Races or Tournaments? [PRELIMINARY AND INCOMPLETE]*

Andrea Blasco Kevin J. Boudreau Karim R. Lakhani
Michael Menietti

Last updated: 18 May, 2017

Abstract

We examine the performance of two different choices of contest design: the race (where the winner is the first to achieve a minimum quality) and the tournament (where the winner is the one with the highest quality in a given period). After characterizing the optimal design, we report results of a field experiment conducted to compare the performance of three alternatives motivated by theory: the race, the tournament, and the tournament with a minimum quality requirement. Outcomes in a race are of comparable quality, supplied faster, and with lower participation rates. Based on these findings, we show the optimal design under several counterfactual situations.

JEL Classification: xxx; xxx; xxx.

Keywords: xxxx; xxxx xxxx.

*Blasco: Harvard Institute for Quantitative Social Science, Harvard University, 1737 Cambridge Street, Cambridge, MA 02138 (email: ablasco@fas.harvard.edu).

```
## Loading required package: knitr

## Loading required package: magrittr

## Loading required package: xtable

## Loading required package: stargazer

##
## Please cite as:

##  Hlavac, Marek (2015). stargazer: Well-Formatted Regression and Summary S

##  R package version 5.2. http://CRAN.R-project.org/package=stargazer
```

1 Introduction

Government agencies, private companies, and many other organizations often sponsor prize-based competitions (contests) to engage workers and other participants in a given task, including solving a technical problem, achieving an innovation, xxxx. A typical goal for these organizations is to design a competition that maximizes competitors efforts while minimizing the time to complete the task. Balancing between these two desirable but often incompatible goals can be very difficult. First, when contest designers lack adequate knowledge of costs and skills of participants, design choices are made under considerable uncertainty. Second, competitors do XXXX, YYY, ZZZ.

In this article, we investigate this trade-off betwen time and output quality byb comparing two different competition formats: the "race," where the first to finish an innovation project wins, and the "tournament," where the best finished project wins. While most of past literatured focused on xx and yy, there is no comparison of xxx and xxx.

Races and tournaments are widespread — and for good reasons. However, poeple do not fully understand the trade-off between one and the other. To fix ideas, imagine a government willing to design an innovation contest aimed at finding solutions to a problem of public health, such as antibiotic resistance.[1] To minimize the risk that the threat of xxxx will materialize before a solution is found, one may choose a tournament competition format with a tight deadline for participants to provide their solutions. The problem is to find the right duration. When the duration of the competition is too short, incentives maybe insufficents for competitors to exert enough effort resulting in inadequate solutions. Alternatively, the government can set up a race competition with a prize being awarded to the first competitor who achieves, or goes beyond, a minimum quality

[1] This example is taken...

threshold. Here the problem of accelarating the timing of innovation should not be a big issue but competitors may work inefficiently, as they have no incentives to exceed the minimum threshold. Fixed the prize structure, both approaches have specific advantages and limitations. However, xxxx.

We proceed in two ways. First, we develop a contest model that encompasses both the race and the tournament in a single framework. Exploring the duality of the model, we compare equilibrium behaviors under both competitive formats and characterize the optimal choice for the contest designer. Then, we design and execute an experiment to test the implications of the theory in the field, and xxxx providing policy recommendations.

Our theoritical approach extends the contest model introduced by Moldovanu and Sela (2001) to a situation in which xxx decide both time and quality. Thus, contests have an all-pay structure by which participants pay an immediate cost for an uncertain future reward. The decision of timing and quality is made under the uncertainty of the costs of the rivals. The contest designer wants to maximize reveues and has preferences for both time and quality. Following the analysis of the model, we show that the optimal design depends on the number of participants and the concavity of their cost function. We also show that XXX, YYY, and ZZZZ.

The context of the field experiment was an online programming competition run on Topcoder at the end of 2016. In a typical programming competition, participants compete writing source code that solves a given problem for winning a monetary prize. We worked together with researchers from the United States National Health Institute (NIH) and the Scripps Research Institute (SCRIPPS) to select a challenging problem for the contest. The selected problem was based on an algorithm called BANNER built by NIH (Leaman, Gonzalez, and others 2008) that uses expert labeling to annotate abstracts from a prominent life sciences and biomedical search engine, PubMed, so disease characteristics can be more easily identified. The goal of the programming competition was to improve upon the current NIH's system by using a combination of expert and non-expert labeling, as described by Good et al. (2014). The competition was hosted online on the platform Topcoder.com (about 1M registered users in 2016). Top submissions were awarded monetary prizes ranging between $100 to $5000 for a total prize pool of more than $40,000.

Our intervention consisted in sorting at random participants into independent virtual rooms of 10 or 15 people. These virtual rooms were then randomly assigned to one of three different competitive settings: a race, a tournament, and a tournament with a reserve score, which is the lowest acceptable score by the platform for a submission to be awarded a prize.

We find that xxxxx [participation in the tournament is xxx compared to the race the reserve.]

We also find that xxxx [submission are quicker in a race, whereas are equally distributed at the end of the competition in the the tournament and in the tournament with quality requirement.]

Another interesting finding is that xxxxx [No evidence trade-off between a race and a tournament in terms of higher scores vs faster submissions. We do find that scores are higher in the tournament but we do not find a strong trade-off in the sense that race had comparable good quality solutions than the tournament.]

2 Literature

This paper is related to the contest theory literature Dixit (1987) Baye and Hoppe (2003), Parreiras and Rubinchik (2010), Moldovanu and Sela (2001), Moldovanu and Sela (2006), Siegel (2009), Siegel (2014). It also relates to the literature on innovation contests Taylor (1995), Che and Gale (2003). And the personnel economics approach to contests Lazear and Rosen (1981), Green and Stokey (1983), Mary, Viscusi, and Zeckhauser (1984).

Empirically, Dechenaux, Kovenock, and Sheremeta (2014) provide a comprehensive summary of the experimental literature on contests and tourments. Large body of empirical works have focused on sports contests Szymanski (2003). More recently, inside firms (xxx) and online contest (xxxx).

This paper is also related to the econometrics of auctions Paarsch (1992), Laffont, Ossard, and Vuong (1995), Donald and Paarsch (1996) and more recently Athey, Levin, and Seira (2011), Athey and Haile (2002), and Athey and Haile (2007).

An extensive literature has discussed the reasons why contests are sometimes preferred to other forms of incentives (e.g., individual contracts). Typically, contests reduce monitoring costs [xxx], incentivize production with common risks [xxx], and deal with indivisible rewards [xxxx], among others. While there is not much debate on why contests should be used, the issue of how to effectively design and deploy a contest still attracts much research.

Several aspects of contest design have been investigated, including the optimal prize structure [XXX, xxxx, xxxx], number of competitors [XXX, XXX], and imposing restrictions to competition such as minimum effort requirements [XXX, XXX]. Also, a great deal of theoretical models of races and tournaments have been developed and applied to a wide range of economic situations including patent races [xxx], arms races [xxx], sports [xxx], the mechanism of promotions inside firms [xxxx], sales tournaments [xxxx], etc.

Harris and Vickers (1987), Grossman and Shapiro (1987) investigate the dynamics issues patent races where the interest is how firms compete for a patent. Bimpikis, Ehsani, and Mostagir (2014) looks at the problem of how to design an information structure that is optimal when the contest is a race and innovation is uncertain (encouragement and competition effect). In the laboratory, Zizzo (2002) finds poor support to predictions of dynamic xxxx. In general we do not know much about the dynamic aspect of contests.

The duality. As pointed out by Baye and Hoppe (2003), many of these models of tournament and race competitions are specific cases of a more general "contest games." And sometimes it is possible to design one or the other in a way to exploit a "duality." In other words, in theory, a competition can be designed as a tournament to do xxx or as a race to do xxx. While theoretically very useful, how to exploit this duality in practice remains largely unknown. Lack of data. As before, xxxx. The main challenge is self-selection. The answer to this optimal design question relates to the cost function of agents with respect to "time" and to "effort." It is hard to say which solution is better. However, it is easier to tell whether you should have one prize or multiple prizes.

3 The model

We now generalize the contest game introduced by Moldovanu and Sela (2001) to a situation where players simultaneously decide $i)$ the quality and $ii)$ how fast to produce a given output. Then we explore the problem of revenue maximization faced by a contest designer with preferences for both quality and time.

3.1 Basic setup

A generalized contest game is an n-player game with asymmetric information in which each player $(i = 1, ..., n)$ competes against the others to win a prize. Players assign a nonnegative value v_k to each prize $(k = 1, ..., q)$ that can be ordered as $v_1 \geq v_2 \geq ... \geq v_q$. A player strategy consists in choosing an output quality y_i and a timing t_i both nonnegative numbers that determine the probability $p_k(y_i, t_i)$ of winning a prize k.

Each player incurs production costs as determined by a cost function $C(\cdot)$ which is increasing in quality and decreasing in timing. The cost function also depends on an individual ability parameter a_i that is private information of each player and is drawn at random from a common distribution F on a bounded interval $[\underline{a}, \bar{a}]$ with $\underline{a} > 0$.[2] For simplicity, we assume the cost function is multiplicative:

$$C(a_i, y_i, t_i) = c_a(a_i)c_y(y_i)c_t(t_i) \tag{1}$$

with $c_a(\cdot)$ and $c_t(\cdot)$ being monotonic decreasing functions (the higher the ability or the time to complete, the lower the cost) and $c_y(\cdot)$ being a monotonic increasing function (the higher the quality, the higher the cost). We impose additional conditions to ensure nonnegative costs

$$c_a(\bar{a}) > 0, c_t(\bar{t}) > 0, \text{ and } c_y(\underline{y}) > 0.$$

[2] This assumption rules out common parametric distributions like the log-normal and forces us to focus the analysis on beta-type distributions. However, results do not hinge on this particular assumption.

Player i's payoff is then

$$\pi_i = \sum_{k=1}^{q} p_k(y_i, t_i)v_k - C(a_i, y_i, t_i). \tag{2}$$

Using the above notation, we denote a (generalized) contest game G with

$$G \equiv \left\{n, \{v_k\}_{k=1}^{q}, F, C, \{p_k\}_{k=1}^{q}\right\}. \tag{3}$$

When players have to meet a time deadline and/or satisfy a minimum quality level to be eligible to win a prize, we say that a contest game G has a "minimum-entry requirement." That is, a contest G has a minimum-entry requirement if, whenever player i's quality is below a target level $q_i < \underline{y}$ or the completion time is above a given deadline $t_i > \bar{t}$, then player i's probability of winning a prize k is zero $p_k(\cdot, \cdot) = 0$. To simplify exposition later, we use the convention that whenever a deadline requirement is not met then the output quality is zero, and whenever the required quality is not met then the completion time is infinity $t_i = \infty$. This simply ensures that players who did not meet any minimum requirement are always ranked last in the contest.

After having introduced a general contest game, we now define tournaments and races as particular cases of it using the following notation: let $y_{(1:n-1)}, \ldots, y_{(n-1:n-1)}$ and $t_{(1:n-1)}, \ldots, t_{(n-1:n-1)}$ denote the order statistics of the y's and the t 's for the $n-1$ opponents of player i. A tournament has the following characteristics.

Definition 1 (Tournament)**.** A tournament is a contest game G with a minimum-entry requirement $\bar{t} > 0$ and $\underline{y} = 0$ where player i's probability of winning a prize is

$$p_k(y_i, t_i) = \begin{cases} \Pr(y_i > y_{n-1:n-1}) & \text{if } k = 1 \\ \Pr(y_{n-k+1:n-1} > y_i > y_{n-k:n-1}) & \text{if } k > 1. \end{cases} \tag{4}$$

when $t_i \leq \bar{t}$, and it is zero otherwise.

In other words, the tournament is a special case of a contest game with minimum-entry requirement in which players have a deadline to meet and the player having achieved the highest output quality within the deadline gets the first prize, the player having achieved the second highest output quality gets the second prize, and so on.

In a similar way, we define a race competition.

Definition 2 (Race)**.** A race is a contest game G with a minimum-entry requirement $\underline{y} > 0$ and

$\bar{t} = \infty$ where player i's probability of winning a prize is

$$p_k(y_i, t_i) = \begin{cases} \Pr(t_i < t_{1:n-1}) & \text{if } k = 1 \\ \Pr(t_{1-k:n-1} > t_i > t_{k:n-1}) & \text{if } k > 1. \end{cases} \tag{5}$$

when $y_i < \underline{y}$ and it is zero otherwise.

That is, the race corresponds to a special case where the player being the first to complete a job with a minimum quality gets the first prize, the player being the second to complete a job with a minimum quality gets the second prize, and so on.

3.2 Equilibrium

In this section, we solve the model for the XXX XXX equilibrium of players. We assume throughout that there are only two prizes of total value normalized to one, where the fraction $\alpha \geq 1/2$ goes to the first placed competitor and $1 - \alpha$ goes to the second placed competitor. We also let $F_{r:n}$ and $f_{r:n}$ be the distribution and density function of the r^{th} order statistic (i.e., the r^{th} lowest realization) of n draws from the ability distribution (i.e., the a's).

3.2.1 Equilibrium in a tournament

At equilibrium, each player i chooses y_i and t_i by maximizing $E[U]\ldots$

The key observation is that the probability of winning a prize in a tournament is affected by the completion time only when deadline is missed (it is zero in that case), otherwise the probability is a constant with respect ot completion time. As a result, picking $t_i = \bar{t}$ is a (weakly) dominant strategy for each player. This also means that, from the point of view of the contest designer, imposing a more distant deadline has the same effect as a reduction in the marginal costs for every participant competitor. [This should go later] Then, the equilibrium y_i must xxx a monotonic function $y^*()$ with an inverse $\phi()$ which satisfies the first-order differential equation

$$\begin{aligned} 0 = &\alpha f_{(1:N-1)}(\phi)\phi' + (1-\alpha)\phi'\{[1 - F_{(1:N-1)}(\phi)] f_{(1:N-2)}(\phi) + \\ &+ f_{(1:N-1)}(\phi) F_{(1:N-2)}(\phi)\} - c_a(a) c_y(\underline{y}) c_t'(t_i) \end{aligned} \tag{6}$$

subject to the boundary condition $\phi(0) = \underline{a}$ (i.e., the lowest-ability competitor's optimal output quality is zero).

As shown by Moldovanu and Sela (2001), the solution is

$$y^*(a_i) = c_y^{-1}\left[c_y(\underline{y}) + \frac{1}{c_t(\bar{t})}\left(\alpha\int_{a_i}^{\bar{a}} A(z)dz + (1-\alpha)\int_{a_i}^{\bar{a}} B(z)dz\right)\right] \tag{7}$$

where

$$A(x) = \frac{1}{c_a(x)} f_{(n-1:n-1)}(x) \tag{8}$$

and

$$B(x) = \frac{1}{c_a(x)}\left\{\left[1 - F_{(n-1:n-1)}(x)\right] f_{(n-1:n-2)}(x) + f_{(n-1:n-1)}(x)F_{(n-1:n-2)}(x)\right\}. \tag{9}$$

Monotonicity of the equilibrium output quality implies that, for every $i = 1, ..., n$, the equilibrium expected payoff from the contest π_i^* depends on the rank of the player's ability relative to the others. As a result, the equilibrium expected payoff net of costs is

$$R(a_i) = \alpha F_{n:n}(a_i) + (1-\alpha)[1 - F_{n:n}(a_i)]F_{n-1:n-1}(a_i). \tag{10}$$

3.2.2 Equilibrium in a race

In a similar way, one can derive the equilibrium strategy in a race. Again the key observation is that any quality below the target gives a zero probability of winning and any quality above the target gives a constant probability of winning. Thus, player i's choice of optimal quality y^* is either zero (with $t_i = \bar{t}$ by convention) or $y^* = \underline{y}$.

Then, the equilibrium xxx for player i is

$$t^*(a_i) = c_t^{-1}\left[c_t(\bar{t}) + \frac{1}{c_y(\underline{y})}\left(\alpha\int_{a_i}^{\bar{a}} A'(z)dz + (1-\alpha)\int_{a_i}^{\bar{a}} B'(z)dz\right)\right] \tag{11}$$

where

$$A(x) = \frac{1}{c_a(x)} f_{(n-1:n-1)}(x) \tag{12}$$

and

$$B(x) = \frac{1}{c_a(x)}\left\{\left[1 - F_{(n-1:n-1)}(x)\right] f_{(n-1:n-2)}(x) + f_{(n-1:n-1)}(x)F_{(n-1:n-2)}(x)\right\}. \tag{13}$$

An important property of XX is that $y^*(a_i)$ has its upper bound in XX and lower bound in XX. Again payoffs are xxxx. Hence, by setting to zero and solving for the ability, gives the marginal ability $\underline{a}$ as

$$\underline{a} = h(n, V, F_A, C, d). \tag{14}$$

3.2.3 Tournament vs races

By comparing equilibrium xxx and xxx, we find that the race and the tournament do not (ex-post) dominate one another with respect to output quality. Whereas the race always dominates the tournament with respect to completion time. [This is only when the deadline is the same. Otherwise, there's always xxxx.] This result is stated below.

Proposition 1. *There always exist an interval of abilities where the output quality is higher in the race than in the tournament. By contrast, every player takes less completion time in the race than in the tournament.*

Proof. Marginal type has utility zero in a race but the same type has a strictly positive utility in the tournament. Since probability of winning is not different in the race or the tournament (the bid is a monotonic transformation of the individual ability or, in other words, rankings are virtually the same), expected payoffs in equilibrium differ only in the cost functions. Hence, to be an equilibrium, the player in the tournament should bid less than the player in the race to earn a strictly positive expected payoff. □

Let's make an example.

```
p <- plnorm   # pdf individual abilities
r <- rlnorm   # Simulate individual abilities
cy <- function(x) x^2 # Cost function performance
ct <- function(x) 2*exp(1-x)  # Cost function timing
```

FIGURE 1. Equilibrium bids in a race and a tournament.

Implications. The above proposition applies only if the target is higher in a race than in a tournament. But what if the two competitions had the same target ? In that case, tournaments and races have the same marginal type. Therefore, the performance of players in the tournament with reserve are always non-lower than those in the race. This does not imply that it is optimal to set the target. On the contrary, we will show that it is optimal to set an optimal target in a tournament that is below the optimal target in a race. Next section.

3.3 The contest designer's problem

Let us now focus on the contest designer's problem. Imagine the contest designer can choose the competition format to be either the race or the tournament. Imagine all other aspects of design are given. The prize structure α has been already chosen. There is a deadline $\bar{t}$, which is the same in both competition formats. [The quality requirement $\underline{y}_c$ in the tournament is smalle than that in the race $\underline{y}_{\text{race}} > \underline{y}_{\text{tour}}$)] We will relax this assumption later to consider a more general setting where these variables are also part of the contest designer's problem.

The contest designer has an objective function that is increasing in the expected quality of the winning solution and decreasing in the corresponding time to completion. Here, to do not complicate exposition, we assume that the contest designer cares about the winning submission only: second placed efforts are not considered. [If the principal values the diversity of the solutions ... but we assume it does not.]

XXX EQUATION XXXX

The optimal choice involves a comparison of the expected profits between the race and the tournament. Given xxxx, we can show that there will be a threshold on the cost of completion time $\hat{\tau}$ above which the race is a better choice than the tournament, and vice versa.

Proposition 2. *There's a tau above which ...*

Proof. In a tournament, the objective function is

$$\begin{aligned} R_{\text{tour}} &= \Pr(t_{(1:n)} \leq \bar{t}) \left\{ \int y^*(x \mid t_{(1:n)} \leq \bar{t}) dF_{n:n}(x) - \tau\bar{t} - 1 \right\} \\ &= \int_{\hat{a}}^{\bar{a}} y^*(x) dF_{n:n}(x) - \tau\bar{t} - 1. \end{aligned} \tag{15}$$

That is, the contest designer's objective function is the sum of the expected output quality for a given deadline, minus the cost τ of having the winner working on the task until completion (i.e., until the deadline), and the cost of the prize pool (recall the prize pool is normalized to one).

[Implicitly, you're assuming that the prize is always large enough to ensure positive effort.] [Second prize too is stochastic!!!!]

In a race, the objective function is

$$\begin{aligned} R_{\text{race}} &= \Pr(a_{(N)} \geq \hat{a}) \left\{ \underline{y} - \alpha - \Pr(a_{(N-1)} \geq \hat{a})(1-\alpha) \right\} - \tau \int_{\hat{a}}^{\infty} t^*(x) dF_{N:N}(x) \\ &= [1 - F_{N:N}(\hat{a})] \left\{ \underline{y} - \alpha - [1 - F_{N-1:N}(\hat{a})](1-\alpha) \right\} - \tau \int_{\hat{a}}^{\infty} t^*(x) dF_{N:N}(x). \end{aligned} \tag{16}$$

Note. $t^*(x) \leq \bar{t}$ for all x's. Thus, a lower bound for the above objective function can be computed:

$$\underline{R_{\text{race}}} = [1 - F_{N:N}(\hat{a})] \left\{ \underline{y} - \alpha - [1 - F_{N-1:N}(\hat{a})](1-\alpha) - \tau\bar{t} \right\} \tag{17}$$

An even simpler lower bound is rewriting the above expression as if $\alpha = 1$ (note if the real alpha was set 1 then also mtype would change and therefore setting alpha hits a lower bound only when mtype does xxxx when alpha is 1).

Note. $y^*(x)$ is lower than $\underline{y}$ for all $a < \hat{a}$. Thus, a lower bound of the tournament's expression is

$$\overline{R_{\text{tour}}} = [1 - F_{N:N}(\hat{a})]\underline{y} + \int_{\hat{a}}^{\infty} y^*(x) dF_{N:N}(x) - \tau\bar{t} - 1. \tag{18}$$

$$\begin{aligned}
\underline{R_{\text{race}}} &\geq \overline{R_{\text{tour}}} \\
[1 - F_{N:N}(\hat{a})](\underline{y} - 1 - \tau\bar{t}) &\geq [1 - F_{N:N}(\hat{a})]\underline{y} + \int_{\hat{a}}^{\infty} y^*(x) dF_{N:N}(x) - \tau\bar{t} - 1 \\
-[1 - F_{N:N}(\hat{a})](\tau\bar{t} + 1) &\geq \int_{\hat{a}}^{\infty} y^*(x) dF_{N:N}(x) - (\tau\bar{t} + 1) \\
F_{N:N}(\hat{a})(\tau\bar{t} + 1) &\geq \int_{\hat{a}}^{\infty} y^*(x) dF_{N:N}(x) \\
\tau &\geq \left[\frac{\int_{\hat{a}}^{\infty} y^*(x) dF_{N:N}(x)}{F_{N:N}(\hat{a})} - 1 \right] \frac{1}{\bar{t}}
\end{aligned} \tag{19}$$

End proof.

When the cost of time τ is sufficiently high, the race is preferred. Interestingly, the threshold is a function of the deadline to complete the job, as xxx. It also depends on the shape of xxxx.

3.3.1 Optimal minimum-entry

Now we turn to discuss the contest designer's choice of an optimal minimum requirement $\underline{y}$. So far, we have assumed that $\underline{y}_{\text{race}} > \underline{y}_{\text{tour}}$. Now, we show that the assumption that xxxx is indeed an optimal choice of the contest designer. This is summarized in the next proposition.

Proposition 3. *Suppose the contest designer can choose the target that max profits in both the race and the tournament. Then, the optimal $\underline{y}$ in tournament is generally lower than that in a race.*

To prove that it is indeed the case. We proceed in two steps. First, we assume that the contest designer does not care about minimizing the timing of the innovation by imposing $\tau = 0$. For simplicity, assume that $\alpha = 1$ (winner-takes-all). In a race, this means that the optimal target will be a value that makes equal the costs in terms of less participation versus the gains in terms of higher values of the winning solutions. Formally, the contest designer's problem in a race is

$$\text{maximize } R^{\text{race}} = [1 - F_{N:N}(\hat{a})](\underline{y}_{\text{race}} - 1). \tag{20}$$

Note that $\hat{a}$ depends on the target. This is clearly concave in $\underline{y}_{\text{race}}$. Thus, the first order condition is also sufficient.

$$\text{FOC } \Rightarrow -F'_{N:N}(\hat{a})\hat{a}'(\underline{y}_{\text{race}} - 1) + [1 - F_{N:N}(\hat{a})] = 0. \tag{21}$$

In a tournament, ...

$$\text{maximize } R^{\text{race}} = \int_{\hat{a}}^{\infty} y^*(x, \underline{y}) dF_{N:N}(x) - [1 - F_{N:N}(\hat{a})]. \tag{22}$$

Convexity is not sure. If not, then the optimal target is zero. Which is lower than the optimal target in a race.

Instead. If the objective function is (strictly) concave then there's an internal solution.

$$\begin{aligned}
\text{FOC } \Rightarrow & \frac{d \int_{\hat{a}}^{\infty} y^*(x, \underline{y}) dF_{N:N}(x))}{d\underline{y}} + F'_{N:N}(\hat{a})\hat{a}' = 0 \\
& \text{(by using Leibniz rule)} \\
\Rightarrow & - y^*(\hat{a}, \underline{y})\hat{a}' F'_{N:N}(\hat{a}) + \int_{\hat{a}}^{\infty} \frac{\partial y^*(x, \underline{y})}{\partial \underline{y}} dF_{N:N}(x) - F'_{N:N}(\hat{a})\hat{a}' = 0 \\
\Rightarrow & - \underline{y}\hat{a}' F'_{N:N}(\hat{a}) + \int_{\hat{a}}^{\infty} \frac{\partial y^*(x, \underline{y})}{\partial \underline{y}} dF_{N:N}(x) - F'_{N:N}(\hat{a})\hat{a}' = 0.
\end{aligned} \tag{23}$$

Using (21) with (23), the optimal target is the same in the race and the tournament only if

$$\int_{\hat{a}}^{\infty} \frac{\partial y^*(x, \underline{y})}{\partial \underline{y}} dF_{N:N}(x) = [1 - F_{N:N}(\hat{a})]. \tag{24}$$

$$\frac{\partial y^*(x,\underline{y})}{\partial \underline{y}} = \frac{c_y'(\underline{y})}{c_y'(y^*(x,\underline{y}))}.$$

Then.

- If $c_y(\cdot)$ is linear, we have that the ratio is one for all x.
- If $c_y(\cdot)$ is convex, then we have that it is less than one. If
- If $c_y(\cdot)$ is concave, then we have that it is higher than one.

As a result, if linear or convex the first order condition is lower than that in the race. Since the obj. function is concave (second order is decreasing), the target should be lower in a tournament than in a race to satisfy the first order condition. (a lower target increases the focs.).

Conjecture. If $\tau > 0$, the $\underline{y}$ in the race is higher.

3.4 Structural econometric model

Readings:

- The winner's curse, reserve prices, and endogenous entry: Empirical insights from eBay auctions
- Entry and competition effects in first-price auctions: theory and evidence from procurement auctions
- Auctions with entry

General two-step strategy:

- First step. Identify the marginal type from the data and the distribution of types.
- Second step. Using the estimated distribution of types.

Basic idea. Equilibrium condition gives:

$$y_i^* = y^*(a_i; F_{\mathcal{A}}). \tag{25}$$

with $y^*(\cdot)$ being an invertible function with ϕ denoting the inverse.

Hence the distribution of bids is

$$F_Y(y) = \Pr(y_i^* \leq y) = \Pr(y^*(a_i) \leq y) = \Pr(a_i \leq \phi(y)) = F_{\mathcal{A}}(\phi(y)). \tag{26}$$

Identification of the model. suggest

```r
races2 <- races
source("help_functions.R")
```

4 The experimental design

4.1 The context

The context of the experiment was an online programming competition conducted between March 2 and 16, 2016 on the online platform Topcoder.com. Registered participants were competing against one another on developing solutions to a hard Information Extraction (IE) problem. This consisted of developing source code to extract structured information from research papers on life sciences and biomedical topics. As an incentive for participation, a total prize pool of $41,000 was offered to top submissions.

Since its launch in 2001, Topcoder hosts on a regular basis similar programming competitions engaging thousands of participants from all over the world (Topcoder has about 1M registered users in 2016). Typical assigned problems include classification, prediction, or natural language processing algorithms applied to datasets of various kind. In most cases, the problem is hard to solve and demands some strong background in machine learning and statistics. To encourage participation, the competition is often designed as a "tournament" where top five submissions are awarded a monetary prize the extent of which depends on the nature and complexity of the problem (generally between $5,000 and $20,000). But it is not uncommon to find competitions called "first-to-finish" that are equivalent to a race. Note that, since these two competition formats are generally employed for different problems, it is problematic using naturally occurring data to make any proper comparison of behavior across competitions. In addition to monetary incentives, all active competitors attain a "skill rating" that provides a metric of their ability as contestants and sometimes play a role in signaling skills to potential employers. However, to better focus on the monetary incentives, our experiment was a non-rated event.

We worked together with researchers from the United States National Health Institute (NIH) and the Scripps Research Institute (SCRIPPS) to select a challenging computational problem for this competition. The selected problem was based on an algorithm called BANNER built by researchers at the NIH (Leaman, Gonzalez, and others 2008) that uses expert labeling to annotate abstracts from a prominent life sciences and biomedical search engine (PubMed) so disease characteristics can be more easily identified. The goal of the programming competition was to improve upon the current NIH's system by using a combination of expert and non-expert labeling, as described by Good et al. (2014).

- Why this was a good problem?
- Efforts to find a threshold
    - Survey of desired threshold
    - Preliminary reserved competition for 4

4.2 The design

A xxx-day preliminary registration phase resulted in 299 pre-registered members.[3] Registrants were then split into 24 separate rooms of either 10 or 15 people. Each room was then randomly assigned to a competitive condition.

We study three competitive conditions: i) a tournament, ii) a race, and iii) a tournament with minimum-quality. In the tournament condition, the first placed competitor was the highest score at the end of the competition. In the race condition, was the first to achieve a score of xxxx. And in the tournament with reserve condition, was the highest among those achiecing a score of xxxx. The level was chosen following a pre-trail experiment with 4 competitors solving the same problem. It also reflected the desired objective of the NIH researchers xxxx. To raise participation, additional . Note that in each condition competitors were sorted at random into 8 separate groups (four with 10 people and other four with 15 people). Hence, as shown in Table XXXX, our design generated a total of 3x8 = 24 groups.

Grand prizes of xxxx were awarded to the top xxx in every conditions.

```
tab <- with(races, table(treatment, room))
rownames(tab) <- capitalize(rownames(tab))
xtab <- xtable(tab, caption = "Experimental design", label = "experimental d
render.xtable2(xtab)
```

Table 1: Experimental design

	1	2	3	4	5	6	7	8
Race	9	10	10	10	15	15	15	15
Tournament	10	10	10	10	15	15	15	15
Reserve	10	10	10	10	15	15	15	15

The competition was announced on the platform and to all community members via email. A preliminary online registration was required to participate, which resulted in 340 pre-registered

[3] Here we excluded xxxx who xxxx requirements.

members. Among these pre-registered members, we selected the 299 with had registered to a programming contest at least once before the present contest. This choice was to ensure that participants were sufficiently experienced and understood the basic rules governing programming competitions.

In each of these groups, contestants were given access to a “virtual room” that is a private web page listing handles of the other participants in the group, a leaderboard to be updated regularly during the competition, and a common chat that they can use to ask clarifying questions with respect to the problem at hand.

A problem statement containing a full description of the algorithmic challenge, the rules of the game, and payoffs was published at the beginning of the submission phase. The submission phase was of 8 days in which participants could submit their computer programs. Each submission was automatically scored and feedback in the form of preliminary scores was published regularly on the website via the leaderboard.

4.3 Data

```
percent <- function(x, digits = 0, ...) {
    round(100 * x, digits, ...)
}
difftime2 <- function(...) {
    as.numeric(difftime(...))
}
attach(races)
rated <- !is.na(mm_rating)
years <- difftime2("2015-03-01", member_date, units = "days")/365
event_py <- mm_reg/years
tothours <- week1 + week2 + week3 + week4

# Default kernel density estimation
density2 <- function(x, ...) density(x, bw = "nrd", kernel = "gaussian",
    ...)

# Skill rating
rating_l <- split(mm_rating[rated], treatment[rated])
rating_l.pdf <- lapply(rating_l, density2, from = 500)
rating_l.test <- kruskal.test(rating_l)
```

```
rating.fig.cap <- sprintf("This picture shows kernel density estimates of tl
    rating_l.test$method, rating_l.test$p.value)

# Risk aversion
nomiss <- !is.na(risk)
risk_l <- split(risk[nomiss], treatment[nomiss])
risk_l.pdf <- lapply(risk_l, density2, from = 0, to = 10)
risk_l.test <- kruskal.test(risk_l)

risk.fig.cap <- sprintf("This picture shows kernel density estimates of the
    risk_l.test$method, risk_l.test$p.value)


# Total hours
nomiss <- !is.na(tothours)
tothours_l <- split(log(tothours[nomiss]), treatment[nomiss])
tothours_l.pdf <- lapply(tothours_l, density2, from = 0)
tothours_l.test <- kruskal.test(tothours_l)

tothours.fig.cap <- sprintf("This picture shows kernel density estimates of
    tothours_l.test$method, tothours_l.test$p.value)
```

For each registered competitor, we collected basic platform data including the time of the initial membership registration and statistics about participation in past programming competitions. The large majority (69 percent) were rated members with a median membership period of 6 years and a median of 16 registrations to past competitions (about 4.4 competitions per year) of which 6 with submissions. The rest were unrated members being on the platform for a median of 2 years with a median of 2 registrations and no submissions.

A key variable to measure was expected ability of competitors. We examined several proxies. For rated competitors, a sensible measure was the skill rating earned for the performance in past competitions. This measure is based on a version of the Elo system used in Chess, which is used by Topcoder to rank-order competitors at the end of each challenge. As shown in Figure 1, the skill rating distribution is right-skewed reflecting the presence of a few competitors with very high skill ratings compared to the average.[4] The figure also shows that the distribution of skill ratings was

[4]This right-skewed property is not specific of our sample but seems to hold more generally for the distribution of skill ratings of the entire platform.

the same in all three competitive conditions. We examined other proxies including the count of past wins, top ten positions, and the total prize money won while being a member of the platform. Though complete for both rated and unrated individuals in our sample, these other proxies contain less information about expected ability because most competitors have never won or earned a prize.

```
plot.density <- function(list) {
    xlim <- range(sapply(list, function(x) range(x$x)))
    ylim <- range(sapply(list, function(x) range(x$y)))
    legend.names <- capitalize(names(list))
    plot.new()
    plot.window(xlim = xlim, ylim = ylim, log = "", asp = NA)
    sapply(1:3, function(i) lines(list[[i]], lty = i, lwd = 2))
    legend("topright", bty = "n", legend.names, lty = 1:3)
    axis(1, at = pretty(seq(xlim[1], xlim[2], length = 20)))
}
plot.density(rating_l.pdf)
title("Skill rating distribution")
```

Additional demographic information was collected via a pre-registration survey where competitors were asked their gender, age, geographic origins, education, and the most preferred programming language. We also asked their willingness to take risks "in general," as a measure of risk aversion (Dohmen et al. 2011) and a forecast on how many hours they expected to be able to work on the problem in the next few days of the challenge.[5] As shown in Figure 2, individuals reported being more willing to take risks than not (the median response was 7 out of 10) and prepared to work on the problem a median of 24 hours in total over the eight day submission period.

```
plot.density(risk_l.pdf)
title("Willingness to take risks in general (risk aversion)")
mtext("Responses to general risk question\n(0=not at all willing, 10=very wi
    1, 3)
mtext("Respondents", 2, 3)
```

[5]The exact question was: "The submission phase begins March 08. Looking ahead a week, how many hours do you forecast to be able to work on the solution of the problem?"

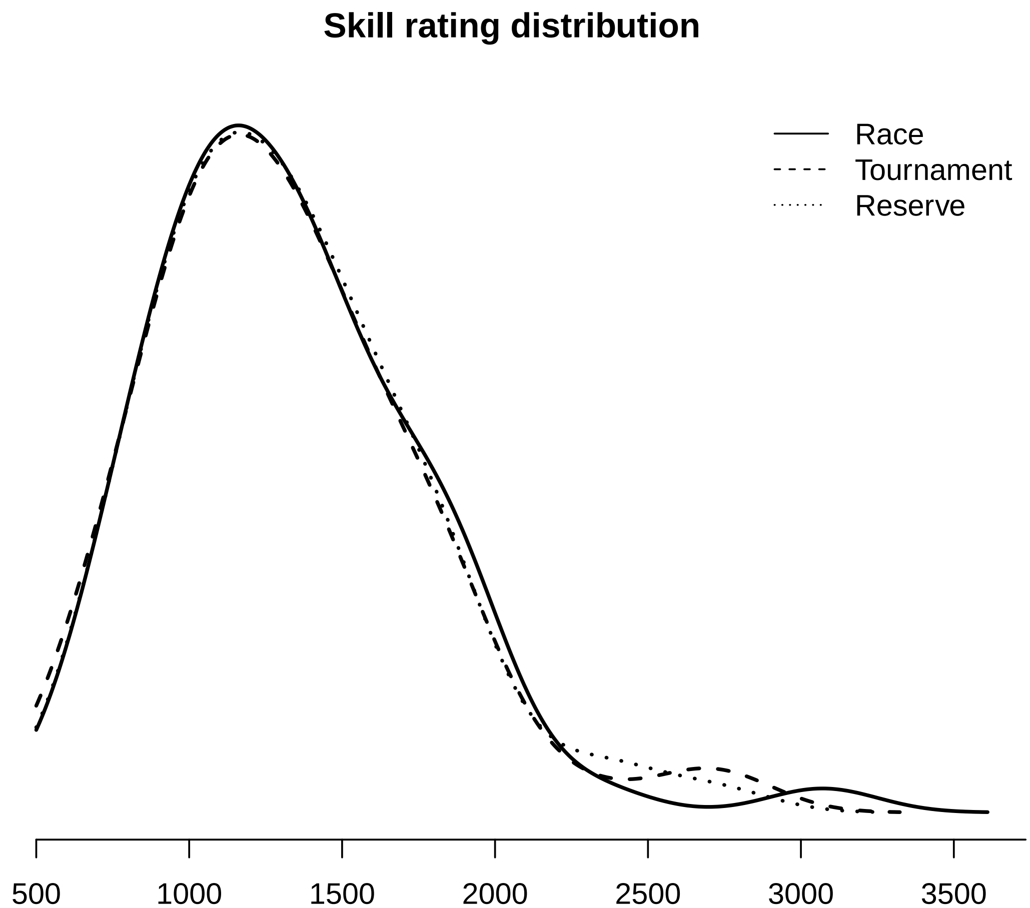


Figure 1: This picture shows kernel density estimates of the distribution of the skill ratings for each competitive condition. For testing whether samples originate from the same distribution we use a Kruskal-Wallis rank sum test that gives a pvalue of 0.989. Thus, we do not reject the null hypothesis of the data being drawn from the same distribution in each condition.

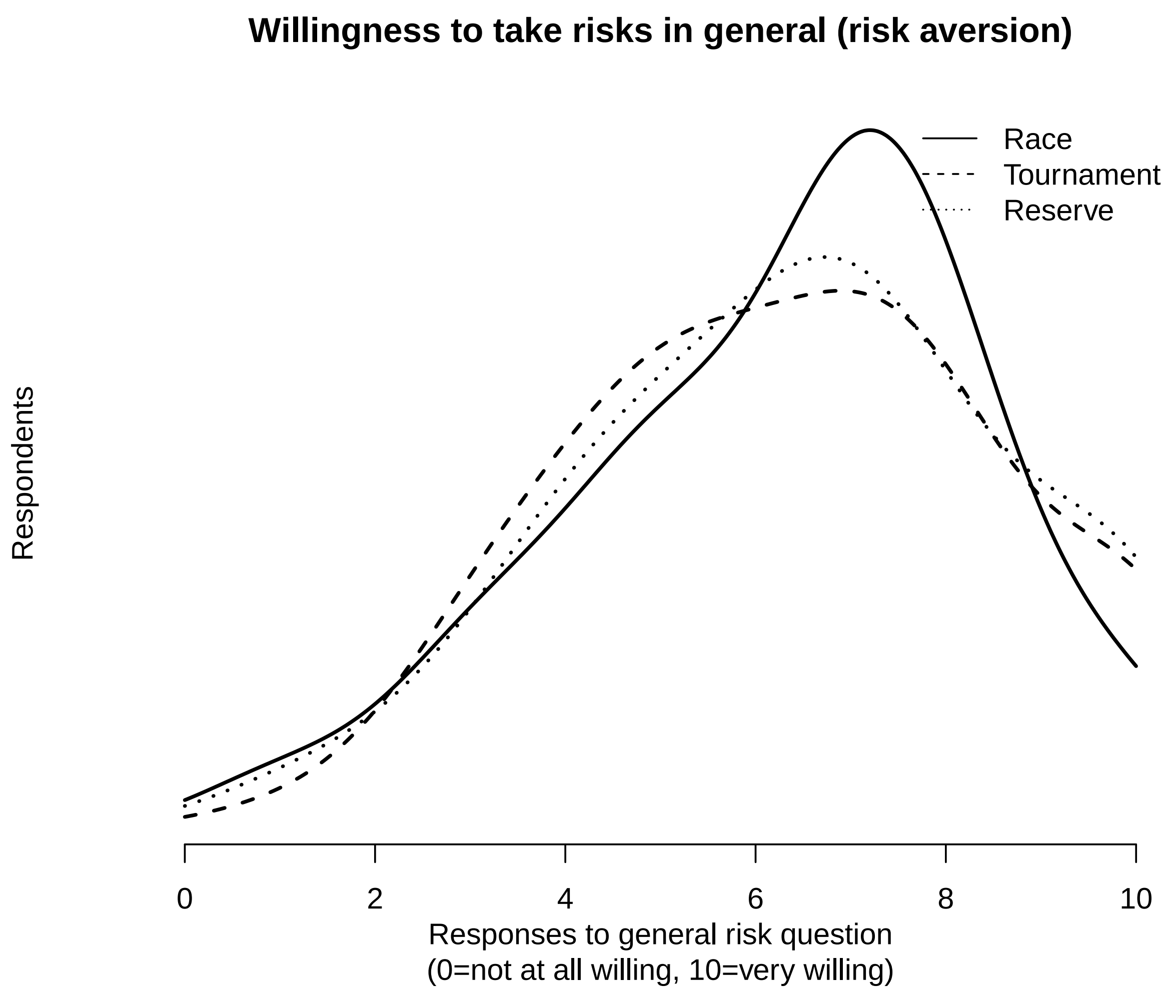
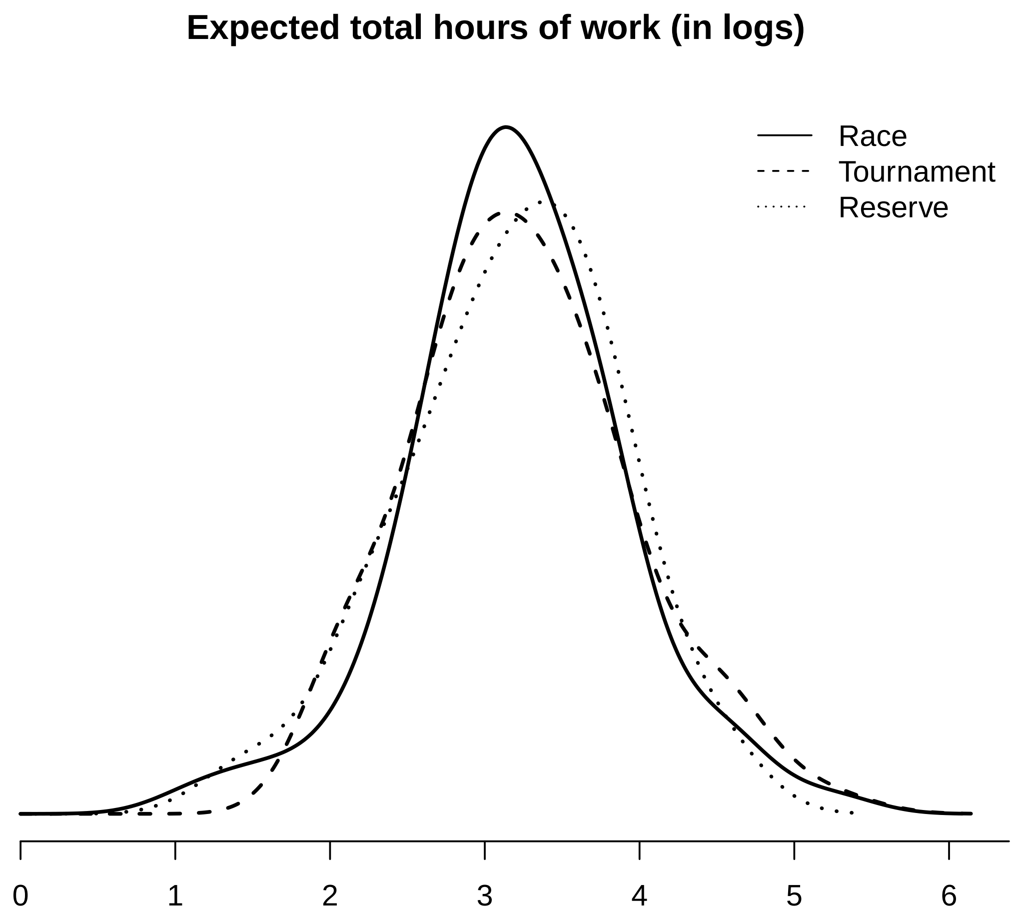


Figure 2: This picture shows kernel density estimates of the distribution of the responses to the question about willingness to take risks "in general" measured on an eleven-point scale for individuals in each competitive condition. For testing whether samples originate from the same distribution we use a a Kruskal-Wallis rank sum test that gives a pvalue of 0.958. Thus, the null hypothesis of the data being drawn from the same distribution cannot be rejected.

```r
plot.density(tothours_l.pdf)
title("Expected total hours of work (in logs)")
```




```r
with(races, boxplot(week1, week2, week3, week4, notch = TRUE,
    xlab = "Days of the competition\n(1=first 2 days, 4=last 2 days)",
    ylab = "Expected hours of work"))
axis(1, 1:4)
```

```r
# t.test(week1, week4) # wilcox.test(week1, week4)
```

```r
# Variables
demo <- cbind(age, gender, educ, plang, risk, tothours)
```

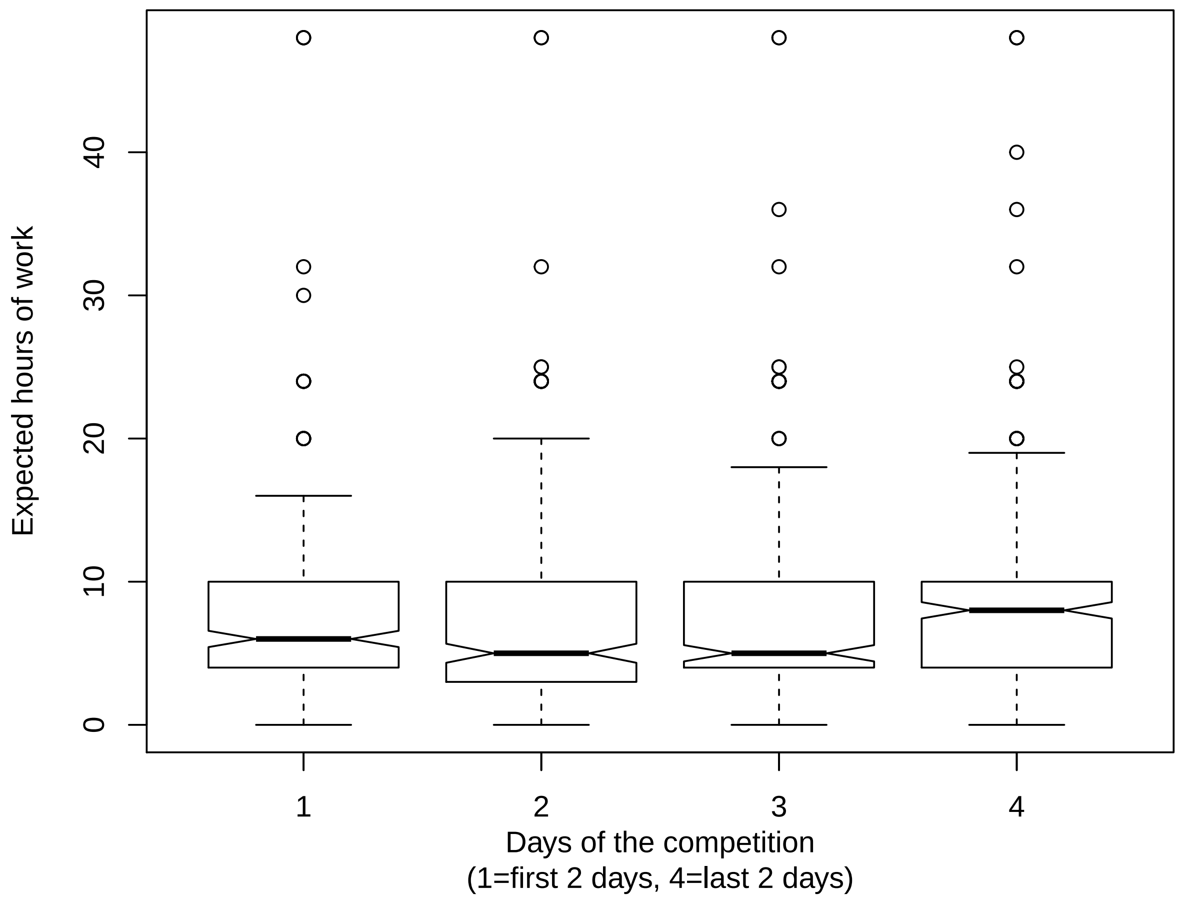


Figure 3: Responses to the question about expected hours of work for every 2 days of the competition.

```r
skills <- cbind(mm_reg, mm_events, mm_rating, nwins, ntop10,
    paid)

# Tests balanced covariates
test.balanced.categ <- function(x, ...) {
    chisq.test(table(x, races$treatment), ...)$p.val
}
test.balanced <- function(x, ...) {
    miss <- is.na(x)
    l <- split(x[!miss], races$treatment[!miss])
    kruskal.test(l, ...)$p.val
}

pval.categ <- apply(demo, 2, test.balanced, simul = TRUE)
pval <- apply(skills, 2, test.balanced, simul = TRUE)
```

The three experimental groups did not differ significantly in terms of the distribution of pre-treatment covariates. Using a Kruskal-Wallis rank sum test we find no difference in the distribution of participation measures (registrations, submissions) and ability proxies (skill rating, wins, top ten positions, earnings) across treatments (the lowest p-value was 0.273). Likewise, using a Pearson's Chi-squared test we find no association between each categorical variable (age, gender, education, programming language, risk attitudes,[6] and expected total hours of work) and treatments (the lowest p-value was 0.375). Hence, the randomization was successful in keeping pre-treatment variables balanced across competitive conditions.

```r
# Submissions
nsub <- tapply(scores$submission, scores$coder_id, max)
submissions <- tapply(scores$submission, scores$coder_id, max)

# Participation
par.percent <- round(100 * tapply(submit, treatment, mean))
par.table <- table(submit, treatment)
par.test <- fisher.test(par.table)
```

[6]Risk attitudes can be also modeled as a continuous variable and tested using the Kruskal-Wallis rank sum test. Results of this test are reported in Figure 2.

```
# Participation & Size
par.percent.size <- round(100 * tapply(submit, room_size, mean),
    1)
par.table.size <- table(submit, room_size)
par.test.size <- fisher.test(par.table.size)
```

5 Results

We obtained a total of 1759 submissions made by 86 participating coders (29 percent of the sample). Our theoretical model predicts that participation should be higher in the tournament condition compared to the other two forms of competition. Our data were only in part consistent with this prediction: the overall response rate in the Tournament groups (33 percent) was higher than that in the Race (26 percent) and in the Reserve (27 percent) condition. However, the overall association between treatments and response rates was not statistically significant (a two-sided Fisher's Exact Test for Count Data gives a p-value of 0.526).

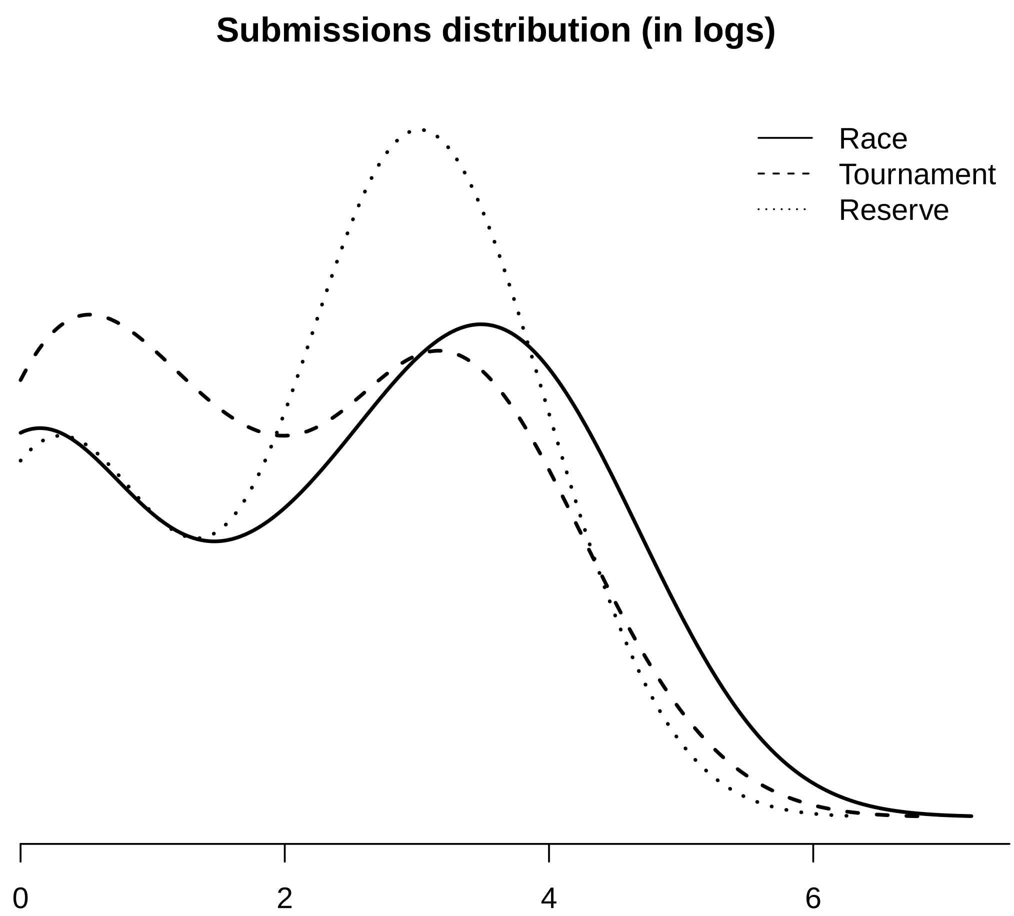
Our model also predicts lower participation for individuals in large rooms (15 competitors) compared to those in small rooms (10 competitors). Under the model, the "marginal type" is increasing in group size and so the individual probability of entry is lower. However, our data proved only negligible differences in participation between large (28.9 percent) and small rooms (28.6 percent). Additionally, we found no significant treatment differences conditional on the room size being large or small (xxxxx). Hence, one may conclude that a group-size difference of 5 people is probably not large enough to be impactful.

```
# Testing size! tab <- table(room_size, submit, treatment)
# fisher.test(tab[,,'race']) fisher.test(tab[,,'tournament'])
# fisher.test(tab[,,'reserve']) ftable(tab)
```

```
# Differences in submission distribution
scores2 <- merge(scores, races, by = "coder_id", all.x = TRUE)
scores3 <- aggregate(submission ~ coder_id + treatment, scores2,
    max)

nsub_l <- split(log(scores3$submission), scores3$treatment)
nsub_l.pdf <- lapply(nsub_l, density, from = 0)
nsub_l.test <- kruskal.test(nsub_l)
```

```r
# Bimodal distribution
plot.density(nsub_l.pdf)
title("Submissions distribution (in logs)")
```




```r
xlab <- pretty(seq(1, 150, length = 10))
```

```r
# Submissions ... prep data
scores$last_submission <- with(scores, ave(submission, coder_id,
    FUN = max))
scores2 <- merge(scores, races[, c("coder_id", "treatment", "room_size")])
scores3 <- subset(scores2, last_submission == submission)
```

```r
# Testing differences of submissions and treatment
submissions_l <- with(scores3, split(submission, treatment))

t.test(submissions_l$race, submissions_l$tournament)
```

```
##
##  Welch Two Sample t-test
##
## data:  submissions_l$race and submissions_l$tournament
## t = 1, df = 50, p-value = 0.3
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -7.96 22.59
## sample estimates:
## mean of x mean of y
##      25.6      18.3
```

```r
wilcox.test(submissions_l$race, submissions_l$tournament, alternative = "gre
```

```
## Warning in wilcox.test.default(submissions_l$race, submissions_l
## $tournament, : cannot compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  submissions_l$race and submissions_l$tournament
## W = 500, p-value = 0.3
## alternative hypothesis: true location shift is greater than 0
```

```r
wilcox.test(submissions_l$race, submissions_l$reserve, alternative = "greate
```

```
## Warning in wilcox.test.default(submissions_l$race, submissions_l$reserve,
## cannot compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
```

```
## data:  submissions_l$race and submissions_l$reserve
## W = 400, p-value = 0.4
## alternative hypothesis: true location shift is greater than 0
```

```r
# Testing differences of submissions and treatment
submissions_l <- with(scores3, split(submission, room_size))
t.test(submissions_l$Large, submissions_l$Small)  # No difference!!!
```

```
##
##  Welch Two Sample t-test
##
## data:  submissions_l$Large and submissions_l$Small
## t = -0.09, df = 80, p-value = 0.9
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -11.2  10.3
## sample estimates:
## mean of x mean of y
##      20.3      20.7
```

The median submission per participant was of 11 submissions, with a minimum of 1 and a maximum of 126 submissions.

6 Participation by experience

```r
attach(races)
```

```
## The following objects are masked from races (pos = 3):
##
##     age, algo_events, algo_rating, algo_reg, coder_id, country,
##     duration, educ, finished, gender, lastround, member_date,
##     mm_events, mm_rating, mm_reg, ntop10, nwins, paid, plang,
##     risk, room, room_size, startdate, submit, timezone, treatment,
##     week1, week2, week3, week4
```

```r
mm_qtl <- cut(mm_reg, quantile(mm_reg), include = T)
chisq.test(table(submit, mm_qtl))  ## p<0.01 !!!
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(submit, mm_qtl)
## X-squared = 10, df = 3, p-value = 0.002
```

```r
(tab <- ftable(treatment, submit, mm_qtl))
```

```
##                    mm_qtl [1,3] (3,9] (9,25] (25,161]
## treatment  submit
## race       FALSE               25    15     16       17
##            TRUE                 7     6      7        6
## tournament FALSE               29    12     16       10
##            TRUE                 7     5      6       15
## reserve    FALSE               24    19     16       14
##            TRUE                 1     6     10       10
```

```r
tothours <- week1 + week2 + week3 + week4
impute.random <- function(x) {
    index <- is.na(x)
    x[index] <- sample(x[!index], size = sum(index), replace = TRUE)
    return(x)
}
tothours_imp <- impute.random(tothours)
hours_qtl <- cut(tothours_imp, quantile(tothours_imp), include = TRUE)

mm_ratio <- mm_events/mm_reg
fit <- rep()
summary(fit$m0 <- glm(submit ~ treatment + mm_qtl + hours_qtl,
    binomial))
```

```
##
## Call:
```

```
## glm(formula = submit ~ treatment + mm_qtl + hours_qtl, family = binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.614  -0.816  -0.672   1.133   2.210
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -2.2115     0.4326   -5.11  3.2e-07 ***
## treatmenttournament    0.2918     0.3309    0.88   0.3780
## treatmentreserve      -0.0597     0.3370   -0.18   0.8595
## mm_qtl(3,9]            0.9197     0.4218    2.18   0.0292 *
## mm_qtl(9,25]           1.1402     0.4018    2.84   0.0045 **
## mm_qtl(25,161]         1.7283     0.4005    4.32  1.6e-05 ***
## hours_qtl(16,24]       0.0721     0.3941    0.18   0.8547
## hours_qtl(24,40]      -0.0802     0.3688   -0.22   0.8278
## hours_qtl(40,192]      1.1760     0.3856    3.05   0.0023 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 358.81  on 298  degrees of freedom
## Residual deviance: 328.51  on 290  degrees of freedom
## AIC: 346.5
##
## Number of Fisher Scoring iterations: 4
```

```r
summary(fit$m0 <- glm(submit ~ treatment + mm_qtl + hours_qtl +
    mm_ratio, binomial))
```

```
##
## Call:
## glm(formula = submit ~ treatment + mm_qtl + hours_qtl + mm_ratio,
##     family = binomial)
##
## Deviance Residuals:
```

```
##    Min      1Q  Median      3Q     Max
## -1.668  -0.810  -0.564   0.975   2.162
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -2.45557    0.45171   -5.44  5.4e-08 ***
## treatmenttournament  0.26603    0.33855    0.79  0.43198
## treatmentreserve    -0.00304    0.34596   -0.01  0.99299
## mm_qtl(3,9]          0.40630    0.45839    0.89  0.37543
## mm_qtl(9,25]         0.56972    0.44021    1.29  0.19559
## mm_qtl(25,161]       1.10642    0.44294    2.50  0.01249 *
## hours_qtl(16,24]    -0.03374    0.40280   -0.08  0.93324
## hours_qtl(24,40]    -0.04606    0.37609   -0.12  0.90252
## hours_qtl(40,192]    1.14263    0.39488    2.89  0.00381 **
## mm_ratio             2.09779    0.63648    3.30  0.00098 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 358.81  on 298  degrees of freedom
## Residual deviance: 317.30  on 289  degrees of freedom
## AIC: 337.3
##
## Number of Fisher Scoring iterations: 4
```

```r
summary(fit$m1 <- glm(submit ~ mm_qtl + hours_qtl + mm_ratio,
    binomial, subset = treatment == "race"))
```

```
##
## Call:
## glm(formula = submit ~ mm_qtl + hours_qtl + mm_ratio, family = binomial,
##     subset = treatment == "race")
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.790  -0.700  -0.513   0.529   2.161
```

```
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -2.762      0.853   -3.24   0.0012 **
## mm_qtl(3,9]              0.800      0.921    0.87   0.3851
## mm_qtl(9,25]             0.884      0.839    1.05   0.2919
## mm_qtl(25,161]           0.470      0.917    0.51   0.6081
## hours_qtl(16,24]         0.128      0.733    0.17   0.8615
## hours_qtl(24,40]        -0.208      0.780   -0.27   0.7893
## hours_qtl(40,192]        2.287      0.879    2.60   0.0093 **
## mm_ratio                 2.212      1.186    1.87   0.0621 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 114.01  on 98  degrees of freedom
## Residual deviance:  93.07  on 91  degrees of freedom
## AIC: 109.1
##
## Number of Fisher Scoring iterations: 5
```

```
summary(fit$m2 <- glm(submit ~ mm_qtl + hours_qtl + mm_ratio,
    binomial, subset = treatment == "tournament"))
```

```
##
## Call:
## glm(formula = submit ~ mm_qtl + hours_qtl + mm_ratio, family = binomial,
##     subset = treatment == "tournament")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.740   -0.790   -0.573    1.029    2.075
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -1.98632    0.63561   -3.13   0.0018 **
```

```
## mm_qtl(3,9]         0.28886    0.81892    0.35   0.7243
## mm_qtl(9,25]        0.01371    0.74679    0.02   0.9854
## mm_qtl(25,161]      1.52904    0.68634    2.23   0.0259 *
## hours_qtl(16,24]   -0.00825    0.70765   -0.01   0.9907
## hours_qtl(24,40]   -0.04202    0.65646   -0.06   0.9490
## hours_qtl(40,192]   0.98072    0.66190    1.48   0.1384
## mm_ratio            1.68897    1.12531    1.50   0.1334
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 126.84  on 99  degrees of freedom
## Residual deviance: 109.80  on 92  degrees of freedom
## AIC: 125.8
##
## Number of Fisher Scoring iterations: 4
```

```
summary(fit$m3 <- glm(submit ~ mm_qtl + hours_qtl + mm_ratio,
    binomial, subset = treatment == "reserve"))
```

```
##
## Call:
## glm(formula = submit ~ mm_qtl + hours_qtl + mm_ratio, family = binomial,
##     subset = treatment == "reserve")
##
## Deviance Residuals:
##    Min       1Q   Median       3Q      Max
## -1.315  -0.832  -0.427   1.128   2.352
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -3.3290     1.0886   -3.06   0.0022 **
## mm_qtl(3,9]          1.5727     1.1579    1.36   0.1744
## mm_qtl(9,25]         2.0209     1.1726    1.72   0.0848 .
## mm_qtl(25,161]       2.2418     1.1888    1.89   0.0593 .
## hours_qtl(16,24]    -0.2750     0.7546   -0.36   0.7155
```

```
## hours_qtl(24,40]   -0.0534     0.6235   -0.09   0.9318
## hours_qtl(40,192]   0.2189     0.7424    0.29   0.7681
## mm_ratio            2.0471     1.2181    1.68   0.0929 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 116.65  on 99  degrees of freedom
## Residual deviance: 100.12  on 92  degrees of freedom
## AIC: 116.1
##
## Number of Fisher Scoring iterations: 5
```

```r
stargazer(fit, type = "text")
```

```
##
## ============================================================
##                                 Dependent variable:
##                        ----------------------------------------
##                                       submit
##                            (1)       (2)       (3)       (4)
## ------------------------------------------------------------
## treatmenttournament    0.266
##                       (0.339)
##
## treatmentreserve      -0.003
##                       (0.346)
##
## mm_qtl(3,9]            0.406     0.800     0.289     1.570
##                       (0.458)   (0.921)   (0.819)   (1.160)
##
## mm_qtl(9,25]           0.570     0.884     0.014    2.020*
##                       (0.440)   (0.839)   (0.747)   (1.170)
##
## mm_qtl(25,161]        1.110**    0.470    1.530**   2.240*
##                       (0.443)   (0.917)   (0.686)   (1.190)
```

```
##
## hours_qtl(16,24]      -0.034     0.128    -0.008    -0.275
##                       (0.403)   (0.733)   (0.708)   (0.755)
##
## hours_qtl(24,40]      -0.046    -0.208    -0.042    -0.053
##                       (0.376)   (0.780)   (0.656)   (0.624)
##
## hours_qtl(40,192]    1.140***  2.290***    0.981     0.219
##                       (0.395)   (0.879)   (0.662)   (0.742)
##
## mm_ratio             2.100***   2.210*     1.690    2.050*
##                       (0.636)   (1.190)   (1.120)   (1.220)
##
## Constant            -2.460*** -2.760*** -1.990*** -3.330***
##                       (0.452)   (0.853)   (0.636)   (1.090)
##
## -----------------------------------------------------------
## Observations            299        99       100       100
## Log Likelihood       -159.000   -46.500   -54.900   -50.100
## Akaike Inf. Crit.     337.000   109.000   126.000   116.000
## ===========================================================
## Note:                           *p<0.1; **p<0.05; ***p<0.01
```

```r
# Tournament --> participation odds for expert coders are
# about 40% higher in tournaments than in the race

# Nice plot with bootstrap
races$mm_qtl <- mm_qtl
races.l <- split(races, treatment)
# table(submit, mm_qtl, treatment) -> tab
```

```r
devtools::session_info()
```

```
## Session info -----------------------------------------------------------------

##  setting  value
##  version  R version 3.3.2 (2016-10-31)
```

```
##  system   x86_64, darwin13.4.0
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  tz       America/New_York
##  date     2017-05-18

## Packages ----------------------------------------------------------------

##  package   * version date       source
##  backports   1.0.5   2017-01-18 CRAN (R 3.3.2)
##  codetools   0.2-15  2016-10-05 CRAN (R 3.3.2)
##  devtools    1.12.0  2016-06-24 CRAN (R 3.3.0)
##  digest      0.6.12  2017-01-27 CRAN (R 3.3.2)
##  evaluate    0.10    2016-10-11 CRAN (R 3.3.0)
##  formatR     1.4     2016-05-09 CRAN (R 3.3.0)
##  highr       0.6     2016-05-09 CRAN (R 3.3.0)
##  htmltools   0.3.5   2016-03-21 CRAN (R 3.3.0)
##  knitr     * 1.15.1  2016-11-22 CRAN (R 3.3.2)
##  magrittr  * 1.5     2014-11-22 CRAN (R 3.3.0)
##  memoise     1.0.0   2016-01-29 CRAN (R 3.3.0)
##  Rcpp        0.12.9  2017-01-14 CRAN (R 3.3.2)
##  rmarkdown   1.3     2016-12-21 CRAN (R 3.3.2)
##  rprojroot   1.2     2017-01-16 CRAN (R 3.3.2)
##  stargazer * 5.2     2015-07-14 CRAN (R 3.3.0)
##  stringi     1.1.2   2016-10-01 CRAN (R 3.3.0)
##  stringr     1.2.0   2017-02-18 CRAN (R 3.3.2)
##  withr       1.0.2   2016-06-20 CRAN (R 3.3.0)
##  xtable    * 1.8-2   2016-02-05 CRAN (R 3.3.0)
##  yaml        2.1.14  2016-11-12 CRAN (R 3.3.2)
```

References

Athey, Susan, and Philip A Haile. 2002. "Identification of Standard Auction Models." *Econometrica* 70 (6). Wiley Online Library: 2107–40.

———. 2007. "Nonparametric Approaches to Auctions." *Handbook of Econometrics* 6. Elsevier: 3847–3965.

Athey, Susan, Jonathan Levin, and Enrique Seira. 2011. “Comparing Open and Sealed Bid Auctions: Evidence from Timber Auctions*.” *The Quarterly Journal of Economics* 126 (1). Oxford University Press: 207–57.

Baye, Michael R, and Heidrun C Hoppe. 2003. “The Strategic Equivalence of Rent-Seeking, Innovation, and Patent-Race Games.” *Games and Economic Behavior* 44 (2). Elsevier: 217–26.

Bimpikis, Kostas, Shayan Ehsani, and Mohamed Mostagir. 2014. “Designing Dynamic Contests.” Working paper, Stanford University.

Che, Yeon-Koo, and Ian Gale. 2003. “Optimal Design of Research Contests.” *The American Economic Review* 93 (3). American Economic Association: 646–71.

Dechenaux, Emmanuel, Dan Kovenock, and Roman M Sheremeta. 2014. “A Survey of Experimental Research on Contests, All-Pay Auctions and Tournaments.” *Experimental Economics*. Springer, 1–61.

Dixit, Avinash Kamalakar. 1987. “Strategic Behavior in Contests.” *The American Economic Review* 77 (5). American Economic Association: 891–98.

Dohmen, Thomas, Armin Falk, David Huffman, Uwe Sunde, Jürgen Schupp, and Gert G Wagner. 2011. “Individual Risk Attitudes: Measurement, Determinants, and Behavioral Consequences.” *Journal of the European Economic Association* 9 (3). Wiley Online Library: 522–50.

Donald, Stephen G, and Harry J Paarsch. 1996. “Identification, Estimation, and Testing in Parametric Empirical Models of Auctions Within the Independent Private Values Paradigm.” *Econometric Theory* 12 (03). Cambridge Univ Press: 517–67.

Good, Benjamin M, Max Nanis, CHUNLEI Wu, and ANDREW I Su. 2014. “Microtask Crowdsourcing for Disease Mention Annotation in Pubmed Abstracts.” In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 282–93. NIH Public Access.

Green, Jerry R, and Nancy L Stokey. 1983. “A Comparison of Tournaments and Contracts.” *The Journal of Political Economy* 91 (3): 349–64.

Grossman, Gene M, and Carl Shapiro. 1987. “Dynamic R&D Competition.” *Economic Journal* 97 (386). Royal Economic Society: 372–87.

Harris, Christopher, and John Vickers. 1987. “Racing with Uncertainty.” *The Review of Economic Studies* 54 (1). Oxford University Press: 1–21.

Laffont, Jean-Jacques, Herve Ossard, and Quang Vuong. 1995. “Econometrics of First-Price Auctions.” *Econometrica* 63 (4). Econometric Society: 953–80.

Lazear, Edward P, and Sherwin Rosen. 1981. “Rank-Order Tournaments as Optimum Labor Contracts.” *The Journal of Political Economy* 89 (5): 841–64.

Leaman, Robert, Graciela Gonzalez, and others. 2008. “BANNER: An Executable Survey of Advances in Biomedical Named Entity Recognition.” In *Pacific Symposium on Biocomputing*, 13:652–63.

Mary, O'Keeffe, W Kip Viscusi, and Richard J Zeckhauser. 1984. "Economic Contests: Comparative Reward Schemes." *Journal of Labor Economics* 2 (1). University of Chicago Press: 27–56.

Moldovanu, Benny, and Aner Sela. 2001. "The Optimal Allocation of Prizes in Contests." *The American Economic Review*. JSTOR, 542–58.

———. 2006. "Contest Architecture." *Journal of Economic Theory* 126 (1). Elsevier: 70–96.

Paarsch, Harry J. 1992. "Deciding Between the Common and Private Value Paradigms in Empirical Models of Auctions." *Journal of Econometrics* 51 (1-2). Elsevier: 191–215.

Parreiras, Sérgio O, and Anna Rubinchik. 2010. "Contests with Three or More Heterogeneous Agents." *Games and Economic Behavior* 68 (2). Elsevier: 703–15.

Siegel, Ron. 2009. "All-Pay Contests." *Econometrica* 77 (1). Wiley Online Library: 71–92.

———. 2014. "Contests with Productive Effort." *International Journal of Game Theory* 43 (3). Springer: 515–23.

Szymanski, Stefan. 2003. "The Economic Design of Sporting Contests." *Journal of Economic Literature* 41 (4). American Economic Association: 1137–87.

Taylor, Curtis R. 1995. "Digging for Golden Carrots: An Analysis of Research Tournaments." *The American Economic Review* 85 (4). American Economic Association: 872–90.

Zizzo, Daniel John. 2002. "Racing with Uncertainty: A Patent Race Experiment." *International Journal of Industrial Organization* 20 (6). Elsevier: 877–902.