ABSTRACT

ACKNOWLEDGEMENTS

Contents

Chapter 1

Introduction

1.1 Background and Motivation

Learning curves may reflect different underlying learning processes. For example, in insight learning, it may follow slightly fluctuated level and sudden dramatic increase at some points. Because the learning is sometimes the result of experience through personal interactions with the environment. And following the occurrence of insight, abrupt realisation of how to solve the problem can be repeated in future similar situations. This means that the associated experiences with insights can be parameterised, and it can be linked to future behaviours.

Therefore, there are several possible functions which might fit learning curves and represent different learning processes; step function, powerlaw, piecewise powerlaw and etc. Through possible functions and learning curves, people can be grouped into several learning patterns. Moreover, by inspecting people in those groups, 'how they practice' to get the learning curve and 'which features' having an affect upon individuals can be found. Then suggestions for better learning for individuals may be categorised and proposed.

For this, online games are a reasonable instrument. As it involves "rapid perception, decision making, and motor responding", as well as it gives rich details of practice history, in order to find individuals' learning curve and investigate their features (Stafford and Dewar, 2013)

Text...

1.2 Problem Definition

There are three main issues in traditional reinforcement learning curve. Because it has smooth power law with diminishing gain derived from average value, all possible individual learning curve are squashed into one learning curve. Thus, it cannot tell which learning curve will belong to a task, behaviour, and learning process. Furthermore, even though identifying bad or good on learnings are important, but it cannot provide sufficient information about those.

Below is the list of problems on current reinforcement learning curve.

- Not being able to categorise learning process.
- Averaging possible individual learning curves
- Not enough information for identification of learning success

1.3 Aims and Objectives

These are the concepts I think you should cover in your introduction:

Skill acquisition & expertise - factors which are known to influence skill acquisition - practice amount, practice spacing, others

Using games to study skill acquisition

Learning curves - different possible underlying functions - problems with averaging over

The ultimate aim, of course, is to identify individuals who learn faster or for longer, and try and relate this to how they practice The aim of this paper is to fit different functions to the Axon game data. In the data, there are 854,064 individuals data about when, where they did the game and other informations, as well as scores of game according to their attempts. And so identify the underlying learning process. Furthermore, it is to inspect features which may have influences on the learning curves.

- Understand several possible functions; step function, powerlaw, piecewise powerlaw and etc.
- Understand different underlying learning processes; insight learning, associative learning, multi-component/process learning and etc.

- Fit functions to individual Axon game data and compare those to identify.
- Test theories of what makes learning most effective, exploiting unsupervised learning, establish which parameter is important for getting learning curve from the data.
- Design more effective learning practices.

1.4 Project Management

Text...

Chapter 2

Data Curation

A large number of data were acquired through the online game named Axon which is developed by Preload for Welcome trust (?, ?). Totally 1,201,516 machine identities (or players) played the game over 4 million times. The raw data set is fundamentally comprised by the score, date, time of plays and so on in accordance with machine identities which may represent each individual. Undoubtedly, it can be possible to extract information from the data set on how people practiced to get a higher score. For example, it shows how much time they played for each score. However, the raw data set of on-line game seems quite noisy to discern those factors which affect player's learning. It contained many 'undefined' or 'unrepresentable values', and some values are not valid, such as starting play time is later than the time play finished. For this reason, the information on what the analysis needs cannot be identified directly from it.

Thus, the data source needs refining processes until the gemstone of data set could gleam with the evidence of valuable information. In other word, the big data set has to be curated. Stonebraker et al. stated that "data curation is the act of discovering a data source(s) of interest, cleaning and transforming the new data, semantically integrating it with other local data sources, and deduplicating the resulting composite" (?, ?).

Below are the tasks in which steps of data curation are largely expected,

1. Cleaning raw data set,
2. Incrementally accommodating new data entities,
3. Normalising scores grouped by the same players,

4. Clustering players, showing the similar patterns.

Especially, data curation tasks followed methods of two previous studies (?, ?, ?). Accordingly, the curation tasks of cleaning, grouping by machine IDs and accommodating will be explained at the following section. After then, another tasks for data deformation and clustering will be examined.

2.1 Data Cleaning

What has to be noted before data cleaning is that it has to be implemented under the grouped data by the same machine IDs. Totally 4,038,802 plays were sorted by players having the same machine IDs, and then 1,201,516 players were detected. The criteria according to the previous two studies (?, ?, ?) are as follow,

1. players who did not play at least 15 games,
2. players who attempted more than 300 times,
3. data which lack valid longitude or timing information for each attempt,
4. players with discontinuous game play attempts.

The reason of discarding those who played less than 15 attempts is because those cannot show a causal relationship between attempts and scores. In this step, xxx individuals were deleted. And the reason of second step is that plays more than 300 attempts did not show many differences against after 300 attempts. ??? out of total number of players attempted the game more than 300 times, and also they were removed. Also discarded players having unrecognisable time information were ???. Finally, game play data for each individual recorded discontinuously. In other word, there are non-recorded game play data between the first attempt and the last attempt for each players. The reason why this problem happened is unknown though, it is obvious that individuals with this issue have to be filtered appropriately. Though cleaning those is simply mentioned in this session, the specific method how to manage players with discontinuous data will be discussed in chapter ???.

As a result, raw data with 4, 038,802 plays with xxx players were cleaned, and then reduced to ??? plays and ??? individuals decreased by ???.

2.2 Data accommodating

The next step of data curation is to incrementally accommodate hidden but obvious features from the previous cleaned data set. It is originally constituted by basic 8 categories. After data accommodating task, 7 additional hidden variable would be derived.

2.2.1 Original variables in the data source

When people play the on-line game, 7 basic information as follow is recorded,

- machine identities,
- scores and attempt numbers,
- date, hour and minute,
- latitude and longitude.

When players access the on-line game, tracking code written by Preloaded record machine identities. Machine IDs may be considered as individuals who actually play the game. Preloaded inserted tracking code to the on-line game, which records a machine identity at each time the game was loaded, as well as the result of the game at the end of plays, and play date and time. Attempts and scores were recorded in the order of the time.

The information of locations where the machine ID accessed the game is also collected approximately at the maximum of city-block level.

2.2.2 Hidden variables calculated from the original variables

From the basic variables which are recorded when people play the game, other features that may imply useful information of how players are influenced during learning processes to get higher marks can be derived as bellow,

- time difference,
- local time,
- gap types (no, long, short and sleep gap),

- a number of total plays.

The basic time information; date, hour and minute is combined together. And then, time differences between successive attempts are calculated by subtracting current game start time from its previous attempt time. Therefore, the first attempts do not have value of time differences.

Next, local times for each play were calculated in seconds, using the formula, "$local\ time = UTCtime + (longitude24 \div 360), modulo24$" (?, ?). Local time would facilitate comparison of players in different time zone at the same level as life style time. There was data that local time could not be calculated because of missing tack of latitude and longitude, and those were filtered at the data cleaning task.

On the basis of time difference, Stafford et al. (2016) classified individuals into 4 groups, according to "the nature of the timing of their first 15 attempts at the game". First group is 'no gap' for those who had less than 15 minute break between each play. Second group is 'wake' in which players are assumed having a rest in there waking hours if they have "a single gap between 7 and 12 hours". Working hours is time between 5 am and 12 pm. And those who rested for between 7 and 12 hours in working hours were categorised as 'sleep'. Finally, all other individuals are classified as "no category".

Furthermore, for more detailed analysis, 'rest' which is defined in 'Gap types' categorisation was divided with 3 types. Long gap, short gap and sleep gap were determined when players had a rest 'between 7 hours and 11 hours', 'below 15 minutes', and 'not in working hours' respectively.

As the term of hidden variables, there are possibilities that more factors can be mined from the data set and be incrementally accommodated to it if appropriate methods are implemented.

2.3 Data Deformation

CSV format has been used as a basic data file format for previous study.

CSV is an abbreviation of Comma-Separated Values. A data record of CSV format file is a line formed by one or more elements, and commas separate those element

values to be distinguishable (plain text format) (?, ?). According to Idris (2014), it is straightforward for those format to load and store data-set from the data storage onto working place; ease to generate, read and edit manually, versatility in most programming languages, but not efficient because exploiting plain text format ”take a lot of space”. And this might result in consuming more time to handle such a big data set than using binary format.

Simply, the data set needs to be stored in memory unit of machine(s) where the analysis processes would operate. However, these are true only when data size is manageable. Because fitting all data-set of such a big data in computer memory takes many spaces. In some case loading on computer memory cannot be possible depending on computer capability and setup. Thus, CSV format is not very efficient way to manage such a big data (?, ?). To solve with those problems, two approaches of being able to reduce storing spaces and of enabling faster speed for analysing were applied; converting CSV file format into Pickle format, and splitting data set into several pieces.

2.3.1 Pickle file format

Pickle file format can provide ”a high level of data compression such as zip, bzip and gzip” and also faster than CSV (?, ?), because it is ”binary protocols for serializing and de-serializing a Python object structure” (?, ?). Furthermore, it provides easy approach to deal with converting Python object hierarchy into a binary stream and vice versa, with the notion of ’pickling’ and ’unpickling’. Moreover, pickle file format can automatically represent a large number of Python types (?, ?).

Therefore, curated data set after cleaning, accommodating, and deformation process is stored as binary file format, and for the binary file format, pickle specified for Python is used.

2.3.2 Data index table and separation

Although the big data set can be converted to binary file, it still takes large amount of storage. And it also causes RAM memory optimisation problem. When data curation tasks performed (of course, as well as data analysis task which will be discussed in the next session), processing speed became slower and slower over time. This might be

because data occupies all spaces of hardware memory, and exploits virtual memory on main memory unit. Basically, RAM memory provides very high speed compared to hard disk, and it is even faster than Solid State Drive (SSD) which is often considered 'fast'. If data takes all RAM memory spaces, it transfer addresses of some part of it to virtual memory. This task causes many interruptions to the tasks, and eventually decrease processing performance remarkably.

Therefore, loading only necessary data to a specific task will be a solution to keep processing performance fast. Whole data set was separated by each type of variables. To secure not to lose player information within the divided data, address of each data has to be indexed directing player ID. As a result of data deformation task, 15 pickle data files were generated; eventLabel, date, hour, minute, latitude, longitude, eventValue, play_filter, comb_time, diff_time, local_time, long_gap, shor_gap, sleep_gap, gap_type, and total_plays, which are pointed by individual ID information, eventAction.

2.4 K-Means Clustering with Competitive Learning Algorithm