A Game-Theoretic Analysis of Kaggle

Sam Schwager, Sam Sklar, John Solitario SUNet ID: sams95, ssklar2, johnny18

December 10, 2018

1 Introduction

Since the advent of the Internet, organizations have devised a variety of methods for sourcing solutions from large-scale communities, and organizations have leveraged "crowdsourcing" to tackle a variety of tasks. Although a multitude of crowdsourcing models exist, we choose to focus on open-entry contests for organizing algorithm competitions, as they apply directly to Kaggle. Furthermore, we explore a subset of recent research that models open-entry, algorithm competitions as side-choosing games and another subset of research that investigates incentive structures meant to maximize the total number of participants.

In our review of the literature, we note that little attention has been paid to the organization of crowdsourced projects. Furthermore, the majority of research assumes a fixed design (i.e. Kaggle's current structure) for crowdsourced projects. In section 5, we further explore Kaggle as a data science competition platform and identify key issues with the platform. In section 6, we propose a new version of the Kaggle platform, which better distributes participants across the various competitions. We then run a simulation of our "new" Kaggle platform, which better distributes participators across current competitions. In sections 7 and 8, we propose and discuss two more alternatives to the current Kaggle platform. Finally, we conclude with a discussion about how our new versions could impact the Kaggle community.

2 Kaggle

Kaggle is a crowd-sourced platform that connects data scientists from all over the world with organizations seeking insights from their data. In 2017, Kaggle passed one million users and was acquired by Google. Currently, Kaggle employs three main incentives to bring users to the platform. First and foremost, many of the competitions feature large cash prizes for the winners. Currently, the most popular competition on the site is hosted by Two Sigma. The hedge fund is offering $100,000 to the participants who can best predict stock price movements from available news data. In addition to monetary incentives, many Kaggle users advertise their activity on the site as a portfolio for potential employers. A history of demonstrated success on the site can help those with nontraditional educational backgrounds gain exposure to top tech companies. Finally, Kaggle fosters a community dedicated to learning and skill building. Kaggle forums are active with idea sharing and collaboration.

Each Kaggle competition is unique, but there are some commonalities. The setup in virtually all cases involves companies submitting labeled data on which users train their models. There is no limit to the number of participants in a single competition and there is no limit to the total number of competitions in which any given user can participate. Companies can, however, limit the number of model submissions by an individual participant. Usually, users can submit no more than two models per day. This forces users to be thoughtful throughout the process of building their models since parameter tweaking and other typical "tricks" will not be conducive to a successful submission. Also, companies tend to asses models with methods typically crafted to penalize those that over fit the given training data.

In one current competition, Quora asks users to predict whether or not a question is sincere. Users can apply whatever model they please, using any programming language, and they simply submit their predictions on a test dataset. In the case of the Quora competition, the F1 score between the user's predictions and the ground-truth values is used to assess performance.

Kaggle tracks performance by maintaining a ranking of its users. The Kaggle progression system buckets users into one of five possible categories (see Figure 1).

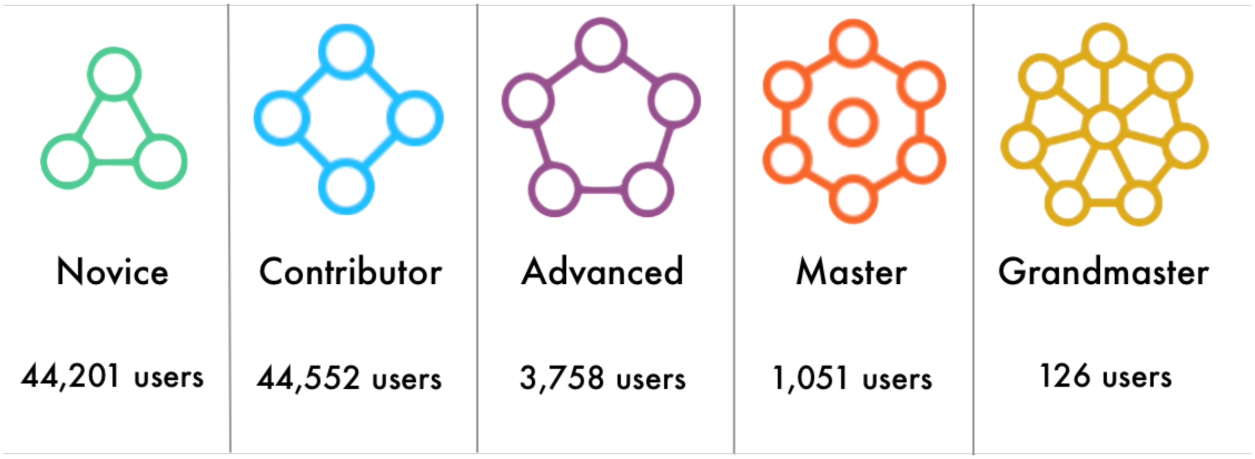


Figure 1: User skill level on Kaggle.

3 Related Works

3.1 Side-Choosing Games

Abdelmeged, Xu, and Liebherr introduce SCGs (referred to as Side-Choosing Games or Scientific Community Games) as mechanisms for managing the creation and dissemination of formal knowledge [6]. To model Kaggle as an SCG, we must first narrow the scope of Kaggle competitions to prediction and classification problems, which require the submission of algorithms to make explicit decisions.

First, an SCG relies on an extensive two-player, zero-sum game, G, between two given players p_x and p_y. In the case of Kaggle, we model G as one of the 12 active paying competitions, and we identify p_x and p_y as models submitted by two unique participants in the competition. Furthermore, GS represents a game state of G, while Q is a proposition on the game state GS. Within a Kaggle competition, GS could represent a specific classification or prediction task (i.e. performed on one row of data), while Q could represent a claim relevant to the given task. For instance, given an image classification task G, GS would represent a given image, while Q would represent a claim about the given image.

Second, based on model characteristics, players p_x and p_y of an SCG have preferred static sides, which depends on whether they believe a given claim, Q or $!Q$, to be true. A specific instance of the side-choosing game $\langle G, GS, Q, p_x, p_y \rangle$ produces a result-row consisting of (1) the winner, (2) the loser, and (3) at most one forced player (a forced player decision is likely to occur at least once per game, as p_x's decisions may mirror p_y's decisions, without loss of generality). A set of game results produced by multiple binary SCGs is called an SCG-Table.

Referring back to Kaggle, the SCG-Table breaks down the classification or prediction results made between two submitted models, and the competitor with the superior model would win the overall SCG. For the SCG characterization to hold, we must assume that a ground truth exists for each claim (i.e. each claim has a ”correct” answer, which typically takes the form of labeled data).

Although Kaggle does not explicitly run SCGs, the SCG format encapsulates the underlying mechanism of comparing any two models within a given competition. Furthermore, although characterizing Kaggle competitions as a series of SCGs requires some assumptions and generalizations, it allows us to formalize several key advantages of the Kaggle format:

A. Distributed evaluation of players without a central authority.

B. Collusion resistance: impossible for players to collude and make a skilled or even perfect player lose.

C. Overhead of learning SCGs is amortized over a large number applications.

D. Low barriers to entry: clever ideas are demonstrated by systematically defending a chosen side.

E. Objective: results depend on how participants solve the computational problems underlying a claim.

3.2 Incentive Structures

A vast amount of research has explored the design of optimal labor contracts, as well as rank-order contests as incentive schemes for procuring labor. Furthermore, over the past several years, additional literature has studied incentive structures as they relate to payouts on crowdsourcing platforms.

Yan et al. categorize Kaggle competitions as specific types of crowdsourcing projects, regarded as trial-and-learn projects [3]. In a trial-and-learn project, participants explore a well-defined problem with a pre-determined evaluation metric, and furthermore, participators can make multiple submissions and improve their submissions by leveraging their expertise. Given the characterization of Kaggle competitions as trial-and-learn projects, Yan et al. prove the following theorem, under their given assumptions:

Theorem (Maximum Number of Contributors): *Given the award A and entering cost c_f, the number of rational contributors n is $\sqrt{\frac{A}{c_f}}$ under "winner-takes all" and $n\prime \in (\sqrt{\frac{A}{c_f}}, 2\sqrt{\frac{A}{c_f}})$, under "Top-3".*

Thus, Yan et al. argue competitions with higher award amounts and Top-3 award structures (i.e. the top-3 participants recieve a portion of the total payout) will attract more contributors. However, Yan et al. rely on the classification of participators as rational contributors.

Easley and Ghosh also explore participation incentives for crowdsourced projects with additional acknowledgement to prospect theory preferences, originally pioneered by Tversky and Kahneman [1], [2]. Specifically, they argue that a small chance of winning a large prize might contribute more than its 'true' share of utility as potential contributors will overweight small probabilities, which in turn creates a larger perceived payoff for the same expected payout. Thus, prospect theory helps explain why participants may flock to the highest paying competitions, even if their probability of winning the given competition remains extremely low.

4 Definitions

For the sake of showing how Kaggle can be improved, we will quantify our currently abstract notions of contributor and organization welfare.

4.1 Individual Welfare

Remember that people can get value from participating in three ways: monetary reward, knowledge, and résumé boost. Let's call welfare w, monetary payout m, knowledge gained k, the résumé boost v, and the user's ranking compared to other competitors i. Although m and v clearly depend on i, we assume that the knowledge gained from participating does not. Thus, we have:

$$w = m_i + v_i + k \tag{1}$$

Organizations have the flexibility to divide prize money among multiple participants. The most common structure pays out the majority of the purse to the first place submission followed by increasingly smaller values for players 2, 3, 4 and 5. Thus, m_i satisfies

$$m_i = 0, \forall i > 5 \tag{2}$$

Moreover, many users report that if you do not consistently place within the top 1% of participants in a competition than recruiters will not be sufficiently impressed. As a result, when we have n total participants:

$$i/n > .01 \implies v_i = 0 \tag{3}$$

Next we will define the welfare derived by the organization that poses the competition.

4.2 Competition Host Welfare

Broadly, organizations can gain two benefits from hosting a Kaggle competition:

A. A better model for understanding a dataset relevant to their field.

B. Access to strong data science candidates who are interested in working with the given company's data.

Now we will make one of our key assumptions about the nature of the host's welfare.

The organization does not derive any value from the least effective participants (i.e. those who are far behind the winning participants). We make this assumption because there is a sharp drop off in the quality of models outside of this range. The lesser quality models either are far simpler than those of the winners or reflect a misunderstanding of the data being studied. Moreover, those participants have failed to display the quality necessary to be a candidate for employment for the company in question.

We will define organizational welfare in terms of the quality of the top participants. Kaggle maintains a score for all of its 93,397 participants that depends on their performance on the platform.

Call the Kaggle score of each user s. Now we can define the welfare to the competition host in terms of the Kaggle scores of its top k participants, $s_1, s_2, ..., s_k$. Thus, we represent organizational welfare as:

$$W = \sum_{i=1}^{k} s_i \tag{4}$$

With this concrete definition of welfare established, all that remains is a proper choice of k. We propose choosing $k = 85$ due to empirical analysis that we discuss in Section **6.4**.

5 Analysis of Kaggle

5.1 Advantages

Kaggle has successfully proctored 292 competitions and currently has more than 16 active competitions. Aside from creating a massive marketplace and earning a hefty acquisition, Kaggle has been successful in several key areas:

A. Giving both the private and public sector a way to crowdsource big data problems.

B. Helping data scientists bolster their résumés, while also competing for prize money.

C. Providing a platform for experienced data scientists to share information both among one another and with newcomers.

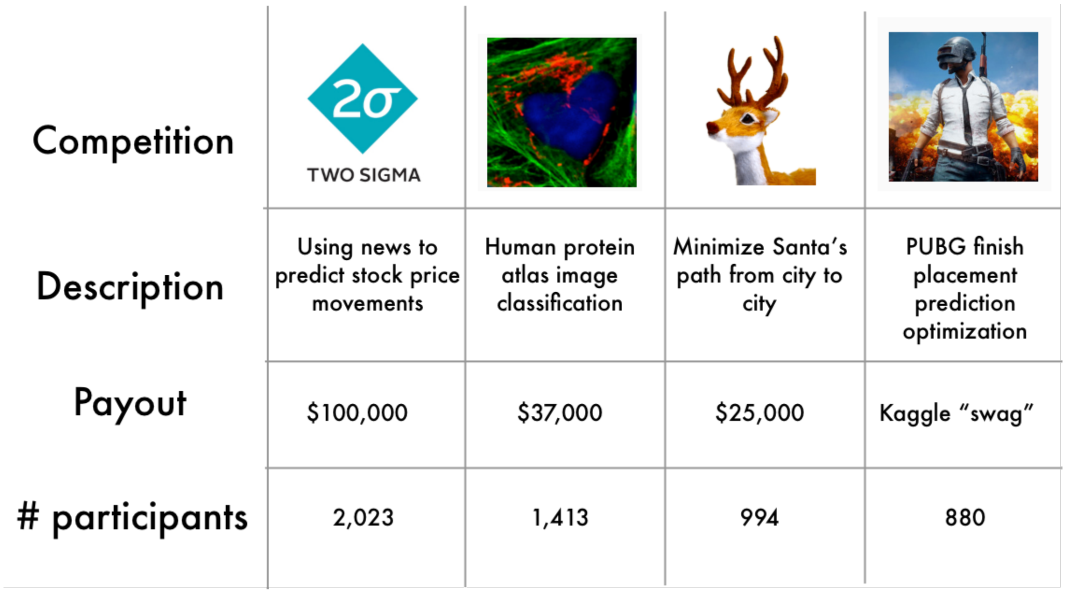
5.2 Issues

5.2.1 Pareto Optimality

We argue that Kaggle in its current form is not a Pareto Optimal platform. To see why this is the case, consider the Two Sigma competition, which is currently the highest paying open Kaggle competition. Now consider some data scientist who comes to Kaggle in search of a new project to work on. Since the Two Sigma payout is significantly larger than the payouts from the other competitions, the data scientist will likely decide to work on the Two Sigma competition. However, this data scientist will most likely provide no benefit to the Two Sigma competition since the best 7 data scientists (the only ones that receive a reward) are most likely already in the group of 2,023 present before this new data scientist joined. Even if we extend the notion of “meaningful” competitors to the top 85 (as proposed in Section **6.4**), it is still likely that the top 85 were present in the initial group of 2,023. Nonetheless, it is likely that there is another Kaggle competition to which this new data scientist could have made a meaningful contribution. Since we could make this other competition better off without making the Two Sigma competition worse off by moving this new data scientist to the other competition, we see that the existing Kaggle system is not Pareto Optimal.

5.2.2 Deviation from Obvious Best Strategy

The majority of people on Kaggle tend to participate in the competition with the highest payout. For instance, Kaggle has 16 active competitions as of December 3, 2018. Out of the 16 active competitions, 4 of those competitions will close within the next month.



Competition	TWO SIGMA			
Description	Using news to predict stock price movements	Human protein atlas image classification	Minimize Santa’s path from city to city	PUBG finish placement prediction optimization
Payout	$100,000	$37,000	$25,000	Kaggle “swag”
# participants	2,023	1,413	994	880

Figure 2: Active Kaggle competitions closing within the next month

From Figure 2, we can see that the majority of active users on the platform participate in the competition with the highest payout. However, as explained in the “Individual Welfare” section, a participant’s welfare from a competition depends on three inputs: monetary payout, knowledge gained, and résumé value. We assume knowledge gained remains fixed from competition to competition, as most competitions present similar learning opportunities. In order for a participant to receive any monetary payout, they must place in the top K competitors, which generally ranges from 3 to 7. More so, a participant must finish in the top 1% to receive any résumé value. By participating in the competition with the highest payout, a participant typically participates in the competition with the most competitors. As mentioned in section 3.2 and following from prospect theory, participants tend to overweight their small chances of winning the most popular competitions, and consequently enter the more popular competitions with larger 'perceived' payouts. However, by participating in the competitions with the most players, the majority of participants actually lower both their expected payout and their expected increase in résumé value.

Instead of participating in the competition with the highest payout, competitors should participate in the competition that maximizes their total utility. More so, competitors can maximize their utility by participating in competitions with somewhat lower monetary payouts but substantially fewer overall participants. In sum, the majority of Kaggle participants tend to deviate from the obvious best strategy of maximizing their total utility by participating in the competitions with the highest payouts and consequently the highest numbers of participants.

5.2.3 Cascading Information and Replication

Each Kaggle competition has a discussion board where participants discuss their strategies, ask questions, and debate relevant data science topics. However, most of the time, mid-tier participants (roughly defined as those ranking between the 5th and 25th percentiles) reveal their specific strategy. For instance, two months into the Human Protein Atlas Image Classification Competition, which has a $37,000 payout, participant *lafoss* made a post titled "pretrained ResNet34 with RGBY (0.460 public LB)," in which he discusses his approach and provides most of the needed code in an accompanying iPython notebook. Although *lafoss*'s approach has only enabled him to achieve a ranking of 258/1411, his post has been upvoted by approximately 290 participants, which makes it the most upvoted post on the discussion board. Given *lafoss* outperforms 1153 other participants, those participants have an incentive to at least try, tweak, and then submit something similar to *lafoss*'s approach.

Therefore, in many Kaggle competitions information cascades occur. As new participants enter a competition, they have an incentive to try previous approaches (made available through the discussion board), which may improve their ranking and bolster their personal profile. However, the majority of these "replicated" submissions tend to be worse than the submission made by the (typically mid-tier) participant who initiated the information cascade. Thus, the "replicated" submissions add little or no value to the Kaggle platform, even though most participants, especially novices, have an incentive to submit them.

6 Version 1

6.1 Description of one seasonal gold competition

In order to more efficiently distribute talent across competitions, we propose establishing a "gold" competition in which only selected users can participate. There will be one gold competition per "season" (i.e. 3-month period), and at any given time the gold competition will have the largest purse of all open Kaggle competitions. We propose only 85 data scientists to participate in the gold competition. Our analysis in support of this value is included in Section **6.4**. Furthermore, based on past competition payout structures we believe only the top ten participants should receive prize money, with the first-place participant earning half of the pot, the second-place participant earning one quarter of the pot, and the remaining one quarter being split among places three through ten.

6.2 Single-item auction

In order to sponsor a gold competition, organizations (i.e. companies, government agencies, nonprofits, etc.) will place bids corresponding to the total purse they are willing to offer. It is up to Kaggle whether or not to enforce a specific payout structure for gold competitions, but we propose enforcing the structure detailed above.

Specifically, we propose implementing a Vickrey auction, which is an auction in which the highest bidder wins and pays a price equal to the second-highest bid. We choose the Vickrey auction format since it is truthful (i.e. bidding truthfully is a dominant strategy), individually rational (i.e. a truthful bidder is guaranteed nonnegative utility), and welfare maximizing (i.e. no other allocation has more total value to the bidders) [8].

6.3 Welfare Increase

The primary benefit of this version of our proposal is its contribution to the total welfare among Kaggle competition sponsors. In order to model the potential increase in welfare, we perform the following experiment

A. Treat the Two Sigma competition (i.e. the outstanding competition with the biggest prize) as the current gold competition.

B. Keep all but the top 85 participants with the highest Kaggle ranking in the Two Sigma competition, and distribute the remaining 1,038 participants among the other 11 open paying competitions.

C. Measure the increase in welfare after allocating the surplus players to the other competitions.

In order to perform the above, we need both an allocation mechanism and a concrete definition for change in welfare. For the allocation mechanism, we propose the following:

Let u be an arbitrary user in the group of 1,038 users leaving the Two Sigma competition. Assign u to competition c_i with probability p_i given by:

$$p_i = \frac{r_i}{\sum_{j \in \{1,\ldots,11\}} r_j} \tag{5}$$

Note that in the equation above r_i denotes the total payout of competition r_i. Intuitively, this probabilistic model accounts for the fact that players gravitate toward competitions with higher total payouts. We recognize that we could have formulated the reallocation of players from the Two Sigma competition to other competitions as a linear program in which we seek to maximize total welfare subject to appropriate constraints. However, our observations indicate that such a formulation would be inconsistent with Kaggle user behavior, as noted via the reference to Prospect Theory in Section **3.2**.

Next, in order to formalize the concept of change in welfare for an arbitrary competition, we simply use our definition of organizational welfare from Section **4.2** to arrive at the following equation:

$$\Delta W = \sum_{i=1}^{85} r'(i) - r(i) \tag{6}$$

Note that in the equation above, $r'(i)$ denotes the Kaggle score of the player with the ith highest Kaggle score in the competition after the allocation of new users to the competition, and $r(i)$ denotes the same but before the allocation.

6.4 Implementation

In order to test Version 1 of our model empirically, we implement a simulation of our proposal, the code for which can be found at the following GitHub repository: `https://github.com/SRS95/CS269I-Final-Project`.

6.4.1 Data Collection

In order to perform the simulation we first need Kaggle user and competition data. In order to collect the user data, we scrape the Kaggle website's public leaderboard, compiling a list of all users in the grandmaster, master, and expert categories.

We then use Kaggle's API to get the leaderboards for all 12 of the competitions we wish to include in our simulation (i.e. the Two Sigma competition along with the other 11 open paying competitions). Combining the user data we already obtained with the leaderboards data, we are able to create mappings from competitions to tuples of the form (username, (tier, points)), where tier is either grandmaster, master, or expert, and points corresponds to the user's Kaggle points.

6.4.2 Simulation

Using the data we've collected, our simulation algorithm proceeds as follows:

```
Data: Kaggle user and competition data
Result: Reallocates competitors no longer in gold competition and outputs resulting welfare gain
initialize userData, competitionData, averageGains;
for numSimulations do
    allocate surplus players;
    compute gains from reallocation for each competition ;
    update averageGains;
end
return averageGains
```

Algorithm 1: Simulation algorithm

Note that our simulation algorithm performs the reallocation procedure multiple times and averages across all trials. This is due to the stochastic nature of our reallocation algorithm, which we detail below:

```
Data: Competitors to reallocate, sorted mapping from each competition to its competitors before
      reallocation, payouts for each competition
Result: Sorted mapping from each competition to its competitors after reallocation
for playerToRealloc do
    Sample $n$ uniformly in range [1,...,$\lfloor numCompetitions/2 \rfloor + 1$];
    Randomly sample $n$ competitions proportionally to their payouts;
    Add playerToRealloc to each of the sampled competitions;
end
for updatedCompetition do
    Sort competitors from highest to lowest Kaggle score
end
return competitionsToCompetitors
```

Algorithm 2: Reallocation algorithm

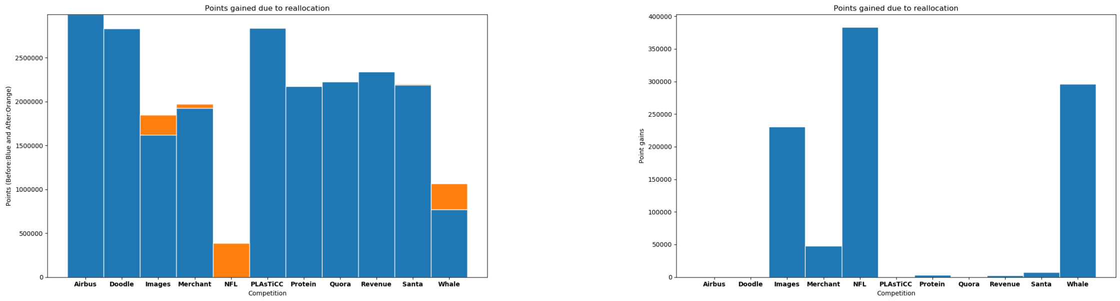
Note that the random sampling in the reallocation algorithm is performed according to the probability distribution detailed in Equation (5) in Section **6.3**. Also, note that we allow competitors to compete in multiple competitions since this is consistent with actual Kaggle user behavior. Finally, note that we return a mapping from competitions to competitors wherein each competition's competitors are sorted from highest to lowest Kaggle score. This is done so that we can easily compute the change in welfare before and after reallocation according to Equation (6) in Section **6.3** as follows:

```
Data: Sorted array of competitors in competition before reallocation, sorted array of competitors in
      competition after reallocation
Result: Change in welfare for given competition
Sum the Kaggle points of the top 85 competitors after reallocation;
Sum the Kaggle points of the top 85 competitors before reallocation;
return after - before
```

Algorithm 3: Change in welfare

Note that the above incorporates our assumption that only the top 85 competitors provide any benefit to a given competition.

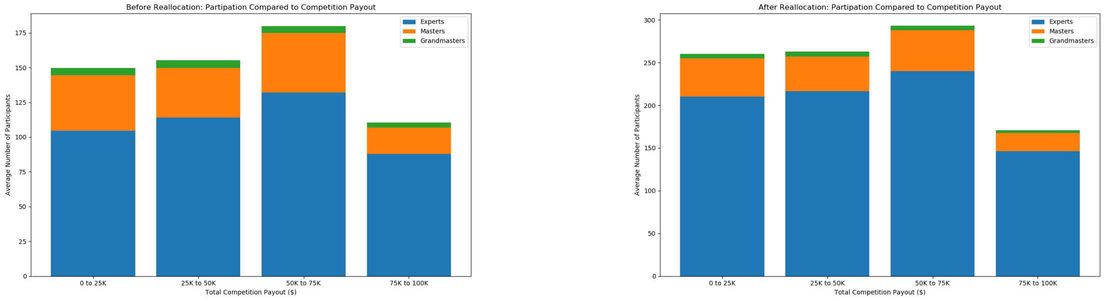
Running the simulation 100 times and averaging gains in welfare across all runs produces the following results:



(a) Point Gain: Before and After

(b) Raw Point Gains

Figure 3: (a) shows the increase in welfare provided to each of the 11 paying competitions we consider in our simulation relative to the welfare of each competition before reallocation. The total welfares of the competitions before reallocation are represented by the blue bars, and the increases in welfare are represented by the orange bars on top of the blue. On the other hand, (b) shows the raw increases in welfare after reallocation. Crucially, we note that our proposal never leads to a decrease in welfare, since in the worst case the top 85 competitors in a given competition are the same after reallocation as they were before.



(a) Payouts vs. Tiers: Before

(b) Payouts vs. Tiers: After

Figure 4: Shows the average number of participants with an expert, master, or grandmaster ranking across competitions with comparable payouts. (a) shows participation before reallocation, while (b) shows participation after reallocation. As expected, participation increases across all competitions and tiers, regardless of payout (note the change in the scale of the y-axis from (a) to (b)).

6.4.3 Optimal Gold Competition Size

Finally, in order to complete our model, we need to find the optimal number of competitors k as defined in our definition of organizational welfare in Section **4.2**. Recall that our model assumes that no benefit is provided to a competition by any competitors outside of the top k in the competition.

In order to find the optimal value for k, we run our simulation many times using different values for k and plot the average gain across all competitions in the simulation vs. k.

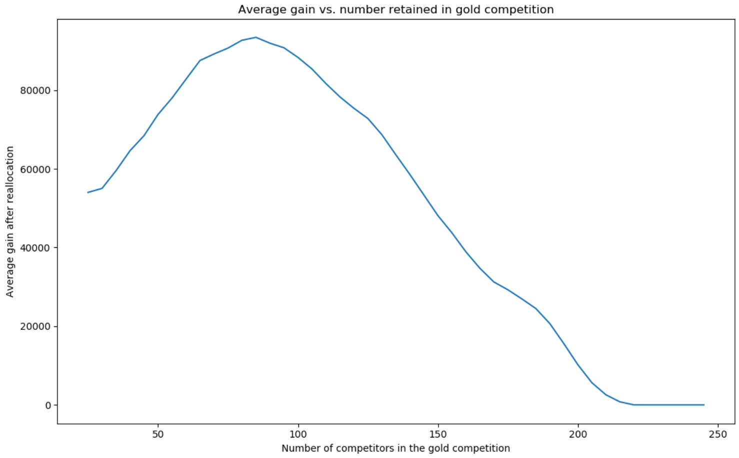


Figure 5: From this plot we deduce that the optimal value for k is 85 since that is the point at which the average gain across competitions is the highest. Since we have access to user data on all Kaggle grandmasters, masters, and experts and averaged across 11 competitions (i.e. the "normal" competitions discussed above), we believe that our empirical justification for $k = 85$ is well-founded.

7 Version 2

7.1 Description of multiple, seasonal, gold competitions

Version 2 simply extends Version 1 by allowing for there to be multiple "gold" competitions. These competitions will still be exclusive, and so as to avoid the same users being placed into all of the gold competitions, we advocate for a system in which a user is only allowed to participate in one gold competition at a time. Furthermore, we believe it would be best to structure the system so that all gold competitions begin and end at the same time, each lasting for the typical 3-month seasonal period we proposed in Version 1.

7.2 Multiple-item auction

As before, in order to sponsor a gold competition, organizations will place bids corresponding to the total purse they are willing to offer. It is up to Kaggle whether or not to enforce a specific payout structure for gold competitions, but we propose enforcing the structure detailed in Version 1.

However, it is crucial to note that we must model this process as a multiple-item auction. Formally, we let g_i denote the ith gold competition, and we enforce the following: $r(g_i) \geq r(g_j)$, $\forall i \leq j$ where $r(g_i)$ is the total payout of gold competition i.

We propose implementing a Vickrey-Clarke-Groves auction that proceeds as follows:

A. Accept a bid from each organization for each outcome (i.e. each assignment of slots). Crucially, assume that a bid is 0 for any outcome in which the bidder does not receive a slot.

B. Choose the allocation of slots that maximizes the total reported social welfare. Namely, maximize:

$$\sum_{i=1}^{n} b_i(\omega) \tag{7}$$

over all ω in Ω. Note that ω denotes a particular assignment of slots, n is the number of bidders, and $b_i(\omega)$ is the bid from organization i for outcome ω

C. Finally, charge each bidder i her externality p_i, meaning the welfare loss caused to the other bidders by i's presence:

$$p_i = (\max_{\omega \in \Omega} \sum_{j \neq i} b_j(\omega) - \sum_{j \neq i} b_j(\omega^*)) \tag{8}$$

Note that ω^* is the oucome chosen in the second step.

We note this auction format is desirable since the VCG mechanism is truthful and individually rational [9].

Finally, once the spots have been allocated to organizations via the VCG auction mechanism, we propose applying the well-know Deferred-Acceptance Algorithm to match data scientists with gold competitions. Specifically, data scientists would submit rankings over each of the n gold competitions, and each of the competitions would submit rankings over k data scientists. These preference lists would then be used as inputs to the Deferred-Acceptance Algorithm, which would then perform the assignment.

Finally, we recognize that more work would have to be done to determine the optimal number of competitors in each of the gold competitions. Our assumption is that the number of competitors in each competition should be equal, but that would have to be empirically or theoretically verified. Nonetheless, minor modifications to our implementation of Version 1 along with an out-of-the-box implementation of the Deferred-Acceptance Algorithm and the VCG mechanism could be used to accomplish what we have described in this section.

8 Version 3

Now that we have demonstrated the value of separating the top players into a league of their own, we might consider the effects of bucketing the rest of the players by skill level.

8.1 Skill-based Match Making

Chen, Xue, Kolen and other collaborators from Electronic Arts, the video game company, established the benefit of skill based games in their paper “An Engagement Optimized Matchmaking Framework” [5].

In EA’s context, matchmaking refers to the process of pairing players in online games. While Kaggle is not structured as “player versus player” in the traditional sense, it could be thought of as an online game that currently hosts open lobbies (since players freely assign themselves to whichever competition they like).

8.2 Churn Rate

By applying the successful methods seen in online gaming, Kaggle can effectively reduce it’s overall churn rate and increase the engagement on the platform. By increasing platform engagement, Kaggle successfully increases its own welfare and in turn passes that gain on to companies and users.

We can approximate Kaggle’s churn rate by looking at the number of Novice users versus others. To achieve Contributor status, a user only needs to finish their profile, participate in one competition, and leave a comment. 47% of users are Novice’s, even accounting for the fact that some of those are new users who will convert, which means Kaggle currently has a high churn rate.

8.3 Applying EOMM to Kaggle

The first and unsurprising result from EOMM is that players who are introduced to the platform with a series of continuous losses leave the most, approximately 5.1% of the time. Alarmingly, this is likely the most common outcome for any new Kaggle participant, as data science problems on the site remain non trivial and the competition is stiff.

The second outcome conducive to user churn is for a player to start with a series of wins followed by a loss (4.9% churn). This is unlikely with Kaggle's current structure, but it is something we avoid in our new set up. What players want, and what we hope to model through matchmaking, is the feeling of steady improvement, as reflected in performance against other players.

Since Kaggle already maintains an internal ranking of players based on skill level, we can approximate each player's predicted outcome based upon skill level versus the collective skill of opponents. Using this knowledge of a player's current play state (her record in previous matches) and our prediction of churn rate based upon a series of outcomes, we simply assign players to reduce the overall sum of expected churn using:

$$\eta^* = \arg\min_{\eta} \sum_{m \in \eta} \sum_{p=1}^{n} c_m(p) \tag{9}$$

Where η is the sum of the churn rate for a given set of matches, m is a match in η, p is all of the players in a match, and $c(p)$ reflects the predicted churn for player p in a match m.

8.4 Caveat

Unlike EA's online games, Kaggle currently only has 16 open competitions and its competitions last months at a time. Moreover, the number of players actively signing in at any given time and looking to join a new match is orders of magnitude smaller than popular online games. This may create a liquidity issue, making it difficult for us to have the necessary flexibility required to see the benefits that EOMM.

Another potential issue is Kaggle's understanding of player skill level. In online gaming, the unique controls and layout of each game means that new players are likely to struggle in the beginning, even if they have significant experience in other games. In data science, university training or professional experience can typically be easily applied to a new data set by a first time Kaggle participant. As a result, it is more difficult to reflect the true skill value of a first time player. If our understanding of player skill level breaks down, we cannot effectively predict match outcomes, which means we cannot reduce overall churn rate.

9 Conclusion

Currently, we live in a world where we have more data-related problems to solve than we have data scientists available to solve them. Fortunately, Kaggle has successfully created a crowd-sourced solution that provides companies with a new means of gaining insights from their extensive troves of data, while simultaneously creating a platform for novice and expert data scientists alike to hone their skills.

Despite its success, Kaggle has several major shortcomings, which lead to suboptimal outcomes for companies hosting competitions on their platform.

After analyzing Kaggle's shortcomings from a game-theoretic standpoint, we decided to hone in on Kaggle's lack of Pareto optimality. Specifically, in section 6 we proposed an improved version of Kaggle that is identical to the current platform with the exception of a single "gold" competition.

Our simulated results of Version 1 of our proposal were promising. Nonetheless, recognizing the need for a system flexible enough to support multiple gold competitions as Kaggle continues to grow, we proposed an enhanced Version 2 addressing these considerations in Section 7.

Lastly, we considered a complete Kaggle overhaul in which we centralize all of the player matchmaking. However, we ultimately found that Kaggle does not have the volume of competitions to support such a setup and, as such would lose some of its appeal.

In conclusion, we propose Kaggle implement Version 1 of our proposal, and ultimately Version 2, contingent on a successful adoption of Version 1. We recognize the potential ramifications of our proposal on user sentiment, but we believe that a gradual incorporation of our proposal into the platform would be palatable and beneficial for all parties.

References

[1] Easley, David, and Arpita Ghosh. *Behavioral Mechanism Design.* ACM SIGecom Exchanges, vol. 14, no. 1, Dec. 2015, pp. 8994., doi:10.1145/2845926.2845932.

[2] Tversky, Amos, and Daniel Kahneman. *Advances in Prospect Theory: Cumulative Representation of Uncertainty.* Readings in Formal Epistemology, 2016, pp. 493519., doi:10.1007/978-3-319-20451-2-24.

[3] Yan, Wangcheng, et al. *Crowdsourcing Data Science for Innovation.* 2017 IEEE International Conference on Data Mining Workshops (ICDMW), 2017, doi:10.1109/icdmw.2017.164.

[4] Dipalantino, Dominic, and Milan Vojnovic. Crowdsourcing and All-Pay Auctions. Proceedings of the Tenth ACM Conference on Electronic Commerce - EC 09, 2009, doi:10.1145/1566374.1566392.

[5] Chen, Zhengxing, Xue, Su, Kolen, John, Aghdaie, Navid, Zaman, Kazi, Sun, Yizhou, and Magy El-Nasr. *EOMM: An Engagement Optimized Matchmaking Framework*, 2017, arXiv:1702.06820v1

[6] Ahmed Abdelmeged, Ruiyang Xu, Raghavendra Gali, Karl Lieberherr, 2015. *Theory of Side-Choosing Games to Create and Disseminate Knowledge in Formal Sciences.* ACM X, X, Article X (February 2015), 19 pages.

[7] Abdelmeged, A. 2014. Organizing *Computational Problem Solving Communities via Collusion-Resistant Semantic Game Tournaments.* Ph.D. Dissertation, Northeastern University, Boston, MA USA. Advisor-Karl Lieberherr.

[8] Roughgarden, T. 2016. ”Lecture 13: Introduction to Auctions.” CS 269I: Incentives in Computer Science. Stanford Universiy, Stanford, CA USA.

[9] Roughgarden, T. 2016. ”Lecture 15: The VCG Mechanism” CS 269I: Incentives in Computer Science. Stanford Universiy, Stanford, CA USA.