volunteers, to assess the usability of the tool to interact with the data and identify patterns. Users can be briefed about the tool and can be asked to answer a set of questionnaires than can assess ease of usability, possible improvements and cognitive load of exploring patterns. Most ideas implemented in this thesis were well researched and user-tested in their original papers, cited in Chapter 3 and Chapter 5, the novelty of this thesis was to put together various existing techniques in a unique way to help solve a bigger problem of leveraging data visualization and machine learning to identify mental health.

The scope of ideas that can be implemented using mobile sensing data is vast, and much more can be achieved. This thesis was initial exploratory research and hopefully will provide reference to researchers who'll further work with the PROSIT dataset.

Bibliography

[1] Bootstrap: A free and open source front end development framework for the creation of websites and web apps. https://getbootstrap.com.

[2] Bootstrap multiselect: A jquery based plugin to provide an intuitive user interface for selecting multiple attributes as inputs. this dropdown menu will contain options as checkboxes. https://github.com/davidstutz/bootstrap-multiselect.

[3] Data-driven documents: D3.js is a javascript library for manipulating documents based on data. d3 helps to bring data to life using html, svg, and css. https://d3js.org.

[4] dgrid: Python implementation of overlap removal technique described in [46]. https://github.com/fpaulovich/dimensionality-reduction.

[5] Flask: A micro web framework written in python. it provides tools, libraries and technologies that allows to build a web application. https://flask.palletsprojects.com/en/2.1.x/.

[6] Iris: This data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. https://archive.ics.uci.edu/ml/datasets/iris.

[7] jquery: A fast, small, and feature-rich javascript library. it makes things like html document traversal and manipulation, event handling, animation, and ajax much simpler with an easy-to-use api that works across a multitude of browsers. https://jquery.com.

[8] "number of smartphone subscriptions worldwide from 2016 to 2027" statista, feb 2022. https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/.

[9] pandas: A fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the python programming language. https://pandas.pydata.org.

[10] Predicting Risks and Outcomes of Social InTeractions (PROSIT). PROSIT is a research study that aims to improve the mental well-being of youth, howpublished = http://prosit.meierlab.info.

[11] Python programming language. https://www.python.org/downloads/.

[12] scikit-learn: A free machine learning library for python programming language. https://scikit-learn.org/.

[13] Who global action plan on physical activity 2018–2030: more active people for a healthier world. https://apps.who.int/iris/bitstream/handle/10665/272722/9789241514187-eng.pdf?sequence=1&isAllowed=y.

[14] Wolfgang Aigner, Silvia Miksch, Heidrun Schumann, and Christian Tominski. *Visualization of time-oriented data*, volume 1. Springer, 2011.

[15] Christopher Andrews, Alex Endert, Beth Yost, and Chris North. Information visualization on large, high-resolution displays: Issues, challenges, and opportunities. *Information Visualization*, 10(4):341–355, 2011.

[16] Natalia Andrienko and Gennady Andrienko. *Exploratory analysis of spatial and temporal data: a systematic approach*. Springer Science & Business Media, 2006.

[17] Etienne Becht, Leland McInnes, John Healy, Charles-Antoine Dutertre, Immanuel WH Kwok, Lai Guan Ng, Florent Ginhoux, and Evan W Newell. Dimensionality reduction for visualizing single-cell data using umap. *Nature biotechnology*, 37(1):38–44, 2019.

[18] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

[19] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[20] Chris Brunsdon, A Stewart Fotheringham, and ME Charlton. An investigation of methods for visualising highly multivariate datasets. *Case Studies of Visualization in the Social Sciences*, pages 55–80, 1998.

[21] Chris Brunsdon, Stewart Fotheringham, and Martin Charlton. Geographically weighted regression. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 47(3):431–443, 1998.

[22] Kerstin Bunte, Michael Biehl, and Barbara Hammer. A general framework for dimensionality-reducing data visualization mapping. *Neural Computation*, 24(3):771–804, 2012.

[23] Sam Cartwright-Hatton, Kirsten McNicol, and Elizabeth Doubleday. Anxiety in a neglected population: Prevalence of anxiety disorders in pre-adolescent children. *Clinical psychology review*, 26(7):817–833, 2006.

[24] Jinette Comeau, Katholiki Georgiades, Laura Duncan, Li Wang, Michael H Boyle, and 2014 Ontario Child Health Study Team. Changes in the prevalence of child and youth mental disorders and perceived need for professional help between 1983 and 2014: evidence from the ontario child health study. *The Canadian Journal of Psychiatry*, 64(4):256–264, 2019.

[25] James Cook, Ilya Sutskever, Andriy Mnih, and Geoffrey Hinton. Visualizing similarity data with a mixture of maps. In *Artificial Intelligence and Statistics*, pages 67–74. PMLR, 2007.

[26] Cathy Creswell, Polly Waite, and Peter J Cooper. Assessment and management of anxiety disorders in children and adolescents. *Archives of disease in childhood*, 99(7):674–678, 2014.

[27] Padraig Cunningham and Sarah Delany. k-nearest neighbour classifiers. *Mult Classif Syst*, 54, 04 2007.

[28] John Curry, Susan Silva, Paul Rohde, Golda Ginsburg, Christopher Kratochvil, Anne Simons, Jerry Kirchner, Diane May, Betsy Kennard, Taryn Mayes, et al. Recovery and recurrence following treatment for adolescent major depression. *Archives of general psychiatry*, 68(3):263–269, 2011.

[29] Sarah Jane Delany. k-nearest neighbour classifiers. 2007.

[30] Afsaneh Doryab, Jun Ki Min, Jason Wiese, John Zimmerman, and Jason Hong. Detection of behavior change in people with depression. In *Workshops at the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[31] Stephen G Eick and Alan F Karr. Visual scalability. *Journal of Computational and Graphical Statistics*, 11(1):22–43, 2002.

[32] Jon D Elhai, Robert D Dvorak, Jason C Levine, and Brian J Hall. Problematic smartphone use: A conceptual overview and systematic review of relations with anxiety and depression psychopathology. *Journal of affective disorders*, 207:251–259, 2017.

[33] Niklas Elmqvist, Pierre Dragicevic, and Jean-Daniel Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE transactions on Visualization and Computer Graphics*, 14(6):1539–1148, 2008.

[34] Mateus Espadoto, Rafael M Martins, Andreas Kerren, Nina ST Hirata, and Alexandru C Telea. Toward a quantitative survey of dimension reduction techniques. *IEEE transactions on visualization and computer graphics*, 27(3):2153–2173, 2019.

[35] Yujie Fang, Hui Xu, and Jie Jiang. A survey of time series data visualization research. In *IOP Conference Series: Materials Science and Engineering*, volume 782, page 022013. IOP Publishing, 2020.

[36] Mark A Ferro, Jan Willem Gorter, and Michael H Boyle. Trajectories of depressive symptoms during the transition to young adulthood: the role of chronic illness. *Journal of Affective Disorders*, 174:594–601, 2015.

[37] Daniel E Ford and Douglas B Kamerow. Epidemiologic study of sleep disturbances and psychiatric disorders: an opportunity for prevention? *Journal of the American Medical Association*, 262(11):1479–1484, 1989.

[38] Jerome H Friedman and Werner Stuetzle. Projection pursuit regression. *Journal of the American statistical Association*, 76(376):817–823, 1981.

[39] Keinosuke Fukunaga. *Introduction to statistical pattern recognition*. Elsevier, 2013.

[40] Katholiki Georgiades, Laura Duncan, Li Wang, Jinette Comeau, Michael H Boyle, and 2014 Ontario Child Health Study Team. Six-month prevalence of mental disorders and service contacts among children and youth in ontario: evidence from the 2014 ontario child health study. *The Canadian Journal of Psychiatry*, 64(4):246–255, 2019.

[41] Golda S Ginsburg, Emily M Becker, Courtney P Keeton, Dara Sakolsky, John Piacentini, Anne Marie Albano, Scott N Compton, Satish Iyengar, Kevin Sullivan, Nicole Caporino, et al. Naturalistic follow-up of youths treated for pediatric anxiety disorders. *Journal of the American Medical Association psychiatry*, 71(3):310–318, 2014.

[42] Golda S Ginsburg, Emily M Becker-Haimes, Courtney Keeton, Philip C Kendall, Satish Iyengar, Dara Sakolsky, Anne Marie Albano, Tara Peris, Scott N Compton, and John Piacentini. Results from the child/adolescent anxiety multimodal extended long-term study (camels): primary anxiety outcomes. *Journal of the American Academy of Child & Adolescent Psychiatry*, 57(7):471–480, 2018.

[43] Tasha Glenn and Scott Monteith. New measures of mental state and behavior based on data collected from sensors, smartphones, and the internet. *Current psychiatry reports*, 16(12):1–10, 2014.

[44] Eduardo Graells and Alejandro Jaimes. Lin-spiration: using a mixture of spiral and linear visualization layouts to explore time series. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, pages 237–240, 2012.

[45] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

[46] Gladys M Hilasaca, Wilson E Marcílio-Jr, Danilo M Eler, Rafael M Martins, and Fernando V Paulovich. Overlap removal of dimensionality reduction scatterplot layouts. *arXiv preprint arXiv:1903.06262*, 2019.

[47] Patrick Hoffman, Georges Grinstein, and David Pinkney. Dimensional anchors: a graphic primitive for multidimensional multivariate information visualizations. In *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM internation conference on Information and knowledge management*, pages 9–16, 1999.

[48] Jeremy F Huckins, Alex W DaSilva, Weichen Wang, Elin Hedlund, Courtney Rogers, Subigya K Nepal, Jialing Wu, Mikio Obuchi, Eilis I Murphy, Meghan L Meyer, et al. Mental health and behavior of college students during the early phases of the covid-19 pandemic: Longitudinal smartphone and ecological momentary assessment study. *Journal of medical Internet research*, 22(6):e20185, 2020.

[49] Jean-François Im, Michael J McGuffin, and Rock Leung. Gplom: the generalized plot matrix for visualizing multidimensional multivariate data. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2606–2614, 2013.

[50] A Inselberg. Parallel coordinates: Visual multidimensional geometry and its applications. 233 spring street, new york, ny 10013, 2008.

[51] Alfred Inselberg. The plane with parallel coordinates. *The visual computer*, 1(2):69–91, 1985.

[52] Alfred Inselberg. Don't panic... just do it in parallel! *Computational Statistics*, 14(1):53–77, 1999.

[53] Alfred Inselberg and Bernard Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *Proceedings of the First IEEE Conference on Visualization: Visualization90*, pages 361–378. IEEE, 1990.

[54] Waqas Javed, Bryan McDonnel, and Niklas Elmqvist. Graphical perception of multiple time series. *IEEE transactions on visualization and computer graphics*, 16(6):927–934, 2010.

[55] Michael I Jordan and Tom M Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.

[56] Dietrich Kammer, Mandy Keck, Thomas Gründer, Alexander Maasch, Thomas Thom, Martin Kleinsteuber, and Rainer Groh. Glyphboard: Visual exploration of high-dimensional data combining glyphs with dimensionality reduction. *IEEE Transactions on Visualization and Computer Graphics*, 26(4):1661–1671, 2020.

[57] Alan E Kazdin. Annual research review: expanding mental health services through novel models of intervention delivery. *Journal of Child Psychology and Psychiatry*, 60(4):455–472, 2019.

[58] Daniel A Keim. Information visualization and visual data mining. *IEEE transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.

[59] Philip C Kendall and Jeremy S Peterman. Cbt for adolescents with anxiety: Mature yet still developing. *American Journal of Psychiatry*, 172(6):519–530, 2015.

[60] Josua Krause, Aritra Dasgupta, Jean-Daniel Fekete, and Enrico Bertini. Seekaview: An intelligent dimensionality reduction strategy for navigating high-dimensional data spaces. In *2016 IEEE 6th Symposium on Large Data Analysis and Visualization (LDAV)*, pages 11–19. IEEE, 2016.

[61] Kurt Kroenke, Robert L Spitzer, and Janet BW Williams. The phq-9: validity of a brief depression severity measure. *Journal of general internal medicine*, 16(9):606–613, 2001.

[62] Kurt Kroenke, Robert L Spitzer, Janet BW Williams, and Bernd Löwe. An ultra-brief screening scale for anxiety and depression: the phq–4. *Psychosomatics*, 50(6):613–621, 2009.

[63] Tania Lecomte, Stéphane Potvin, Marc Corbière, Stéphane Guay, Crystal Samson, Briana Cloutier, Audrey Francoeur, Antoine Pennou, Yasser Khazaal, et al. Mobile apps for mental health issues: meta-review of meta-analyses. *JMIR mHealth and uHealth*, 8(5):e17458, 2020.

[64] John A Lee and Michel Verleysen. *Nonlinear dimensionality reduction*, volume 1. Springer, 2007.

[65] Yu-Kang Lee, Chun-Tuan Chang, You Lin, and Zhao-Hong Cheng. The dark side of smartphone usage: Psychological traits, compulsive behavior and technostress. *Computers in human behavior*, 31:373–383, 2014.

[66] Dirk J Lehmann, Georgia Albuquerque, Martin Eisemann, Marcus Magnor, and Holger Theisel. Selecting coherent and relevant plots in large scatterplot matrices. In *Computer Graphics Forum*, volume 31, pages 1895–1908. Wiley Online Library, 2012.

[67] Jing Li, Jean-Bernard Martens, and Jarke J Van Wijk. Judging correlation from scatterplots and parallel coordinate plots. *Information Visualization*, 9(1):13–30, 2010.

[68] Jake Linardon, Pim Cuijpers, Per Carlbring, Mariel Messer, and Matthew Fuller-Tyszkiewicz. The efficacy of app-supported smartphone interventions for mental health problems: A meta-analysis of randomized controlled trials. *World Psychiatry*, 18(3):325–336, 2019.

[69] Lucy MacLeod, Banuchitra Suruliraj, Dominik Gall, Kitti Bessenyei, Sara Hamm, Isaac Romkey, Alexa Bagnell, Manuel Mattheisen, Viswanath Muthukumaraswamy, Rita Orji, et al. A mobile sensing app to monitor youth mental health: observational pilot study. *JMIR mHealth and uHealth*, 9(10):e20638, 2021.

[70] Tamara Munzner. *Visualization analysis and design*. CRC Press, 2014.

[71] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT Press, 2012.

[72] Dag Neckelmann, Arnstein Mykletun, and Alv A Dahl. Chronic insomnia as a risk factor for developing anxiety and depression. *Sleep*, 30(7):873–880, 2007.

[73] Luis Gustavo Nonato and Michael Aupetit. Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. *IEEE Transactions on Visualization and Computer Graphics*, 25(8):2650–2673, 2018.

[74] Maurice M Ohayon. Epidemiology of insomnia: what we know and what we still need to learn. *Sleep medicine reviews*, 6(2):97–111, 2002.

[75] Scott A Paluska and Thomas L Schwenk. Physical activity and mental health. *Sports medicine*, 29(3):167–180, 2000.

[76] George C Patton, Carolyn Coffey, Helena Romaniuk, Andrew Mackinnon, John B Carlin, Louisa Degenhardt, Craig A Olsson, and Paul Moran. The prognosis of common mental disorders in adolescents: a 14-year prospective cohort study. *The Lancet*, 383(9926):1404–1411, 2014.

[77] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901.

[78] Holly G Prigerson, Timothy H Monk, Charles F Reynolds III, Amy Begley, Patricia R Houck, Andrew J Bierhals, and David J Kupfer. Lifestyle regularity and activity level as protective factors against bereavement-related depression in late-life. *Depression*, 3(6):297–302, 1995.

[79] Mashfiqui Rabbi, Shahid Ali, Tanzeem Choudhury, and Ethan Berke. Passive and in-situ assessment of mental and physical well-being using mobile sensors. In *Proceedings of the 13th international conference on Ubiquitous computing*, pages 385–394, 2011.

[80] Yatharth Ranjan, Zulqarnain Rashid, Callum Stewart, Pauline Conde, Mark Begale, Denny Verbeeck, Sebastian Boettcher, Richard Dobson, Amos Folarin, RADAR-CNS Consortium, et al. Radar-base: open source mobile health platform for collecting, monitoring, and analyzing data using sensors, wearables, and mobile devices. *JMIR mHealth and uHealth*, 7(8):e11734, 2019.

[81] Darius A Rohani, Maria Faurholt-Jepsen, Lars Vedel Kessing, and Jakob E Bardram. Correlations between objective behavioral features collected from mobile and wearable devices and depressive mood symptoms in patients with affective disorders: systematic review. *JMIR mHealth and uHealth*, 6(8):e9691, 2018.

[82] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

[83] Sohrab Saeb, Mi Zhang, Christopher J Karr, Stephen M Schueller, Marya E Corden, Konrad P Kording, David C Mohr, et al. Mobile phone sensor correlates of depressive symptom severity in daily-life behavior: an exploratory study. *Journal of medical Internet research*, 17(7):e4273, 2015.

[84] John W Sammon. A nonlinear mapping for data structure analysis. *IEEE Transactions on computers*, 100(5):401–409, 1969.

[85] Saul Shiffman, Arthur A Stone, and Michael R Hufford. Ecological momentary assessment. *Annu. Rev. Clin. Psychol.*, 4:1–32, 2008.

[86] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *The craft of information visualization*, pages 364–371. Elsevier, 2003.

[87] Sônia Fernandes Silva and Tiziana Catarci. Visualization of linear time-oriented data: a survey. In *Proceedings of the first international conference on web information systems engineering*, volume 1, pages 310–319. IEEE, 2000.

[88] Charles Spearman. " general intelligence" objectively determined and measured. 1961.

[89] Clemens Stachl, Quay Au, Ramona Schoedel, Samuel D Gosling, Gabriella M Harari, Daniel Buschek, Sarah Theres Völkel, Tobias Schuwerk, Michelle Oldemeier, Theresa Ullmann, et al. Predicting personality from patterns of behavior collected with smartphones. *Proceedings of the National Academy of Sciences*, 117(30):17680–17687, 2020.

[90] Joshua B Tenenbaum, Vin de Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[91] Sara Thomée, Annika Härenstam, and Mats Hagberg. Mobile phone use and stress, sleep disturbances, and symptoms of depression among young adults-a prospective cohort study. *BioMed Central(BMC) Public Health*, 11(1):1–11, 2011.

[92] Warren S Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 17(4):401–419, 1952.

[93] Aditya Vaidyam, John Halamka, John Torous, et al. Enabling research and clinical use of patient-generated health data (the mindlamp platform): digital phenotyping study. *JMIR mHealth and uHealth*, 10(1):e30557, 2022.

[94] Julie Vallée, Emmanuelle Cadot, Christelle Roustit, Isabelle Parizot, and Pierre Chauvin. The role of daily mobility in mental health inequalities: the interactive influence of activity space and neighbourhood of residence on depression. *Social science & medicine*, 73(8):1133–1144, 2011.

[95] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(11), 2008.

[96] Laurens Van Der Maaten, Eric Postma, Jaap Van den Herik, et al. Dimensionality reduction: a comparative. *Journal of Machine Learning Research*, 10(66-71):13, 2009.

[97] Jarkko Venna, Jaakko Peltonen, Kristian Nybo, Helena Aidos, and Samuel Kaski. Information retrieval perspective to nonlinear dimensionality reduction for data visualization. *Journal of Machine Learning Research*, 11(2), 2010.

[98] Elio Ventocilla and Maria Riveiro. Visual growing neural gas for exploratory data analysis. In *14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, February 25-27, 2019, Prague, Czech Republic*, volume 3, pages 58–71. SciTePress, 2019.

[99] Elio Ventocilla and Maria Riveiro. A comparative user study of visualization techniques for cluster analysis of multidimensional data sets. *Information visualization*, 19(4):318–338, 2020.

[100] William H Walker, James C Walton, A Courtney DeVries, and Randy J Nelson. Circadian rhythm disruption and mental health. *Translational psychiatry*, 10(1):1–13, 2020.

[101] Shaun Wallace, Hua Guo, and David Sasson. Visualizing self-tracked mobile sensor and self-reflection data to help sleep clinicians infer patterns. In *proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 2194–2200, 2017.

[102] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T Campbell. Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 3–14, 2014.

[103] Matthew O Ward. A taxonomy of glyph placement strategies for multidimensional data visualization. *Information Visualization*, 1(3-4):194–210, 2002.

[104] Matthew O Ward, Georges Grinstein, and Daniel Keim. *Interactive data visualization: foundations, techniques, and applications*. AK Peters/CRC Press, 2010.

[105] Marc Weber, Marc Alexa, and Wolfgang Müller. Visualizing time-series on spirals. In *InfoVis*, volume 1, pages 7–14, 2001.

[106] Kilian Q Weinberger and Lawrence K Saul. An introduction to nonlinear dimensionality reduction by maximum variance unfolding. In *Association for the Advancement of Artificial Intelligence*, volume 6, pages 1683–1686, 2006.

[107] Alice Zheng and Amanda Casari. *Feature engineering for machine learning: principles and techniques for data scientists*. O'Reilly Media, Inc., 2018.

[108] Xinyu Zhou, Yuqing Zhang, Toshiaki A Furukawa, Pim Cuijpers, Juncai Pu, John R Weisz, Lining Yang, Sarah E Hetrick, Cinzia Del Giovane, David Cohen, et al. Different types and acceptability of psychotherapies for acute anxiety disorders in children and adolescents: a network meta-analysis. *Journal of the American Medical Association psychiatry*, 76(1):41–50, 2019.

Appendix A

Implementation Detail

As part of this thesis, we developed an interactive visualization system for analyzing data collected during PROSIT projects. Tech stack in web app development usually has the client-side (front-end) and server-side (back-end) parts. Because of the development flexibility that these front-end technologies provide, we decided to go ahead with the plan of creating a web application. HTML/CSS combined with JavaScript and d3.js [3] were suitable for building custom dashboard templates and visualizations needed for this thesis. We also needed a Back-End server that could serve the processing needs required for different custom visualizations. Flask [5] was the preferable choice for this purpose. Flask is lightweight and is often referred to as a microframework. It also has a shorter learning curve, making it the ideal fit to our tech stack. For all the preprocessing needs, we used the pandas [9] framework, a fast, powerful, flexible and easy-to-use open-source data analysis and manipulation tool built on top of the Python programming language. Table A.1 shows the list of technologies/frameworks/Programming_languages used in the implementation of our visual system.

SL No	Architecture	**Tech stack**	**Usage description**
1	front-end	HTML/CSS/JavaScript	front-end User Interface(UI) development
2		Bootstrap [1]	front-end input features
3		bootstrap-multiselect [2]	front-end dropdown feature
4		Jquery [7]	Event handling and client-server data connectivity
5		d3.js [3]	Custom charting/visualization
6	back-end	Python3 [11]	Data preprocessing and data manipulation at Back-end
7		pandas [9]	Data preprocessing and feature Engineering
8		scikit-learn [12]	Implementation of ML algorithms
9		Distance Grid(DGrid) [4]	Removal of overlap on DR glyphs scatter plot

Table A.1: Tech stack: list of technologies/frameworks/programming_languages used in development of visual system