# Monopsony in Online Labor Markets

Arindrajit Dube, Jeff Jacobs, Suresh Naidu, and Siddarth Suri

July, 2018
NBER Summer Institute

# Searching for monopsony in labor markets

- Can there be substantial market power even on online labor markets?
  - "On-demand" and online work relatively new (Katz and Krueger 2016).
  - In particular market for crowdsourced "data services".

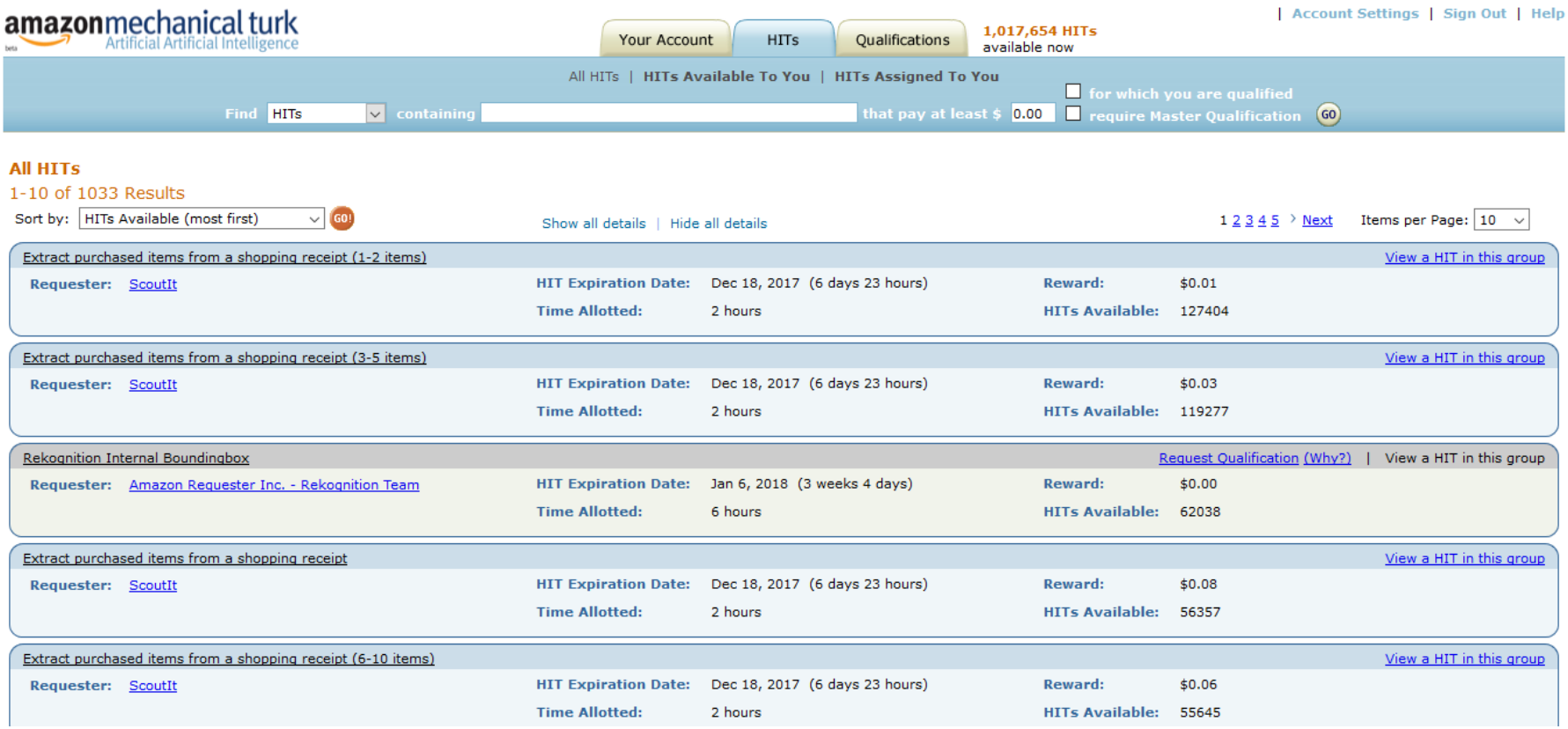
# Searching for monopsony in labor markets

- Can there be substantial market power even on online labor markets?
  - "On-demand" and online work relatively new (Katz and Krueger 2016).
  - In particular market for crowdsourced "data services".
- "Tough test" for monopsony relative to other contexts
  - Large number of jobs posted and seemingly low cost of search.
  - Little labor regulation.
  - Little work looking at market structure.

# Searching for monopsony in labor markets

- Can there be substantial market power even on online labor markets?
  - "On-demand" and online work relatively new (Katz and Krueger 2016).
  - In particular market for crowdsourced "data services".
- "Tough test" for monopsony relative to other contexts
  - Large number of jobs posted and seemingly low cost of search.
  - Little labor regulation.
  - Little work looking at market structure.
- Offers a laboratory for quantifying labor market power, especially experimentally.

# Looking for work on Amazon Mechanical Turk



# What we do

- Observational estimate
- Key idea: sensitivity of the duration of time-to-fill-task to posted wage (labor supply elasticity) is theoretical measure of labor market power.
  - Large scraped dataset from MTurk
  - Use text of requester and job descriptions to form high-dimensional set of covariates to isolate causal effect.
  - Use Double Machine Learning estimator proposed by Chernozhukov et al. (2016)
  - Find remarkably stable, low sensitivity suggesting high degree monopsony power, with little heterogeneity by reward.
- Experimentally validate degree of monopsony power on Mechanical Turk using archived data from previous experiments.

# What we do

- Observational estimate
- Key idea: sensitivity of the duration of time-to-fill-task to posted wage (labor supply elasticity) is theoretical measure of labor market power.
  - Large scraped dataset from MTurk
  - Use text of requester and job descriptions to form high-dimensional set of covariates to isolate causal effect.
  - Use Double Machine Learning estimator proposed by Chernozhukov et al. (2016)
  - Find remarkably stable, low sensitivity suggesting high degree monopsony power, with little heterogeneity by reward.
- Experimentally validate degree of monopsony power on Mechanical Turk using archived data from previous experiments.
  - Estimate sensitivity of job acceptance probability to posted wage

# What we do

- Observational estimate
- Key idea: sensitivity of the duration of time-to-fill-task to posted wage (labor supply elasticity) is theoretical measure of labor market power.
  - Large scraped dataset from MTurk
  - Use text of requester and job descriptions to form high-dimensional set of covariates to isolate causal effect.
  - Use Double Machine Learning estimator proposed by Chernozhukov et al. (2016)
  - Find remarkably stable, low sensitivity suggesting high degree monopsony power, with little heterogeneity by reward.
- Experimentally validate degree of monopsony power on Mechanical Turk using archived data from previous experiments.
  - Estimate sensitivity of job acceptance probability to posted wage
- Distinguish between "recruitment" and "retention" margin

# What we do

- Observational estimate
- Key idea: sensitivity of the duration of time-to-fill-task to posted wage (labor supply elasticity) is theoretical measure of labor market power.
  - Large scraped dataset from MTurk
  - Use text of requester and job descriptions to form high-dimensional set of covariates to isolate causal effect.
  - Use Double Machine Learning estimator proposed by Chernozhukov et al. (2016)
  - Find remarkably stable, low sensitivity suggesting high degree monopsony power, with little heterogeneity by reward.
- Experimentally validate degree of monopsony power on Mechanical Turk using archived data from previous experiments.
  - Estimate sensitivity of job acceptance probability to posted wage
- Distinguish between "recruitment" and "retention" margin
  - "recruitment": bringing someone into a task.
  - "retention": getting someone to do another task, once you've brought them in.
  - Experimental evidence finds both of these are surprisingly low.

# What we do

- Observational estimate
- Key idea: sensitivity of the duration of time-to-fill-task to posted wage (labor supply elasticity) is theoretical measure of labor market power.
  - Large scraped dataset from MTurk
  - Use text of requester and job descriptions to form high-dimensional set of covariates to isolate causal effect.
  - Use Double Machine Learning estimator proposed by Chernozhukov et al. (2016)
  - Find remarkably stable, low sensitivity suggesting high degree monopsony power, with little heterogeneity by reward.
- Experimentally validate degree of monopsony power on Mechanical Turk using archived data from previous experiments.
  - Estimate sensitivity of job acceptance probability to posted wage
- Distinguish between "recruitment" and "retention" margin
  - "recruitment": bringing someone into a task.
  - "retention": getting someone to do another task, once you've brought them in.
  - Experimental evidence finds both of these are surprisingly low.

# Some Theory

- Not first to notice MTurk may be non-competitive.

# Some Theory

- Not first to notice MTurk may be non-competitive.
- Kingsley, Gray and Suri (2015) show skewed distribution of job posting, suggesting labor market concentration.

# Some Theory

- Not first to notice MTurk may be non-competitive.
- Kingsley, Gray and Suri (2015) show skewed distribution of job posting, suggesting labor market concentration.
- But concentration not necessarily good measure of market power: could simply be productivity dispersion.

# Some Theory

- Not first to notice MTurk may be non-competitive.
- Kingsley, Gray and Suri (2015) show skewed distribution of job posting, suggesting labor market concentration.
- But concentration not necessarily good measure of market power: could simply be productivity dispersion.
- Instead, want to directly estimate the sensitivity of job acceptance to wage/reward.

# Some Theory

- Not first to notice MTurk may be non-competitive.
- Kingsley, Gray and Suri (2015) show skewed distribution of job posting, suggesting labor market concentration.
- But concentration not necessarily good measure of market power: could simply be productivity dispersion.
- Instead, want to directly estimate the sensitivity of job acceptance to wage/reward.
  - Competitive market should be very responsive.
  - Monopsonistic market should be pretty non-responsive.

# Scraped Data

- Scrape MTurk data (alas no longer possible).
- Panos Ipseiros scraped between Jan 2014-Feb 2016.
- We scraped from May 2016 to August 2017.
- Both theoretically capture a near-census of posted HITs.
- Observe: time posted, time removed, reward, requester, and short text describing job.
- Idea: Duration = time removed - time posted is another way to estimate recruitment elasticity.
- **Assumption:** HITs observed at constant rate $\lambda$.
- High paying jobs should disappear quickly.
  - But plenty of other determinants of duration (e.g. requester cancellation, other task characteristics).

# Densities of Log Durations

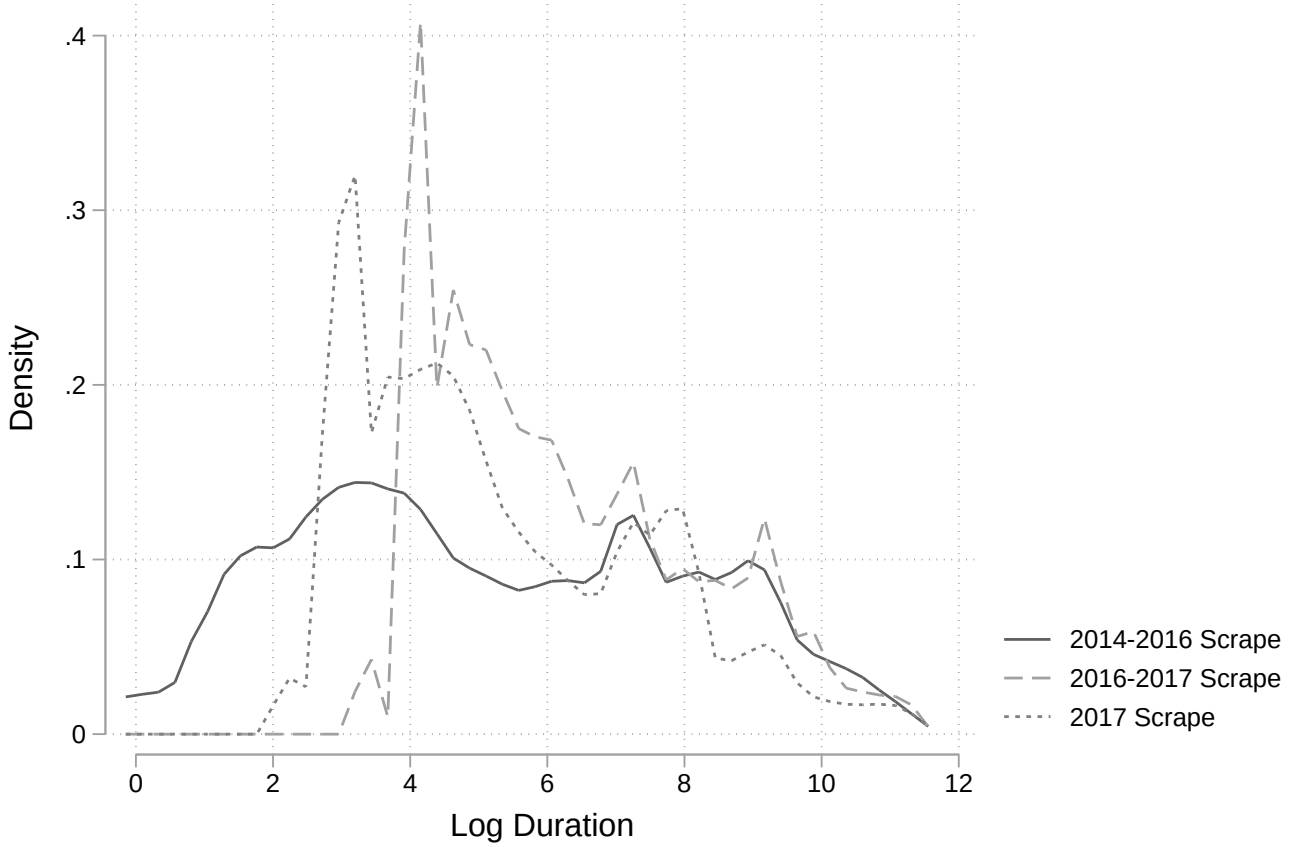



Figure: Kernel density plots of log duration for the 3 different samples used in the analysis.

# Regression Framework

$$\log(duration_h) = -\eta log(reward_h) + \nu_h + \epsilon_h \quad (1)$$

Suppose $E[\epsilon|\nu] = 0$, but if $\nu_h$ unobserved and not equal to 0 then resulting $\eta$ estimate is biased.

# Identification

- Approach 1: Fixed Effects
- FE strategy: control for requester, time allotted + batch size deciles, and time posted FE.
- But how do we know we have controlled for as much as we possibly can?
- Machine learning to the rescue!

# Double-ML estimator

- General method for leveraging "big data" to approach causality.
- Caveat: assumes only selection on observables, but pushes it as far as possible.
- Idea: use machine learning (we use Random Forests) to control flexibly for observable determinants ($X^{reward}, X^{duration}$) of rewards and durations.
- Get functions for predicted rewards $E[log(reward)|X^{reward}]$ and durations $E[log(duration)|X^{duration}]$ and validate using standard out-of-sample methods.
- Robinson (1988) Intuition: coefficient $-\eta$ is recovered from regression of $\widehat{\zeta} = log(duration) - E[log(duration)|X^{duration}]$ on $\widehat{\mu} = log(reward) - E[log(reward)|X^{reward}]$.
- Check: Adding any other covariates has little effect on point estimates.

# Features Used

- Create a variety of requester and time level characteristics (mean reward, etc).
- **N-grams**: Sliding windows of length 1–3 over the text, where the feature value is the (unweighted) frequency of the gram within the title or description.
- **Topic Distributions**: LDA with topics $K \in \{5, 10, 15, 20\}$ are run on all descriptions, use topic proportions as features..
- **Doc2Vec embeddings**: 100-dimensional embedding vector corresponding to each description was used as a feature for that description.

A split of 40% training, 10% validation, and 50% test was used, with 5-fold cross-validation for hyperparameter (number of trees/ features considered in each split) tuning, resulting in 2-fold sample-splitting for the final $\check{\theta}$ value.

# Prediction Model

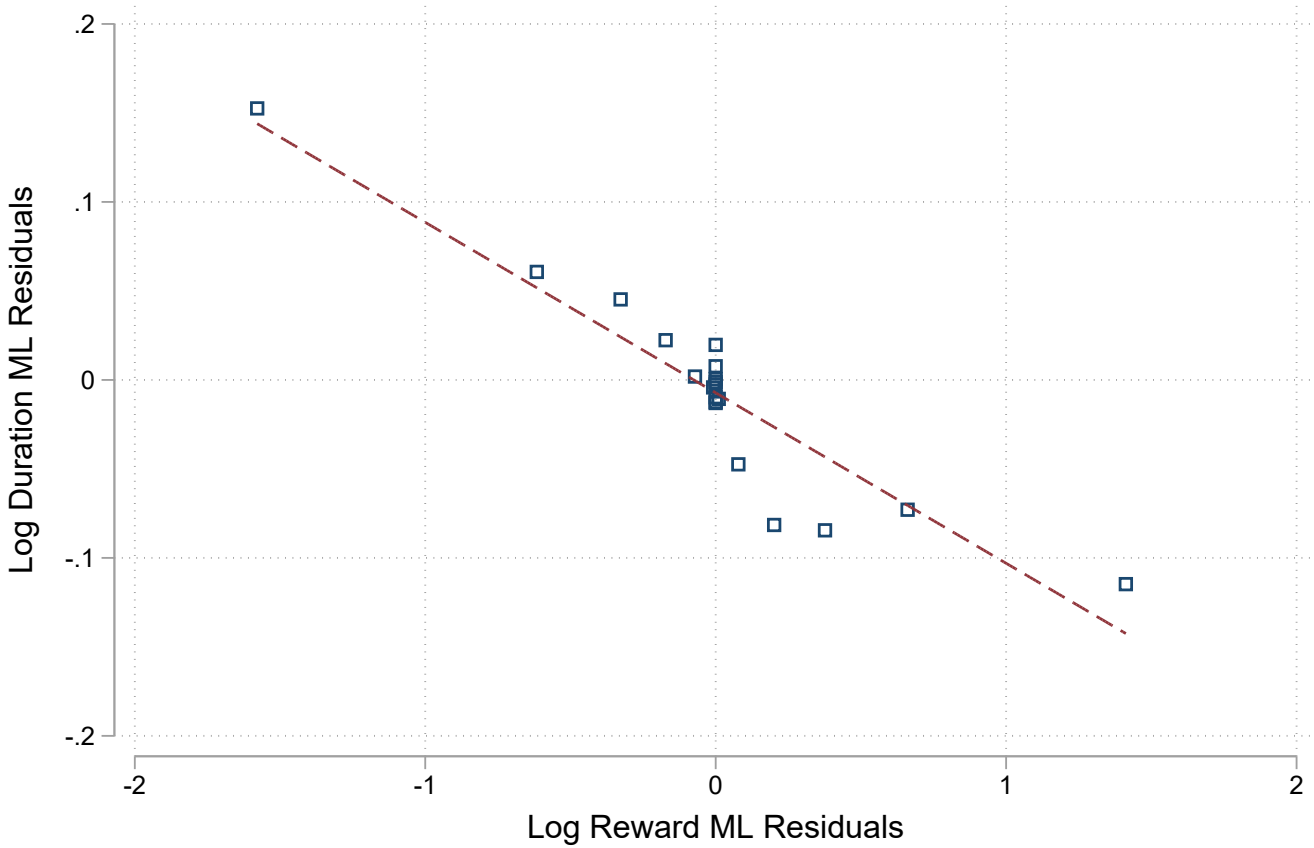
Out of sample $R^2$ values:

| | Jan 2014 - Feb 2016 | | May 2016 - May 2017 | | May 2017 - Aug 2017 | |
|---|---|---|---|---|---|---|
| | $A \to B$ | $B \to$ A | $A \to B$ | $B \to A$ | $A \to B$ | $B \to A$ |
| Reward | 0.7716 | 0.7764 | 0.8951 | 0.8949 | 0.8982 | 0.8984 |
| Duration | 0.8968 | 0.8980 | 0.4379 | 0.4404 | 0.5085 | 0.5035 |

Most predictive features:

| Rank | Feature Name | Gini |
|---|---|---|
| 0 | Requester Mean Reward | 0.408642 |
| 1 | Log Requester Mean Reward | 0.392017 |
| 2 | Time Allotted | 0.040536 |
| 3 | Requester Batch Count | 0.016639 |
| 4 | Title Length | 0.014999 |

# Residuals Binned Scatterplot



# Results

Table: Duration Elasticities from Observational MTurk Data

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) |
|---|---|---|---|---|---|---|---|
| Log Reward | 0.186 | -0.0600 | | | | | |
| | (0.0947) | (0.0585) | | | | | |
| Log Reward-ML res. | | | -0.0958 | -0.0787 | -0.198 | -0.181 | -0.0299 |
| | | | (0.00558) | (0.00651) | (0.0281) | (0.0161) | (0.00402) |
| N | 644873 | 629756 | 644873 | 629756 | 93775 | 292746 | 258352 |
| Clusters | 41167 | 26050 | 41167 | 26050 | 6962 | 18340 | 24923 |
| Type | OLS | FE | ML | ML-FE | ML | ML | ML |
| Data | Pooled | Pooled | Pooled | Pooled | 2017 | 2016-2017 | 2014-2016 |

# Experimental measures of labor market power

- Two different margins for requester labor supply elasticity
- "Recruitment": responsiveness of worker to initial posted wage.
- "Retention": responsiveness of worker to bonus wage offered after worker accepts job.

# Experimental Retention Elasticities

- Obtain replication data from two experiments (Horton and Zeckhauser 2011, Dube, Manning, Naidu 2016).
  - Design: post initial HIT at fixed reward.
  - Collect demographics and have do initial task (e.g. tag fugitive slave column in 1850 census).
  - Then ask if they would like to do more for additional randomized wage.

$$Pr(Accept_i) = \beta reward_i + \epsilon_i \tag{2}$$

- Labor supply elasticity facing the firm is $\eta = \widehat{\beta} \times \frac{E[reward]}{E[Accept]}$

# Results

| | Horton et al 2011: Probability of Accepting Offer | | | |
|---|---|---|---|---|
| Reward | 0.127 | 0.140 | 0.0861 | 0.0973 |
| | (0.0219) | (0.0241) | (0.0292) | (0.0333) |
| N | 328 | 307 | 125 | 107 |
| $\eta$ | 0.234 | 0.241 | 0.192 | 0.202 |
| SE | 0.0334 | 0.0364 | 0.0594 | 0.0664 |
| Avg. Reward | 11.60 | 11.63 | 11.37 | 11.50 |
| Sophisticated | No | No | Yes | Yes |
| Controls | No | Yes | No | Yes |
| | Dube et al. 2017: Probability of Accepting Offer | | | |
| Reward | 0.0267 | 0.0486 | 0.0764 | 0.0782 |
| | (0.0171) | (0.0202) | (0.0348) | (0.0329) |
| Controls | No | Yes | No | Yes |
| N | 5184 | 5017 | 1702 | 1618 |
| $\eta$ | 0.052 | 0.077 | 0.118 | 0.114 |
| SE | 0.0333 | 0.0322 | 0.0534 | 0.0479 |
| Avg. Reward | 9 | 9 | 9 | 9 |
| Sophisticated | No | No | Yes | Yes |

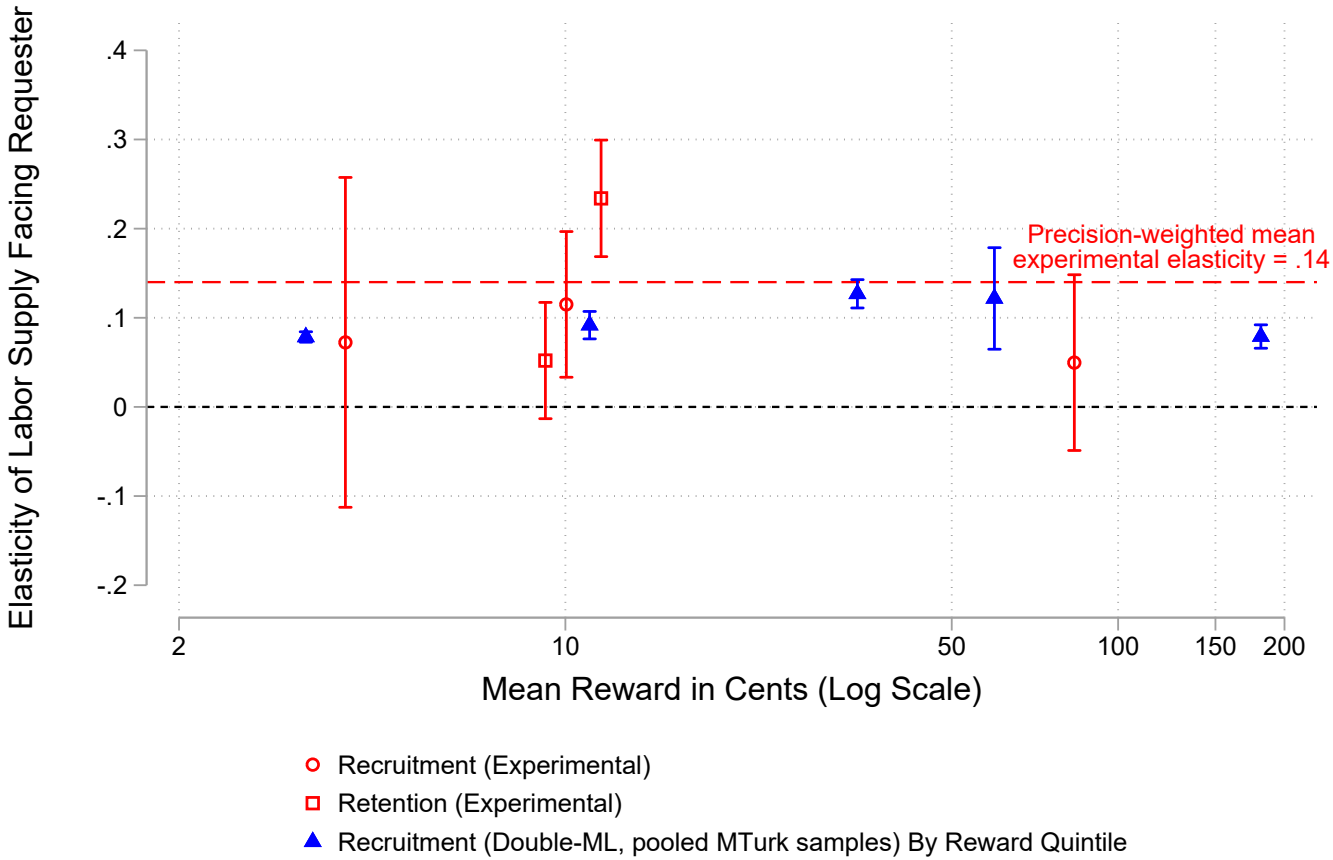
# Experimental Recruitment Elasticities

- Obtain replication data from three experiments (Hsieh 2016, Ho et al. 2015, Yin et al. 2018).
  - Design: post initial HIT at fixed reward.
  - Collect demographics and get Mturk worker id.
  - Randomize HIT postings to be seen by subset of worker ids ("honeypot" design).
  - Again estimate probability of accepting as function of reward and compute $\eta$.

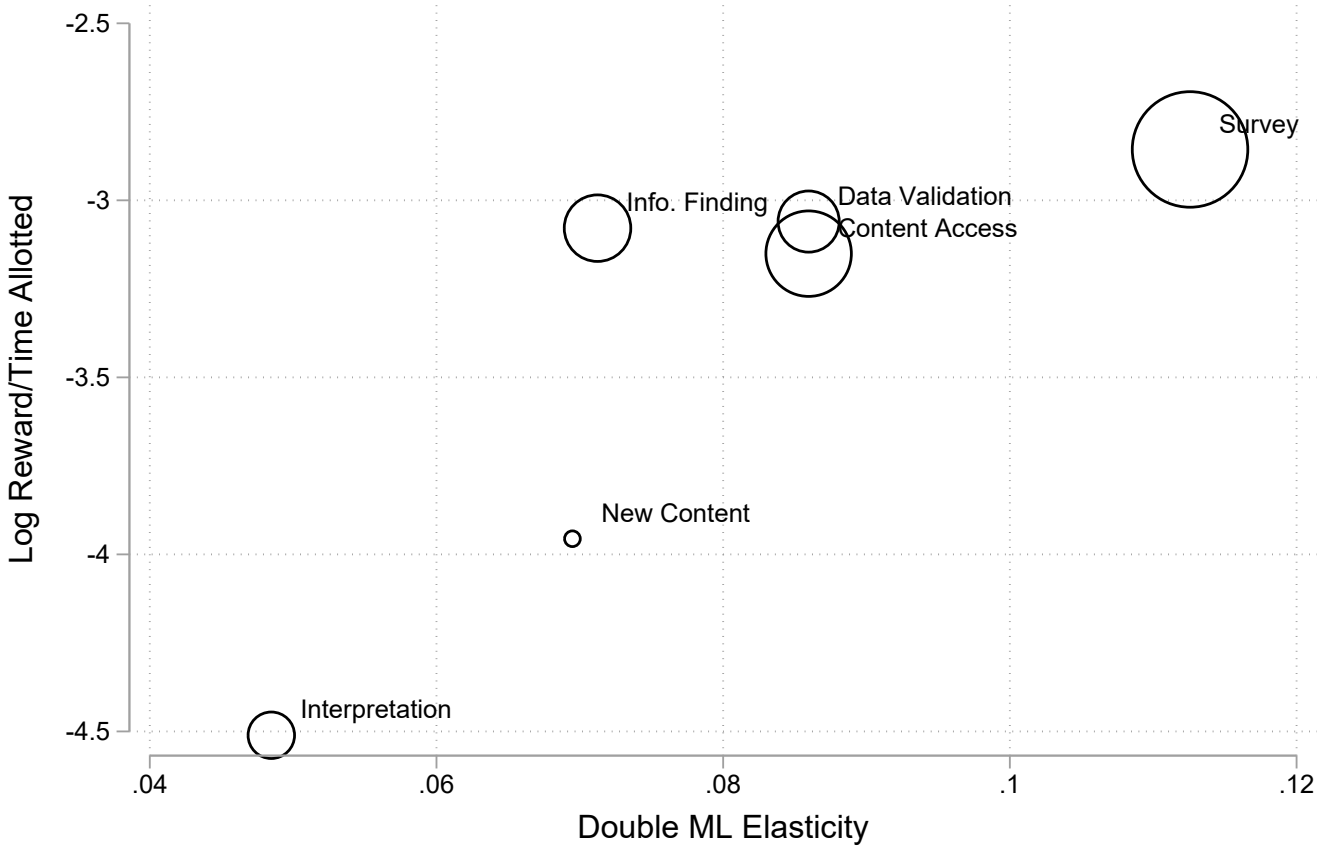
# Results

| | Recruitment Elasticities From Three Experiments | | | |
|---|---|---|---|---|
| | (1) | (2) | (3) | (4) |
| Reward | 0.00186 | 0.0451 | 0.0287 | 0.00744 |
| | (0.00188) | (0.0587) | (0.0104) | (0.00385) |
| N | 600 | 1800 | 338 | 2738 |
| $\eta$ | 0.0497 | 0.0724 | 0.115 | 0.0610 |
| SE | 0.0503 | 0.0944 | 0.0417 | 0.0290 |
| Avg. Reward | 83.33 | 4 | 10.04 | 22.13 |
| Experiment | Spot Diff. | Classify Reviews | Brainstorming | Pooled |

# Remarkably Stable Estimates - Experimental and Observational Approaches



# By Task Type: Employers Are Using Their Monopsony Power



# Conclusion

- Labor supply elasticities facing requesters on MTurk quite low.
- Implies optimizing employers are paying workers less than 12% of productivity.
- Implies training sample sizes and statistical power lower than would be obtained under competitive market?
- Why is there this monopsony power? Job differentiation likely candidate.
- Solutions: Rating systems (e.g. Turkopticon), alternative platforms (Dynamo).
- Solutions: MTurk allowing more wage discrimination, but crowdsourcing market rewards pro-requester platforms.

# Summary Statistics

| | 2014-2016 Scrape | | 2016-2017 Scrape | | 2017 Scrape | |
|---|---|---|---|---|---|---|
| | Mean | Std Dev | Mean | Std Dev | Mean | Std Dev |
| Duration (Minutes) | 3370.360 | 9414.101 | 3519.257 | 9721.523 | 2293.174 | 8375.199 |
| Reward (Cents) | 38.014 | 63.741 | 70.397 | 92.420 | 61.774 | 87.358 |
| Log Reward ML Prediction | 2.639 | 1.229 | 3.431 | 1.416 | 3.286 | 1.362 |
| Log Duration ML Prediction | 5.210 | 2.642 | 6.223 | 1.414 | 5.301 | 1.589 |
| Log Duration ML Residuals | -0.004 | 0.892 | -0.013 | 1.432 | 0.003 | 1.466 |
| Log Reward ML Residuals | -0.001 | 0.679 | -0.003 | 0.483 | -0.001 | 0.459 |
| Time Allotted (Minutes) | 77.793 | 204.495 | 595.510 | 2916.676 | 434.435 | 2102.791 |
| Max No. of HITs in Batch | 83.413 | 1303.061 | 59.867 | 1627.825 | 53.539 | 931.335 |
| Observations | 258352 | | 292746 | | 93775 | |