Predicting DGMS Player Type using Online Steam Activity

Aaron Tabor
University of Saskatchewan
Saskatoon, SK
aaron.tabor@usask.ca

ABSTRACT

In this study, data collected from Steam was used to train an SVM classifier to predict DGMS player type – a personality index tailored specifically to videogames. Data was collected from 168 participants, and included questionnaire responses, participant genre preferences, personal game collection, hours logged within each game, and in-game achievements collected. The classifier was built using Python's SciKit-Learn library, and features were added throughout a series of iterative development phases. While minimal improvements made over the course of these iterations, the mean overall classification accuracy achieved was 57%. The best classification accuracy was achieved when predicting membership in the DGMS-moral category (71%), while the worst accuracy occurred when predicting membership in the DGMS-performance category (35%). Per-development phase classification performance was assessed with a 10-fold cross validation, with a final, unbiased, evaluation perform on 10% of the dataset reserved for final testing.

Keywords

Machine Learning; Supervised Learning; Binary Classification; Support Vector Machine; Digital Games Motivation Scale; Steam

1. INTRODUCTION

Steam is an online community centered around video games where people who enjoy games can purchase, play, discuss, trade, and review videogames. However, the large amount of content available on Steam makes it difficult for players to find new content that they will enjoy. Similarly, it is difficult for the developers and distributors of new content to directly engage their specific target audience because of the huge community of members belonging to the steam site.

Validated questionnaires, such as the Digital Games Motivation Scale (DGMS), exist to categorize an individual's player type, a collection of traits and preferences that describe the games and style-of-play that the individual will enjoy (essentially, a personality index tailored to videogames). The information provided by the DGMS could allow Steam to better recommendation games to players based on their personal preferences, increasing overall sales for the site. However, if player type can be predicted based solely on data captured during gameplay and other activities within the Steam platform, these personalized game recommendations could be made to players without the need for them to first complete the DGMS questionnaire.

In this project, data was collected from 168 participants in the form of DGMS questionnaires responses and data queried from the publicly available Steam web-API. All participants were recruited through the Amazon Mechanical Turk crowdsourcing marketplace. The data was used to construct features based on participant's game genre preferences, participant's game collection, hours that participants have logged in each of their games, and achievements collected during gameplay. These features were used to train a Support Vector Machine (SVM) to predict membership in each of the eight DGMS motivational categories. All machine learning code was written in Python, using the SciKit-Learn library.

Features were added to the model using an iterative development approach, and classification accuracy was assessed at each phase with a 10-fold cross validation. Ten percent of the dataset was reserved for a final, unbiased evaluation, during which an overall classification accuracy of 57% was achieved. The highest classification accuracy was achieved when predicting membership in the DGMS-moral category (71%), while the lowest accuracy occurred when predicting DGMS-performance (35%).

2. BACKGROUND

2.1 Personality Type Questionnaires

Research on personality is a rich field of study within the greater context of psychology. As such, tools have been developed that help researchers identify an individual's personality traits. One early example of such a tool is the Myers-Briggs personality index questionnaire [8], which classifies personality based on four pairs of opposing personality characteristics (e.g., extraversion vs. introversion; reliance on rational thought vs. on feelings or intuition). A more recently developed questionnaire is the "Big Five" personality index [6], which classifies the degree to which an individual adheres to five broad personality traits (extraversion, agreeableness, openness, conscientiousness, and neuroticism). The separate scores achieved for each personality trait come together to form a personality index, which describes an individual's overall personality. Tools like the Myers-Briggs and Big Five questionnaires are important beyond simply describing personality, since their results can also hold predictive power, such as when selecting teams or group members increase the performance in the workplace [9].

2.2 Predicting Personality Type

Many studies have investigated the use of machine learning for predicting personality traits. In these studies, researchers typically have participants complete a personality questionnaire, such as the Big Five, and use the resulting scores as labels for a supervised learning task. The motivation for performing these types of studies is that, if personality can be accurately predicted from easily available information, the benefits of knowing an individual's personality can be achieved without first administering a personality questionnaire.

Researchers have attempted to train personality classification accuracies with different datasets derived from individuals' online activity and behavior while interacting with technology. For example, Chitteranjan et al. [2] used smartphone data collected over an eight-month period (e.g., call logs, SMS messages, app usage) to train an SVM classifier to predict Big Five personality traits. In this prediction task, personality trait scores were simplified to boolean features using the per-trait median score as a cutoff point (i.e., participants scoring above the median were considered to possess the trait). A total of 83 participants took part in their study, and per-trait classification accuracies between 69% and 76% were achieved.

Similarly, Golbeck et al. used data collected from the Facebook [4] and Twitter [5] profiles of 279 participants to train a machine learning personality classifier. Even though the raw data collected from the sites was too sparse to for use in the feature set, Golbeck was able to use summary statistics to accurately predict personality. For example, while participants shared very few Facebook friends in common, and the content of tweets varied widely across participants, summary statistics such as number of friends and bag-of-words style text analysis were sufficient features. In both studies, a per-trait multi-linear regression was used to predict Big Five personality trait scores, and prediction accuracies of within 11% and within 18% were achieved with the Facebook and Twitter datasets, respectively.

While the studies discussed so far have been limited by their small dataset sizes, larger studies have also been conducted using data collected through Facebook. Backrach et al. [1] and Youyou et al. [12] used machine learning algorithms to predict personality with datasets of 5,000 and 80,000 Facebook users, respectively. In a similar fashion as Goldbeck [4], Backrach [1] largely made use of summary features such as counts of friends, photos, and wall posts. On the other hand, Youyou et al. [12] trained their classifier to predict personality using only the participant's "likes". Both studies derived labels from a modified Big Five personality questionnaire, and trained a multi-linear regression model when predicting personality.

2.3 DGMS Player Type

The Digital Games Motivation Scale (DGMS) is a personality questionnaire, much like the Myers-Briggs or Big Five questionnaires discussed so far. However, the DGMS is tailored specifically to identify personality with respect to video games [3]. The questionnaire determines which aspects of video games an individual will find most motivating. The questionnaire's 43 five-point Likert scale questions are divided into eight motivation categories, with scoring computed per-category by finding the average score across the responses in each category. Scores range from 1.0 (not at all motivated by a particular aspect of gameplay), to 5.0 (highly motivated by an aspect of gameplay). These eight category scores can be interpreted as an individual's player type: a summary of the individual's gaming preferences. An example player type is shown in Figure 1.

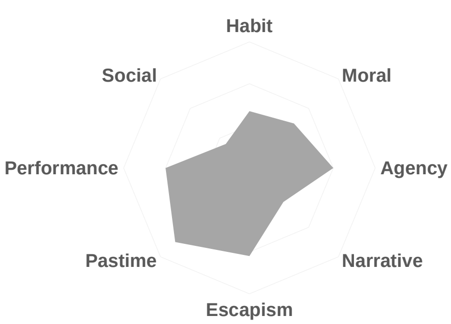

Figure 1: An example DGMS *Player Type*. The graphic summarizes scores achieved in each of the eight DGMS questionnaire categories. High scores are shaded further from the center (e.g., Pastime), while low scores are shaded close to the center (e.g., Social).

Player type can be used to infer which type of games an individual will find enjoyable. For example, a player who scores highly in the "social" and "performance" categories of the DGMS is likely to enjoy games like Call of Duty or Dota 2, which create a social online environment and promote a competitive multiplayer experience. In contrast, a player that scores highly in the "narrative" and "escapism" will be more likely to enjoy games like Mass Effect or Dragon Age, which include a strong storyline and rich character development.

2.4 Steam Online Gaming Community

Fundamentally, Steam is a popular online vendor of video games. However, the Steam website is also a community for gamers within which they can discuss, review, and compare video games with other players. Steam also provides publicly accessible API that can be used to query information about players including the games they own and have recently played, as well as information about their social activity within the site (e.g., comments, discussion posts, game reviews, fan art). In this project, the Steam API (steamcommunity.com/dev) has been used to track participant behavior over a three-month period. Much like the personality prediction studies discussed earlier, the goal of our study was to use data collected from the steam platform to predict a player's DGMS player type.

3. DATA DESCRIPTION

The dataset used in this study was collected from 168 participants, recruited through the Amazon Mechanical Turk platform. Each participant completed a set of questionnaires at the beginning of data collection, and consented to the logging of their online activity within the Steam site for the following three months. The dataset consists of the results of these questionnaires, as well as the Steam data collected over the following three months.

3.1 Labels

As the goal of this project was to perform supervised learning, results from the DGMS questionnaires completed by each participant were used as labels. This resulted in a total of eight label columns (one per DGMS category), each having a value in the range [1.0, 5.0]. Figure 2 presents a summary of participant scores in each of the eight categories.

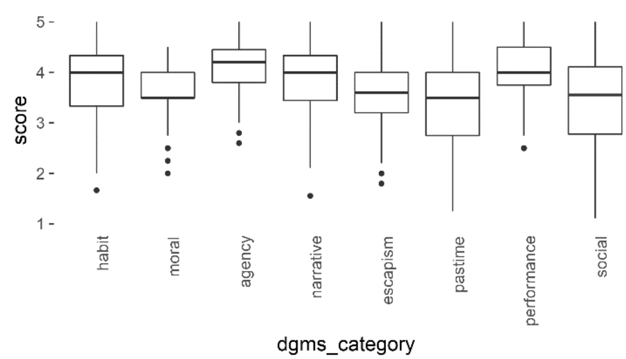

Figure 2: Summary of scores obtained in each of the eight DGMS questionnaires categories. Category scores range from 1.0 – not at all motivating, to 5.0 – highly motivating. Note that mean scores obtained in each category are greater than the 3 – neutral response.

The dataset was originally unbalanced and contained many more positive than negative examples (i.e., more participants who are motivated by a DGMS category than participants who are not motivated by that category). While techniques such as oversampling could have been used to mitigate this issue, the label were instead converted to binary features as was previously done by Chitteranjan et al. [2], taking on the value 1 when a participant scored above the per-category median in each of the eight DGMS categories and 0 otherwise. This approach had the additional benefit of reducing the complexity of the problem from a regression to a binary classification (i.e., *which subset of the DGMS categories is an unseen participant a member of?* – instead of *to what degree is an unseen participant motivated by each DGMS category?*).

3.2 Features

Dataset features came from four categories, selected based on the insights they provide into participant preferences and gaming habits. Each feature category will be introduced immediately, then discussed in more detail below.

1. ***Self-Reported Preferences***
   These features provide a rough insight into game genres that participants enjoy, and were collected as a "check all that apply" style question administered to each participant at the beginning of the study.
2. ***Game Collection***
   These features provide further insight into participant preferences, as these games have been purchased by the participant. This data was collected through the Steam API.
3. ***Hours Logged***
   These features, yet again, provide further insight into participant preferences, as participants will likely log more time on games they enjoy playing. This data was collected through the Steam API.
4. ***Achievements Collected***
   These features provide insight into the progress that participants have made in each of their games. Since people often enjoy activities they are good at, this may provide even further insight into participant preferences. This data was collected through the Steam API.

While the Steam API provides the data for feature categories 2, 3, and 4 on a per-game basis, this resulted in a very sparse dataset (e.g., steam provides a list of all games each participant owns, but there is minimal overlap between participants). Therefore, much like the work of Golbeck et al. with Facebook [4] and Twitter [5] data, summary features were computed and used for classifier training instead. A 3rd party web-service (www.steamspy.com) was used to query genre information for each game that appeared in the dataset, and a set of per-genre summary features was computed for categories 2, 3, and 4. These summary features were computed as the percentage of items within the category that were associated with each genre. As an example, these summary features included the percentage of games in a player's collection that were FPS (first-person-shooter) genre, or the percentage of time spent playing RPGs (role playing games). Since many games are multi-genre, these percentages did not necessarily sum to 100% (e.g., Call of Duty is both a multiplayer and FPS game, and therefore was counted towards both genres). The set of genres used was selected to align with those appearing in the initial "Self-Reported Preferences" questionnaire (see below for details)

3.2.1 *Self-Reported Preferences*

In addition to the DGMS questionnaire completed at the beginning of the study, each participant also indicated their videogame genre preferences in a "check all boxes that apply" style question. A total of 12 genres were included in the available choices, and Figure 3 presents the percentage of participants who indicated a preference for each of the genres. RPG (role playing games), FPS (first-person shooters), action, adventure, and strategy genres were the most popular among participants, while MOBA (massive online battle arena), platformer, and sports games were least popular.

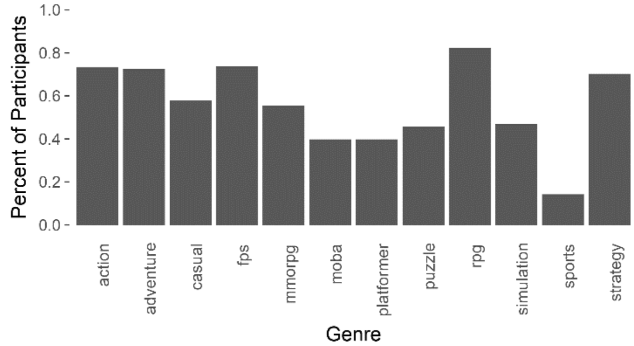

Figure 3: Self-Reported Videogame Genre Preferences. Each bar represents the percentage of participants who indicated that they enjoy playing the respective genre of videogames.

3.2.2 *Game Collection*

Using the list of games owned by each participant provided by the Steam API, a set of per-genre summary features were computed. Since adding a game to your collection typically involves purchasing it, these features provide more information about the genre of games that participants enjoy playing enough to purchase. Figure 4 presents the distribution of participants game collections across each genre. It is interesting to note that, while more participants reported preferring RPG games, participants tend to own more action and adventure games. Participants game collection tended to consist predominantly of action and adventure games, followed by RPG, strategy, FPS, and casual games.

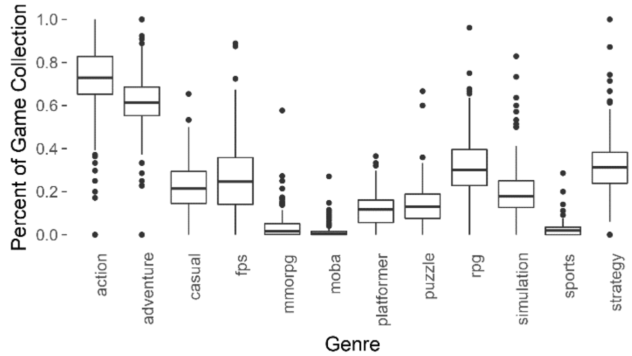

Figure 4: Distribution of participants game collections across genres. Note that, since many games belong to more than one genre, the per-genre percentages across a participant's collections need not sum to 100%.

3.2.3 *Hours Logged*

In addition to the list of games owned by each participant, Steam also provides the amount of time the participant has spent playing each game. This information is valuable because playing a game for a long time would suggest that a participant has even more interest in the game than based simply on the initial purchase of that game alone. Figure 6 presents the percentage of play time that each participant spent playing games from each genre. Again, action and adventure games are most popular, followed by RPG, strategy, and FPS.

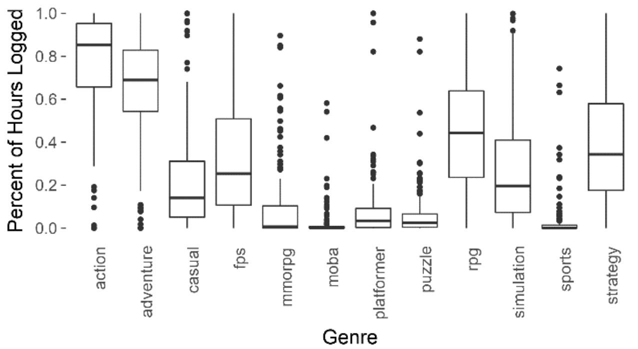

Figure 6: Distribution of time spent playing games from each genre category. Note again that, since many games belong to more than one genre, a participant's per-genre percentages need not sum to 100%.

3.2.4 Achievements Collected

Finally, the Steam API provides the in-game achievements that each participant has collected in each of their game. Beyond time spent playing the game, achievements indicate that the player is progressing through the games objectives. This information could be useful when predicting DGMS player type; for example, players motivated by social or performance aspects of gameplay may collect more achievements than other players motivated by a strong narrative or those who play games simply as a pastime. Figure 7 presents the breakdown of all achievements collected within games of each genre. As was this case in previous categories, the majority of achievements were collected in action and adventure games, followed by FPS, RPG, and strategy games.

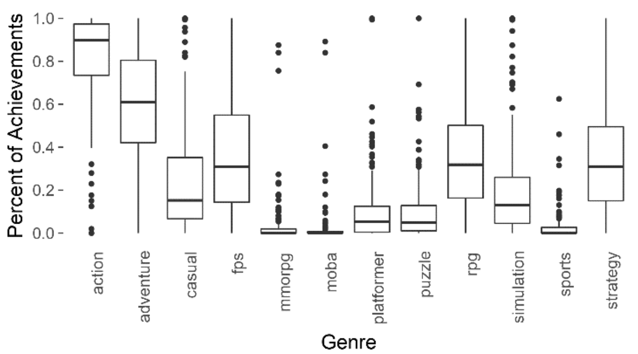

Figure 7: Breakdown of achievements collected while playing each genre of game. Note again that, since many games belong to more than one genre, the per-genre percentages across a participant's collections need not sum to 100%.

4. ALGORITHM DESCRIPTION

The goal of this project was to train a classifier using the features described above such that it could predict which subset of DGMS categories an unseen participant would be a member of. A support vector machine (SVM) was used to perform this classification.

4.1 How does an SVM work?

In its simplest form, an SVM is a binary classifier; an SVM distinguishes one *class* of data points from all other data points. The SVM accomplishes this task by drawing a *maximum separating hyper-plane* that creates a boundary in feature-space between members and non-members of the class. In two-dimensional feature-space, this boundary is simply a line, as can be seen in Figure 5a. In three-dimensional feature-space the boundary becomes a 2-dimensional plane, while in the general

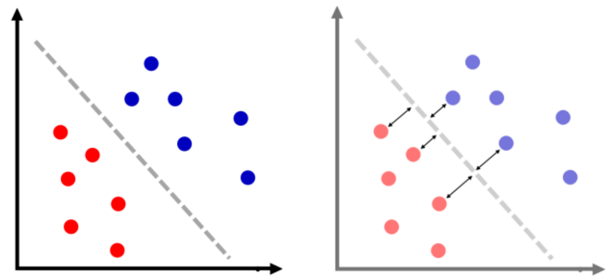

Figure 5: Visual Depiction of a Support Vector Machine (SVM) in 2-Dimensional Feature-Space. A) The SVM works by drawing a boundary (grey dashed line) between one class (blue dots) and the remaining data points (red dots). B) This boundary is said to be *maximally separating*, that is, it maximizes the distance between all support vectors on both sides of the boundary (black arrows).

case, the boundary is an N-1 dimensional hyper-plane (with N being the dimensionality of feature-space). The boundary is said to be *maximally separating*, as it is positioned in such a way that maximizes the mean distance to each data point closest to the boundary (on both the membership and non-membership side of the separating boundary; see Figure 5b). These "closest" data points are termed support vectors, and "closeness" is measured using Euclidian distance. Therefore, it is critical to scale each feature to a common range to ensure a uniform influence across all features.

In the most basic form, the maximum separating hyper-plane is required to be a strict plane in the mathematical sense (i.e., it must be "flat"). SVM's that adhere to this strict requirement are therefore only suitable for predicting classes that are linearly separable from the remaining dataset. However, this constraint can be relaxed using a different kernel function. One commonly used kernel, and the default option in Python's SciKit-Learn library [cite] is the *radial bias function* (RBF). This kernel relaxes the strict linearity requirements of the SVM, and provides two parameters for tuning the model: *C* and *gamma*. The *C* parameter adjusts how flexible or curved the separating boundary can be (i.e., the trade-off between boundary linearity and the inclusion of data points that would otherwise lie on the wrong side of the boundary), while the *gamma* parameter specifies how quickly the influence of the support vectors decays with distance (i.e., how much more influence a support vector that is close to the separating boundary has in determining the boundary's position compared to another support vector that is farther from the boundary). While these parameters can lead to improved classification accuracy, they also increase the risk of overfitting to the specifics of current training set. Since a very small dataset was used in this project, the strictly linear version of the SVM was used for all classification to avoid overfitting.

Whether the boundary is flat or curved, one interesting implication of using a concrete boundary is that the SVM classifier is not probabilistic. Unlike the Naïve Bayes or Hidden Markov Models studied in class, the SVM does not report a probability distribution for the class of an unseen data point. Since the SVM's prediction is based solely on the position of the unseen data point relative to the maximally separating boundary, predictions will be binary: either a member (i.e., data point lies beyond the boundary), or non-member (i.e., data point lies before boundary) of the class of interest. While this binary decision may initially seem limiting when compared to other models like Naïve

Bayes because it eliminates the notion of a level of confidence in a prediction, it has the benefit of being extremely computationally efficient, even in high dimensional feature spaces; once the SVM has been trained, a prediction is made by simply checking which side of the boundary the new data point lies in.

4.2 Tailoring Dataset for use with an SVM

I made several pre-processing modifications to the features in this dataset before using them to train the SVM classifier. As mentioned earlier, each of the eight DGMS category scores was simplified from a continuous variable in the range [1.0, 5.0] to a binary feature indicating a participant's membership in that category. While, as previously discussed, this was done in part to simply the complexity of the problem, mitigating the small dataset, it also creates a machine learning situation that is naturally suited to be addressed by an SVM classifier. More specifically, eight separate SVM classifiers – one per DGMS category – were trained and used to predict the per-category membership of an unseen participant.

Following this label simplification, each feature of the dataset was scaled to a common range ([0,1] for continuous features, 0 – false, 1 – true for boolean features) to ensure a uniform influence across all features during the SVM's Euclidian distance calculations.

Finally, a principle component analysis (PCA) was used to reduce the number of features in feature categories #2, #3, and #4 from twelve features to three features (i.e., $3 = \texttt{floor}(\log_2(12))$). The motivation for performing this dimensionality reduction was because games appearing in the dataset that belonged to more than a single genre were captured in more than one feature. For example, since Call of Duty is both a multiplayer and FPS title, any hours that a participant logged playing Call of Duty contributed to both their percent of time spent playing multiplayer games and first-person-shooter games, and leads to interdependent features. Since the best classification performance is typically achieved with a small number of independent features, the PCA was performed to consolidate these features, conceptually identifying the most pronounced intra-genre patterns within the dataset.

4.3 Development and Evaluation

The dataset used in this project was collected using custom python scripts. The Steam (steamcommunity.com/dev) and SteamSpy (www.steamspy.com) APIs were used to query participant and game genre information, respectively. All machine learning classifiers came from the Python SciKit-Learn library [10]. Specifically, the LinearSVM classifier, and StandardScaler and PCA utility classes were used during development.

An iterative design and development approach was adopted. At the beginning of the project, 10% percent of the dataset was set aside to be used as final, unbiased, evaluation of classification accuracy. Using the remaining 90% of the dataset, a 10-fold cross validation was used to evaluate the classifier at each phase of the iterative development. At each step of the iterative design, one of the four feature categories was created and added to the classifiers feature set.

With such a small data set, achieving high classification accuracies is unlikely. However, previous studies have achieved reasonably accurate predictions of personality traits (e.g., "to within 11% of their actual values") using datasets of a similarly small size (N = 279) [golbeck2011-facebook, golbeck2011-twitter]. Furthermore, as discussed above, predictions in this project have been simplified from regression to binary classification, a decision made to mitigate the small number of data samples.

In this project, even marginal classification accuracies will be considered acceptable for the problem domain as the risks associated with mis-classification are quite low (e.g., a player may be recommended a game that the will not like, or may miss out on a game they could have enjoyed). Therefore, gaining any additional insight into a player's player type – and consequently their gaming interests – will be considered as success.

5. RESULTS

Classification accuracies for each of the eight DGMS categories are presented in Table 1. In each iterative phase of development, the classification accuracy reported is the mean accuracy achieved across the 10-fold validation, while accuracies reported for the final evaluation were obtained by training the classifier with the full dataset used in development, then tested on the reserved 10% of data samples. The classifier for DGMS-moral achieved the best classification accuracy (71% in final evaluation), while the classifier for DGMS-performance achieved the worst (35% in final evaluation).

	User-Specified Preferences	Game Collection	Hours Logged	Achievements Collected	Final Evaluation
Habit	0.62	0.60	0.65	0.65	0.64
Moral	0.76	0.74	0.76	0.76	0.71
Agency	0.52	0.58	0.56	0.59	0.65
Narrative	0.61	0.58	0.66	0.63	0.53
Escapism	0.58	0.62	0.60	0.57	0.53
Pastime	0.54	0.57	0.53	0.54	0.65
Performance	0.66	0.64	0.63	0.66	0.35
Social	0.50	0.58	0.56	0.60	0.53
Overall	**0.60**	**0.61**	**0.62**	**0.62**	**0.57**

Table 1: Classification Accuracies. Results are presented per-category, as a separate classifier was trained for each DGMS category. The first 4 columns present accuracies achieved after each iterative phases of development, while the final column presents accuracies achieved with 10% of the dataset withheld during development.

With such a small dataset (the reserved testing set contained only 17 samples), it is difficult to explain this behavior; classifier performance will be sensitive to the specific data samples in the reserved testing set. However, I believe that the per-category classifier performance is related to the variance in the original DGMS category scores (Figure 2). Even though these scores were reduced to binary features, low variances in DGMS category score could suggest more uniform behavior across participants with respect to that motivation category. For example, the low variance of the DGMS-moral category suggests that participants in this study exhibited very similar behavior with respect to this category, and – since the behavior was so uniform across participants – the classifier was able to learn distinguishing patterns between morally motivated participants and non-morally motivated participants. This does not mean, however, that the classifier would be able to accurately predict members of the morally-motivated DGMS category in general.

I find several aspects of these results particularly interesting. First, only very marginal (if any) gains in classification performance were achieved through my iterative design phases. Even though my new features brought additional information about each player's interests, they also increased the dimensionality of the machine learning problem. Secondly, while classification performance remained stable across iterative development phases, sharp changes occurred in some categories when predicting the class of new data points during the final evaluation. For example, DGMS-performance dropped substantially (from 66% → 35% accuracy), while DGMS-pastime achieved increased performance (from 54% → 65%). Again, it is difficult to infer any general insight from these fluctuations because of the small dataset size.

6. DISCUSSION

One aspect of this study that must be carefully considered before generalizing these results is that the dataset consisted largely of participants who have highly self-identified as "gamers". While a dataset skewed towards a specific demographic is undesirable in general, the DGMS was designed specifically for use with video games and gamers, so in this situation the dataset may be appropriate.

However, DGMS category scores were also skewed towards the upper end of the scale (e.g., most participants reported being highly motivated by most aspects of gameplay – see Figure 2). In this project, I treated the median score of each category as a cut-off point for membership within the category. However, this was an arbitrary cut off point and may not be a valid assumption in general, since many of the participants who scored above "3 – neutral" are ultimately not labeled as member of that DGMS category in my dataset. Before generalizing these results, it will be important to determine and address the cause of the highly skewed DGMS scores observed in the dataset.

Another potential issue with the study is that the genres used to summarize feature categories were selected largely though an arbitrary process. In Steam, game genres are assigned through a user-driven tagging process. For a particular game, the genre is considered to be the single tag (or set of tags) with the highest number of user votes. This presented several challenges.

First, there are an incredibly large and diverse number of user-defined tags in use on the Steam website. I chose to work with only the small subset of these tags that most closely aligned with the list of genres presented to participants in the initial background questionnaires at the beginning of the study. Some genre-tag mappings were quite easy to identify (e.g., "Adventure" genre – "Adventurer" Steam tag), but others were more ambiguous ("Multiplayer" genre – "Multiplayer" tag, "Online Multiplayer" tag, "Massively Multiplayer" tag). In situations where the mapping was ambiguous, I opted for the more commonly used tag across the entire Steam site. Since each game can be assigned many tags, I believe this was a valid assumption.

Furthermore, many games on Steam are highly tagged as more than one genre (e.g., FPS and Multiplayer). Whenever a cross-genre game was encountered, the game was counted once for each of its genres. This has the consequence that the set of per-genre features may not necessarily sum to 1.0 for each participant, but I believe it was an appropriate choice because it fully represents the characteristics of the game.

One final issue to consider with this project is that only marginal performance improvements occurred throughout the iterative development phases. While the motivation for including these additional features in the dataset was introduce more insight into players underlying motivations, the shape of each of these feature categories is very similar (i.e., there are many similarities between Figures 3, 4, 5, and 6). This suggests that the new features did, in fact, not introduce a substantial amount of new information about participants, and the cost of their added dimensionality likely outweighs any benefit they provide. Therefore, it may be worthwhile to further investigate the performance of DGMS classification accuracy using a feature set consisting solely of achievements collected.

7. FUTURE WORK

In this project, only a fraction of the player information available through the Steam API was used. The API also exposes many text-rich fields such as the comments and game reviews that players have posted to the site. Analyzing the fields with bag-of-words or sentiment analysis techniques could provide a substantial amount of information about player's likes and dislikes, which could further improve DGMS classification performance. Similarly, fields like the player's profile will likely hold much more feature-rich information that could be used to perform machine learning.

8. SUMMARY

In this study, data collected from Steam was used to train an SVM classifier to predict membership in each of the eight DGMS player type motivational categories. The dataset was constructed using questionnaire responses and data collected from 168 participants. The overall classification accuracy achieved was 57%, with highest classification accuracy in DGMS-moral (71%), while the lowest occurred with DGMS-performance (35%).

9. REFERENCES

[1] Bachrach, Yoram, Michal Kosinski, Thore Graepel, Pushmeet Kohli, and David Stillwell. *Personality and patterns of Facebook usage.* In Proceedings of the 4th Annual ACM Web Science Conference, pp. 24-32. ACM, 2012.

[2] Chittaranjan, Gokul, Jan Blom, and Daniel Gatica-Perez. *Who's who with big-five: Analyzing and classifying personality traits with smartphones*. In Wearable Computers (ISWC), 2011 15th Annual International Symposium on, pp. 29-36. IEEE, 2011.

[3] De Grove, Frederik, Johannes Breuer, Vivian Hsueh Hua Chen, Thorsten Quandt, Rabindra Ratan, and Jan Van Looy. *Validating the digital games motivation scale for comparative research between countries.* Communication Research Reports 34, no. 1 (2017): 37-47.

[4] Golbeck, Jennifer, Cristina Robles, and Karen Turner. *Predicting personality with social media.* In CHI'11 extended abstracts on human factors in computing systems, pp. 253-262. ACM, 2011.

[5] Golbeck, Jennifer, Cristina Robles, Michon Edmondson, and Karen Turner. *Predicting personality from twitter.* In Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on, pp. 149-156. IEEE, 2011.

[6] John, O. P., Donahue, E. M., & Kentle, R. L. (1991). The Big Five Inventory--Versions 4a and 54. Berkeley, CA:

University of California,Berkeley, Institute of Personality and Social Research.

[7] Murphy, Kevin P. *Machine learning: a probabilistic perspective – chapter 14.5: Kernels – Support Vector Machines*. Cambridge, Mass.: MIT Press, 2013. Textbook: Machine Learning,

[8] Myers, I.B., 1962. The Myers-Briggs Type Indicator: Manual (1962).

[9] Omar, Mazni, Sharifah-Lailee Syed-Abdullah, and Naimah Mohd Hussin. *Developing a team performance prediction model: A rough sets approach.* In International Conference on Informatics Engineering and Information Science, pp. 691-705. Springer, Berlin, Heidelberg, 2011.

[10] "Scikit-learn." Scikit-learn: machine learning in Python — scikit-learn 0.19.1 documentation. Accessed December 13, 2017. http://scikit-learn.org/stable/index.html.

[11] Shalev-Shwartz, Shai, and Shai Ben-David. *Understanding machine learning:from theory to algorithms – chapter 7: Linear Models.* New York: Cambrige University Press, 2016.

[12] Youyou, Wu, Michal Kosinski, and David Stillwell. *Computer-based personality judgments are more accurate than those made by humans.* Proceedings of the National Academy of Sciences 112, no. 4 (2015): 1036-1040.