Races or Tournaments?[1]

[DRAFT: Please, do not distribute]

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

Last updated: 01 December, 2016

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

Contents

Abstract

A wide range of economic and social situations are decided by either a race or a tournament. In such contests, agents choose whether and how much to exert some costly effort to increase the probability of being awarded a prize under uncertainty about the other agents types or actions. In theory, whenever the sponsor of the competition prefers competitors' performance over the time to complete a particular task, the expected outcomes of a tournament setup should be either equal or greater than those of a race. Yet, a race might be more efficient from an economic point of view as it may prevent unnecessary costs due to an excess of participation. We examine this trade-off empirically. We report the results of a field experiment conducted on a leading crowdsourcing platform where we compare the outcomes (efforts, quality, and diversity of outputs) of three alternative competitive situations motivated by theory: the race, the tournament, and the tournament with a quality requirement.

JEL Classification: XX; XX; XX;

Keywords: xx; xx; xx;

1 Introduction

Contests are a very important source of incentives in the economy. In the United States, the government regularly sponsors open competitions aimed at solving various challenging problems of public health, education, energy, environmental issues, and so on.[1] In the private sector, firms use contests as an effective tool either to rapidly expand their innovative ability via open innovation initiatives or as a means to lead workers to exert higher levels of effort via the competition for bonuses, pay increases, and promotions. Thus, understanding how to effectively design a contest is an important economic issue.

In this study, we focus on the difference between two quite popular choices of contest design: the "race" (a competition to be first) and the "tournament" (a competition to be best). In particular, we aim at addressing the following research questions: How this choice affect the choices of contestants? When the sponsor of a competition should pick one or the other? What is the main trade-off? How this decision interact with other key choices of design such as the distribution of prizes among winners?

To address these questions we proceed in two ways. First, we generalize the well known incomplete information contest model of @moldovanu2001optimal. This is enables a direct comparison of equlibrium behaviors under both the race and the tournament within a single theoretical framework. Then, we collect data from a field experiment that we desiged and run to test some of the implications of the theory, and provide policy recommendations.

Economists have a long tradition in studying races, of various kinds (patent and arms races), and tournaments (sales tournaments). A large body of works have investigated several aspects of contest design, including the optimal prize structure [XXX], number of competitors [XXX], and the

[1] On a monthly basis, the web portal www.challenge.gov publishes calls for new online challenges seeking problem solvers from all around the world.

appeal of imposing minimum effort requirments [XXX]. Seldom economists have considered the competitive setting being either a race or a tournament as the result of a deliberate choice of contest design. Usually the nature of the competition is seen as exogenous and due to the environemental factors (for example, in a patent race, the type of competition is determined by the legal environment and cannot be easily changed by policy makers). One important exception is @baye2003strategic that identifies a strategic equivalence between games of complete information modeled as races and tournaments. This paper is important because connects extends to races the results from a wide literature on the optimal design of a tournament. However, it does not shed light on when it would be desirable to use one or the other.

By this perspective, we are able to show that races cannot be justified simply by the goal of maximizing average effort. And the reason is intuitive. A race awards a prize to first to hit a particular target. Those who will judge the target to hard to achieve will not join the competition and will drop out. On the contrary, those who are able to achieve the target at low costs will not try to exceed the target. As a result, the race is comparable to a competition with fixed "entry costs" or a fixed entry requirement, where agents will decide to either enter and pay a fixed prize, or stay out of the competition. Then, the possible gains in terms of expected revenues from a race are limited to those who would enter the competition and would exert less effort that that required to hit the target. These potential benefits can be obtained under a tournament as well by imposing a a fixed requirement to be eligible for prizes. So, races are not chosen to maximize expected effort of competitors, at least, in the traditional "auction-theoretical" sense.

What are races for? We examine a few hypothesis and we provide some examples. First hypothesis is that the sponsor of the race is not primarily interested in total output but also in the time to complete a particular task. In a tournament, this type of preferences can be satisfied by fixing a deadline. Say time within which competitors are asked to provide their efforts.

However, assuming competitors have costs from making less time in performing a task and there complementarities in costs, increasing the deadline in a tournament is similar to raising the marginal cost for everyone, which might not be an optimal solution. In a race, by contrast, increasing the deadline will affect entry but, conditional on entry, the time to complete the task will always be less than the deadline. Which means that those with low costs will be mostly affected by the deadline, whereas xxxx. Which may be a superior choice than the tournament.

To fix ideas let consider the following example. The government wants to solve a global public health problem such as "antibiotics resistance." The overuse of antibiotics leads to the phenomenon of "resistance" which is a loss in the power of antibiotics to treat certain infections. This is an increasing threat for public health. The government has the choice of making a contest to engage people in solving this problem. The government has preferences for time in the sense that the government wants to minimize to have the first submission. So, the government fix a requirement in terms of costs of the solution and award a prize to the first to meet this requirement. Example. UK governemet goes for a race. EU xxxx goes for a tournament with a deadline in 2016. (. . .) 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.

There is also a case for efficiency. Consider a platform with many competitions. The platform may want to engage competitors for short period of time provide that solutions are above a certain quality level.

To test our theory we further examine experimental data on competitors making sumibssion in an online computer programming contest. We randomized competitors into 3 groups: 1. race 2. tournament 3. tournament with a quality requirement we study participation, timing of submission and final scores.

We find that, as our theory suggest, participation is higher in the tournament and lower in the race and in the tournament with entry costs. We further find that 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. With respect to final scores, theory predicts as 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 @dixit1987strategic @baye2003strategic, @parreiras2010contests, @moldovanu2001optimal, @moldovanu2006contest, @siegel2009all, @siegel2014contests. It also relates to the literature on innovation contests @taylor1995digging, @che2003optimal. And the personnel economics approach to contests @lazear1981rank, @green1983comparison, @mary1984economic.

Empirically, @dechenaux2014survey provide a comprehensive summary of the experimental literature on contests and tourments. Large body of empirical works have focused on sports contests @szymanski2003economic. More recently, inside firms (xxx) and online contest (xxxx).

This paper is also related to the econometrics of auctions @paarsch1992deciding, @laffont1995econometrics, @donald1996identification and more recently @athey2011comparing, @athey2002identification, and @athey2007nonparametric.

3 Races and tournaments

3.1 The basic theoretical model

Consider a contest with k available prizes of value $V_1 > V_2 > ... > V_k$. Each agent $(i = 1, ..., n + 1)$ moves simultaneously to maximize the chances of winning a prize. To be elegible for a prize, each agent has to complete a task within a deadline $d > 0$. Outcomes are then evaluated and ranked along two dimensions: the output quality y_i and the time to completion t_i both being nonnegative real numbers.

In a tournament, the agent having achieved the highest output quality within the deadline gets the first prize, the agent having achieved the second highest output quality gets the second prize, and so on. In a race, by contrast, the first agent to achieve an output quality of at least $\underline{y}$ within the deadline wins the first prize, the second to achieve the same target gets the second prize, and so on.

Since agents move simultaneously, they do not know the performance of others when deciding their efforts. On the other hand, it is assumed that they know the number of competitors as well as their cost functions to complete the task up to a factor a_i being the agent's private ability in performing the task. Each agent knows his ability but does not know the ability of the others. However, it is common knowledge that abilities are drawn at random from a common distribution F_A that is assumed everywhere differentiable on the support $A \subseteq [0, \infty)$.

Each agent is risk neutral and maximizes the chances of winning a prize $\Pr(\cdot)$ minus the costs of effort $C(\cdot)$ as

$$\text{maximize} \quad \textstyle\sum_{j=1}^{k} \Pr(\text{ranked } j\text{'th}) V_j - C(y_i, t_i, a_i). \tag{1}$$

Costs are assumed multiplicative

$$C(y_i, t_i, a_i) = c_y(y) \cdot c_t(t) \cdot a_i^{-1} \tag{2}$$

with $c_y(0) \geq 0$, $c_y' > 0$, $c_t(d) \geq 0$, and $c_t' < 0$.

The contest designer chooses the rules of the competition, including prize structure and target quality, to maximize the output quality while keeping low the time to completion. Net of the prize paid, the objective function is

$$\text{maximize} \quad \int_0^\infty y^*(a)dF_A(a) - \tau \int_0^\infty t^*(a)dF_A(a) \tag{3}$$

where the parameter $\tau \in [0, 1]$ denotes the utility weight attached by the sponsor to an increase in the average time to completion.

3.1.1 Equilibrium in a tournament

We provide here the symmetric equilibrium with two prizes and $n > 3$ agents. In the appendix, we provide a general formula for $k > 2$ prizes. Also, let normalize the prize pool $V_1 + V_2 = 1$ and use the percentage $\alpha \geq 1/2$ to denote the fraction of the prize pool going to the first placed competitor.

In a tournament, the unique symmetric equilibrium of the model gives, for every $i = 1, ..., n$, the optimal time to completion $t^*(a_i)$ equal to the deadline d and the optimal output quality $y^*(a_i)$ as

$$y^*(a_i) = c_y^{-1}\left[c_y(0) + \frac{1}{c_t(d)}\left(\alpha \int_{a_i}^{\infty} ...dz + (1-\alpha)\int_{a_i}^{\infty} ...dz\right)\right] \tag{4}$$

if $a_i \geq \underline{a}$ [see @moldovanu2001optimal], and equal to zero otherwise.

The main difference with @moldovanu2001optimal's one-dimensional equilibrium outcome can be understood by noting that, for every positive output quality $y > 0$, any time to completion $t < d$ is strictly dominated by $t = d$, and, for $y^* = 0$, the choice of the time to completion is irrelevant. So, the optimal choice of time to completion is either making use of all the time available or quitting the contest.

An important property of (4) is that $y^*(a_i)$ has its upper bound in XXXX and lower bound in zero. Also, equilibrium output quality is monotonic increasing in the agent's ability [see @moldovanu2001optimal]. Thus, for every $i = 1, ..., n+1$, the equilibrium expected reward $R(a_i)$ depends only on the rank of his ability relative to the others and can be written as

$$R(a_i) \equiv \alpha F_A(a_i)^n + (1-\alpha)n[1 - F_A(a_i)]F_A(a_i)^{n-1} \tag{5}$$

Hence, by setting (5) equal to costs and solving for the ability, gives the marginal ability $\underline{a}$ as

$$\underline{a} = \sup\{a \in A : R(a_i) - C(0, d, a) = 0\}. \tag{6}$$

This gives the functional relationship between the $\underline{a}$ and the parameters

$$\underline{a} = h(n, \alpha, F_A, C_0) \tag{7}$$

with $C_0 \equiv c_y(0)c_t(d)$.

3.1.2 Equilibrium in a race

In a race, the unique symmetric equilibrium of the model gives, for every $i = 1, ..., n$, the optimal quality $y^*(a_i)$ equal to the minimum requirement $\underline{y}$ and the optimal time to completion $t^*(a_i)$ as

$$t^*(a_i) = c_t^{-1}\left[c_t(d) + \frac{1}{c_y(0)}\left(\alpha \int_{a_i}^{\infty} ...dz + (1-\alpha)\int_{a_i}^{\infty} ...dz\right)\right] \tag{8}$$

if $a_i \geq \underline{a}$, and equal to zero otherwise.

Again, the equilibrium time to completion is monotonic (decreasing) in the ability and, therefore, as in the case of a tournament, we can derive the same functional relationship between the marginal type and the parameters

$$\underline{a}_{\text{race}} = h(n, \alpha, F_A, C_{\underline{y}}) \tag{9}$$

with $C_{\underline{y}} \equiv c_y(\underline{y})c_t(d)$ instead of C_0.

By comparing (7) with (9) one important property emerges that is summarized in the next proposition.

Proposition 1. *All else equal, participation is higher in a tournament compared to a race.*

A race introduces higher entry costs that drive low-ability competitors out of the contest. This effect can be beneficial to the contest sponsor in two ways. First, it may lower the total costs due to the time spent by competitors on the problem. Second, the target may induce some agents to increase effort and, hence, output quality compared to what they would do in a tournament. This second effect is summarized in the next proposition.

Proposition 2. *All else equal, there always exists a subset of types $\tilde{A} \subset A$ for which the output quality is higher in a race compared to a tournament.*

This result implies that, regardless of the time to completion, a tournament never "dominates" a race in terms of output quality ex-post, although it might do so ex-ante.

3.1.3 Example

To fix ideas, we consider the following example. Imagine cost functions are non-linear as $c_t(t) = 2e^{1-t}$ and $c_y(y) = y^2$, as shown in Figure XXXX below. We further assume that abilities are drawn from a common log-Normal distribution.

Figure XXX shows the theoretical equilibrium bidding functions for a race and a tournament with varying competitors and prize structures. As can be seen, the gains from a race tend to disappear as the number of competitors grows and with a winner-takes-all prize structure.

Note, these pictures do not show the differences in time to completion, which should be also taken into account to make a fuller comparison of gains for the contest design.

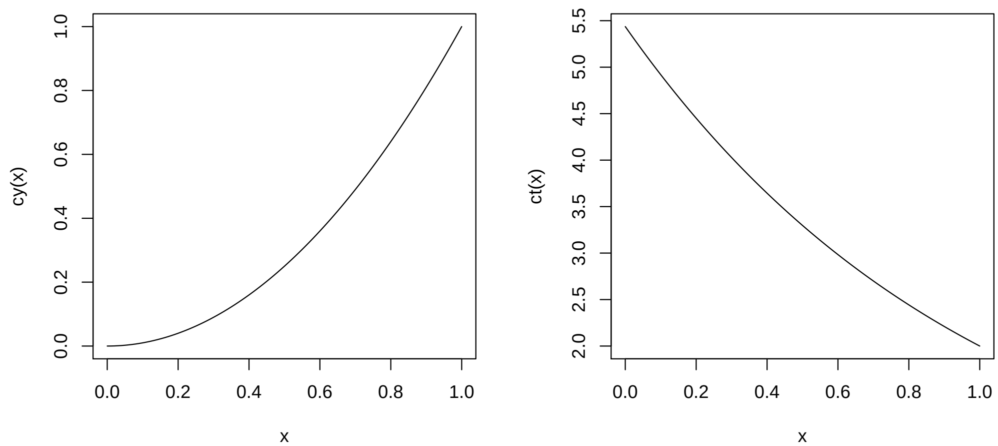


Figure 1: Example of non-linear cost functions with respect to output quality (left panel) and time to completion (right panel).

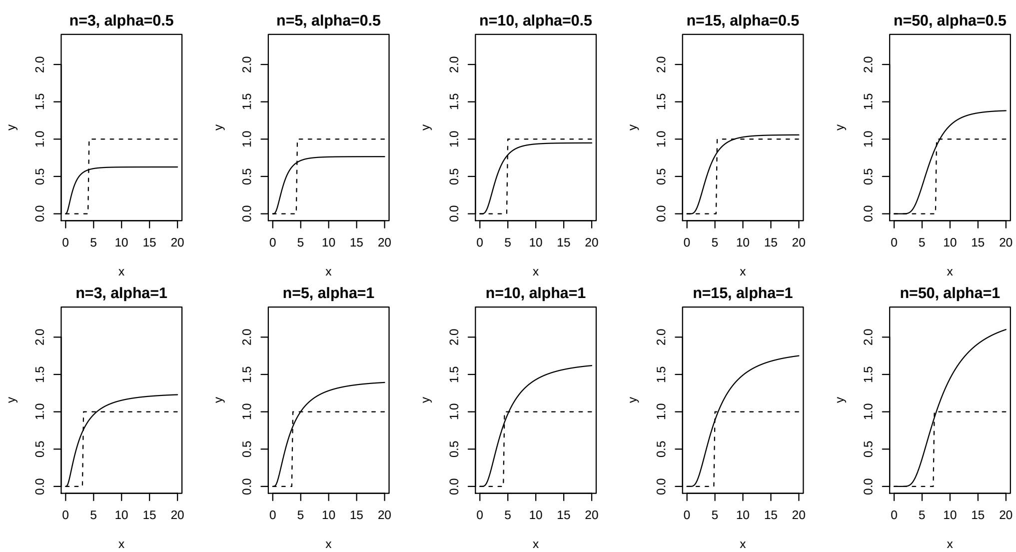


Figure 2: Equilibrium bidding functions of quality in a tournament (—) and in a race (- - -) with abilities drawn from a log Normal distribution. The number of competitors and the fraction of prize pool to the winner α is at the top of each panel.

3.2 Structural econometric model

We adopt a structural approach to the empirical comparison of races and tournaments. This means that the econometric model for the observed outcomes is derived from the theoretical model of contests described above. Thus the basic parameters that determine the outcomes are the number of competitors, the costs, the rewards, and the distribution of private abilities. This modeling approach allows us to estimate the distribution of private values, which is useful for policy analysis.

Consider the choice of making a submission or not. As shown before, the optimal choice for an agent depends on its private ability. It is optimal to make a submission only if the agent's private ability a is higher than that of the marginal type $\underline{a}$ (or $\underline{a}_{\text{race}}$ in a race). This marginal value depends on some known parameters, such as the number of competitors and the rewards, as well as on the distribution of abilities, that is unknown.

We shall adopt a parametric formulation for F_A and, for every i, we assume the private ability is drawn from a log-normal distribution with mean μ and standard deviation σ. Equivalently, we have that the ability is $a_i = e^{\mu+\sigma Z_i}$ with a random variable $Z_i \sim \text{Normal}(0,1)$.

The structural model describing the optimal choice of entry is then

$$\Pr(a_i > \underline{a}) = \Pr(e^{\mu+\sigma Z_i} > \underline{a}) = \Pr(Z_i > \frac{\underline{a} - \mu}{\sigma}) = \Phi\left(\frac{\mu - \log(\underline{a})}{\sigma}\right). \quad (10)$$

The $\underline{a}$ can now be estimated by the method of moments or mle. Let consider the method of moments:

$$\hat{p} = n\Phi\left(\frac{\mu - \log(\underline{a})}{\sigma}\right) \quad (11)$$

we need to impose $\sigma = 1$ and $\mu = 0$ to identify the xxx.

The log-likelihood function to maximize is

$$ll = \sum_{i=1}^{n} y_i \log(e^{-\underline{a}^\lambda}) + (1 - y_i) \log(1 - e^{-\underline{a}^\lambda}). \tag{12}$$

We can use the Weibull together with the zero-profit condition XXXX to get an equation for the $\underline{a}$.

$$\alpha e^{n\underline{a}^k} + (1 - \alpha)n[1 - e^{n\underline{a}^k}]e^{(n-1)\underline{a}^k} - C_0\underline{a}^{-1} = 0. \tag{13}$$

This simplifies to:

$$\underline{a}e^{(n-1)\underline{a}^\lambda} \left[\alpha e^{\underline{a}^\lambda} + (1 - \alpha)n(1 - e^{n\underline{a}^\lambda})\right] = C_0 \tag{14}$$

which has no closed solution, :(, but can be solved numerically.

Also, since $C_0 > 0$, $\underline{a}$ has its lower bound in xxxx.

4 A comparison in the field

4.1 The experimental setup

In this section, we illustrate the preceding econometric methods by an empirical analysis of the outcomes of a field experiment that we conducted to compare races and tournaments. The context of the experiment was an online programming competition among expert software developers, engineers, and computer scientists. This competition was conducted on the online platform Topcoder.com from March 2 to March 16, 2016.

Since its launch in 2010, Topcoder.com hosts on a weekly basis competitive programming contests for thousands of competitors from all over the world. Typical assigned problems are data science problems (e.g., classification, prediction, natural language processing) that involve some background in machine learning and statistics to be solved. All Topcoder members – about 1M registered users in 2016 – can compete and attain a "rating" that provides a metric of their ability as a competitor. Other than attaining a rating, the competitors having made the top five submissions usually win a monetary prize. The extent of these awards can range considerably depending on the nature and complexity of the problem, generally between $5,000 to $20,000.

In this study, we worked together with researchers from the National Health Institute (NIH) and the Scripps Research Institute (SCRIPPS) to select a challenging problem for an experimental programming competition. The selected problem was based on an algorithm, called BANNER, built by NIH that uses expert labeling to annotate abstracts from PubMed, the most prominent life sciences and biomedical search engine, so disease characteristics can be more easily identified [@leaman2008banner]. The goal of the challenge was to improve upon the current NIH's system by using a combination of expert and non-expert labeling along the lines of @good2014microtask.

The contest was announced to all community members via email. Following a three days registration period, signed up competitors were randomly assigned to separate groups. Each group had to solve the same computational problem, as described above, within a period of 2 weeks.

Groups were then randomly assigned to one of three different competitive settings: a race, a tournament, and a tournament with a minimum quality requirement. In all groups, the first placed competitor was awarded a prize of $1,000, and an additional, consolatory prize of $100 was awarded to the second placed competitor. However, in a race, xxxx. In a tournament, xxxx .And in a tournament with minimum quality requirement, xxxx.

Outcomes were matched with data from each competitor's online web profile on the platform. This typically includes the date in which the member registered, the rating, the number of past competitions, and so on. Additional personal information, was collected via a mandatory initial and a final survey. In the initial survey, registered competitors were asked basic demographics, including a measure of risk aversion. They were also asked to forecast the number of hours they would be able to spend competing in the next few days (the exact question was: “looking ahead xxxx”). In the final survey, they were asked to look back and tell us their best estimate of the time spent working on the problem.

4.2 Data

A total of 299 competitors signed up for the challenge. All were registered members of the platform, with the median time as member of the platform of 5 years (min = 1 and max = 15 years). The distribution of experience in competing was highly skewed, with competitors in the 90th percentile having taken part in 28 more competitions and with a skill rating of 999 higher points than those in the 10th percentile; see Figure 3.

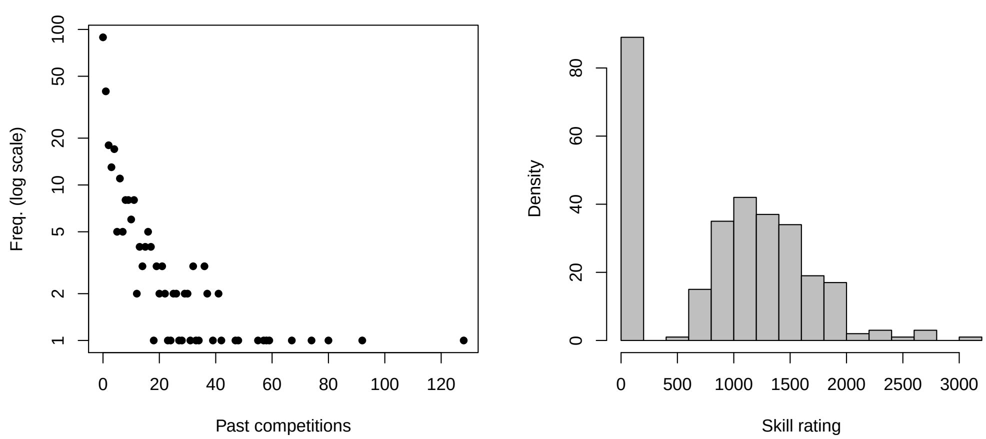


Figure 3: Distribution of the count of past contests (left panel) and the skill ratings (right panel) of the signed-up competitors.

After the two-week submission period, 86 competitors made 1759 submissions overall. The distribution of submissions was rather skewed, with participants in the 90th percentile having made 50 more submissions than those in the 10th percentile.

This result does not seem to correlate well with the competitor's experience or skills, as the Pearsons's correlation coefficient between the count of past competitions or the rating and the count of submissions is positive but generally low; see Table XXX. Thus, differences in submissions appear idiosyncratic and perhaps related to the way to organize the work rather than systematically associated with underlying differences in experience or skills.

```
##                  nsub mm.rating   mm.count
## nsub       1.00000000 0.1147754 0.04634501
## mm.rating 0.11477537 1.0000000 0.33330457
## mm.count  0.04634501 0.3333046 1.00000000
```

The timing of submissions was rather uniform during the submission period with a peak of submissions made in the last of the competition. (explain more)

```
scores$submax <- ave(scores$subid, scores$id, FUN=max)
par(mfrow=c(2, 1), mar=c(4,4,2,2))
plot(subid==1 ~ as.POSIXct(subts), data=scores, type='h', yaxt='n'
    , xlab='', ylab='', main='Dispersion time first submission')
plot(subid==submax ~ as.POSIXct(subts), data=scores, type='h'
    , yaxt='n', xlab='', ylab='', main='Dispersion time last submission')
```

Scores: xxxx

5 Empirical analysis

5.1 Treatment differences

Difference in participation by treatments are show in Table XX.

```
Fisher's Exact Test for Count Data
```

data: tab p-value = 0.5255 alternative hypothesis: two.sided

Margins computed over dimensions in the following order: 1: 2: entry % latex table generated in R 3.2.3 by xtable 1.8-2 package % Wed Nov 30 12:20:11 2016

	No submission	Submission	Total
Race	73	26	99
Tournament	67	33	100
Reserve	73	27	100
Total	213	86	299

We find no differences in the room size.

```
Fisher's Exact Test for Count Data
```

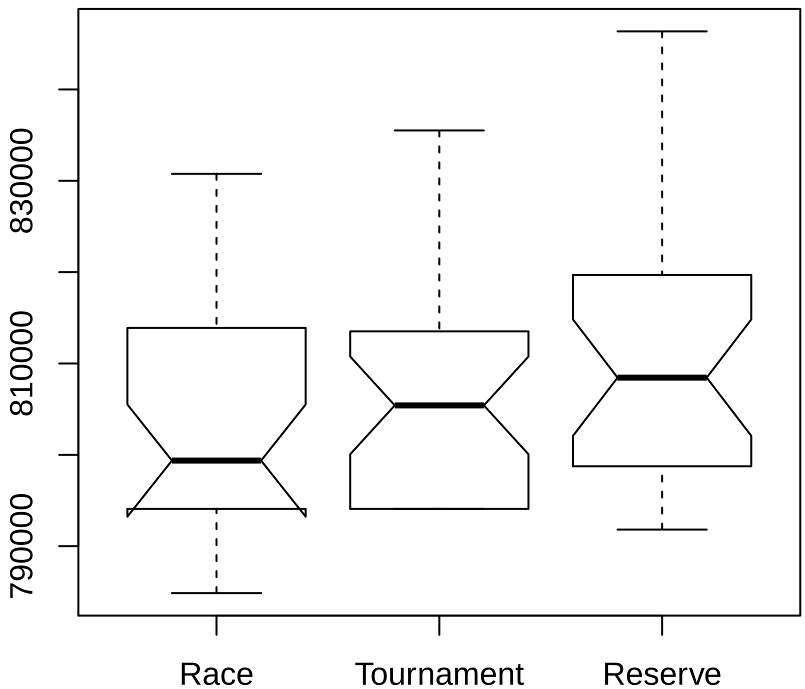
data: tab p-value = 1 alternative hypothesis: true odds ratio is not equal to 1 95 percent confidence interval: 0.5689643 1.6912635 sample estimates: odds ratio 0.9846648

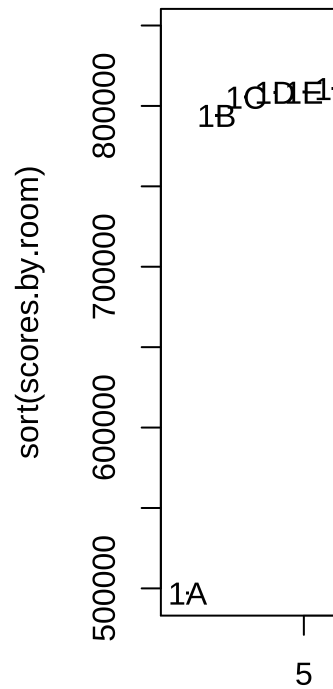
Margins computed over dimensions in the following order: 1: 2: % latex table generated in R 3.2.3 by xtable 1.8-2 package % Wed Nov 30 12:20:11 2016

	FALSE	TRUE	Total
Large	128	52	180
Small	85	34	119
Total	213	86	299

Ex-post

```
## Warning in bxp(structure(list(stats = structure(c(784858.58, 794084.93, :
## some notches went outside hinges ('box'): maybe set notch=FALSE
```





Timing: early vs late

Using a Chi-square test of independence, we find no significant differences in participation rates associated with the assigned treatments (p-value: 1); see Table XX.

Further, we model participation rates as a logistic regression. We use a polynomial of third degree for the count of past competitions to account for non-linear effects of experience; and we use an indicator for whether the competitor had a win or not. Also, taking into account differences in ability, participation rates are not significantly different.

5.2 Estimation results

Participation to the competition by treatment is shown in Figure ??. Participation here is measured by the proportion of registered participants per treatment who made any submission during the eight-day submission period. Recall that competitors may decide to enter into the competition and work on the problem without necessarily submitting. In a tournament, for example, competitors are awarded a prize based on their last submission and may decide to drop out without submitting anything. However, this scenario seems unlikely. In fact, competitors often end up making multiple submissions because by doing so they obtain intermediate feedback via preliminary scoring (see Section XXX for details). In a race, competitors have even stronger incentives to make early submissions as any submission that hits the target first wins.

`Table xxx`

We find that the propensity to make a submission is higher in the Tournament than in the Race and in the Tournament with reserve, but the difference is not statistically significant (a Fisher's exact test gives a p-value of `round(fisher.test(nsub.tab)$p.val,3)`). As discussed in Section XXX, we may not have enough power to detect differences below 5 percentage points. However, we find the same not-significant result in a parametric regression analysis of treatment differences with controls for the demographics and past experience on the platform; see Table ??. Adding individual covariates reduces variability of outcomes, potentially increasing the power of our test. In particular, Table ?? reports the results from a logistic regression on the probability of making a submissions. Column 1 reports the results from a baseline model with only treatment dummies. Column 2 adds demographics controls, such as the age, education, and gender. Column 3 adds controls for the past experience on the platform. Across all these specifications, the impact of the treatment dummies (including room size) on entry is not statistically significant.

5.3 Simulation results

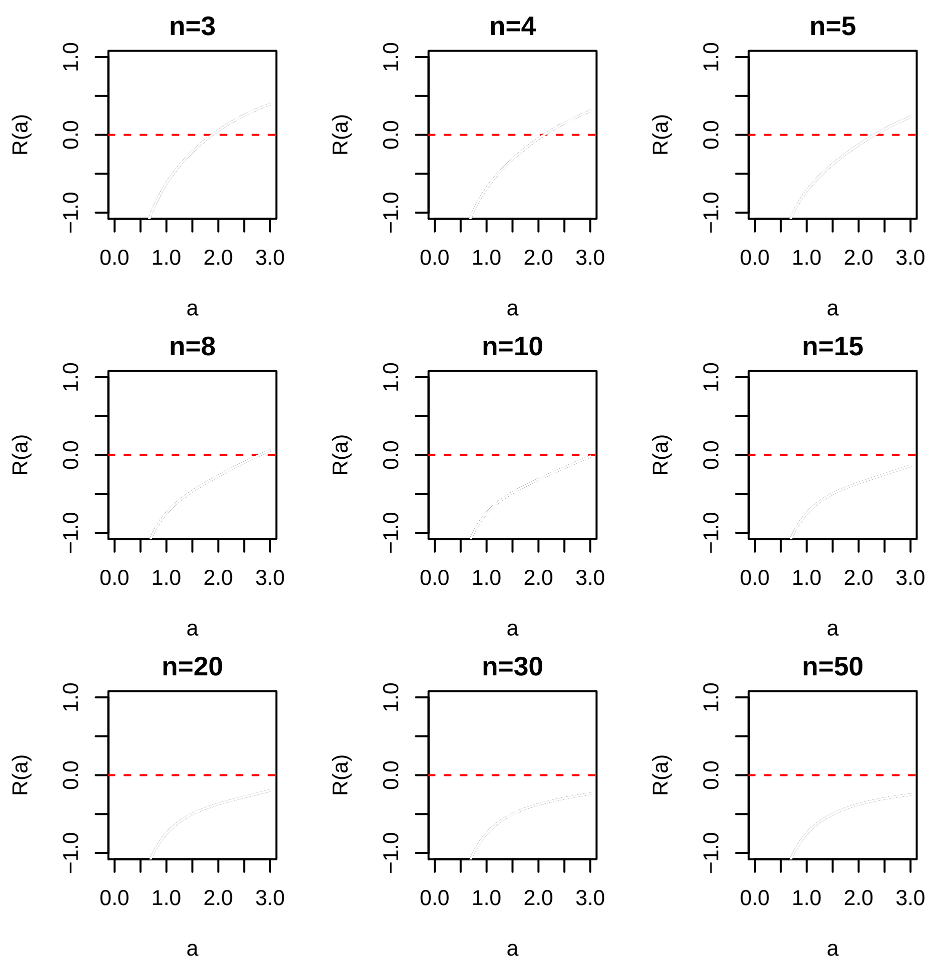
6 Structural estimation

```r
load("races.RData") # Data

########################################
# Simulation to gain confidence
# Parameters:
#   alpha = 1
#   c0    = 0.75
#   n     = 10
########################################
p <- plnorm
r <- rlnorm
zeroprofit <- function(a, n, c0, ...) {
  p(a, ...)^n - c0 * a^(-1)
}
marginal <- function(n, cost, ...) {
  uniroot(f=zeroprofit, n=n, c0=cost, interval=c(0.0001, 10), ...)$root
}

# Plot 1
old.par <- par(mfrow=c(3,3), mar=c(4,4,2,2))
for (nn in c(3, 4, 5, 8, 10, 15, 20, 30, 50)) {
  curve(zeroprofit(x, n=nn, c0=0.75), from=0.001, to=3
      , main=sprintf("n=%i",nn), ylab="R(a)", xlab="a"
      , ylim=c(-1, 1))
  abline(h=0, lty=2, col=2)
```

```
  for (ss in seq(0.5, 5, length=10))
    curve(zeroprofit(x, n=nn, c0=0.75), add=TRUE, col=gray(ss/5))
}
```

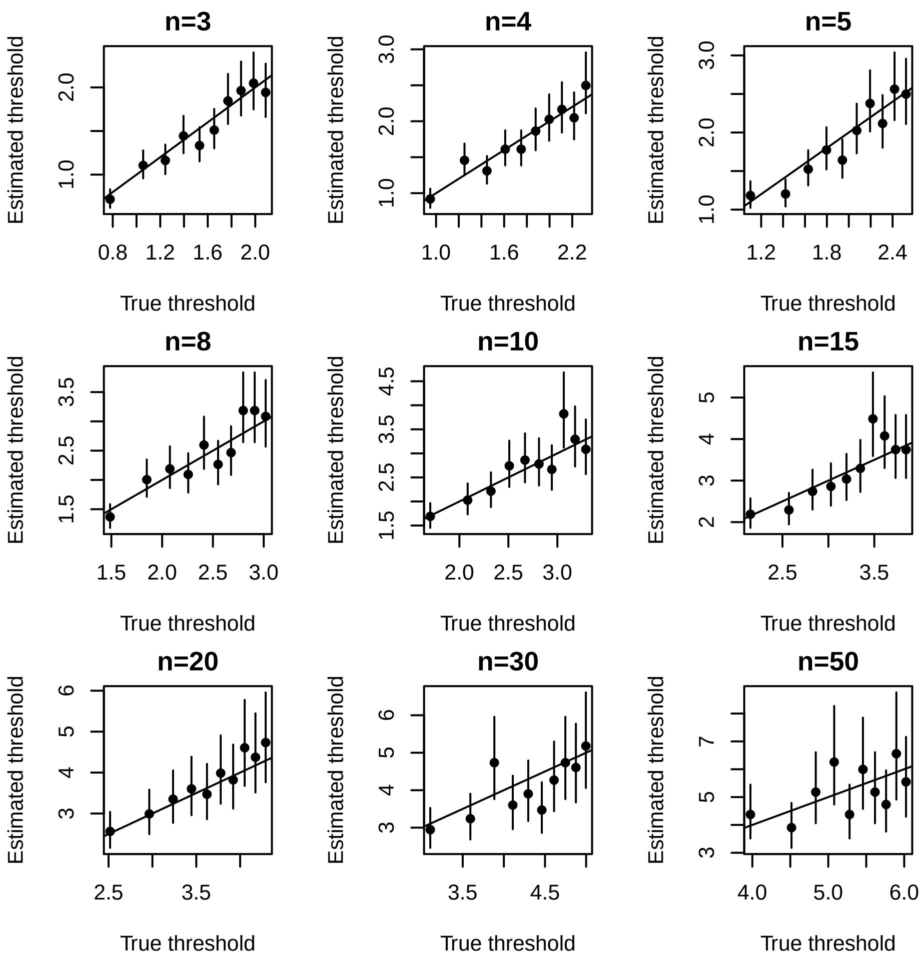



```
par(old.par)

# Function to simulate and estimate data
estimate <- function(n, obs, n.seq) {
  costs <- seq(0.05, 0.95, length=n.seq)
  m <- matrix(ncol=4, nrow=length(costs))
```

```
  colnames(m) <- c("a", "a.hat", "a.ci95hi", "a.ci95lo")
  for (i in 1:length(costs)) {
    # Sim
    th <- marginal(n=n, cost=costs[i])
    y <- ifelse(r(obs) > th, 1, 0)
    # Fit
    fit <- glm(y ~ 1, family=binomial('probit'))
    # Save
    a.hat <- exp(-coef(fit))
    a.ci95hi <- exp(-coef(fit) + 2*coef(summary(fit))[2])
    a.ci95lo <- exp(-coef(fit) - 2*coef(summary(fit))[2])
    m[i, ] <- c(th, a.hat, a.ci95hi, a.ci95lo)
  }
  return(m)
}

# Plot 2
old.par <- par(mfrow=c(3,3), mar=c(4,4,2,2))
for (nn in c(3, 4, 5, 8, 10, 15, 20, 30, 50)) {
  m <- estimate(nn, 300, 10)
  plot(m, pch=16, ylim=range(m)
       , main=sprintf("n=%i",nn)
       , ylab="Estimated threshold", xlab="True threshold")
  segments(x0=m[,1], y0=m[, 3],y1=m[, 4])
  abline(a=0, b=1)
}
```




```
par(old.par)

# Analysis
entry <- ifelse(is.na(dat$nsub), 0, 1)
treatment <- dat$treatment
fit <- glm(entry ~ treatment, family=binomial('probit'))

# Baseline (race)
bhat <- coef(fit)
race <- exp(-bhat[1])
```

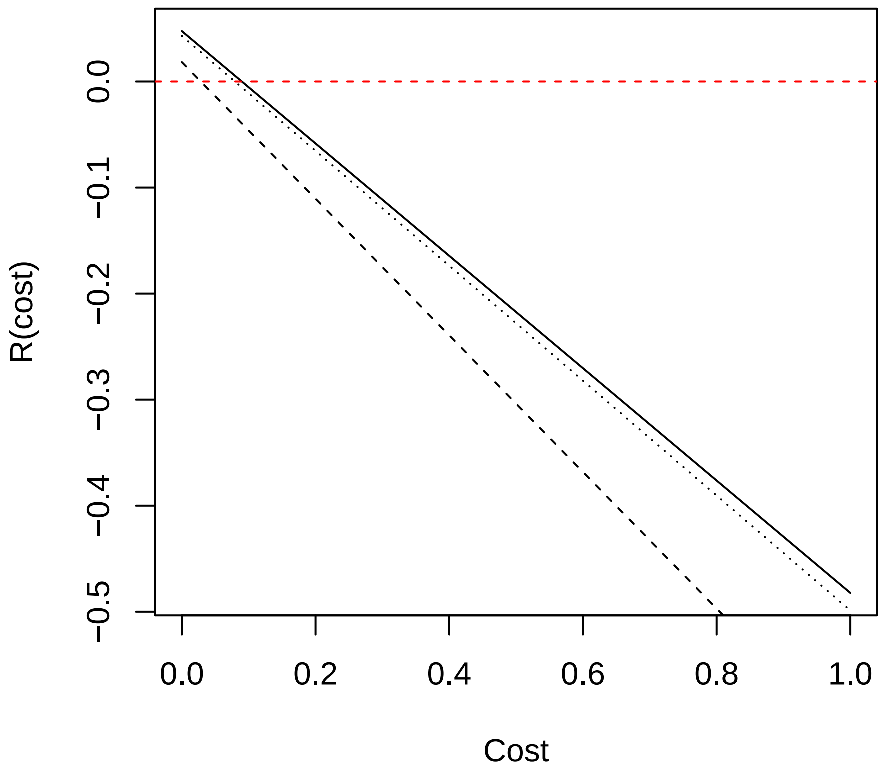
```r
tour <- exp(-sum(bhat[-3]))
targ <- exp(-sum(bhat[-2]))

# Estimated thresholds in ability
cbind(race, tour, targ)

##                 race     tour     targ
## (Intercept) 1.887531 1.552572 1.845616

# Estimated costs
zeroprofit.cost <- function(c0, n, a, ...) {
  zeroprofit(a, n, c0, ...)
}
marginal.cost <- function(n, a, ...) {
  uniroot(f=zeroprofit.cost, n=n, a=a, interval=c(0.0001, 0.999), ...)$root
}

curve(zeroprofit.cost(x, n=10, a=race)
      , xlab="Cost"
      , ylab="R(cost)")
curve(zeroprofit.cost(x, n=10, a=tour), add=TRUE, lty=2)
curve(zeroprofit.cost(x, n=10, a=targ), add=TRUE, lty=3)
abline(h=0, lty=2, col=2)
```



A Appendix

A.1 Simulating the model in R

```
f <- function(n, cy, ct, cy.inv=NULL, ct.inv=NULL
              , xlim=c(0, Inf), ylim=c(0, Inf), tlim=ylim
              , p=plnorm, d=dlnorm, r=rlnorm
              , deadline=1, target=1, V=1, inf.tol=1e2
              , type=c("bids", "utility"), ...) {
```

```
#------------------------------------------------#
# Bidding function for races and tournaments.
#
# Args:
#  n: Competitors.
#  cy: Cost function of quality.
#  ct: Cost function of time.
#  ylim, tlim: Range values of costs.
#  cy.inv, ct.inv: Inverse cost function.
#  p, d, r: Probability distribution of ability.
#  upper: upper limit of ability.
#  V: vector of prizes structures.
#  deadline, target: model parameters.
#  plot.results: plot bidding curves.
#
# Returns:
#   Nothing.
#
# Example:
# cy <- function(x) x^2
# ct <- function(x) 2*exp(1-x)
# z.lnorm <- f(n=c(3, 5, 10, 15, 50), cy=cy, ct=ct)
# z.wbull <- f(n=c(3, 5, 10, 15, 50), cy=cy, ct=ct
#              , p=pweibull, d=dweibull, r=rweibull, shape=1.5)
# ymax <- range(c(z.lnorm$ability$y, z.wbull$ability$y))[2]
# plot(z.lnorm$ability, typ='l', ylim=c(0, ymax))
# lines(z.wbull$ability, typ='l', lty=2)
#------------------------------------------------#
if (any(V < 1/2 | V > 1)) stop("V must be higher then 1/2 and less than 1.")
type <- match.arg(type)
```

```
inverse <- function (f, lower, upper) {
  if (upper==Inf) upper <- inf.tol
  function (y) uniroot((function (x) f(x) - y), lower=lower, upper=upper)$root
}
if(is.null(cy.inv)) cy.inv <- inverse(cy, lower=cy(ylim[1]), upper=cy(ylim[2]))
if(is.null(ct.inv)) ct.inv <- inverse(ct, lower=ct(tlim[1]), upper=ct(tlim[2]))

# Density of an order statistic.
dord <- function(x, k, n, ...) {
  z <- lfactorial(n) - lfactorial(k-1) - lfactorial(n-k) +
    (k-1)*p(x, log.p=TRUE, ...) +
    (n-k)*p(x, log.p=TRUE, lower.tail=FALSE, ...) +
    d(x, log=TRUE, ...)
  return(exp(z))
}

# Distribution of an order statistic.
pord <- function(x, k, n, ...) {
  z <- 0
  for (j in k:n) {
    z <- z + lfactorial(n) - lfactorial(j) - lfactorial(n-k) +
    (j) * p(x, log.p=TRUE, ...)
    (n-j) * p(x, log.p=TRUE, lower.tail=FALSE, ...)
  }
  return(exp(z))
}

# Define bidding function
b <- function(x, n, V, ...) {
```

```
  integrand <- function(x, k, n, ...) dord(x, k, n, ...) * x
  integrand2 <- function(x, k, n, ...) {
    (- dord(x, k, n, ...) * pord(x, k-1, n-1, ...) +
    dord(x, k-1, n-1, ...) * (1-pord(x, k, n, ...))) * x
  }
  integral <- function(x, n, ...) {
    integrate(integrand, k=n, n=n, upper=x, lower=xlim[1], ...)$val
  }
  integral2 <- function(x, n, ...) {
    integrate(integrand2, k=n, n=n, upper=x, lower=xlim[1], ...)$val
  }
  cy.inv(cy(ylim[1]) + (V*integral(x, n, ...) + (1-V)*integral2(x, n, ...)) / c
}

# Equilibrium expected utility
u <- function(x, ystar, tstar, k, n, V, ...) {
    if (x==0) return(-Inf)
    V * pord(x, k, n, ...) +
    (1-V) * (1-pord(x, k, n, ...)) * pord(x, k-1, n-1, ...) -
    cy(ystar)*ct(tstar) / x
}

# Equilibrium score in a race
y.race <- function(x, n, V, ...) {
  xbar <- uniroot(u, ystar=target, tstar=deadline, k=n, n=n, V=V, interval=c(0,
  ifelse(x < xbar, 0, target)
}

# Equilibrium score in a tournament
y.tournament <- function(x, n, V, ...) {
```

```
    sapply(x, b, n=n, V=V, ...)
  }

  if (type=="bids") {
    par(mfrow=c(length(V), length(n)), mar=c(4,4,2,2))
    ybar <- max(target, max(sapply(n, y.tournament, x=xlim[2], V=max(V), ...)))
    for (j in 1:length(V)) {
      for (i in 1:length(n)) {
        curve(y.tournament(x, V=V[j], n=n[i], ...), from=xlim[1], to=min(inf.tol,
            , ylim=c(0, ybar), ylab='y', xlab='x'
            , main=sprintf("n=%s, v=%s", n[i], signif(V[j], 2)))
        curve(y.race(x, V=V[j], n=n[i], ...), add=TRUE, lty=2)
      }
    }
  }
  if (type=="utility") {
      print("xxxx")
  }
}
```

A.1.1 Individual ability distribution

Ability $a \in [0, \infty)$ is iid and drawn from a common (absolutely continuous) distribution $a \sim F$.

A.1.2 Cost functions

Multiplicative cost function (Cobb-Douglas) $C(y, t) = c_y(y)c_t(t)/a$ where the ability a is a cost shifting parameter.

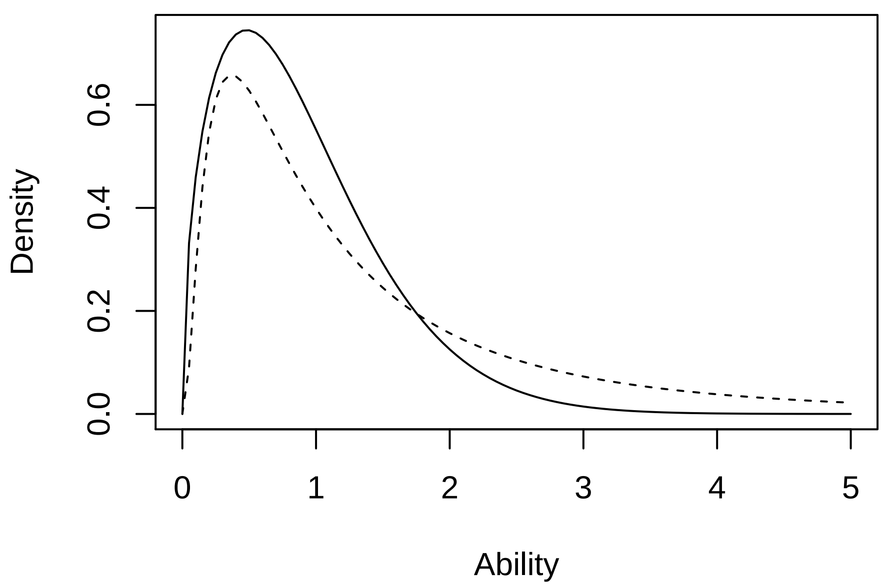


Figure 4: Players' abilities that are independently and identically drawn from the log-normal (——) and the Weibull (- - -) distribution (shape parameter of 1.5).

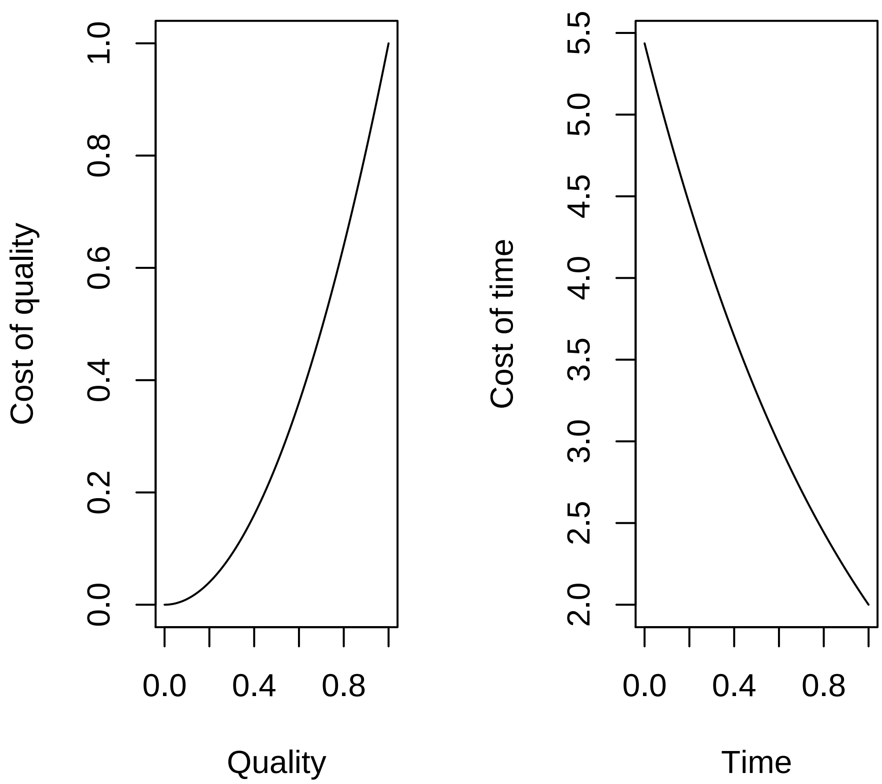
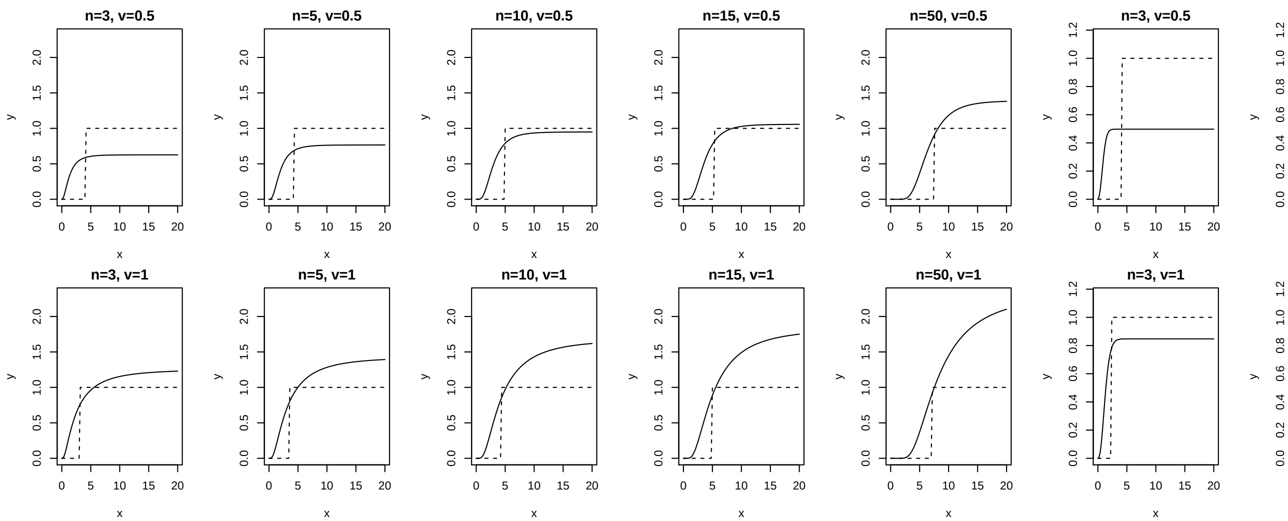


Figure 5: Cost functions for quality and time.

A.1.3 Example of bidding functions

In a tournament, it is a dominant strategy to submit at the deadline $t^* = d$ or quit the competition $q^* = 0$. The equilibrium quality is as shown in the figures. By contrast, in a race, it is a dominant strategy to meet the target quality $q^* = \bar{q}$ or quit the competition $q^* = 0$.



A.2 Further notes about identification

- Since there is imperfect information each agent will view all y_j with $j \neq i$ as random variables. Let $F_{Y_{j:n}}$ denote the distribution of the jth smallest value of the (n less the agent) Y_j's

- The problem faced by each agent i is

$$\text{maximize} \quad \Pr(y > Y_{n:n})V_1 + \Pr(Y_{n:n} > y > Y_{n-1:n})(y)V_2 - C(y,t)/x \quad (15)$$

Using the distribution we have:

$$\text{maximize} \quad F_{Y_{n:n}}(y)V_1 + [1 - F_{Y_{n:n}}(y)]F_{Y_{n-1:n-1}}(y)V_2 - C(y,t)/x \qquad (16)$$

$$\Pr(y \geq Y_{n:n})V_1 + \sum_{r=1}^{k-1} \Pr(Y_{n+1-r:n} > y \geq Y_{n-r:n})V_{r+1};$$

and it is zero otherwise. That can be written

$$[1 - F_{Y_{n:n}}]V_1 + \sum_{r=1}^{k-1}[1 - F_{Y_{1:n+1-r}}]F_{Y_{n-r:n}}V_{r+1}.$$

- Suppose that $t_i = T$. Suppose further that the performance of other players is a monotone decreasing function of their type. So we have $Y_j = b(X_j)$ and $b^{-1}(Y_j) = X_j$.

- This implies that we can use a simple change of variable to rewrite the payoff as a function of F_X. And we have:

  $$[1 - F_{X_{n:n}}(b^{-1}(y_i))]V_1 + \sum_{r=1}^{k-1} \Pr(Y_{n+1-r:n} > y_i \geq Y_{n-r:n})V_{r+1};$$

  Using Bayesian rule and one of the properties of *iid* order statistics to express conditional probability, we have

  $$[1 - F_{X_{n:n}}(b^{-1}(y_i))]V_1 + \sum_{r=1}^{k-1}(1 - F_{X_{1:n-r}})F_{X_{n-r:n}}V_r$$

B References