NTNU
Norwegian University of
Science and Technology

Data Analysis for the Mobile Application of the selfBACK Decision Support System

Yu He

Embedded Computing Systems
Submission date: July 2018
Supervisor: Kerstin Bach, IDI

Norwegian University of Science and Technology
Department of Computer Science

Summary

The aim of the thesis is to find user behavior patterns by applying unsupervised learning methods on the SELFBACK app usage data. The recognized patterns will be used as references to select interviewees in the process evaluation. The focus of this thesis lies in what unsupervised learning methods can be applied on the given data, how to apply them and how to choose the best clustering results. Five clustering methods and four evaluation methods are explored in the thesis. For all clustering results, comparisons are made both in vertical and in horizontal to choose the best results. The optimal clustering results show that the behavior patterns for different types of data can be recognized in good quality. The experimental results are promising and can be used as direct references for the process evaluation.

Preface

This thesis is a result of work conducted in Department of Computer Science at Norwegian University of Science and Technology (NTNU) between February 2018 to July 2018.

I would like to thank my supervisor Associate Professor Kerstin Bach for her patient guidance and care during this period. Besides, I would like to thank Professor Kjetil Svarstad, the coordinator in NTNU of EMECS, for his help and encouragement during the time I have spent in NTNU. I would also like to thank all the staff of EMECS for giving me the opportunity to study further in Embedded Computing Systems. Last, I would like to thank my families and friends for their support and care.

Table of Contents

List of Tables

List of Figures

Abbreviations

AID	=	Activity Detection Algorithm
API	=	Application Programming Interface
ARBS	=	Advice Rule Based System
BIC	=	Bayesian Information Criterion
CEP	=	Complex Event Processor
CPU	=	Central Processing Unit
FCM	=	Fuzzy C-means
FPC	=	Fuzzy Partition Coefficient
GBD	=	Global Burden of Disease
GMM	=	Gaussian Mixture Models
GP	=	General Practitioner
GPAM	=	Generalized Population Admixture Model
GUI	=	Graphical User Interface
HMM	=	Hidden Markov Model
LBP	=	Low Back Pain
LOC	=	Lines of Code
MDA	=	Mobile Data Analytics
MECC	=	Mobile Edge Cloud Computing
OBD	=	On-Board Diagnostic
PAM	=	Population Admixture Model
PCA	=	Principle Component Analysis
QoE	=	Quality of Experiences
RCT	=	Randomized Control Trial
RDF	=	Resource Description Framework
RMDQ	=	Roland Morris Disability Questionnaire
RPM	=	Revolutions Per Minute
SMS	=	Short Message Service
SVM	=	Support Vector Machines
UI	=	User Interface
URL	=	Uniform Resource Locator
VCCQ	=	Virtual Patient Care Questionnaire
YLD	=	Years Lived with Disability

Chapter 1

Introduction

This chapter introduces the SELFBACK project in a few aspects: the background, the project description, and the task of the thesis. The main goal and research questions are also stated. The chapter ends with a structure explanation of the thesis.

1.1 Low Back Pain Introduction

Low back pain (LBP) is a common symptom occurring all over the world, which can be a result of different known or unknown diseases or abnormalities. Since the causes of LBP are difficult to be identified accurately, most cases are termed non-specific. LBP happens in all age groups, from children to elderly. People with higher risks of having LBP include those with physically demanding jobs, smokers, obese patients, and those with physical and mental comorbidities. Most episodes of LBP only last for a short time, but recurrence is common. Some people end up with persistent disabling pain due to many factors.

According to Hartvigsen et al. (2018), in the Global Burden of Disease (GBD) 2015 study of disease burdens for 315 causes in 195 countries, LBP caused approximately 60.1 million years lived with disability (YLD) in 2015, which was increased by 54% since 1990. Today, LBP is one of the most common diagnosis in primary care and a main cause for activity limitation, sick leave and physical disability.

A large amount of funds has been invested on the prevention and treatment of LBP. However, LBP is still a big problem in healthcare. Foster et al. (2018) stated that the biggest challenge is to reduce unnecessary treatments, stop harmful practices, and assuring effective and affordable healthcare. In order to alleviate the burdens on the world-wide healthcare system caused by LBP, cost-effective and context-specific strategies are needed.

1.2 SELFBACK Introduction

The SELFBACK project is funded by the European Unions Horizon 2020 research and focuses on promoting the self-management of patients with non-specific LBP (Bach et al., 2016a). It is a recommendation system built on case-based reasoning which provides personalized advice for patients with non-specific LBP and improves their physical functionality.

The SELFBACK system consists of three main parts: a wristband, a smart phone app and a data server (Bach et al., 2016b). The system overview is shown in Figure 1.1. A user firstly signs up on a web page and answers screening questions to initiate the case-based reasoning stage and generate the first exercise plan. Then the smart phone app collects objective and subjective data to refine the exercise plan. The objective data is the activity stream containing information about the users activity status like sleeping, sitting, walking, etc. It is collected by the smart phone app from the wristband continuously. The subjective data is collected by answering weekly questions by the user, including the level of pain, the degree of functionality, and so on. Both types of data are sent to the data server by the smart phone app. The objective data and subjective data are measured using separate metrics for case matching in the case-based reasoning stage. The system reuses the individual advice from the best matching case to generate a new weekly customized exercise plan for the user. After the new recommendations are produced, the data server sends them to the user by the smart phone app.

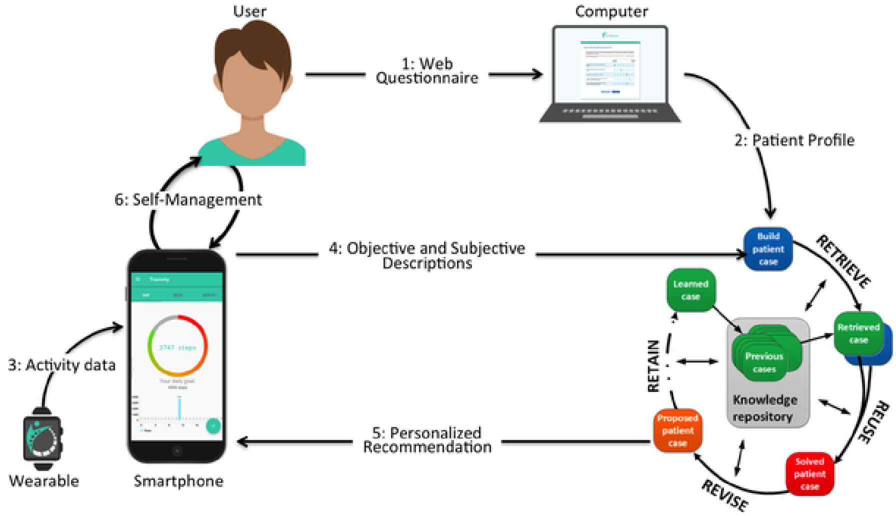


Figure 1.1: System Overview of SELFBACK (http://www.selfback.eu/about-the-project.html).

The total cost of SELFBACK for each user is estimated around 130 euros and users can start after a simple education. The SELFBACK project does not request the direct medical supervision from professional medical staffs, thus it can be applied to plenty of people easily. There are two main challenges for SELFBACK. One is to detect and identify the activity patterns based on the abstracted activity data, the other one is to find the optimal match of the case in the case base of existing case descriptions.

1.3 Project Description

During the randomized control trial (RCT) of the SELFBACK decision support system, participants are recruited from primary and secondary care units, such as general practitioners (GP), physiotherapists or chiropractors. Once participants found unsuitable enrolling in the SELFBACK approach, they are not considered further in the SELFBACK RCT. These participants can be seriously ill, pregnant, unable to do physical exercises or have had back surgery before.

The qualified participants are given a web questionnaire to document their basic information and starting status. The questionnaire has approximately 30 items. Then these participants are randomly divided into 2 groups by ratio 1:1. One group continues to use traditional treatment as the control group. The other group is given the SELFBACK app to try the new treatment as the intervention group. After 6 weeks period, the participants from both groups are given the same web questionnaire to get their latest status. Then both groups will continue the trial for the next 7 months. Between month 3-4 after signing up, an additional Virtual Patient Care Questionnaire (VCCQ) will be asked. The same web questionnaire will be asked again. VCCQ has 4 items and is to provide feedback for the SELFBACK app. The web questionnaire will be given to the participants of both groups every 3 months to get their periodical feedback.

The click-though app usage data contains large amounts of information about users. By analyzing the app usage data with machine learning methods, user behavior patterns can be recognized. The behavior patterns can help to understand how users interact with the SELFBACK app and to improve the quality of experiences (QoE) of the app. Combining with the feedback from participants about their conditions of low back pain in long term, the behavior patterns also contribute to make conclusions about if the SELFBACK app helps to relief low back pain and to what extent the SELFBACK app helps the participants.

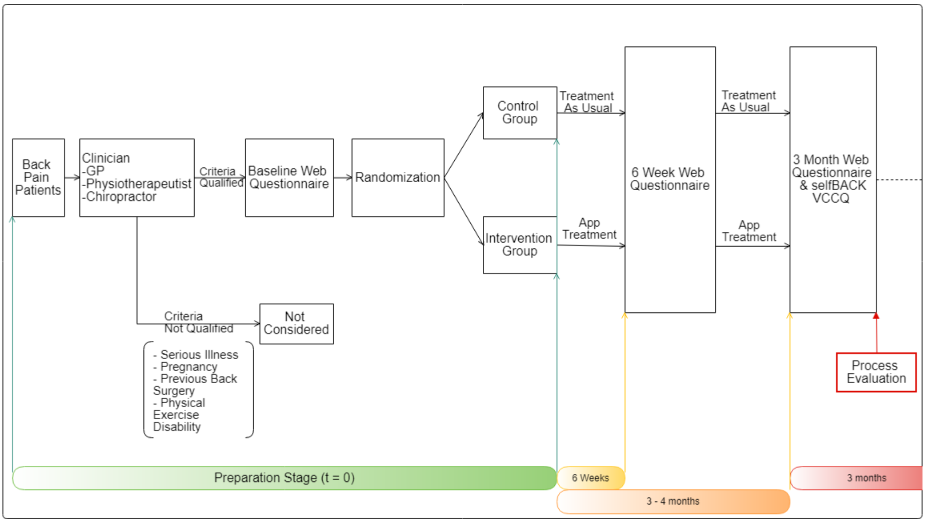


Figure 1.2: Diagram discussing the SELFBACK process for onboarding and following up on RCT participants.

The aim of this project is to use machine learning methods to analyze the click-through app usage data from the intervention group and separate the participants in the intervention group into clusters for the process evaluation after the first 3 months trial. During the process evaluation, only 5-7 participants (30% of all the participants) will be randomly selected for one cluster as representatives, according to the size of the cluster. The selected participants will participate in a face-to-face interview with a researcher to give feedback about using the SELFBACK app as a treatment tool. A diagram of the system is shown in Figure 1.2.

1.4 Main goal and Research Questions

As stated in project description, the main goal of this thesis is to find different usage patterns of the SELFBACK app by applying different machine learning methods on the app usage data. For each pattern, representative participants will be randomly selected for the interview in the process evaluation.

Based on the main goal, there are four research questions generated to guide the thesis, which will help to achieve the main goal. The research questions are shown below.

RQ1. What machine learning methods can be applied on the SELFBACK app usage data?

RQ2. How to apply those machine learning methods on the given data?

RQ3. How to choose the best clustering results among the results from different methods?

RQ4. How well do the applied machine learning methods perform on the given data?

1.5 Thesis Structure

The thesis is structured in 6 chapters. Chapter 1 introduces the background of the SELFBACK project and the task of the master thesis. Chapter 2 presents the systematic literature review process, including the review plan, the screening process and the results. Chapter 3 demonstrates the basic theories used in the project for data preprocessing and clustering. Chapter 4 describes the process of the experiments with results attached. Chapter 5 discusses the experimental results. Chapter 6 makes conclusions of the thesis and discusses future thoughts.

Chapter 2

Literature Review

This chapter presents the systematic literature review process of the thesis, which helps to find more relevant literature without personal bias. The goal of the literature review is to find the state of art of analytics of the usage of an app with unsupervised learning methods.

2.1 Literature Review Plan

The plan of the systematic literature review is based on the method demonstrated by Kofod-Petersen (2015) with adaptions made for the thesis. The process includes 4 steps: literature searching, literature removing, abstract screening and full text quality criteria screening. Both abstract screening and full text quality criteria screening are for the literature quality assessment, which is the final selection process.

2.1.1 Literature Searching

In literature searching, literature research questions, search terms and digital libraries used to conduct searching are decided.

Firstly, according to the task of the thesis, 4 literature research questions are proposed. The following steps are conducted in order to answer these 4 questions.

- What are the existing unsupervised learning solutions to the problem of analyzing the usage data of an app?
- How do the existing solutions differ from each other?
- What is the strength of the evidence in support of different solutions?
- What implications does each solution have?

Secondly, the search terms are defined. The search terms are key words of the task of the thesis and closely related to the first research question. The search terms are shown in Table 2.1 and Table 2.2. The search strings are formed by grouping these terms.

Mobile application	Usage	Analytics	Healtchare	User behavior
App				User profiling
Mobile app				

Table 2.1: Search Terms for App Analytics.

Unsupervised learning	Clustering	Dimensionality reduction	Healthcare

Table 2.2: Search Terms for Unsupervised Learning.

Four digital libraries are chosen to conduct the literature searching: IEEE Xplore, ACM Digital Library, SpringerLink, and ScienceDirect.

2.1.2 Literature Removing

Removal criteria are implemented for selecting primary literature. There are 4 criteria used in this step. The studies meeting these 4 conditions are eliminated from the research.

- Duplicates
- The same study published in different sources
- Studies published before a certain date, 2008
- Studies after certain page number, e.g. 4

2.1.3 Abstract Screening

The abstract screening is conducted by reading the abstract of each literature. 4 criteria are used to examine the abstracts.

- The study's main concern is relevant to the task of the thesis.
- The study is a primary study presenting empirical results.
- The study focuses on the constraints.
- The study describes an algorithm.

The first criterion is obligatory to be satisfied. The rest three are optional. The more criteria the literature meets, the more relevant the literature is.

2.1.4 Full Text Quality Criteria Screening

The full text quality criteria screening is conducted after the abstract screening. There are 10 questions to assess the quality of a literature.

Q1. Is there a clear statement of the aim of the research?

Q2. Does the study use the data from a mobile app?

Q3. Are algorithm decisions justified?

Q4. Is the test data set reproducible?

Q5. Is the experimental procedure thoroughly explained and reproducible?

Q6. Is there visualization in the paper?

Q7. Are there algorithm comparisons in the paper?

Q8. Are the performance metrics used in the study explained and justified?

Q9. Are the test results thoroughly analyzed?

Q10. Does the test evidence support the findings presented?

For each question, there are 4 options to choose: yes referring to 1 point, no referring to 0 point, partly referring to 0.5 points, and not applicable. The sum of points of the 10 answers are calculated and a threshold is set according to the statistical result of the answers. The literature with marks higher than the threshold will be finally selected.

2.2 Results

2.2.1 Search Results

The literature searching is conducted for two rounds. The first round is to get basic statistic information about the target literature. Thus, the first round is conducted by searching key terms only in titles. The second round is to supplement the results acquired from the first round. The searching range is bigger and key terms are searched in both titles and abstracts.

Round 1

In the first round, the key terms are searched only in titles. The statistic results of literature found about app analytics are shown in Table 2.3. The statistic results about unsupervised learning are shown in Table 2.4. And the statistic results about both app analytics and unsupervised learning are shown in Table 2.5.

	Search group	IEEE Xplore	ACM DL	ScienceDirect	SpringerLink
1	Mobile application & analytics	8	14	2	0
2	Mobile app & analytics	1	4	1	1
3	Mobile data & analytics	29	40	3	1
4	Mobile application & usage	22	33	8	6
5	Mobile app & usage	5	14	7	2
6	Mobile data & usage	14	12	5	0
7	Mobile application & usage & analytics	0	0	0	0
8	Mobile app & usage & analytics	0	0	0	0
9	Mobile data & usage & analytics	0	0	0	0
10	Mobile application & analytics & healthcare	1	1	0	0
11	Mobile app & analytics & healthcare	0	0	0	0
12	Mobile data & analytics & healthcare	0	0	1	0
13	Mobile application & analytics & user behavior	0	0	0	0
14	Mobile app & analytics & user behavior	0	0	0	0
15	Mobile data & analytics & user behavior	0	0	0	0
16	Mobile application & user behavior	8	9	4	0
17	Mobile app & user behavior	1	3	1	0
18	Mobile data & user behavior	9	1	2	0
19	Mobile application & user profiling	1	4	1	0
20	Mobile app & user profiling	0	1	1	0
21	Mobile data & user profiling	3	4	0	0
22	Mobile application & analytics & user profiling	0	1	0	0
23	Mobile app & analytics & user profiling	0	0	0	0
24	Mobile data & analytics & user profiling	0	0	0	0
25	Mobile application & analytics & healthcare & user behavior	0	0	0	0
26	Mobile app & analytics & healthcare & user behavior	0	0	0	0
27	Mobile data & analytics & healthcare & user behavior	0	0	0	0
28	Mobile application & analytics & healthcare & user profiling	0	0	0	0
29	Mobile app & analytics & healthcare & user profiling	0	0	0	0
30	Mobile data & analytics & healthcare & user profiling	0	0	0	0
	Sum = 289	102	141	36	10

Table 2.3: Statistic Search Results of App Analytics in Round 1.

	Search group	IEEE Xplore	ACM DL	ScienceDirect	SpringerLink
1	Unsupervised learning & healthcare	0	0	0	0
2	Clustering & healthcare	9	5	22	0
3	Dimensionality reduction & healthcare	0	0	0	0
4	Clustering & dimensionality reduction & healthcare	0	0	0	0
5	Dimensionality reduction & clustering	38	8	8	0
	Sum = 90	47	13	30	0

Table 2.4: Statistic Search Results of Unsupervised Learning in Round 1.

	Search group	IEEE Xplore	ACM DL	ScienceDirect	SpringerLink
1	Mobile application & user profiling/behavior & Unsupervised learning	0	0	0	0
2	Mobile application & user profiling/behavior & Clustering	0	0	0	0
3	Mobile application & user profiling/behavior & Dimensionality reduction	0	0	0	0
4	Mobile application & user profiling/behavior & Clustering & dimensionality reduction	0	0	0	0
5	Mobile application & analytics & Unsupervised learning	0	0	0	0
6	Mobile application & analytics & Clustering	0	0	0	0
7	Mobile application & analytics & Dimensionality reduction	0	0	0	0
8	Mobile application & analytics & Clustering & dimensionality reduction	0	0	0	0
9	Mobile application & analytics & user profiling & Clustering & dimensionality reduction	0	0	0	0
10	Mobile application & analytics & user behavior & Clustering & dimensionality reduction	0	0	0	0
	Sum = 0	0	0	0	0

Table 2.5: Statistic Search Results of both App Analytics and Unsupervised Learning in Round 1.

In Round 1, there are 289 results found for app analytics, 90 results for unsupervised learning, and 0 for both app analytics and unsupervised learning. Thus, there are altogether 379 results found, including 31 duplicates. Hence, 348 primary studies are found, which is not enough to avoid personal bias in the systematic literature review. As shown in the three tables above, for some search groups, there are 0 results. Thus, the literature searching is conducted for a second round.

Round 2

In the second round, the key terms are searched in both titles and abstracts. According to the results of Round 1, most studies found about unsupervised learning are not related to app analytics. Thus, unsupervised learning terms are not searched again separately in Round 2. The unsupervised learning part are searched together with the app analytics. The statistic results of app analytics in Round 2 are shown in Table 2.6. The statistic results of both app analytics and unsupervised learning are shown in Table 2.7.

	Search group	IEEE Xplore	ScienceDirect	SpringerLink	ACM DL
1	Mobile application & analytics	557	39	917	154
2	Mobile app & analytics	67	10	388	28
3	Mobile data & analytics	975	74	249	151
4	Mobile application & usage	2406	402	4794	379
5	Mobile app & usage	185	105	1105	153
6	Mobile data & usage	2515	466	1191	339
7	Mobile application & usage & analytics	42	5	515	18
8	Mobile app & usage & analytics	6	1	232	9
9	Mobile data & usage & analytics	68	9	143	18
10	Mobile application & analytics & healthcare	51	3	207	11
11	Mobile app & analytics & healthcare	4	2	92	4
12	Mobile data & analytics & healthcare	86	6	36	20
13	Mobile application & analytics & user behavior	35	5	102	10
14	Mobile app & analytics & user behavior	11	0	54	5
15	Mobile data & analytics & user behavior	80	8	39	19
16	Mobile application & user behavior	1209	231	875	264
17	Mobile app & user behavior	169	82	217	94
18	Mobile data & user behavior	1609	275	294	300
19	Mobile application & user profiling	543	95	1090	90
20	Mobile app & user profiling	59	14	202	31
21	Mobile data & user profiling	629	93	316	80
22	Mobile application & analytics & user profiling	15	2	92	4
23	Mobile app & analytics & user profiling	7	1	45	3
24	Mobile data & analytics & user profiling	31	4	27	7
25	Mobile application & analytics & healthcare & user behavior	3	0	21	1
26	Mobile app & analytics & healthcare & user behavior	406	0	10	1
27	Mobile data & analytics & healthcare & user behavior	5	0	6	2
28	Mobile application & analytics & healthcare & user profiling	1388	0	28	0
29	Mobile app & analytics & healthcare & user profiling	227	0	11	0
30	Mobile data & analytics & healthcare & user profiling	1	0	6	0
31	Mobile application & user patterns	1173	138	36	200
32	Mobile app & user patterns	118	24	5	66
33	Mobile data & user patterns	1646	166	17	259
	Sum = 34668 (6994)	16326	2260	13362	2720

Table 2.6: Statistic Search Results of App Analytics in Round 2.

	Search group	IEEE Xplore	ScienceDirect	SpringerLink	ACM DL
1	Mobile application & user behavior & Unsupervised learning	9	0	18	2
2	Mobile application & user profiling & Unsupervised learning	2	0	19	0
3	Mobile application & user behavior & Clustering	69	14	244	10
4	Mobile application & user profiling & Clustering	25	5	270	3
5	Mobile application & user profiling/behavior & Dimensionality reduction	3	0	11	0
6	Mobile application & user profiling/behavior & Clustering & dimensionality reduction	0	0	7	0
7	Mobile application & analytics & Unsupervised learning	9	0	7	0
8	Mobile application & analytics & Clustering	48	4	136	5
9	Mobile application & analytics & Dimensionality reduction	1	0	0	0
10	Mobile application & analytics & Clustering & dimensionality reduction	0	0	0	0
11	Mobile application & analytics & user profiling & Clustering & dimensionality reduction	0	0	0	0
12	Mobile application & analytics & user behavior & Clustering & dimensionality reduction	0	0	0	0
	Sum = 921 (571)	166	23	712	20

Table 2.7: Statistic Search Results of both App Analytics and Unsupervised Learning in Round 2.

For the search groups with over 100 results in one digital library, the results are refined by the year between 2008 to 2018 and sorted by relevance. Only the first 100 results in each digital library are reviewed. Books and chapters are excluded.

In Round 2, there are 34668 results found for app analytics and 921 results for both app analytics and unsupervised learning. Altogether, there are 35589 studies found including duplicates. And 7565 of them, duplicates included, are selected for the abstract screening. Among the 7565 results, 6994 results are for app analytics and 571 results are for both app analytics and unsupervised learning.

2.2.2 Abstract Screening Results

According to the abstract screening criteria, there are 31 papers selected. Since there are 7 papers only related to unsupervised learning, they are removed from the list. For the rest 24 papers in the list, key introductions are written according to the abstracts, shown in Table 2.8.

Table 2.8: Key Introductions for 24 Papers

Paper ID	Title	Key introduction
P1	Mobile Data Analytics(Abolfazli and Lee., 2017)	Discussion about the most recent advances in mobile data analytics.
P2	Execution Models for Mobile Data Analytics(ur Rehman et al., 2017)	Discussion about various options for execution models design in mobile data analytics and related challenges.
P3	Still in flow long-term usage of an activity motivating app for seniors(Lins et al., 2016)	Usage statistics of a mobile application (app) for seniors that encourages physical and mental activity of 82 users for about two years were processed and show that the active elderly users can be clustered in two groups with either increasing or decreasing and very little constant activity.

Table 2.8 continued from previous page

P4	SAMOA – A Visual Software Analytics Platform for Mobile Applications(Minelli and Lanza., 2013)	A platform to analyze mobile application data.
P5	RECKON: an analytics framework for app developers(Parate et al., 2016)	A framework to identify and extracts task-level information from an unlabeled data stream of user actions for mobile app analytics.
P6	AppFunnel: a framework for usage-centric evaluation of recommender systems that suggest mobile applications(Böhmer et al., 2013)	Analyze user's app preference by analyzing different stages of application engagement.
P7	A Cloud-Based Mobile Data Analytics Framework: Case Study of Activity Recognition Using Smartphone(Yuan and Herbert., 2014)	Use supervised learning and unsupervised learning to analyze mobile data and produce models which can be used to identify user activities like walking or running.
P8	A framework to support educational decision making in mobile learning(Fulantelli et al., 2015)	A task-interaction framework based on the relationships between the different types of interactions occurring in a mobile learning activity and the tasks which are pedagogically relevant for the learning activity.
P9	An unsupervised framework for sensing individual and cluster behavior patterns from human mobile data(Zheng and Ni., 2012)	A probabilistic framework to identify behavior patterns and predict user activities by analyzing mobile data with unsupervised learning methods.
P10	A mobile application to support collection and analytics of real-time critical care data(Vankipuram et al., 2017)	An application to track activities during a trauma code and provide feedback.
P11	A habit mining approach for discovering similar mobile users(Ma et al., 2012)	An approach to discover similar mobile users by identifying behavior patterns with the raw context log data and Bayesian Matrix Factorization model.
P12	Big Data Analytics for User-Activity Analysis and User-Anomaly Detection in Mobile Wireless Network(Parwez et al., 2017)	Use K-means and Neural Network to detect and predict anomalous behavior in mobile wireless network by analyzing mobile data.

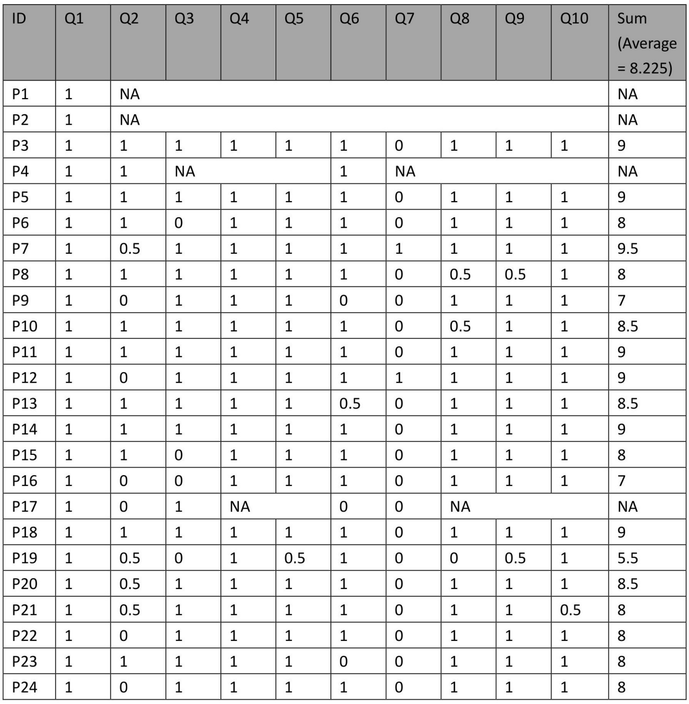
Table 2.8 continued from previous page

P13	Detection of Churned and Retained Users with Machine Learning Methods for Mobile Applications(Gener et al., 2014)	A study to find the different behavior patterns of churned and retained mobile application users using machine learning approach SVM with the data gathered from the users of a mobile application (iPhone & Android).
P14	Employing a data mining approach for identification of mobile opinion leaders and their content usage patterns in large telecommunications datasets(Chen et al., 2018)	Use clustering to identify web usage patterns of mobile opinion leaders from big data systems.
P15	Evaluating the usefulness of mobile services based on captured usage data from longitudinal field trials(Jensen and Larsen., 2007)	A framework to evaluate the mobile service usage by analyzing the usage data.
P16	Exploring the usage of a mobile phone application in transplanted patients to encourage medication compliance and education(Zanetti-Yabur et al., 2017)	A comparison trail between patients with the app and those without the app to show if the app helps the patients to recover.
P17	Managing diabetes: Pattern discovery and counselling supported by user data in a mobile platform(Machado et al., 2017)	An approach to analyze the data gathered from a diabetes mobile app to give individual advice by identifying behavior patterns.
P18	Modelling user behavior data in systems of engagement(Bent et al., 2017)	An architecture of modeling student behavior data, captured from different activities a student performs during the process of learning.
P19	MyHealthAvatar: A Lifetime Visual Analytics Companion for Citizen Well-being(Deng et al., 2016)	Key interactive visual analytics components in MyHealthAvatar to facilitate health and lifestyle data presentation and analysis, including 3D avatar, dashboard, diary, timeline, clockview and map to achieve flexible spatio-temporal lifestyle visual analysis to promote citizens' well-being.
P20	Passive profiling of mobile engaging behaviours via user-end application performance assessment(Chen et al., 2016)	Identify user behaviors by analyzing the user participation in mobile apps with the help pf the Hidden Markov Modelling to cluster.
P21	Sentiment-based User Profiles in Microblogging Platforms(Gutierrez and Poblete., 2015)	User profiling by sentiment behaviors of Twitter users using clustering.

Table 2.8 continued from previous page

P22	Service Usage Classification with Encrypted Internet Traffic in Mobile Messaging Apps(Fu et al., 2016)	A new system to classify service usages of mobile messaging apps by jointly modeling user behavioral patterns, network traffic characteristics, and temporal dependencies.
P23	Temporal Analytics for Software Usage Models(Andrei and Calder., 2017)	A new probabilistic model to analyze software usage with the parameters inferred from logged time series data of user-software interactions.
P24	Vehicular data acquisition and analytics system for real-time driver behavior monitoring and anomaly detection(Nirmali et al., 2017)	A vehicular data acquisition and analytics system for real-time driver behavior monitoring, anomaly detection, and alerting by identifying a driver's behavior using a Markov model and K-means clustering algorithm.

2.2.3 Full Text Quality Criteria Screening Results



ID	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Sum (Average = 8.225)
P1	1	NA									NA
P2	1	NA									NA
P3	1	1	1	1	1	1	0	1	1	1	9
P4	1	1	NA			1	NA				NA
P5	1	1	1	1	1	1	0	1	1	1	9
P6	1	1	0	1	1	1	0	1	1	1	8
P7	1	0.5	1	1	1	1	1	1	1	1	9.5
P8	1	1	1	1	1	1	0	0.5	0.5	1	8
P9	1	0	1	1	1	0	0	1	1	1	7
P10	1	1	1	1	1	1	0	0.5	1	1	8.5
P11	1	1	1	1	1	1	0	1	1	1	9
P12	1	0	1	1	1	1	1	1	1	1	9
P13	1	1	1	1	1	0.5	0	1	1	1	8.5
P14	1	1	1	1	1	1	0	1	1	1	9
P15	1	1	0	1	1	1	0	1	1	1	8
P16	1	0	0	1	1	1	0	1	1	1	7
P17	1	0	1	NA		0	0	NA			NA
P18	1	1	1	1	1	1	0	1	1	1	9
P19	1	0.5	0	1	0.5	1	0	0	0.5	1	5.5
P20	1	0.5	1	1	1	1	0	1	1	1	8.5
P21	1	0.5	1	1	1	1	0	1	1	0.5	8
P22	1	0	1	1	1	1	0	1	1	1	8
P23	1	1	1	1	1	0	0	1	1	1	8
P24	1	0	1	1	1	1	0	1	1	1	8

Table 2.9: Full Text Quality Criteria Screening Results.

The 24 papers listed in Table 2.8 are examined by the ten questions of full text quality criteria. The results are shown in Table 2.9. In the Table 2.9, papers are labeled with the same IDs in Table 2.8.

Most papers marked over 7 in the full text quality criteria screening. There are four papers (P1, P2, P4, P17) not applicable to the metrics, either because they are comparison papers about background theory or because they present studies without testing. They are kept in the literature list because they are closely related to the topic of the thesis. Paper 19 marked 5.5 which is much lower than the average mark 8.225 because it mainly describes the visualization options of the proposed application. The paper is kept in the literature list because visualization is a significant part in the thesis and the visualization options in the paper can be a good reference. Above all, all the 24 papers listed are kept for the detail review process.

2.2.4 Paper Grouping

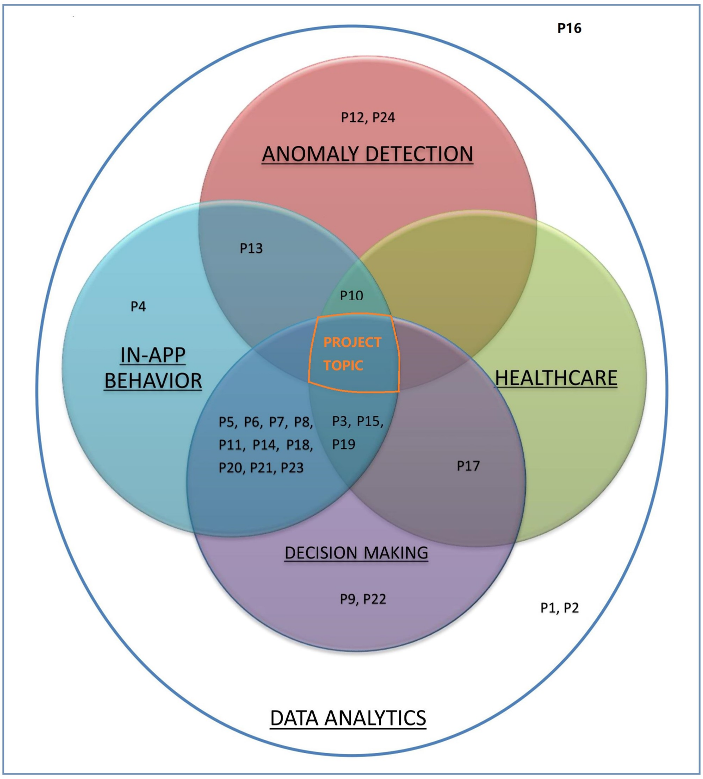
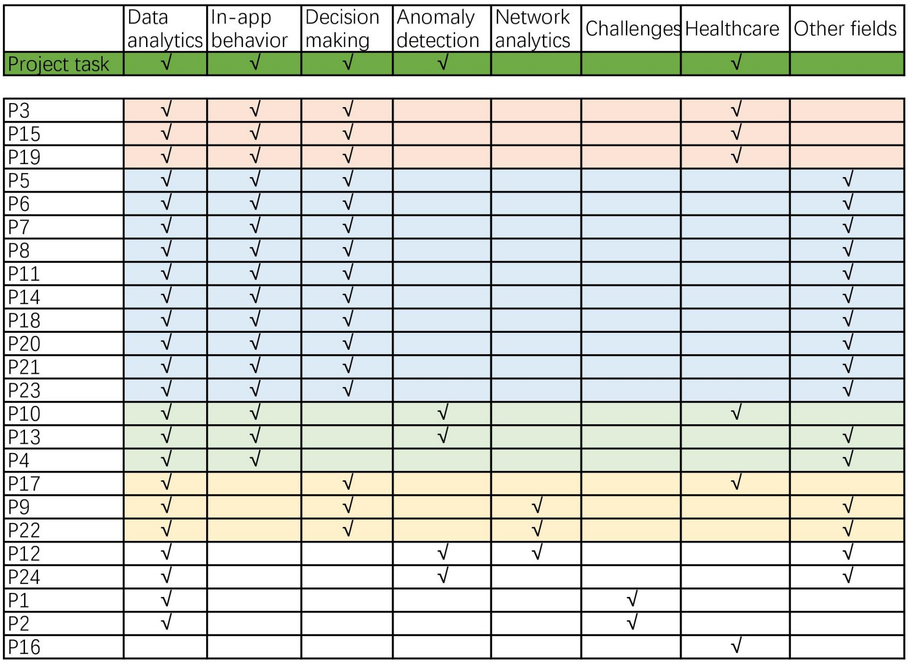


Figure 2.1: Paper Grouping Venn Diagram.

The listed 24 papers are grouped according to their topics. The paper grouping results are visualized in Venn diagram, shown in Figure 2.1.

According to the task of the thesis, there are five basic sets defined in the Venn diagram: data analytics, in-app behavior, anomaly detection, healthcare and decision making. The papers are categorized into different groups according to the five sets. The topic of this master thesis is the intersection part of all sets, which is marked in orange in the diagram. Detail information about the grouping process is shown in Table 2.10.



	Data analytics	In-app behavior	Decision making	Anomaly detection	Network analytics	Challenges	Healthcare	Other fields
Project task	√	√	√	√			√	
P3	√	√	√				√	
P15	√	√	√				√	
P19	√	√	√				√	
P5	√	√	√					√
P6	√	√	√					√
P7	√	√	√					√
P8	√	√	√					√
P11	√	√	√					√
P14	√	√	√					√
P18	√	√	√					√
P20	√	√	√					√
P21	√	√	√					√
P23	√	√	√					√
P10	√	√		√			√	
P13	√	√		√				√
P4	√	√						√
P17	√		√				√	
P9	√		√		√			√
P22	√		√		√			√
P12	√			√	√			√
P24	√			√				√
P1	√					√		
P2	√					√		
P16							√	

Table 2.10: Paper grouping Results.

As shown above, there are mainly 5 groups for all the papers, shown in different background colors. The groups are listed according to the relevance to the task of the thesis. Details about each group and each paper are demonstrated in the next part.

2.3 Literature

As stated above, the 24 papers can be categorized into five groups according to their key words and relevance to the task of the thesis.

2.3.1 Group 1: Data analytics of in-app behavior for decision making behavior in healthcare

The first group includes paper 3, paper 15 and paper 19, which contains four related key words. They present studies about analyzing in-app data to make decisions in healthcare.

In paper 3, Lins et al. (2016) analyzed the usage data of a mobile application on tablets. The application was designed to motivate the elderly to increase their physical and mental activities every day. The data was collected from retirement homes and assisted living facilities with 82 seniors aging from 63 to 96. The participants were asked to log their physical and mental activities with the application. Every touch on the application's GUI was logged with user ID, timestamp, position on the screen and a target link. The data analysis was done by the R statistics software. The data analysis was set in two steps. First, for each residence location, the daily average number of touch events of all users at this location was examined. Then using the K-means algorithm, the users were clustered according to touch events and usage projection. The results showed that the Concept of Flow can be applied in this situation. The seniors will decrease or even stop using the application when they think the tasks are either too difficult or too easy.

In paper 15, Jensen and Larsen. (2007) introduced a framework to evaluate the utility of mobile applications by analyzing the mobile usage data in longitudinal field trials. There are five main modules in the framework: capture, reporting, interpretation, analysis and output. The quantitative mobile data is captured and analyzed automatically on the mobile devices. The logging data captured includes when the application is started and stopped, UI events, screen transitions, frequency and duration spent in each screen, etc. The framework was tested on a mobile eHealth application named DiasNet Mobile for diabetes management. The interaction data of a diabetic user for three months was analyzed. The performance of main functionalities and the usage of the application were explored. The performance was evaluated by the duration of the user spent each time for every functionality. The usage of the application was evaluated at different granularities of time, such as a day, an hour and a session. Experiment results showed that new observations were found in the field trial which were missed in the lab and the evaluation of utility of DiasNet Mobile by analyzing usage data was proven to be gainful.

In paper 19, Deng et al. (2016) introduced a mobile application named MyHealthAvatar for collecting, aggregating, and visualizing life-logging data from mobile applications and wearable devices Fitbit, Withings and Moves. Fitbit and Withings are wearable devices which record steps, distance, calories, heart rate and so on. Moves records steps, locations, calories and distance automatically and uses the data to recognize activity types like walking, running, transport, etc. MyHealthAvatar uses dashboards, diaries, timelines, clock views and maps to provide spatio-temporal data analysis and visualization. The dashboard shows a user's latest health status in summary with significant notifications. The dairy is calendar-based to display daily data and events. The timeline includes five visualization styles to present time-varying data and events in a linear layout. The clock view shows the activities in one day in a radial layout. The map analyzes and visualizes the Moves data based on Google Maps. MyHealthAvatar also provides an integrated view LifeTracker to show integrated spatio-temporal visualization and analysis using diary, map and clock view. Experiment results showed that MyHealthAvatar can record, store and reuse the unified and structured individual health data in the long term successfully.

2.3.2 Group 2: Data analytics of in-app behavior for decision making behavior in various domains

Group 2 includes paper 5, paper 6, paper 7, paper 8, paper 11, paper 14, paper 18, paper 20, paper 21 and paper 23. They present studies about in-app data analytics for decision making in other fields.

In paper 5, Parate et al. (2016) introduced a framework Reckon for mobile application analytics about end-to-end user experience in completing tasks efficiently. Reckon uses an automated algorithm to identify and extract task-level information from unlabeled datastreams of user actions. Reckon outputs various useful metrics to the app developers to evaluate how easy it is to complete tasks with the app, like the time spent in completing a specific task, the frequency of a particular task being executed, the number of actions users take to complete a specific task, the abandonment rate of each screen along a particular execution path, and etc. Reckon can be applied in many use cases, such as UI evaluation, contextual crash analysis, task abandonment analysis, task popularity analysis and user categorization.

In paper 6, Böhmer et al. (2013) introduced a usage-centric evaluation framework AppFunnel. AppFunnel is used to evaluate mobile application recommender systems. In AppFunnel, there are four stages along user's interaction sequences in the conversion funnel after a mobile application was recommended: view, installation, direct usage and long-term usage. The framework uses conversion rates as evaluation metrics instead of click-through-rates and download statistics. The conversion rates represent the number of action sequences which users follow from one stage to another. Three conversion rates were analyzed in this paper: view to installation, installation to direct usage and installation to long-term usage. The framework was tested in wild and four recommender engines based on personalization and context awareness were tested. The results showed that context-aware engines led to higher long-term usage of installed applications while non-contextualized engines had better performance in direct usage.

In paper 7, Yuan and Herbert. (2014) introduced a framework to monitor and classify daily activities using both supervised learning and unsupervised learning methods with the help of cloud computing. 5 algorithms were studied in this paper: Support Vector Machine, Bayesian Network, Decision Tree, K-Nearest Neighbor and Neural Network. The data of each user from a smartphone and its cooperative sensor is firstly used to build a universal classification model with supervised learning methods. The universal model is downloaded to smartphones first to perform real-time activity analysis. In order to improve accuracy, the universal model is adapted using unsupervised learning methods on the cloud as more data gathered, which generates an adapted model for each individual. The adapted models are downloaded to smartphones to perform accurate real-time activity analysis. Cross-validation is used to determine which machine learning method performs best for a given dataset. The results showed that for each individual, the best unsupervised learning method differed. And the adapted models performed better than the universal model.

In paper 8, Fulantelli et al. (2015) introduced a task-interaction based framework for educational decision-making in mobile learning. Teachers assess and evaluate students according to the pedagogical tasks and the relationships between different interactions in a mobile learning activity. The learning experiences are classified according to 6 factors:

context, control, tools, communication, subject and objective. The framework was tested in 2 mobile learning scenarios which differed based on the school curriculum, learning objectives and pedagogical models. Students' activities were tacked by an RDF model which can use the meaning expressed by semantics in the relationships between concepts. Teachers can visualize students' activities according to the 6 factors in a dashboard and make educational decisions based on the participation levels of students in learning activities. The participation levels are monitored by mobile data indicators, like number of posted messages, number of downloaded documents, connection durations in time, and etc. Machine learning methods like clustering are used to aggregate students with similar behaviors. The framework has potentials in education-decision making in mobile learning, such as supporting teachers with highlighting the most important indicators for a specific scenario.

In paper 11, a new approach was introduced by Ma et al. (2012) to discover mobile users with similar habits by identifying behavior patterns. The data used is collected from users' mobile phones, including timestamps, profiles, cell IDs and interaction records like playing e-games. The approach takes two steps. First, raw context logs of each user are normalized by transforming location-based context data like cell IDs and user interaction records into more universal representations like home and work place. Then, a constraint-based Bayesian Matrix Factorization model is used to extract common habits from behavior patterns and transform behavior pattern vectors into common habit vectors named hyper behavior patterns in a denser space. The cosine distance is calculated to compare every two records to find the similarity degree and generate clusters with similar behaviors. Experiment results showed that the approach can reduce the sparseness of behavior pattern vector space and discover similar users according to their habits effectively.

In paper 14, Chen et al. (2018) introduced an approach to identify mobile opinion leaders and cluster them according to their web content usage patterns. The approach uses statistics, data mining and pattern recognition to analyze the dataset from a Taiwanese telecommunications company. Four characteristics were used in the test to identify opinion leaders: enduring involvement, exploratory behavior, innovation and mobile competence. The four characteristics refer to user attributes like tenure, data plan, mobile phone brands and mobile data usage. Users with higher levels of all these four characteristics are labeled as opinion leaders. The identified opinion leaders are clustered into categories based on their web content usage like information searching and social networking. The usage is calculated by review times of each content category. The resulted clusters include all-services user, e-news browser, e-video viewer, e-shopper and three combination clusters. Experiment results showed that the four main characteristics can be used to identify opinion leaders and seven usage patterns were found from opinion leaders' mobile data.

In paper 18, Bent et al. (2017). introduced an architecture to model user behaviors by analyzing the mobile behavior data captured from different activities in a learning process. The system includes client-side application design, event stream data capturing, cloud-enabled data management, analytics and visualization. The data is collected from three kinds of interaction events: application event, sentiment and contextual event, and sensor event. The data collected and analyzed includes timestamps, duration time, page numbers, locations, etc. Engagement is measured by the sum of events for a specific task in a session. The data is stored as JSON objects and processed using MapReduce functions on a cloud-enabled backend. Systems of engagements are evaluated by the 5Vs of the user interaction data: volume, velocity, variety, veracity and value. An experiment was conducted to assess the performance of the system in modeling student behaviors, the results of which showed that the architecture can model student behaviors effectively and can be applied on other applications.

In paper 20, Chen et al. (2016) introduced a framework to profile mobile engaging behaviors passively in order to characterize the dynamics of user participation in real contexts of mobile applications. The framework includes three main parts: a characterization of mobile traffic data and engaging behaviors from the view of end users, a profiling of user behavior dynamics in mobile usage participation and interactions, a Hidden Markov Model (HMM) based clustering to identify behavior patterns. In the mobile traffic analysis, the activity detection (AID) algorithm was used to store entity relationships and properties. Different sets of metrics for behaviors, perceived application performance and contextual factors were used to measure the user engaging behavior in context. Hellinger distance for probability distributions were used to define the distance between two HMMs in clustering. In the test, a mobile phone dataset and a campus Wi-Fi dataset were used. The data collected in the mobile phone dataset includes network packets, application status, locations and user touch actions. The data for the campus Wi-Fi dataset includes network packets, locations and user identities. The mobile phone dataset was used to evaluate the AID algorithm and the campus Wi-Fi dataset was used to profile mobile user engaging behaviors. Experiment results showed that user engaging behaviors were more sensitive to the application quality at less familiar locations and user engaging behaviors were mainly affected by the interactions of principle engaging states.

In paper 21, Gutierrez and Poblete. (2015) presented a research in identifying and characterizing user sentimental profiles in social media Twitter from users' published text contributions and timelines. There are four stages for identifying sentiment profiles: collecting public tweets from user timelines to build the dataset, preprocessing tweets and extracting sentiment polarity for each tweet, clustering users by their sentiment polarity distributions, characterizing user profiles. The activity level is defined to present the number of tweets published in a certain time session. In experiment, five groups were identified by the activity level. Based on the five levels obtained, K-means clustering and hierarchical clustering were implemented to cluster users in each level according to similar sentiment strength across every sentiment polarity axis. Research results showed that a large number of Twitter users can be clustered into nine profiles based on the sentiment strength and polarity. Tweet-level metrics like the number of retweets, hashtags, mentions and URLs were used to characterize the nine sentiment profiles. No statistically important strong correlation was found which can hold for all profiles. The proposed methodology can be used in two applications: link recommendation following sentiment and mood detection and intervention.

In paper 23, Andrei and Calder. (2017) introduced an approach to analyze temporal features for software usage. Two models of software usage are defined based on two latent variable Markov models: Population admixture model (PAM) and Generalized population admixture model (GPAM). The two models are admixtures of activity patterns and generated from interaction logging data over different time intervals. PAM and GPAM present different perspectives on usage behaviors because the transition probabilities between states in an activity pattern are different. Two sets of temporal logic properties are defined. Generic properties are for analyzing personal activity patterns and identifying main characteristics of patterns. GPAM-specific properties are for analyzing combined patterns and focusing on unusual results from the general analysis. An experiment was conducted using a mobile application AppTracker which monitors the opening and closing of other apps. It has four main functions: overall usage, last 7 days, select by period, and settings. The data collected from AppTracker includes information about users' devices, start and end usage data, and session lists. Experiment results showed that AppTracker had three main activity patterns: overall viewing, in-depth viewing and glancing. Besides, the analysis results are sensitive to the chosen time interval for the logged data. The experiment proved that the proposed approach was tractable and useful.

2.3.3 Group 3: In-app Data Analytics for Anomaly Detection in Healthcare or other fields

Group 3 includes paper 10, paper 13 and paper 4. They present studies about in-app data analytics for anomaly detection in healthcare or other fields. Paper 10 talks about anomaly detection in healthcare. Paper 13 talks about anomaly detection in other fields. Paper 4 shows in-app data analytics in other fields.

In paper 10, Vankipuram et al. (2017) introduced a web-based mobile application which can be used to track clinicians' activities in trauma codes, provide real-time deviations from guidelines and protocols, and provide feedback with decisions. The events are stored with timestamps and the trauma codes are represented as timelines. The deviations are divided into 4 types: error, innovation, proactive and reactive. If a task fails to follow the guidelines and protocols in time or in sequence, it will be checked against rules to get its deviation type. The application deploys a summary data viewer to visualize activities with decisions made and task classifications. The visualizations are interactive and can be modified according to individual needs. Experiment results showed that the application can be used in critical care environments to capture data and present it to show accurate reflections of work activities in real-time without significant workflow interruptions.

In paper 13, Gener et al. (2014) introduced an approach to detect churned and retained users of a mobile application to improve user experience design by identifying user patterns with Support Vector Machines (SVM). The mobile application data used in the approach includes user static information like device and operating system attributes, user action information like clicked buttons and opening or closing the application with timestamps and locations adhered, and user dynamic information like battery status. The data is preprocessed to provide appropriate input for SVM, such as outlier detection, noise detection, filtering and normalization. Six features are used in SVM to classify churned user dataset between the first usage and last usage: total usage time, maximum usage time in one session, maximum usage time in one day, number of sessions, number of days the application is used, and average daily usage time. The attributes related to time are counted by minutes. Experiment results showed that the churned users can be detected effectively with the proposed approach and the next usage time of a user can be predicted rather precisely.

In paper 4, Minelli and Lanza. (2013) introduced a platform SAMOA to analyze mobile applications and present the mined data with visualization techniques. SAMOA uses 3 factors for the analysis to understand the structure and evolution of apps: source code, usage of external libraries and historical data. Three different granularities are provided in SAMOA for visualization. The snapshot view shows a specific revision of an app with two main components: the central section and the ring. The central section presents the entire app with classes and lines of code (LOC). The ring shows the 3rd-party API calls made by the app. Stacked bar charts and line charts are used to present different evolutionary information, like LOC, 3rd-party calls and core elements. The evolution view depicts the evolution of an app over its entire history. The ecosystem view presents a few apps at the same time, using stacked bar charts or grid layouts. The visualizations provided by SAMOA are interactive. Users can choose different ways to display data freely.

2.3.4 Group 4: Network Data Analytics in Decision Making in Healthcare or other fields

Group 4 includes paper 17, paper 9 and paper 22. They present studies about network data analytics in decision making in healthcare or other fields. Paper 17 focuses on healthcare. Paper 9 and paper 22 talk about network data analytics in other fields.

In paper 17, Machado et al. (2017) proposed an approach to guide diabetic patients and analyze the data gathered to give them individual advice. The system offers users with generic advice at beginning and advises them more specifically later on with constant learning. Data mining methods association rules and Bayesian networks are used to discover usage patterns which can be transformed into particular contexts to advise users and predict crisis. The data is collected by a mobile application named MyDiabetes, which fosters users to register their daily data like when and where they inject how much insulins. MyDiabetes includes three components: user interface, database and inference. The data mining is done in the inference component by the Advice Rule Based System (ARBS). The ARBS uses three rules to decide the advice given to the user: system rules, advice query rules and medical rules. This approach hasn't been tested in field yet.

In paper 9, Zheng and Ni. (2012) introduced a probabilistic framework to learn users' daily behavior patterns to predict user activities from mass mobile data using unsupervised learning. The spatial and temporal attributes used in the framework are the cell tower IDs and time. Gaussian components and latent states are also used to create probabilistic models. A Bayesian network is constructed first to model single user's specific activity pattern. The two attributes reinforce each other in the learning process until convergence. The network is then extended to a multi-user model by learning a few typical behavior patterns from many users, which models the similarity and differences among users to cluster behavior patterns. The behavior patterns learnt by the framework is used to predict users' future locations based on time using standard inference techniques. Experiment results showed that the single-user model uncovered clear and meaningful daily behavior pattern for each user. It was also revealed that the multi-user model separated behavior patterns from mixed data successfully and overcame the sparsity problem.

In paper 22, Fu et al. (2016) introduced a system CUMMA to classify service usage types using encrypted Internet traffic data collected from mobile messaging apps. The types of service usage studied in the system include text, picture, audio note, stream video call, location sharing, short video, news feed and outlier. There are four modules in the system: traffic segmentation, traffic feature extraction, service usage prediction and outlier detection and handling. The collected traffic flows are segmented into sessions and dialogs hierarchically. The packet length features and time delay features are then extracted to train service usage classifiers using Random Forest, which can classify dialogs into single-type usage. The dialogs with mixed usages are detected as anomaly and segmented into sub-dialogs of single-type usage using clustering and their usage types are predicted using an HMM model. An experiment was conducted on real world data collected from WeChat and WhatsApp. In the experiment, correlation analysis, comparisons of different features and classifiers, robustness and efficiency check, app-level quality of experiences (QoE) were conducted. Experiment results showed that the proposed system performed well in scoring QoE and profiling user behaviors for in-app usage analytics.

2.3.5 Group 5: Others

Group 5 consists of the rest papers, which can be categorized as others, including paper 12, paper 24, paper 1, paper 2 and paper 16. Paper 12 talks about network data analytics for anomaly detection in other fields. Paper 24 talks about data analytics for anomaly detection in other fields. Paper 1 and paper 2 present the challenges in data analytics. Paper 16 shows relevant study in healthcare.

In paper 12, Parwez et al. (2017) introduced an anomaly detection approach in mobile network using clustering methods. The spatial and temporal information contained in the mobile data (call detail record) is used to analyze user activities. The data used contains square IDs, timestamps, inbound call activity, outbound call activity, inbound SMS activity and outbound SMS activity. The sum of those four activities is used in the approach. The unusual high total user activities are categorized into anomalies since they cause unusual high traffic demands. Using K-means clustering and hierarchical clustering, anomaly user activities are clustered and identified, which helps to identify regions of interest and take proper traffic allocation actions. In the test, the objects in the anomaly clusters were verified against the anomaly objects found through ground truth data. The results showed that there was a match, which meant clustering can be used for anomaly detection. A comparison experiment using anomalous data and anomaly-free data to train neural network-based model for anomaly prediction showed that training the neural network with anomaly-free data led to less mean squared error.

In paper 24, Nirmali et al. (2017) introduced a system to monitor driver behaviors and detect anomaly driving by analyzing vehicular data. The system includes an On-Board Diagnostic (OBD) unit, a mobile app and a Complex Event Processor (CEP). The vehicular data is collected by OBD and transferred to CEP by the mobile app. The data processing and analyzing happen in CEP. A Markov model and K-means clustering are used to identify each driver's driving pattern for anomaly detection. Adaboost algorithm is used to monitor safe driver behaviors. The vehicular data collected includes speed, RPM, acceleration, throttle position, fuel level and engine load. In the Markov models, if the ratio between the number of anomalous events and the number of events in 20 seconds is bigger than a threshold value, it will be identified as an abnormal behavior. In K-means clustering, clusters with least number of data items will be considered as abnormal clustera and a range will be calculated for these clusters to identify the new incoming data.

Adaboost models are generated to classify data as safe or unsafe according to the alpha value. The driver will be alerted by the mobile app if there is an abnormal pattern detected. The data analyzed data is visualized on the mobile app in graphs to show the entire driving pattern of a driver with the identified abnormal instances. Experiment results showed that the system achieved over 90% accuracy in different driving simulations.

In paper 1, Abolfazli and Lee. (2017) discussed the challenges of mobile data analytics (MDA). With the proliferation of mobile devices, many advanced machine learning algorithms have been leveraged to analyze real-time data and make decisions. There are a few key challenges of MDA. Processing deficiency is an important one since the collection, storage and processing of mobile data are limited by the CPU power, memory capacity and battery. The storage limitation is another significant challenge. The mobile data is high-dimensional and high-volume. Reducing data dimensions and shrinking data volume are necessary to realize MDA solutions because the I/O operations involved by data storage and retrieval are resource-intensive operations in computing. The dimensionality reduction algorithms and data compression or pruning techniques efficiently save the mobile storage at the cost of minimal information and accuracy loss. Besides, heterogeneity, seamless connectivity, privacy and security are also crucial.

In paper 2, ur Rehman et al. (2017) defined an MDA application as a seven-stage application-execution process. It also discussed different factors which affect the design and the performance of an MDA application in a mobile edge cloud computing (MECC) system and discussed their related challenges. The seven stages are defined as data acquisition, data adaptation, data preprocessing, data fusion, data mining, knowledge integration, and knowledge management and visualization. The output of each preceding stage is the input of its later stage. Considering heterogeneity, the design and complexity of execution models are affected by several factors, like model types, data processing granularity, data management, and data transfer. For model type, training learning models in resource-constrained mobile devices is a big challenge. For data processing granularity, mobile devices need computational support from other devices and systems for heavyweight processing in distributed models. For Data management, resource consumption is increased by the onboard data storage. For data transfer, persistent Internet connection is required.

In paper 16, Zanetti-Yabur et al. (2017) discussed a comparison trial in which transplanted patients were given a mobile application to encourage their medication compliance and education. The mobile application named Transplant Hero aims at assisting transplant recipients with taking their medicines and also educating them. The efficacy of using the mobile application to promote medication adherence was compared with a group without using the application. The trial was examined by three questionnaire and documentations about patients' serum tacrolimus and creatinine levels. Results showed that patients using the mobile application can remember more about their immunosuppression regimen and they had less negative views about medication in the early peri-operative period. The trial showed that using mobile applications as a tool in the transplant patients' healthcare can potentially decrease non-adherence and foster patients' education.

2.4 Limitations and Conclusions

Considering the large amount of primary results found during the literature search, a few selection criteria are implemented, which may cause the selection results to be incomplete.

Refining the results by year 2008 to 2018 has nearly no effect on the results because the topic is new (the first iPhone was released in 2007) and relevant studies were published in recent years. However, for each search group, only the first 100 results in every digital library are considered, which may have led to omissions in the literature searching stage. The abstract screening criteria may also evict some relevant papers if the abstracts of those papers are not well written.

The 24 papers selected are closely related to the core of the thesis and all of them answer the four literature research questions well. Even if there may be some papers omitted, they can form the background base of the thesis in good quality and quantity. Thus, the systematic literature review can be marked as complete.

In the studies of in-app data analytics, the data analyzed includes start and end usage time, session lists, duration time in each screen, screen transitions, touch events, etc. Those data types can be a good reference for data selection and extraction in the thesis.

Of all the studies of pattern recognition, 8 machine learning methods were successfully used: K-means clustering, Random Forest, hierarchical clustering, Bayesian Network, Neural Network, Support Vector Machine, K-Nearest Neighbor and Decision Tree. Markov models were used in some cases to help clustering. For data analytics in healthcare, K-means clustering and Bayesian Network were used. For in-app data analytics, K-means clustering, Bayesian Network, hierarchical clustering and Random Forest were used. Since the data in the thesis are unlabeled, unsupervised learning methods are needed. Thus, K-means clustering, hierarchical clustering and Neural Network, these 3 methods will be first considered in the thesis. The rest methods will also be referenced.

Chapter 3

Basic Theory

This chapter introduces the data preprocessing methods, the clustering methods and the evaluation methods used in each clustering method.

3.1 Data preprocessing methods

3.1.1 Principle Component Analysis (PCA)

PCA is a statistical procedure for analyzing and simplifying datasets, first proposed by Pearson (2010). PCA is often used to reduce the dimensionality of datasets while maintaining the features contributing the most to the variance of the dataset. This is done by retaining the low-order principal components and ignoring the higher-order principal components, because low-order components often retain the most important aspects of the data. The method is mainly about the feature decomposition of the covariance matrix to obtain the main components of the data (i.e., feature vectors) and their weights (i.e., the eigenvalues). PCA is used in this thesis for dimensionality reduction.

3.1.2 Feature scaling

For some datasets, the range of variables can vary widely, which results in improper working of machine learning algorithms. In data preprocessing, feature scaling can be used to normalize the values of a dataset in order to eliminate the effects of some gross influences. Normally, the range of variables after feature scaling is [0,1] or [-1,1]. The formula of feature scaling is given in (3.1). In the equation, x is an original value and x' is the corresponding value after normalization.

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{3.1}$$

3.2 Clustering methods

Based on the conclusion of Chapter 2, K-means clustering, hierarchical clustering and Neural Network should be first considered to apply. Considering that the sizes of the data sets in this thesis are small, only K-means clustering and hierarchical clustering are suitable to apply on the data sets. In order to eliminate the limitations of these two methods, three more clustering methods are applied on the data sets. They are K-medoids clustering, Fuzzy C-means clustering and Gaussian Mixture Models clustering. Thus, there are five clustering methods applied in this thesis.

3.2.1 K-means clustering

The principle of K-means clustering was first proposed by Steinhaus (1957). Hartigan (1975) described the K-means clustering algorithm in detail. Hartigan and J. A. (1979) proposed a more efficient version of K-means clustering algorithm. For n data points, the K-means clustering algorithm partitions each data point to the cluster whose centroid has the shortest mean Euclidean distance to the data point, which forms clusters. A centroid locates at the mean coordinates of all data points in one cluster. In each iteration, the centroid of each cluster is recalculated and data points are reassigned to different clusters according to the distance, until the clusters stop changing. The result of K-means clustering algorithm depends on the number of clusters and the initial centroids, which can be determined by users.

3.2.2 K-medoids clustering

The principle of K-medoids clustering was first proposed by Kaufman and Rousseeuw (1987). K-medoids is a clustering method similar to K-means. In K-medoids, a sum of pairwise dissimilarities is calculated instead of a sum of squared Euclidean distances in K-means. A centroid is a real data point in the cluster called medoid, whose average dissimilarity to other data points in the same cluster is the smallest. After the initial medoids are selected, in each iteration, data points are assigned to the cluster of the medoid who has the minimal pairwise dissimilarity. New medoids are calculated after the clusters are formed. The result of K-medoids clustering algorithm depends on the number of clusters and the initial medoids. In the implementation, the initial medoids are chosen from the dataset randomly and only the best clustering results are considered.

3.2.3 Agglomerative clustering

Agglomerative clustering is a type of hierarchical clustering (Hastie et al., 2009). Agglomerative clustering starts from the bottom. At bottom, each data point is a cluster. In each iteration, the two clusters with the smallest intergroup dissimilarity are merged into one cluster. The intergroup dissimilarity between two groups is calculated by the distance of two data points from the two groups. There are three measures of the intergroup dissimilarity. *Single linkage* calculates the intergroup dissimilarity by the minimal distance between two data points from two groups. *Complete linkage* computes the intergroup dissimilarity by the maximal distance between two points from two groups. *Group average* uses the average distance between two data points from two groups. Dendrograms are used in agglomerative clustering to show the entire hierarchy. A dendrogram is a binary tree. The leaves on the bottom represent each data point. The height of each node depends on the intergroup dissimilarity between its two daughter nodes. The result of Agglomerative clustering depends on the dissimilarity measure chosen by users. In the implementation of the thesis, *Group average* is used as the measure of intergroup dissimilarities.

3.2.4 Fuzzy C-means (FCM) clustering

Fuzzy C-means clustering is proposed by Dunn (1973) and improved by Bezdek and C. (1981). It is based on K-means clustering. Different from K-means clustering, in FCM, each data point is allowed to belong to more than one cluster. It is a ”soft” clustering algorithm comparing to K-means clustering. The clustering result depends on the probabilities of each data point belonging to each cluster. At beginning, random probabilities are assigned to data points. In each iteration, centroids of clusters are calculated by the mean probability of all data points in the cluster. For each data point, probabilities of belonging to each cluster are calculated. Iterations stop when the change of probabilities between two iterations is smaller than a given threshold. A data point belongs to the cluster with the highest probability.

3.2.5 Gaussian Mixture Models (GMM) clustering

Similar to FCM, GMM is also a ”soft” clustering algorithm. A Gaussian mixture model is a probabilistic model (Yang and Ahuja, 1998). In GMM, the dataset is assumed to obey the Mixture Gaussian Distribution. In principle, by increasing the number of models, we can approximate any continuous probability density distribution arbitrarily. Each GMM is consist of k Gaussian distributions, named as components. The k components add together linearly to form the probability density function of GMM. The process of GMM clustering is the process of training k Gaussian distributions, which also refer to k clusters. In iterations, probabilities of each data point generated by each component are calculated. With the probabilities, parameters determining the probabilities of generating the data points from a component are calculated. A Likelihood function is then computed which is the product of the parameters. The iteration stops when the value of the Likelihood function converges. A data point belongs to the cluster with the highest probability. In implementation, the expectation-maximization (EM) algorithm is used to fit Gaussian mixture models.

3.3 Evaluation methods

3.3.1 Silhouette analysis

In Silhouette analysis, each cluster is represented by a silhouette according to its tightness and separation (Rousseeuw, 1987). The number of clusters (the value of k) is determined by the mean distance (the average silhouette score) between clusters. The value of the average silhouette score is between [-1,1]. Every data point will get a silhouette score after each clustering. If the score is 1, it means that the data point is far away from neighbor clusters, indicating a good clustering result. If the silhouette score is 0, it means that the data point may lie on the border of two clusters. If it is negative, the data point might be assigned to a wrong cluster. The average silhouette score is the mean score of all data points. The higher the average silhouette score, the better the clustering result.

Apart from the average silhouette score, the widths of silhouettes are also considered. The width of a silhouette is directly proportional to the number of data points in its corresponding cluster. Normally, its better to choose a k value so that all the silhouettes have similar widths.

The silhouette analysis is used in the K-means clustering and K-medoids clustering to choose the best number of clusters in this thesis. The PCA algorithm is applied on the data sets to make them 2-dimensional before silhouette analysis.

3.3.2 Elbow method

In the elbow method, the number of clusters starts from 2 and keeps increasing by 1. For each number, the cost of training the clusters is calculated. The cost drops dramatically at one point and decreases very slowly after that point. This point is the elbow of the cost curve and the corresponding number is the optimal number of clusters (Kodinariya and Dan Makwana, 2013).

The elbow method is used in the agglomerative clustering to choose the best number of clusters k in this thesis. The value of k is chosen by finding the clustering step where the acceleration of distance growth has the biggest value. The acceleration of distance growth is calculated by the 2nd derivative of the distances. Sort the distances by values. The point where the distance decreases the fastest (the strongest elbow) corresponds to the point where the acceleration of distance growth is the biggest. That point is the optimal value of k.

3.3.3 Fuzzy partition coefficient (FPC)

Partition coefficient was proposed by Dunn for fuzzy clustering methods as a measure of fuzziness (Trauwaert, 1988). Dunn defined the partition coefficient as the mean of the sum of all membership functions. FPC shows how clearly a dataset is described by a clustering model. In another word, FPC shows how far a fuzzy partition is from a hard solution, in which each data point is assigned to the cluster with the biggest probability. The range of the FPC value is between [1/k,1], and k is the number of clusters. If the value is 1, for each data point, the probability of belonging to one cluster is unity and probabilities of belonging to the rests are zero, indicating a hard clustering. The value 1/k indicates complete fuzziness. The higher the FPC values, the better the clustering.

FPC is used in the Fuzzy C-means clustering to choose the optimal number of clusters in this thesis.

3.3.4 Bayesian information criterion (BIC)

BIC was developed by Schwarz (1978) with a Bayesian argument for adopting it. In soft clustering methods, its possible to increase the probabilities by increasing the number of parameters. But it may lead to overfitting. BIC solves the problem by adding a penalty term for the number of parameters, which refers to the number of clusters in this thesis. BIC is an increasing function of the error variance of a model and also an increasing function of the number of parameters, indicating that dependent variables' unknown variations and the number of explanatory variables will increase the value of BIC. The lower the BIC, the fewer explanatory variables, the better fitting.

BIC is used in GMM clustering to choose the best number of clusters in this thesis.

Chapter 4

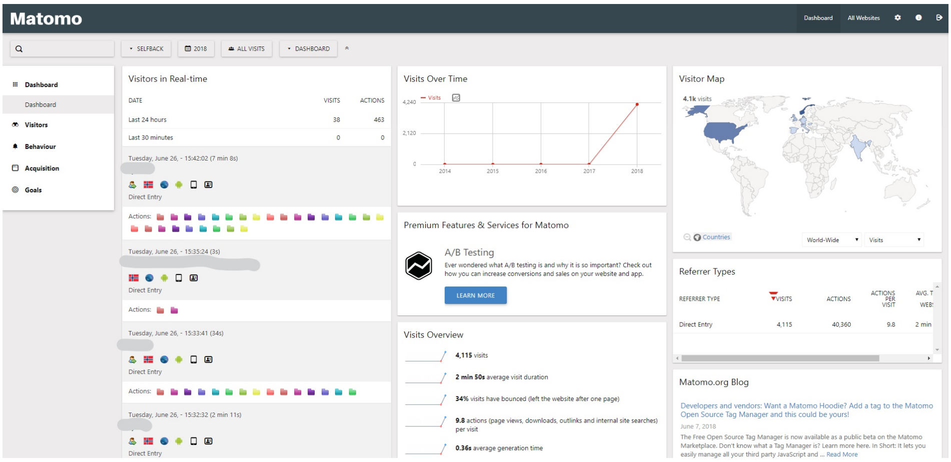
Methods and Experiments

This chapter introduces the data processed in the thesis and how the data is processed before and during clustering. The clustering results are also shown with analysis.

4.1 Data description

In this thesis, there are 6 types of data to be clustered. They are baseline data, activity data, education data, exercise data, user flow data and sessions data. The baseline data is about the basic information about users, like age, gender, time length of low back pain, etc. The activity data is about the steps users take every week. The education data is about the education tips users read every week. The exercise data is about the exercises users do every week. The user flow data is about how users visit app pages in sequence in each visiting session (i.e. user journey). The sessions data includes the user flow data and the time users spend on each page visited. The baseline data, activity data, education data and exercise data are retrieved from the back-end of *Matomo*. The user flow data and sessions data are retrieved from *Matomo* by API modules and methods defined in *Matomo*.

Matomo is an open analytics platform for web applications. It helps to gather and analyze important information about web application users. *Matomo* focuses on web analytics, ecommerce analytics, server log analytics, and intranet analytics. In this thesis, *Matomo* is used for web analytics. For web analytics, *Matomo* can track key performance indicators such as visits, goal conversion rates, downloads, keywords, and so on. It also shows how users engage on web applications. An example user interface of *Matomo* for the SELFBACK project is shown in Figure 4.1.



Visitors in Real-time

DATE	VISITS	ACTIONS
Last 24 hours	38	463
Last 30 minutes	0	0

Tuesday, June 26, - 15:42:02 (7 min 8s)

[redacted]

Direct Entry

Actions:

Tuesday, June 26, - 15:35:24 (3s)

[redacted]

Direct Entry

Actions:

Tuesday, June 26, - 15:33:41 (34s)

[redacted]

Direct Entry

Actions:

Tuesday, June 26, - 15:32:32 (2 min 11s)

[redacted]

Direct Entry

Premium Features & Services for Matomo

A/B Testing

Ever wondered what A/B testing is and why it is so important? Check out how you can increase conversions and sales on your website and app.

Visits Overview

4,115 visits

2 min 50s average visit duration

34% visits have bounced (left the website after one page)

9.8 actions (page views, downloads, outlinks and internal site searches) per visit

0.36s average generation time

Referrer Types

REFERRER TYPE	VISITS	ACTIONS	ACTIONS PER VISIT	AVG. T WEB
Direct Entry	4,115	40,360	9.8	2 min

Matomo.org Blog

Developers and vendors: Want a Matomo Hoodie? Add a tag to the Matomo Open Source Tag Manager and this could be yours!

June 7, 2018

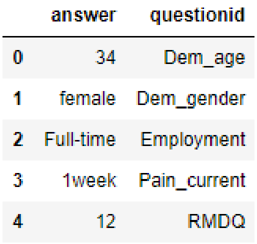
The Free Open Source Tag Manager is now available as a public beta on the Matomo Marketplace. Don't know what a Tag Manager is? Learn more here. In Short: It lets you easily manage all your third party JavaScript and ... Read More

Figure 4.1: Example UI of Matomo

4.2 Feature selection and extraction

4.2.1 Baseline data

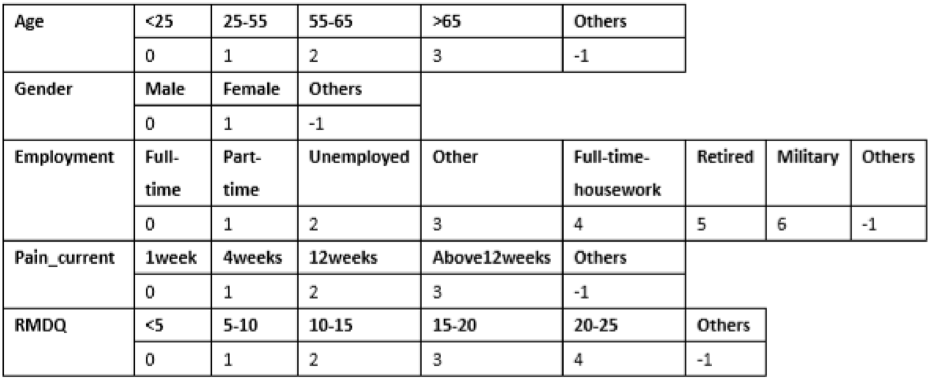
The baseline data includes *age*, *gender*, *employment*, *pain_current* and *RMDQ* (Roland Morris Disability Questionnaire) scores. An example of baseline data is shown in Table 4.1.



	answer	questionid
0	34	Dem_age
1	female	Dem_gender
2	Full-time	Employment
3	1week	Pain_current
4	12	RMDQ

Table 4.1: Baseline data example.

All the five features are used for clustering. The data of five features are transformed to integers according to different metrics shown in Table 4.2.

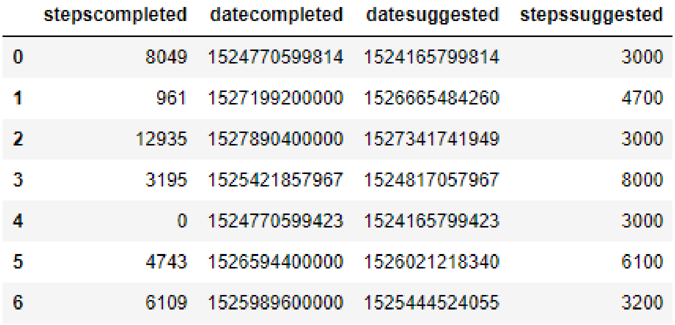


Age	<25	25-55	55-65	>65	Others			
	0	1	2	3	-1			
Gender	Male	Female	Others					
	0	1	-1					
Employment	Full-time	Part-time	Unemployed	Other	Full-time-housework	Retired	Military	Others
	0	1	2	3	4	5	6	-1
Pain_current	1week	4weeks	12weeks	Above12weeks	Others			
	0	1	2	3	-1			
RMDQ	<5	5-10	10-15	15-20	20-25	Others		
	0	1	2	3	4	-1		

Table 4.2: Baseline data feature extraction table.

4.2.2 Activity data

The activity data includes *completed steps*, *suggested steps*, *completed date*, and *suggested date*. An example of activity data is shown in Table 4.3.



	stepscompleted	datecompleted	datesuggested	stepssuggested
0	8049	1524770599814	1524165799814	3000
1	961	1527199200000	1526665484260	4700
2	12935	1527890400000	1527341741949	3000
3	3195	1525421857967	1524817057967	8000
4	0	1524770599423	1524165799423	3000
5	4743	1526594400000	1526021218340	6100
6	6109	1525989600000	1525444524055	3200

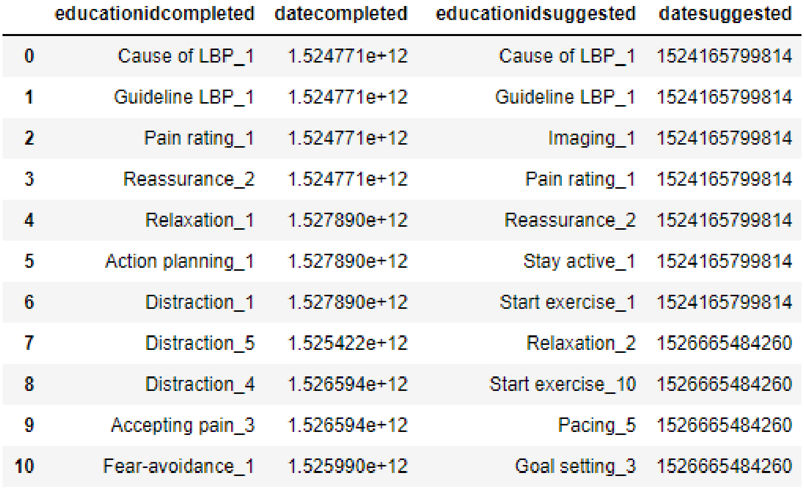
Table 4.3: Activity data example.

Completed date and *suggested date* are shown in timestamps. All the four columns are used to construct three features for clustering. The timestamps are transformed into dates before using.

- First, the average time difference in days between two activity weeks are calculated. Different activity weeks have different dates.
- Second, the complete ratio for each activity week is calculated by comparing the *completed steps* and *suggested steps* for each activity week.
- Third, the average steps for one activity week is calculated.

4.2.3 Education data

The education data includes *completed education IDs*, *completed date*, *suggested education IDs*, and *suggested date*. An example of education data is shown in Table 4.4.



	educationidcompleted	datecompleted	educationidsuggested	datesuggested
0	Cause of LBP_1	1.524771e+12	Cause of LBP_1	1524165799814
1	Guideline LBP_1	1.524771e+12	Guideline LBP_1	1524165799814
2	Pain rating_1	1.524771e+12	Imaging_1	1524165799814
3	Reassurance_2	1.524771e+12	Pain rating_1	1524165799814
4	Relaxation_1	1.527890e+12	Reassurance_2	1524165799814
5	Action planning_1	1.527890e+12	Stay active_1	1524165799814
6	Distraction_1	1.527890e+12	Start exercise_1	1524165799814
7	Distraction_5	1.525422e+12	Relaxation_2	1526665484260
8	Distraction_4	1.526594e+12	Start exercise_10	1526665484260
9	Accepting pain_3	1.526594e+12	Pacing_5	1526665484260
10	Fear-avoidance_1	1.525990e+12	Goal setting_3	1526665484260

Table 4.4: Education data example.

Completed date and *suggested date* are shown in timestamps. The timestamps are transformed into dates before using. The four columns are used to construct three features for clustering.

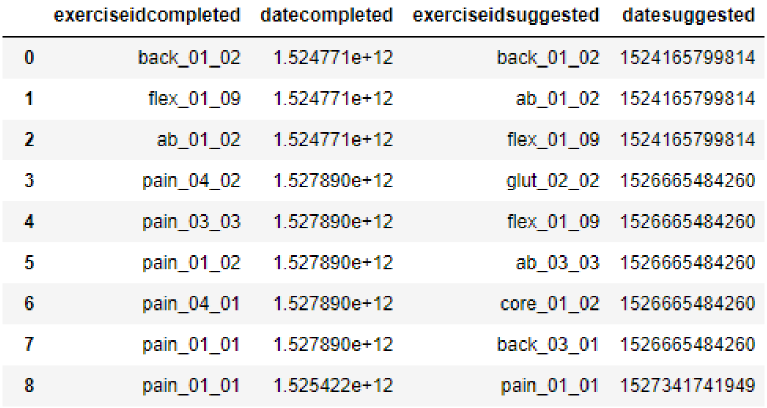
- First, the completed frequencies of all education IDs in all education weeks are put into a list initiated with 0s. Add 1 at the position corresponding to an education if it is visited. For example, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1] is a frequency list of education tips. In this week, this user read education tip 14 for twice, education tip 18, 21 and 26 for once. This feature shows the number of education tips visited and the visiting frequency of each education tip.

- Second, the completed education IDs are separated into weeks according to completed dates. The average number of education tips visited per education week is calculated.

- Third, the complete ratio for each education week is calculated by comparing the number of suggested education IDs and the number of completed education IDs in one education week.

4.2.4 Exercise data

Like the education data, the exercise data includes *completed exercise IDs*, *completed date*, *suggested exercise IDs*, and *suggested date*. An example of exercise data is shown in Table 4.5.



	exerciseidcompleted	datecompleted	exerciseidsuggested	datesuggested
0	back_01_02	1.524771e+12	back_01_02	1524165799814
1	flex_01_09	1.524771e+12	ab_01_02	1524165799814
2	ab_01_02	1.524771e+12	flex_01_09	1524165799814
3	pain_04_02	1.527890e+12	glut_02_02	1526665484260
4	pain_03_03	1.527890e+12	flex_01_09	1526665484260
5	pain_01_02	1.527890e+12	ab_03_03	1526665484260
6	pain_04_01	1.527890e+12	core_01_02	1526665484260
7	pain_01_01	1.527890e+12	back_03_01	1526665484260
8	pain_01_01	1.525422e+12	pain_01_01	1527341741949

Table 4.5: Exercise data example.

Like activity data and education data, *completed date* and *suggested date* are shown in timestamps, which are transformed into dates before using. The four columns are used to construct three features for clustering.

- First, the completed frequencies of all exercise IDs in all exercise weeks are put into a list initiated with 0s. Add 1 at the position corresponding to an exercise if it is visited. The frequency list of exercises is formed in the same way as that of education tips. This feature shows the number of exercises visited and the visiting frequency of each exercise.
- Second, the completed exercise IDs are separated into weeks according to completed dates. The average number of exercises visited per exercise week is calculated.
- Third, the complete ratio for each exercise week is calculated by comparing the number of suggested exercise IDs and the number of completed exercise IDs in one exercise week.

4.2.5 User flow data

To show the user flows of all users, from-to charts are used to construct user flow matrices for each session. Here, one session refers to one visit. Each user has 10 sessions. The users are clustered in 10 dimensions, which are the 10 sessions. There are mainly three steps in the feature selection and extraction.

- Get page list
  - Use method 'Actions.getPageUrls' to get page titles starting with '/selfback' and put them into a list.
  - Compare the page title list above and the ones visited by all visitors. Supplement the page title list with missing titles and create a full-page title list.
- Get visitor ID list

  - Use method ’UserId.getUsers’ to get visitor names and visitor IDs.
  - Sort the visitors by alphabets.

- Get each visitors 10 session logs and construct user flow matrix

  - Use method ’Live.getVisitorProfile’ and the visitor IDs in step 2 to get the visit details in last 10 sessions of all visitors.
  - For each session, get the URLs visited. Remove the prefix and suffix of each URL to be comparable with the ones in the page titles list.
  - For each URL above, find its index in the page title list. For each session, put indices in an index list. For each user, put index lists for 10 sessions into a matrix.
  - For each session, create a matrix initiated with all 0s. The matrix represents a from-to chart showing the user journey among pages in one session. The number of rows and columns of the matrix are the length of the page title list. For each index list, make every neighboring indices into a pair. Use index pairs and the sequence of two indices to fill in 1s in the matrix. If there has been a 1 at the corresponding position, add 1 to it, and so on. The matrix not only shows the user flow, but also shows the visiting frequency of each step in the flow.

    For example, the index of page ’/selfback.Login’ is 1 and the index of page ’/selfback.Plan’ is 5. If a user goes from page *Login* to page *Plan*, then fill in 1 at the position [1][5] in the matrix. If a user goes from *Plan* to *Login*, then fill in 1 at the position [5][1]. If there is only one index in the index list, fill in 1 at the position of [index][index], like [1][1] if only *Login* was visited.

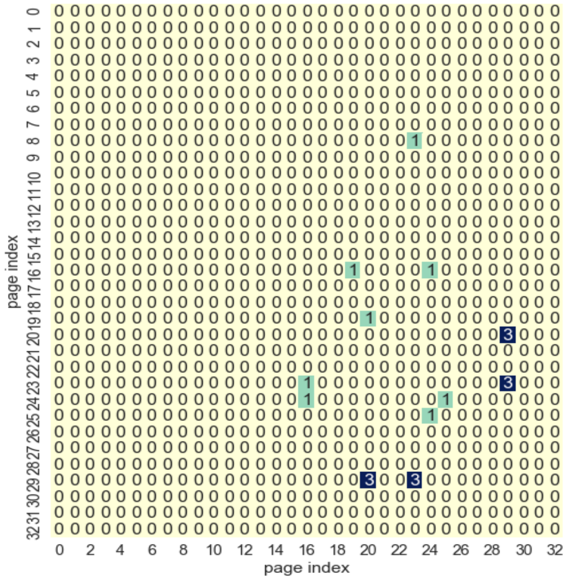


Figure 4.2: Example of from-to-matrix in user flow data

An example of the matrix created is shown in Figure 4.2. There are 33 pages in the SELFBACK app. All the pages are sorted by alphabets and indexed from 0 to 32. Take the position [29][20] as an example. 3 is filled at this position. It indicates that this user goes from page 29 to page 20 for three times in this session.

- For each user, put the 10 from-to matrices into a list. For users with less than 10 sessions, fill 0s in the from-to matrices for the empty sessions.
- Form a big list by putting the 10 from-to matrices of each user together.

4.2.6 Sessions data

The data for sessions clustering includes the URLs visited by users and the time spent on the URLs. They are used to construct two features which are the user flows on different pages and the time spent on each page in a visit. The user flows are represented by user flow matrices. The time spent is represented by a one-dimension list initiated with 0s.

- For the user flow matrix, the same method of constructing user flow matrices in user flow data is used here.
- For the time spent list, accumulate the time of each visit at the position corresponding to each page visited. [0, 0, 41, 0, 0, 114, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] is an example of time spent list. It indicates that this user visits page 2 for 41 seconds and page 5 for 114 seconds during this visit.

4.3 Dimensionality reduction

The dimensionality reduction method principal component analysis (PCA) is used before clustering on education data, exercise data, user flow data and sessions data. It is not applied on baseline data and activity data because there is only one number to represent one feature for each user in these two types of data.

- For the three features of education data, there is only a number to represent a user's average number of educations per week. But there is a list of visiting frequency for each education week of each user, which means the visiting frequency of each user is represented by a matrix. There is also a list of complete ratios for each user. To cluster education data in three dimensions which are the three features, PCA is used on the visiting frequency matrices and complete ratio lists to represent each user with one number to show one feature.
- For the three features of exercise data, like the education data, PCA is used on the visiting frequency matrices and complete ratio lists in order to use one number to represent one feature.
- For the user flow data, PCA is applied twice to cluster in 10 dimensions. First, PCA is applied on each user flow matrix to transform it into a one-dimension list. Then, PCA is applied on each users 10 sessions to represent each session with one number.

- For sessions data, PCA is applied twice on the user flow feature and once on the time spent feature to use one number to represent one feature.

4.4 Feature scaling

Feature scaling is applied on the sessions data after dimensionality reduction. The values of two features in sessions data vary widely, which affects the clustering result. Thus, the data is normalized into range [-1, 1] and [0, 1]. Feature scaling is also applied on activity data, education data and exercise data before visualization with heatmaps.

4.5 Clustering

The five clustering methods introduced in Chapter 3 are all applied on the six types of data. The four evaluation methods are used to choose the optimal number of clusters.

For baseline data, activity data and education data, there are 25 users available for clustering, which is less than the expectations during the RCT. According to the project description in Chapter 1, there will be 30% of users been interviewed. Assume the users clustered here represent the real distribution, there can be 7 clusters at maximum here, which is 30% out of 25. The cluster numbers 2, 3, 4, 5, and 6 are studied.

For exercise data, there are 23 users who are available to be clustered. Thus, the max number of clusters is 6. The cluster numbers 2, 3, 4, 5, and 6 are studied.

For user flow data, there are 28 users available to be studied in this thesis. For each user, since only last 10 sessions can be retrieved from *Matomo*, another 27 users are created by getting the last 10 sessions of the 28 users every 7 to 10 days in three weeks. Finally, there are 55 users to be clustered. The max number of clusters is 16. The cluster numbers 2 to 16 are studied.

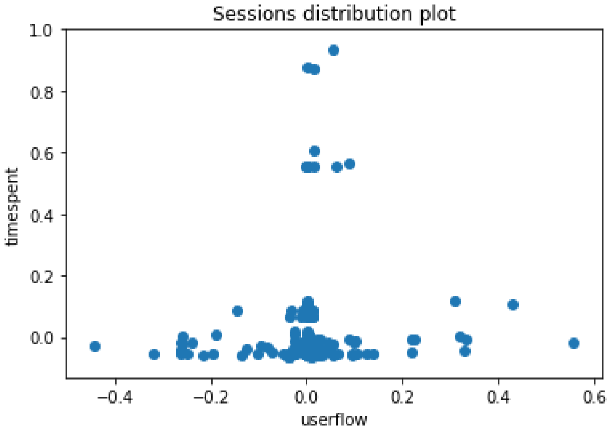


Figure 4.3: Sessions distribution plot

For sessions data, there are 223 sessions to be clustered. The scatter plot of all sessions is shown in Figure 4.3. Considering the distribution of the scatters, the cluster numbers 2, 3, 4, 5, 6, 7, 8 are studied.

The results of applying the evaluation methods are shown below. In order to eliminate the limitations of studying only one cluster number, the best three cluster numbers are chosen to be studied for each evaluation result.

4.5.1 Silhouette analysis

There are three standards used to choose the best three numbers of clusters. The Silhouette score should be as high as possible. The silhouette width should be even. There should be no negative Silhouette scores.

Baseline data

The silhouette analysis result of baseline data is shown in Table 4.6.

Clusters	Average Silhouette score	Silhouette width	Negative score
2	0.5785263659759827	Almost even	No
3	**0.6141317615066791**	**Almost even**	**No**
4	0.5529004391060902	Not even	No
5	0.5365212201783961	Not even	No
6	0.6388448162916582	Almost even	Yes

Table 4.6: Silhouette analysis result of baseline data

Considering the three standards together, 3 is the best choice. For 3 clusters, the average Silhouette score is high among all the average Silhouette scores. The widths of silhouettes are almost even. And there is no negative Silhouette score for all the data points. The other good options can be 2 clusters or 6 clusters.

Activity data

The silhouette analysis result of activity data is shown in Table 4.7.

Clusters	Average Silhouette score	Silhouette width	Negative score
2	0.6914302511730216	Almost even	No
3	**0.7276900119645395**	**Almost even**	**No**
4	0.5727971604510249	Not even	No
5	0.5808795120282223	Almost even	No
6	0.649032333636704	Almost even	No

Table 4.7: Silhouette analysis result of activity data

Based on the three standards, 3 is the best choice for the number of clusters. For 3 clusters, the average Silhouette score is the highest among all choices. The silhouette widths are similar and there is no negative Silhouette scores. The other good options can be 2 or 6.

Education data

The silhouette analysis result of education data is shown in Table 4.8.

Clusters	Average Silhouette score	Silhouette width	Negative score
2	**0.6474534573050242**	**Almost even**	**No**
3	0.595844634037881	Not even	No
4	0.5100350248722253	Almost even	Yes
5	0.5124499234373351	Not even	Yes
6	0.4968856261615948	Not even	Yes

Table 4.8: Silhouette analysis result of education data

Considering the three standards together, 2 is the optimal choice for the number of clusters. For 2 clusters, the average Silhouette score is the highest. The widths of silhouettes are almost even. There is no negative score for all data points. The other good choices can be 3 or 4.

Exercise data

The silhouette analysis result of exercise data is shown in Table 4.9.

Clusters	Average Silhouette score	Silhouette width	Negative score
2	**0.7757076658928596**	**Not even**	**No**
3	0.7044682129867152	Not even	No
4	0.5755608889624952	Not even	No
5	0.5337093424791152	Not even	No
6	0.526020713884879	Not even	No

Table 4.9: Silhouette analysis result of exercise data

According to the three standards, 2 is chosen as the optimal cluster number. For 2 clusters, the average Silhouette score is the highest and there is no negative Silhouette scores. Although the widths of silhouettes are not even, considering the silhouette widths are not even in all options, 2 is still the best. The other good options can be 3 or 4.

User flow data

Clusters	Average Silhouette score	Silhouette width	Negative score
2	0.47998839481129624	Not even	No
3	**0.534889787279326**	**Not even**	**No**
4	0.5400978274121135	Not even	Yes
5	0.45070359747709077	Not even	No
6	0.497484945260439	Not even	Yes
7	0.47806469341111324	Not even	Yes
8	0.4949192273048953	Not even	Yes
9	0.4051786668506182	Not even	Yes
10	0.3974549309605101	Not even	Yes
11	0.4137404323624616	Not even	No
12	0.36845056248794966	Almost even	Yes
13	0.37255525890299845	Almost even	Yes
14	0.3896879416146476	Almost even	Yes
15	0.38277412850130405	Almost even	No
16	0.3893645017872551	Amost even	No

Table 4.10: Silhouette analysis result of user flow data

The silhouette analysis result of user flow data is shown in Table 4.10.

Considering the three standards together, 3 is the best choice. The average Silhouette score is high among all options. There is no negative Silhouette scores. Although the widths of silhouettes are not even, 3 is still the best choice. The other good options can be 2 or 4.

Sessions data

The silhouette analysis result of sessions data is shown in Table 4.11.

Clusters	Average Silhouette score	Silhouette width	Negative score
2	**0.8409521752538408**	**Not even**	**No**
3	0.7144476845518968	Not even	No
4	0.7518729553878488	Not even	No
5	0.6811177698012335	Not even	No
6	0.6877503600532796	Not even	No
7	0.6986080461667229	Not even	Yes
8	0.6033714934443039	Not even	Yes

Table 4.11: Silhouette analysis result of sessions data

Based on the three standards, 2 is chosen to be the best number of clusters. The average Silhouette score of 2 clusters is the highest. There is no negative Silhouette scores. Although the widths of silhouettes are not even, they are not even in all other options either. The other good options can be 3 or 4.

4.5.2 Elbow method

As stated in Chapter 3, the number of clusters are chosen by finding the clustering step where the acceleration of distance growth is the biggest. Thus, the standard to choose the best three numbers of clusters is to choose the ”strongest elbow” of the distances, which is the biggest 2nd derivative distance.

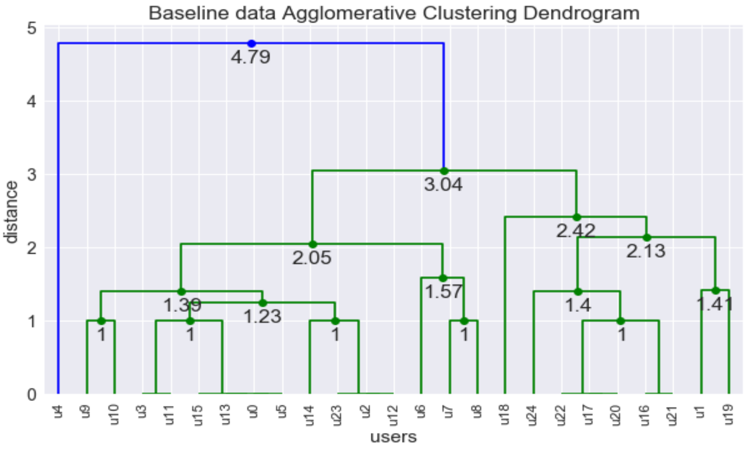


Figure 4.4: Dendrogram of baseline data

Figure 4.4 is a dendrogram example of baseline data in agglomerative clustering. In agglomerative clustering, each merging step reduces the number of clusters by 1. From the top of the dendrogram, the last k merges are the last k clusters. Since merging is based on distances between two nodes, the distances of the last k merges are used to choose the number of clusters.

Baseline data

The elbow method result of baseline data is shown in Figure 4.5. Although only 2 to 6 are studied here, the figure shows the results of 2 to 10, which are the last 10 merges. The results of 7 to 10 can be a reference.

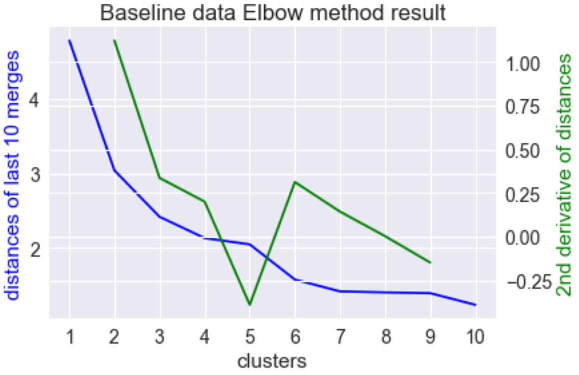


Figure 4.5: Elbow method result of baseline data

From the trend of the green line (the 2nd derivative of distances), we can see that there are a few big values. Thus, there are a few ”strong elbows” in the blue line (the distances).

The ”strongest elbow” appears at 2. According to the standard, 2 is chosen as the best number of clusters. There are ”strong elbows” at 3 and 6 as well. Thus, the other good options can be 3 or 6.

Activity data

The elbow method result of activity data is shown in Figure 4.6. Like baseline data, the figure shows the results of 2 clusters to 10 clusters. The results of 2 clusters to 6 clusters are studied and the rest results are referenced.

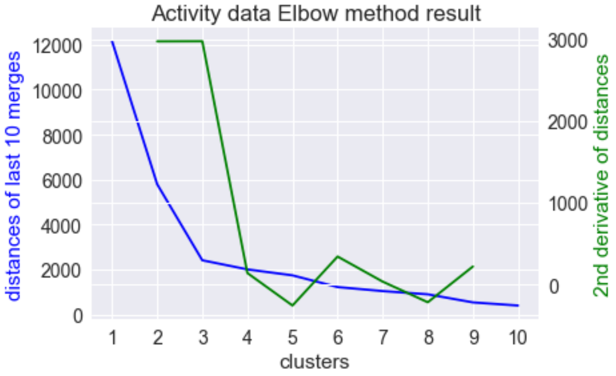


Figure 4.6: Elbow method result of activity data

As shown above, the green line has big values at 2 and 3. Combined with the trend of the blue line, the biggest value of the green line appears at 2. Thus, the ”strongest elbow” locates at 2. According to the standard, 2 is the best choice for the number of clusters. There is a relatively big value of green line at 6. Therefore, the other good options can be 3 or 6.

Education data

The elbow method result of education data is shown in Figure 4.7. The results of 2 clusters to 10 clusters are shown in the figure. Only the results of 2 clusters to 6 clusters are studied.

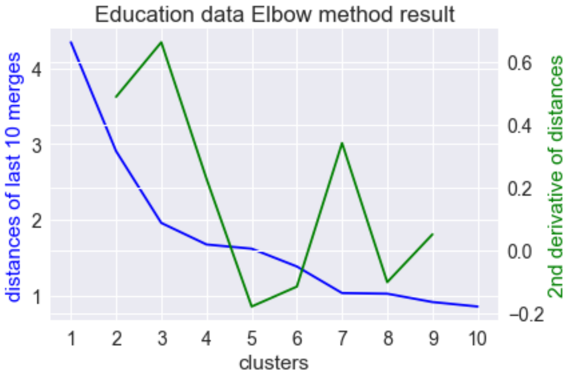


Figure 4.7: Elbow method result of education data

The green line is not monotonically decreasing. There are a few big values of green line, which refer to a few ”strong elbows” of the blue line. The ”strongest elbow” appears at 3. Thus, based on the standard, 3 is the optimal number of clusters. The ”elbows” at 2 and 4 are also strong. The other good choices can be 2 or 4.

Exercise data

The elbow method result of exercise data is shown in Figure 4.8. The results of 2 clusters to 10 clusters are shown in the figure. But only 2 clusters to 6 clusters are further studied. The rest are referenced.

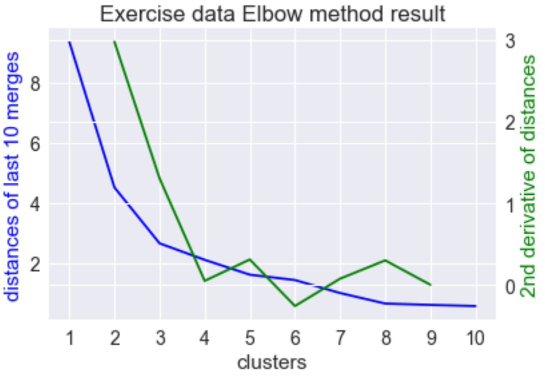


Figure 4.8: Elbow method result of exercise data

It’s clear to see the biggest value of the green line appears at 2. Thus, the ”strongest elbow” locates at 2. According to the standard, 2 is chosen as the best cluster number. The values of the 2nd derivative of distances (the green line) at 3 and 5 are relatively big. Therefore, the other good options can be 3 or 5.

User flow data

The elbow method result of user flow data is shown in Figure 4.9. Since 2 clusters to 16 clusters are studied, the last 16 merges are shown here.

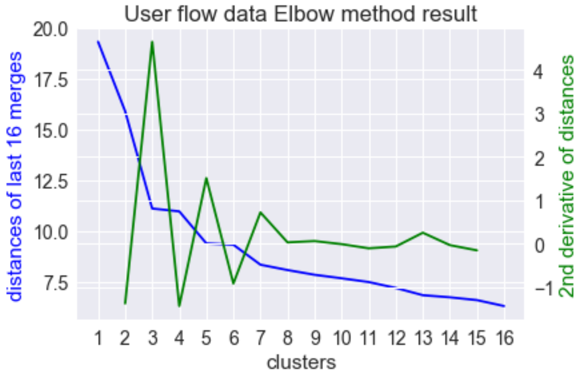


Figure 4.9: Elbow method result of user flow data

There are a few peaks in the green line. The peaks locate at 3, 5 and 7. The highest peak appears at 3, which is the ”strongest elbow”. Thus, 3 is chosen as the optimal number of clusters based on the standard. The other good options can be 5 or 7.

Sessions data

The elbow method result of sessions data is shown in Figure 4.10. Because 2 clusters to 8 clusters are studied, the last 8 merges are shown in the figure.

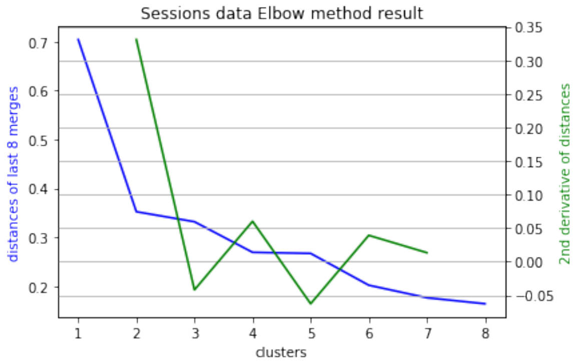


Figure 4.10: Elbow method result of sessions data

The biggest value of the green line appears at 2. Thus, the ”strongest elbow” is at 2. According to the standard, 2 is the best choice of cluster number. The values of the 2nd derivative of distances at 4 and 6 are relatively big. Therefore, the other good choices can be 4 or 6.

4.5.3 Fuzzy Partition Coefficient

According to the introduction of Fuzzy Partition Coefficient in Chapter 3, the standard to choose the best three numbers of clusters is to choose the biggest FPC values.

Baseline data

The FPC result of baseline data is shown in Figure 4.11.

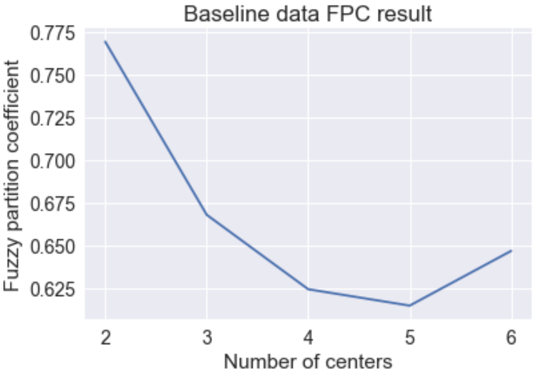


Figure 4.11: FPC result of baseline data

The FPC values are monotonically decreasing until 5. There is a small rise at 6. The biggest FPC value appears at 2. According to the standard, 2 is chosen as the best number of clusters. The FPC values at 3 and 6 are relatively big. Thus, the other good options can be 3 or 6.

Activity data

The FPC result of activity data is shown in Figure 4.12.

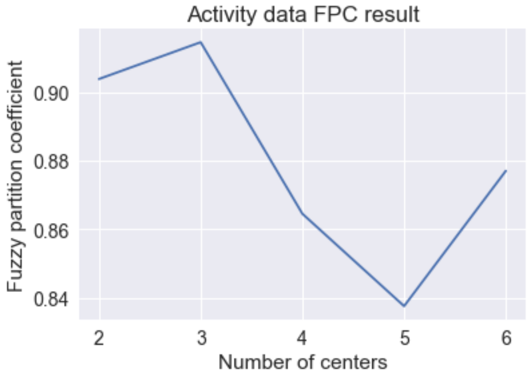


Figure 4.12: FPC result of activity data

The trend of the FPC values is like a zigzag. The biggest FPC values locates at 3. According to the standard, 3 is optimal choice for the number of clusters. The FPC values at 2 and 6 are relatively big. The other good options can be 2 or 6.

Education data

The FPC result of education data is shown in Figure 4.13.

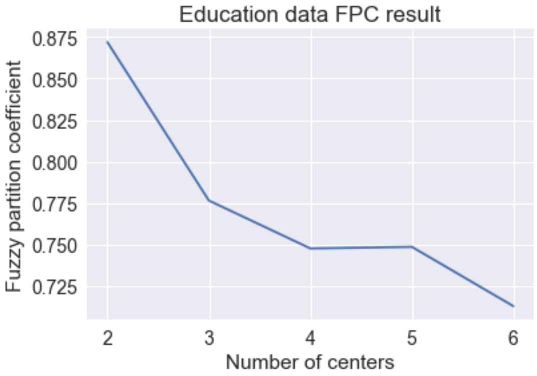


Figure 4.13: FPC result of education data

The FPC values are monotonically decreasing. The biggest FPC value shows at 2. Thus, 2 is chosen to be the best number of clusters. The FPC value at 3 is the second biggest and the one at 4 is the third biggest. Thus, the other options can be 3 or 4.

Exercise data

The FPC result of exercise data is shown in Figure 4.14.

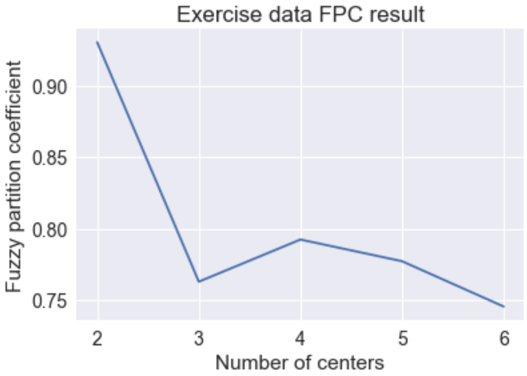


Figure 4.14: FPC result of exercise data

There is a big drop of FPC values between 2 and 3. The FPC values change in a small range after 3. Since the biggest FPC values appears at 2, according to the standard, 2 is chosen as the optimal number of clusters. The FPC values at 4 and 5 are relatively big. Therefore, the other good options can be 4 or 5.

User flow data

The FPC result of user flow data is shown in Figure 4.15.

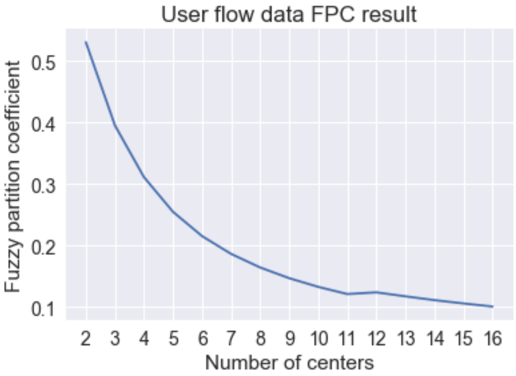


Figure 4.15: FPC result of user flow data

The FPC values is monotonically decreasing until 11. There is a small rise from 11 to 12. After 12, the FPC values continue decreasing. The biggest FPC values locates at 2. According to the standard, 2 is optimal number of clusters. The other good options can be 3 or 4.

Sessions data

The FPC result of sessions data is shown in Figure 4.16.

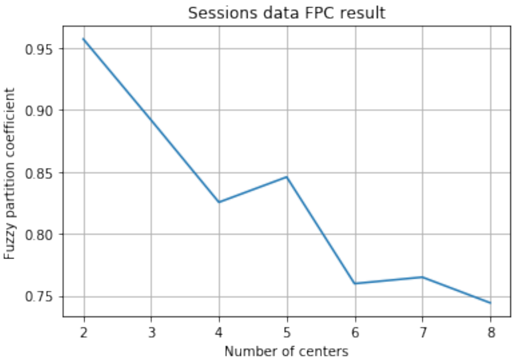


Figure 4.16: FPC result of sessions data

The trend of the FPC values is not monotonous. The biggest FPC value appears at 2. Based on the standard, 2 is the best choice of cluster number. The FPC values at 3 and 5 are relatively big. Thus, the other good options can be 3 or 5.

4.5.4 Bayesian Information Criterion

Based on the introduction of Bayesian Information Criterion in Chapter 3, the standard to choose the best three numbers of clusters is to choose the smallest BIC values.

Baseline data

The BIC result of baseline data is shown in Figure 4.17.

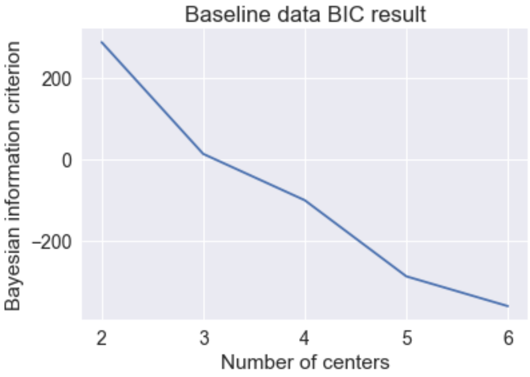


Figure 4.17: BIC result of baseline data

The BIC values are monotonically decreasing. The smallest BIC value appears at 6. According to the standard, 6 is chosen as the optimal number of clusters. The BIC value at 5 is the second smallest and the one at 4 is the third smallest. Thus, the other good options can be 4 or 5.

Activity data

The BIC result of activity data is shown in Figure 4.18.

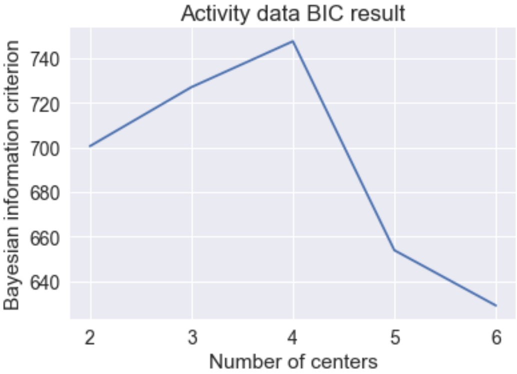


Figure 4.18: BIC result of activity data

The BIC values are increasing before 4 and decreasing after 4. There is a big drop from 4 to 5. The smallest BIC value locates at 6. According to the standard, 6 is chosen to be the best number of clusters. The BIC values at 2 and 5 are relatively small. The other good options can be 2 or 5.

Education data

The BIC result of education data is shown in Figure 4.19.

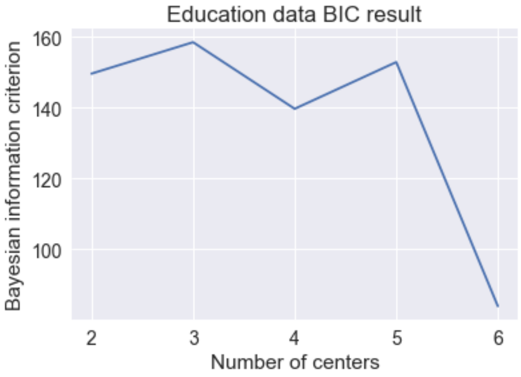


Figure 4.19: BIC result of education data

The trend of BIC values is like a zigzag until 5. There is a big drop from 5 to 6. The smallest BIC value shows at 6. Thus, 6 is the best choice of cluster number. The BIC values at 2 and 4 are relatively small. Therefore, the other options can be 2 or 4.

Exercise data

The BIC result of exercise data is shown in Figure 4.20.

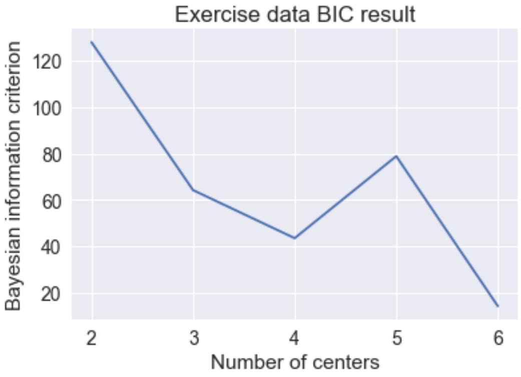


Figure 4.20: BIC result of exercise data

The trend of BIC values is not monotonous. The smallest BIC value appears at 6. And 6 is chosen as the optimal number of clusters based on the standard. The BIC values at 3 and 4 are relatively small. Therefore, the other good options can be 3 or 4.

User flow data

The BIC result of user flow data is shown in Figure 4.21.

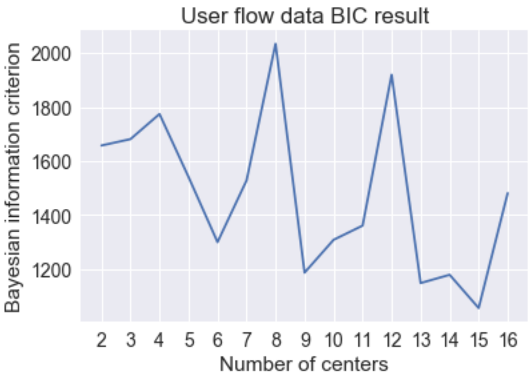


Figure 4.21: BIC result of user flow data

There are a few peaks and valleys in the trend of the BIC values. The smallest BIC value locates at 15. According to the standard, 15 is the optimal number of clusters. The BIC values at 13 and 14 are also relatively small. Thus, the other good options can be 13 or 14.

Sessions data

The BIC result of sessions data is shown in Figure 4.22.

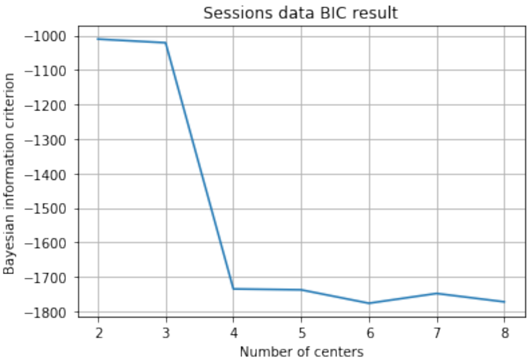


Figure 4.22: BIC result of sessions data

The trend of the BIC values is like a zigzag. The BIC values changes slowly after 4. The smallest value appears at 6. Based on the standard, 6 is chosen to be the best number of clusters. The other good options can be 7 or 8.

For each type of data, the best three numbers of clusters from each evaluation methods are shown in Table 4.12. For the same data set, the numbers of clusters from BIC are bigger than the ones from Silhouette analysis, the elbow method and FPC. The results from Silhouette analysis, the elbow method and FPC are similar. The results in Table 4.12 are applied in the clustering and the clustering results are shown in the next section.

Method	Silhouette analysis	Elbow method	FPC	BIC
Baseline data	2, 3, 6	2, 3, 6	2, 3, 6	4, 5, 6
Activity data	2, 3, 6	2, 3, 6	2, 3, 6	2, 5, 6
Education data	2, 3, 4	2, 3, 4	2, 3, 4	2, 4, 6
Exercise data	2, 3, 4	2, 3, 5	2, 4, 5	3, 4, 6
User flow data	2, 3, 4	3, 5, 7	2, 3, 4	13, 14, 15
Sessions data	2, 3, 4	2, 4, 6	2, 3, 5	6, 7, 8

Table 4.12: The best three cluster numbers of all types of data

4.6 Result analysis

The clustering results of six types of data from five clustering methods with different numbers of clusters are compared both horizontally and vertically to select the best clustering results. Due to space limitations, only the best clustering results are shown here while other clustering results are omitted.

4.6.1 Baseline data

The results of Silhouette analysis, the elbow method, FPC and BIC of baseline data are shown in Table 4.13.

Method	Silhouette analysis	Elbow method	FPC	BIC
Cluster number	2, 3, 6	2, 3, 6	2, 3, 6	4, 5, 6

Table 4.13: Cluster numbers of baseline data

Based on the results in Table 4.13, 2 clusters and 3 clusters are applied in K-means, K-medoids and agglomerative clustering. 4 clusters and 5 clusters are applied in GMM. 6 clusters are applied in K-means, K-medoids, agglomerative clustering, Fuzzy C-means and GMM.

Since the baseline data is clustered in five dimensions (five features), the clustering results will be presented and analyzed by heatmaps. Heatmaps can show the features of the baseline data more straight forward than other visualization methods like scatter plots. The heatmap of all users is shown in Figure 4.23.

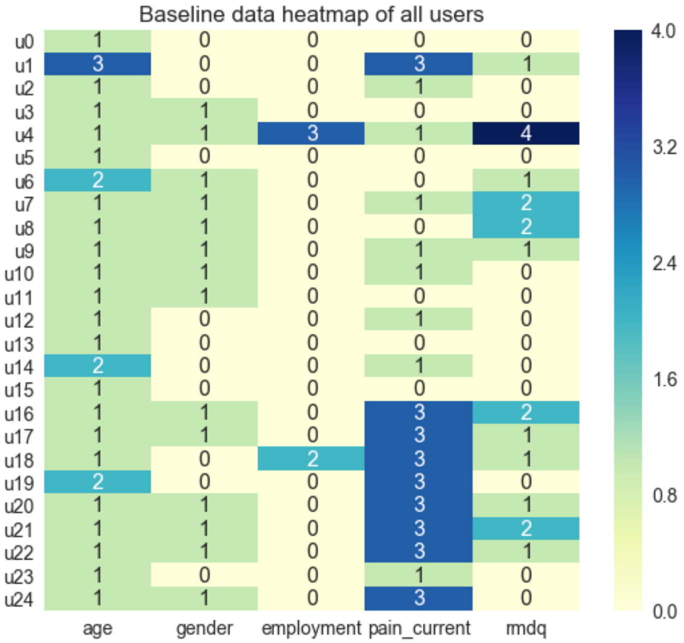


Figure 4.23: Baseline data heatmap of all users

Comparing the different clustering results from different methods in different numbers of clusters, the 3 clusters from K-means and agglomerative clustering are the same and describe the baseline data the best. The heatmap of each cluster is shown in Figure 4.24, Figure 4.25 and Figure 4.26.

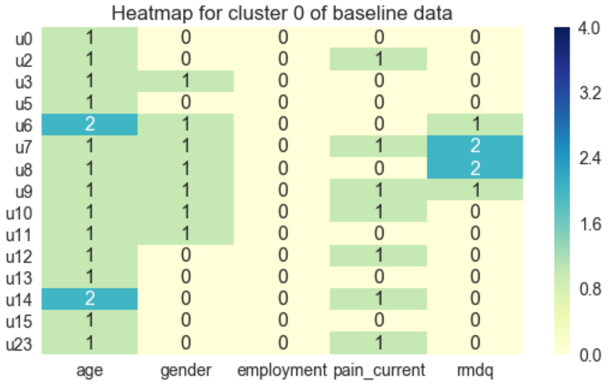


Figure 4.24: Cluster 0 of baseline data

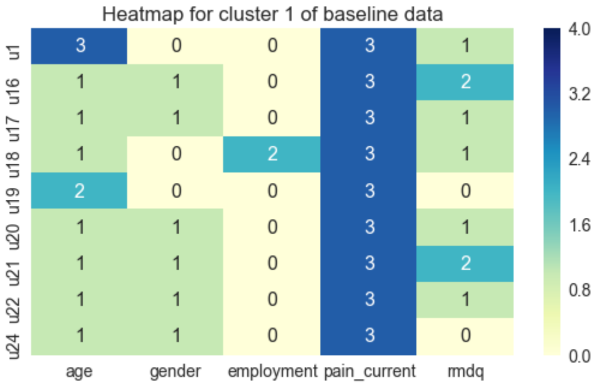


Figure 4.25: Cluster 1 of baseline data

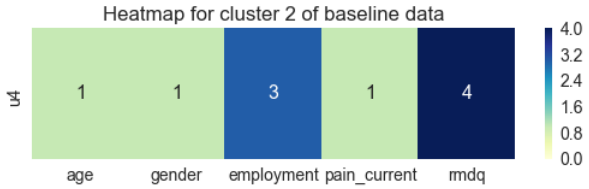


Figure 4.26: Cluster 2 of baseline data

Cluster 0 includes users with low scores in all the five features. Cluster 1 includes users with high scores in *pain_current* and different scores in other features. Cluster 2 includes a user with high scores in *employment* and *RMDQ*.

With reference to Table 4.2, users in cluster 0 are aged from 25 to 65. Approximately half of them are males and half of them are females. They are all full-time employed. They have low back pain for at least 1 week and up to 4 weeks. Their *RMDQ* scores are between 0 and 2. The values of five features indicate that these users are in the middle age group and full-time employed. Their low back pain is short and not serious.

Most users in cluster 1 are aged from 25 to 65. There is one user aged over 65. Most users are females. Except one unemployed user, other users are full-time employed. All the users have low back pain for over 12 weeks, with different levels of disability. The values of five features indicate that these users have suffered from low back pain for a long time.

The only user in cluster 2 is a female and aged between 25 to 55. Her employment status is ’other’. She has suffered from low back pain for 4 weeks with high disability.

4.6.2 Activity data

The results of Silhouette analysis, the elbow method, FPC and BIC of activity data are shown in Table 4.14.

Method	Silhouette analysis	Elbow method	FPC	BIC
Cluster number	2, 3, 6	2, 3, 6	2, 3, 6	2, 5, 6

Table 4.14: Cluster numbers of activity data

Based on the results in Table 4.14, 2 clusters and 6 clusters are applied in K-means, K-medoids, agglomerative clustering, Fuzzy C-means and GMM. 3 clusters are applied in K-means, K-medoids, agglomerative clustering and Fuzzy C-means. 5 clusters are applied in GMM.

Scatter plots are used to present the clustering result. Heatmaps are used to show the features of each cluster. Activity data is normalized before showing in clusters with heatmaps. In order to see the features of each cluster clearly, a heatmap of all users before normalization is shown in Figure 4.27 to be referenced later.

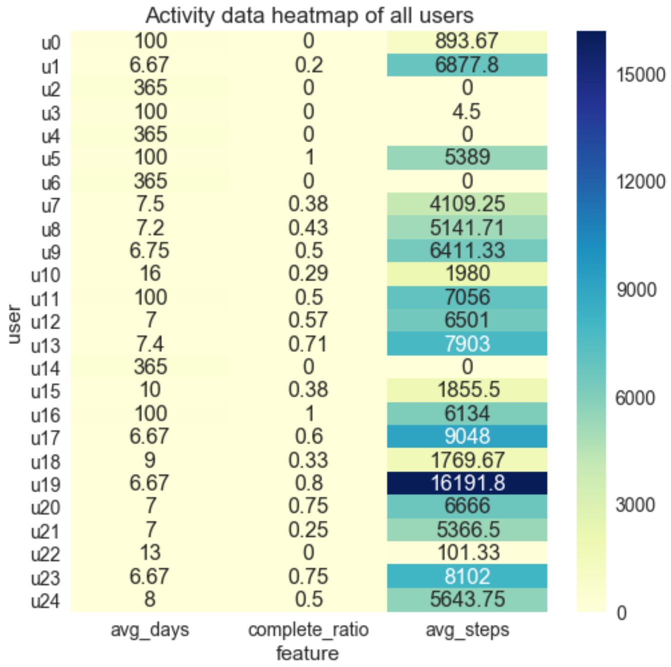


Figure 4.27: Activity data heatmap of all users before normalization

Comparing the different clustering results from different methods in different numbers of clusters, the 5 clusters from GMM describe the activity data the best. The scatter plot of the 5 clusters is shown in Figure 4.28.

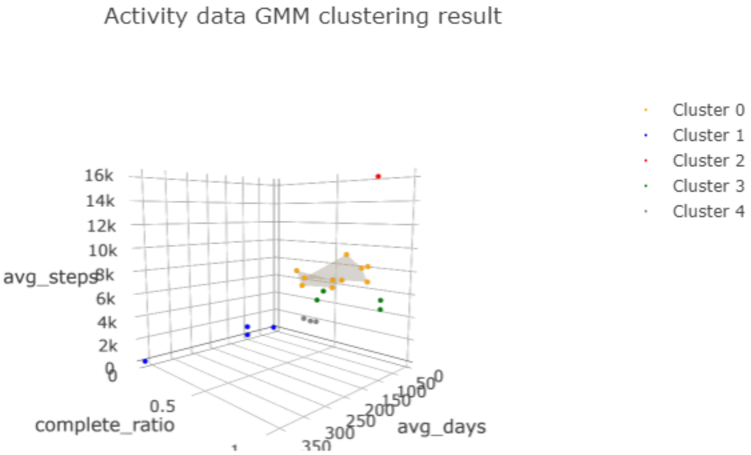


Figure 4.28: Scatter plot of activity data

The heatmaps of cluster 0 and cluster 1 are shown in Figure 4.29 and Figure 4.30. The heatmaps of cluster 2, cluster 3 and cluster 4 are shown in Figure 6.1, Figure 6.2 and Figure 6.3 in the appendix.

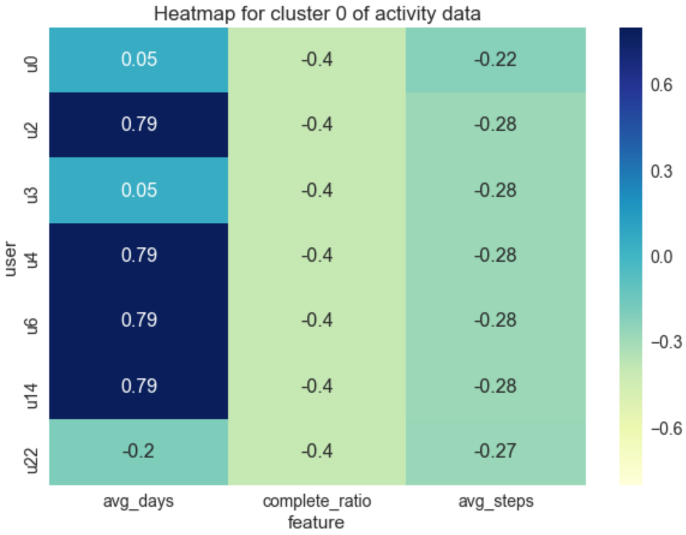


Figure 4.29: Cluster 0 of activity data

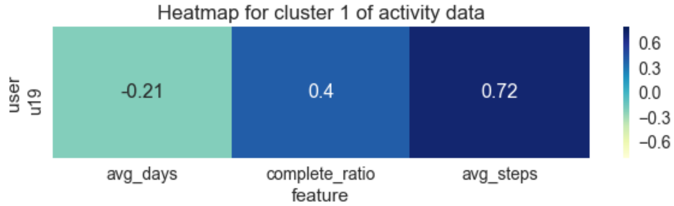


Figure 4.30: Cluster 1 of activity data

In cluster 0, most users have high scores in *avg_days*. All the users have the same low score in *complete_ratio* and similar low scores in *avg_steps*. Cluster 1 includes a user with low score in *avg_days* and high scores in *complete_ratio* and *avg_steps*. In cluster 2, most users have low scores in *avg_days* and middle scores in *complete_ratio* and *avg_steps*. Cluster 3 is a mixed cluster with two score characteristics. Two users have middle scores in *avg_days* and high scores in *complete_ratio*. The other two users have low scores in *avg_days* and middle scores in *complete_ratio*. All the four users have middle scores in *avg_steps*. In cluster 4, users have low scores in *avg_days* and *avg_steps*. They have middle scores in *complete_ratio*.

With reference to Figure 4.27, cluster 0 includes users with complete ratio 0 and average steps approximately 0. Most users have average days bigger than 100, indicating they have at most one activity. Thus, cluster 0 is for users with approximately zero activity.

Cluster 1 includes only 1 user, who has average days 6.67, complete ratio 0.8, and average steps about 16000. This user has activities in the best quality and quantity among all users.

Cluster 2 includes users with average steps over 5000. Most of them have average days smaller than 8 and complete ratios bigger than 0.5. Thus, cluster 2 is for users with activities in good quality and quantity.

Cluster 3 is a mixed cluster with users having two different activity behaviors. Two users have average days 100, complete ratio 1, and average steps around 6000, indicating they have only one activity week in good quality. Two users have average days smaller than 8, complete ratio smaller than 0.5, and average steps around 5000, indicating they have activities in good quantity but bad quality.

Cluster 4 includes users with average days smaller than 16, complete ratio smaller than 0.5 and average steps around 2000. Thus, this cluster is for users with activities in good quantity but bad quality.

Although most users are assigned to the right clusters, there are few users mis-clustered. The activity patterns will be more straight forward if user 1 in cluster 2, user 21 in cluster 2, user 7 in cluster 3, and user 8 in cluster 3 belong to cluster 4.

4.6.3 Education data

The results of Silhouette analysis, the elbow method, FPC and BIC of education data are shown in Table 4.15.

Method	Silhouette analysis	Elbow method	FPC	BIC
Cluster number	2, 3, 4	2, 3, 4	2, 3, 4	2, 4, 6

Table 4.15: Cluster numbers of education data

Based on the results in Table 4.15, 2 clusters and 4 clusters are applied in K-means, K-medoids, agglomerative clustering, Fuzzy C-means and GMM. 3 clusters are applied in K-means, K-medoids, agglomerative clustering and Fuzzy C-means. 6 clusters are applied in GMM.

Scatter plots are used to present the clustering result. Heatmaps are used to show the features of each cluster. Education data is normalized before showing in clusters with heatmaps. In order to see the features of each cluster clearly, a heatmap of all users before normalization is shown in Figure 4.31 to be referenced later.

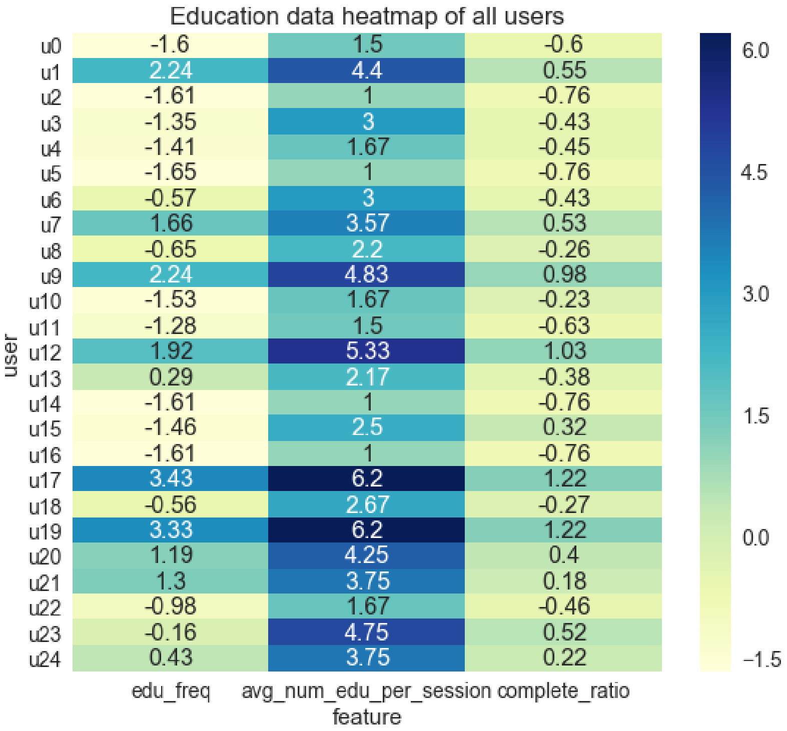


Figure 4.31: Education data heatmap of all users before normalization

In the heatmap of all users' education data, the values of *education frequency* and *complete ratio* have been processed with PCA. For *complete ratio*, the values are between -0.76 to 1.22. The higher the complete ratio score in the heatmap, the higher the original complete ratio. For example, complete ratio score 1.22 refers to complete ratio 1, while complete ratio score -0.76 refers to complete ratio 0. The values of education frequency in the heatmap work in the same way.

Comparing the different clustering results from different methods in different numbers of clusters, the 4 clusters from K-means describe the education data the best. The scatter plot of the 4 clusters is shown in Figure 4.32.

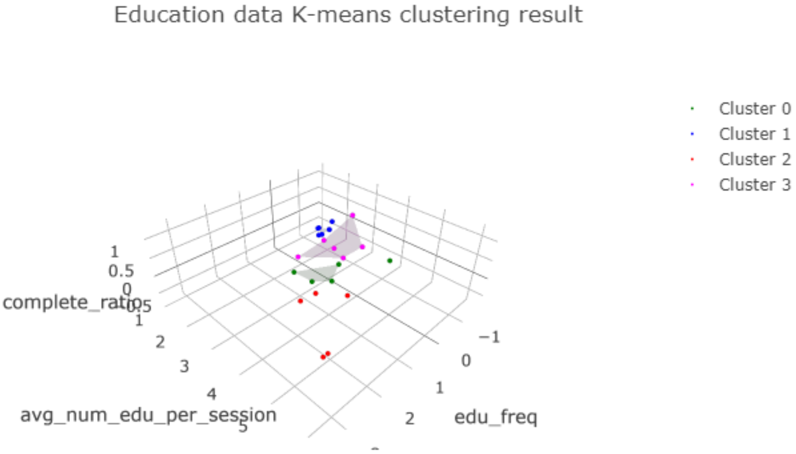


Figure 4.32: Scatter plot of education data

The heatmaps of cluster 0 and cluster 1 are shown in Figure 4.33 and Figure 4.34. The heatmaps of cluster 2 and cluster 3 are shown in Figure 6.4 and Figure 6.5 in the appendix.

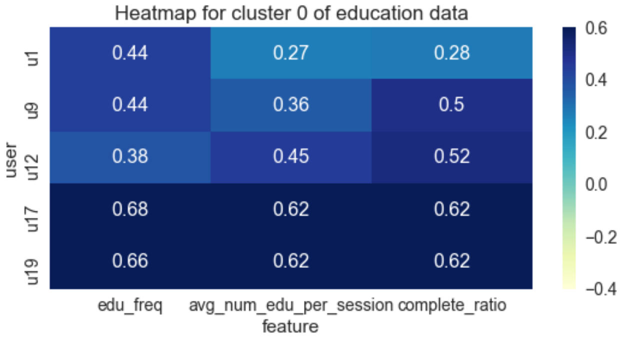


Figure 4.33: Cluster 0 of education data

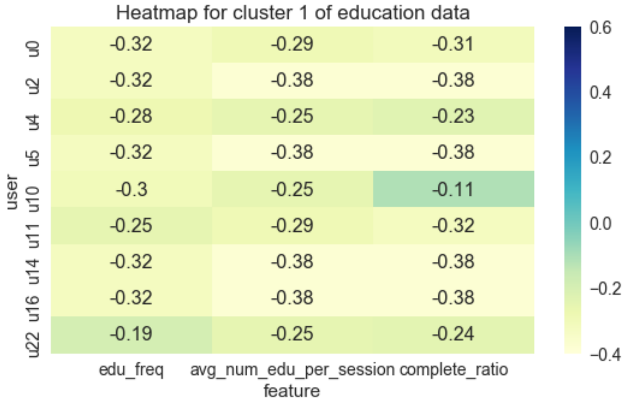


Figure 4.34: Cluster 1 of education data

The 4 clusters represent 4 levels of scores in three features. In cluster 0, users have high scores in three features. In cluster 1, users have low scores in three features. In cluster 2, users have mid-low scores. In cluster 3, users have mid-high scores.

In the app, there is one education tip suggested for each user everyday. Thus, in one education week, there are 7 education tips suggested for each user. With reference to Figure 4.31, users in cluster 0 have education frequency scores bigger than 1.92, average number of educations per week bigger than 4.4, and complete ratio scores bigger than 0.55. The values of education frequency indicate that users in cluster 0 read many different education tips. The values of the average number of educations per week and complete ratio indicate that these users read education tips for more than 4 days in a week.

In cluster 1, users have education frequency scores smaller than -0.98, average number of educations per week smaller than 1.67, and complete ratio scores smaller than -0.23. The values of education frequency indicate that users in cluster 1 read only a few different education tips. The values of the rest two features show that these users read education tips for at most 2 days in a week.

In cluster 2, users have education frequency scores between -1.46 and 0.29, average number of educations per week between 2.17 and 3, and complete ratio scores between -0.43 and 0.32. The values of education frequency show that users in cluster 2 read mid-low number of different education tips, which are more than users in cluster 1 but less than users in cluster 0. The values of the rest two features show that these users read education tips for 2 to 3 days a week.

In cluster 3, users have education frequency scores between -0.16 and 1.66, average number of educations per week between 3.57 and 4.75, and complete ratio scores between 0.18 and 0.53. The values of education frequency indicate that users in cluster 3 read mid-high numbers of different education tips, which are more than users in cluster 2 but less than users in cluster 0. The values of the rest two features show that these users read education tips for 3 to 5 days in a week.

4.6.4 Exercise data

The results of Silhouette analysis, the elbow method, FPC and BIC of exercise data are shown in Table 4.16.

Method	Silhouette analysis	Elbow method	FPC	BIC
Cluster number	2, 3, 4	2, 3, 5	2, 4, 5	3, 4, 6

Table 4.16: Cluster numbers of exercise data

Based on the results in Table 4.16, 2 clusters are applied in K-means, K-medoids, agglomerative clustering, and Fuzzy C-means. 3 clusters are applied in K-means, K-medoids, agglomerative clustering, and GMM. 4 clusters are applied in K-means, K-medoids, Fuzzy C-means and GMM. 5 clusters are applied in agglomerative clustering and Fuzzy C-means. 6 clusters are applied in GMM.

Scatter plots are used to present the clustering result. Heatmaps are used to show the features of each cluster. Exercise data is normalized before showing in clusters with heatmaps. In order to see the features of each cluster clearly, a heatmap of all users before normalization is shown in Figure 4.35 to be referenced later.

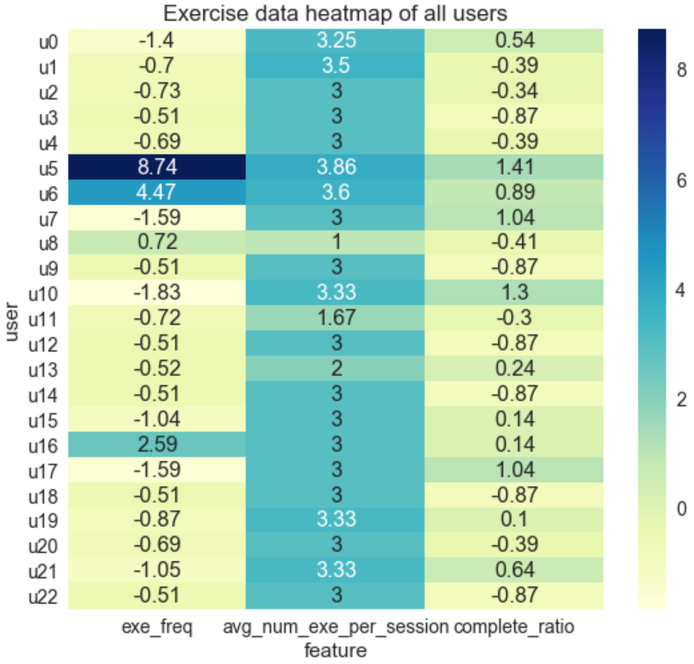


Figure 4.35: Exercise data heatmap of all users before normalization

The values of *exercise frequency* and *complete ratio* in the heatmap of all users' exercise data have been processed with PCA. The higher the value of exercise frequency in the heatmap, the higher the values of original exercise frequency. The negative values indicates small values of original data. The values of complete ratios work in the same way.

Comparing the different clustering results from different methods in different numbers of clusters, the 5 clusters from Fuzzy C-means describe the exercise data the best. The scatter plot of the 5 clusters is shown in Figure 4.36.

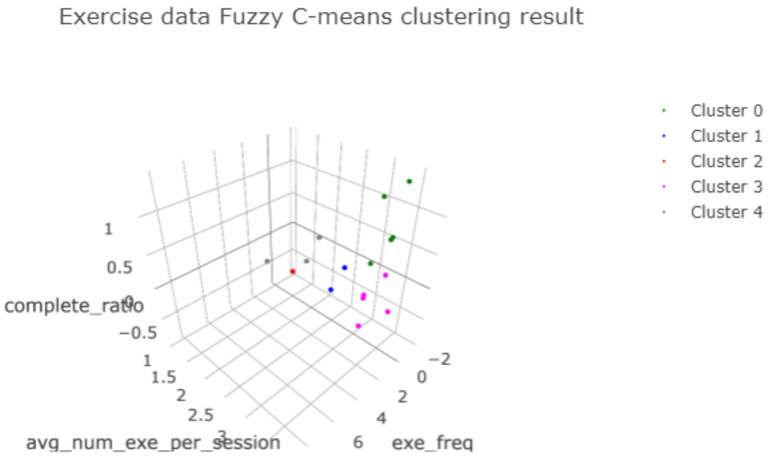


Figure 4.36: Scatter plot of exercise data

The heatmaps of cluster 0 and cluster 1 are shown in Figure 4.37 and Figure 4.38. The heatmaps of cluster 2, cluster 3 and cluster 4 are shown in Figure 6.6, Figure 6.7 and Figure 6.8 in the appendix.

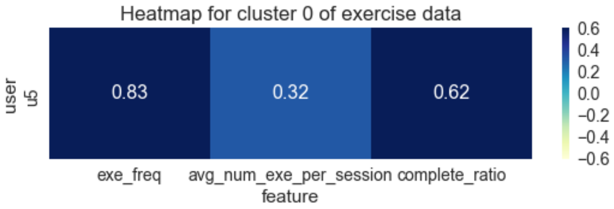


Figure 4.37: Cluster 0 of exercise data

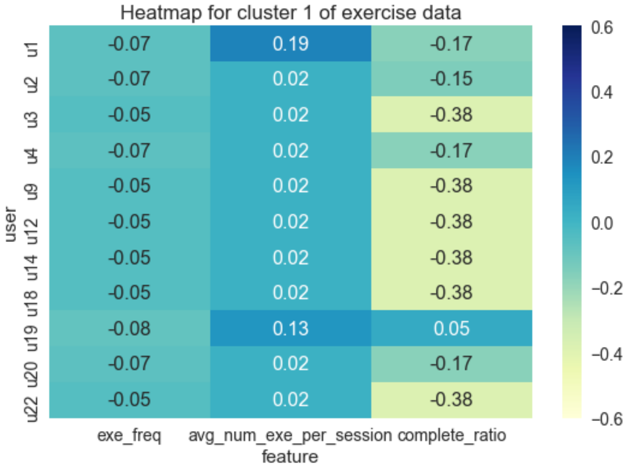


Figure 4.38: Cluster 1 of exercise data

Cluster 0 includes a user with the highest scores in three features. In cluster 1, most users have low scores in three features. Cluster 2 includes users with low scores in exercise frequency, middle scores in average number of exercises per week, and high scores in complete ratio. Cluster 3 includes users with the lowest scores in average number of exercises per week. They have middle scores in other two features. Cluster 4 includes users with mid-high scores in three features.

In the app, normally users are suggested to have 3 to 5 exercises per week, which is per exercise week. With reference to Figure 4.35, cluster 0 includes a user with exercise frequency score 8.74, average number of exercises per week 3.86, and complete ratio score 1.41. The value of exercise frequency shows that this user have done many different exercises. The values of average number of exercises per week and complete ratio show that this user do exercises at least three times a week, which is enough for a week as suggested.

In cluster 1, users have exercise frequency scores between -0.87 and -0.51, average number of exercises per week between 3 and 3.5, and complete ratio score between -0.87 and 0.1. The values of exercise frequency indicate that users in cluster 1 have done mid-small numbers of different exercises. The values of the rest two features indicate that these users do exercises at least three times a week, which is enough for a week as suggested.

In cluster 2, users have exercise frequency scores between -1.83 and -1.04, average number of exercises per week between 3 to 3.33, and complete ratio scores between 0.14 and 1.3. The values of exercise frequency indicate that users in cluster 2 have done small numbers of different exercises. The values of the rest two features indicate that these users do exercises at least three times a week, which is enough for a week as suggested.

In cluster 3, users have exercise frequency scores between -0.72 and 0.72, average number of exercises per week less than 2, and complete ratio scores between -0.41 and 0.24. The values of exercise frequency show that users in cluster 3 have done middle numbers of different exercises. The values of the rest two features show that these users do exercises twice a week at most, which is not enough as suggested.

In cluster 4, users have exercise frequency score between 2.59 and 4.47, average number of exercises per week between 3 to 3.6, and complete ratio scores between 0.14 to 0.89. The values of exercise frequency show that users in cluster 4 have done many different exercises. The values of the rest two features show that these users do exercises at least 3 times a week, which is enough as suggested.

Although most users are partitioned into right clusters, there is one user mis-clustered. The exercise patterns will be more straight forward if user 15 in cluster 2 belongs to cluster 1.

4.6.5 User flow data

The results of Silhouette analysis, the elbow method, FPC and BIC of user flow data are shown in Table 4.17.

Method	Silhouette analysis	Elbow method	FPC	BIC
Cluster number	2, 3, 4	3, 5, 7	2, 3, 4	13, 14, 15

Table 4.17: Cluster numbers of user flow data

Based on the results in Table 4.17, 2 clusters are applied in K-means, K-medoids, and Fuzzy C-means. 3 clusters are applied in K-means, K-medoids, agglomerative clustering and Fuzzy C-means. 4 clusters are applied in K-means, K-medoids and Fuzzy C-means. 5 clusters and 7 clusters are applied in agglomerative clustering. 13 clusters, 14 clusters, and 15 clusters are applied in GMM.

Since the user flow data is clustered in 10 dimensions (10 sessions), heatmaps are used to show the clustering results instead of scatter plots because heatmaps can show the features of each cluster more straight forward. In order to see the features of each cluster clearly, a heatmap of all users is shown in Figure 4.39 to be referenced later.

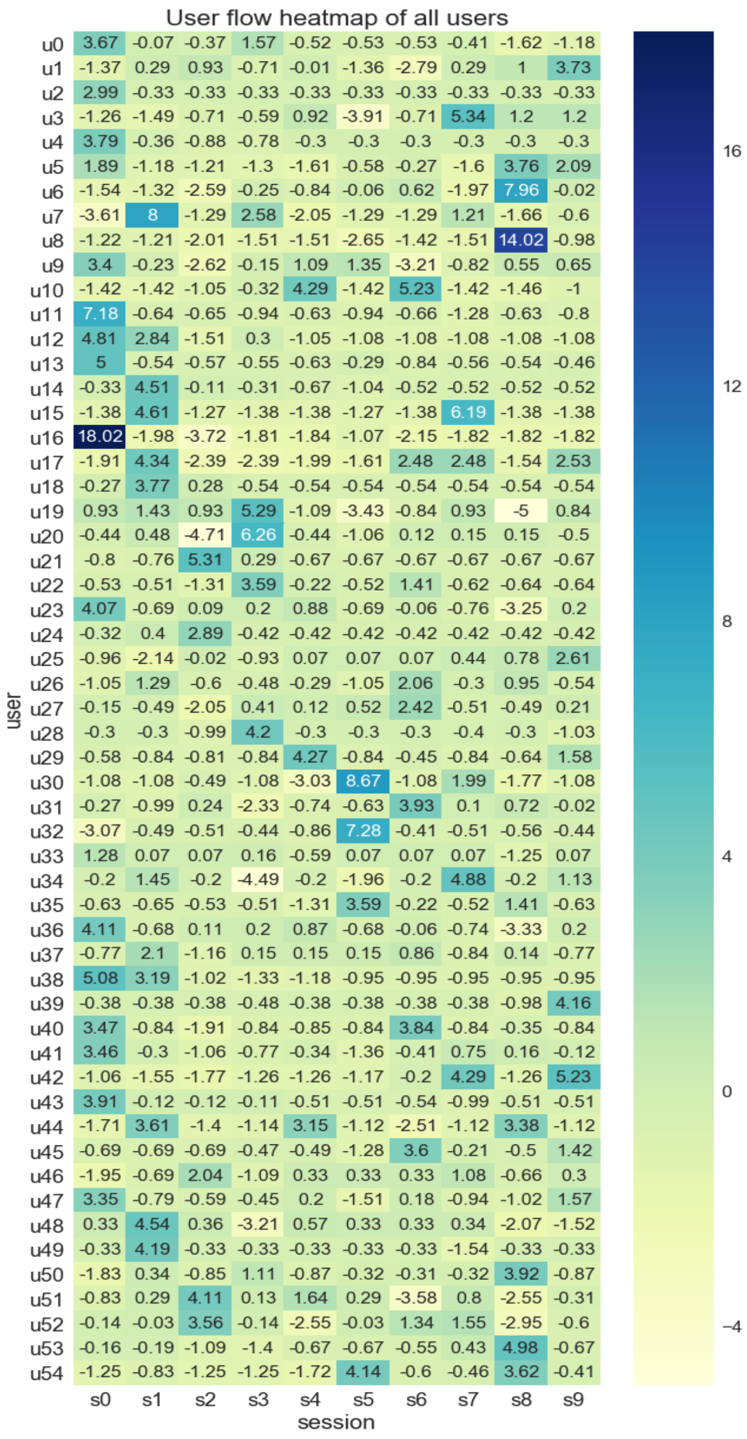


Figure 4.39: User flow data heatmap of all users

Comparing the different clustering results from different methods in different numbers of clusters, the 4 clusters from Fuzzy C-means describe the user flow data the best. The heatmaps of cluster 0 and cluster 1 are shown in Figure 4.40 and Figure 4.41. The heatmaps of cluster 2 and cluster 3 are shown in Figure 6.9 and Figure 6.10 in the appendix.

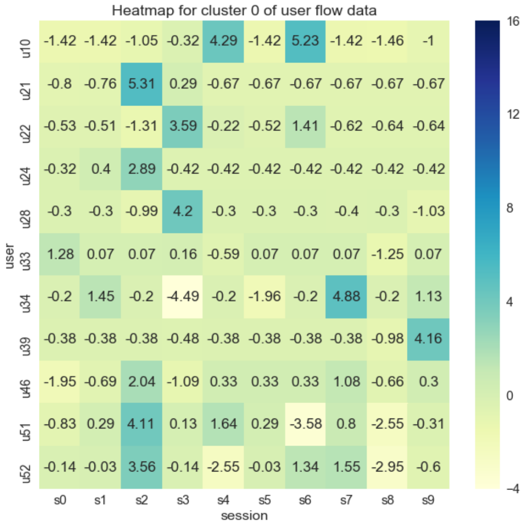


Figure 4.40: Cluster 0 of user flow data

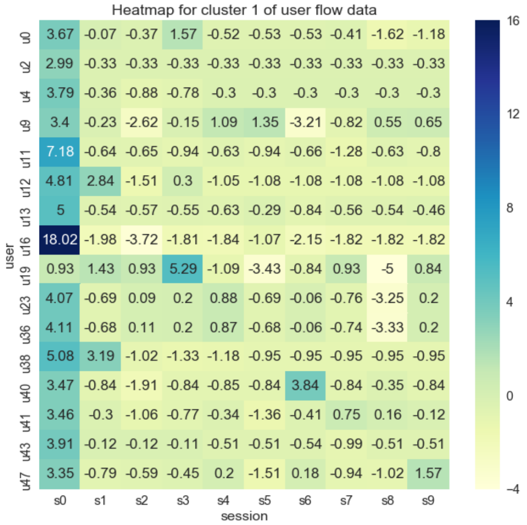


Figure 4.41: Cluster 1 of user flow data

Cluster 0 includes users with large scores concentrated in session 2 to session 4. Cluster 1 includes users with large scores concentrated in session 0. Cluster 2 includes users with large scores concentrated in session 5 to session 9. Cluster 3 includes users with large scores concentrated in session 1.

Big user flow scores indicate big numbers of page visiting in a session, and vice versa. From Figure 4.39 we can see that for every user, most sessions have small scores, indicating each user visits a certain small number of pages in one session in most cases. Each user has at least one big user flow score, indicating each user visits many pages in a session for at least once. The clustering results show the four behavior patterns of visiting pages in a session. In cluster 0, most users visit big numbers of pages in session 2 to session 4. In cluster 1, most users visit big numbers of pages in session 0. In cluster 2, most users visit big numbers of pages in session 5 to session 9. In cluster 3, most users visit big numbers of pages in session 1.

Although most users are assigned into right clusters. There are few users mis-clustered. User 20 in cluster 2, user 29 in cluster 2 and user 19 in cluster 1 should belong to cluster 0. User 33 in cluster 0 should belong to cluster 1. User 34 and user 39 in cluster 0 should belong to cluster 2.

4.6.6 Sessions data

The results of Silhouette analysis, the elbow method, FPC and BIC of sessions data are shown in Table 4.18.

Method	Silhouette analysis	Elbow method	FPC	BIC
Cluster number	2, 3, 4	2, 4, 6	2, 3, 5	6, 7, 8

Table 4.18: Cluster numbers of sessions data

Based on the results in Table 4.18, 2 clusters are applied in K-means, K-medoids, agglomerative clustering and Fuzzy C-means. 3 clusters are applied K-means, K-medoids and Fuzzy C-means. 4 clusters are applied in K-means, K-medoids and agglomerative clustering. 5 clusters are applied in Fuzzy C-means. 6 clusters are applied in agglomerative clustering and GMM. 7 clusters and 8 clusters are applied in GMM.

Considering the big number of sessions clustered, only scatter plots are used to present the clustering result. Comparing the different clustering results from different methods in different numbers of clusters, the 4 clusters from K-means describe the sessions data the best. The scatter plot of the 4 clusters is shown in Figure 4.42.

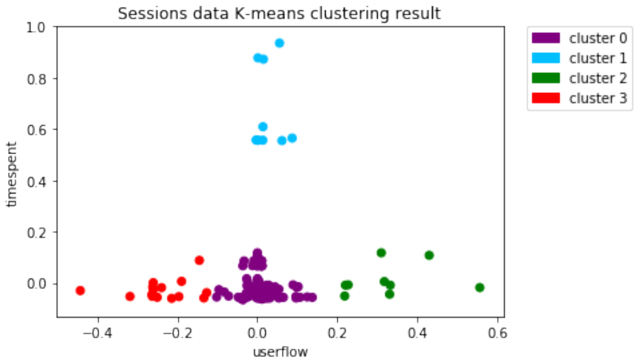


Figure 4.42: Sessions data K-means clustering result

As shown above, the clustering result presents four behavior patterns. Cluster 0 includes sessions with middle scores in user flow and low scores in time spent, indicating users visit middle numbers of pages in a session and spend short time on pages. Cluster 1 includes sessions with middle scores in user flow and high scores in time spent, indicating users visit middle numbers of pages in a session and spend long time on pages. Cluster 2 includes sessions with low scores in user flow and time spent, indicating users visit small numbers of pages in a session and spend short time on pages. Cluster 3 includes sessions with high scores in user flow and low scores in time spent, indicating users visit big numbers of pages in a session and spend short time on pages.

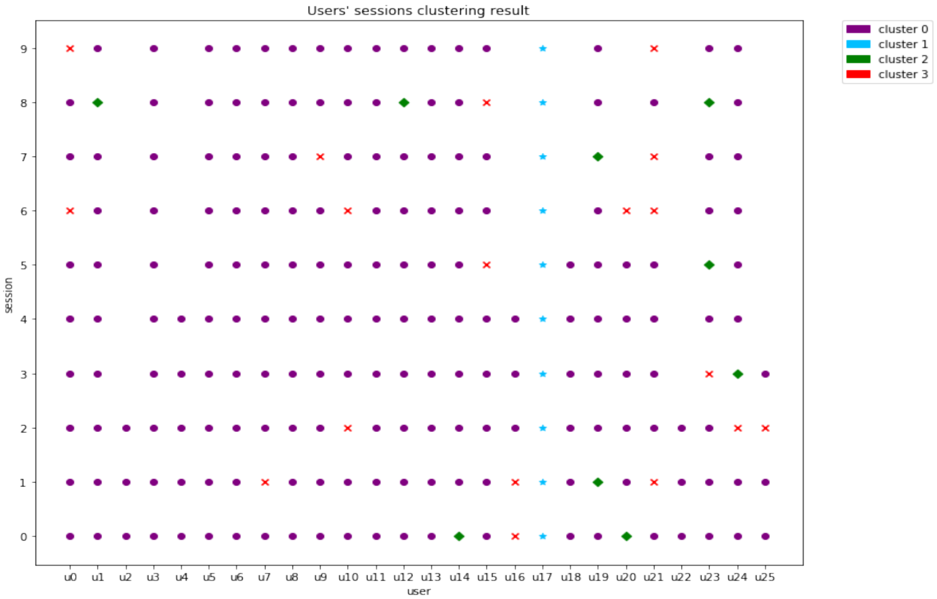


Figure 4.43: Users’ sessions clustering result

For each user, the clustering result of each session is shown in Figure 4.43. The x axis is users and the y axis is sessions. Each session is represented as a dot in the color of the cluster it belongs to. Take user 0 as an example. User 0 has 10 sessions, so there are 10 dots for user 0. Session 6 and session 9 belong to cluster 2, so these two dots are marked in green which is the color of cluster 2. The rest eight sessions belong to cluster 0, so these 8 dots are marked in purple which is the color of cluster 0.

As shown in Figure 4.43, most sessions belong to cluster 0. Most users have cluster 0 as their representative cluster. Thus, in most cases, most users visit middle numbers of pages in a session and spend short time on pages. In some cases, users visit either only a few pages or many pages in a session and spend short time on pages. In rare cases, users visit middle numbers of pages in a session and spend long time on pages. User 17 has cluster 1 as his/her representative cluster, indicating he/she visits middle numbers of pages in a session and spends long time on pages.

Chapter 5

Discussion

This chapter discussed the experiment results from two aspects: main goal and research questions, and challenges. In main goal and research questions, the clustering process is discussed with the main goal and research questions mentioned in Chapter 1 as standards. Challenges existed in the thesis are discussed in the second section.

5.1 Main Goal and Research Questions

As stated in Chapter 1, the main goal of this thesis is to find out how to recognize app usage patterns by analyzing the app usage data with machine learning methods. In order to answer this question, the four research questions are answered first.

RQ1. What machine learning methods can be applied on the SELFBACK app usage data?

The SELFBACK app usage data is unlabeled. Thus, unsupervised learning methods can be applied on it. There are five clustering methods and four evaluation methods applied in this thesis.

The five clustering methods are K-means, K-medoids, agglomerative clustering, Fuzzy C-means and Gaussian Mixture Models. Among these five methods, K-means and agglomerative clustering are chosen based on the result of literature review in Chapter 2. In order to eliminate the limitations of K-means and agglomerative clustering, three more methods are also applied on the data sets. Considering the small sizes of the given data sets, K-means is less robust to noises than K-medoids. For example, if there are only a few data points, such as (1,1), (1,2), (2,1) and (100,100). Among them, (100,100) is the noise. In K-means, the centroid will be roughly in the middle of (1,1) and (100,100), which is obviously not what we want. However, K-medoids can avoid this situation. The centroid will be selected from the first three points of (1,1), (1,2), (2,1) and (100,100) to minimize the absolute error of the cluster. Thus, K-medoids is also applied on the data sets. Besides, K-means K-medoids, and agglomerative clustering are ”hard” clustering methods, which may lead to some wrong clustering results due to their nature. Thus, ”soft” clustering methods Fuzzy C-means and Gaussian Mixture Models are also applied.

There are four evaluation methods applied in the thesis to choose the optimal number of clusters for each type of data set. They are Silhouette analysis, the elbow method, Fuzzy Partition Coefficient and Bayesian Information Criterion. Silhouette analysis and the elbow method are used for ”hard” clustering, while Fuzzy Partition Coefficient and Bayesian Information Criterion are used for ”soft” clustering. Considering K-means and K-medoids are almost the same except the way of selecting centroids, the results of Silhouette analysis are used in both of them. In agglomerative clustering, distances between two levels of nodes can be a good indication for clustering and they are easy to be presented with the elbow method. Therefore, the results of the elbow methods are used in agglomerative clustering. Fuzzy Partition Coefficient was proposed for fuzzy clustering by Dunn who also first proposed fuzzy clustering. Thus, the results of Fuzzy Partition Coefficient is applied in Fuzzy C-means. In literature, Bayesian Information Criterion is the most popular criterion for Gaussian Mixture Models and it is implemented in the package *GaussianMixture* in python. Therefore, the results of Bayesian Information Criterion are used in Gaussian Mixture Models. The four evaluation methods focus on different aspects. Their limitations are eliminated by combining them with corresponding clustering methods, leading to more scientific clustering results.

RQ2. How to apply those machine learning methods on the given data?

The given data is raw data. Before applying clustering methods on it, data preprocessing needs to be applied on it to transform the raw data into matrices which can be used in clustering methods. The data preprocessing consists of three parts: feature selection and extraction, dimensionality reduction and feature scaling. In feature selection and extraction, the selected raw data is transformed into numbers according to different metrics based on the features which are used as dimensions in clustering. In dimensionality reduction, PCA is applied on the data to use only one number to represent one feature for each user or each session. In feature scaling, data is normalized into small ranges to limit the effects of wide ranges of data.

After preprocessing, the data is ready to be clustered. For the five clustering methods, the number of clusters is required as a parameter. Hence, before applying clustering methods on the data, the four evaluation methods are applied on the data to see the distribution characteristics of the data and to decide the numbers of clusters for applying the clustering methods. For each type of data, the best three cluster numbers of each evaluation method are chosen to be studied. The cluster numbers are applied in the clustering methods as the key parameter. After the data is preprocessed and the cluster numbers are chosen, the five clustering methods are applied on the data.

RQ3. How to choose the best clustering results among the results from different methods?

With reference to Table 4.12, we can see that for baseline data, activity data, education data and exercise data, the results of four evaluation methods remain consistent with little variance. However, for user flow data and sessions data, the cluster numbers from Bayesian Information Criterion are much bigger than the ones generated by Silhouette analysis, the elbow method and Fuzzy Partition Coefficient. Considering that the sizes of user flow data and sessions data are much bigger than the sizes of baseline data, activity data, education data and exercise data, this is an interesting found about the result of Bayesian Information Criterion. This can be further studied in the future when there is more data in baseline, activity, education and exercise. The different values of cluster numbers from different evaluation methods show that applying different methods will produce different results. By analyzing all the results, the limitations of only applying one method are eliminated and more reliable results can be selected.

Based on the results of the four evaluation methods, the clustering results generated in different numbers of clusters from different clustering methods are compared both vertically and horizontally. Take the baseline data as an example. As shown in Table 4.13, 2, 3, 4, 5 and 6 can be the candidates of the cluster number. 2 clusters and 3 clusters are applied in K-means, K-medoids and agglomerative clustering. 4 clusters and 5 clusters are applied in GMM. 6 clusters are applied in K-means, K-medoids, agglomerative clustering, Fuzzy C-means and GMM. In comparison, the clustering results are first compared in the same number of clusters, like comparing the 2 clusters from K-means, K-medoids and agglomerative clustering and choosing the best 2 clusters from them. The same procedure is applied for 3 clusters, 4 clusters, 5 clusters and 6 clusters. After selecting the local best result in each group, the local best results from all groups are compared together to select the global best clustering result. For baseline data, the local best results are the 2 clusters from K-means and Fuzzy C-means, 3 clusters from K-means and agglomerative clustering, 4 clusters from GMM, 5 clusters from GMM, and 6 clusters from agglomerative clustering. After the comparison, the 3 clusters from K-means and agglomerative clustering show the patterns of the baseline data the best and are selected as the final clustering result. The process of selecting the best results makes a comprehensive comparison of all candidate results, making the final result highly scientific and convincing.

RQ4. How well do the applied machine learning methods perform on the given data?

The clustering results of each type of data show that the machine learning methods applied perform well in finding different app usage patterns.

The four evaluation methods have similar cluster numbers in most cases and Bayesian Information Criterion can be further studied with more data as mentioned. Among the five clustering methods, K-means and Fuzzy C-means have the same clustering results in most cases. K-medoids, agglomerative clustering and Gaussian Mixture Models have their own clustering characteristics.

The best clustering results for each type of data and the corresponding evaluation methods are shown in Table 5.1.

Data type	Baseline	Activity	Education	Exercise	User flow	Sessions
Clustering method of the best result	K-means, agglomerative clustering	GMM	K-means	Fuzzy C-means	Fuzzy C-means	K-means
Evaluation method of the best result	Silhouette analysis, Elbow method, FPC	BIC	Silhouette analysis, Elbow method, FPC, BIC	Elbow method, FPC	Silhouette analysis, FPC	Silhouette analysis, Elbow method

Table 5.1: Best clustering methods and evaluation methods of each data type

From Table 5.1, we can see that for different types of data, the best clustering method and evaluation method differ. K-means and Fuzzy C-means are applied on most types of data to get the best clustering results. In most cases, the best number of clusters are acquired from Silhouette analysis, the elbow method and FPC. Thus, those methods can be first considered for most of the future data. For activity data, GMM and BIC should be first considered in the future.

Although there are few mis-clustered data points in some cases, it is acceptable considering the sizes of the data sets. The clustering results can be improved when more data is available.

The answers to the four research questions help to achieve the main goal of the thesis. The optimal clustering results show that for each type of data, there are certain app usage patterns recognized. Based on those usage patterns, representative users can be selected for the interview in the process evaluation. Although there are few data points clustered wrongly, the clustering results in total are positive. Further experimenting can be constructed with more data to acquire better results.

5.2 Challenges

The biggest challenge of this thesis is the small sizes of data sets. Small data sets are not variant enough and may have negative effects on the clustering results. For instance, there are only 23 users in exercise data to be clustered on three features. With reference to the scatter plot of exercise data in Figure 4.36, we can see that most data points have their own characteristics and can be self-clustered. If the number of clusters is big like 8, there may exist the over-fitting problem. If the number of clusters is small like 3, there may exist wrongly clustered data points. If there are more users to be clustered, users with similar characteristics will be distributed closer, which will make the clusters more representative and convincing.

Apart from the number of users studied in the thesis, the data of each user's sessions retrieved from *Matomo* also leads to the small amount of available data. Only the last 10 sessions can be acquired from *Matomo* due to its default settings. Efforts are made to try to change the settings to get more sessions. But the problem still exists.

Besides, the data processed in this thesis are from both study users who are the participants of the SELFBACK project and test users who are the staff of the project. Although the test users use the app in similar ways as study users, sometimes there are some abnormal behaviors like too long time or too many pages in a session. These behaviors are rare but still can effect the clustering results. Considering the sizes of the data sets, it is difficult to choose between keeping or evicting the abnormal data.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, five clustering methods and four evaluation methods are explored and applied on the SELFBACK app usage data. For each type of app usage data, user behavior patterns are recognized. The numbers of recognized patterns and their corresponding qualities are shown in Table 6.1.

Data type	Baseline	Activity	Education	Exercise	User flow	Sessions
Number of patterns	3	5	4	5	4	4
Quality of patterns	1	0.84	1	0.96	0.89	1

Table 6.1: Patterns' quantity and quality of each data type

The quality of patterns are calculated by the ratio between the number of right clustered data points and the total number of data points. If there is no data point mis-clustered, the quality will be 1. The higher the ratio, the better the clustering results. The quantity and quality of patterns recognized for each data type show that the clustering results are very positive.

Based on the data acquired, the thesis accomplishes the available goals of process evaluation. For baseline data from back-end, *gender*, *age*, *occupational status*, *duration of low back pain* and *Roland Morris scores* are analyzed in the five features in the baseline data section. For activity data from back-end, *the steps count goals*, *if goals achieved* and *step count progress across study period weekly* are analyzed in the three features in the activity data section. For education data from back-end, *education screens looked at with frequency* and *completion rates* are analyzed in the three features in the education data section. For exercise data from back-end, *exercise screens looked at with frequency* and *completion rates* are analyzed in the three features in the exercise data section. For analytics data retrieved from *Matomo*, the *user journeys* are analyzed in the user flow data section. The *frequency of opening the app* and *duration of using different sections of the app* in the last 10 sessions of each user are analyzed as the two features in the sessions data section.

The clustering results are in good quality and can be used as direct references to choose interviewees in the process evaluation.

6.2 Future Work

With reference to the challenges in the thesis, in the future, improvements can be done in three main aspects.

First, more data can be analyzed when there are more users using the app regularly. When the sizes of data sets are big enough, more clustering methods, like Neural Network, can be applied to further eliminate the limitations of the five clustering methods used in this thesis.

Second, the problem of only getting last 10 sessions of each user should be solved as soon as possible. This problem is the main cause of insufficient analytics data. The behavior patterns in user journeys and duration on different sections will be more obvious and convincing if all the sessions of each user can be acquired.

Third, although the test users analyzed in this thesis behave similarly as study users who have real low back pain problems, some of their abnormal data, like the data when they test the app, can still be obstructions of getting the best behavior patterns. Thus, the behavior patterns recognized will be closer to the ones in the real world of low back pain if only the participants of the SELFBACK project are studied.

Bibliography

Abolfazli, S., Lee., M. R., 2017. Mobile data analytics. IT Professsional 19 (3).

Andrei, O., Calder., M., 2017. Temporal analytics for software usage models. In: Software Engineering and Formal Methods. Trento, Italy.

Bach, K., Mork, P. J., Aamodt., A., 2016a. Can data-driven self-management reduce low back pain? Ercim News (104).

Bach, K., Szczepanski, T., Aamodt, A., Gundersen, O. E., Mork., P. J., 2016b. Case representation and similarity assessment in the SELFBACK decision support system. In: The International Conference on Case-Based Reasoning. Atlanta, GA, USA.

Bent, O., Dey, P., Weldemariam, K., Mohania., M. K., 2017. Modelling user behavior data in systems of engagement. Future Generation Computer Systems 68, 456–464.

Bezdek, C., J., 1981. Pattern Recognition with Fuzzy Objective Function Algorithms.

Böhmer, M., Ganev, L., Kruger., A., 2013. Appfunnel: A framework for usage-centric evaluation of recommender systems that suggest mobile applications. In: International Conference on Intelligent User Interfaces. Santa Monica, CA, USA.

Chen, C. P., Weng, J.-Y., Yang, C.-S., Tseng., F.-M., 2018. Employing a data mining approach for identification of mobile opinion leaders and their content usage patterns in large telecommunications datasets. Technological Forecasting Social Change.

Chen, X., Qiang, S., Wei, J., Jiang, K., Jin., Y., 2016. Passive profiling of mobile engaging behaviors via user-end application performance assessment. Pervasive and Mobile Computing 29, 95–112.

Deng, Z., Zhao, Y., Parvinzamir, F., Xia Zhao, e. a., 2016. Myhealthavatar: A lifetime visual analytics companion for citizen well-being. In: Edutainment 2016. Hangzhou, China.

Dunn, J. C., 1973. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. Journal of Cybernetics 3 (3), 32–57.

Foster, N. E., Anema, J. R., Cherkin, D., Roger Chou, e. a., 2018. Prevention and treatment of low back pain: evidence, challenges, and promising directions. The Lancet.

Fu, Y., Xiong, H., Lu, X., Yang, J., Chen., C., 2016. Service usage classification with encrypted internet traffic in mobile messaging apps. IEEE Transactions On Mobile Computing 15 (11).

Fulantelli, G., Taibi, D., Arrigo., M., 2015. A framework to support educational decision making in mobile learning. Computers in Human Behavior 47, 50–59.

Gener, M., Bilgin, G., zgr Zan, Voyvodaolu., T., 2014. Detection of churned and retained users with machine learning methods for mobile applications. In: International Conference of Design, User Experience, and Usability. Crete, Greece.

Gutierrez, F. J., Poblete., B., 2015. Sentiment-based user profiles in microblogging platforms. In: Acm Conference on Hypertext Social Media. Guzelyurt, Northern Cyprus.

Hartigan, J. A., Wong, M. A., 1979. Algorithm as 136: A k-means clustering algorithm. Journal of the Royal Statistical Society, Series C 28 (1), 101–108.

Hartigan, J., 1975. Clustering algorithms. John Wiley Sons.

Hartvigsen, J., Hancock, M. J., Kongsted, A., Quinette Louw, e. a., 2018. What low back pain is and why we need to pay attention. The Lancet.

Hastie, Trevor, Tibshirani, Robert, Friedman, Jerome, 2009. The Elements of Statistical Learning (2nd ed.).

Jensen, K. L., Larsen., L. B., 2007. Evaluating the usefulness of mobile services based on captured usage data from longitudinal field trials. In: International Conference on Mobile Technology, Applications, and Systems and the International Symposium on Computer Human Interaction in Mobile Technology. San Juan, Puerto Rico.

Kaufman, L., Rousseeuw, P., 1987. Clustering by means of medoids. Statistical Data Analysis Based on the L1Norm and Related Methods, 405–416.

Kodinariya, T., Dan Makwana, P., 2013. Review on determining of cluster in k-means clustering. International Journal of Advance Research in Computer Science and Management Studies 1, 90–95.

Kofod-Petersen, A., 2015. How to do a structured literature review in computer science. Petersen.

Lins, C., Hein, A., Halder, L., Gronotte., P., 2016. Still in flow long-term usage of an activity motivating app for seniors. In: IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom). Munich, Germany.

Ma, H., Cao, H., Yang, Q., Chen, E., Tian., J., 2012. A habit mining approach for discovering similar mobile users. In: International Conference on World Wide Web. Lyon, France.

Machado, D., Paiva, T., Dutra, I., Costa, V. S., Brando., P., 2017. Managing diabetes: Pattern discovery and counselling supported by user data in a mobile platform. In: IEEE Symposium on Computers and Communications. Crete, Greece.

Minelli, R., Lanza., M., 2013. Samoa a visual software analytics platform for mobile applications. In: IEEE International Conference on Software Maintenance. Eindhoven, Netherlands.

Nirmali, B., Wickramasinghe, S., Munasinghe, T., Amalraj, C., Dilum., H., 2017. Vehicular data acquisition and analytics system for real-time driver behavior monitoring and anomaly detection. In: IEEE International Conference on Industrial and Information Systems. Sri Lanka.

Parate, A., Jain, P., Kim., K.-H., 2016. Reckon: An analytics framework for app developers. In: The Workshop on Experiences in the Design and Implementation of Smart Objects. New York City, NY, USA.

Parwez, M. S., Rawat, D. B., Garuba., M., 2017. Big data analytics for user-activity analysis and user-anomaly detection in mobile wireless network. IEEE Transactions On Industrial Informatics 13 (4).

Pearson, K., 2010. 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.

Rousseeuw, P. J., 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Computational and Applied Mathematics 20, 53–65.

Schwarz, G., 1978. Estimating the dimension of a model. The Annals of Statistics 6 (2), 461–464.

Steinhaus, H., 1957. Sur la division des corps matriels en parties. Bull. Acad. Polon. Sci. 4 (12), 801–804.

Trauwaert, E., 1988. On the meaning of dunn's partition coefficient for fuzzy clusters. Fuzzy Sets and Systems 25, 217–242.

ur Rehman, M. H., Batool, A., Liew, C. S., Teh, Y.-W., ur Rehman Khan., A., 2017. Execution models for mobile data analytics. IT Professsional 19 (3).

Vankipuram, A., Vankipuram, M., Ghaemmaghami, V., Patel., V. L., 2017. A mobile application to support collection and analytics of real-time critical care data. Computer Methods and Programs in Biomedicine 151, 45–55.

Yang, M.-H., Ahuja, N., 1998. Gaussian mixture model for human skin color and its applications in image and video databases. Proc. SPIE 3656, Storage and Retrieval for Image and Video Databases VII, 458–466.

Yuan, B., Herbert., J., 2014. A cloud-based mobile data analytics framework: Case study of activity recognition using a smartphone. In: IEEE International Conference on Mobile Cloud Computing, Services, and Engineering. Oxford, UK.

Zanetti-Yabur, A., Rizzo, A., Hayde, N., Watkins, A. C., Rocca, J. P., Graham., J. A., 2017. Exploring the usage of a mobile phone application in transplanted patients to encourage medication compliance and education. The American Journal of Surgery 214.

Zheng, J., Ni., L. M., 2012. An unsupervised framework for sensing individual and cluster behavior patterns from human mobile data. In: Acm Conference on Ubiquitous Computing. Pittsburgh, USA.

Appendix

In this appendix, the omitted heatmaps of clusters in Chapter 4 are shown here.

The heatmaps of rest clusters of activity data are shown in Figure 6.1, Figure 6.2 and Figure 6.3.

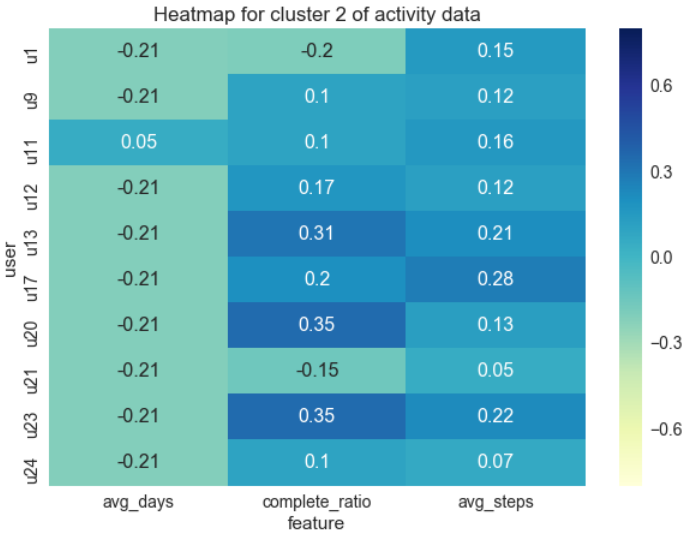


Figure 6.1: Cluster 2 of activity data

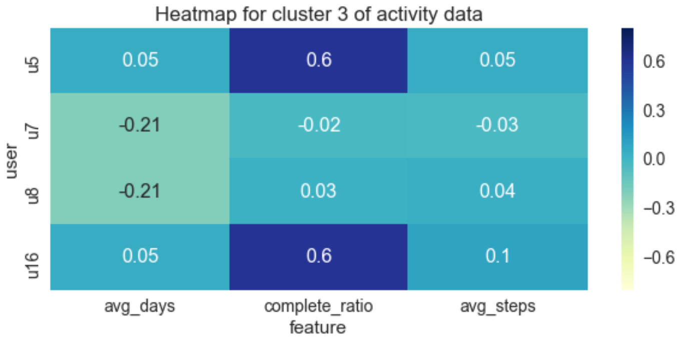


Figure 6.2: Cluster 3 of activity data

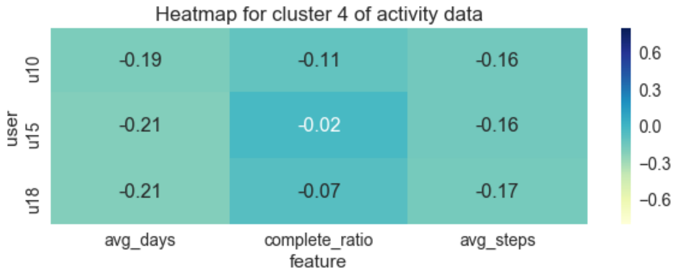


Figure 6.3: Cluster 4 of activity data

The heatmaps of rest clusters of education data are shown in Figure 6.4 and Figure 6.5.

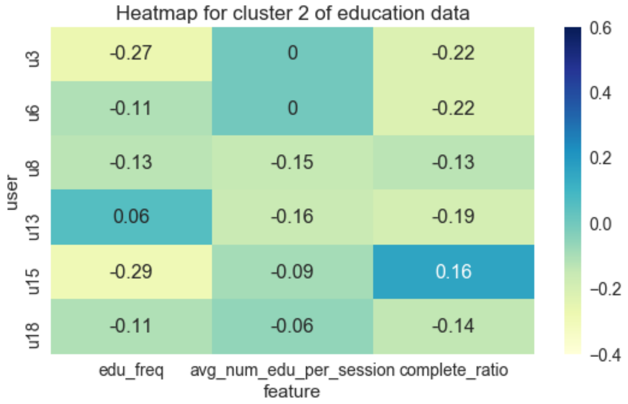


Figure 6.4: Cluster 2 of education data

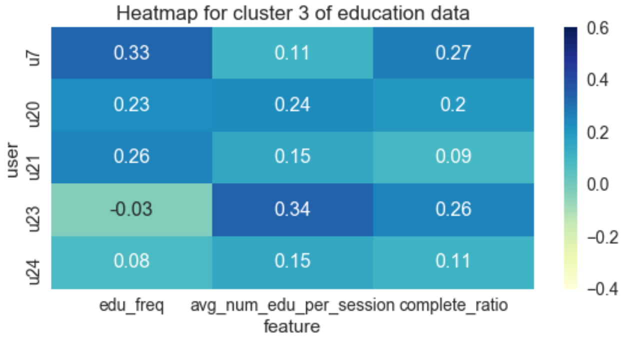


Figure 6.5: Cluster 3 of education data

The heatmaps of rest clusters of exercise data are shown in Figure 6.6, Figure 6.7 and Figure 6.8.

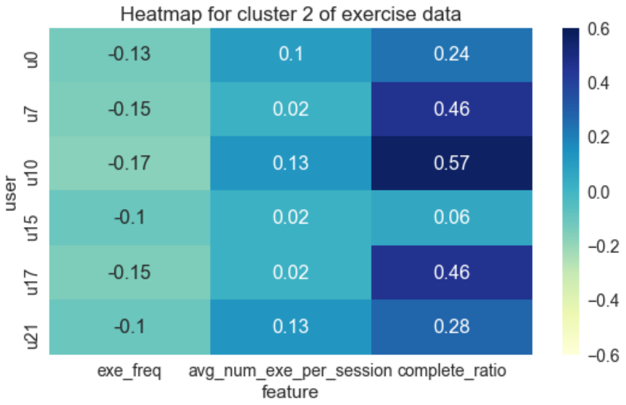


Figure 6.6: Cluster 2 of exercise data

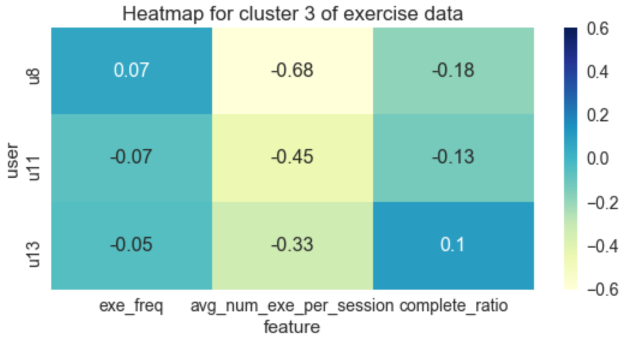


Figure 6.7: Cluster 3 of exercise data

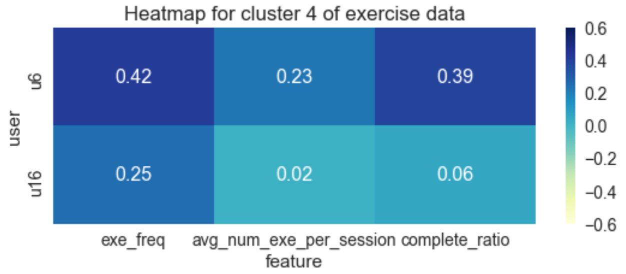
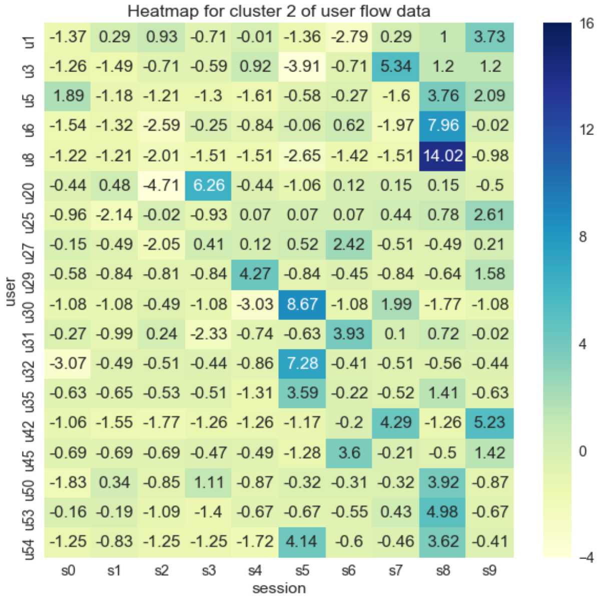


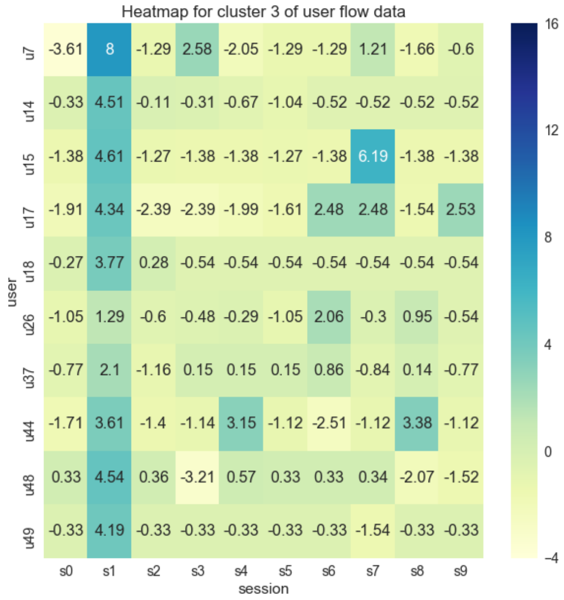
Figure 6.8: Cluster 4 of exercise data

The heatmaps of rest clusters of user flow data are shown in Figure 6.9 and Figure 6.10.



	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9
u1	-1.37	0.29	0.93	-0.71	-0.01	-1.36	-2.79	0.29	1	3.73
u3	-1.26	-1.49	-0.71	-0.59	0.92	-3.91	-0.71	5.34	1.2	1.2
u5	1.89	-1.18	-1.21	-1.3	-1.61	-0.58	-0.27	-1.6	3.76	2.09
u6	-1.54	-1.32	-2.59	-0.25	-0.84	-0.06	0.62	-1.97	7.96	-0.02
u8	-1.22	-1.21	-2.01	-1.51	-1.51	-2.65	-1.42	-1.51	14.02	-0.98
u20	-0.44	0.48	-4.71	6.26	-0.44	-1.06	0.12	0.15	0.15	-0.5
u25	-0.96	-2.14	-0.02	-0.93	0.07	0.07	0.07	0.44	0.78	2.61
u27	-0.15	-0.49	-2.05	0.41	0.12	0.52	2.42	-0.51	-0.49	0.21
u29	-0.58	-0.84	-0.81	-0.84	4.27	-0.84	-0.45	-0.84	-0.64	1.58
u30	-1.08	-1.08	-0.49	-1.08	-3.03	8.67	-1.08	1.99	-1.77	-1.08
u31	-0.27	-0.99	0.24	-2.33	-0.74	-0.63	3.93	0.1	0.72	-0.02
u32	-3.07	-0.49	-0.51	-0.44	-0.86	7.28	-0.41	-0.51	-0.56	-0.44
u35	-0.63	-0.65	-0.53	-0.51	-1.31	3.59	-0.22	-0.52	1.41	-0.63
u42	-1.06	-1.55	-1.77	-1.26	-1.26	-1.17	-0.2	4.29	-1.26	5.23
u45	-0.69	-0.69	-0.69	-0.47	-0.49	-1.28	3.6	-0.21	-0.5	1.42
u50	-1.83	0.34	-0.85	1.11	-0.87	-0.32	-0.31	-0.32	3.92	-0.87
u53	-0.16	-0.19	-1.09	-1.4	-0.67	-0.67	-0.55	0.43	4.98	-0.67
u54	-1.25	-0.83	-1.25	-1.25	-1.72	4.14	-0.6	-0.46	3.62	-0.41

Figure 6.9: Cluster 2 of user flow data



	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9
u7	-3.61	8	-1.29	2.58	-2.05	-1.29	-1.29	1.21	-1.66	-0.6
u14	-0.33	4.51	-0.11	-0.31	-0.67	-1.04	-0.52	-0.52	-0.52	-0.52
u15	-1.38	4.61	-1.27	-1.38	-1.38	-1.27	-1.38	6.19	-1.38	-1.38
u17	-1.91	4.34	-2.39	-2.39	-1.99	-1.61	2.48	2.48	-1.54	2.53
u18	-0.27	3.77	0.28	-0.54	-0.54	-0.54	-0.54	-0.54	-0.54	-0.54
u26	-1.05	1.29	-0.6	-0.48	-0.29	-1.05	2.06	-0.3	0.95	-0.54
u37	-0.77	2.1	-1.16	0.15	0.15	0.15	0.86	-0.84	0.14	-0.77
u44	-1.71	3.61	-1.4	-1.14	3.15	-1.12	-2.51	-1.12	3.38	-1.12
u48	0.33	4.54	0.36	-3.21	0.57	0.33	0.33	0.34	-2.07	-1.52
u49	-0.33	4.19	-0.33	-0.33	-0.33	-0.33	-0.33	-1.54	-0.33	-0.33

Figure 6.10: Cluster 3 of user flow data