Proceedings of the Third International Symposium on AI & Games, Daniela M. Romano and David C. Moffat (Eds.),
at the AISB 2010 convention, 29 March – 1 April 2010, De Montfort University, Leicester, UK

Towards Automatic Player Behaviour Characterisation using Multiclass Linear Discriminant Analysis

Robin Baumgarten[1]

Abstract. The automated classification of player behaviour and a subsequent adaptation of game rules in video games is an area of research of interest to both the games industry and to academia, requiring a combination of AI, HCI, psychology, statistics, data mining and machine learning. In this paper, we describe an experiment to answer the question whether it is possible to measure and distinguish behaviour of a player playing a simple game with very limited inputs and a limited variety of possible actions. We use the arcade style action game *PacMan* to record a variety of game-related metrics in an online survey comprised of 245 players, each playing 5 sessions of PacMan. Then, the resulting data is analysed using discretization methods and linear discriminant analysis. The results indicate that differentiation between different players and different playing styles is possible, and that the methods used are suitable to automatically identify the most significant distinguishing features of player behaviour.

1 TEST BEDS FOR PLAYER BEHAVIOUR

A suitable game to research player behaviour on has to allow for a reasonable spectrum of different actions. Games with too limited input dimensions might not allow for sufficiently diverse behaviour patterns. This might make it harder or even impossible to separate player groups according to their actions. Examples of games with a low dimension of inputs are games with a very small state space such as *Noughts and Crosses* and *Rock-paper-scissors*, luck based board games such as *Ludo* and *Snakes and Ladders* and card games such as *Uno* and *Blackjack*.

On the other hand, games that are very complex and allow for a vast number of actions are not suitable for this project, because this freedom of choice often results in a high player behaviour complexity which might adversely affect the clustering process. That is, when a player has a lot of actions to choose from, it is harder to generalise behaviour patterns, because actions are repeated less often and game data (such as keyboard strokes or mouse input) is more diverse. This increased complexity in behaviour patterns might make it harder to separate points in the game data space that is to be clustered, thus requiring more dimensions through additional measurement metrics to mitigate this complexity.

Therefore, a game with an input dimensionality as large as required for a reliable differentiation in behaviour patterns but as small as possible to keep the clustering process efficient has to be found. As the development of video games has seen a steady increase in game complexity, recent titles are less likely to fall into this category than older titles. In fact, old arcade games such as *Pac-Man* and *Space Invaders* are both limited in its input dimensions but at the same time provide sufficient freedom for the player to accommodate for different playing behaviours.

There are additional advantages when using such simple arcade games. In particular, because there is no to little commercial interest in most of these games, it is easier to get hold of games with supplied source code. There are dedicated web pages such as *Galaxy Arcade*[2], *Flash Game Code*[3], and *Geheee*[4] where source code of games can be downloaded.

Furthermore, using small games such as *Pac-Man* allows for online distribution as a browser-based flash game, which simplifies the dissemination of our survey and broadens the target audience significantly. However, Internet based surveys restrict measurement methods to direct inputs to the game, namely keyboard strokes and mouse movement. While it would be interesting to augment the player actions with external input, such as cameras for facial expression detection, acceleration sensors, brain activity scanners, and specialised input hardware that supplies physiological information, e.g. mouse devices with skin conductivity sensors or keyboards with pressure sensors, this information will not be available when the classification algorithm is run in real-world environments, i.e., when used as a tool by game developers to adapt game rules online in video games.

In *Pac-Man* (compare figure 1), the player can navigate in 2 dimensions (up, down, left, right) to eat all pills scattered over the playing area. He has to avoid touching the four ghosts, which hunt him through the labyrinth. He can temporarily turn invulnerable to ghosts when he eats one of the four *power-pills* in the level, and gets bonus points when eating ghosts during this phase. Eating all four ghosts under the influence of a power-pill results in a significant amount of bonus points that are key to obtaining a good score. Occasionally a fruit appears somewhere in the level that can be eaten for bonus points as well.

2 SURVEY METHODOLOGY

In our survey we asked each participant to play 5 rounds of *PacMan*. This enabled us to optimise the result space under the assumption that these five games of a player convey his playing style consistently without too much variation.

[1] Imperial College London, UK, email: Robin.Baumgarten06@doc.ic.ac.uk

[2] Galaxy Arcade: `http://www.galaxyarcade.co.uk/custom_games/whitelabel_games.php`, checked 24/05/2008.

[3] Flash Game Code: `http://flashgamecode.net/tetris`, checked 24/05/2008.

[4] Geheee: `http://www.geheee.com/source.html`, checked 24/05/2008.

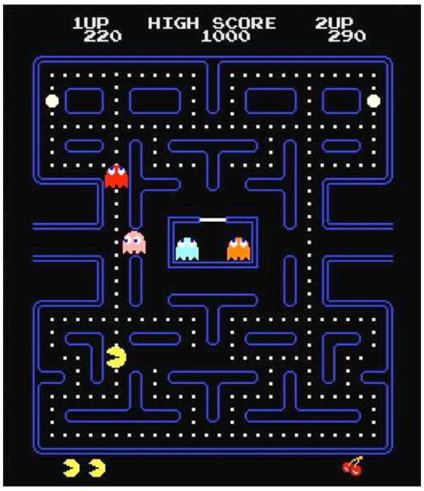

Figure 1. Original PacMan

2.1 Technical Details

A data-logging version of *Pac-Man* has been implemented using Adobe Flash. Using Facebook as survey platform simplified handling user authentication and allows for the collection of simple statistical metrics such as age and gender groups. The game uses a *MySQL* database as back-end and stores all relevant game information, so that games can be fully reconstructed post-hoc. This was then used to identify suitable metrics which are extracted and used in the classification process.

2.2 Recorded Metrics

Apart from the mentioned low level input measurements (keyboard and mouse data), game specific metrics can be recorded. These metrics, including concrete instances for *Pac-Man* are given below.

- Player actions, or type of actions (i.e., aggressive, passive, defensive, mistakes, bonus[5]). In *Pac-Man*, these actions are: eating a pill, eating a power-pill, eating a ghost, getting eaten by a ghost, eating a fruit, finishing a level, and changing the movement direction.
- Rate of player actions, which are often ratios of actions per action or actions per time. In *Pac-Man*, this is pills per minute, ghosts eaten per power pill, ghosts eaten per minute, power pills per minute, key strokes per pill, changes in movement directions per pill[6]
- Reaction time to game events. In *Pac-Man*, the only such events are appearing fruits and the direction changes of ghosts, thus the time-span of fruit appearance to fruit eating can be recorded and possibly ghost-player interaction, which requires a delicate rule to decide when a player input is triggered by ghosts directly. In the current implementation, reaction times are not yet considered.
- Rate of progress through the game. In *Pac-Man*, this is given by the level completion time.

While this information depends on the game used, it is desirable to keep the information sufficiently general, as the classification process is aimed to be game independent[7].

In this survey, the metrics described above have been extracted from the raw MySQL data using a python script, which also prepared the data for input into an *R* algorithm[8].

2.3 Linear Discriminant Analysis

Discriminant analysis is used in statistics and machine learning to characterise or separate classes of objects based on a set of measurable features and class information of these objects. Linear discriminant analysis (*LDA*) utilises a linear combination of these features to (attempt to) separate the groups of objects. This combination can be used as a linear classifier or for dimensionality reduction. In other words, it maximizes the ratio of between-class variance to the within-class variance in any particular data set, thereby guaranteeing maximal separability.

In the context of our experiment, we use LDA to both perform a dimensionality reduction and extract information about player behaviour from the resulting transformed space. The weights of the features in the first dimensions of the LDA transformed space (which we will call *LDA space* from now on) indicate the most important features that determine the behaviour of a player and how it differs from other players. A positive side-effect of this method is that unimportant features are eliminated automatically, and it is not harmful if features are correlated, which is bound to happen as we look at rates of player actions such as pills per minute and ghosts per minute.

As we treat each set of 5 sessions of a player as one class, we need multi-class LDA. This can be understood as an algorithm that finds axes that best separate the classes. The axes define an optimal space through the n-dimensional cloud of data that best separates the session-groups. The rationale of using this approach is that if the sessions of a player are grouped together and the features recorded in the game are used to separate players as much as possible, these distinguishing features become apparent and we have found a space that reflects the playing style and skill of a player. Furthermore, players with similar behaviour and playing style will have a small distance in this LDA space, effectively grouping similar playing styles together. In this sense we can use LDA as a classifier for player behaviour; not necessarily predicting the correct class (i.e., player) for a given object (playing session), but the local group of classes will give an indication of the playing style this player is exhibiting.

Of course, the quality of this indication of playing style hinges on the quality of the linear discriminant analysis. If the measured features are unsuitable for distinguishing player behaviour, or the game does not provide for enough expressivity, a good class separation will not be possible, thus creating a space that does not represent player behaviour and skill well.

A technique closely related to LDA is principal component analysis (*PCA*). While LDA is focused on the difference between the classes of data, PCA does not take these differences into account

[5] A bonus action leads to a higher score but otherwise does not affect the goal of the game, for example eating a fruit in *Pac-Man*

[6] The distinction of key strokes and changes in movement direction aims to capture the difference of interaction behaviour with the game of players; it is possible to press a direction key several times with only one resulting direction change.

[7] This can of course not be validated in a survey with a single game and is therefore subject of further research.

[8] *R* is a programming language specifically developed for mathematical tasks and supports many statistical methods

and only looks at the features of objects. It effectively transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible, thus potentially reducing the number of required dimensions to describe the data. In our experiments, we want to make use of class information as explained above, and thus will not use PCA for our dataset.

3 RELATED WORK

Linear discriminant analysis has been used in areas such as face recognition, bankrupcy prediction[6], marketing and product management[5]. In face recognition research, it is mostly used together with PCA as means of dimensionality reduction of the feature vectors (the resulting vectors are then called *Fisher faces* when LDA is used and *Eigenfaces* in case of PCA)[1, 7].

LDA has also been applied to estimate feature weights for evaluation functions in an Othello game tree[2]. Buro found that choosing quadratic over linear discriminants did not necessarily provide an improvement in estimation accuracy, and the increase in complexity should be carefully considered against the speed and simplicity of LDA. He also applied logistic regression which gave promising results as well.

4 RESULTS

The survey garnered a good response, mainly due to its ease of access and being (indirectly) linked to by several social network sites such as reddit[9] and facebook[10]. This resulted in a dataset of 245 players, each playing at least 5 sessions of *PacMan*. The data was collected over the span of two months. Logging every input, action and event during each session resulted in a dataset of about 4000 entries per session. This was then reduced to the metrics mentioned in section 2.2, and fed into the script that produced a linear discriminant analysis of the data.

4.1 LDA Results

As the axes in a LDA space are ordered from accounting for most between-group variance to least, which is why LDA is commonly used in dimensionality reduction[3, 4]. The first 5 dimensions are enough to account for 98% of between-group variance in our dataset.

A five-dimensional point cloud containing 1225 points and 245 classes is hard to visualise, so we will use projections and subsets of the data here. Figure 2 shows the entire dataset projected onto the first two LDA dimensions. Each number represents a player, and each number will appear five times, its position marking the coordinates of the feature vector defined by one session of that player. Additionally, all sessions of a player are colour-coded. Figure 3 shows a detail of that plot.

To understand what the individual axes of the LDA-transformed space indicate, we look at the base transformation matrix that transforms the result space with (normalised) features as dimensions into the LDA result space. The weights for the first five LDA dimensions are shown in table 1. The biggest absolute values (within 70% of the maximum) in each column have been highlighted.

[9] *http://www.reddit.com*
[10] *http://www.facebook.com*

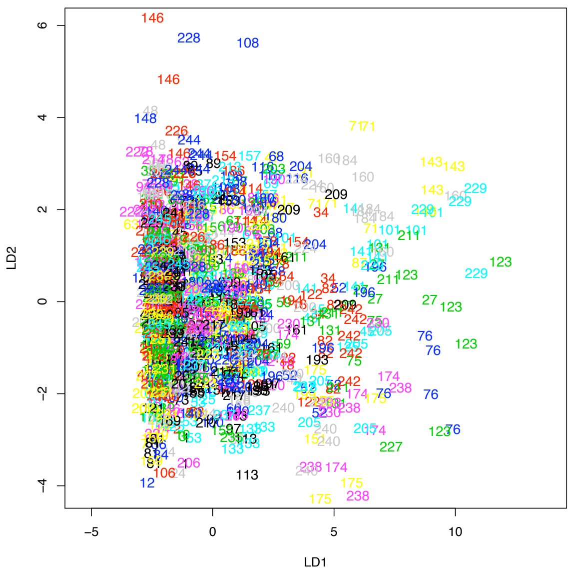

Figure 2. LDA transformed data space, projected onto the first two LDA dimensions. Each number represents a player, and each number will appear five times, its position marking the coordinates of the feature vector defined by one session of that player.

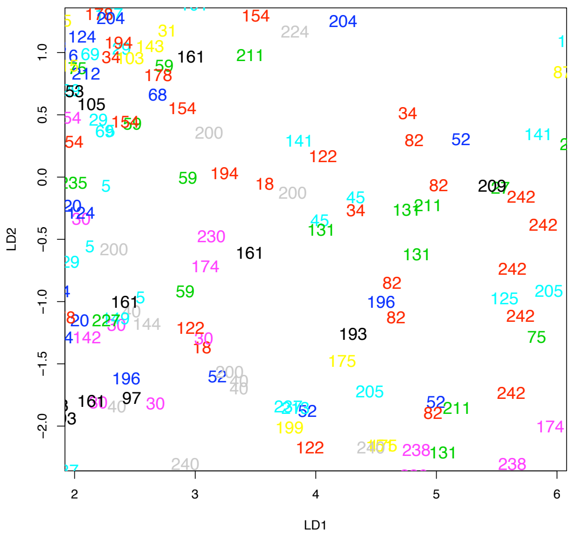

Figure 3. Detail of the LDA transformed data space, projected onto the first two LDA dimensions. Each number represents a player, and each number will appear five times, its position marking the coordinates of the feature vector defined by one session of that player.

	LD1	LD2	LD3	LD4	LD5
totalPills	-0.20	0.43	-1.85	2.22	-1.05
pillsPerLife	0.37	-0.33	1.68	0.38	-1.10
pillsPerMinute	0.21	0.09	0.01	-0.16	0.79
totalGhostsEaten	0.03	0.72	**-3.15**	-2.00	-0.63
ghostsEatenPerMinute	0.05	0.27	-1.00	0.08	**2.09**
ghostsEatenPerPowerPill	-0.09	0.45	0.73	0.04	-0.83
totalPowerPills	-0.14	**-1.32**	**2.92**	**-3.48**	**3.48**
powerPillsPerMinute	-0.25	**-0.95**	-0.38	-0.19	-1.27
eatenByGhosts	0.10	-0.09	0.46	-0.14	-0.32
totalTimePlayed	-0.29	-0.40	-0.23	**-3.17**	-0.51
totalfruitsEaten	-0.03	0.07	0.21	0.09	0.02
totalKeystrokes	0.41	-0.06	-1.59	-0.19	-1.33
keyStrokesPerPill	**2.42**	-0.32	1.16	-0.13	0.86
totalPacTurns	0.25	0.80	-1.03	**2.95**	**3.45**
pacTurnsPerPill	-0.15	0.12	-0.10	-0.15	0.24
totalLevelsCompleted	-0.20	0.43	-1.85	2.22	-1.05
score	-0.04	0.03	**3.92**	1.79	**-2.27**

Table 1. Weights for the base transformation from feature space into LDA space. Each value indicates the influence of a feature to the linear discriminant dimension.

4.2 Interpretation

The projection of the data onto the first two LDA dimension suggests that the sessions of the players share similarities and the chosen features can be used to classify player behaviour. In the graph we can see groups of the five sessions from the same player clustered together, which indicates a similarity between these sessions, in other words the playing style of the player, has been identified. Additionally, the in-class distances are generally much lower than the average in-between-class distances. Although LDA tries to optimise this relationship, data without features that allow for a distinction of these intra- and extra-class behaviour would not show the mentioned properties. The features used in our dataset are sufficient to reveal these differences and we can therefore interpret the LDA space as an approximation to a space where the distance between sessions is determined by player behaviour and style.

To identify what exactly constitutes to player behaviour and style, we look at the linear combinations of features that define the axes in the LDA space.

The first dimension of the LDA transformed space is dominated by the average number of keystrokes required for each pill (compare table 1). Therefore, the most important feature to distinquish different players (and thus playing styles) is how the player physically interacts with the computer. The amount of keypresses required to navigate a level (normalised by the amount of pills eaten to accomodate for the duration of a session[11]) represents both the per-turn efficiency of a player (how often do they press a key for PacMan to change the direction once?) and also the skill to navigate the level (how many turns do they need to eat the pills?).

The second dimension is influenced by several related features: There are large negative weights on the number and speed of power pills eaten and positive weights on the number of ghosts eaten and the number of turns. This indicates that this dimension models a game concept of PacMan: Chasing ghosts with a high efficiency, i.e., eating many ghosts while not using up many power pills. Chasing ghosts usually requires a lot of maneuverability, thus increasing the corners PacMan takes. Or, if the player is on the other end of the spectrum, this means inefficiently eating power-pills without eating many ghosts. One could label this axis as the ghost chaser vs. ignorer dimension.

The third dimension has a high negative weight on ghosts eaten, and a positive weight on score and power-pills. It is less clear how to interpret this dimension, but we can still see that sessions that end up having a high score without the player eating a lot of ghosts will receive a high value in this dimension, i.e., the players that do well but play it safe without taking many risks versus players that end up playing too risky and eating many ghosts but dying earlier.

[11] Sessions do not necessarily have the same length; the player might lose all his lifes quickly and die, thus having a smaller number of keystrokes, or managing to complete the level, which obviously requires more keyboard interaction. The number of keys they need to press for a given time frame is more accurately described by giving a pill-dependent ratio.

5 CONCLUSIONS AND FUTURE WORK

The analysis of the survey data presented in this paper indicates successful detection of player behaviour in game data recorded in a relatively simple computer game. We have shown that linear discriminant analysis is a useful tool to process the data, reduce significant dimensions and identify important features, and will continue to use it as a first step in our player behaviour analysis. Of course, more analysis will be required to further quantify the correctness and usefulness of this approach and compare it to other methods such as principal component analysis and bayesian network classification.

In our data, we were able to determine that the most important gameplay features returned by LDA are also what defines a game in general: physical interaction with the device, and secondly the main goals of the game. We discovered that players can easily be aligned along the dimensions of risk averse over risk accepting to risk seeking. In particular, in PacMan that meant that players tried to avoid ghosts, or had an efficient strategy to deal with them, or risked too much and lost lives early.

These findings are valuable for our project of understanding and optimising adaptive games. After having shown the basic feasibility of recognizing player behaviour through basic data logging, we can now proceed to create an adpative game on the basis of these findings and will use a similar classification system as outlined in this paper to understand a player's playing style.

REFERENCES

[1] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, et al., 'Eigenfaces vs. fisherfaces: Recognition using class specific linear projection', *IEEE Transactions on pattern analysis and machine intelligence*, **19**(7), 711–720, (1997).

[2] M. Buro, 'Statistical feature combination for the evaluation of game positions', *Journal of Artificial Intelligence Research*, **3**(1), 373–382, (1995).

[3] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern classification*, Citeseer, 2001.

[4] M. Loog, RPW Duin, and R. Haeb-Umbach, 'Multiclass linear dimension reduction by weighted pairwise Fisher criteria', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**(7), 762–766, (2001).

[5] M. Mazzocchi, *Statistics for marketing and consumer research*, Sage, 2008.

[6] T.K. Sung, N. Chang, and G. Lee, 'Dynamics of modeling in data mining: interpretive approach to bankruptcy prediction', *Journal of Management Information Systems*, **16**(1), 63–85, (1999).

[7] W. Zhao, R. Chellappa, and PJ Phillips, 'Subspace linear discriminant analysis for face recognition', *Center for Automation Research, University of Maryland, College Park, Technical Report CAR-TR-914*, **137**, 138–144, (1999).