PROJECT NOTTINGHAM: A STUDY OF USER BEHAVIOR ON MOBILE INVESTING APPLICATIONS

by

Harsh Manohar Gawai

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

at

Dalhousie University
Halifax, Nova Scotia
August 2022

This thesis is dedicated to my family, friends and Professors who supported me in every situation

Table of Contents

List of Tables

List of Figures

List of Abbreviations Used

DBSCAN Density-Based Spatial Clustering of Applications with Noise. 10, 25, 28, 30, 32

DTW Dynamic Time Warping. 11, 31

EGM Electronic Gaming Machines. 9

EPS Epsilon. 10

IG Information Gain. 12, 13

OLG Ontario Lottery and Gaming Corporation. 7

PCA Principal Component Analysis. 28

RG Responsible Gambling. 7, 8

SVM Support Vector Machine. 14, 36, 37, 44

Abstract

Investing or trading in the stock market has become a task just a button click away. Financial trading applications made this possible with their effortless and user-friendly design to buy and sell stocks. It has been observed that risky trading behavior on such applications has caused volatility in financial markets. This thesis explores the prediction of risky behavior using users actions on an investing application. To achieve this, we conducted an experiment where participants generated behavioral data from transactions with a simulated trading app. An unsupervised learning approach was undertaken to cluster users based on the time series data collected from the simulation. We identified distinct clusters of users based on app usage data which reflected degrees of risky behavior. To determine their risky behavior, assessment of the clusters based on the degree of intrinsic risk associated with their common actions, as well as responses to survey questions made by the participants before the task was conducted. The algorithm which distinguished the user behavior in most appropriate way was 'TimeSeriesKMeans'. The survey data were used as labels for classification task to explore the reliability of psychometric surveys to predict user behavior and 'SVM (Support Vector Machine)' and 'Logistic Regression' classifier provided the most accurate results among other classification algorithms. Moreover, the factors involved in user enjoyment from simulation were also explored. This work demonstrates a step towards identifying a method for conducting and assessing clustering and classification models for the purpose of risky behavior detection using psychometric measures for evaluation and the reliability of those measures for identifying risky behavior apart from the actual behavior.

Acknowledgements

I would like to thank my friends and family who were always been my support system. Big thanks to Soheil Latifi whose advices helped me to stay on right path through my thesis. I am also grateful to Samantha Taylor for providing additional support in the participant recruitment process. I would like to express my deepest gratitude to my co-supervisor Prof. Colin Conrad for his continuous guidance and mentoring not only in my thesis but also regarding career growth and taking life decisions. Lastly, I am extremely grateful to my supervisor Dr. Vlado Keselj for providing me with all the knowledge regarding NLP, supervising and guiding me in the proper direction to complete my thesis.

Chapter 1

Introduction

Financial Markets have always been volatile though today we live in an era wracked by extremes. In 2020, the global pandemic has provided us with the main reason why investments are necessary to provide security through hard times, though it also illustrated their high risk. During the pandemic, many investors invested in Bitcoin and received huge returns in 2021. After that, the market dropped dramatically during 2022 [2]. Investment holders should be prepared to handle the volatility of the market, though they are not always prepared for this.

In January 2021, financial markets were rocked by a new type of unexpected cause. Over the course of two weeks, millions of Reddit users leveraged popular financial trading apps such as Robinhood to propel the stock value of GameStop, a failing company, by 1,500% [21]. Though many app users profited from this activity, it also had a negative impact on hedge funds which had bet on the stock failing. Millions of Reddit users also lost money as the stock price subsequently crashed [11]. This activity subsequently caused stock hysteria in various similar assets, ultimately culminating in market volatility and calls for regulation of these apps [36]. The questions must be raised: what is causing people to conduct risky financial trading this way? Are there features of trading apps that influence risky decisions? If so, are there app designs that limit such risk taking behaviour?

The factors which causes such investing behavior might include that the users who are inclined towards investing in meme stocks or crypto-currency are risk takers. Economists believed that risk perception is a personality trait that can predict attitudes towards investments, as well as actual investment behavior such as saving for retirement. Another factor which causing this could be people enjoy using the application. Users' happiness and tendency to utilise applications are known to be influenced by hedonistic aspects like enjoyment, especially when social media is involved. This was proposed as a determinant of whether people engage in dangerous

behaviour on mobile investment platforms by Costola *et al.* [13].

From a behavioural perspective, the GameStop and Robinhood phenomenon is novel and multifaceted, incorporating social, technical, psychological, and user experience dimensions in an environment where large amounts of data can be collected easily. Researchers have explored factors that contribute to collective social behaviours on social media such as those exhibited on Reddit [38]. There are also well-established theories that investigate the role that enjoyment or cognitive absorption plays in information technology use [3], as well as the role that risk perception may play in driving risky and impulsive IT behaviour [39]. More recently, e-commerce researchers have applied similar techniques to the assessments of the effects of gamification on online shopping behaviour [14]. These social scientific approaches can offer considerable insight into the latent causes of risky behaviour, though offer relatively little insight into specific design factors that influence it.

By contrast, machine learning approaches could offer insight into the relationship between features and specific behavioural patterns in application use. For example, machine learning research has been conducted to model user navigation patterns [22], which might be improved using a combination of psychological data and modern machine learning techniques [19]. One possible reason that influences the user's behavior is persuasive gamification. In recent years, the Persuasive gamified systems have become more popular, especially when sustainable development is considered. These are mainly used in several areas such as managing the diseases, avoiding the behavior which includes high levels of risk, fitness, eating healthy foods, etc. These structures particularly address motivation, which allows within the behavioral traits that permits customers to work in positive ways. The persuasion systems that are gamified are more potent. It deals with the emotional angle of the user.

In light of behavioral analysis, there has been several work which analyzes the gambling patterns, risks and problems related with it. Peres *et al.* [26] have used machine learning in studying the data. The behavior traits are also being discussed with the gambling services. The data of the players collected and analysis of their behaviors is the main aim of the research. The creation of clusters depending on the gambling level helps to find more solutions and get a clear idea. The authors used clustering in depicting different kinds of gambling activities in different areas based

on different strategies.

Nowadays, gambling has become extensive because of the widespread use of communication platforms. Gambling comes with disadvantages, and it creates a problem for society. It affects the mental health of people and creates a disturbance at the psychological level in human beings. It also affects the economic situation in the country. The studies also suggested that the players should be self-aware of the effects of gambling. Through this process, they can reduce the effects. The usage of AI tools in the attempt of detecting problem players or risk takers will be helpful.

The Classification and Regression models were developed to list out the people addicted to gambling. Personal feedback related to the money spent by the player can also let the users change their perspective. By analyzing the past behavioral patterns of the user, we can predict the future scope of gambling of the individual. These patterns can even help limit-setting the money spent by players [4]. All in all, this gave us the motivation to study the user's behavior and their actions on any online system which relates to their risk taking decisions.

1.1 Research Problem

In this thesis, we describe a formative interdisciplinary study that combines behavioural and machine learning approaches to the prediction of risky financial trading behaviour. Inspired by the use of simulation games in the study of psychological immersion and flow [20], we developed a simulation of a stock trading app called "Project Nottingham" through which users bought and sold simulated currency in a fast-paced trading environment based on historical market data. The simulation is designed to mimic Robinhood's interface and included a variety of features that were analogous to their platform, such as real-time visualizations, easy trading options, and performance comparisons. Simulated trading behaviours were recorded and logged in a database and were used in this analysis. Though this represents only the first steps towards identifying factors that influence risky trading behaviour, the study contributes by offering a novel interdisciplinary approach to research in artificial intelligence. The second half of study involves clustering the users based on their risky behavior and the classification between behavioral or App-use features and target variables as survey data that we obtained from our simulation which helps in depicting the risks and

enjoyment levels of the investor. One of the predicted records mining techniques is classification. It aids in the identity of target audiences. Low risk, high risk, low enjoyment, and high enjoyment are the four groups focused in this study. It would also be beneficial to explore whether in-game bonus prompts factors in users' enjoyment. Bonus rounds or incentives are one of the persuasive strategies which influences users to utilize any game, application or system continuously. Our research questions can be articulated as follows:

1. Are there effective methods for clustering risky investment app trading behaviours based on user actions?

2. Are psychometric risk surveys useful as labels for classifying trading behavior?

3. Do bonus prompts influence users enjoyment on investing application?

The remainder of this thesis includes `Chapter 2` where we discussed the related work and the baseline topics which encouraged us to experiment with behavioral data. We also introduced some unsupervised and supervised learning models that was used in this thesis. In `Chapter 3`, the process flow from creating the application to performing Machine learning on collected data is explained. `Chapter 4` discusses the experiments and the results of clustering and classification of users based on their behavioral data. `Chapter 5` provides descriptive analysis of the data along with an analysis of the clustering results and the limitations of this work. At last `Chapter 6` derive conclusions from the results and discussed possible work that can be attained in future.

Chapter 2

Background and Related Work

In this literature review, we discussed studies related to user behavior on stock market, persuasive applications and problem gambling. We also explored it and how these various areas encouraged us to analyze data collected from a user study with the help of machine learning and psychological survey approaches.

2.1 Psychological approaches to simulation and gaming research

Plieger et al. [28] mentioned that with the consideration of the extraversion, the people with this personality trait tend to be more open to the different experiences, which helps in taking the risky decisions. Some factor other than personality is the environment. Mental stress and anxiety also affect the decision-making of a person. The profession of an individual can also lead to a difference in the decision making. For example, if an individual on a daily basis is adapted to the risks in their profession, then they have a high probability of taking the risky decisions. If these persons are compared to the person, who is doing the opposite kind of job, which has low-risk involvement.

There are many psychological factors that can lead to greater or reduced risk taking in applications. Plieger et al. conducted a study where they analyzed the behavior of different genders in the case of various asset allocations, while investing capital, volatile stocks, etc. [28]. They created a simulation of stock market and ran the study for three weeks while constantly taking updates and their thoughts about their personality, investment behavior, and life-stress using open-ended follow-up questions. The following study observes that the women are more concerned with losses which come about from time to time. Thus it could be analyzed that women are much more likely to make higher monetary selections after a protracted time.

Every simulation model will have limitations. There is a criticism that some of the parameters are constant such as simulation time and data of different stocks used [28].

These don't change on a realistic basis, which holds for the simulation software. But when compared to the behavioral assumptions this model is far more successful in the field of stock marketing. The task is not to be wholly dependent on the tool but to study the stocks carefully and invest in them. This process will lead to the reduction of the risks. The investment in stocks with adequate knowledge will lead to better savings and low-risk levels. Another issue with online simulation is that if we use the online questionnaire, it cannot detect the financial position of the investors but it helps to see how different are the user belief and their actions. This simulation tool helps in assessing financial outcomes and provides a subjective analysis of the stocks and helps in educating the investors to get the best deal out of their portfolios. Simulation is thus a promising method that includes both the individual factors and variables related to the trading behavior, which increases the predictive power of the individuals and results in the success in the trading arena.

Persuasive systems are the systems designed by studying and assessing the content of it using accepted psychological research theories and techniques. It aims to affect users' opinions and behaviour through social influence and persuasion. Such systems are mostly used in Human-Human or Human-Computer interactions. Decision taking behavior of any person varies based on different age groups, genders, and personality types. To get a better insight into the personalities, Ndulue *et al.* [24] investigated five different persuasive strategies and analyzed them. These strategies helped in finding the change in the behavior based on the rewards associated with the work, competition between the users, cooperation between the users, personalization, and lastly, the normative influence. In this, they have described that people with conscientiousness are highly motivated with the help of competition, norms, and rewards. Whereas emotionally stable people are motivated by cooperation. The competition builds the willingness inside the individual to compete with fellow users and to improve their standards. Thus in our simulation, we have added briefing page which comes up after the user finished with simulation rounds that shows the top 3 performers till now who made more profit out of the given money to start with to see whether users get influenced by the top performers to play the simulation again.

There also are a few obstacles within the persuasive gamified systems. As persuasive systems are mainly based on people's performance, it is necessary to reduce the

cheating and overuse of the system as it changes the main aim of the whole concept. Sometimes, due to over competitiveness, some users will cheat or spend a lot of time finding ways to perform better at the task, which can be affected mentally based on what kind of persuasive system they are using. The persuasive system can have a high impact on the investment side. As the investment is mainly a brain game, the software with the analysis of the risk patterns can change the perspective of the investors and it can help them to reduce the risks. The computation and depiction of the enjoyable levels of the portfolios can affect the investors. So the usage of these types of systems can have a bright future and high economic intelligence.

In regards of the user behavior in gambling area, Harrigan *et al.* [17] showed a case study about players in Ontario. The percentage of gamblers, when compared to non-gamblers, was nearly more than thrice. This study intended to provide a clear idea of the progression of players. The paper proposed that self-control games as a vital aid for the problems related to gambling. The utilization of the barriers displaying warning symbols has a tendency to have an effect on the conduct of the person and might assist in lowering the playing nature of a person. The psychological effect can bring out a major change in human activity. By making the gambling activity less fun or finding any other game which is more fun and healthy can reduce the problems that are being faced due to gambling.

Responsive gambling, referred to by Harrigan [17], is a commendable step toward changing the behavior. This model helped the Ontario Lottery and Gaming Corporation (OLG) in analyzing the crucial data of the players and in the creation of the potential data-sets regarding the players. Responsible Gambling (RG) helps in creating the focus groups. This initiative consists of a slot machine that will give information to the players about the risky choices they are making. Thus, in turn, it affects the behavior. The main characteristics of the slot machine model are the percentage at which the player gets payback, the bonus, and the volatility. The following work also provides the label classification. This type of system creates a sense of responsibility. A positive impact has been created with this type of model. The proposed version may be the muse step in the direction of the studies focused on this area.

Casinos are primarily made up of slot machines. At the time of their inception in

the twentieth century, slot machines did not require any special abilities. However, technological advancements and new games have ushered in a massive shift in recent years. Due to this, nowadays in the market, the skill is required. Chen *et al.* [12] have provided their studies on several areas regarding gambling, motivation, and demographics. The studies include the reasons for gambling in a separate section and analyze them. The vital ones include Ego-Driven, Learning/Evaluating, and Relaxation. The cluster-based analysis is done for a better understanding of the data.

The authors used questionnaires to accumulate the data. The questionnaire has mainly three types of questions. They are behavioral, reasons for choosing to gamble, the importance of the environment of the app, questions related to gender, income, etc. From the collected data, the authors have segregated the characteristics and the demographics. They used this separated data in analyzing the features. The researchers have asked different questions to the players like time spent on games, jackpot machines, themed games, etc. This study helps in managing the database of the players, their behavior, and even the players. The main limitation of the proposed work is that it is done through the minimum data collected. Nowadays, as online gambling becomes more popular, there is a high future scope in this industry.

As gambling comes with numerous risks, it becomes vital to check the negative aspect of the field. Protection of the people involved in gambling and minimizing the adverse impact becomes the checkpoint. The RG tools, as discussed, provide a clear set of limitations to the players and reduce the harm. If the players' gaming time is reduced automatically, it avoids the players from wasting a lot of money. Lots of studies are going on regarding the limit-setting strategy. This strategy is often treated as having a positive impact on players. The limit-setting can be done in the deposits, bets, loss, and amount of time invested by the player. If the limits are mandatory in gambling then it can reduce the harmful impact caused by gambling. Most individuals find fun in gambling and treat it as a legal and socially acceptable activity. Gambling is one of the concerns in today's world as it affects mental health. There is a lot of research and surveys conducted and analyzed in the gambling field. By keeping this in mind, our simulation provides users with limit of virtual $10,000 to start investing in assets. Their task was to make profit out of the provided money

and our task was to analyze their action that in the process of making profit whether they undertake risky behavior or not.

Another work that we influenced by was Latifi [1] research into Electronic Gaming Machines (EGM) and the problem gambling on it. He analyzed several data sets, such as data generated from lottery games using complex algorithms and studied the patterns and the consequences of the behavior. The author explores the data of users on gambling machines and applied clustering algorithms to identify customer personae based on their playing behavior. In addition to this, the clusters generated algorithm was used as labels for playing sessions to create a classifier which detects playstyle of the users. Overall, the work helped in detecting behavioral patterns on EGM. This work was effective at modeling user risk and their playstyle based on their behaviour, though there was no way to cross-reference those findings with psychological information about the users because they were anonymous. The development of the simulation tool regarding the stock market helps in assessing user risk in a different domain. Stock market simulation platforms can help in assessing trading patterns and analyzing the risks related to them, which also allowing us to gather psychological data from the users. Stock market simulations elicit risky decisions with the data available by encouraging users to find high-performing stocks. These simulations calculate the portfolio on a day-to-day basis and provides the information to the hypothetical or simulated investor.

Philander [27] used the supervised learning algorithms and data mining methods for identifying the risk involved in the case of online players. With the help of real-time monitoring and studying the information about the players, we can depict the behavioral pattern of the players. The studies on monitoring the player's information and figuring out the patterns in their conduct facilitates in depicting which of the gamers is exceedingly related to gambling. By using these methods, they have divided the players into three different categories. The categories are the players who are not further interested in the gambling area, dissatisfied players as a result of unforeseen events and gambling-related concerns, etc. The author used the rain forest algorithms, classification, and regression algorithms and compared the results of all three. In our study we undergo following approaches.

2.2 Unsupervised Learning

This section gives overview of the clustering models used to differentiate the users based on their actions.

2.2.1 DBSCAN

DBSCAN stands for Density-Based Spatial Clustering of Applications with Noise. The main components of DBSCAN is 'Core point', 'EPS', 'Minpts'. The algorithm starts with randomly selecting core point from all the data points. Then it will consider all its neighbouring points. Epsilon (EPS) is the radius of the core points under which it will consider its neighbouring points and the minimum number of points to be accepted in that core point's cluster is defined by 'Minpts'. For each point it creates a circular area of radius 'eps' and if two clusters having 'Minpts' under that circle are close to each other, they join to become a cluster. This step goes on till no other points are considered in the cluster. Then it jumps to another data points which were not included in the first cluster and starts the same process. This algorithm is sequential means if the datapoint is in similar distance with two different cluster then the first cluster which is in the process to add neighbours in it will absorb that point first. The points which are not included in any of the clusters created will be considered as 'Outliers'.

2.2.2 KMeans

K-Means is the oldest unsupervised machine learning techniques used to identify clusters from the dataset. It requires the number of clusters K to be specified in order to initialize the clustering from the data. The algorithm randomly selects K points as centroid of clusters and calculates the distance between centroid and each point from the data and then assigns the point with minimum distance to its respective cluster. It continues the next iteration by computing mean of the each clusters as centroid points and again perform the same steps. The algorithm stops when the centroid value doesn't change. On the other hand, if the algorithm enters in to the loop of switching between sets of centroids, another criteria is added for stopping such as iterations limit or stopping when there is very small difference between centroid sets.

The computational complexity of the algorithm depends on the value of K.

2.2.3 TimeSeriesKMeans

TimeSeriesKMeans is a clustering algorithm which works better for clustering time series data. It uses several distance metrices used to calculate similarity between two data points out of which 'Euclidean' and 'Dynamic Time Warping' are very popular. Euclidean distance is the common approach for calculating distances in standard clustering algorithms. It works as calculating point to point distance between two similar lengths of time series. On the other hand, Dynamic Time Warping (DTW) can also calculate distance of time series data of dissimilar length. Euclidean distance does not work better for time series, it is invariant to time shifts. For instance, if two time series is highly similar but one of them is shifted even a little then it will consider it whole differently while DTW is suitable for time series which are not aligned exactly in time or length. Distance formula for DTW [37, 30] is given by 'The squared root of the sum of the squared distances between each element in x and its closest point in y.':

$$DTW(x, y) = \min_{\pi} \sqrt{\sum_{(i,j)\in\pi} d(x_i, y_j)^2}$$

where $\pi = [\pi_0, \ldots, \pi_K]$ is a path that satisfies the following properties:

- it is a list of index pairs $\pi_k = (i_k, j_k)$ with $0 \leq i_k < n$ and $0 \leq j_k < m$
- $\pi_0 = (0, 0)$ and $\pi_K = (n-1, m-1)$
- for all $k > 0$, $\pi_k = (i_k, j_k)$ is related to $\pi_{k-1} = (i_{k-1}, j_{k-1})$ as follows:

  1) $i_{k-1} \leq i_k \leq i_{k-1} + 1$

  2) $j_{k-1} \leq j_k \leq j_{k-1} + 1$

The algorithm fits all the time series data in the model and predicts the closest cluster that each time series belongs.

2.3 Supervised Learning

2.3.1 Decision Tree

Decision Tree is a classification algorithm used to distinguish data or items into different categories. There is an internal node which represents a feature, a branch represents a decision rule, and each leaf node indicates the conclusion in a decision tree, which resembles a tree structure. It gains the ability to divide data according to attribute values. It follows recursive partitioning which is the process of repeatedly dividing a tree. The decision tree is a non-parametric or distribution-free strategy that does not rely on the assumptions of a probability distribution. The algorithm selects attributes based on attribute selection measure to split the data. The common selection measures are Information Gain, Gain Ratio and Gini Index.

Information Gain refers to decrease in entropy where entropy is randomness or impurity of input set. Based on the values of the specified attributes, information gain calculates the difference between the average entropy after splitting and the entropy before splitting of the dataset.

$$Entropy(D) = -\sum_{i=1}^{m} p_i \cdot \log_2 p_i$$

where, p_i is the probability that any record in D (i.e.target column) belongs to class C.

$$IG(D, A) = Entropy(D) - \sum_{j=1}^{V} \frac{|D_j|}{|D|} \cdot Entropy(D_j)$$

where, D is the target column, A is the variable column for testing and j refers to each value in A column. The variable A with highest Information Gain (IG) will be selected as splitting attribute at any given node.

Information Gain is biased towards choosing attributes with large values such as some kind of id or code. Gain Ratio is the modification to IG by considering the intrinsic information of a split

$$GainRatio = \frac{InformationGain}{Entropy}$$

The selection process will include getting information gain for all attributes and then calculating the average IG. After that, calculating the gain ratio of all attributes

whose IG is greater than or equals to calculated average IG. The attribute with the higher gain ratio will be selected for splitting.

Gini index calculates the probability of a feature which is classified incorrectly when selected randomly. It ranges from 0 to 1, where 0 indicates that all elements belongs to specified class while 1 represents that elements are distributed randomly across classes. The value 0.5 shows the uniform distriibution of elements across classes.

$$GiniIndex = 1 - \sum_{i=1}^{n} (P_i)^2$$

where, P_i indicates the probability of an element classified for a distinct class.

2.3.2 Random Forest

Random Forest classifier includes a large number of separate Decision Trees that work as an ensemble. Each Tree in the random forest gives class prediction and the class or label with highest number of counts becomes the model's prediction. Random Forest utilizes 'Bagging' which is an ensemble technique where random sampling of data with replacements and repetition but having the same number of data in every sample takes place and passed into each decision trees. Thus, the model is trained independently and the resultant output is based on the majority voting from the results of all the models. Random Forests are comparatively slow as it requires more computational time due to number of decision trees used. The main hyperparameters used are 'n_estimators' which is the number of trees algorithm builds, 'max_features' which includes the maximum number of features it uses to split a node and 'min_sample_leaf' which states the minimum number of leaves required to split an internal node.

2.3.3 Logistic Regression

Logistic Regression is statistical learning method which measures the relationship between dependent and independent variable by approximating probabilities using logistic function such as 'Sigmoid'. The formula for Sigmoid function is shown as follow:

$$P(Y|X) = \frac{1}{1 + e^{-f(x)}}$$

where $f(x)$ is a function with all the features (x) and their coefficient (β) in a linear form,

$$f(x) = x_0 + x_1\beta_1 + ... + x_k\beta_k + \epsilon$$

Here ϵ represents the constant or noise. Logistic Regression is representeed in a way that linear regression is defines using equation of a straight line. The difference here is that the output would be binary. The equation for logisic regression is given by:

$$y = \frac{e^{b_0+b_1X}}{1 + e^{b_0+b_1X}}$$

2.3.4 GaussianNB

Naive Bayes classifier is based on Bayes theorem which is used to calculate conditional probability.

$$P(A|B) = \frac{P(A) \cdot P(B|A)}{P(B)}$$

where $P(A)$ is probability of A occured, $P(B)$ is probability of B occured, $P(A|B)$ is probability of A given B and $P(B|A)$ is probability of B given A.

The classifier is useful when the dimensionality of data is high. It assumes that the features are independent in nature with each other. In order to classify any data point to respective class, it calculates the posterior i.e., individual probability of the class and likelyhood of features given that class. It calculates same for all the classes and the one with highest value as output will be assigned to that data point as label. The module in 'sklearn' named 'GaussianNB' performs the Naïve Bayes classification task.

2.3.5 Support Vector Machine

SVM is a supervised learning algorithm that helps in computing the perfect decision boundary with which the segregation of the data can be done. Through SVM, we can get a hyperplane with the extreme points which are called support vectors. If the data set can be classified into two sets using simply a straight line called decision boundary, then we can use the linear SVM. If not, then we can use non-linear SVM. The best hyperplane is the one whose to the nearest element of both side is highest. SVM is highly effective and works well, and is also more efficient. The main disadvantage is that it is not compatible with a large amount of data.

Chapter 3

Methodology

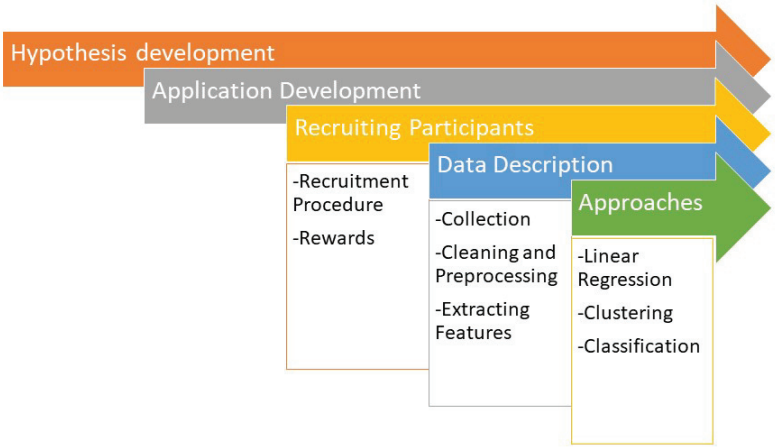

Figure 3.1: Methodology Process

Figure 3.1 describes the process undertaken in order to conclude our research questions in this section. It starts with description of development of the simulation for the study followed by recruitment procedure of participants, collecting data and extracting valuable features and at last applying machine learning models on the gathered data.

3.1 Hypothesis development

Our research questions were based on hypothesis that can be articulated as follows:

- H1 - Users' risky trading behavior and investing decisions are distinguishable on a simulation

- H2 - Risk appetite and enjoyment of the user can be predicted by risky decision on the investing application

The idea behind the second hypothesis is to explore whether the psychometric surveys are reliable as labels to analyze risky decisions and enjoyment. To gain empirical evidence for these hypotheses, we opted to create a simulation of a trading application and analyze usage data from the simulation. The features extracted from the simulation about the investing behavior of the user which are the indicators of their risky decisions were used as an independent variables to determine whether users can be distinguished based on their risky decision or not which helps to prove first hypothesis. Survey data asked before and after the simulation contained information about risk appetite and enjoyment of the users. Linear Regression helped to find out if the features were able to predict users risk propensity; i.e., risk survey data, then the we can rely on survey data as an indicator to users risky decisions. In addition to this, if the features were able to classify the survey data using supervised learning methods then it will be strong evidence to support second hypothesis along with linear regression results.

3.2 Application development

The initial aspect of this research was to create a web application that simulates and simplifies stock trading task. To accomplish this, we used 'Plotly Dash' which is an open-sourced library in Python for developing dashboards and customized user interface. Dash is written on top of `Flask` [16], `Plotly.js` and `React.js`. We used *SQLAlchemy* [7] which is an Object Relational Mapper Python toolkit to perform database operations. The simulation leverages stock market indices, equities and cryptocurrency data. The application was hosted on 'Heroku' which is a cloud platform as a service. The application also administered a survey before the task and one following the task, as well as a debriefing screen. The Robinhood investing application is adaptive in terms of accessibility in all devices, displaying stock graph and portfolio value in a single page which includes your total number of shares, equity value and average cost. It is user friendly in terms of interaction and performing actions such as buying and selling with a single click. To protect the ecological validity of the human-computer interaction tasks, the application design adhered as much

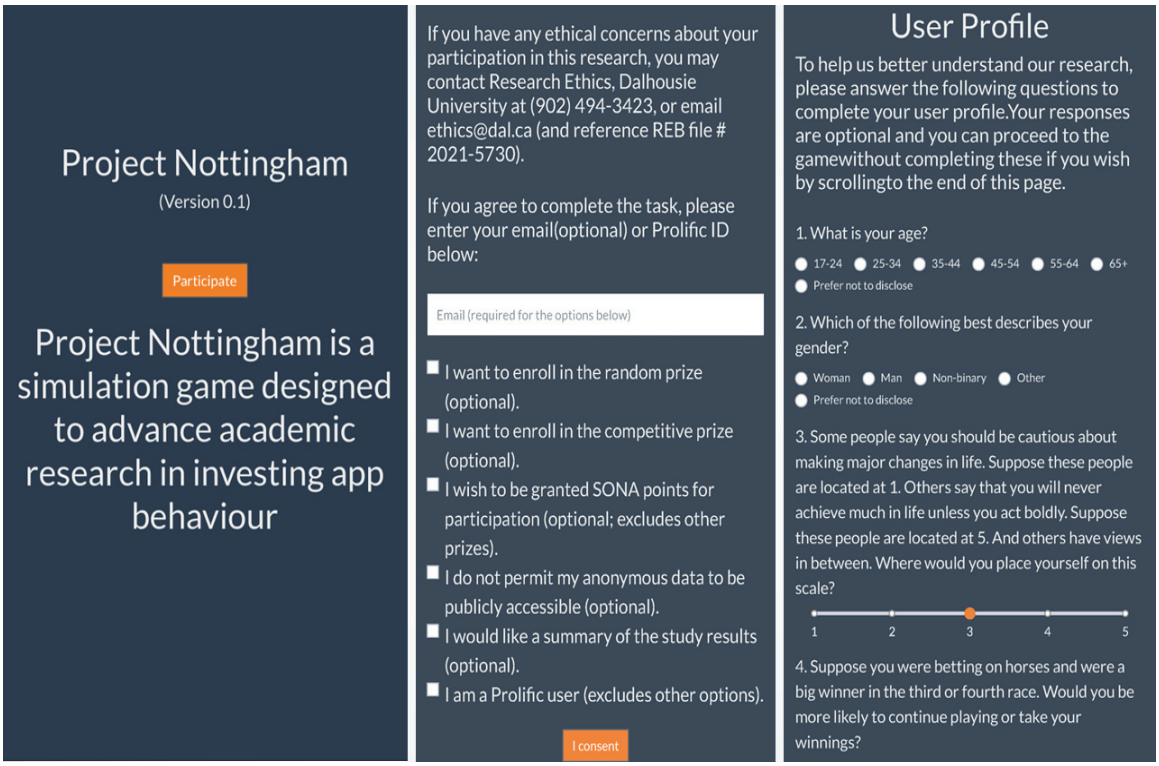

Figure 3.2: Sample screens of the web app, designed for mobile. 1) Left: Welcome screen 2) Center: Consent screen 3) Right: Pre-game survey

as possible to that of the Robinhood app and collected data concerning the users' responses to survey questions and application use transactions (i.e. buy, sell).

Figure 3.2 shows the contents that were included in the application when the user goes to the application link. The first screen is the welcome screen followed by research consent and then survey about the users' risk propensity before the task. To understand the connection between the users perception about risk and their actions or decisions reflects risk or not we came up with survey to log their perception and experience. The risk survey consisted of 12 questions which included 7 previously validated questions about general risk appetite [18], 3 questions about willingness to gamble lifetime income [6], as well as demographic questions such as Age and Gender.

After submitting the survey, the application took user to main simulation page Figure 3.3 where there were two tabs namely Instructions and Invest. Instructions tab had details about the assets user can buy and brief tasks about the simulation which can be performed by users. In the invest tab participants can start the simulation and carry out actions such as buying and selling stocks. The assets were based on historical

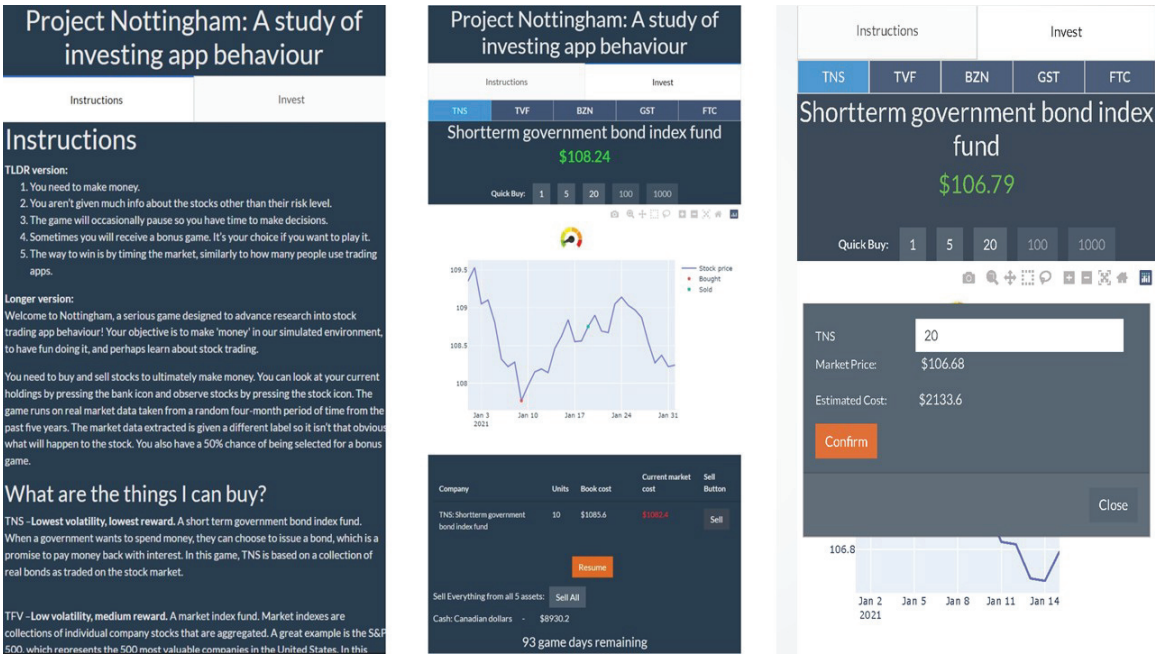

Figure 3.3: Sample screens of the web app, designed for mobile. 1) Left: Instructions 2) Center: Stock Simulation 3) Right: Sell button prompt

stock and cryptocurrency data. They were limited to five options and were placed in order of volatility: government bonds (lowest volatility, lowest reward), a major stock index fund, a single bank stock, a meme stock (e.g. Gamestop, Blackberry), and a cryptocurrency (e.g. Bitcoin, Litecoin; most volatile, highest reward). Users were not provided with the name of actual stock asset whereas the asset and the time period they were presented were generated randomly every time user starts a new session. Users were provided with the portfolio of different assets on the same screen just by switching over 5 distinct tabs of assets. The graph updated every 1500ms. They had option to enter the number of stocks they wanted to sell by clicking 'Sell' button or can sell everything from their portfolio by clicking 'Sell All'. In the sell prompt, the first field is where user entered the number of stocks to sell, 'Market Price' indicates the current price of single stock of the asset and the 'Estimated Cost' shows the total amount of number of stocks user entered for present simulation day. The numbers above buying buttons showed the current stock price of the stock as the graph moves. The Green and Red color indicated the increase or decrease in stock price compared to previous day respectively. On the bottom of the screen, the days remaining till the end of the simulation to invest was displayed.

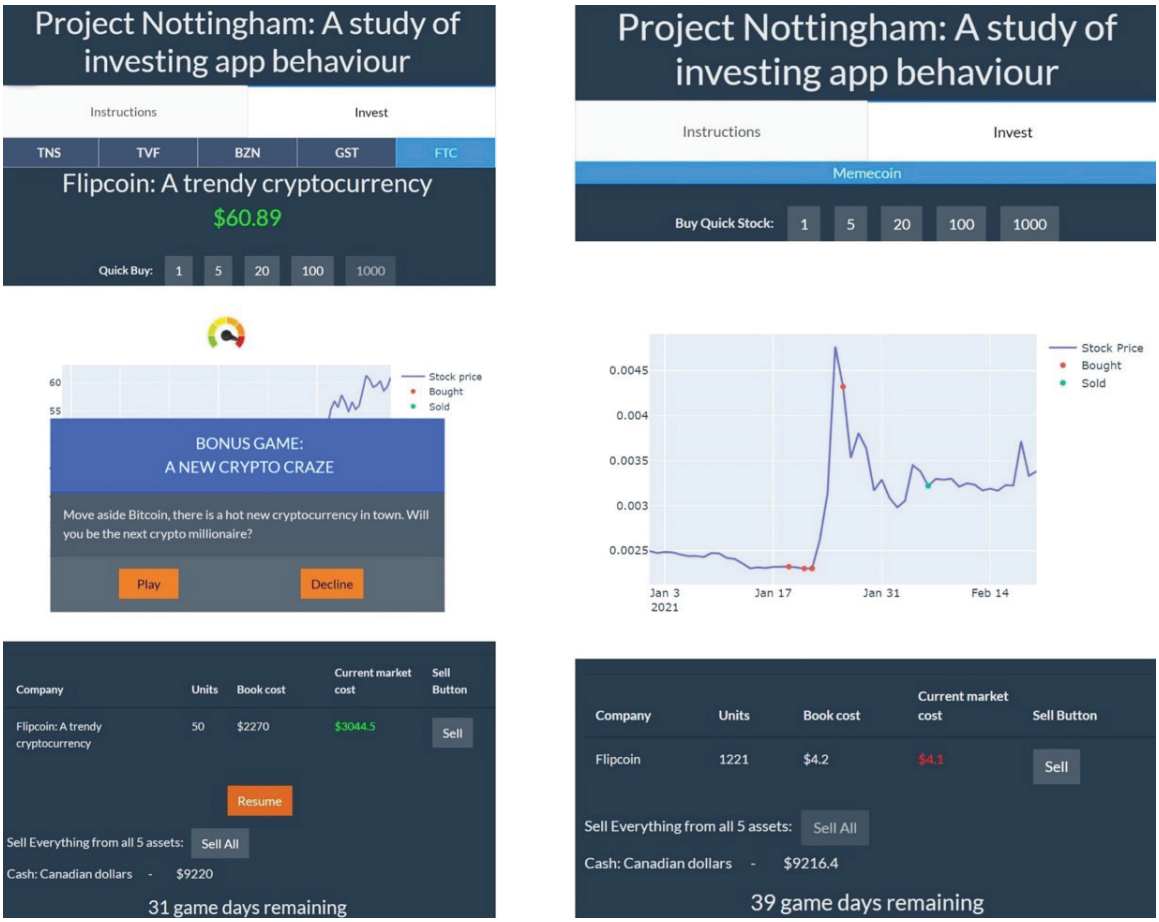

Figure 3.4: Sample screens of the web app, designed for mobile. 1) Left: Bonus Popup 2) Right: Bonus round

As the participant uses the application, Figure 3.4 shows a bonus pop-up prompted to them based on their engagement and buying behavior on the simulation. It includes to invest on the latest cryptocurrency. They had the option to accept or decline the bonus round.

Figure 3.5 represents that after the simulation days finished user had given an optional survey to fill which includes their inclination and enjoyment about the simulation. Submitting the survey leads participants to the briefing screen where they can also check out the average and top 3 portfolio value from users till now. A table describes the participants portfolio at the end of the simulation. They were given the option to play the simulation again by clicking the `'Play Again'` button.

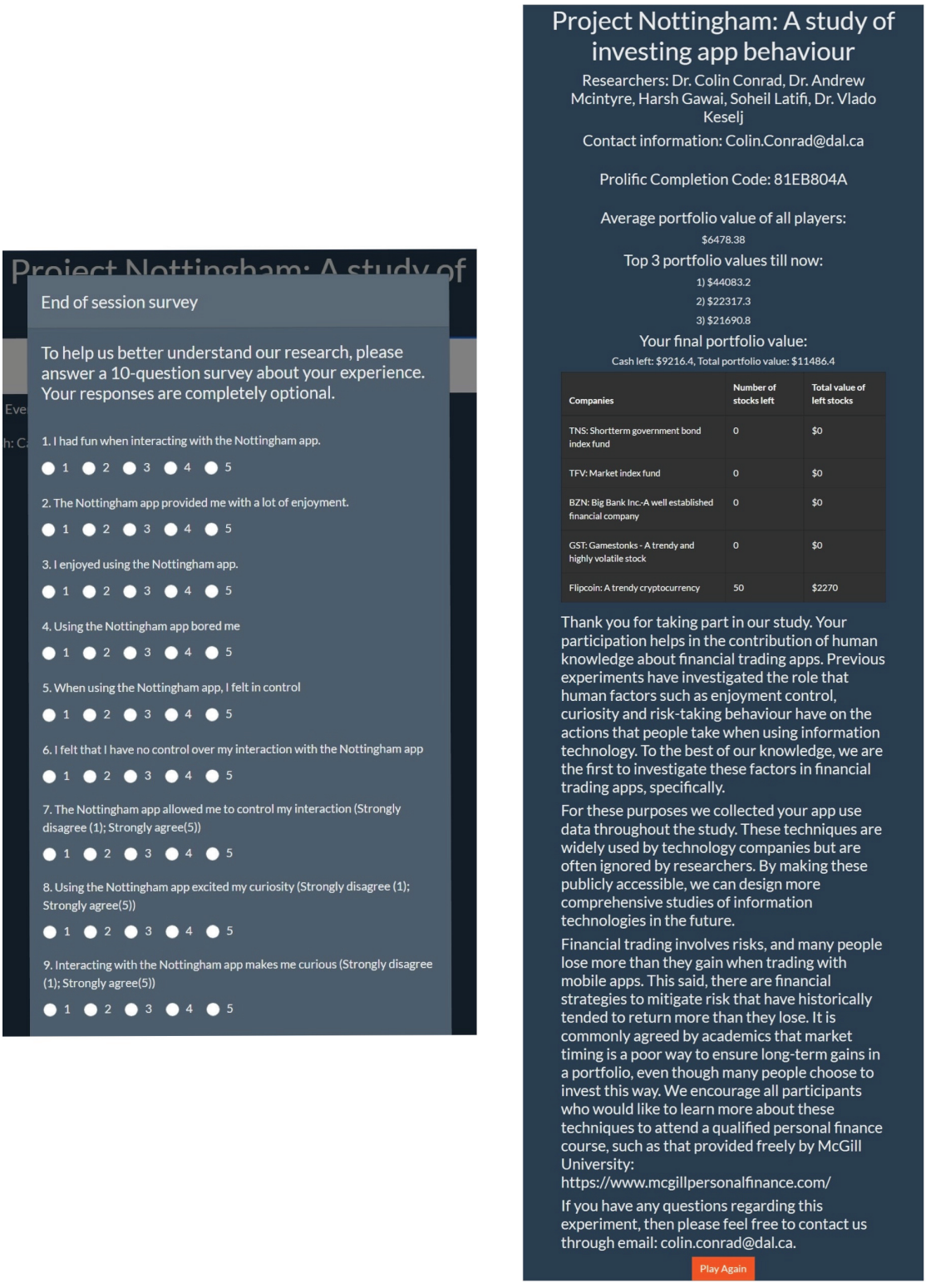

Figure 3.5: Sample screens of the web app, designed for mobile. 1) Left: Post-Session Survey 2) Right: Briefing Screen

3.3 Participants and Procedure

Participants were recruited through email, social media and paid panel. We contacted students through our labs and faculty mailing lists and invited them to participate for a chance to win a random draw of a $50 CAD gift certificate, as well as a competitive prize granted to the top 3 performers. We also recruited participants on the Prolific platform, who were each paid £2 for their time and compensated regardless of whether they completed the task. After receiving and clicking a link of the application, participants were presented with a consent screen, and consent was given by agreeing to proceed past the consent screen. Participants then completed the risk questionnaire and completed the stock trading simulation, which consisted of four rounds consisting of 46–47 seconds during which participants could buy and sell simulated assets, which were arranged in order of volatility. The simulation was paused in between each round and resumed by the participant at will which gives them chance to analyze their portfolio and take actions based on it for their future investments. The second survey at last concerned the enjoyment that participants experienced and consisted of 11 questions based on a well-known enjoyment measure [3]. We did not collect any identifying information on either survey.

3.4 Data Description

The data that were collected from the application was comprised of 4 different tables namely `'Userinfo'`: the pre-game demographic and risk perceptions data, `'Sessioninfo'`: Played Session information, `'Postsurveydata'`: Post-session survey(Enjoyment) information, and `'Playbehavior'`: Buy/Sell transactions information respectively. Figure 3.6 shows the sample of first table where we stored users demographic and survey information. Each user has been identified with unique id `"userid"` column. Figure 3.7 describes the second table where 'start-time' means the time when user clicks `'Invest'` button and 'count' means number of times user played the session. The columns 'amountLeft' and 'portfolioValue' includes the remaining balance at the end of the session and the total of remaining balance + cost of invested assets that are not sold respectively. The session played by each user had been differentiated by column `"sessionid"`.

userid	Email	consents	Age	Gender	Que3	Que4	Que5	Que6	Que7	Que8	Que9	Que10	Que11	Que12
295	None	[]	17-24	Woman	4.0	5.0	5.0	5.0	5.0	5.0	3.0	Yes	No	Yes
515	None	None	NaN	NaN	3.0	3.0	3.0	3.0	3.0	3.0	3.0	NaN	NaN	NaN
575	None	None	NaN	NaN	4.0	3.0	3.0	3.0	3.0	3.0	3.0	NaN	NaN	NaN
935	None	None	17-24	Woman	4.0	5.0	3.0	1.0	2.0	5.0	1.0	Yes	Yes	No
985	None	None	17-24	Woman	5.0	5.0	5.0	5.0	5.0	1.0	3.0	No	No	No
1145	None	[1]	17-24	Woman	3.0	3.0	5.0	1.0	4.0	4.0	2.0	No	No	No
1185	None	[1, 2, 3, 4, 5]	17-24	Man	4.0	4.0	5.0	5.0	5.0	5.0	4.0	Yes	Yes	Yes
1325	None	[1, 2, 3, 5]	25-34	Woman	4.0	2.0	2.0	3.0	2.0	3.0	3.0	No	No	Yes
1505	None	[1, 2, 3, 4, 5]	NaN	NaN	3.0	3.0	3.0	3.0	3.0	3.0	3.0	NaN	NaN	NaN
1525	None	None	NaN	NaN	3.0	3.0	3.0	3.0	3.0	3.0	3.0	NaN	NaN	NaN
1575	None	None	25-34	Man	3.0	2.0	4.0	4.0	4.0	2.0	2.0	No	No	No
1595	None	None	25-34	Man	3.0	2.0	4.0	3.0	3.0	2.0	3.0	No	No	No
1635	None	None	NaN	NaN	3.0	3.0	3.0	3.0	3.0	3.0	3.0	NaN	NaN	NaN

Figure 3.6: Userinfo table

	sessionid	user_id	starttime	counts	amountLeft	portfolioValue	bonusRoundPlayed
22	405	1355	19:52.2	1	5884.9	5884.9	Yes
159	2235	4785	51:33.2	1	13.8	10394.0	No
160	2305	5085	52:08.4	1	17654.0	17654.0	No
163	2335	5215	15:08.9	1	10786.7	10786.7	No
166	2365	5145	17:10.6	1	10064.2	10064.2	Yes
170	2405	5345	24:59.1	1	8063.7	9729.3	No
176	2465	5405	15:09.1	1	9920.3	9920.3	Yes
231	3015	7295	19:16.8	1	8304.2	10439.4	No
249	3195	7305	43:27.1	5	2296.4	10094.2	Yes
325	3965	8735	05:32.1	1	9540.2	9540.2	Yes

Figure 3.7: Sessioninfo table

The next table in Figure 3.8 has all the post-session survey answers related to 'Enjoyment' whereas the Figure 3.9 shows the table where information related to investments are stored. Every transaction that user has made are associated with `"playid"` column. The 'timestamp' columns depicts the time when user clicked button to buy/sell the stock. The 'company_name' has the name of the assets of which user bought stock on 'game_day'th day of the simulation. The columns 'buy_sell' and 'buy_sell_count' states which action did user performed and how much stock did he/she bought at that particular time respectively. Moreover 'market_price' includes

price of single stock of associated asset that user bought or sold while the $total\_cost = buy\_sell\_count * market\_price$ which shows that this much amount had user bought or sold in a single transaction.

	postid	user_id	session_id	Que1	Que2	Que3	Que4	Que5	Que6	Que7	Que8	Que9	Que10
0	45	775	245	5.0	3.0	5.0	2.0	5.0	4.0	5.0	5.0	4.0	4.0
1	115	1025	315	3.0	4.0	3.0	1.0	3.0	3.0	3.0	3.0	4.0	5.0
2	125	1355	405	5.0	2.0	3.0	4.0	4.0	5.0	3.0	4.0	2.0	5.0
3	135	1955	665	2.0	3.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	145	2175	725	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
5	235	2205	755	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
6	245	2205	765	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
7	255	2205	775	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
8	265	2265	805	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	5.0
9	285	2775	965	4.0	3.0	4.0	1.0	4.0	1.0	4.0	4.0	5.0	4.0

Figure 3.8: Postsurveydata table

playid	session_id	timestamp	company_name	game_day	buy_sell	buy_sell_count	market_price	total_cost
5895	625	2021-11-19 18:34:01.324	TNS: Shortterm government bond index fund	5	Buy	20	80.22	1604.40
5905	625	2021-11-19 18:34:03.471	TNS: Shortterm government bond index fund	6	Buy	1	80.19	80.19
5915	625	2021-11-19 18:34:16.604	TNS: Shortterm government bond index fund	15	Sell	6	80.00	480.00
5925	625	2021-11-19 18:34:38.569	TNS: Shortterm government bond index fund	30	Sold All	15	80.05	1200.80
5935	625	2021-11-19 18:34:38.569	TFV: Market index fund	30	Sold All	0	22.99	0.00
5945	625	2021-11-19 18:34:38.569	BZN: Big Bank Inc.-A well established financia...	30	Sold All	0	64.45	0.00
5955	625	2021-11-19 18:34:38.569	GST: Gamestonks - A trendy and highly volatile...	30	Sold All	0	19.41	0.00
5965	625	2021-11-19 18:34:38.569	Flipcoin: A trendy cryptocurrency	30	Sold All	0	3665.30	0.00
5975	625	2021-11-19 18:34:45.376	TNS: Shortterm government bond index fund	33	Buy	1	80.06	80.06
5985	625	2021-11-19 18:34:47.401	TNS: Shortterm government bond index fund	35	Buy	100	80.03	8003.00
5995	665	2021-11-19 19:53:35.273	TNS: Shortterm government bond index fund	2	Buy	5	78.26	391.30

Figure 3.9: Playbehavior table

We calculated the relevant features from the dataset mentioned in the figures 3.7

and 3.9 that were used as input for the models. The features shown in the figure 3.10 are extracted that were likely to be the indicative for 'Risk Propensity' of participants play behavior while using the application and similarity with the features used for the users data of gambling machine analysis [1]. As the application had 3 pauses on `31st`, `62nd` and `93rd` simulation day, we got total 4 rounds. So we refined the data into 5 different tables as `'Round 1'` — data between game day 1 to 31, `'Round 2'` — data between game day 32 to 62, `'Round 3'` — data between game day 63 to 93, `'Round 4'` — data between game day 94 to 124 and 'Total Rounds' of the mentioned features where the last table contains the summation of all four rounds of extracted data.

Features	Description
TRANSACTIONS_PER_MINUTE	Number of Transactions users made per minute
STOCKS_PER_MINUTE	Total number of stocks bought per minute
COSTBUY_PER_MINUTE	Total amount of stocks bought per minute
BOUGHT	Total number of stocks bought in a single session
NUMBER_OF_TRANSACTIONS	Total number of transactions made in a session
BUY_VALUE	Total amount of stocks bought in a session
MIN_BUY_AMNT	Minimum amount of stock bought in a session
MAX_BUY_AMNT	Maximum amount of stock bought in a session
RISK_METER	Frequently bought stocks of asset from 5 assets
TNS_AVGTIME	Average time taken between buy and sell of "Lowest volatility, Lowest reward" assets
TFV_AVGTIME	Average time taken between buy and sell of "Low volatility, Medium reward" assets
BZN_AVGTIME	Average time taken between buy and sell of "Medium volatility, Medium reward" assets
GST_AVGTIME	Average time taken between buy and sell of "High volatility, High Potential reward" assets
FLIPCOIN_AVGTIME	Average time taken between buy and sell of "Highest volatility, Highest Potential reward" assets
PROFIT/LOSS	Profit or Loss
RISE_BUY	Number of times user bought stocks when it was increased compared to previous day
FALL_BUY	Number of times user bought stocks when it was decreased compared to previous day

Figure 3.10: Extracted Features

3.5 Machine Learning Approaches to Data Analysis

This section provides the brief overview of the machine learning approaches such as unsupervised and supervised learning methods used with our collected data.

3.5.1 Clustering

As per our goal, we had to segment participants according to their 'Risk Propensity' such as 'High Risk' or 'Low Risk' based on their play behavior on simulation. This differentiation can be carried out with the help of Clustering Algorithms. The algorithms that we've selected for our experiments are `K-Means` clustering, `DBSCAN`, and `TimeSeriesKMeans`. We also tried Spectral clustering and Birch but they gave similar results to that of DBSCAN and Kmeans. The clusters that we obtained from the models was being analyzed with all the features and the 'High Risk' and 'Low Risk' potential behavior has been determined for that clusters. The most important thing that we investigated is comparing the clusters with the data that were calculated based on the psychometric scale from the pre-session survey which ultimately are the indication of their perception about risk. Apart from this, we explored whether there is any influence of features and clusters on `'Enjoyment'` of the participants by comparing those with post-session survey results which sums up how much participants had enjoyed the simulation.

3.5.2 Classification

The main idea behind the classification task is to figure out the risky behavior from the survey data as well as from engagement style on simulation. We did two experiments on this task. In first one, we took features as mentioned in figure 3.10 of 'Total Rounds' and the labels as pre-session survey questions which are summed up to get a value and after determining the threshold by choosing median of the data, we split the data based on people with 'High Risk' and 'Low Risk' perception as labels. For the second one, we used the same features but the labels as post-session survey labels which shows people with 'High Enjoyment' and 'Low Enjoyment'. This experiment helps us to predict the 'Enjoyment' of the participants based on their engagement on the simulation.

Chapter 4

Machine Learning Experiments

After the completion of our study on simulation, we managed to gather 254 users out of which 147 users actually completed the session on the application and the session data that we used to perform our analysis had 214 records. The reason for higher number of records compared to users is a single user had played session more than once. Due to the small number of data points we had, we were restricted by not using the complex models or Neural Networks.

4.1 Data Preprocessing and Feature Selection

The users who completed the sessions were filtered out based on the 'portfolioValue' in 'Sessioninfo' table as it was update at the end of the session when player finished the simulation or else it contained 'Null'. As some of the features in our dataset had different range of values like 'risk_meter' or 'transactions_per_minute' was in the range of tens, 'stocks_per_minute' were in the range of hundreds, and 'buy_value' or 'costbuy_per_minute' were in the range thousands, so it is crucial to normalize the data before performing supervised or unsupervised learning on it as the high range values will affect more on the results. To overcome this, we used 'MinMaxScalar' from sklearn package. It converted the data between [0,1]. It preserves the shape of the distribution of the data and the information present in the original data is not materially altered.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

We visualized the correlation of features with each other using heatmap. This helped us to eliminate one of the feature which are highly correlated with each other. From the figure 4.1, the red color indicates that the features are highly correlated with each other, so we eleminated 'Bought', 'Number_of_Transactions', 'Buy_Value', and 'Max_Buy_Amnt' as these features were highly correlated with other extracted features.

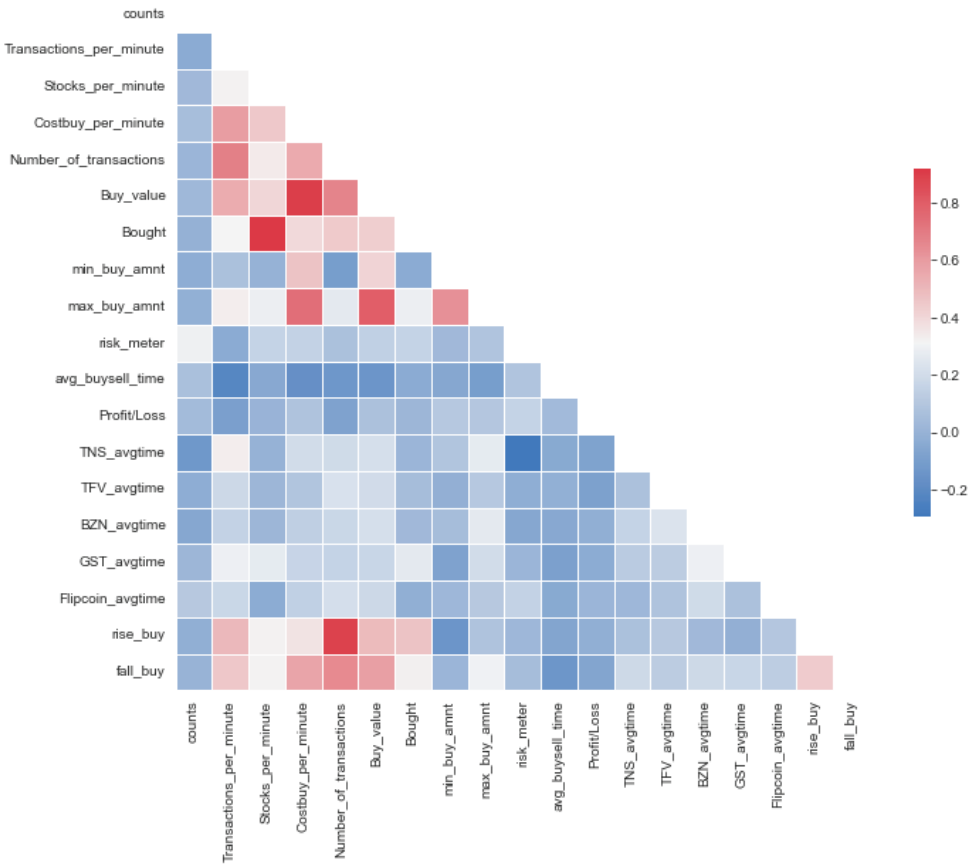

Figure 4.1: Correlation Heatmap

4.2 Clustering the Participants

This sections explains how different clustering algorithm performed for our dataset and visual evaluation of cluster results with the features from data.

4.2.1 DBSCAN Clustering

We started with taking features shown in figure 3.10 as input for our model. The features we used here are the summation of the values of features for all four rounds. This model's behavior is regulated by mainly two parameters `eps` and `min_samples` where 'eps' defines the neighbourhood of the clusters; i.e., two points are declared as neighbours if the distance between them are less than or equal to `eps` value. The later parameter specifies the minimum number of neighbour a point should have in order to consider it in a cluster. We used `NearestNeighbors` class from sklearn package which

calculates the distance between the data point and the specified minimum neighbours to get the optimal value of `eps` [29]. The optimal value should be the elbow of the curve figure 4.2 but in this case, we tried with different values of 'eps' from the range of 2 to 5 from graph to get the best clusters and came up with the value of 4.9. For parameter `min_samples`, we selected the values based on the formula [31]

$$min\_sample = 2 * TotalNumberofFeatures \tag{4.1}$$

where 'TotalNumberofFeatures' is 17 for our dataset thus the value we get is 34.

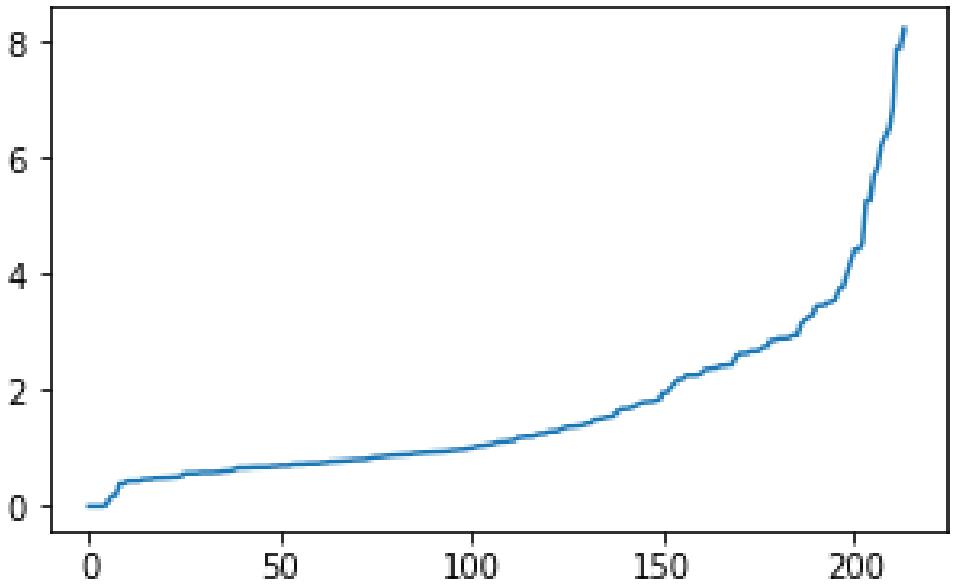

Figure 4.2: Nearest Neighbours Distances (eps)

To visualize the clusters made from DBSCAN we used Principal Component Analysis (PCA) [34] which helped us to visualize the our multi-dimensional data with just 2 principal components that mostly explained the variance in our dataset. Figure 4.3 shows the model formed one cluster in 'Blue' and detected the outliers in 'Red'. We compared the labels with the features that are most likely to help label the clusters. As shown in figure 4.4, each label has been compared sideways with the selected features such as 'transactions_per_minute', 'stocks_per_minute', 'costbuy_per_minute', 'number_of_transactions', 'rise_buy', 'risk_meter'. The X-axis represents the value of the feature while the Y-axis represents the number of users in that label. This helped

us to determine the abnormal behavior of cluster −1 compared to cluster 0 as in Cluster −1 people are aggressive buyers which can be confirmed by transactions, costbuy and stocks per minute from top two graphs.

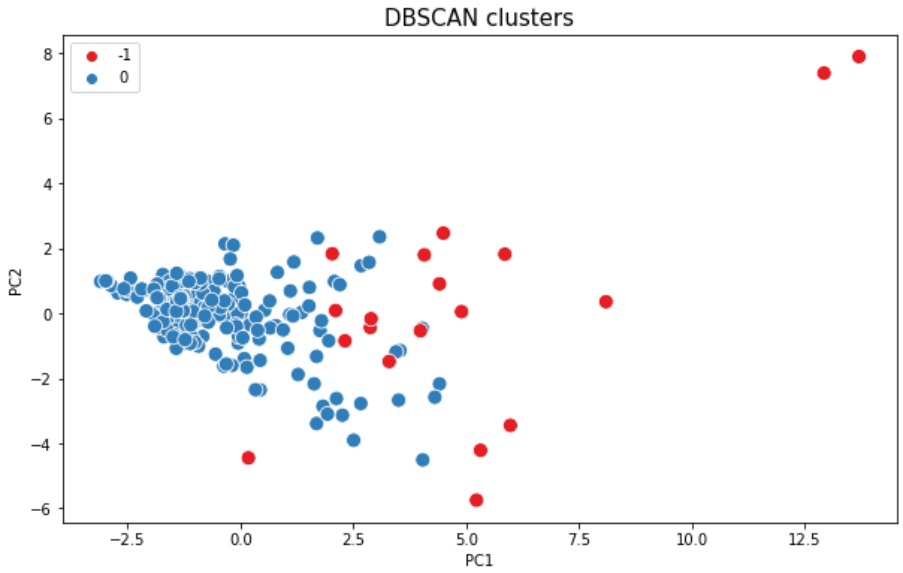

Figure 4.3: Clusters visualized using PCA

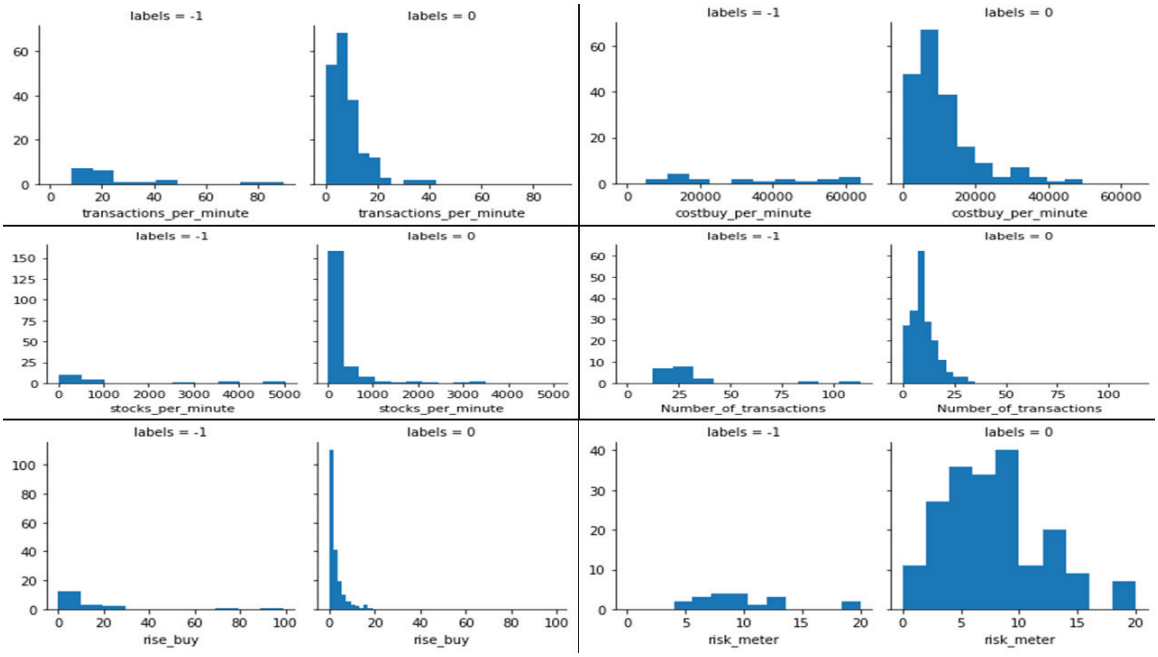

Figure 4.4: Comparison of DBSCAN clusters with features

4.2.2 KMeans Clustering

Kmeans algorithm requires to mention the K number of clusters that we want to create from the dataset [25]. We ran the algorithm on our dataset to get the clusters. As it is a centroid based algorithm, it starts with randomly initializing K centroids and then calculating the 'Euclidean' distance [5] between each point and both the centroids. This way it assigns the points to the closest centroid and thus creating clusters. Figure 4.5 shows the clusters created from KMeans. It performed almost similar to DBSCAN but has clearly separated the points into two clusters. Though there is still ambiguity in labeling the 'High Risk' or 'Low Risk' behavior to clusters in figure 4.6, the users in cluster 0 took higher risk as there are points where users had too many transactions with higher values of purchasing amount.

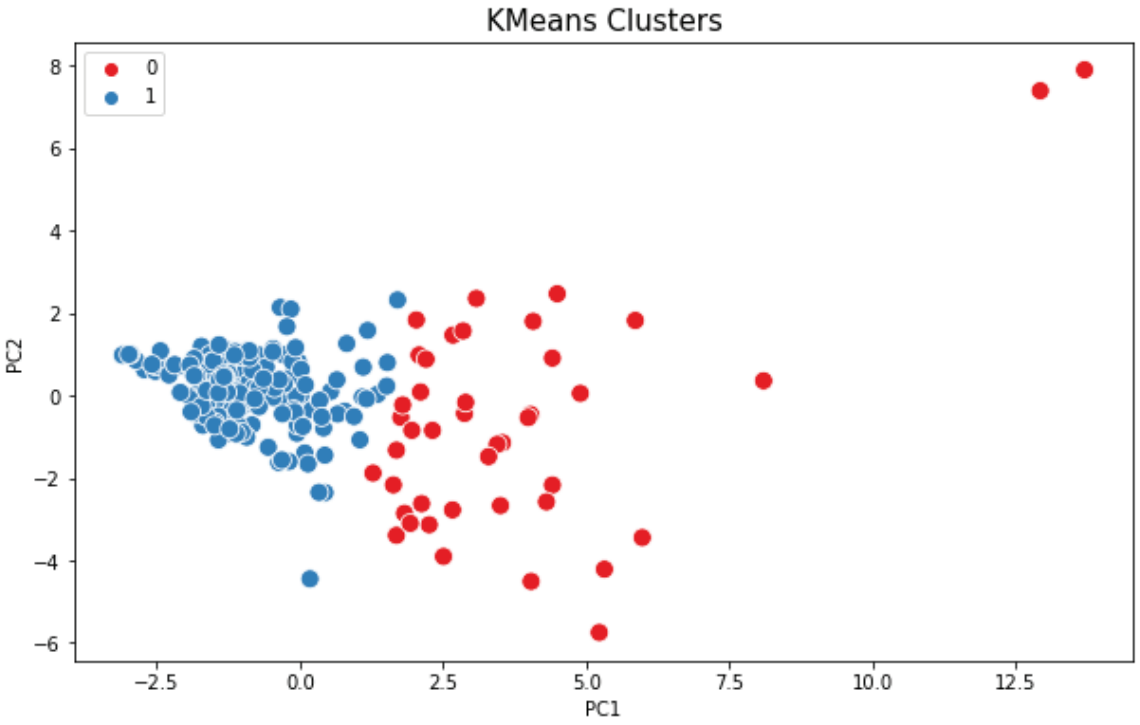

Figure 4.5: Clusters visualized using PCA

4.2.3 TimeSeriesKMeans

Though previous models do cluster the users into two categories, it was not clearly distinguishable why we can consider those clusters as 'High Risk' or 'Low Risk' takers

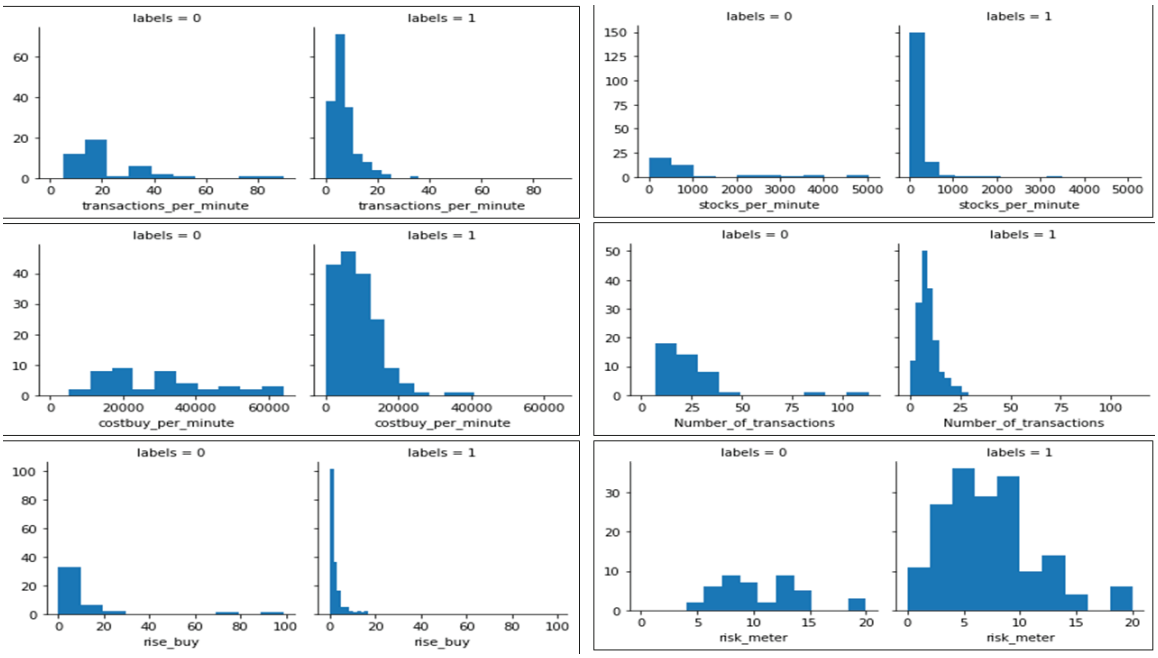

Figure 4.6: Comparison of Kmeans clusters with features

with the feature values gathered at the end of simulation of each users. Treating data as time series helped us to see a bigger picture when visualized as how people behaved on simulation and their actions through out their task. As the extracted data is in four different parts based on the paused rounds, it can be considered as time series. We used `TimeSeriesKMeans` method provided in the Python `'tslearn'` package [37]. This model performs k-means clustering but with improvements that are tailored to time series data. We employed Dynamic Time Warping [8, 33] distance metric, which calculates distance even between two dissimilar lengths of data. DTW is mostly used when the there is time shifts between two time series data but we did experiment with both 'Euclidean' distance and DTW and found better results with the later one. It compares the value of one series at time T with the value of another series at nearest points at time T. Figure 4.7 shows the resultant clusters formed. Cluster 0 and Cluster 1 are expressed in 'Blue' and 'Orange' color respectively. The numbers on the X-axis defines the rounds of the simulation which we considered it as time series. Cluster 1 are the aggressive buyers as it is clear from the graph that their transactions, per minute buy value and number of stocks bought per minute are higher as compared to Cluster 0. On the right side of the figure, the graph shows the 'risk_meter' feature which describes that Cluster 1 were buying high volatility assets

throughout each round as compared to Cluster 0 where they focused on buying low volatile assets as the round goes. Bottom graph depicts that during all four rounds, Cluster 1 were buying stocks when there is increase in price compare to previous day which shows their risky investments and their buying count are more compared to Cluster 0 in every round. This analysis shows that people in Cluster 1 are 'High Risk' takers while those in Cluster 0 are 'Low Risk' takers.

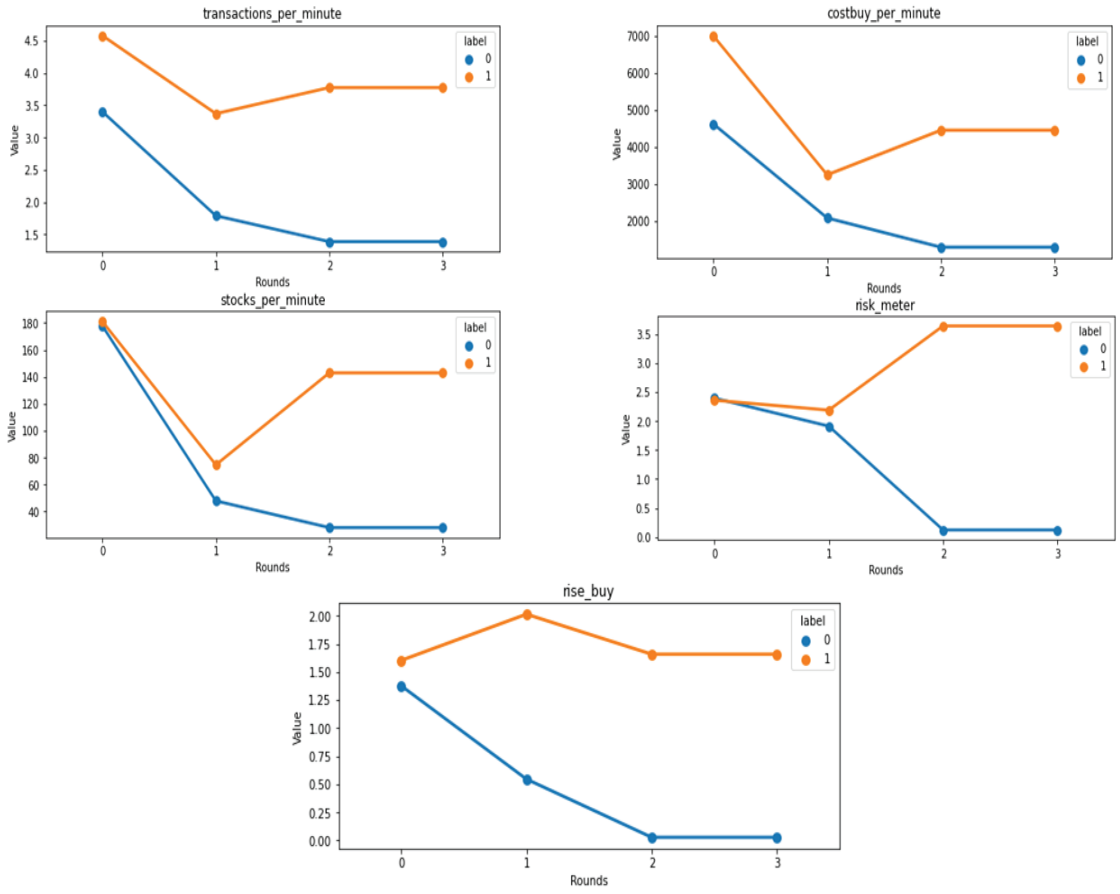

Figure 4.7: Comparison of TimeSeriesKMeans clusters with risk related features

TimeSeriesKMeans gave better insights by distinguishing participants based on their risk propensity compared to DBSCAN and KMeans.

4.3 Linear Regression for Risk and Enjoyment Prediction

The main idea behind using Linear Regression is to describe the relationship between Clusters created by TimeSeriesKMeans by means of independent variable; i.e., Features, and the dependent variable; i.e., Risk and Enjoyment survey. We used Ordinary Least Square regression method from Python's statsmodel [32] package. By minimising the sum of squares in the difference between the observed and predicted values of the dependent variable constructed as a straight line, the approach evaluates the relationship. If the difference between the actual values and the model's predicted values are modest and unbiased, the model fits the data well. We got the following results for both the model where one had risk survey as dependent variable while other had enjoyment survey as dependent variable.

$$\begin{aligned} Risk/Enjoyment = \beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + \beta_3 * X_3 + \beta_4 * X_4 + \beta_5 * X_5 + \beta_6 * X_6 \\ + \beta_7 * X_7 + \beta_8 * X_8 + \beta_9 * X_9 + \beta_{10} * X_{10} + \beta_{11} * X_{11} + \beta_{12} * X_{12} \\ + \beta_{13} * X_{13} \end{aligned}$$

For Risk Survey:

$$R\text{-}squared = 0.230 \qquad pvalue > 0.05$$

For Enjoyment Survey:

$$R\text{-}squared = 0.260 \qquad pvalue > 0.05$$

The R-squared value is the statistical measure which indicates how near the data is to the fitted regression line or in other words it shows the percentage of variance in the dependent variable explained by the independent variables when taken together. It ranges between 0 to 100 percent. Generally, higher R-square determines that the model fits the data. Usually in psychology, lower R-squared value does not mean that model is not working properly as the goal states to predict the human behavior which is way harder in real life to predict it. That being said, if the predictors are statistically significant then also we can interpret that changes in them are linked with the target variable. But in this case, we got the p-value higher than critical

value 0.05 which should be lower than it in order to consider data as significant. All of the predictors correspond to dependent variable risk perception gives p-value greater than 0.05 while in case of enjoyment survey as dependent variable except 'Transactions_per_minute' and 'Rise_Buy' where p-value corresponds to them were 0.013 and 0.005 respectively. This experiment did not gave proper results with the amount of data we gathered and the continuous nature of the labels.

4.4 Classifying the Risk Propensity and Enjoyment

This task helps us to accomplish the notion to classify the Risk Propensity of users based on their buying behavior on simulation. As our previous experiment with continuous target variable did not give good results, we considered last 3 questions from pre-session survey data about participants willingness to gamble lifetime income as our labels. The questions are as follows:

```
1) Suppose that you are the only income earner in your family, and
you have a good job guaranteed to provide you an average income
every year for life.  You are given the opportunity to take a new
and equally good job, with a 50-50 chance that it will double your
income and a 50-50 chance it will cut your income by a third.  Would
you take the new job?

2) If yes, suppose the chances were 50-50 that it would double your
income and 50-50 that it would cut it in half.  Would you still take
the new job?

3) If no to the first question, suppose the chances were 50-50 that
it would double your income and 50-50 that it would cut it by 20
percent.  Would you then take the new job?
```

If the participants said 'Yes' to the first question and 'No' or 'Yes' to the second questions, then they were categorized as 'High Risk' takers while if they said 'No' to the first question and 'Yes' or 'No' the third question then they were considered as 'Low Risk' takers. We used the same dataset for this task with extra features extracted from tables 3.7 and 3.9 as 'counts' — number of times user played

the simulation, 'avg_buysell_time' — average time taken by user between two consecutive buy, and sell transaction and 'Remaining Stock' — number of stocks left at the end of the simulation because introducing this features increased the model accuracy by few percent. We used total five models for classification. First one is `DecisionTreeClassifier` with parameter 'criterion' which determines the optimum split of features. We used 'entropy' as criterion which is defined by the formula where p_j is the probability of label j.

$$Entropy = -\sum_j p_j \cdot \log_2 p_j$$

The second model we used is '`RandomForestClassifier`' which comprises of multiple decision trees with parameter 'n_estimators' which defines the number of trees to be used in the model. We set the value of 100 for 'n_estimators' means it will construct 100 decision trees. Third one is '`LogisticRegression`' classifier which mostly is used to predict the probability of the target variable or labels. The fourth one is '`GaussianNB`' classifier which uses Bayes Theorem that is based on conditional probability and the last one is '`SVM`' classifier. K-Fold cross-validation method were used to train the models and it is mostly used for performance estimation of model and if the dataset is small or limited. This method splits the dataset into K non-overlapping parts and each part is kept as a hold out for testing purpose and rest folds were used for training in for K iterations. The average performance of K iterations are considered as final output of the model. The optimal value for K used is 10 which was mentioned by Borra *et al.* [10]. We compared all four models with different evaluation metrics to assess model's performance [15, 41].

	Decision Tree	Random Forest	Logistic Regression	Naive Bayes	Support Vector Machine
Accuracy	0.537013	0.541991	0.640476	0.495022	0.645022
Precision	0.652257	0.624297	0.643290	0.680310	0.645022
Recall	0.615385	0.732967	0.992857	0.397802	1.000000
F1 Score	0.627260	0.671865	0.780560	0.497151	0.784052

Figure 4.8: Evaluation of Classification models with Risk survey labels

One of the metrics that was used is 'Accuracy' which defines as fraction of number

	Decision Tree	Random Forest	Logistic Regression	Naive Bayes	Support Vector Machine
Accuracy	0.533618	0.654131	0.676923	0.616239	0.68433
Precision	0.662828	0.706342	0.681766	0.682556	0.68433
Recall	0.637427	0.846784	0.988889	0.823977	1.00000
F1 Score	0.643241	0.769273	0.807036	0.745778	0.81249

Figure 4.9: Evaluation of Classification models with Enjoyment survey labels

of correct predictions to total number of predictions. It is also defined as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP is True Positive, TN is True Negative, FP is False Positive and FN is False Negative. Precision is the fraction of true positives to all the positives predicted. It helps to determine the proportion of anticipated positives are actually 'Positive'.

$$Precision = \frac{TP}{TP + FP}$$

Recall states the proportion of true positives to all the positive in the dataset. It shows the proportion of actual positives are correctly classified.

$$Recall = \frac{TP}{TP + FN}$$

F1-score is the combination of both Precision and Recall. It is the weighted average of both.

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

Figure 4.8 shows the comparison of four metrics on all of our models. Accuracy, Precision and F1 score of Logistic Regression and Support Vector Machine were almost similar and higher than other models. All in all SVM performed better than Logistic Regression by fraction of values.

This experiment is followed by the task where we considered our labels as post-session enjoyment data. We wanted to transform the continuous value of enjoyment survey in bins in order to categorize it as 'High Risk' or 'Low Risk' for classification. We discretized the data based on the median value as cutoff. These boundaries are not subjective choice but rather an intuitively understandable for wide audience in order to interpret the values as the dataset is so small and will helpful for categorizing

it based on ‘High Risk’ and ‘Low Risk’. As shown in figure 4.9, the comparison helps us to determine that overall performance of Logistic Regression Classifier and SVM classifier are better as compared to others.

Chapter 5

Descriptive Analysis and Discussion

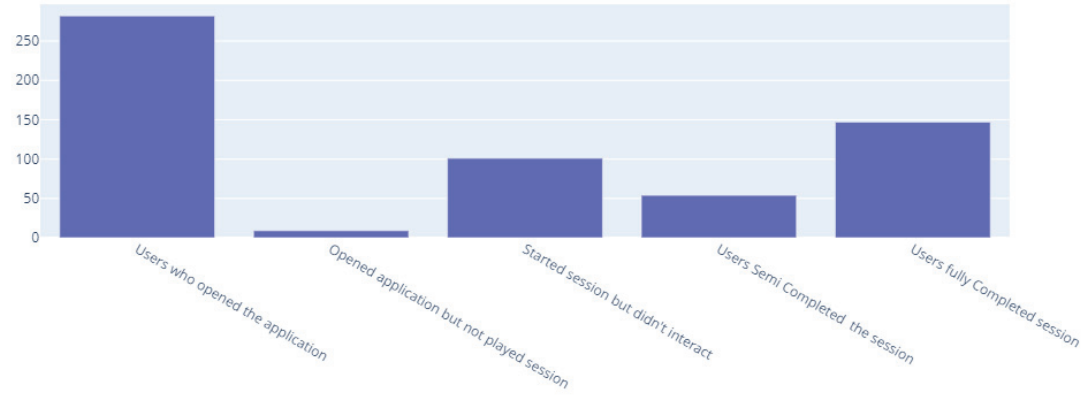

Figure 5.1: User Analysis

We ran an analysis on how users interacted with the simulation and found out that 282 people opened the simulation by clicking the link out of which 9 participants opened the application but did not played the session, 101 participants started the session but did not interact with the simulation, 54 users left the session in the middle and 147 users completed the whole session 5.1.

Figure 5.2 gives an insight about the people how with their risk behavior made profit or loss. It is clear that most of the people around 40% of the users that we've analyzed were 'Low Risk' takers and made profit, followed by 27% of the users who were low risk takers and made loss whereas only 16% of the users that were detected as 'High Risk' takers but made loss on simulation. Majority of the participants were low risk takers on the simulation.

According to our collected dataset, as shown in figure 5.3 15 participants were prompted the bonus round which included to trade with new cryptocurrency and their enjoyment factor after bonus round was higher than the mean of the all the player

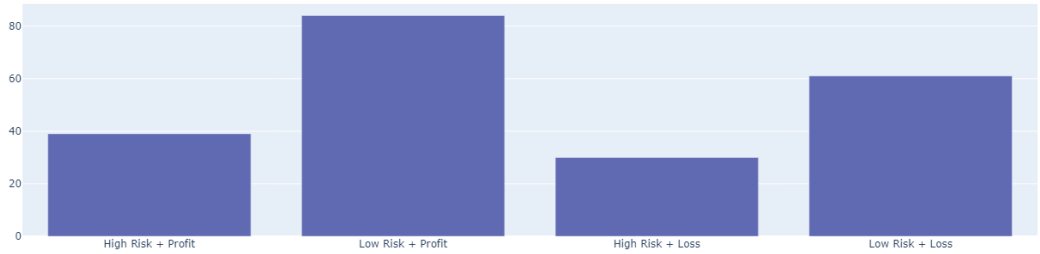

Figure 5.2: Type of Risk takers and Profit/Loss

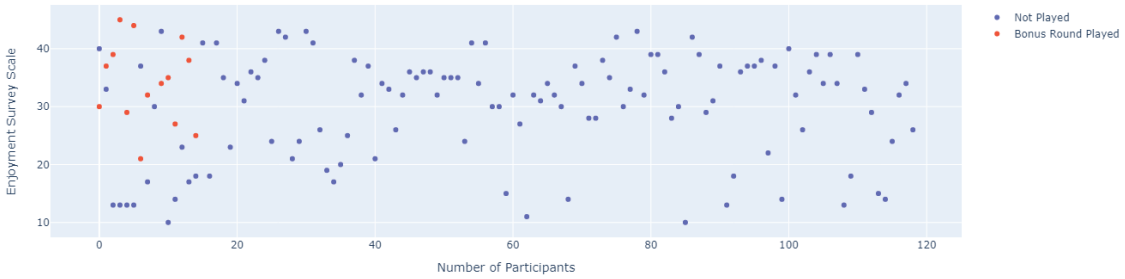

Figure 5.3: Enjoyment of Users played bonus round vs Not played

who filled the enjoyment survey. This gives a light to our research question that game prompts are likely to influence cognitive absorption factor such as 'Enjoyment'.

After getting the promising results from clustering technique `TimeseriesKMeans`, we analyzed the features that created the clusters with psychometric data: pre-session survey 'Risk Perception' using correlation test from `scipy`'s stats module [40]. Although there are chances that their perception about risk and their action might not always align with each other, our study tells that even with such small dataset, it is still somewhat significant in table 5.1 between the risk perception and three features. The test statistics explains us that the how much the sample data deviates from the Null hypothesis which states that there is no relation between variables while the p-value tells us that if the null hypothesis were true, we'd expect to observe data as extreme as our sample roughly 1% of the time due to chance.

In figure 5.4, the features mentioned the average time taken by users between

Features	T-statistic	P-value	Correlation
risk_meter	3.279	0.001	0.219
counts	6.165	< 0.001	0.389
flipcoin_avgtime	1.984	0.04	0.135

Table 5.1: Significance test of Risk Survey with Features

buy and sell of that particular asset. In case of disengagement of users in buying assets through application, the high risk people; i.e., 'Cluster 1', were less engaging as they were taking long time between consecutive buy and sell compared to low risk people; i.e., 'Cluster 0'. On the other hand, 'Cluster 0' people were not at all engaging with application during round 3 and round 4. Possible reason might be they were waiting for stock prices to go high to sell it. This concludes that high risk takers are more engaging in such applications in comparison with low risk takers who invest and waited for their assets to give them profit.

These findings led interesting theoretical contributions. They contend that user risk-taking behaviour on stock trading simulation was partially influenced by social characteristics including risk appetite and enjoyment, which may also apply to the meme stock phenomenon. User enjoyment is known to be key factor in predicting hedonic information system use [42, 35] and risk appetite is known to influence decision making [39]. We are led to assert that these findings generalize to the case of investing applications. In addition to this, we demonstrate the potential of a novel strategy for behavioral modelling and theory development. We showed how mixed methods approaches including questionnaires might support machine learning approaches to behavioural research issues in response to the request for computationally intensive models of theory generation [9, 23]

The overall contribution involves distinguishing meaningful risk-taking behavior and that the survey can play a role in providing additional insight to our model. While the differences in the high risk behaviors could differentiate the clusters, the congruence between the survey measures and the clusters suggests that the unsupervised learning not only captured behavior, but were also indirectly indicative of the participants' psychology. The evaluation measures used to observe this was measures of psychological traits, our results suggest that clusters likely reflect behavior which is

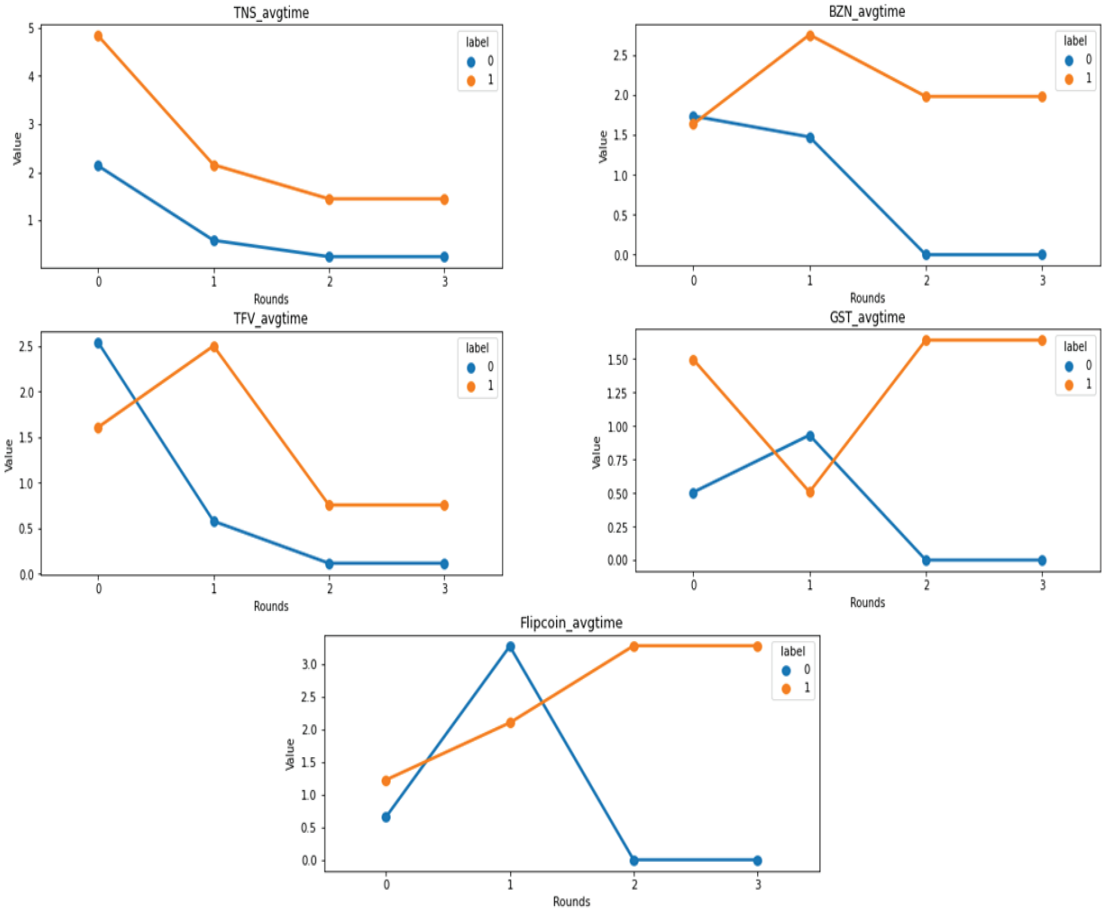

Figure 5.4: Comparison of TimeSeriesKMeans clusters with user engagement features (cluster 0 — low-risk, cluster 1 — high-risk participants)

at least in-part influenced by an individual's risk appetite. This was supported by the classification task where the features were able to predict the psychological trait of the users with decent accuracy. This comparison has applications beyond this study, as it demonstrates how unsupervised learning discoveries concerning human behavior could be evaluated using psychometric or other survey techniques.

5.1 Research Questions Revisited

The analysis of the collected data and the experiments led us to the answers of our research questions:

Research Question 1: *Are there effective methods for clustering risky investment app trading behaviours based on user actions?*

The unsupervised learning method gave us insights that the user's action on simulation or their transaction behavior are distinguishable to 'High Risk' and 'Low Risk' takers. Considering the dataset as timeseries throughout the simulation rounds provided a clear picture of their playing behavior. The assertion was supported by the 'TimeSeriesKMeans' algorithm which helped in clustering the users based on their behavior.

Research question 2: *Are psychometric risk surveys useful as labels for classifying trading behavior?*

The results from the supervised learning experiment showed that risk and enjoyment surveys can be used as labels to predict the trading behavior. Even with such small dataset, the algorithms were able to acheive more than 60% accuracy. The increase in quantity as well quality data will lead to higher accuracy and thus will be an useful resource for the classifying behavior.

Research question 3: *Do bonus prompts influence users enjoyment on investing application?*

In terms of bonus rounds, these are used as persuasive strategies to influence users to utilize any application more. The main motivation introducing the bonus prompt was to find out whether such strategy works for investing application and we found out that it holds true for trading simulation too. These features motivates users' to continuously use application as it influence them as a fun factor or it susceptible towards earning more.

5.2 Limitations

The main limitation in our research was data sample size. Our findings are limited by challenges with recruitment and data quality. Though 282 participants consented to participate out of which some of the participants were recruited from Prolific, which is a platform of convenience, where users can be recruited for study by means of payment. Moreover, most of our collected data was incomplete as only 147 participants completed the full task and engaged in the simulation through out the session and also answered both pre-session and post-session survey. The amount of data is not

sufficient for comparing clusters and the psychometric questionnaires though we get little insight with it. The sample size prohibits the use of more complex techniques such as neural networks or deep learning. Moreover, the limited data of bonus round played users were not significant enough to find the relation about the influence of bonus prompts in their trading behavior. The current data does not explain much about the behavioral pattern binds with bonus rounds. In addition to this, many participants skip the survey component of the task which suggest that data quality may not be robust. The results we obtained can be considered complete but an uncomprehensive analysis.

Apart from this, there can be theoretical challenges in deriving conclusions from simulation. Despite the fact that the task was created to replicate the Robinhood investing application, participants might not have the same level of motivation as people who really use the real-world trading application. Future research can get around this problem by getting feedback from users of the Robinhood app in addition to inferring conclusions from the behavioural task.

Chapter 6

Conclusion and Future Work

In this study, we investigated the potential drivers of risky trading behaviour among users of mobile trading platforms to facilitate the risk-taking decisions and enjoyment of users which can be used to interpret real life behavior. Our first step was to create a simulation of a mobile trading platform where we provided the users with hypothetical amount in their wallet for investing. The actions users took on the simulation helped us to create clusters of users based on their trading behavior. The efficient way to approach this task was to consider data as time series throughout the simulation and the machine learning model which accomplishes the clustering of time series data was 'TimeSeriesKMeans'. Unsupervised approach on people's actions can be used to differentiate their behavior on live platform by making clusters.

We used supervised learning approach on user's risk perception and enjoyment as dependent variable and their behavior on simulation as independent variables to state the relationship between both whether it can classify or not. The results were quite acceptable and were able to classify their behavior by mapping psychometric measures. SVM classifier worked better than other classifiers for the small dataset that we have collected.

Due to the limitation in data, most of the relationship we found on clusters and psychometric surveys were not significant. Future research will include collecting more data of users to understand the bigger picture of population behavior. Structuring the surveys which consist of several latent features such as enjoyment, engagement, and attractiveness of design features which can be categorize for using in Machine Learning models and hypothesis testing will lead to more concrete results in future. Researchers can make use of this knowledge to improve machine learning methods for behavioural modelling and to broaden their investigation into several domain such as the driving forces behind why people utilise social media to disrupt financial markets, factors and features that lead users to spend a lot of time on specific gaming,

applications which led people to increase their screen time on mobile and many more.

There are also practical implications suggesting for later work. Design elements used in investing apps like Robinhood encourage users to make riskier choices and reinforce hedonic affordances. For instance, Robinhood has a referral network and a rewards system that resemble those found in social media and mobile games. Application designers or other stakeholders can restrict or promote features that either identify risk-taking, promote enjoyment, or prohibit these things depending on their goals. These results could be improved in the future by testing particular characteristics that either stimulate or restrict the risk or hedonic elements. It is true that we won't get actual behavior of the users with these simulations but experiments can be refined to get near results of users' behavior. In future, the design changes can be made on 'Nottingham' simulation such as introducing the sound feature when user clicks any button, appreciation notes when they made profit based on their decision, brief description about the news of the stocks or asset they would be on, and increasing the length of the simulation to provide them enough time to read the news and make transactions. The more we research the intricate elements that influence dangerous trading, the more we'll be able to develop a strategy that won't result in terrible financial repercussions for trading platforms.

Bibliography

[1] Electronic gaming machine playstyle detection and rapid playstyle classification using multivariate convolutional LSTM neural network architecture, 2021.

[2] Bitcoin (btc) price, charts, and news: Coinbase: bitcoin price, btc price, bitcoin coinbase, 2022.

[3] Ritu Agarwal and Elena Karahanna. Time flies when you're having fun: Cognitive absorption and beliefs about information technology usage. *MIS Quarterly*, 24(4):665–694, 2000. Publisher: JSTOR.

[4] Michael Auer and Mark D Griffiths. Predicting limit-setting behavior of gamblers using machine learning algorithms: A real-world study of norwegian gamblers using account data. *Int. J. Ment. Health Addict.*, 20(2):771–788, April 2022.

[5] Paul Barrett. Euclidean distance: raw, normalized, and double-scaled coefficients, September 2005.

[6] Robert B Barsky, F Thomas Juster, Miles S Kimball, and Matthew D Shapiro. Preference parameters and behavioral heterogeneity: An experimental approach in the health and retirement study. *The Quarterly Journal of Economics*, 112(2):537–579, 1997.

[7] Michael Bayer. Sqlalchemy. In Amy Brown and Greg Wilson, editors, *The Architecture of Open Source Applications Volume II: Structure, Scale, and a Few More Fearless Hacks*. aosabook.org, 2012.

[8] Donald J. Bemdt and James Clifford. Using dynamic time warping to find patterns in time series, Apr 1996.

[9] Nicholas Berente, Stefan Seidel, and Hani Safadi. Research commentary—data-driven computationally intensive theory development. *Inf. Syst. Res.*, 30(1):50–64, March 2019.

[10] Simone Borra and Agostino Di Ciaccio. Measuring the prediction error. a comparison of cross-validation, bootstrap and covariance penalty methods. *Computational Statistics Data Analysis*, 54(12):2976–2989, 2010.

[11] Abram Brown. Reddit Traders Have Lost Millions Over GameStop. But Many Are Refusing To Quit. Section: Social Media.

[12] Sandy C Chen, Stowe Shoemaker, and Dina Marie V Zemke. Segmenting slot machine players: a factor-cluster analysis. *Int. J. Contemp. Hosp. Manag.*, 25(1):23–48, February 2013.

[13] Michele Costola, Matteo Iacopini, and Carlo R M A Santagiustina. On the "mementum" of meme stocks. *Econ. Lett.*, 207(110021):110021, October 2021.

[14] Alejandro García-Jurado, Mercedes Torres-Jiménez, Antonio L. Leal-Rodríguez, and Pilar Castro-González. Does gamification engage users in online shopping? *Electronic Commerce Research and Applications*, 48:101076, July 2021.

[15] Margherita Grandini, Enrico Bagli, and Giorgio Visani. Metrics for multi-class classification: An overview. 2020.

[16] Miguel Grinberg. *Flask Web Development: Developing Web Applications with Python*. O'Reilly Media, Inc., 2018.

[17] Dan ; Barton Kevin Harrigan, Kevin A. ; Brown. Classification of slot machines in ontario: Providing relevant information to players. 2017.

[18] Cindy D Kam and Elizabeth N Simas. Risk orientations and policy frames. *The Journal of Politics*, 72(2):381–396, 2010.

[19] Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Samuel Harford. Multivariate LSTM-FCNs for time series classification. *Neural Networks*, 116:237–245, August 2019.

[20] Pierre-Majorique Léger, Fred D Davis, Timothy Paul Cronan, and Julien Perret. Neurophysiological correlates of cognitive absorption in an enactive training context. *Computers in Human Behavior*, 34:273–283, 2014.

[21] Yun Li. GameStop mania explained: How the Reddit retail trading crowd ran over Wall Street pros, January 2021. Section: Markets.

[22] Haibin Liu and Vlado Kešelj. Combined mining of Web server logs and web contents for classifying user navigation patterns and predicting users' future requests. *Data & Knowledge Engineering*, 61(2):304–330, May 2007.

[23] Shaila M Miranda, University of Oklahoma, Inchan Kim, Jama D Summers, University of Oklahoma, and University of Tennessee. Jamming with social media: How cognitive structuring of organizing vision facets affects IT innovation diffusion. *MIS Q*, 39(3):591–614, March 2015.

[24] Chinenye Ndulue, Oladapo Oyebode, Ravishankar Subramani Iyer, Anirudh Ganesh, Syed Ishtiaque Ahmed, and Rita Orji. Personality-targeted persuasive gamified systems: exploring the impact of application domain on the effectiveness of behaviour change strategies. *User Modeling and User-Adapted Interaction*, 32(1-2):165–214, March 2022.

[25] Dr. S. S. Prabhune Niraj N Kasliwal, Shrikant Lade. Introduction of clustering by using k-means methodology. *International Journal of Engineering Research Technology (IJERT)*, 1, 2012. ISSN: 2278-0181.

[26] Fernando Peres, Enrico Fallacara, Luca Manzoni, Mauro Castelli, Aleš Popovič, Miguel Rodrigues, and Pedro Estevens. Time series clustering of online gambling activities for addicted users' detection. *Appl. Sci. (Basel)*, 11(5):2397, March 2021.

[27] Kahlil S Philander. Identifying high-risk online gamblers: a comparison of data mining procedures. *Int. Gambl. Stud.*, 14(1):53–63, January 2014.

[28] Thomas Plieger, Thomas Grünhage, Éilish Duke, and Martin Reuter. Predicting stock market performance. *J. Individ. Differ.*, 42(2):64–73, April 2021.

[29] Nadia Rahmah and Imas Sukaesih Sitanggang. Determination of optimal epsilon (eps) value on DBSCAN algorithm to clustering data on peatland hotspots in sumatra. *IOP Conf. Ser. Earth Environ. Sci.*, 31:012012, January 2016.

[30] H Sakoe and S Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Acoust.*, 26(1):43–49, February 1978.

[31] Kriegel Hans-Peter Xu Xiaowei Sander Jörg, Ester Martin. Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery*, 2:169–194, January 1998.

[32] Skipper Seabold and Josef Perktold. Statsmodels: Econometric and statistical modeling with python. *Proceedings of the 9th Python in Science Conference*, 2010, 01 2010.

[33] Pavel Senin. Dynamic time warping algorithm review. December 2008.

[34] Jonathon Shlens. A tutorial on principal component analysis. April 2014.

[35] Deb Sledgianowski and Songpol Kulviwat. Using social network sites: The effects of playfulness, critical mass and trust in a hedonic context. *Journal of Computer Information Systems*, 49:74–83, 06 2009.

[36] Emily Stewart. GameStop. Dogecoin. Now AMC. Do meme traders need to be protected from themselves?, June 2021.

[37] Romain Tavenard, Johann Faouzi, Gilles Vandewiele, Felix Divo, Guillaume Androz, Chester Holtz, Marie Payne, Roman Yurchak, Marc Rußwurm, Kushal Kolar, and Eli Woods. Tslearn, a machine learning toolkit for time series data. *Journal of Machine Learning Research*, 21(118):1–6, 2020.

[38] Hsien-Tung Tsai and Richard P. Bagozzi. Contribution Behavior in Virtual Communities: Cognitive, Emotional, and Social Influences. *MIS Quarterly*, 38(1):143–164, 2014. Publisher: Management Information Systems Research Center, University of Minnesota.

[39] Anthony Vance, Bonnie Anderson, C. Brock Kirwan, and David Eargle. Using Measures of Risk Perception to Predict Information Security Behavior: Insights from Electroencephalography (EEG). *Journal of the Association for Information Systems*, 15(10), October 2014.

[40] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[41] Zeljko . Vujović. Classification model evaluation metrics. *International Journal of Advanced Computer Science and Applications*, 12(6), 2021.

[42] Robin L Wakefield and Dwayne Whitten. Mobile computing: a user study on hedonic/utilitarian mobile device usage. *Eur. J. Inf. Syst.*, 15(3):292–300, June 2006.