# Mining Rules from Player Experience and Activity Data

**Jeremy Gow** and **Simon Colton**
Department of Computing
Imperial College London
London SW7 2RH, UK
{j.gow, s.colton}@imperial.ac.uk

**Paul Cairns**
Department of Computer Science
University of York
York YO10 5GH, UK
paul.cairns@york.ac.uk

**Paul Miller**
Formerly of Rebellion Developments Ltd
Oxford OX2 0ES, UK
paulm@ieee.org

## Abstract

Feedback on player experience and behaviour can be invaluable to game designers, but there is need for specialised knowledge discovery tools to deal with high volume playtest data. We describe a study with a commercial third-person shooter, in which integrated player activity and experience data was captured and mined for design-relevant knowledge. We demonstrate that association rule learning and rule templates can be used to extract meaningful rules relating player activity and experience during combat. We found that the number, type and quality of rules varies between experiences, and is affected by feature distributions. Further work is required on rule selection and evaluation.

## Introduction

Data analytics has become increasingly popular in the games industry in recent years, with high volume log data collection supporting a range of data-centred design approaches. This presents opportunities to combine game data with measurements of player personality and experience, to produce deeper insights into game design (Yee et al. 2011) and allow greater personalisation of game content (Yannakakis and Togelius 2011). However, the development of specialised tools and techniques to support data-centred designers lags behind our ability to collect mountains of data.

In this paper, we describe a novel approach to generating design-relevant knowledge from integrated game log and player feedback data. *Experience rules* describe the conditions under which specific experiences were reported. Using a simple categorisation of features, we define experience rule templates corresponding to various aspects of game design: the design of level content, the design of adaptive mechanisms, and reflection on connections between player experiences. We present a study of 24 players of the commercial third-person shooter *Rogue Trooper*, in which detailed experience and activity data was captured, and demonstrate that our templates and association rule mining can be used to generate meaningful and design-relevant experience rules. We discuss lessons learned and suggest some directions for further work.

## Background

Player and player experience modelling from log data has been an increasingly well-researched topic over the last few years. Approaches have used a wide range of AI techniques to analyse log and/or experience data. This typically involves classifying players according to an existing model of personality or affect. For example, online summary statistics on player behaviour in *World of Warcraft* have been used to find relationships with, and then predict, player personality profiles (Yee et al. 2011). In contrast, novel player models have been generated from game data using unsupervised approaches, including applying self-organising maps to high volume summary statistics from *Tomb Raider: Underworld* (Drachen, Canossa, and Yannakakis 2009), and identifying player style traits in *Rogue Trooper* with linear discriminant analysis (Gow et al. 2012).

One direction particularly relevant to our work is using game data to model and predict player experience. For instance, using a clone of *Super Mario Bros*, Pedersen et al. (2009) trained a neural network to predict the player experiences of fun, challenge and frustration based on level content. In the educational domain, dynamic Bayesian networks have been used estimate student affect in a simple factorisation game (Conati and Maclaren 2005). Once experience can be predicted reasonably accurately from known data, it becomes possible to generate or adapt content to induce certain experiences, e.g. Shaker et al. (2010) used experience prediction for *Super Mario* to automatically generate level designs. For a more detailed overview of these areas, see Yannakakis & Togelius (2011).

Association rule learning was originally developed to analyse associations between items in supermarket transactions (Agrawal, Imielinski, and Swami 1993). Given a set of items, a transaction database describes a list of observed itemsets. An association rule $A \Rightarrow B$, for disjoint itemsets $A$ and $B$, is a statement about the transactions: whenever a transaction contains the items in $A$, it also contains the items in $B$. Agrawal and colleagues originally introduced the *support-confidence* framework: the *support* for an itemset is the probability a transaction contains it, and the *confidence* of a rule is then $p(B|A) = p(AB)/p(A)$. Rule mining algorithms such as FP-Growth (Han, Pei, and Yin 2000) can generate rules according to predefined minimum support and confidence constraints. A range of alternative rule metrics

| Rule type | Template |
|---|---|
| General | $All^+ \Rightarrow All^+$ |
| Class | $All^+ \Rightarrow All$ |
| Experience | $All^+ \Rightarrow PX$ |
| **Contextual** | $\{IN, CN, OB\}^*, PX^+ \Rightarrow PX$ |
| Observable | $\{IN, CN, OB\}^+ \Rightarrow PX$ |
| **Adaptative** | $\{IN, CN\}^*, OB^+ \Rightarrow PX$ |
| Content | $\{IN, CN\}^+ \Rightarrow PX$ |
| **Dynamic content** | $IN^*, CN^+ \Rightarrow PX$ |
| **Static content** | $IN^+ \Rightarrow PX$ |

Table 1: Experience rule templates. The set of all features $All = IN \cup CN \cup OB \cup PX$.

have been researched (Geng and Hamilton 2006), e.g. lift $p(B|A)/p(A)$, conviction $p(A)p(\neg B)/p(A\neg B)$ and leverage $p(B|A) - p(A)p(B)$. Association rules are a conceptually simple and well-researched area of data mining, with several open source implementations available, e.g. Weka (Hall et al. 2009), presenting a very low barrier to entry for game designers.

## Experience rule templates

Our approach assumes the activity and experience data is structured as a set of *episodes*, each of which corresponds to a period of gameplay. Each episode has an arbitrary number of defined features which we discretise into nominal attributes, giving us a list of episodes (transactions), each defined by a set of attribute/value pairs (itemsets) suitable for association rule learning. In this paper, the episodes correspond to individual combat between the player and a group of NPCs (non-player characters), but they could represent any arbitrary period of gaming activity, e.g. a puzzle, a level, or a month of play.

In order to distinguish rules that might be of interest to designers, we first categorise the episode features:

**Player Profile** ($PP$) Any information known about the player, e.g. genre preferences.

**Combat Design** ($CD$) Features which are predetermined by the episode content, as determined by the game designer, e.g. the initial NPC health and relative positions.

**Initial** ($IN = PP \cup CD$) Initial conditions of an episode, i.e. predetermined features of the player and content.

**Controllable** ($CN$) Features of the game play during the episode that can be manipulated by the designers, e.g. how much NPCs fire.

**Observable** ($OB$) Features which describe the interaction between player and content, which cannot be controlled, but which can be computed directly from the game log.

**Player Experience** ($PX$) Measurements of player experience for the episode (see "Data Capture" below).

In Table 1, various types of association rule are defined using these feature categories. An experience rule has a single PX feature as the consequent, i.e. a class rule for an experience feature. We distinguish four mutually exclusive types of experience rule which might play a role in design:

**Contextual rules** These rules describe the context of an experience: those experiences observed at the same time as the consequent experience. The premise contains at least one experience (PX), along with any other features. These rules might help a designer understand connections between distinct experiences in various gaming contexts.

**Adaptive rules** These capture the directly observable situation associated with an experience. The premise must contain an observable feature (OB), along with other observable and controllable features. These rules could be used in the design of adaptive mechanisms that monitor player experience for an episode and adjust the content of upcoming episodes accordingly. Indeed, a rule-based adaptive system could use the rules directly — a scenario we hope to explore in future work.

**Dynamic content rules** These associate controllable features of the episode activity with a specific experience. The premise contains at least one controllable feature (CN) and other controllable features or initial conditions. They could be used to design dynamic game content aimed at inducing specific player experiences, e.g. the control of NPCs.

**Static content rules** These describe how the initial conditions of an episode can impact on player experience. The premise contain features describing the player's background (PP) and the initial episode configuration (IN). They could be used to reflect on how different types of game content affect different types of player.

Collectively, we refer to these as CADS rules.

## Data capture

To explore the generation and use of CADS experience rules, we conducted a study to capture activity and experience data for combat episodes in the commercial third-person shooter *Rogue Trooper* (Eidos/Rebellion). An instrumented version of the game was developed which, every 0.2 seconds, logged position, orientation and state data for the PC (Player Character) and all NPCs within a given radius, along with in-game events such as damage or item use.

An initial study, in which 10 *Rogue Trooper* players recorded post-game commentaries (Gow et al. 2010), identified 9 dimensions of experience commonly associated with player activity. A 9-item questionnaire was then devised to elicit ratings for these experiences, with each item presenting a pair of opposing statements. The respondent could slightly agree, agree or strongly agree with one or neither statement, providing a rating for that experience on a 7 point scale ($+3$ to $-3$).

The rated experiences were: **Aware** (I was fully aware of the situation / I didn't know what was happening); **Care** (I was careful / I jumped straight in); **Challenge** (The enemy were a challenge / The enemy were easily defeated); **Danger** (I felt exposed to danger / I felt safe from harm); **Engage** (I felt engaged / I felt bored); **Independence** (I was working on my own / I relied on my allies); **Lost** (I was lost / I knew where I was going); **New** (This part felt new / This part felt repetitive); **Purpose** (I knew what to do next / I didn't know how to progress).

For this study, 24 players were asked to play from the start of the first level for at least 20 minutes. They could continue playing for as long as they liked, up to the end of level 3. Players were interrupted every 5 to 10 minutes — when a natural break in play was observed — and asked to complete the experience questionnaire. Hence player experience ratings were recorded for a succession of play segments.

As an aid to recall, the questionnaire was accompanied by a *storyboard prompt*, showing numbered screenshots of the level content played in the preceding segment. These allowed more detailed feedback to be given on how experience varied during the segment: as well an *overall* score for each experience, specific parts of the level could be rated separately by recording the associated screenshot numbers. The questionnaire form was designed to accommodate this kind of detailed feedback. This method allowed players to quickly rate each 5–10 minute segment of play, while also allowing them to record any exceptional experiences within that segment. It supported (but did not demand) feedback on sections of play as small as a few seconds and, if they wished, players were able rate individual combat scenarios.

Along with the player activity and experience data, some brief demographic data was collected: age, gender, frequency of gaming, and a list of their favourite games. A mix of expert and novice gamers were recruited in order to elicit a wide range of behaviours and experiences. The log data, but not the experience data, from this study has been previously used to generate player models (Gow et al. 2012).

## Combat features

In order to define a set of features for each combat episode, we first extracted the segments of log data corresponding to the player engaging in combat. To do this, we identified 43 separate groups of enemy NPCs that the player could fight over the first three levels of *Rogue Trooper*. The groups are encountered in a linear order and typically separately from other groups — although it was possible to fight two groups simultaneously by ignoring one group and moving past them to another, this was rarely observed. For a given PC life, a combat between an NPC group was defined as starting with whichever of the following events occurred first: one of the NPCs fired, an NPC entered a hostile AI state, an NPC received damage, or an NPC damaged the player. As enemy NPCs could sometimes fire at allied NPCs (friendly to the player) long before combat began, firing events that occurred more than 3 seconds before another 'start' event were ignored. Combat between an NPC group and the player ends when the PC's life ends, or the NPCs are no longer logged (death or moved out of range of the PC).

To test how accurately periods of combat-related player activity were identified, the start points of 25 randomly selected combats were determined manually by reviewing screen capture video. On average the two methods were within 0.5 seconds, with no large discrepancies, an improvement over the method used in (Gow et al. 2012).

A total of 633 combat episodes were extracted from 45 levels played. For each combat episode, 118 features were computed: 16 initial conditions describing the type and spatial arrangement of NPCs; 30 controllable features measuring aspects of NPC behaviour, such as how often and at what distance they fired; 61 observable features such as player weapon use, combat time and distance of kills; 2 profile features (how frequently they played games and whether they had a shooter game among their list of favourites); and the 9 player-generated experience ratings for the combat.

A set of nominal features was then defined for each combat by discretising numeric features into High, Medium and Low classes using the unsupervised frequency-based binning filter from the Weka data mining library (Hall et al. 2009). Nominal features were left untouched, except experience features which were also converted to three values. Preliminary experiments showed that it was important that no class value (High, Medium or Low) dominated the discretised feature, i.e. was of much higher frequency than the other two values. Because large values will be associated with many different feature combinations, they can dominant the subsequent rule mining, producing a large number of low-quality rules. For non-experience features we replaced large ($\geq 70\%$) values with an undefined value.

Three of the discretised experience features had dominant ($> 50\%$) values: high Aware and Purpose, and low Lost. Replacing these with undefined values would have destroyed valuable information about these experiences. However, our preliminary results showed these dominant experience-value pairs lead to poor quality rule sets, where the dominant value dominated the experience class rules (68.6%), at the expense of other values. In all three cases, the dominant value represented "agree or strongly agree" with one of the experience statements. To mitigate this effect, we adjusted the discretisation of these three experience features by hand: merging the two other values into a single value, and splitting the dominant value into a large "agree" value and a smaller "strongly agree" value.

## Rule mining

The popular open source Weka data mining library (Hall et al. 2009) was used to mine association rules from the nominal combat feature data. The library provides a number of rule learning methods: we chose FP-Growth (Han, Pei, and Yin 2000) for its superior performance, and used four metrics: confidence, lift, conviction and leverage. The nominal features were converted to binary features for use with FP-Growth. The results below were obtained with Weka 3.7.6.

Minimum metric values were chosen based on the distribution from preliminary results, in order to remove very low quality rules: 0.5 confidence, 1.1 lift and conviction and 0.01 leverage. A minimum support level of 0.1 was chosen so that rules were based on at least 63 combat episodes — we also knew from preliminary results that the number of rules increases dramatically slightly below that point due to a combinatorial explosion. However, further studies could mine rules below that level of support to explore less frequent associations between experience and activity.

FP-Growth was used to generate all rules above a minimum support and metric value (the primary metric), which we then filtered using the CADS rule templates and remaining metric constraints. In theory, the choice of primary metric should not affect the results. However, in practice we

found that Weka returned slightly different results for each metric. For example, using FP-Growth with lift as the primary metric returned a few more rules very near the confidence=0.5 boundary than when using confidence, i.e. Weka appears to be not returning valid rules near the primary metric boundary, perhaps due to using rounded values. For completeness, we ran FP-Growth with each of the metrics and took the union of the rule sets.

## Results

In total, 7395 rules were generated that conformed to the CADS templates and the metric constraints. The rule search and filtering took 14 minutes on a 2.4Ghz MacBook with 4GB of memory available to Java. Of the generated rules, 3266 (44.2%) were contextual rules, 2796 (37.8%) adaptive, 969 (13.1%) dynamic content, and 364 (4.9%) static content rules. Unsurprisingly, the rules generated for each template decreases as the templates get more restrictive.

Three experiences were the consequent of over 1500 rules each: Lost had 1685 (22.8%), Purpose 1624 (21.0%) and Aware 1562 (21.1%). These were three of the four experiences with unbalanced discrete distributions, i.e. they had the one underpopulated category and two highly populated categories. A large number of rules were generated for these large value categories. For example, there were 1328 and 357 rules for high and medium levels of Lost, but none for low levels. Again, this is not surprising: the more combats that belong to a category (e.g. Lost=high) the more premises that will be strongly associated with it. We should be careful when interpreting such categories and rules, as they cover a wide range of player experiences.

Of the remaining experiences, Challenge had 1053 rules (14.2%), Danger 637 (8.6%), Engage — the other unbalanced experience feature — 524 (7.1%), Independence 142 (1.9%), New 132 (1.8%) and Care only 36 (0.5%).

To identify rules that might be of interest to designers, we can use the rule metrics to further filter the results. We define the *top set* as those rules in the top 20% for at least one of the four metrics, which consists of 2149 rules, or 30.4% of the original set. Figure 1 shows the top set broken down by rule type and consequent experience, and Figure 2 by consequent experience and value. We can see that, even for high scoring rules, Aware (22.1%), Lost (16.9%) and Purpose (14.7%) still account for a large proportion of rules due to their unbalanced distributions. However, the proportion of Challenge (26.3%) and Danger (16.2%) rules has risen significantly— in fact, Challenge has the highest proportion of top set rules. Engage (1.9%) and Independence (0.2%) have a reduced proportion of rules, while New (1.4%) and Care (0.2%) remain low.

Only Aware, Challenge and Danger have a large number of high-quality rules for each CADS type, with Contextual being the only rule type that has a high-quality rule for every experience. Care is the only experience with rules for Low, Medium and High levels of experience — the others all contrast two levels with a third neglected. For Challenge and Danger this is High versus Low, but for the remainder the Medium level is contrasted with High or Low, due to the underlying score distributions.

From the distribution of high-scoring rules, we infer that High and Low Challenge and Danger are the easiest experiences to model from this data. Aware, Purpose and Lost have all had their rule sets inflated by their underlying score distributions, which is likely to affect how well the rules model those experiences. It also seems that Independence and Care were the hardest to model using this data and approach. Independence is based by the relationship the player has to friendly NPCs, and our feature set did not measure friendly NPC activity. For Care, it may be that player's belief about how careful they are being is not particularly associated with the actions they take, i.e. there are no good behavioural correlates. Alternatively, there may be too much diversity between different player types for any associations to have been learned.

### Example experience rules

To illustrate the kinds of rules generated by our approach, Tables 2 and 3 show a selection of 29 rules, with rule type, metric values (leverage excluded for space) and rule support, i.e. $p(AB)$. These were chosen because they are short and relatively clear to interpret, score highly for the four metrics, and illustrate the rule types and combat features. Due to limited space, we only briefly discuss how the selected rules in Table 2 could be interpreted.

**Aware** For players who do not favour the shooter genre, Low awareness is associated with being lost and unengaged (C1). In fact, this can be predicted with 0.66 confidence just from a medium level of enemy NPC health (S4), which suggests these players often feel not fully aware of their combat situation. This rises to 0.76 when the NPCs are likely to be actively hostile (D3), i.e. not killed before they notice the player. In contrast, we can predict shooter genre fans are aware of the situation when combats are short (A2).

**Care** Low levels of Care are associated with having a feeling of purpose but not feeling under threat (C5).

**Challenge** When the NPCs are taking little damage, high Challenge is associated with high Danger (C6). Conversely, low Challenge is likely when the player feels safe and independent (C7). Unsurprisingly, death indicates a high level of Challenge when the player has a high rate of receiving damage (A8). For non-shooter fans, combats with an enemy emplacement (bunker) on higher ground are challenging (S9).

**Danger** Challenging and unfamiliar combat makes the player feel in danger (C10). If NPCs are on higher ground and not taking much damage, then we can predict feelings of danger (A11). For people who play games less than once a month, Danger can be predicted with confidence 0.76 (S13), rising to 0.80 when engaging actively hostile NPCs (D12).

**Engage** Low engagement is associated with repetitive combats where the player knows what to do (C14). For shooter fans, starting combats with, and maintaining, high levels of ammunition indicates they are engaged with the game (A15), although the confidence is low at 0.66. For these players, engagement can be predicted with similar levels of confidence just because NPCs are injured rather than
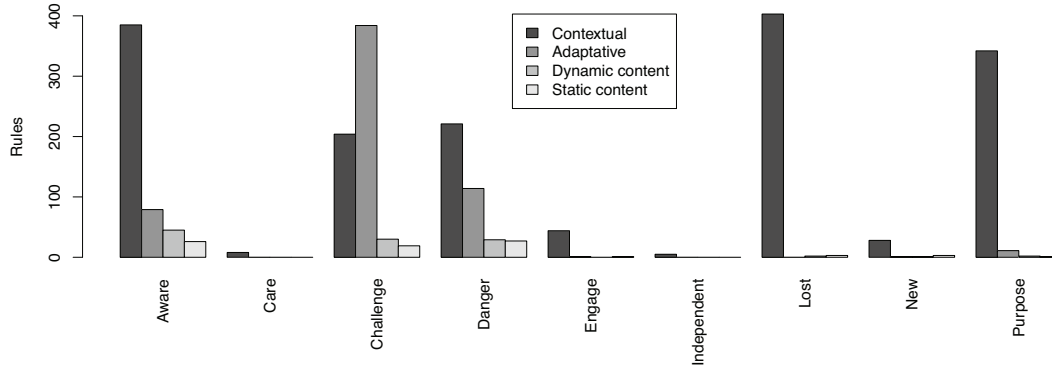
Figure 1: High scoring rules (top 20% for some metric) by consequent experience and rule type.
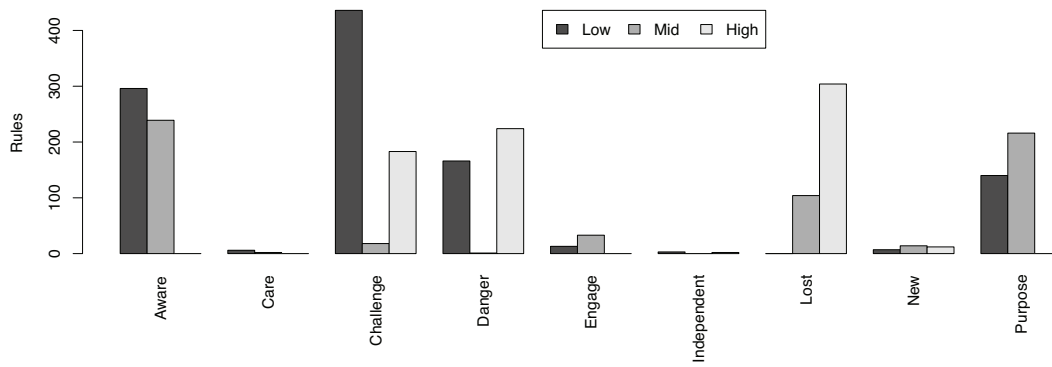


Figure 2: High scoring rules (top 20% for some metric) by consequent experience and value.

| Rule | Premise | Consequent | Sup. | Conf. | Lift | Conv. |
|------|---------|-----------|------|-------|------|-------|
| C1 | $Engaged{=}low, Lost{=}high, genre^{pp}{=}F$ | $Aware{=}low$ | 0.11 | 0.89 | 1.92 | 4.30 |
| A2 | $duration^{ob}{=}low, genre^{pp}{=}T$ | $Aware{=}mid$ | 0.13 | 0.74 | 1.64 | 2.05 |
| D3 | $p.acted^{cn}{=}high, mean.inflict^{cn}{=}mid, genre^{pp}{=}F$ | $Aware{=}low$ | 0.11 | 0.72 | 1.56 | 1.85 |
| S4 | $mean.init.health^{in}{=}mid, genre^{pp}{=}F$ | $Aware{=}low$ | 0.19 | 0.66 | 1.42 | 1.55 |
| C5 | $Danger{=}low, Purpose{=}mid$ | $Care{=}low$ | 0.14 | 0.60 | 1.79 | 1.62 |
| C6 | $mean.take^{ob}{=}low, Danger{=}high$ | $Challenge{=}high$ | 0.12 | 0.88 | 3.12 | 5.48 |
| C7 | $Danger{=}low, Independence{=}high$ | $Challenge{=}low$ | 0.11 | 0.82 | 2.20 | 3.29 |
| A8 | $died^{ob}, pc.dam.rate^{ob}{=}high$ | $Challenge{=}high$ | 0.10 | 0.81 | 2.87 | 3.54 |
| S9 | $has.emplacement^{in}, mean.set.vert^{in}{=}high, genre^{pp}{=}F$ | $Challenge{=}high$ | 0.12 | 0.73 | 2.58 | 2.56 |
| C10 | $Challenge{=}high, New{=}high$ | $Danger{=}high$ | 0.14 | 0.92 | 2.81 | 7.33 |
| A11 | $mean.set.vert^{in}{=}high, mean.take^{ob}{=}low$ | $Danger{=}high$ | 0.11 | 0.75 | 2.30 | 2.60 |
| D12 | $p.acted^{cn}{=}high, often^{pp}{=}Less, genre^{pp}{=}F$ | $Danger{=}high$ | 0.10 | 0.80 | 2.45 | 3.17 |
| S13 | $often^{pp}{=}Less$ | $Danger{=}high$ | 0.12 | 0.76 | 2.33 | 2.72 |
| C14 | $New{=}low, Purpose{=}mid$ | $Engage{=}low$ | 0.11 | 0.75 | 1.87 | 2.28 |
| A15 | $mean.ammo^{ob}{=}high, start.ammo^{ob}{=}high, genre^{pp}{=}T$ | $Engage{=}mid$ | 0.13 | 0.66 | 1.44 | 1.56 |
| D16 | $p.injured^{cn}{=}high, genre^{pp}{=}T$ | $Engage{=}mid$ | 0.12 | 0.60 | 1.31 | 1.33 |
| S17 | $often^{pp}{=}Weekly, genre^{pp}{=}T$ | $Engage{=}mid$ | 0.16 | 0.65 | 1.41 | 1.51 |

Table 2: Selected rules for Aware, Care, Challenge, Danger and Engage. Rule types: C=Contextual, A=Adaptive, D=Dynamic content, S=Static content. A premise $f^c = v$ states feature $f$ is in category $c$ and has value $v$. Value $T = true$ and $F = false$.

| Rule | Premise | Consequent | Sup. | Conf. | Lift | Conv. |
|------|---------|------------|------|-------|------|-------|
| C18 | $Lost{=}high, Purpose{=}low, genre^{pp}{=}F$ | $Indep.{=}low$ | 0.11 | 0.59 | 2.02 | 1.68 |
| A19 | $mean.take^{ob}{=}high, often^{pp}{=}Weekly$ | $Indep.{=}mid$ | 0.10 | 0.54 | 1.25 | 1.21 |
| C20 | $Aware{=}low, Challenge{=}mid, Purpose{=}low$ | $Lost{=}high$ | 0.12 | 0.94 | 1.94 | 7.12 |
| C21 | $Purpose{=}mid, often^{pp}{=}Daily, genre^{pp}{=}T$ | $Lost{=}mid$ | 0.10 | 0.86 | 2.00 | 3.83 |
| S22 | $often^{pp}{=}Less, genre^{pp}{=}F$ | $Lost{=}high$ | 0.10 | 0.68 | 1.40 | 1.56 |
| S23 | $often^{pp}{=}Daily, genre^{pp}{=}T$ | $Lost{=}mid$ | 0.11 | 0.63 | 1.46 | 1.50 |
| C24 | $Danger{=}low, genre^{pp}{=}F$ | $New{=}low$ | 0.11 | 0.71 | 2.80 | 2.50 |
| C25 | $Challenge{=}high, Danger{=}high$ | $New{=}high$ | 0.14 | 0.64 | 1.89 | 1.81 |
| A26 | $mean.hostile^{ob}{=}high, genre^{pp}{=}F$ | $New{=}high$ | 0.11 | 0.50 | 1.47 | 1.30 |
| C27 | $Aware{=}low, Lost{=}high, New{=}mid$ | $Purpose{=}low$ | 0.11 | 0.94 | 2.16 | 8.21 |
| A28 | $p.move^{ob}{=}low, dist.rate^{ob}, genre^{pp}{=}F$ | $Purpose{=}low$ | 0.11 | 0.71 | 1.63 | 1.90 |
| D29 | $mean.p.fire^{cn}{=}high, genre^{pp}{=}T$ | $Purpose{=}mid$ | 0.11 | 0.70 | 1.51 | 1.74 |

Table 3: Selected rules for Independence, Lost, New and Purpose.

killed instantly (D16) or even because they play about once a week (S17). This suggests the rule set's model of engagement is quite weak.

## Conclusions

We have described how association rule learning can be used to mine log and experience data for rules about player experience and its relationship to player activity. These rules encode several types of communicable knowledge about player experience that could inspire and be shared between game designers, or even used to build rule-based adaptive systems. Our current results demonstrate that meaningful and potentially useful rules can be generated from a realistic amount of playtest data.

We have not yet addressed how such rule sets should be evaluated, nor the wider problem of how designers might select and exploit rules in practice. The utility of this approach could be enhanced by specialised tools for filtering and generalising from large rule sets, and relating rules back to specific combat episodes and level content.

This study has shown that some experiences were modelled better than others: Challenge and Danger had a good selection of rule types describing high and low levels of experience. Other experiences were less well captured, perhaps because they lacked behavioural correlates, or because our data did not include relevant features. Experience rating distributions with overly dominant discretised values also affected rule quality. Further work might address how player experience could be recorded so that ratings are less skewed, and not dominated by one of the "agree" responses. Results might also be improved with better feature selection, better discretisation techniques, and separate rule mining attempts focused on specific experience-value pairs. Overall, many high-quality rules used the player profile features, suggesting that more extensive player profiling — perhaps including player traits learned from the combat data (Gow et al. 2012) — would be a fruitful direction of study.

## References

Agrawal, R.; Imielinski, T.; and Swami, A. 1993. Mining associations between sets of items in large databases. In *ACM SIGMOD Conf. on Management of Data*, 207–216.

Conati, C., and Maclaren, H. 2005. Data-driven refinement of a probabilistic model of user affect. In *Proc. 10th Int. Conf. User Modeling (UM 2005)*, 40–49.

Drachen, A.; Canossa, A.; and Yannakakis, G. 2009. Player modeling using self-organization in Tomb Raider: Underworld. In *IEEE Symp. on Comp. Intelligence & Games*, 1–8.

Geng, L., and Hamilton, H. J. 2006. Interestingness measures for data mining: A survey. *ACM Comput. Surv.* 38(3).

Gow, J.; Cairns, P.; Colton, S.; Miller, P.; and Baumgarten, R. 2010. Capturing player experience with post-game commentaries. In *Proc. 3rd Int. Conf. on Computer Games, Multimedia & Allied Technology (CGAT 2010)*.

Gow, J.; Baumgarten, R.; Cairns, P.; Colton, S.; and Miller, P. 2012. Unsupervised modelling of player style with LDA. *IEEE Trans. on Computational Intelligence & AI in Games*. Forthcoming.

Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; and Witten, I. H. 2009. The WEKA data mining software: An update. *SIGKDD Explorations* 11(1).

Han, J.; Pei, J.; and Yin, Y. 2000. Mining frequent patterns without candidate generation. In *ACM SIGMOD Conf. on Management of Data*, 1–12.

Pedersen, C.; Togelius, J.; and Yannakakis, G. 2009. Modeling player experience in Super Mario Bros. In *Proc. IEEE Symp. on Comp. Intelligence & Games*, 132–139.

Shaker, N.; Yannakakis, G.; and Togelius, J. 2010. Towards automatic personalized content generation for platform games. In *Proc. AI & Interactive Digital Entertainment (AIIDE 2010)*. AAAI Press.

Yannakakis, G., and Togelius, J. 2011. Experience-driven procedural content generation. *IEEE Trans. on Affective Computing* 2(3):147–161.

Yee, N.; Ducheneaut, N.; Nelson, L.; and Likarish, P. 2011. Introverted elves & conscientious gnomes: The expression of personality in World of Warcraft. In *Proc. Int. Conf. on Human Factors in Comp. Systems (CHI 2011)*, 753–762.